A Numerical Study of Frictional Contact

Sayantan DATTA,
School of Computer Science,
McGill University, Montreal

December 17, 2018

A thesis submitted to McGill University in partial fulfillment of the requirements for the degree of Master of Science

under the supervision of
Prof. Paul G. Kry,
Prof. Derek Nowrouzezahrai.

MCGILL UNIVERSITY, MONTREAL

Abstract

School of Computer Science

Master of Science

A Numerical Study of Frictional Contact

by Sayantan DATTA

Friction is a complex phenomenon resulting from elastic and plastic deformations coupled with molecular interaction along the contact boundary. When two surfaces touch, their roughness, and normal force determines the actual area under contact, governing the process of deformation and molecular interaction. A typical macroscopic interaction may involve millions of microscopic contacts and the aggregate of these forces give rise to the phenomenon of friction. In this thesis, our goal is to simulate the phenomenon of friction assuming unlubricated contact and elastic deformation at the contact asperities. We collect data by varying many parameters that affect friction between two surfaces and build a function approximator exploiting the correlation in data. Such an approximator is a computationally inexpensive, versatile and more accurate substitute for friction coefficient tables currently in use with various physically based simulators.

Traduction en français

Une Étude Numérique du Contact Frictionnel

par Sayantan DATTA

La friction est un phénomène complexe résultant de déformations élastiques et plastiques couplées à une interaction moléculaire le long de la frontière de contact. Lorsque deux surfaces se touchent, leur rugosité et leur force normale déterminent la surface réelle en contact qui régit le processus de déformation et d'interaction moléculaire. Une interaction macroscopique typique peut impliquer des millions de contacts microscopiques, c'est l'agrégation de ces forces qui provoque le phénomène de friction. Dans cette thèse, notre objectif est de simuler le phénomène de frottement en supposant un contact non lubrifié ainsi qu'une déformation élastique aux aspérités de contact. Nous recueillons des données en faisant varier de nombreux paramètres qui affectent le frottement entre deux surfaces et construisons une approximation exploitant la corrélation des données. Un tel approximation est peu coûteux, polyvalent et plus précis que les tables de coefficients de frottement actuellement utilisées dans divers simulateurs physiques.

Acknowledgements

I wish to thank everyone who supported me during this study. I would like to express my deep gratitude to Professor Paul G. Kry and Professor Derek Nowrouzezahrai, my research supervisors, for their invaluable advice during the course of this study. Their courses provided me with a solid foundation in computer graphics and also motivated me to take up graphics research as my career. Paul helped me with his useful advice about theory, methods, and literature in elastic theory and contact mechanics. Derek's input was crucial for the development of the project and his suggestions regarding documentation and presentation were invaluable. I would also like to extend my thanks to Professor Kenny Erleben and Professor Sheldon Andrews for their valuable feedback during the development of the project. Finally, I thank my family for supporting me during my stay far from home and their constant encouragement.

Declaration of Authorship

I, Sayantan DATTA, declare that this thesis titled, "A Numerical Study of Frictional Contact" and the work presented in it are my own. I confirm being the sole contributor for all chapters from Chapter 1 to Chapter 5.

Contents

Abstract **iii**

Traduction en français iv

Acknowledgements **v**

Declaration of Authorship **vii**

1 Introduction **1**

1.1 Overview of Friction Force 1

1.2 Historical Remarks 3

1.3 Motivation 3

1.4 Current Issues 4

1.5 Our Goals and Expectations 5

2 Literature Survey **7**

2.1 Couloumb's Law 7

2.2 Origin of Friction 8

2.3 Relationship to Surface Roughness 9

2.4 Categories of Friction 9

2.5 Modeling 10

2.5.1 Hertzian Contact 12

Two Sphere Contact 13

2.5.2 Local Friction 14

2.5.3 Friction in Elastomer 15

2.6 Summary of Tribology Literature Survey 15

2.7 Friction in Computer Graphics Literature 16

2.8 Rigid Body Models 17

2.8.1 Resolving Contact Acceleration 18

2.8.2 Resolving Contact Impulses 21

Friction Pyramid Approximation 24

2.8.3 Approximate Friction Cone 25

2.8.4 Data Driven Techniques 29

2.9 Soft Body Models . . . 29
2.9.1 Soft Body Contact . . . 30
2.9.2 True Friction Cone . . . 32
2.9.3 Data Driven Techniques . . . 33

3 Implementation **35**
3.1 Deformation Map and Deformation Gradient . . . 35
3.2 Energy Density and Internal Force . . . 36
3.3 Neohookean Material . . . 37
3.4 Discretization . . . 39
3.5 Generating Surfaces . . . 41
3.6 Rayleigh Damping . . . 43
3.7 Computing Stiffness Matrix (K) . . . 45
3.8 Computing Critical Damping . . . 45
3.9 Soft Body Collision . . . 48
3.9.1 Spatial Hashing . . . 48
3.9.2 Penalty Forces in 2D . . . 48
3.9.3 Penalty Forces in 3D . . . 51
3.10 Time Stepping . . . 53
3.11 Summary . . . 54

4 Results **55**
4.1 Friction in 2D . . . 55
4.1.1 Methodology . . . 56
4.1.2 Results . . . 56
4.2 Friction in 3D . . . 61
4.2.1 Methodology - Dynamic Friction . . . 62
Results . . . 62
4.2.2 Methodology - Static Friction . . . 63
Results . . . 63
4.3 Real World Measurements . . . 64
4.3.1 Methodology . . . 64
4.3.2 Results . . . 68
4.4 Conclusion . . . 68

5 Conclusion **69**
5.1 Future Work . . . 69

Bibliography **71**

List of Figures

1.1 **The image on the left demonstrates the effect of friction as an operating mechanism for brake calipers. The image on the right shows wearing of a gear due to friction.** . . . 1
1.2 **Frictional contact at different scales.** . . . 2
1.3 Diagrams dating back to 1500 A.D. showing attempts to analyze the friction on a cylinder supported in a half-bearing. Adopted from Codex Arundel, British Library, London (Arundel folio 41r c. 1500-05). . . . 3

2.1 **Contact between two elastic spheres.** . . . 13
2.2 **Local contact model. The local friction force r is in the XY plane opposing the relative motion of the two bodies. The local normal pressure p_n acts along axis Z.** . . . 14
2.3 **Rigid body contact without friction** . . . 18
2.4 **Friction pyramid.** . . . 21
2.5 **Approximate linearized friction cone. The convex hull marked with blue is given by the condition $e^T\beta_i = \mu f_{Ni}$. The direction of the net tangential force $D_i\beta_i$ is shown using dashed arrows.** . . . 26
2.6 **Discretization of soft a body. The mass of the soft body is distributed over lumped masses m_i and internal forces f_{ij} hold the masses together.** 30
2.7 **Soft body contact for two arbitrary soft bodies with different parameterizations. A position p_x on soft body X (in red) is a function of parameters s_x and q_x while position $p_y$on soft body Y is a parameterized by s_y and q_y** . . . 30

3.1 **Deformation map denoted by ϕ.** . . . 36
3.2 **Deformation of discrete elements.** . . . 39
3.3 **Noise samples (top) and their associated frequency spectrum (bottom).See that Gabor noise shows anisotropic properties both in signal space and frequency space.** . . . 42
3.4 **Delaunay triangulation of the boundary markers to generate a hollow mesh.** . . . 42

3.5 **Cross section of an adaptively discretized volume. Size of the tetrahedrons reduces as we go from flat base to noisy top.** 43
3.6 **System response under different level of damping.** 46
3.7 **Modeling edge-particle collision. Position P is the projection of AC on AB. We do not allow the particle C to come any closer than distance H from the edge AB.** 48
3.8 **Modeling tetrahedron-tetrahedron collision. The repulsive forces act between the sampled points on the tetrahedron.** 52
3.9 **Modeling repulsion under close proximity. H_1 and H_2 represents the distance of the centroid from corresponding vertex.** 53

4.1 **An output frame from our 2D simulator.** 55
4.2a **Variation in the coefficient of static friction with Young's modulus. The shaded tube indicates the standard deviation in measurement at the point of observation.** 56
4.2b **Variation in the coefficient of static friction with Poisson ratio with associated error bound.** 57
4.2c **Variation in the coefficient of static friction with Rayleigh damping with associated error bound.** 57
4.2d **Variation in the coefficient of static friction with mass density with associated error bound.** 58
4.3 **Variation in the coefficient of static friction with elastic parameters.** . 60
4.4 **An output frame from our 3D simulator.** 61
4.5 **The plot of tangential forces over time. The normal force is held constant at 15 Newton with a relative velocity of 0.2 m/s.** 62
4.6 **The plot of tangential feedback forces over time. The normal force is held constant at 15 Newton while tangential force is increased over time.** 64
4.7a **Variation in the coefficient of static friction with Young's modulus (E) of the two materials in contact.** 66
4.7b **Variation in the coefficient of static friction with Poisson ratio (ν) of the two materials in contact.** 67
4.7c **Variation in the coefficient of static friction with mass density (ρ) of the two materials in contact.** 67

List of Tables

4.1 **Table of static friction coefficient (μ_{mean}) and the reliability of measurement (μ_{std}) in the 2D case for varying soft body parameters.** 59
4.2 **The coefficient of static friction obtained by our simulator for metal like material.** 63
4.3 **Table of static friction coefficient (μ_{static}) obtained from real world measurements. The data in the table is compiled from various sources available online.** 65

List of Symbols

We use uppercase letters to denote matrices, lowercase letters for scalars and, bold letters and symbols to denote column vectors. All symbols are explained in place.

Chapter 1

Introduction

Friction is a ubiquitous phenomenon and we all have experienced the phenomenon or lack thereof in our daily lives. Friction is the reason why we can hold a coffee mug in our hand or write on a blackboard. Without friction, it would be almost impossible to carry out regular tasks in our daily lives. In fact, we are so used to friction, that we hardly acknowledge its presence and only realize its importance when it is missing e.g. walking on an icy sidewalk during winter or stepping on a banana peel. At the same time, friction also makes doing other things harder. Friction is the reason why it is so difficult to slide furniture on a floor or ride a bicycle on snow.

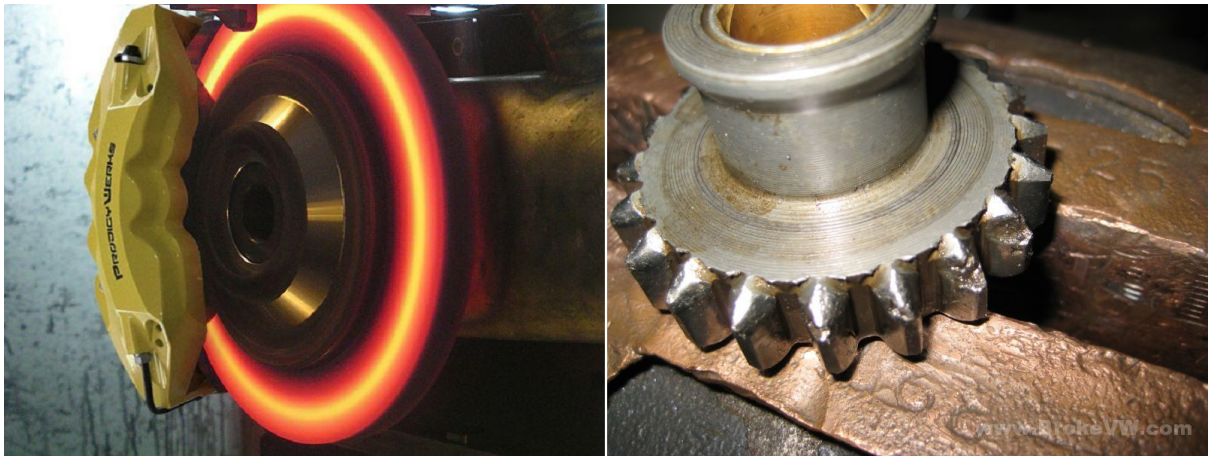

FIGURE 1.1: **The image on the left demonstrates the effect of friction as an operating mechanism for brake calipers. The image on the right shows wearing of a gear due to friction.**

1.1 Overview of Friction Force

Friction is a contact force – meaning it appears whenever two surfaces are in contact. The force is non-conservative and opposes the motion between two surfaces that are in contact. Work done due to non-conservative forces depend on the trajectory of motion.

Such forces are responsible for energy losses in a system. As seen in Figure 1.1, friction is both useful and adversarial. It is the primary reason for wear in machines and causes energy dissipation and efficiency losses. On the other hand, it is used as an operational mechanism in many applications such as brakes in vehicles, treads on a tire, contact forces in robotic limbs and many more.

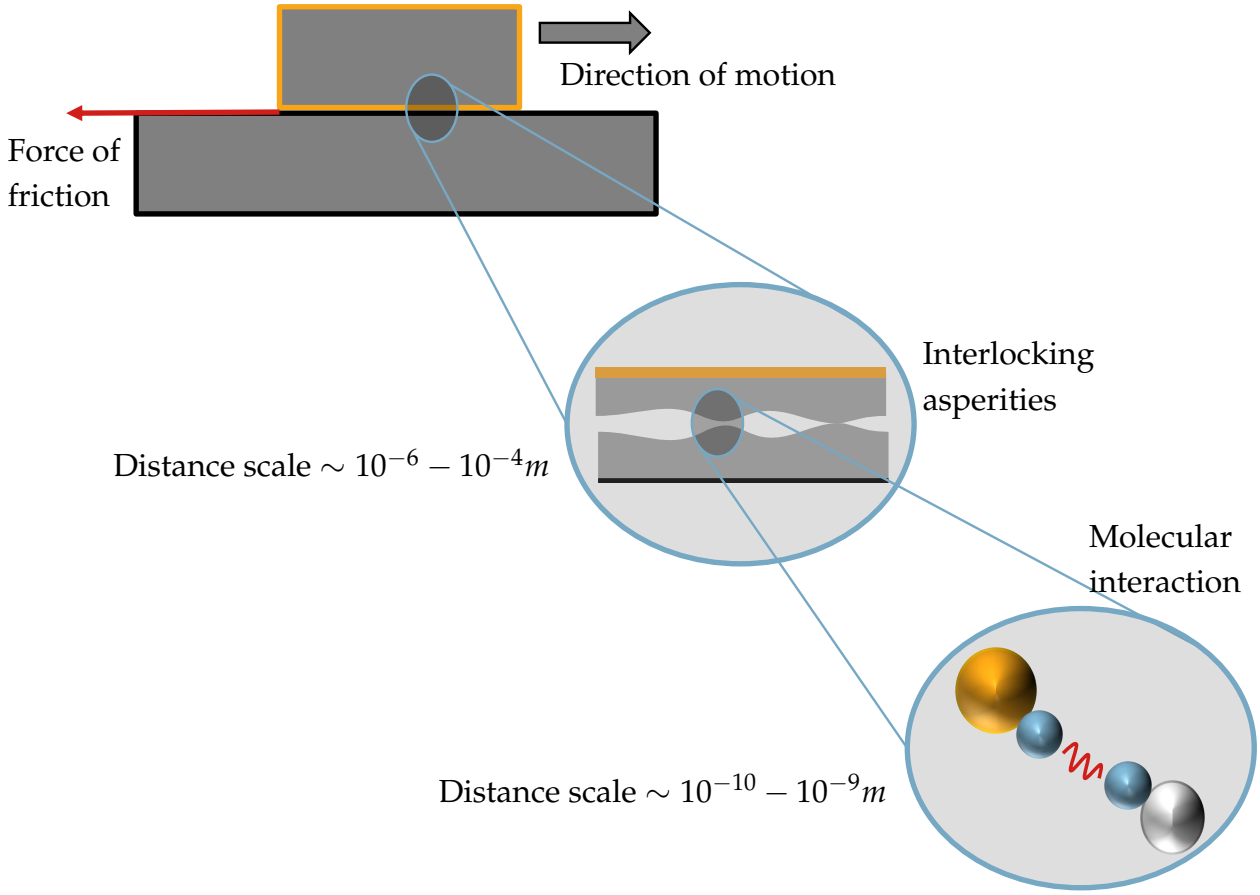

FIGURE 1.2: **Frictional contact at different scales.**

Let us look at the operating mechanism of friction. As shown in Figure 1.2, friction force always act in the direction opposing the relative motion. When looked closely under a microscope, the surface of a material rarely looks smooth. In fact, one can see peaks and valley like structure all over the surface. These peaks are commonly referred to as *asperities*. One cause of friction force is the entanglement of asperities between the surfaces. Horizontal forces required to deform the asperities or break them constitutes a part of friction forces. If we further zoom at the contact between two asperities, molecular interactions also come into play. In many cases, molecular interactions constitute a significant portion of the friction forces. Thus, the friction force is a complex combination of deformation of asperities and molecular interaction. We will investigate these factors in more details in the next chapter.

1.2 Historical Remarks

First systematic study of friction dates back to more than 500 years. Leonardo da Vinci was among the first to study friction from an engineering standpoint. As shown in Figure 1.3, he developed rolling elements aimed at reducing friction and wear and studied several tribological subtopics such as friction, wear, bearing materials, lubrication systems, gears, screw-jacks, and rolling elements. In fact, da Vinci discovered and documented the Laws of Friction nearly 150 years before Amontons' Laws of Friction were introduced.

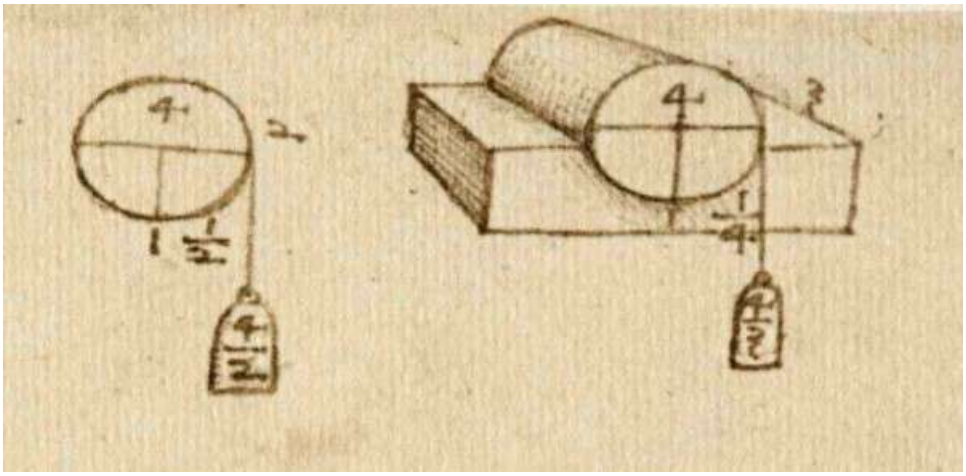

FIGURE 1.3: Diagrams dating back to 1500 A.D. showing attempts to analyze the friction on a cylinder supported in a half-bearing. Adopted from Codex Arundel, British Library, London (Arundel folio 41r c. 1500-05).

Other pioneers in the field of tribology are Guillaume Amontons (1663-1705), John Theophilus Desaguliers (1683-1744), Leonard Euler (1707-1783), and Charles-Augustin Coulomb (1736-1806). These pioneers brought tribology to a standard, and their laws still apply to many engineering problems today.

1.3 Motivation

Understanding friction reliably and predictably can simplify many issues relevant in computational fabrication, robotics, and animation. With the commercialization of 3D printing, it is now possible to fabricate parts with unprecedented precision. However, we have to rely on computational models to accurately predict the behavior of parts before physically printing them. Modeling the objects and simulating them in a virtual environment can significantly reduce design errors and minimize the cost of production. There have been significant advances along this direction [7, 10] and is an active field of research. Similarly, in robotics and reinforcement learning, it is more

cost effective to train a robot in virtual environments than the actual environment, especially when the environment is hazardous or inaccessible. Friction is an important part of any physically based system, and accurate modeling of friction is of paramount importance. In computer animation, many realistic visual effects like cloth, and volumetric soft bodies require accurate modeling of friction to produce finer details like wrinkles on cloth or skin.

1.4 Current Issues

We use an approximation of Coulomb's law to model friction. Coulomb's law originated during the late 18^{th} century and is astonishing that we continue to use this model through 21^{st} century. The main reason for the popularity of Coulomb's law is its simplicity and its ability to provide a sufficiently good approximation of the friction phenomenon in many cases. However, even with all the simplicity, exact implementation of Coulomb's law is difficult owing to its non-linear nature. Non-linearity violates many solvability guarantees offered by its linearized approximation. Most of the popular physics simulator like Bullet, PhysX and MuJoCo use a linearized polyhedral approximation of Coulomb's friction. We discuss the linearized polyhedral approximation in more details in the upcoming chapter.

When using Coulomb's law of friction for rigid bodies, we do not consider the geometry of the surface of the two materials in contact. The current approach relies heavily on friction coefficients obtained from experimental observations, and it is not always possible to run experiments on every pair of surfaces we intend to simulate. Since friction depends on the geometry of the surfaces in contact, known coefficients may not be accurate due to differences in the surface geometries [2].

In most models, we ignore internal damping and the effects of plastic deformation along the contact surface. Plastic deformation is the condition when a material is pushed beyond its elastic limit, and the material is no longer able to regain its original shape. When pushed further, the material may fracture causing wear in the material. Due to plastic deformation and wear, the surface geometry may change, affecting the friction coefficient. Also, the stress-strain curve is no longer linear when operating in the plastic region which can also affect friction properties of a material.

Lastly, no distinction is made between the static and kinetic coefficient of friction even though in practice they are very different. Usually, the simulators use the same coefficient of friction in both static and dynamic case.

1.5 Our Goals and Expectations

We aim to study the nature of friction and build a computational framework for tackling issues relevant in physically based simulators. Our primary goal is to build an approximate function that outputs the coefficient of friction based on various elastic parameters affecting the phenomenon. Generating an accurate coefficient of friction instead of force has the advantage that it can be easily integrated into existing frameworks. Since our approach requires a lot of data, we build a simulator which can capture the effects of changing various parameters on friction. The data can also be obtained from real world measurements but using a simulator has an added advantage that we can build any surface geometry and material which may not exist in the real world or are costly or difficult to fabricate. Using a simulator is also cheaper than buying robots for measurements.

Our objective is to capture dry friction, meaning without any lubrication at the interface, for both static and dynamic case using a simulator. We do not simulate plastic deformations and limit ourselves to the elastic case. Our work involves building a simulator which can capture the effects of changing various parameters on friction. We start with a thorough background in both tribology and computer graphics to get the idea of prevalent techniques and understand the factors affecting frictional contact in detail.

Chapter 2

Literature Survey

In this chapter, we perform a survey of the tribology and computer graphics literature. Our aim is two-fold; first, we wish to understand static and dynamic friction and differentiate the models used to describe the two categories; second, we identify the parameters which affect the two categories friction. Our tribology literature survey is described in Section 2.1 and Section 2.6.

In the second part, starting from Section 2.7, we discuss various models used to simulate friction in computer graphics. We discuss the problems associated with modeling friction in rigid body mechanics and their prevalent solutions. We end the chapter with a discussion on soft bodies and an approach for modeling friction in soft bodies.

2.1 Couloumb's Law

Coulomb's law of friction was postulated by Charles-Augustin Coulomb during the 18^{th} century and still apply to many engineering problems today. The findings of Coulomb and his predecessors are summarized in the following three laws:

- Law 1: The force of friction is directly proportional to the applied load (Amontons 1st Law).
- Law 2: The force of friction is independent of the apparent area of contacts (Amontons 2nd Law).
- Law 3: Kinetic friction is independent of the sliding velocity (Coulomb's Law).

The three laws only apply to dry friction. It was well known from ancient time that lubrication changes tribological properties. All three laws can be condensed in the so-called Coulomb's law (or Amonton-Coulomb law), which relates friction force F_t and the load acting normally across the surfaces F_N by

$$F_t = \mu F_N, \tag{2.1}$$

where the scalar μ is the coefficient of friction which depends on the properties of the material in contact and the geometry of the interface. Notice that the law does not differentiate between static and kinetic friction, but it has been empirically found that $\mu_s \geq \mu_k$ where μ_s is the coefficient of static friction whereas μ_k is the coefficient of kinetic friction.

2.2 Origin of Friction

Tribology is the general field of science which concentrates on contact mechanics of moving interfaces. Tribology encompasses friction, lubrication, wear, and adhesion. An important aspect of tribology is the dissipation of energy through the breaking of molecular bonds, plastic deformation and wear. In this thesis, we focus primarily on friction. Whenever two surfaces are in contact, friction acts as a force that resists tangential motion between the objects in contact. The study of friction at the atomic level involves two entangled sub-issues. First is the roughness of surfaces in contact and second is the interfacial mechanism for conversion of mechanical energy to other forms [22]. In absence of plastic deformation or wear, friction largely originates at an atomic scale. However, when loading is large, plastic deformation and wear forms the important component of friction force while the other component being the atomic forces.

At the atomic scale, the friction force is dependent on both contact speed v_c and real area of contact A_c, with the "force law" expressed as

$$F_t = \eta v_c A_c; \eta = \frac{\rho}{\tau}. \tag{2.2}$$

Here, ρ is the aerial mass density and τ is a characteristic "slip time", which corresponds to the time for the moving object's speed to fall to $1/e$ of its original value, assuming it is stopped by frictional forces alone. Here, "e" is the Euler's number.

At microscopic level or at the atomic scale, the friction force is dependent on the true area of contact and independent of the normal force applied. However, at the macroscopic level, friction is largely independent of the area of contact but depends on normal force. The difference in the form between the microscopic and macroscopic friction laws is rationalized by assuming the true real area of contact between macroscopic objects is likely to be proportional to the loading force. Also notice that microscopic friction depends on contact speed v_c while Equation 2.1 is independent of speed. It is assumed that the instantaneous sliding velocity at the microscopic contacts

is much larger than the sliding velocity of the material itself i.e. $v_c \gg v$. Hence, the aggregate force due to friction is the property of the junction itself rather than the sliding velocity.

2.3 Relationship to Surface Roughness

Friction is a complex phenomenon. When two surfaces come into contact, only certain parts of the surface will carry the applied load. The sum of these discrete contact areas forms the real contact area. After Section 2.2, it is somewhat counterintuitive that flat surfaces tend to have lower friction coefficients compared to rough surfaces. We know friction depends on the real area under contact which is supposedly much greater for smooth surfaces compared to rough surfaces. However, the friction force due to an area under contact only considers interfacial friction i.e. friction due to molecular bonds. Another large component of friction is due to plastic deformation of asperities under contact. When two surfaces are rubbed together, their asperities interlock [20]. Depending upon the normal load, the asperities either slide over each other or deform. When the normal load is low, the tendency of the surface is to slide over each other without causing much deformation. Thus, friction force under low normal load is mostly due to molecular adhesion of contacts at the asperities of the two surfaces. As the loading increases, the real area of contact increases due to deformation. At the same time, due to increased loading, it is now more difficult to simply slide over the asperities. Hence, the asperities have little other option than deforming laterally or causing shearing. Some extra energy is required for causing the shear, but energy is lost only in case deformation is plastic. The combined effect of interfacial friction and interlocking of asperities results in a higher coefficient of friction in rough surface compared to smoother surfaces.

There is, however, an exception to this rule. When surfaces that are extremely smooth like glass on glass, the molecular interaction dominates the friction phenomenon. In fact, molecules are so close they tend to act as glues.

2.4 Categories of Friction

Coulomb friction is categorized into two different categories namely static and kinetic friction. This categorization is an oversimplification of the entire mechanism but is still a good approximation for engineering purposes.

- **Static Friction** - If a body is at rest on another surface, to start its sliding motion, a tangential force exceeding the static friction force has to be applied to it.

$$F_t \leq \mu_s F_N. \tag{2.3}$$

- **Kinetic Friction** - If a body is sliding across a surface, to maintain its velocity, a tangential force in the direction of the motion, and equal in magnitude to the dynamic friction force has to be applied.

$$F_t = \mu_k F_N. \tag{2.4}$$

There is a third type called the **rolling** friction. However, for perfect rolling or rolling without slipping, the type of friction between the wheel and the surface is of static type. This is because when the wheel touches the surface, the surface of the wheel deforms into a small flat patch. As long as the patch is in contact with the surface, there is no motion between the patch and the surface. When the wheel turns, the old patch simply ceases to exist, and a new contact patch is formed.

It should be noted that static friction acts as a constraint in a sense that if the tangential force does not exceed $\mu_s F_N$, the object can not slide. Hence there is no loss of energy due to static friction. In practice, under an unlubricated condition, also known as dry friction, $\mu_s \geq \mu_k$. Again, this statement is grossly simplified but can be explained by arguing that once the body starts moving the number of molecular bonds at any given instance is reduced. From Equation 2.2, which gives the forces due to molecular interaction, we can say slip time τ decreases when the body is in motion.

2.5 Modeling

We will briefly survey the tribology literature and figure out the prominent pattern in modeling friction. Modeling friction can be conceptually partitioned into two task - modeling the surface and modeling the contact. In a majority of cases the two tasks are extremely intertwined and are difficult to treat them as independent subjects. In tribology literature, people have modeled rough surfaces using statistical and spectral analysis tools. To simplify the contact problem, people assume the contacts to have regular geometric shapes like spheres, cylinders, planes, and bristles. Further, the contact can be modeled as plastic or elastic. These variations have resulted in a wide range of models capturing different aspects of friction.

It is well known that surface roughness may cause the real area under contact to be much smaller than the nominal area. The calculation of the area of contact and the

prediction of how this varies with load is not so straightforward. The prior art used the Hertzian theory of contact between spheres to individual contact spots [16]. However, there are two difficulties with this approach. The area of the contact spot depends on the radius of the asperity, which is not usually known. Second, it states that contact area is proportional to the applied load non-linearly as area $\propto$ load$^{(\frac{2}{3})}$. Both these obstacles were removed through the idea that although the overall stresses are in the elastic range the local stresses at the contact spots are much higher so that the elastic limit will be exceeded and the contact will yield plastically [18, 15]. Bowden et al. [8], used a similar approach in their model which assumes the asperities are plastically deformed. They found that the mean normal pressure corresponds to about three times the yield pressure or 2.8 times effective Young's modulus. The lateral stress due to adhesion was assumed to be equivalent to the shear strength of the soft material. Their model no longer needed the computation of real area of contact as the contact patch is assumed to have uniform yield pressure due to plastic deformation. This model was successfully used for the prediction of friction coefficient for various metal – metal contacts.

$$\mu_k = \frac{\text{Tangetial force}}{\text{Normal Force}} = \frac{\sigma A_c}{2.8 E A_c} = \frac{\sigma}{2.8 E^*}. \tag{2.5}$$

The above equation describes the relationship between the coefficient of friction and various elastic parameters described by Bowden et al. [8]. Quantity σ is the shear strength of the softer material, while E^* is the effective Young's modulus computed as Equation 2.12. Note that the true area under contact A_c is no longer required.

Archard et al. [2] pointed out that it is reasonable to assume plastic flow for the first few traversals of one body over another but the assumption losses credibility when same part come into contact multiple times. He showed that although the simple Hertzian theory did not predict the observed proportionality between contact area A and load, a generalized model in which each asperity is covered with micro asperities recursively – an idea like fractals, gave successively closer approximations to the Coulomb's law. Archard et al. showed that for any physically plausible surface if the number of contacts remains constant while the size of the asperities increases with load, the effective or real area is proportional to load$^{(\frac{2}{3})}$. On the other hand, if we increase the number of contacts while keeping the size of the asperity constant, the real area is proportional to the load.

To model the lateral forces due to friction, it is essential to consider molecular adhesive forces at the contact junctions. Gane et al. [14] performed detailed analysis of adhesive junction under light load or more importantly under an elastic condition for metals. They showed the relationship between electrical conductivity, applied load, material properties, and adhesive forces. Ogilvy et al. [30] removed the need for the

simplifying assumptions about the geometry of asperities by presenting a numerical model to predict the friction force between rough surfaces. The Ogilvy model considers contact between two numerically-generated Gaussian surfaces and uses the concept of equivalent surface roughness. The equivalent surface is a contact between an equivalent rough surface and a flat plane. The equivalent rough surface is obtained by summing up the heights of the two real surfaces and an effective modulus that represents their elastic moduli. Ogilvy calculated elastic deformation of asperities using the Hertz theory and accounted for their plastic deformation with a simple plasticity theory. Under conditions of steady sliding, they calculated friction force due to adhesion by using two methods that she called *microscopic* and *macroscopic*.

When using the microscopic method, the total friction force becomes the sum of the adhesive forces required to break every junction is given by

$$F = \sum_i f_i = \sum_i 2\pi r_i \Delta\gamma. \tag{2.6}$$

Here, r_i is the is the radius of curvature of the contacting peak and $\Delta\gamma$ is the adhesion surface energy of the two surfaces. The latter quantity is given by

$$\Delta\gamma = \gamma_1 + \gamma_2 - \gamma_{12}, \tag{2.7}$$

where scalars γ_1 and γ_2 are the surface energy for the two materials and γ_{12} is the surface energy of the combined surfaces.

In the macroscopic approach, the total friction force is related to the total true contact area under the assumption that the shear strength of adhesive junctions are constant, but without considering the details of each asperity under contact.

$$F = s \sum_i A_i, \tag{2.8}$$

where A_i is the real area of contact computed from Hertzian contact equation, and s is the shear strength of the interface. She observed a qualitative agreement between the numerical predictions of her model and the corresponding experimental results.

2.5.1 Hertzian Contact

At the end of the nineteenth century, Hertz et al. solved the contact problem of elasticity theory for the compression of ideally smooth bodies with primary contact along lines and at a point. The main application of the Hertz contact model is to compute the real Area of contact under vertical loads and the pressure distribution of the load at

the contact patch. The following assumptions are made in determining the solutions of Hertzian contact problems:

- The strains are small and within the elastic limit.
- Each body can be considered an elastic half-space, i.e., the area of contact is much smaller than the characteristic radius of the body.
- The surfaces are continuous and non-conforming.
- The bodies are in frictionless contact.

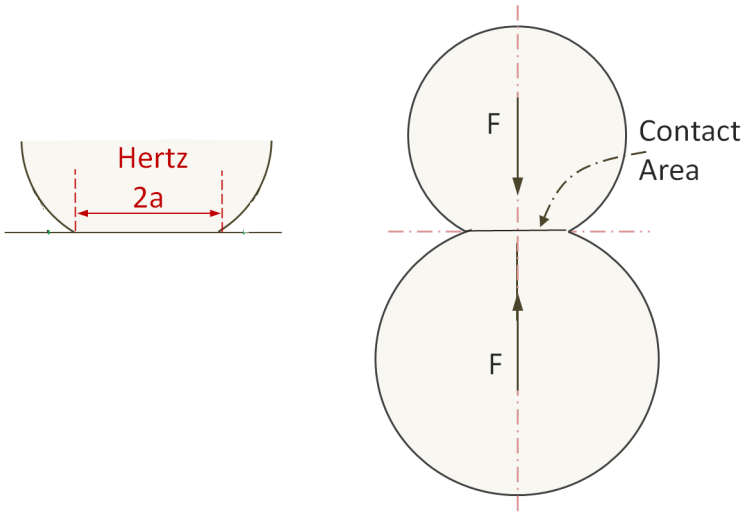

FIGURE 2.1: **Contact between two elastic spheres.**

Two Sphere Contact

Here we assume R_1 and R_2 are the radii of two spheres in contact. E_1 and E_2 are the Young's modulus and ν_1 and ν_2 are the Poisson ratio of the two bodies in contact. F is the normal load.

$$\text{Radius of contact patch: } a = \sqrt[3]{\frac{3FR_e}{4E^*}} \tag{2.9}$$

Pressure distribution as a function of distance r from the center of the contact patch is given as

$$p(r) = p_0\sqrt{1 - \frac{r^2}{a^2}} \tag{2.10}$$

where,

$$\frac{1}{R_e} = \frac{1}{R_1} + \frac{1}{R_2}, \tag{2.11}$$

$$\frac{1}{E^*} = \frac{1-\nu_1^2}{E_1} + \frac{1-\nu_2^2}{E_2}, \text{ and} \tag{2.12}$$

$$p_0 = \sqrt[3]{\frac{6FE^{*2}}{\pi^3 R_e^2}}. \tag{2.13}$$

2.5.2 Local Friction

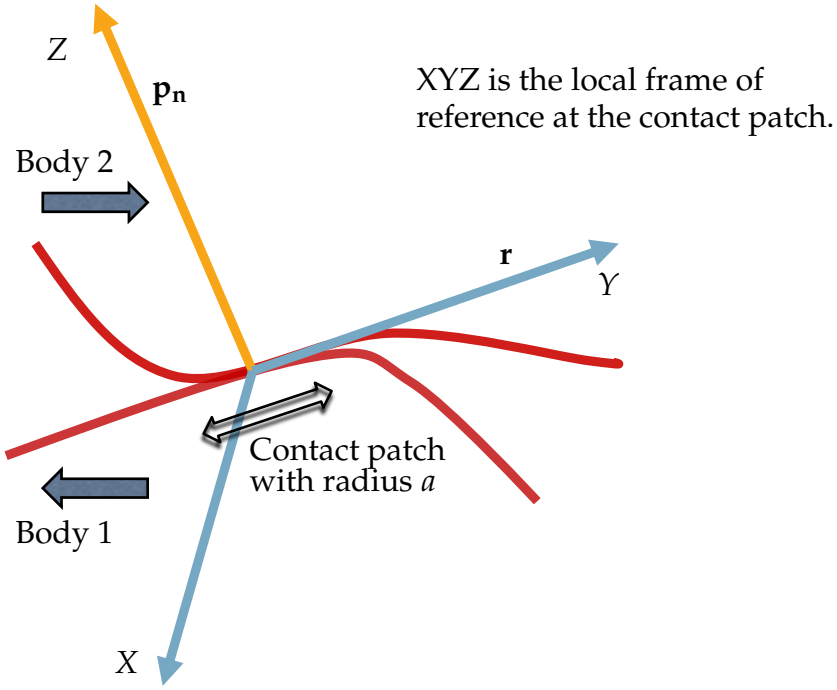

FIGURE 2.2: **Local contact model. The local friction force r is in the XY plane opposing the relative motion of the two bodies. The local normal pressure $\mathbf{p_n}$ acts along axis Z.**

To model the molecular interaction, researchers have employed empirical models for capturing local friction over a contact patch [20]. In this section, we will discuss an empirical model known as the binomial law. The resistance of an adhesive junction to shearing takes place in the local tangential direction and, in effect, constitutes the local friction.

The law is given as

$$r = a + bp_n, \tag{2.14}$$

where p_n is pressure along the local normal direction at the patch of contact and r is the resistive shear stress on the local XY plane opposing the direction of motion. As per Karpenko et al. [20] metals and polymers have b = 0.02-–0.25; for metal-–metal sliding a is between 2.5-30 MPa and for metal—polymer pairs a = 0.2-–0.5 MPa. .

The model computes the net normal force acting along the direction perpendicular to the **mean** surface as

$$N = \int\int_A (p_n cos\theta_x - r sin\theta_x) cos\theta_y \, dx \, dy. \tag{2.15}$$

In the above equation, A is the true contact area whereas θ_x, and θ_y represent the angle of the contact slopes with respect to axis-X and axis-Y directions in the global coordinate system.

Similarly, the net tangential force or force of friction opposing the direction relative motion along the mean surface is given as

$$F = \int\int_A (p_n sin\theta_x + r cos\theta_x) cos\theta_y \, dx \, dy. \tag{2.16}$$

2.5.3 Friction in Elastomer

Kinetic friction in rubber-like material or elastomer can be attributed to the dissipation of energy as heat inside the body of the material rather than plastic deformation or adhesion at the interface. This is a key concept which is true for highly viscous materials and is dependent on the relative velocity of the two material. The short wavelength asperities of the rough surface would excite rubber periodically at high frequencies with periodicity dependent on the velocity. This builds up vibrations inside the body of elastomer and the energy is dissipated as damping – a process more similar to forced vibrations in elastic solids [33, 35].

2.6 Summary of Tribology Literature Survey

It is important to understand the nature of friction based on the domain of application we wish to explore. With a wide variety of models to choose from, it is essential to consider the issues specific to the problem and select a model which fits bests to our problem description. In this chapter, we limit our discussion to dry friction in the context of static and kinetic friction.

A large majority of the models discussed in tribology literature focus on static friction. Since static friction deals with bodies that are not yet in motion, there is no dissipation of energy due to friction. This makes plastic deformation and adhesion like effects optional in case of static friction. However, if the surfaces are smooth or polished like glass, one must consider adhesion even for static case. A vast majority of literature predict static friction based on elastic deformation. The resistance to motion comes primarily from resistance to deforming the asperities. Greenwood et al. [15]

proposed a model composed of hemispherically tipped asperities with a uniform radius of curvature. They approximated the distribution of asperity heights about the mean plane as Gaussian. For mathematical convenience, contact between two rough surfaces was simulated as a contact between an equivalent rough surface and a rigid flat plane. The model used the elastic theory for the contact problem. Nayak et al. [28] extended the idea and introduced the techniques from random process theory into the analysis of Gaussian rough surfaces. He related surface statistics such as the distribution of summit heights, the density of summits, the mean surface gradient, and the mean curvature of summits to the power spectral density of a profile of the surface.

In the case of kinetic friction, effects like plastic deformation and adhesion are important. For a majority of real materials, it is no longer sufficient to just consider elastic deformation of asperities. Unlike the static case, we now have bodies which are in relative motion and dissipation of energy is important. Elastic deformation and material damping are insufficient to account for the energy losses in a majority of materials with the exception of elastomer. For soft and viscous material like rubber, the internal damping of the material may be sufficient in producing kinetic friction as detailed in Section 2.5.3. The common trend in modeling kinetic friction is through elastic contacts while introducing an energy dissipating element at the local junction. One such model is described in Section 2.5.2. The energy dissipation effects due to plastic deformation, wearing and adhesion can be introduced by tuning the parameters at local contacts. This model, however, cannot account for changes in surface geometry due to wearing and plastic deformation. For example, friction may reduce after rubbing two rough surfaces several times together due to breakage and plastic deformation of asperities. On the other hand, rubbing a rough hard metal against rubber may produce entirely different effects. HA et al. [13] generalized the asperity models described by Nayak et al. [28] to model sliding friction between rough surfaces. To model the sliding resistance at true contacts, they used empirical relations based on the data collected from various other sources in the tribology literature.

2.7 Friction in Computer Graphics Literature

The aspect of friction is not new in computer graphics literature. The first systematic study of friction in the context of graphics was done by Baraff et al. [4]. He showed that under Coulomb's law, even simplest of the rigid body contacts may not have a solution when using forces and acceleration for resolving contacts with friction and proved that deciding if non-impulsive contact forces are enough to prevent interpenetration is NP-complete. Hence, no efficient polynomial time algorithm is known to exist for computing contact forces. However, if impulsive forces are allowed, then a

solution exists for a wider class of problems and is guaranteed to have a solution for single contact [41] with linearly discretized friction cone. An LCP formulation for contact forces has been investigated by Baraff et al. [4], 1991 and Baraff et al. [5], 1994. In this approach, first the contact forces are computed and then a time integration step is performed. A more popular approach is to combine the acceleration resolution step with the integration step for velocities. This is known as the impulse based formulation. In this approach, the discontinuities of the contact forces are smoothed over by the time integration step resulting in a more tractable solution. The impulse based approach can be further differentiated into two categories - position based and velocity based. Position based approaches [41] guarantees at the end of each time step, the numerically generated trajectories do not violate rigid constraint at position level. While in the velocity based approaches [1, 12], the trajectories only take into account rigid constraint at velocity level. Using circular or elliptical cone results in a non-linear complementarity formulation. While such a non-linear complementarity formulation which is more accurate, there is less supporting theory to solve the equations arising from non-linear formulations. A more common approach is to use polyhedral approximations to the true cone which can approximate the true friction cones as accurately as desired [1, 12, 23] by increasing the number of vertices in the polyhedron. Kaufman et al. [21] approached the contact problem in linear vector space of rigid body velocities called se(3). In this space, a rigid body can be viewed as a point mass with multiple contact constraints acting on it. However, normal contact impulse is resolved independently of the friction while friction is introduced as a function of resolved impulse and friction coefficient. As far as true cones are concerned, we are yet to see its application in case of velocity based or position-based formulation for rigid bodies. However, true friction cone has been used for soft body hair simulation by Descoubes et al. [6]. In their approach, they fold the three Coulomb's law into a functional approximation and use non-smooth Newton method to solve the Nonlinear complementarity problem for point masses. Pabst et al. [31] uses friction tensors to model isotropic and spatially varying friction for fabrics. Their approach solves the contact problem by assuming contact to be frictionless and introduce friction as a function of normal force and tangential velocity obtained from the contact resolution step. Their approach is not entirely physically based but captures anisotropic effects with minimal computation.

2.8 Rigid Body Models

In this section, we discuss the approaches to incorporating friction in rigid body systems. We start with acceleration based approaches and discuss their issues before

moving to the more preferred velocity based approaches.

2.8.1 Resolving Contact Acceleration

In this section, we investigate contact resolution at the acceleration level and the associated issues in presence of friction. We start with the frictionless case and see how friction makes the contact problem unsolvable in many cases.

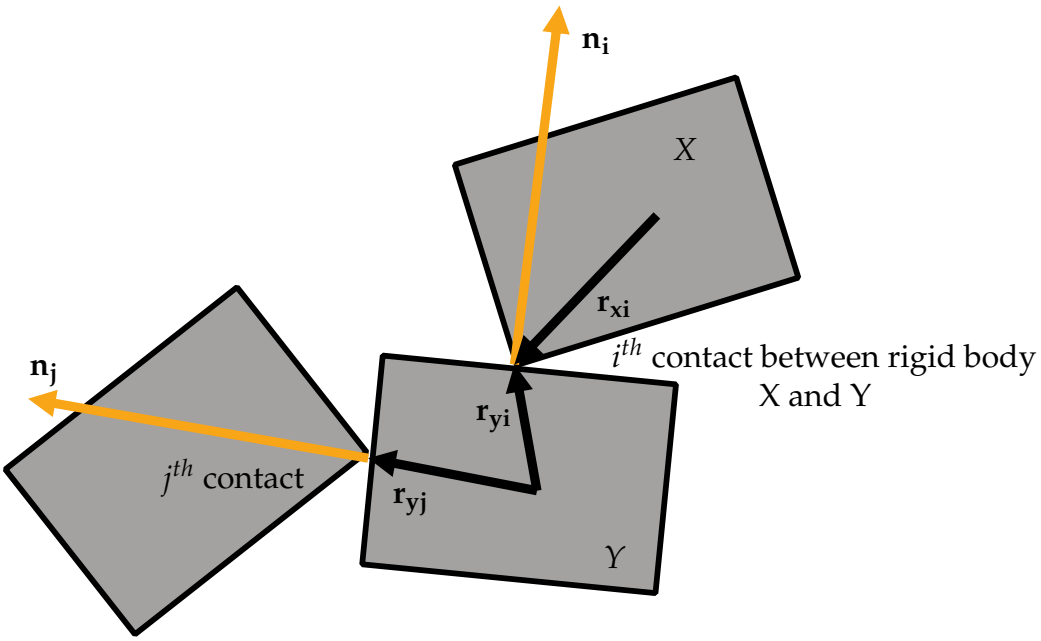

FIGURE 2.3: **Rigid body contact without friction**

Referring to Figure 2.3, the contact normal $\mathbf{n_{xi}}$ is associated with body X and $\mathbf{n_{yi}}$ with body Y for the i^{th} contact such that for $\mathbf{n_{xi}} = -\mathbf{n_{yi}} = \mathbf{n_i}$. The magnitude of normal contact force acting at the contact is f_i and is always positive. Thus, force acting on body X and Y is $f_i\mathbf{n_{xi}}$ and $f_i\mathbf{n_{yi}}$ respectively. We denote the relative normal acceleration of the two bodies for the i^{th} contact as a_i and is given by

$$a_i = (\mathbf{a_x} - \mathbf{a_y}) \cdot \mathbf{n_i}, \tag{2.17}$$

where we define vectors $\mathbf{a_x}$ and $\mathbf{a_y}$ as the acceleration for body X and Y at the i^{th} point of contact under the influences of external forces [3]. The accelerations are given as

$$\begin{aligned}\mathbf{a_x} = m_x^{-1}(\mathbf{f_x} + \sum_{k \in S_x} f_k \mathbf{n_{xk}}) + I_x^{-1}(\boldsymbol{\tau}_x + \sum_{k \in S_x} \mathbf{r_{xk}} \times f_k \mathbf{n_{xk}}) - \boldsymbol{\omega}_x \times I_x \boldsymbol{\omega}_x) \times \mathbf{r_{xi}} \\ + \boldsymbol{\omega}_x \times (\boldsymbol{\omega}_x \times \mathbf{r_{xi}}),\end{aligned} \tag{2.18}$$

$$\mathbf{a_y} = m_y^{-1}(\mathbf{f_y} + \sum_{k \in S_y} f_k \mathbf{n_{yk}}) + I_y^{-1}(\boldsymbol{\tau}_y + \sum_{k \in S_y} \mathbf{r_{yk}} \times f_k \mathbf{n_{yk}}) - \boldsymbol{\omega}_y \times I_y \boldsymbol{\omega}_y) \times \mathbf{r_{yi}} + \boldsymbol{\omega}_y \times (\boldsymbol{\omega}_y \times \mathbf{r_{yi}}), \tag{2.19}$$

where m is the mass, $\boldsymbol{\omega}$ is the angular velocity and I is the inertial tensor of a rigid body. Vector $\mathbf{f}$ and $\boldsymbol{\tau}$ denotes the force and torque due to non-contact forces. Sets S_x and S_y denotes all the contacts for body X and Y. It should be noted that the direction of normal n_k for the k^{th} contact should be such that normal force acting on contact pair is equal and opposite.

Equation 2.18 and 2.19 suggest that relative normal acceleration a_i is a linear combination of contact force magnitudes. For n contacts in the system, we can rewrite Equation 2.17 as

$$a_i = \sum_{j=1}^{n} a_{ij} f_j + b_i. \tag{2.20}$$

Comparing equation 2.17 and 2.20 we get,

$$b_i = (m_x^{-1}\mathbf{f_x} - m_y^{-1}\mathbf{f_y} + I_x^{-1}(\boldsymbol{\tau}_x - \boldsymbol{\omega}_x \times I_x \boldsymbol{\omega}_x) \times \mathbf{r_{xi}} - I_y^{-1}(\boldsymbol{\tau}_y - \boldsymbol{\omega}_y \times I_y \boldsymbol{\omega}_y) \times \mathbf{r_{yi}} + \boldsymbol{\omega}_x \times (\boldsymbol{\omega}_x \times \mathbf{r_{xi}}) - \boldsymbol{\omega}_y \times (\boldsymbol{\omega}_y \times \mathbf{r_{yi}})) \cdot \mathbf{n_i}. \tag{2.21}$$

The scalar a_{ij} denotes the effect of j^{th} contact on the i^{th} contact.

$$a_{ij} = \begin{cases} m_x^{-1} + m_y^{-1} + (\mathbf{r_{xi}} \times \mathbf{n_i})^T I_x^{-1} (\mathbf{r_{xi}} \times \mathbf{n_i}) & \\ + (\mathbf{r_{yi}} \times \mathbf{n_i})^T I_y^{-1} (\mathbf{r_{yi}} \times \mathbf{n_i}), & \text{if } j = i \\ m_x^{-1} \mathbf{n_{xj}} \cdot \mathbf{n_i} + (\mathbf{r_{xi}} \times \mathbf{n_i})^T I_x^{-1} (\mathbf{r_{xj}} \times \mathbf{n_{xj}}), & \text{if } j \in S_x \\ -m_y^{-1} \mathbf{n_{yj}} \cdot \mathbf{n_i} - (\mathbf{r_{yi}} \times \mathbf{n_i})^T I_y^{-1} (\mathbf{r_{yj}} \times \mathbf{n_{yj}}), & \text{if } j \in S_y \\ 0, & \text{otherwise} \end{cases} \tag{2.22}$$

For all a_i, b_i and f_i, $i \in 1, 2, 3 \ldots n$ we can write Equation 2.20 as

$$\mathbf{a} = A\mathbf{f} + \mathbf{b}, \tag{2.23}$$

where vector $\mathbf{a} = [a_1, a_2, \ldots a_n]^T$, $\mathbf{b} = [b_1, b_2, \ldots b_n]^T$ and $\mathbf{f} = [f_1, f_2, \ldots f_n]^T$. The elements of matrix A are given by a_{ij} defined by Equation 2.22. In absence of friction A is symmetric positive definite [4]. It should be noted that both $\mathbf{a}$ and $\mathbf{f}$ are unknown

and must satisfy the following conditions

$$f_i \geq 0,\ a_i \geq 0 \text{ and } f_i\, a_i = 0 \text{ for all } i \in 1, 2, \ldots n. \tag{2.24}$$

The conditions state that when contact forces are non-zero, the relative normal acceleration must be zero in order to avoid penetration. When relative normal acceleration is positive it means that contact is just about to break and thus no force is exerted by the contact. This is known as complementarity condition. The Equation 2.24 can be reformulated as following minimization problem:

$$\min_{\mathbf{f}} \mathbf{f}^{\mathbf{T}}(A\mathbf{f} + b),\ \text{s.t. } A\mathbf{f} + \mathbf{b} \geq 0 \text{ and } \mathbf{f} \geq 0. \tag{2.25}$$

Baraff et al. [4], 1991 solves Equation 2.25 directly using Lemke's algorithm. Baraff et al. [5], 1994 solves Equation 2.24 by iteratively fixing each pair f_i and a_i using Dantzig's algorithm which is more efficient compared to Lemke's algorithm.

Baraff et al. [4], 1991 studies the case of dynamic friction, where friction contact force is modeled as $f_i(\mathbf{n_i} + \mu\mathbf{t_i})$ where $\mathbf{t_i}$ is the direction opposing relative tangential velocity. Using Equation 2.22, it is easy to see that in case of dynamic friction, $a_{ij} \neq a_{ji}$ because $\mathbf{t_i} \cdot \mathbf{n_j} \neq \mathbf{t_j} \cdot \mathbf{n_i}$. Hence the matrix A is no longer symmetric. In such cases Lemke's algorithm no longer guarantees a solution for Equation 2.25 even for the cases which have solution. The author switches over to an impulse based formulation when Lemke's algorithm fails to find a solution. The case of static friction is modeled using dynamic friction by linearly interpolating the friction force between 0 to μf_n as a function of relative tangential velocity.

Baraff et al. [5], 1994 handles the case of static friction more elegantly by introducing tangential constraints. An orthogonal local frame of reference given by $[\mathbf{t_x}, \mathbf{t_y}, \mathbf{n}]$ is created at the point of contact as shown in Figure 2.4. The magnitude of contact forces along each basis vector are given by f_X, f_Y and f_N while the relative acceleration in each direction is a_X, a_Y and a_N. The approach differs from Baraff 1991 due to the fact that the new approach comes with two extra degrees of freedom for solving static friction. However, in dynamic case f_X, f_Y are no longer independent and depends upon f_N thus limiting the solvability of Equation 2.26 and 2.27.

Constraints in the normal direction are given as

$$\begin{aligned} &f_{Ni} \geq 0, \\ &a_{Ni} \geq 0 \text{ and} \\ &f_{Ni}\, a_{Ni} = 0 \text{ for } i \in 1, 2, \ldots n. \end{aligned} \tag{2.26}$$

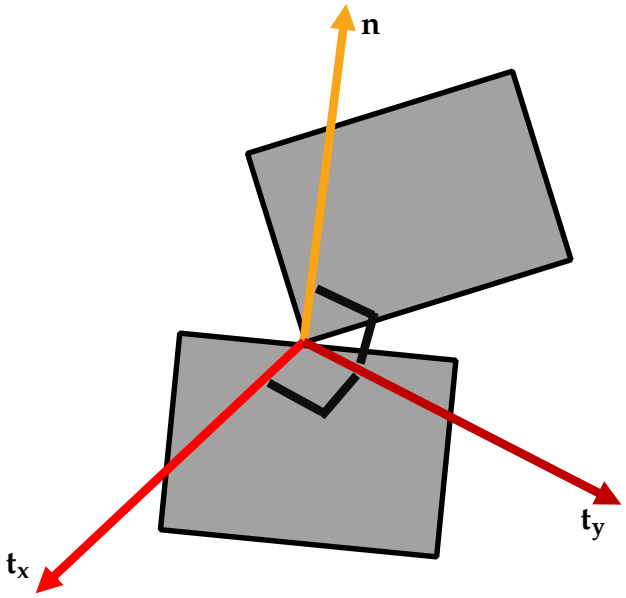

FIGURE 2.4: **Friction pyramid.**

Constraints in the tangential direction are given as

$$\begin{aligned} &\sqrt{f_{Xi}^2 + f_{Yi}^2} \leq \mu f_{Ni}, \\ &a_{Xi} f_{Xi} + a_{Yi} f_{Yi} \leq 0 \text{ and} \\ &\sqrt{a_{Xi}^2 + a_{Yi}^2} \left(\mu f_{Ni} - \sqrt{f_{Xi}^2 + f_{Yi}^2} \right) = 0 \text{ for } i \in 1, 2, \ldots n. \end{aligned} \tag{2.27}$$

The first condition in Equation 2.27 limits the magnitude of friction to a maximum of μf_{Ni}. The second condition is essentially saying the direction of friction force should oppose tangential acceleration. Note that this condition may not result in maximum energy dissipation due to friction as it doesn't require the tangential force and acceleration to be in exactly opposite direction. The last condition ensures that when tangential acceleration is non-zero, the magnitude of friction force reaches its maximum μf_{Ni}.

2.8.2 Resolving Contact Impulses

In this section, we will investigate resolving contacts using impulses. It is well known that contact forces and accelerations are insufficient to solve the complementarity problem defined by Equation 2.24 in presence of friction. The principle of constraints states that constraints should be satisfied by non-impulsive forces if possible; otherwise, impulsive forces should be used to satisfy constraints. However, deciding whether

non-impulsive contact forces are enough to prevent interpenetration is in itself NP-complete. Many have argued that the principle of constraint is too restrictive and there is no real justification for the principle [41, 4, 1]. On the other hand, an impulse based formulation-irrespective of whether non-impulsive forces have a solution or not is proposed which can tackle a far broader class of problems.

We begin by describing rigid body dynamics in velocity space. Let's define v_N the relative normal velocity between the two rigid bodies at the point of contact. We define v_N as

$$v_N = \begin{bmatrix} -\mathbf{n}^\mathbf{T} & -(\mathbf{r_x} \times \mathbf{n})^\mathbf{T} & \mathbf{n}^\mathbf{T} & (\mathbf{r_y} \times \mathbf{n})^\mathbf{T} \end{bmatrix} \begin{bmatrix} \mathbf{v_x} \\ \boldsymbol{\omega}_x \\ \mathbf{v_y} \\ \boldsymbol{\omega}_y \end{bmatrix}, \tag{2.28}$$

where $\mathbf{v}$ is the velocity of the center of mass and $\boldsymbol{\omega}$ is the angular velocity of the rigid bodies in contact. The vector $\mathbf{r}$ indicates the direction of the point of contact from the center of mass. Assuming the number of contacts is n and the number of rigid bodies in the system is m, we can write the relative normal velocities for all rigid bodies in vector form as follows

$$\mathbf{u} = \begin{bmatrix} \mathbf{v_1} & \boldsymbol{\omega}_1 & \mathbf{v_2} & \boldsymbol{\omega}_2 & \dots & \mathbf{v_m} & \boldsymbol{\omega}_m \end{bmatrix}^T. \tag{2.29}$$

Here, $\mathbf{u} \in R^{6m}$ is the concatenated vector containing the velocities of center of mass and angular velocities of the rigid bodies in the system. We can write equation 2.28 in vector form as

$$\begin{aligned} \mathbf{v_N} &= J_N \mathbf{u}, \text{ s.t. } J_N \in R^{n \times 6m} \\ &= [v_0, v_1, ..., v_n]^T, \end{aligned} \tag{2.30}$$

where the scalar components of $\mathbf{v_N}$ are denoted using v_i and gives the relative normal velocity at the point of i^{th} contact. The sparse matrix J_N is also known as contact Jacobian is obtained by placing one constraint in each row. The columns of J_N is indexed using the index of the rigid bodies in the system. In each row, just two columns are non-zero indicating the two rigid bodies in contact. For example, if two rigid bodies X and Y are in contact, then x^{th} and y^{th} column is $[-\mathbf{n}^\mathbf{T} \ -(\mathbf{r_x} \times \mathbf{n})^\mathbf{T}]$ and $[\mathbf{n}^\mathbf{T} \ (\mathbf{r_y} \times \mathbf{n})^\mathbf{T}]$ respectively.

To avoid interpenetration between two rigid bodies at the point of contact we must have

$$\mathbf{v_N} \geq \mathbf{0}. \tag{2.31}$$

Since constraint forces act along the constraint normal, we have

$$\begin{aligned} \mathbf{F_N} &= J_N^T \mathbf{f_N} \text{ and} \\ \mathbf{f_N} &= [f_0,\ f_1, \ldots f_n]^T, \end{aligned} \tag{2.32}$$

where f_i, for $i \in 0 \ldots n$ is the magnitude of impulse acting along the normal direction. The vector $\mathbf{F_N}$ contains the normal constraint impulse and angular impulse pair for all the rigid bodies in the system.

To avoid interpenetration, we need the normal forces to be repulsive when the rigid body pair is in contact and must be zero when they separate. This yields the complementarity constraint defined as follows

$$v_i \geq 0 \text{ compl. } f_i \geq 0 \text{ for all } i \in 1...n. \tag{2.33}$$

We write the time discretized equation of motion as

$$M(\mathbf{u_{t+1}} - \mathbf{u_t}) = \mathbf{F_N} + \mathbf{F_{ext}} \Delta t, \tag{2.34}$$

where $M \in R^{6m \times 6m}$ is the generalized mass matrix given by

$$M = \begin{bmatrix} m_1 I & & & & & & 0 \\ & I_1 & & & & & \\ & & m_2 I & & & & \\ & & & I_2 & & & \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ & & & & & m_m & \\ 0 & & & & & & I_m \end{bmatrix}. \tag{2.35}$$

In the above equation, scalar m_i is the mass of the i^{th} body, I as the identity matrix, I_i as the corresponding inertia tensor. Other forces are given by $\mathbf{F_{ext}} \in R^{6m}$ and Δt is step size. Using Equation 2.30 and 2.34, we can write the relative normal velocity in vector form as

$$\begin{aligned} \mathbf{v_N} &= J_N \, \mathbf{u_{t+1}} \\ &= J_N M^{-1} J_N^T \mathbf{f_n} + J_N \, M^{-1} \, \mathbf{F_{ext}} \Delta t + J_N \, \mathbf{u_t}. \end{aligned} \tag{2.36}$$

We can rewrite the above equation more compactly as

$$\mathbf{v_N} = A\mathbf{f_n} + b \geq \mathbf{0}, \text{ s.t. } A = J_N M^{-1} J_N^T \text{ and } b = J_N(M^{-1} \, \mathbf{F_{ext}} \Delta t + \mathbf{u_t}). \tag{2.37}$$

Using complementarity condition in Equation 2.33 and Equation 2.37, we solve for the collision impulses $\mathbf{f_n}$. There are a variety of ways to solve the problem. Erleben et al. [12] used Projected Gauss-Seidel. Other techniques for solving the complementarity problem include Gauss Jacobi, Lemke's algorithm or Dantzig's algorithm.

Friction Pyramid Approximation

In this section, we augment our model described in the previous section with approximate Coulomb friction. We use friction pyramid approximation of Coulomb friction using two orthogonal direction vectors on the friction tangent plane. The friction force acting along the two directions must be resolved such that it opposes the relative tangential motion of the two objects in contact.

We first define the relative velocities along the two tangential directions $\mathbf{t_1}$ and $\mathbf{t_2}$ as

$$v_{Tk} = \begin{bmatrix} -\mathbf{t_k^T} & -(\mathbf{r_x} \times \mathbf{t_k})^\mathbf{T} & \mathbf{t_k^T} & (\mathbf{r_y} \times \mathbf{t_k})^\mathbf{T} \end{bmatrix} \begin{bmatrix} \mathbf{v_x} \\ \boldsymbol{\omega}_x \\ \mathbf{v_y} \\ \boldsymbol{\omega}_y \end{bmatrix} \text{ for } k = 1 \text{ and } 2. \tag{2.38}$$

Similar to Equation 2.30, we can write the relative tangential velocities of all bodies in vector form as follows

$$\mathbf{v_T} = J_T \mathbf{u}, \text{ s.t. } J_T \in R^{2n \times 6m}, \text{ and} \tag{2.39}$$

$$\mathbf{v_T} = [v_{t1_1}, v_{t2_1}, ..., v_{t1_n}, v_{t2_n}]^T \text{ s.t. } \mathbf{v_T} \in R^{2n}. \tag{2.40}$$

The matrix J_T is the contact Jacobian corresponding to the tangential impulses. We build the matrix J_T similar to J_N except that we introduce two rows per contact corresponding to the two tangential constraints instead of just one in the normal(J_N) case. We can write tangential impulses $\mathbf{F_T}$ acting on the rigid bodies as

$$\mathbf{F_T} = J_T^T \mathbf{f_T}, \text{ where} \tag{2.41}$$

$$\mathbf{f_T} = [f_{t1_0}, \ f_{t2_0}, ..., f_{t1_n}, \ f_{t2_n}]^T, \text{ s.t. } \mathbf{f_T} \in R^{2n}. \tag{2.42}$$

In the above equation, scalars f_{1i} and f_{2i} for $i \in 1 \ldots n$ are the magnitude of contact impulses along the two tangential direction $\mathbf{t_1}$ and $\mathbf{t_2}$. Next, we define the constraints

imposed by Coulomb's law as

$$\begin{aligned} &|f_{t1_i}| \leq \mu f_i \\ &v_{t1_i} f_{t1_i} \leq 0 \\ &v_{t1_i}(\mu f_i - |f_{t1_i}|) = 0 \text{ for all } i \in 1 \dots n. \end{aligned} \tag{2.43}$$

In the above equations, the first condition corresponds to the fact that the magnitude of contact impulse along the first tangent is not allowed to exceed μf_i. The second condition corresponds to the dissipation of energy due to friction forces. Essentially, the tangential force and velocity must be in opposite direction. The final condition asserts the static friction case. As long as $(\mu f_i - |f_{ti}|)$ is non zero, the tangential velocity must be zero. Similarly, we can write the conditions for the second tangential direction as

$$\begin{aligned} &|f_{t2_i}| \leq \mu f_i, \\ &v_{t2_i} f_{t2_i} \leq 0, \\ &v_{t2_i}(\mu f_i - |f_{t2_i}|) = 0 \text{ for all } i \in 1 \dots n. \end{aligned} \quad . \tag{2.44}$$

However with the above constraints 2.43 and 2.43, the maximum possible friction force is $\sqrt{2}$ times μf_i, which is inaccurate. Also, the two constraints only ensure dissipation of energy and not maximal dissipation. Maximal dissipation occurs when $\mathbf{v_t}$ and $\mathbf{f_t}$ are aligned exactly opposite to each other. We can improve the quality by increasing the number of directions used for spanning the friction plane as discussed in the next section. Similar to Equation 2.36 we can write the relative velocities at the contact along the normal and tangential direction as

$$\begin{bmatrix} \mathbf{v_N} \\ \mathbf{v_T} \end{bmatrix} = \begin{bmatrix} J_N \\ J_T \end{bmatrix} M^{-1} \begin{bmatrix} J_N^T & J_T^T \end{bmatrix} \begin{bmatrix} \mathbf{f_N} \\ \mathbf{f_T} \end{bmatrix} + \begin{bmatrix} J_N \\ J_T \end{bmatrix} M^{-1} \mathbf{F_{ext}} \Delta t + \begin{bmatrix} J_N \\ J_T \end{bmatrix} \mathbf{u_t}. \tag{2.45}$$

To resolve the collisions implementing friction pyramid, we need to solve Equation 2.45 subject to constraints given by Equation 2.33, 2.43 and 2.43. Frictional contact is commonly solved using Gauss-Siedel. A Recent study [26] proposed a new method for solving frictional contact using accelerated projected gradient descent (APGD) having good convergence in several test cases and parallelizability.

2.8.3 Approximate Friction Cone

In this section, we see how we can accurately approximate Coulomb friction cone without introducing the inherent non-linearity associated with a friction cone. The Coulomb friction law requires the contact force to lie in a circular cone (or elliptic cone

for anisotropic friction). This is a difficult condition to deal with since it leads to a nonlinear complementarity problem which is much more difficult to analyze and/or to solve numerically. Therefore, the cone is approximated by a polygonal cone. The idea is very similar to the friction pyramid setting where instead of using just two tangents, we use multiple tangents which spans the friction plane at the point of contact. Some other linear constraints are also imposed so that the friction forces do not exceed a certain value.

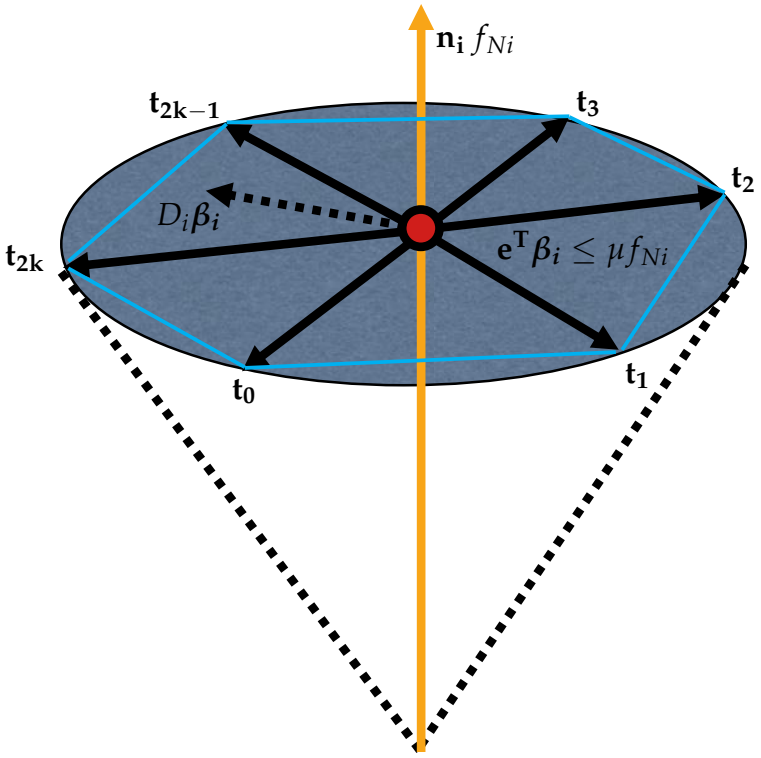

FIGURE 2.5: **Approximate linearized friction cone. The convex hull marked with blue is given by the condition $\mathbf{e}^T\beta_i = \mu f_{Ni}$. The direction of the net tangential force $D_i\beta_i$ is shown using dashed arrows.**

The net contact forces acting at the point of contact $\mathbf{p}$ is given as

$$FC(\mathbf{p}) = \{f_{Ni}\mathbf{n_i} + D_i\boldsymbol{\beta_i} |\ f_{Ni} \geq 0;\ \boldsymbol{\beta_i} \geq \mathbf{0};\ \mathbf{e}^T\boldsymbol{\beta_i} \leq \mu f_{Ni}\}. \tag{2.46}$$

The above function FC is known as the linearized friction cone. We define $\mathbf{n_i}$ as the outward unit normal at contact point i and a set of paired unit tangent vectors $\mathbf{t_{ji}}$ for $j = 1 \ldots 2k$, such that, $|\mathbf{t_{ji}}| = 1$ and $\mathbf{t_{(2j-1)i}} = -\mathbf{t_{(2j)i}}$, for $j = 1 \ldots k$. In simpler words, the tangent vectors appear in opposing pairs as shown in Figure 2.5. We assemble the tangents column wise into matrix $D_i = \{t_{0i},\ t_{1i} \ldots t_{(2k)i}\}$. Scalar f_{Ni} is magnitude of the normal force or impulse at the point of contact. The vector $\boldsymbol{\beta_i} \in R^{2k}$ is the tangential counterpart of the normal forces. Each scalar component of $\boldsymbol{\beta_i}$ is the

magnitude of the force or impulse acting along the tangential directions. The vector $\mathbf{e} = \{1, 1, \ldots 1\}$.

The net force acting along the tangential direction is given by $D_i\boldsymbol{\beta_i}$. While in friction pyramid setting, we did not put any limit on the magnitude of the net tangential forces whereas in polygonal approximation of friction cone, sum of the magnitude of tangential forces are not allowed to exceed μf_{Ni} which forces the net tangential force to remain within the boundary marked by the blue convex hull as shown in Figure 2.5.

Using the principle of maximum energy dissipation, the net force in the tangent plane must oppose the relative tangential velocity (if any). We can write the condition as

$$\begin{aligned} &\min_{\boldsymbol{\beta}}(D\boldsymbol{\beta}) \cdot \mathbf{v_{rel}} \\ &= \min_{\boldsymbol{\beta}} \boldsymbol{\beta}^T (D^T \mathbf{v_{rel}}) \\ &= \min_{\boldsymbol{\beta}} \boldsymbol{\beta}^T \left(\begin{bmatrix} -\mathbf{t_0^T} & -(\mathbf{r_x} \times \mathbf{t_0})^\mathbf{T} & \mathbf{t_0^T} & (\mathbf{r_y} \times \mathbf{t_0})^\mathbf{T} \\ \vdots & & \vdots & \\ -\mathbf{t_{2k}^T} & -(\mathbf{r_x} \times \mathbf{t_{2k}})^\mathbf{T} & \mathbf{t_{2k}^T} & (\mathbf{r_y} \times \mathbf{t_{2k}})^\mathbf{T} \end{bmatrix} \begin{bmatrix} \mathbf{v_x} \\ \boldsymbol{\omega}_x \\ \mathbf{v_y} \\ \boldsymbol{\omega}_y \end{bmatrix} \right). \end{aligned} \tag{2.47}$$

Using the above equation and inequality constraints from Equation 2.46, we can write the Lagrangian for the system as

$$L = \boldsymbol{\beta}^T \left(\begin{bmatrix} -\mathbf{t_0^T} & -(\mathbf{r_x} \times \mathbf{t_0})^\mathbf{T} & \mathbf{t_0^T} & (\mathbf{r_y} \times \mathbf{t_0})^\mathbf{T} \\ \vdots & & \vdots & \\ -\mathbf{t_{2k}^T} & -(\mathbf{r_x} \times \mathbf{t_{2k}})^\mathbf{T} & \mathbf{t_{2k}^T} & (\mathbf{r_y} \times \mathbf{t_{2k}})^\mathbf{T} \end{bmatrix} \begin{bmatrix} \mathbf{v_x} \\ \boldsymbol{\omega}_x \\ \mathbf{v_y} \\ \boldsymbol{\omega}_y \end{bmatrix} \right) - \boldsymbol{\alpha}^T \boldsymbol{\beta} - \lambda(\mu f_N - \mathbf{e^T}\boldsymbol{\beta}). \tag{2.48}$$

Differentiating w.r.t $\boldsymbol{\beta}$ and λ we obtain the following complementarity conditions

$$\begin{bmatrix} -\mathbf{t_0^T} & -(\mathbf{r_x} \times \mathbf{t_0})^\mathbf{T} & \mathbf{t_0^T} & (\mathbf{r_y} \times \mathbf{t_0})^\mathbf{T} \\ \vdots & & \vdots & \\ -\mathbf{t_{2k}^T} & -(\mathbf{r_x} \times \mathbf{t_{2k}})^\mathbf{T} & \mathbf{t_{2k}^T} & (\mathbf{r_y} \times \mathbf{t_{2k}})^\mathbf{T} \end{bmatrix} \begin{bmatrix} \mathbf{v_x} \\ \boldsymbol{\omega}_x \\ \mathbf{v_y} \\ \boldsymbol{\omega}_y \end{bmatrix} + \mathbf{e}\lambda \geq \mathbf{0}, \text{ compl. } \boldsymbol{\beta} \geq \mathbf{0} \tag{2.49}$$

$$\mu f_N - \mathbf{e^T}\boldsymbol{\beta} \geq 0, \text{ compl. } \lambda \geq 0.$$

We write the discretized equation of motion for all rigid bodies in the system as

$$\mathbf{u_{t+1}} = M^{-1} \begin{bmatrix} J_N^T & J_T^T \end{bmatrix} \begin{bmatrix} \mathbf{f_N} \\ \tilde{\boldsymbol{\beta}} \end{bmatrix} + \mathbf{u_t} + M^{-1}\mathbf{F_{ext}}\Delta t \text{ where} \tag{2.50}$$

$$\tilde{\boldsymbol{\beta}} = \begin{bmatrix} \boldsymbol{\beta_0} & \boldsymbol{\beta_1} & \dots & \boldsymbol{\beta_n} \end{bmatrix}^T. \tag{2.51}$$

In Equation 2.50, similar to friction pyramid setting, the constraint Jacobian for tangential constraints is $J_T \in R^{2kn \times 6m}$, where n is the number of contacts, m is the number of rigid bodies and k is the number of tangents spanning the tangent plane at each contact. Vector $\tilde{\boldsymbol{\beta}}$ is constructed by concatenating the tangential impulses $\boldsymbol{\beta_i}$ for all n contacts.

Next, we rewrite complementarity condition outlined by Equation 2.49 for all rigid bodies as

$$\begin{aligned} J_T \mathbf{u_{t+1}} + E\tilde{\boldsymbol{\lambda}} \geq \mathbf{0} \text{ compl. } \tilde{\boldsymbol{\beta}} \geq \mathbf{0} \\ U\mathbf{f_N} - E^T\tilde{\boldsymbol{\beta}} \geq \mathbf{0} \text{ compl. } \tilde{\boldsymbol{\lambda}} \geq \mathbf{0}. \end{aligned} \tag{2.52}$$

In the above equation, we define matrices E, U, and the vector $\tilde{\boldsymbol{\lambda}}$ as

$$\begin{aligned} E = diag(\mathbf{e^0}, \mathbf{e^1}, ..., \mathbf{e^n}),\ E \in R^{2kn \times n} \\ U = diag(\mu_0, \mu_1, ..., \mu_n),\ U \in R^{n \times n} \\ \tilde{\boldsymbol{\lambda}} = \begin{bmatrix} \lambda_0 & \lambda_1 & \dots & \lambda_n \end{bmatrix}^T. \end{aligned} \tag{2.53}$$

In the above equation, the function *diag* creates a matrix as follows

$$diag(\mathbf{e^0}, \mathbf{e^1} ... \mathbf{e^n}) = \begin{bmatrix} \mathbf{e^0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{e^1} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{e^n} \end{bmatrix} \text{ and}$$

$$\mathbf{e^0} = \mathbf{e^1} = \dots \mathbf{e^n} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \in R^{2k \times 1}.$$

We now rewrite Equation 2.50 and complementarity condition 2.52 in matrix notation as [1]

$$\begin{bmatrix} I & M^{-1}J_N^T & M^{-1}J_T^T & 0 \\ J_N & 0 & 0 & 0 \\ J_T & 0 & 0 & E \\ 0 & U & -E^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{u_{t+1}} \\ \mathbf{f_N} \\ \tilde{\boldsymbol{\beta}} \\ \tilde{\boldsymbol{\lambda}} \end{bmatrix} + \begin{bmatrix} \mathbf{u_t} + \mathbf{F_{ext}}\Delta t \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \tilde{\boldsymbol{\rho}} \\ \tilde{\boldsymbol{\sigma}} \\ \tilde{\boldsymbol{\zeta}} \end{bmatrix} \tag{2.54}$$

$$\begin{bmatrix} \mathbf{f_N} \\ \tilde{\beta} \\ \tilde{\lambda} \end{bmatrix}^T \begin{bmatrix} \tilde{\rho} \\ \tilde{\sigma} \\ \tilde{\zeta} \end{bmatrix} = \mathbf{0}, \quad \begin{bmatrix} \mathbf{f_N} \\ \tilde{\beta} \\ \tilde{\lambda} \end{bmatrix} \geq \mathbf{0}, \text{ and } \begin{bmatrix} \tilde{\rho} \\ \tilde{\sigma} \\ \tilde{\zeta} \end{bmatrix} \geq \mathbf{0} \tag{2.55}$$

This kind of formulation is known as dual formulation and variables $\tilde{\rho}$, $\tilde{\sigma}$, $\tilde{\zeta}$ are slack variables. Anitescu et al. [1] showed that Equation 2.54 and 2.55 can be solved using Lemke's algorithm; however there can be more than one solution to the problem.

2.8.4 Data Driven Techniques

Jiang et al. [19] proposed a data driven technique where they augment traditional contact with data driven approach for rigid bodies. They train a classifier and a regressor using synthetic data. The classifier is used to identify the regime in which the contact operates i.e. static, dynamic or separating. In the static case, they use a traditional solver with an additional constraint forcing the point in contact to have zero velocity. For separating case, they simply don't apply any impulses. In the dynamic case, they decouple the constraint impulse into a normal and tangential component. The normal component of the impulse is again solved using traditional techniques such that it doesn't violate the non-penetration constraint. For the tangential component, they use the regressors to predict the tangential friction force. The author showed the validity of their method in the 2D case for a single rigid object as well as for articulated joints. The data is obtained from DART simulator which uses an approximate coulomb friction.

2.9 Soft Body Models

As shown in Figure 2.6, a straightforward approach for modeling soft body systems is to use lumped masses with internal forces between the point masses holding the system together. The internal forces can be modeled using a mass-spring system or using finite element elastic models. The equation of motion for general soft body system is given by

$$M\ddot{\mathbf{x}} + (\alpha M + \beta K(\mathbf{x}))\dot{\mathbf{x}} + \mathbf{f_{int}}(\mathbf{x}) = \mathbf{f_{ext}}. \tag{2.56}$$

In the above equation, vector $\mathbf{x} \in R^{3n}$ is the position of the particles in the system. The matrix M is the diagonal mass matrix. The matrix K, also known as the stiffness matrix is usually a function of the configuration of the particles in the system. The stiffness matrix indicates the connectivity among the particles and how strong the interaction between them is. The scalar quantity α and β are known as the Rayleigh

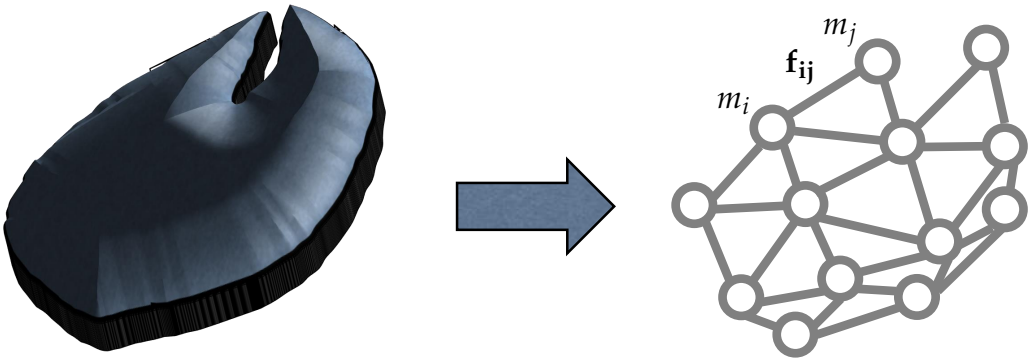

FIGURE 2.6: **Discretization of soft a body. The mass of the soft body is distributed over lumped masses m_i and internal forces $\mathbf{f_{ij}}$ hold the masses together.**

damping parameters. Increasing α is like increasing viscous damping where motion in any direction is damped equally proportional to its mass and coefficient α. Increasing the parameter β causes relative motion between a pair of particles to reduce. The vector $\mathbf{f_{int}}$ gives the internal forces acting on each particle whereas $\mathbf{f_{ext}}$ models the external forces like gravity and collisions forces.

2.9.1 Soft Body Contact

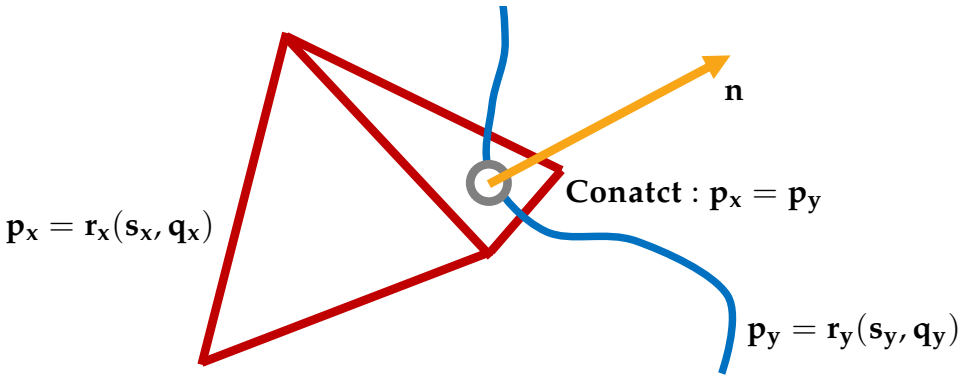

FIGURE 2.7: **Soft body contact for two arbitrary soft bodies with different parameterizations. A position $\mathbf{p_x}$ on soft body X (in red) is a function of parameters $\mathbf{s_x}$ and $\mathbf{q_x}$ while position $\mathbf{p_y}$on soft body Y is a parameterized by $\mathbf{s_y}$ and $\mathbf{q_y}$**

A position $\mathbf{p}$ on a body can be defined as a function of its degrees of freedom $\mathbf{q}$ and a parameter $\mathbf{s}$ which uniquely identifies the position on the body. Degrees of freedom explains the configuration of the body in space. For example, in a lumped

mass tetrahedron, the number of degrees of freedom is 12 i.e. 3 for each vertex. The number of degrees of freedom required to specify a rigid body of any shape is just 6. Given the configuration of the body, the parameter **s** uniquely identifies any position on the body. For example, barycentric coordinates of triangle or tetrahedron. Usually we consider **s** independent of **q**. Hence for any soft or rigid model

$$\mathbf{p} = \mathbf{r}(\mathbf{s}, \mathbf{q}), \ \mathbf{p} \in R^3. \tag{2.57}$$

We find the velocity of a point on the body assuming **s** is not changing with time as

$$\mathbf{v} = \frac{d\mathbf{p}}{dt} = \frac{\partial \mathbf{p}}{\partial \mathbf{q}}\dot{\mathbf{q}}. \tag{2.58}$$

As shown in Figure 2.7, let us consider two soft bodies in contact. The relative motion between the two bodies X and Y at the point of contact is given by

$$\mathbf{u_{xy}} = \begin{bmatrix} \frac{\partial \mathbf{p_x}}{\partial \mathbf{q_x}} & -\frac{\partial \mathbf{p_y}}{\partial \mathbf{q_y}} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}}_\mathbf{x} \\ \dot{\mathbf{q}}_\mathbf{y} \end{bmatrix}. \tag{2.59}$$

We can write the relative contact velocity for all contacts in the system as

$$\mathbf{u} = J\dot{\mathbf{q}}, \ s.t., \ J \in R^{3n \times m}. \tag{2.60}$$

The scalar n is the number of contacts and m is the total number of degrees of freedom for the system. Matrix J is the contact Jacobian. In the Equation 2.60, the i^{th} row of J is given as

$$J_i = \begin{bmatrix} \mathbf{0} & \dots & \frac{\partial \mathbf{p_x}}{\partial \mathbf{q_x}} & \dots & -\frac{\partial \mathbf{p_y}}{\partial \mathbf{q_y}} & \dots & \mathbf{0} \end{bmatrix}. \tag{2.61}$$

We can write the equation of motion as

$$M\ddot{\mathbf{q}} + \mathbf{f_{int}} = J^T\mathbf{f}. \tag{2.62}$$

In the above equation, **f** is the force acting at the point of contacts and is unknown. To avoid interpenetration, the normal component of **u** must be zero. This condition is along with Equation 2.62 can be used to solve the equation of motion in the frictionless case.

2.9.2 True Friction Cone

In this section, we provide a functional characterization of Coulomb friction [6]. We find a function F in $R^3 \times R^3 \rightarrow R^3$ such that

$$F(\mathbf{u_i}, \mathbf{f_i}) = \mathbf{0},\ s.t.\ (\mathbf{u_i}, \mathbf{f_i}) \in \text{Contact}(\mathbf{n_i}, \mu_i)\ for\ all\ i \in 1...n. \tag{2.63}$$

Here, the contact normal $\mathbf{n_i}$ and coefficient of friction μ_i uniquely identifies the i^{th} contact. The vector $\mathbf{u_i}$ and $\mathbf{f_i}$ denotes the relative velocity at the contact and contact force respectively. The function F is split into normal and tangential component as follows

$$F(\mathbf{u_i}, \mathbf{f_i}) = \begin{bmatrix} F_N(u_{Ni}, f_{Ni}) \\ F_T(\mathbf{u_{Ti}}, \mathbf{f_{Ti}}) \end{bmatrix}. \tag{2.64}$$

The normal and tangential components are given as

$$F_N(\mathbf{u_i}, \mathbf{f_i}) = \begin{cases} R^3 \times R^3 \rightarrow R \\ P_{R+}(f_N - u_N) - f_N \end{cases} \tag{2.65}$$

$$F_T(\mathbf{u_i}, \mathbf{f_i}) = \begin{cases} R^3 \times R^3 \rightarrow R^2 \\ P_{B(\mathbf{0},\ \mu_i r_n)}(\mathbf{f_{Ti}} - \mathbf{u_{Ti}}) - \mathbf{f_{Ti}}. \end{cases} \tag{2.66}$$

Here, function $P_{R+}(x)$ clamps the argument in $[0, \infty)$ range. Function $B(\mathbf{0}, x)$ defines a 2D circle centered at zero with radius x. The function $P_B(\mathbf{0}, a)(\mathbf{x})$ clamps the length of $\mathbf{x}$ between 0 to a, without changing its direction.

The function F is simple to understand if we look at each case individually. In all three cases - static, dynamic and separating, the function F must evaluate to zero. First, we consider the case when $u_N > 0$. In this case $f_N = 0$ and $\mathbf{f_T} = \mathbf{0}$ since there is no contact or the contact is breaking. Hence $F_N(\mathbf{u}, \mathbf{f}) = 0$. Using the fact that $P_{B(\mathbf{0},0)}(x) = \mathbf{0}$, we can verify $F_T(\mathbf{u}, \mathbf{f}) = \mathbf{0}$. In the sticking case, we have $\mathbf{u} = \mathbf{0}$ and $f_N > 0$. Therefore, $P_{R+}(f_N) = f_N$ and hence $F_N(\mathbf{u}, \mathbf{f}) = 0$. Similarly, $F_T(\mathbf{u}, \mathbf{f}) = \mathbf{0}$ as $P_{B(\mathbf{0},\ \mu_i r_n)}(\mathbf{f_T}) = \mathbf{f_T}$. Finally in the sliding case, we have $u_N = 0$ and due to principle of maximum dissipation we have direction of $\mathbf{f_T}$ and $\mathbf{u_T}$ opposing each other. Hence, $P_{B(\mathbf{0},\ \mu_i r_n)}(\mathbf{f_T} - \mathbf{u_T}) = P_{B(\mathbf{0},\ \mu_i r_n)}(\alpha \mathbf{f_T}) = \mathbf{f_T}$, where $\alpha \geq 1$. Hence both F_N and F_T are zero in sliding case.

Now we need to solve Equation 2.62 and 2.63 simultaneously. We first discretize Equation 2.62 as

$$\dot{\mathbf{q}}_{\mathbf{t+1}} = M^{-1} J^T \mathbf{f} - M^{-1} \mathbf{f_{int}} + \dot{\mathbf{q}}_{\mathbf{t}}. \tag{2.67}$$

Here, we assume the forces are impulsive in type. Using Equation 2.60 we have

$$\mathbf{u} = J\dot{\mathbf{q}}_{\mathbf{t+1}} = JM^{-1}J^T\mathbf{f} - JM^{-1}\mathbf{f}_{\mathbf{int}} + J\dot{\mathbf{q}}_{\mathbf{t}}. \tag{2.68}$$

We can write the above equation in simplified notation as

$$\mathbf{u} = W\mathbf{f} + \mathbf{b}. \tag{2.69}$$

Substituting $\mathbf{u}$ in Equation 2.63 we get our final equation which we solve using non smooth Newton method

$$F(W\mathbf{f} + \mathbf{b}, \mathbf{f}) = \mathbf{0}. \tag{2.70}$$

2.9.3 Data Driven Techniques

Chen et al. [11] use data obtained from real world measurement to augment their soft body collision model. They use mechanical setup to capture friction force to load curves for various materials. The normal force is obtained using non-interpretation constraint which is used to find the associated friction force from the data captured using measurements. However, unlike the approach described in the last section, their approach does not solve the contact assuming coupled normal and friction forces. Similar to Pabst et. al., 2009 they use a $R^{2\times2}$ matrix to describe anisotropy. Each element of the matrix is a scalar function of normal load derived from real world measurements.

In the next Chapter, we will discuss the theory and implementation details of our simulator.

Chapter 3

Implementation

We provide a detailed overview of our simulator. Our simulator setup consists of two non-rigid elastic bodies which are brought into contact with desired normal force and brushed against each other. We record various parameters during our simulation and analyze the data to compute the coefficient of friction. We start with a brief description of the fundamental principles of elastic solids and how we discretize them for implementation [38] on a digital computer. We then discuss Rayleigh damping and computation of damping parameters. Damping is important for numerical stability and for investigating its impact on coefficient friction. For simulating quasi-static dynamic friction, the case when the relative motion is small, we discuss computation of the parameters to critically damp the system to remove spurious vibrations. We already know the geometry of the contact has a profound impact on friction. We provide a recipe for generating rough surfaces and how we convert the surface to volumetric soft body. The most challenging aspect of soft body contact modeling is collision detection and resolution [11, 17]. We provide an in-depth overview of our collision detection and resolution in both 2D and 3D case. Finally, we discuss our time stepping scheme and how we factor in collision detection and resolution in the integration scheme.

3.1 Deformation Map and Deformation Gradient

In many ways, elastic solids are similar to a mass-spring system but are significantly different in terms of computing the internal forces. One can think of elastic solid deformation as a transformation map applied to each position of an undeformed solid. The transformation may or may not preserve the original geometry. In fact, the relative distance between two points on the body may be closer or farther apart after the transformation. The transformation function is called the deformation map.

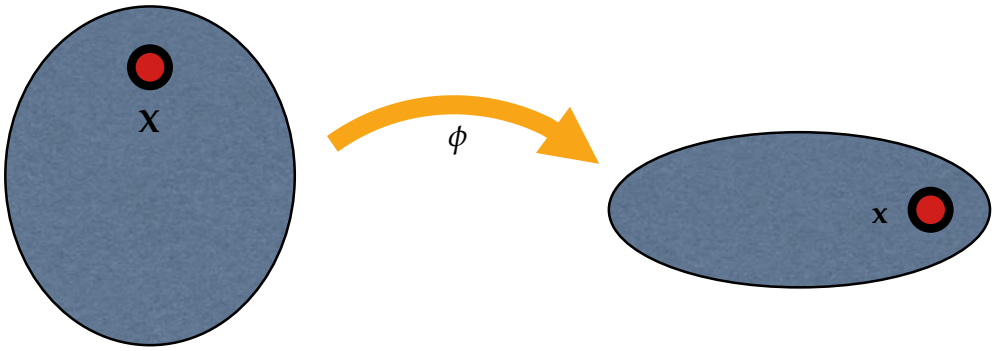

FIGURE 3.1: **Deformation map denoted by ϕ.**

A point on the undeformed shape denoted by **X** and the same point after deformation is denoted by **x**. The two are related using deformation map as

$$\mathbf{x} = \phi(\mathbf{X}). \tag{3.1}$$

It should be noted that both affine and non-affine transformations are encoded by ϕ. Usually, we want the internal energy of the soft body invariant to rotation and translation.

Deformation gradient is a very important quantity which denotes how much **x** changes when **X** is changed. Intuitively, we can think of the deformation gradient as an indicator of how much amplified or diminished the motion of the material is due to perturbation in **X**. Hence the determinant of F gives the relative change in the volume of the soft body due to deformation. Since it is a matrix, it also indicates the direction along which material deforms due to a perturbation in **X**.

$$F(\mathbf{X}) = \frac{\partial \mathbf{x}}{\partial \mathbf{X}} = \frac{\partial \phi(\mathbf{X})}{\partial \mathbf{X}}. \tag{3.2}$$

Here, $F \in R^{3\times3}$ is the deformation gradient and is a crucial ingredient for building other tensors.

3.2 Energy Density and Internal Force

A material is called hyper-elastic when its associated internal potential energy depends only on the final configuration of the material. We define an energy density function which indicates the energy stored per unit volume of the material. The energy density function is not a constant because a material may not be uniformly deformed over its entire volume. Some parts of the material may deform more and store

more energy in those parts. For hyperelastic materials, we define energy density as follows

$$\psi(\mathbf{x}) = \psi(\phi(\mathbf{X})). \tag{3.3}$$

Here, $\psi : R^3 \rightarrow R$ defines the energy stored in infinitesimal volume around $\mathbf{X}$. We now look at how deformation map and energy density are related. For small deformation, using Taylor series expansion of deformation map ϕ centered around $\mathbf{X_0}$ we have,

$$\begin{aligned} \phi(\mathbf{X}) &= \phi(\mathbf{X_0}) + \frac{\partial \phi}{\partial X}(\mathbf{X} - \mathbf{X_0}) \\ &= F(\mathbf{X_0})\mathbf{X} + \mathbf{t}. \end{aligned} \tag{3.4}$$

The constant factor $\mathbf{t}$ in Equation 3.4 is equivalent to a translation of the material and does not cause any change in potential energy of the material. In the case of rotationally invariant material, rigid rotation of the body does not result in an increase in potential energy. However, rigid rotations are embedded within deformation gradient F and it is possible to factor out the rigid rotations using Singular Value Decomposition or other matrix decomposition techniques. *Corotational linear elasticity* takes this approach. It is also possible to make ψ rotationally invariant by using $F^T F$ instead of F. Using Equation 3.4 we can say that for small deformation energy density ψ is a function of deformation gradient F and is given by $\psi(F(\mathbf{X}))$.

Now we have defined energy density and established its dependence on the deformation gradient, we would like to extract a more useful quantity, internal force. We first compute the potential energy of the configuration $\mathbf{x}$ around domain Ω as

$$E(\mathbf{x}; \Omega) = \int_{\Omega} \psi(F(\mathbf{X}))d\mathbf{X}. \tag{3.5}$$

Next we compute the force acting at position $\mathbf{x}$ due to volume Ω as

$$\mathbf{f^T} = -\frac{\partial E(\mathbf{x}; \Omega)}{\partial \mathbf{x}}. \tag{3.6}$$

In Section 3.4, we will apply Equation 3.6 to discretized soft body.

3.3 Neohookean Material

There are many different formulations of energy density ψ which conforms to different properties in an elastic material. We choose Neohookean material because it is rotationally invariant, isotropic and resistant to forcible compression. The model for

Neohookean elasticity, defined in terms of strain energy as

$$\psi(F) = \frac{\mu}{2}(I-3) - \mu \log(J) + \frac{\lambda}{2}\log^2(J), \tag{3.7}$$

where μ and λ are *Lame coefficients*, which are related to *Young's modulus* k and *Poisson ratio* ν as

$$\begin{aligned} \mu &= \frac{k}{2(1+\nu)} \\ \lambda &= \frac{k\nu}{(1+\nu)(1-2\nu)}. \end{aligned} \tag{3.8}$$

In Equation 3.8, *Young's Modulus* k is a measure of resistance to stretching while *Poisson ratio* ν is a measure of incompressibility. The quantity I and J are defined as follows

$$\begin{aligned} J &= det(F) \\ I &= trace(F^T F) \end{aligned}. \tag{3.9}$$

The $\log^2$ term in Equation 3.7 makes the material resistant to deformation. As the material is compressed, J tends to zero while log terms go towards infinity. Intuitively, J is the relative volume of the material with respect to the rest configuration. It should also be noted that Neohookean material ψ is rotationally invariant and isotropic because the quantity J and I remain unchanged if we pre or post multiply deformation gradient F with a rotation matrix. Mathematically, we define rotational invariance and isotropy as

$$\begin{aligned} &\text{Rotational invariance: } \psi(RF) = \psi(F) \\ &\text{Isotropy: } \psi(FQ) = \psi(F) \end{aligned} \tag{3.10}$$

In the equations above, matrices Q and R are rotation matrices. We call Equation 3.7 rotationally invariant because the strain energy ψ does not change due to rigid rotations defined by matrix R. Also deforming the material by equivalent amount(F) along any of its axis defined by matrix Q results in same potential energy. Hence, we call the material isotropic. It should be noted, even though theoretically it is impossible to compress the material to zero, due to numerical errors and discrete time stepping, the material might get into invalid configurations or invert itself. Thee are numerous ways to recover from inversion. Smith et al. [39] handles the issue by preventing the use of log terms in strain energy. Instead they use $-\mu(J-1) + \frac{\lambda}{2}(J-1)^2$ replacing the two log terms in Equation 3.7. It is interesting to note that the new definition is a first

order Taylor expansion of function $\log(J)$ at $J = 1$. The new energy definition allows the material to eventually return to stable state once it is inverted. In our case we prevent inversion by storing the last valid configuration and load it whenever $J \leq 0$.

3.4 Discretization

So far, we have only concerned ourselves with the continuous case, where the internal stresses vary continuously over its entire volume. In this section, we discuss the discretization of the continuous space into piecewise linear elements using tetrahedral mesh [38]. We use TetGen (3D) [37] and Triangle (2D) [36] for discretization of a hollow mesh into tetrahedrons (3D) or triangles (2D).

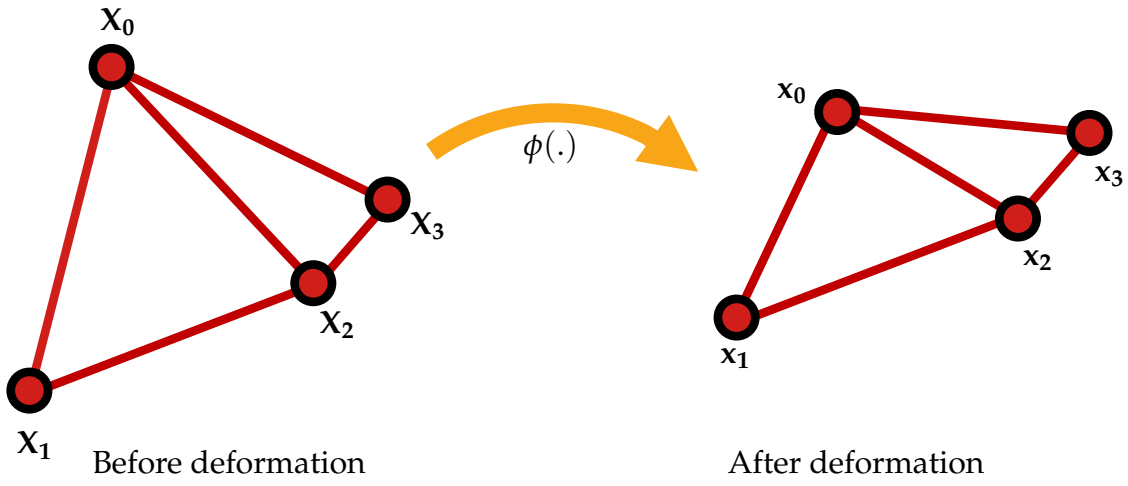

FIGURE 3.2: **Deformation of discrete elements**.

Let us first look at a single tetrahedral element. For any location $\mathbf{X}$ within this element, we can write the deformed configuration $\mathbf{x}$ using tailor series expansion centered around X^* as

$$\mathbf{x} = F(\mathbf{X}^*)\mathbf{X} + C(\mathbf{X}^*). \tag{3.11}$$

For linear tetrahedral element, we assume $F(X^*)$ and $C(X^*)$ to be constant over the volume of a tetrahedron. This introduces another approximation error but is minimized when the material is stiff or the tetrahedrons are small. Writing Equation 3.11 for the four vertices of the tetrahedron we have

$$\begin{aligned} \mathbf{x_0} &= F\mathbf{X_0} + C \\ \mathbf{x_1} &= F\mathbf{X_1} + C \\ \mathbf{x_2} &= F\mathbf{X_2} + C \\ \mathbf{x_3} &= F\mathbf{X_3} + C. \end{aligned} \tag{3.12}$$

We can rearrange Equation 3.12 and compute F in terms of deformed configuration and reference configuration as follows

$$\begin{aligned} F &= \begin{bmatrix} \mathbf{x_0} - \mathbf{x_3} & \mathbf{x_1} - \mathbf{x_3} & \mathbf{x_2} - \mathbf{x_3} \end{bmatrix} \begin{bmatrix} \mathbf{X_0} - \mathbf{X_3} & \mathbf{X_1} - \mathbf{X_3} & \mathbf{X_2} - \mathbf{X_3} \end{bmatrix}^{-1} \\ &= D_s D_m^{-1}. \end{aligned} \tag{3.13}$$

We see that matrix $D_m \in R^{3\times3}$ does not change with time as it only depends on the rest configuration. Hence should be precomputed before starting the simulation.

So far we have introduced error from three sources:

- Error due to spatial discretization.
- Error due to truncating Taylor series in Equation 3.11.
- Error due to the assumption that F and C are invariant within the tetrahedral element.

Once we have deformation gradient, we can compute energy density using Equation 3.5 as follows

$$E(\mathbf{x};\Omega) = \int_\Omega \psi(F(\mathbf{X}))d\mathbf{X} = \psi(F)\int_\Omega d\mathbf{X} = \psi(F)\,V. \tag{3.14}$$

Here, V is the volume of the domain Ω, which in case of Equation 3.14 is the volume of the tetrahedral element. Next, using Equation 3.6 we can compute the force exerted by the domain Ω on i^{th} vertex $\mathbf{x_i}, i \in 0..3$ as

$$\mathbf{f_i^T} = -\frac{\partial E(\Omega)}{\partial \mathbf{x_i}} = -V\frac{\partial \psi(F)}{\partial \mathbf{x_i}}. \tag{3.15}$$

Now let x_{ij} denote the j^{th} component of $\mathbf{x_i}$. Using chain rule for matrix functions we can write

$$f_{ij} = -V\,tr\left[\left(\frac{\partial\psi(F)}{\partial F}\right)^T \frac{\partial F}{\partial x_{ij}}\right] = -V\,tr\left[\frac{\partial\psi(F)}{\partial F}\left(\frac{\partial F}{\partial x_{ij}}\right)^T\right]. \tag{3.16}$$

In the above equation, $\partial\psi(F)/\partial F$ is also known as *Piola stress tensor* and can be derived by differentiating equation 3.7 with respect to F as

$$\frac{\partial\psi(F)}{\partial F} = \mu(F - F^{-T}) + \lambda\, log(J)\, F^{-T}. \tag{3.17}$$

Similarly, differentiating Equation 3.13 with respect to x_{ij} for $i = 0, 1, 2$, we obtain

$$\frac{\partial F}{\partial x_{ij}} = \mathbf{e_j e_i}^T D_m^{-1}. \tag{3.18}$$

Plugging the values of $\partial\psi(F)/\partial F$ and $\partial F/\partial x_{ij}$ into equation 3.16 we have

$$f_{ij} = \left[V \frac{\partial \psi(F)}{\partial F} D_m^{-T} \right]_{ij} \text{ for vertex 0,1 and 2.} \tag{3.19}$$

We can write equation 3.19 in vector form as

$$\begin{bmatrix} \mathbf{f_0} & \mathbf{f_1} & \mathbf{f_2} \end{bmatrix} = V \frac{\partial \psi(F)}{\partial F} D_m^{-T} \tag{3.20}$$

Now using the condition for static equilibrium, we have

$$\mathbf{f_3} = -(\mathbf{f_0} + \mathbf{f_1} + \mathbf{f_2}). \tag{3.21}$$

A node or vertex may be shared with multiple tetrahedrons and we need to accumulate forces from all tetrahedron sharing a node. In our implementation we loop over all tetrahedra, accumulating the force at its vertices. We also use multiple threads for computing the forces and use mutex locks to update the accumulation buffer.

3.5 Generating Surfaces

We can split surface generation into following steps - constructing a height field, subsampling, adding depth and boundary markers to the surface to form a volume, meshing and finally discretize the volume.

The first step in generating a surface is to generate a height field. There are multiple ways to generate height field - texture maps, normal maps or noise. In our case, we use noise because it is procedural and offers easier control over various parameters. We have a variety of noise algorithms to choose from. The Figure 3.3 shows different noise samples and their magnitude spectrum. All noise signals in the figure have the same energy but differ in the way the energy is distributed over its frequencies. Gabor [24] noise allows the control of parameters like orientation, power spectrum and bandwidth of the noise. By controlling these parameters it is possible to generate anisotropic surfaces similar to wood or brushed plastic.

Next step is to subsample the height field. Subsampling allows further control over the frequency spectrum of the generated noise. We use Poisson disc sampling [9] to subsample the height field because it produces tightly packed sample points while

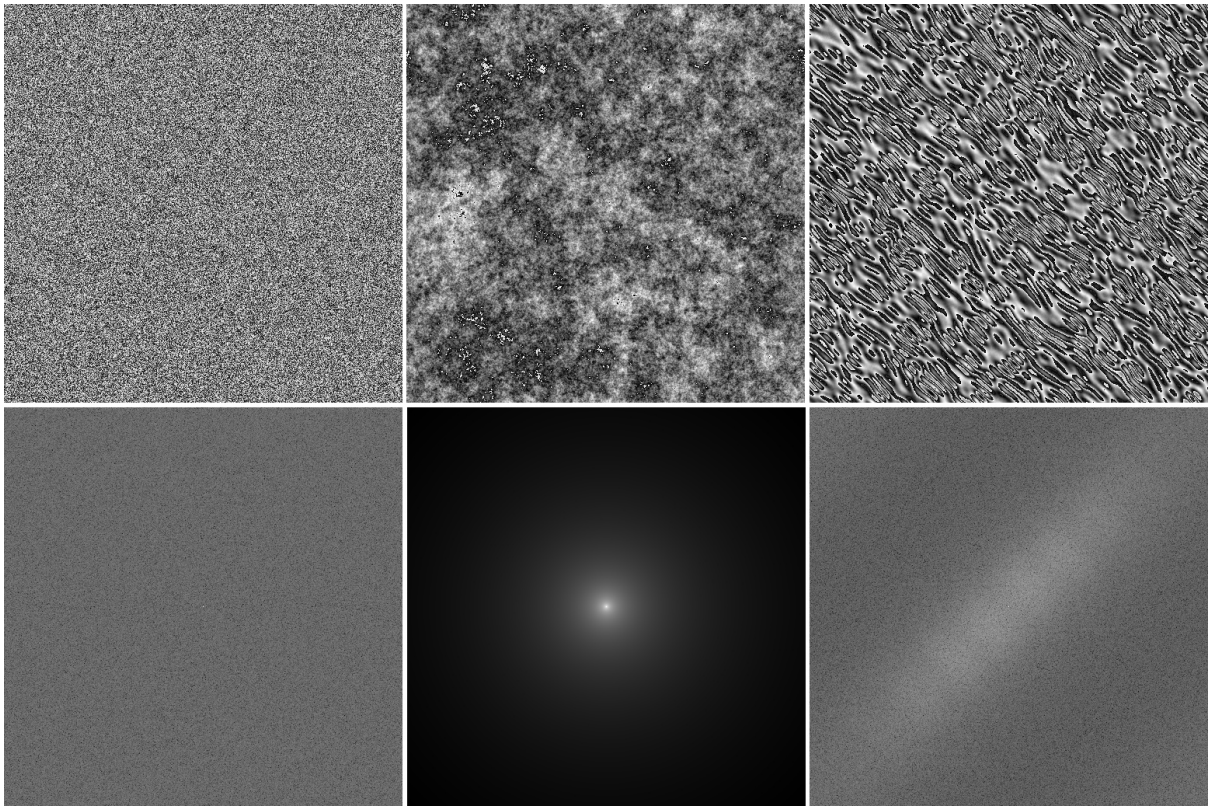

FIGURE 3.3: **Noise samples (top) and their associated frequency spectrum (bottom).See that Gabor noise shows anisotropic properties both in signal space and frequency space.**

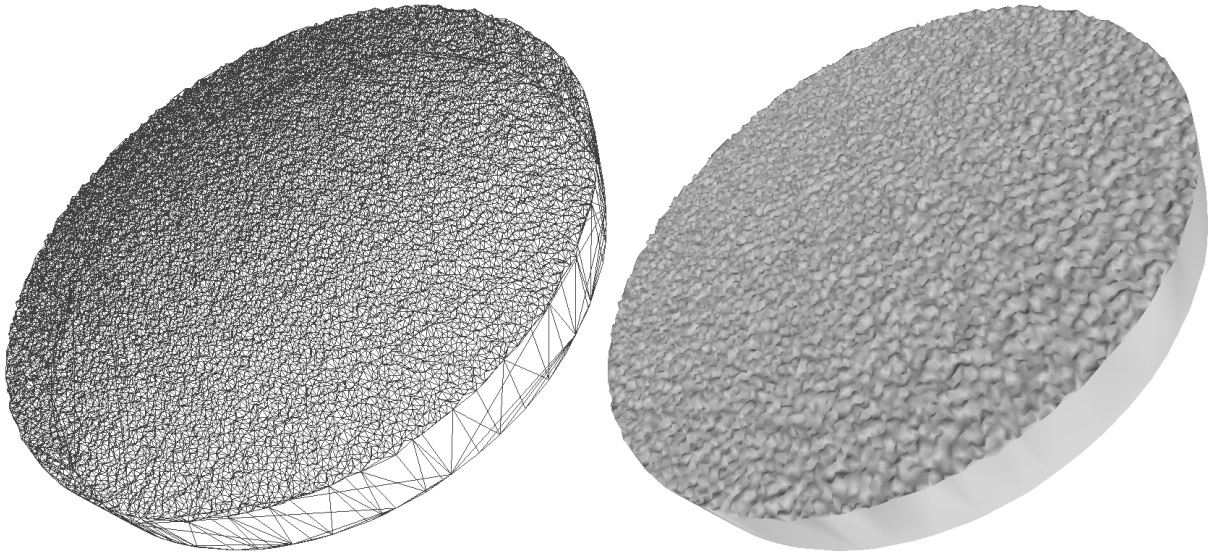

FIGURE 3.4: **Delaunay triangulation of the boundary markers to generate a hollow mesh.**

maintaining a minimum distance between them. The minimum distance is computed as

$$d_{min} = \frac{1}{2f_{max}}. \tag{3.22}$$

In the above equation, f_{max} is the maximum frequency we would like to keep in our surface. Next we add additional boundary markers to define the thickness of the material and use Delaunay triangulation to generate the boundary mesh as shown in Figure 3.4.

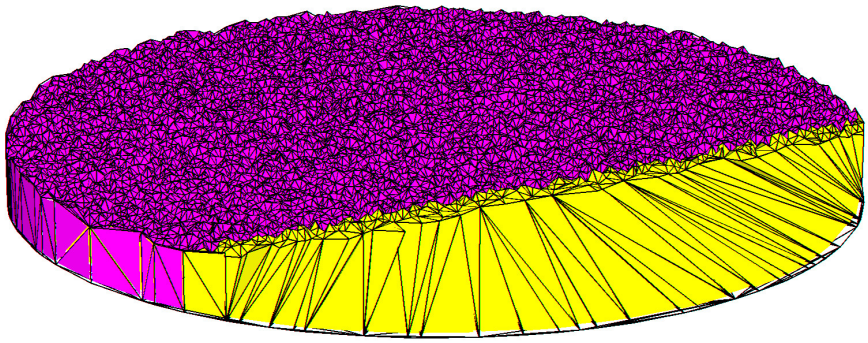

FIGURE 3.5: **Cross section of an adaptively discretized volume. Size of the tetrahedrons reduces as we go from flat base to noisy top.**

Finally as shown in Figure 3.5, we use Tetgen [37] to adaptively discretize the volume into a tetrahedral mesh. Adaptive discretization uses larger tetrahedrons in the interior of the volume while using smaller tetrahedrons near the surface. Tetgen automatically adds points in the interior of the volume, known as Steiner points to break the volume adaptively. Adaptive discretization drastically reduces the number of tetrahedrons required compared to uniform discretization.

3.6 Rayleigh Damping

So far, we have not introduced any damping effects other than numerical damping in our simulation. Damping is also important for numerical stability and we also found damping to affect friction between the materials.

Rayleigh damping model is composed of two parts, first is the mass-based damping and the second is the stiffness based damping. The net force due Rayleigh damping is as follows

$$\mathbf{f_{damp}} = (\alpha M + \beta K(\mathbf{x}))\dot{\mathbf{x}}, \tag{3.23}$$

where M is a diagonal mass matrix while $K(\mathbf{x})$ is the stiffness matrix for the soft body. Scalars, α and β are proportionality constants and we already discussed their effects in Section 2.9. We do not explicitly assemble matrix M or K but compute their effects

in a distributed fashion. It should be noted that while mass-based damping damps out the motion of the soft body as a whole while stiffness based damping damps the relative motion between the nodes. Hence stiffness based damping needs to obey conservation of momentum similar to the internal forces. For each vertex in the soft body, we compute the force due to mass-based damping as follows

$$\mathbf{f_{mass}} = m\alpha\dot{\mathbf{x}}, \tag{3.24}$$

where m is the point mass of the node, $\dot{\mathbf{x}}$ is the velocity of the node and $\mathbf{f_{mass}}$ is the force acting on the vertex due to mass-based damping.

We compute the stiffness based damping for each tetrahedron in the same way as internal forces. Looking at the equation of stiffness and stiffness based damping we have

$$\mathbf{f_{stiff}} = \beta \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \frac{d\mathbf{x}}{dt}, \tag{3.25}$$

where $\mathbf{f}$ is the net force acting on a vertex and $\mathbf{x}$ is the position of the vertex. The quantity $\frac{\partial \mathbf{f}}{\partial \mathbf{x}}$ is known as the stiffness. In our case, we cannot find stiffness analytically, but we can compute the product of stiffness and velocity by numerical means. We introduce the concept of force differential defined as

$$\delta\mathbf{f} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}} d\mathbf{x}. \tag{3.26}$$

Our aim is to directly find the force differential $\delta\mathbf{f}$ numerically. The deformation gradient F depends on the configuration $\mathbf{x}$. Following Equation 3.13, the differential of the deformation gradient is given by

$$\begin{aligned}
\delta F &= \begin{bmatrix} \delta\mathbf{x_0} - \delta\mathbf{x_3} & \delta\mathbf{x_1} - \delta\mathbf{x_3} & \delta\mathbf{x_2} - \delta\mathbf{x_3} \end{bmatrix} \begin{bmatrix} \mathbf{X_0} - \mathbf{X_3} & \mathbf{X_1} - \mathbf{X_3} & \mathbf{X_2} - \mathbf{X_3} \end{bmatrix}^{-1} \\
&= (\begin{bmatrix} \dot{\mathbf{x}}_\mathbf{0} - \dot{\mathbf{x}}_\mathbf{3} & \dot{\mathbf{x}}_\mathbf{1} - \dot{\mathbf{x}}_\mathbf{3} & \dot{\mathbf{x}}_\mathbf{2} - \dot{\mathbf{x}}_\mathbf{3} \end{bmatrix} dt) \begin{bmatrix} \mathbf{X_0} - \mathbf{X_3} & \mathbf{X_1} - \mathbf{X_3} & \mathbf{X_2} - \mathbf{X_3} \end{bmatrix}^{-1} \\
&= \delta D_s D_m^{-1},
\end{aligned} \tag{3.27}$$

where $\mathbf{x_i}$ are the vertices of the tetrahedron and $\dot{\mathbf{x}}_\mathbf{i}$ are the velocities of the corresponding vertices. We compute δD_s from the difference of the nodal velocities and multiplying with time step. Next, we compute the differential of Piola stress tensor derived from Equation 3.17 and is defined as

$$\delta \frac{\partial \psi(F)}{\partial F} = \mu\, \delta F + [\mu - \lambda\, log(J)] F^{-T} \delta F^T F^{-T} + \lambda\, tr(F^{-1}\delta F) F^{-T}. \tag{3.28}$$

We plug in the differential of the deformation gradient from Equation 3.27 and

evaluate the above equation. Similar to Equation 3.20, we can write force differential in compact form as

$$\begin{bmatrix} \delta \mathbf{f_0} & \delta \mathbf{f_1} & \delta \mathbf{f_2} \end{bmatrix} = V \, \delta \frac{\partial \psi(F)}{\partial F} D_m^{-T}. \tag{3.29}$$

Since the differential forces are internal forces, the forces must balance. We obtain the force differential for the fourth vertex as

$$\delta \mathbf{f_3} = -(\delta \mathbf{f_1} + \delta \mathbf{f_2} + \delta \mathbf{f_3}). \tag{3.30}$$

Finally, we compute stiffness damping force as

$$\mathbf{f_{stiff}} = \beta \frac{\delta \mathbf{f_i}}{dt} \text{ for } i \in [0, 3]. \tag{3.31}$$

Once we have the damping forces for all four vertices of the tetrahedron, we accumulate these forces in a similar manner to the internal forces.

3.7 Computing Stiffness Matrix (K)

We do not need to compute the stiffness matrix explicitly for our simulation, we need to find the matrix K for the purpose of computing critical damping factor. We use the standard technique called unit displacement method. Looking at Equation 3.26, if we assume a 1D system and set $dx = 1$ and $dt = 1$, then the force differential is same as stiffness. In other words, computing the force differential using unit velocity is equivalent to computing stiffness. We extend the same idea to general n-dimensional case by assuming the velocity a unit vector in n-dimension. To get the k^{th} column of stiffness matrix, we set the velocity of the vertices in the soft body as

$$\dot{\mathbf{x}} = \mathbf{e_k} \; s.t. \; \mathbf{e_k} \in R^{3n} \text{ and } k \in [0, 3n-1] \tag{3.32}$$

In the above equation, n is the number of vertices in 3D. Using the new velocities and setting dt and β to 1, we compute the Equation 3.31 for all tetrahedra and accumulate the forces on each node. These nodal forces, when concatenated into a vector, form the k^{th} column of the stiffness matrix. We repeat the step for all $3n$ columns.

3.8 Computing Critical Damping

Our motivation for computing critical damping is to make our system quasi-static. For experimenting with static friction and the transition to kinetic friction, we do not want large particle velocities in our system. We can control the average motion of the

soft body by controlling the applied external forces, but individual particles remain beyond explicit control. Hence, we try to control the individual particle velocity by critically damping the soft body.

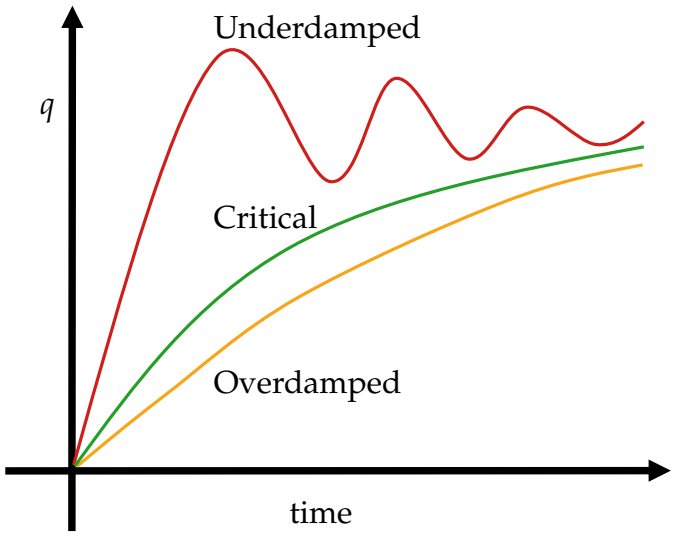

FIGURE 3.6: **System response under different level of damping.**

For any soft body system, the constituting vertices can move in some directions more than the others. In most cases, the system has few main modes along which body's motion is concentrated. Like a mass-spring system, elastic materials also exhibit periodic oscillations or vibrations along these modes. Critical damping is the minimum damping required to eliminate all vibration along a mode. See Figure 3.6. In our case, we use critical damping as our initial set point for the parameters α and β.

We first look at the natural motion of the body. If we imagine plucking a guitar string, the motion of the string after excitation is its natural motion. Hence, the natural motion of the body is defined as the motion of the body without any external forces and is given as

$$M\ddot{\mathbf{x}} + (\alpha M + \beta K(\mathbf{x}))\dot{\mathbf{x}} + \mathbf{f_{int}}(\mathbf{x}) = \mathbf{0}, \tag{3.33}$$

where we can think of $\mathbf{x}$ as the configuration immediately after plucking the string. We should compute K around this excited configuration but in practice, we only compute K around the reference position. This is a good approximation for stiff rigid bodies which do not produce much deformation. We also replace $\mathbf{f_{int}}(\mathbf{x})$ by approximation $K\mathbf{x}$. Note that these simplifying approximations are only used for computing the critical damping factor.

$$\ddot{\mathbf{x}} + (\alpha + \beta M^{-1}K)\dot{\mathbf{x}} + M^{-1}K\mathbf{x} = \mathbf{0}. \tag{3.34}$$

Since, K is not necessarily diagonal, it is not be possible to decouple Equation 3.34 into multiple independent differential equations directly. Hence, we use eigenvalue decomposition of $M^{-1}K$ to decouple Equation 3.34. Matrix $M^{-1}K$ is symmetric positive indefinite since K is symmetric positive indefinite around reference position and M is a diagonal matrix [38]. Using eigenvalue decomposition of $M^{-1}K$, we have

$$M^{-1}K = U\Lambda U^T. \tag{3.35}$$

The matrix Λ is a diagonal matrix of eigenvalues. Each column of matrix U represents an eigenvector and its corresponding eigenvalue along the diagonal of Λ. Eigenvalues for positive indefinite matrices are real. We may have six eigenvalues which are zero magnitude that corresponds to rigid translations and rotations. Pre-multiplying Equation 3.34 by U^T we obtain

$$\ddot{\mathbf{q}} + (\alpha + \beta\Lambda)\dot{\mathbf{q}} + \Lambda\mathbf{q} = \mathbf{0}. \tag{3.36}$$

In the above equation, we substituted $U^T\mathbf{x}$ as $\mathbf{q}$. Since Λ is the diagonal matrix, we can write the above equation as multiple independent differential equations. In our case, we compute α and β for the largest eigenvalue denoted by Λ_0. Hence we can write the one dimensional differential equation as

$$\ddot{q}_0 + (\alpha + \beta\,\Lambda_0)\dot{q}_0 + \Lambda_0\, q_0 = 0. \tag{3.37}$$

Studying the characteristic equation for the above differential equation, we obtain

$$\alpha + \beta\,\Lambda_0 = 2\sqrt{\Lambda_0}. \tag{3.38}$$

The values of α and β that satisfies the above equation produce critically damped motion in the highest frequency mode. We can make other lower frequency modes over damped if α and β satisfy the following additional constraints

$$\alpha \geq \frac{2(\sqrt{\Lambda_0}\Lambda_1 - \sqrt{\Lambda_1}\Lambda_0)}{\Lambda_1 - \Lambda_0} \text{ and } \beta \leq \frac{2(\sqrt{\Lambda_0} - \sqrt{\Lambda_1})}{\Lambda_0 - \Lambda_1}, \tag{3.39}$$

where Λ_1 is the second largest eigenvalue.

3.9 Soft Body Collision

Collision detection and resolution [27] for soft bodies with arbitrary geometry is the most computationally expensive step in our simulation. It accounts for around 50 percent of compute time in the 2D case and more than 70 percent in 3D. Using continuous collision detection is prohibitively expensive because of the large number of triangles (in 2D) or tetrahedra (in 3D). Hence we use spatial hashing with penalty forces [17, 25] to detect and resolve collisions.

3.9.1 Spatial Hashing

We use a spatial hash structure to accelerate the collision detection process. Spatial hash is a hash table where the key is position in space and the table itself represents the space as a uniform grid. We choose the cell size of the hash table such that the repulsive forces diminish nearly to zero after one cell distance. This minimizes the discontinuity in the force field for a particle moving in or out of a cell. At each time step, the particles are hashed into cells. We loop over each edge (in 2D) or tetrahedron (in 3D) and perform a lookup into the table to query the neighborhood particles. The pair obtained is passed for collision resolution using repulsive forces.

3.9.2 Penalty Forces in 2D

In the 2D case, we implement edge-particle and particle-particle collision using repulsive forces.

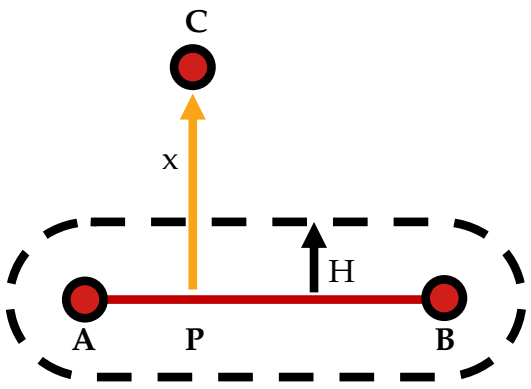

FIGURE 3.7: **Modeling edge-particle collision. Position P is the projection of AC on AB. We do not allow the particle C to come any closer than distance H from the edge AB.**

In the above figure, scalar H is the minimum allowed distance between an edge and a particle. At the end of each time step, no particle should be inside the region bounded by the dashed capsule in Figure 3.7. Scalar x is the distance between an edge and a particle. We define x as follows

$$\alpha = \frac{\mathbf{AC} \cdot \mathbf{AB}}{||\mathbf{AC}||\,||\mathbf{AB}||} \tag{3.40}$$

$$x = \begin{cases} ||\mathbf{AC} - \alpha\mathbf{AB}|| & \text{if } 0 \leq \alpha \leq 1 \\ ||\mathbf{AC}|| & \text{if } \alpha < 0 \\ ||\mathbf{BC}|| & \text{otherwise} \end{cases} . \tag{3.41}$$

The magnitude of repulsive forces are given by

$$f_{rep} = \begin{cases} \frac{K}{(x-H)^2} & if\ x \geq H \\ \frac{K}{x^2} & if\ x < H \end{cases} \tag{3.42}$$

We apply the principle of conservation of energy and momentum to solve for collision impulses. Our algorithm first computes the work done by the force field due to the relative velocity between edge and particle in one time step. Assuming the edge as a point mass concentrated at position P (see Figure 3.7) and using conservation of energy and momentum, we compute the impulse velocity needed to resolve collision for the particle and simplified edge. Finally, we distribute the collision impulse for the simplified edge among its constituent two particles such that the energy of the system does not increase.

Overall algorithm for collision detection and response is given by Algorithm 1. There are three functions which we did not define inside the body of algorithm - *killNormalVelocity()*, *solve()* and *distribute()*.

We compute the effective velocity of the edge by weighing the velocities of the two vertices on the edge by α obtained using Equation 3.40. Note that α is unrelated to the coefficient of mass based damping in Rayleigh damping model. We compute the effective mass as

$$\frac{1}{m_{edge}} = \frac{1-\alpha}{m_A} + \frac{\alpha}{m_B} \tag{3.43}$$

The direction of normal is given by the direction of x obtained from Equation 3.41. We compute the relative normal speed between the edge and particle as follows

$$v_{rel} = (\mathbf{v_C} - \mathbf{v_{edge}}) \cdot \mathbf{n} \tag{3.44}$$

Algorithm 1: Collision detection and resolution algorithm.

Input: v_{rel}, x, dt
Output: Collision impulses along normal: Δv_C Δv_A Δv_B
$dx = v_{rel}\ dt$;
`// kill relative normal velocity between edge and particle`
flag = false;
if $v_{rel} < 0$ *and* $x + dx < H$ **then**
  flag = true;
`// Find work done by the force field` ΔE
if $x < H$ **then**
  $\Delta E = \frac{K\,dx}{x(x+dx)}$
else
  $\Delta E = \frac{K\,dx}{(x-H)(x+dx-H)}$
$limit = -\frac{v_{rel}^2}{2(m_{edge}^{-1}+m_C^{-1})}$;
`// System of equations are inconsistent`
if $\Delta E < limit$ **then**
  flag = true;
if *flag is true* **then**
  Δv_C, Δv_{edge} = killNormalVelocity();
else
  Δv_C, Δv_{edge} = solve(ΔE, v_{rel});
Δv_A Δv_B = distribute(Δv_{edge});
return Δv_C, Δv_A Δv_B;

The first function *killNormalVelocity()* is used when the time step is too large or when the system of equations is unsolvable. The purpose of this function is to force v_{rel} to zero at the end of time step. We solve the following equations for Δv_c and Δv_{edge}.

$$v_{rel} + \Delta v_c - \Delta v_{edge} = 0, \text{ and} \tag{3.45}$$

$$m_C\,\Delta v_c + m_{edge}\,\Delta v_{edge} = 0. \tag{3.46}$$

Equation 3.45 corresponds to the fact that we want zero relative normal velocity at the end of the time step whereas Equation 3.46 is conservation of momentum between edge and particle.

The second function *solve()* is used when the system of equations are consistent and to solve for Δv_c and Δv_{edge} using energy and momentum conservation. Energy conservation is given by ΔE = *Change in Kinetic Energy* and momentum conservation

is the second Equation in 3.47.

$$\begin{aligned} &\Delta E = 0.5\, m_C\, \Delta v_c (2\mathbf{v_C} \cdot \mathbf{n} + \Delta v_c) + 0.5\, m_{edge}\, \Delta v_{edge} (2\mathbf{v_{edge}} \cdot \mathbf{n} + \Delta v_{edge}) \ , \\ &m_C\, \Delta v_c + m_{edge}\, \Delta v_{edge} = 0. \end{aligned} \tag{3.47}$$

Energy change of the edge-particle pair due to the force field is given by ΔE and is computed by integrating Equation 3.42 from 0 to $v_{rel}\, dt$. The exact expression for ΔE is provided inside the body of Algorithm 1. It should be noted that Equation 3.47 has two solutions. We choose the one that increases the relative normal velocity, that is Δv_c - $\Delta v_{edge} > 0$.

The last function *distribute()* is used to divide the impulse $m_{edge}\, \Delta v_{edge}$ among its constituents $m_A\, \Delta v_A$ and $m_B\, \Delta v_B$. We distribute the impulses such that

$$m_{edge}\, \Delta v_{edge}^2 \geq m_A\, \Delta v_A^2 + m_B\, \Delta v_B^2. \tag{3.48}$$

We found solving for the following simple heuristic to work well

$$\begin{aligned} \Delta v_{edge} &= (1-\alpha)\Delta v_A + \alpha \Delta v_B \ \ and \\ \Delta v_A &= \begin{cases} 0 & if \alpha \geq 0.5 \ . \\ \Delta v_{edge}/(1-\alpha) & otherwise \end{cases} \end{aligned} \tag{3.49}$$

Our algorithm is robust and interpenetration free even for large time steps. Since collisions are solved analytically, it is also faster compared to continuous collision detection. The only drawback is we need to tune the stiffness parameter K to make the collisions more accurate and stable.

3.9.3 Penalty Forces in 3D

We have a vastly greater number of tetrahedrons and order of magnitude more contacts at each time step in 3D compared to the 2D case. It is not possible to consider face-particle, edge-edge, edge-particle and particle-particle collisions for so many contacts in a meaningful time budget for each frame. Hence, we use a simpler strategy and make use of SIMD intrinsic for better scalability with the number of contacts.

We first sample points on the boundary tetrahedra of the two interacting surfaces. These points serve as the nodes of interaction between two adjacent tetrahedra belonging to the two surfaces. The repulsive forces act between the points sampled as shown in Figure 3.8. We use sampling density as our parameter to control the density of points on the surface. For each sampled point, we keep track of the index of the vertex closest to the sampled point and the index of the tetrahedron on which the

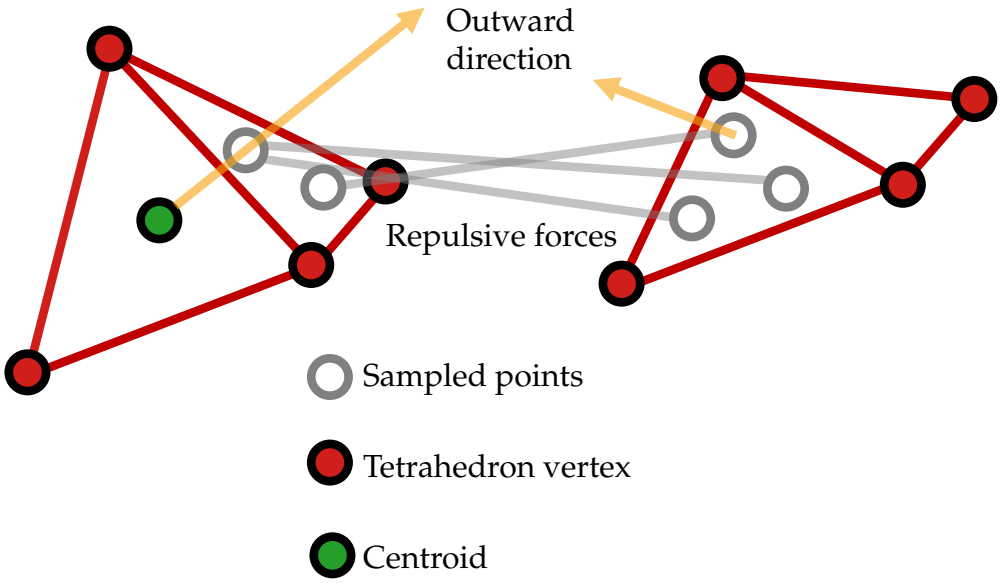

FIGURE 3.8: **Modeling tetrahedron-tetrahedron collision. The repulsive forces act between the sampled points on the tetrahedron.**

point was sampled. The sampled points are stored in barycentric coordinates and converted to world coordinates during simulation. These steps are performed only once at initialization.

Inside the simulation loop, we compute the repulsive force between each pair of sampled points. We also compute the outward direction as the difference between the centroid and the sampled point as shown by the yellow vector in Figure 3.8. The forces between two points are non-zero only if their outward directions are facing each other. Since we only have two soft bodies in our system, we further optimize by building the hash table only for the sampled positions in one of the two soft bodies. Doing so also eliminates the possibility of a self-collision.

Next, we define the repulsive force between two sampled points.

$$H = \min(H_1, H_2) \tag{3.50}$$

In the above equation, H_1 and H_2 are the size of the two tetrahedra $\mathfrak{T}_1$ and $\mathfrak{T}_2$ measured as the distance of the centroid from its vertices. The dashed edge in Figure 3.9 represents the size of a tetrahedron.

$$\mathbf{f_{far}} = \begin{cases} \mathbf{0} & if\ ||\mathbf{x_p}||_2 \leq H \\ \mathbf{0} & if\ \mathbf{d_1} \cdot \mathbf{d_2} \geq 0 \\ \frac{K_1\,\mathbf{x}}{||\mathbf{x_p}||_2^3} & otherwise \end{cases} \tag{3.51}$$

In the above equation, $\mathbf{x_p}$ is the vector between two sampled positions. Vectors $\mathbf{d_1}$

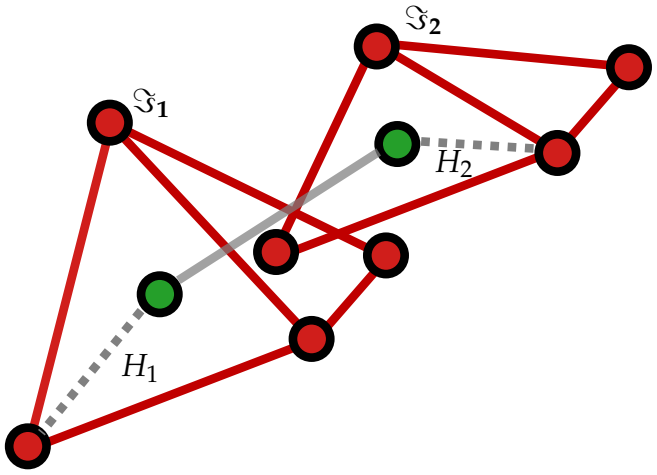

FIGURE 3.9: **Modeling repulsion under close proximity.** H_1 **and** H_2 **represents the distance of the centroid from corresponding vertex.**

and $\mathbf{d_2}$ are the outward direction at the sampled positions. The force $\mathbf{f_{far}}$ is applied to the tetrahedron vertex closest to the sampled position.

Finally, to minimize interpenetration, we use a strong repulsive force acting between the centroids of two tetrahedra. This force is in addition to the forces between the sampled positions. The additional force is distributed equally among the vertices of the tetrahedron, however, is non-zero only when the centroids of two tetrahedra are closer than H.

$$\mathbf{f_{near}} = \begin{cases} \mathbf{0} & if\ ||\mathbf{x_c}||_2 \geq H \\ \frac{K_2\,\mathbf{x_c}}{||\mathbf{x_c}||_2^3} & otherwise \end{cases} \tag{3.52}$$

In the above equation, $\mathbf{x_c}$ is the vector between two centroids of the tetrahedron. When two tetrahedrons are already interpenetrating, $\mathbf{f_{far}}$ pushes them further into each other. Hence, when $\mathbf{f_{near}}$ is non-zero, $\mathbf{f_{far}}$ must be zero. We also make the stiffness K_2 very large so that the centroid of two tetrahedrons cannot cross each other easily and get into an invalid configuration we cannot recover from.

3.10 Time Stepping

We use the Symplectic Euler integration scheme. Symplectic Euler allows for natural integration of collision impulses in the integration step and is computationally inexpensive compared to Backwards Euler. Backwards Euler is more suitable for cases where we require large time steps or extremely stiff systems. As we are aiming for accuracy rather than large time steps, Symplectic Euler makes more sense.

In the 2D case, for each particle, we first compute the internal forces and accumulate all other external forces like gravity and constraint forces required for pressing and rubbing the soft bodies against each other. We compute the change in velocities as

$$\Delta \mathbf{v} = M^{-1}(\mathbf{f_{int}} + \mathbf{f_{ext}})\, dt. \tag{3.53}$$

We add the collision impulses and step again to obtain the positions as

$$\Delta \mathbf{x} = (\Delta \mathbf{v} + \Delta \mathbf{v_{collision}})\, dt. \tag{3.54}$$

In the 3D case, we compute the collision forces instead of collision impulses and update the position and velocities of each particle as follows

$$\Delta \mathbf{v} = M^{-1}(\mathbf{f_{int}} + \mathbf{f_{ext}} + \mathbf{f_{collision}})\, dt \tag{3.55}$$

$$\Delta \mathbf{x} = \Delta \mathbf{v}\, dt. \tag{3.56}$$

3.11 Summary

We briefly described the theory behind elastic solids, their discretization, and implementation. We first derived the equation describing the internal forces inside an elastic material in continuous space and then discretized the continuous space assuming linear tetrahedral elements. We also investigated the generation of rough tetrahedral meshes for our simulation. We discussed the Rayleigh damping model and technique for integrating damping into our framework using force differentials. We provided a recipe for building a full stiffness matrix and computation of critical damping factor α and β. We discussed collision detection and resolution in both 2D and 3D case and how we merge them with our integration scheme. In the next chapter, we look at the effects of varying various elastic parameters in our system on the coefficient of friction.

Chapter 4

Results

We explore various factors affecting friction and discuss the results we obtain from our simulator in 2D and 3D case. We discuss the parameter setup of our simulator and various factors affecting the results. We simulate static friction in the 2D case and both static and dynamic friction in the 3D case and collect data from real world measurement for qualitative validation of results obtained.

4.1 Friction in 2D

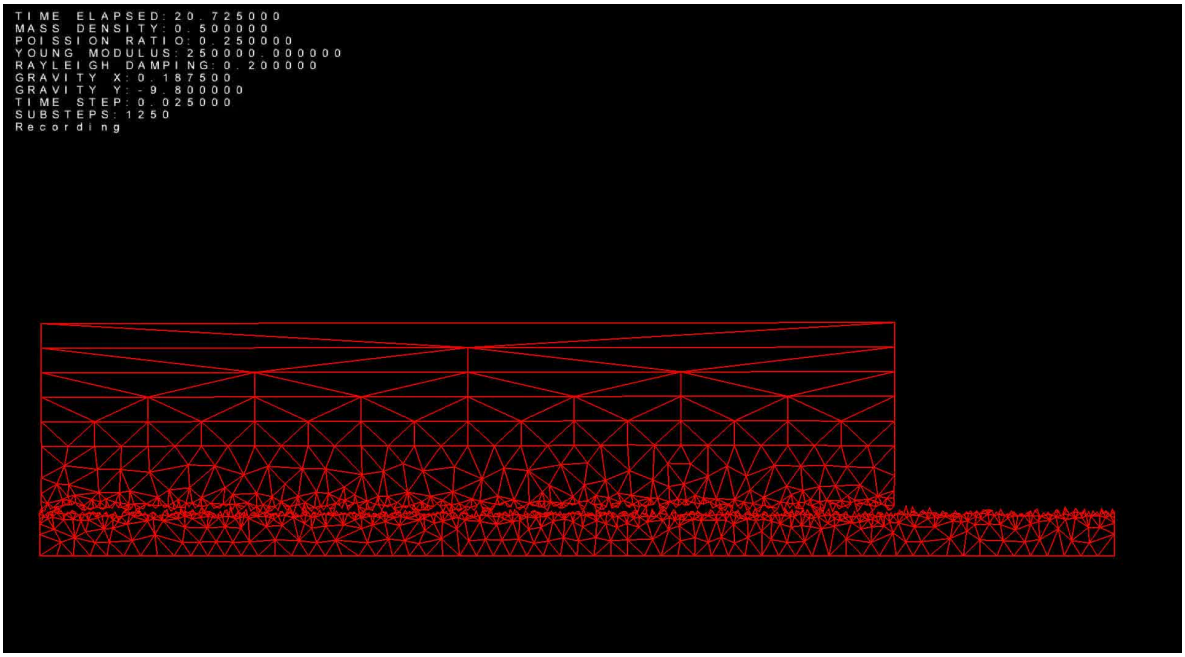

FIGURE 4.1: **An output frame from our 2D simulator.**

In the 2D case, the number of samples per unit length for generating the height field is 2 per unit length and the total number of triangles in the system is around 8000. The height field is generated using pink noise while the standard deviation of the height field is 5 units. For reference, the length of the longer soft body in Figure 4.1 is 1500 units. We also use a hash table of size 2×2 unit for collision detection.

4.1.1 Methodology

To simulate friction between two soft bodies, we make one of the soft body stationary while we drag the other over it. In our case, we simply make the lower soft body stationary by pinning the particles at the lowermost layer. We slowly increase the horizontal force on the upper body until it starts sliding. We define the coefficient of static friction as the ratio between horizontal and vertical force at the moment when the upper body just starts to slide. The vertical force in our case is gravity. We run each test twelve times using different surfaces drawn from the same distribution and average the results to get the friction coefficient for one set of parameters. In 2D case, we derive our surface samples from pink noise with fixed spectral energy. The time required for each individual simulation depends on the stiffness of the material and is between 8 hours to 24 hours on average using a single core with AVX instructions. We run multiple instances of our simulator on a cluster.

4.1.2 Results

We test the effect of varying elastic parameters like Young's modulus, Poisson ratio, Rayleigh damping (stiffness/β damping) and mass density on static friction coefficient. The plots in Figure 4.2 and 4.3 has been obtained from data tabulated in Table 4.1.

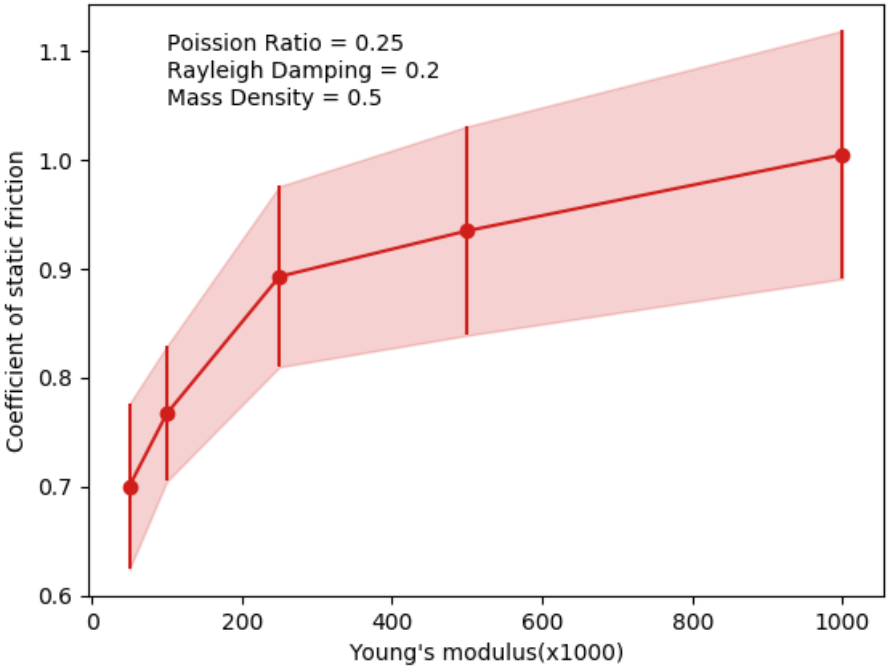

FIGURE 4.2A: **Variation in the coefficient of static friction with Young's modulus. The shaded tube indicates the standard deviation in measurement at the point of observation.**

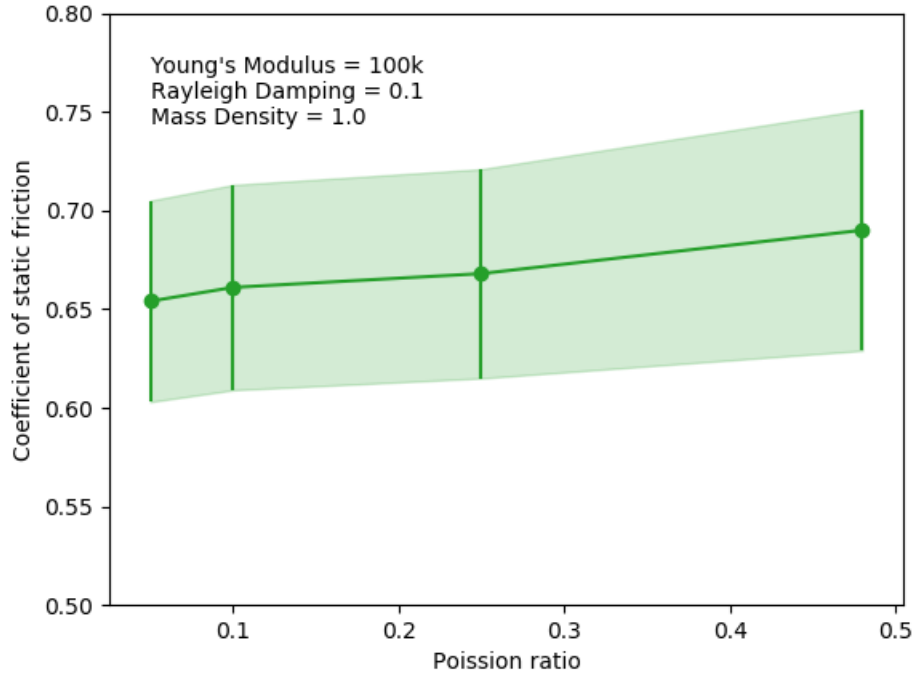

FIGURE 4.2B: **Variation in the coefficient of static friction with Poisson ratio with associated error bound.**

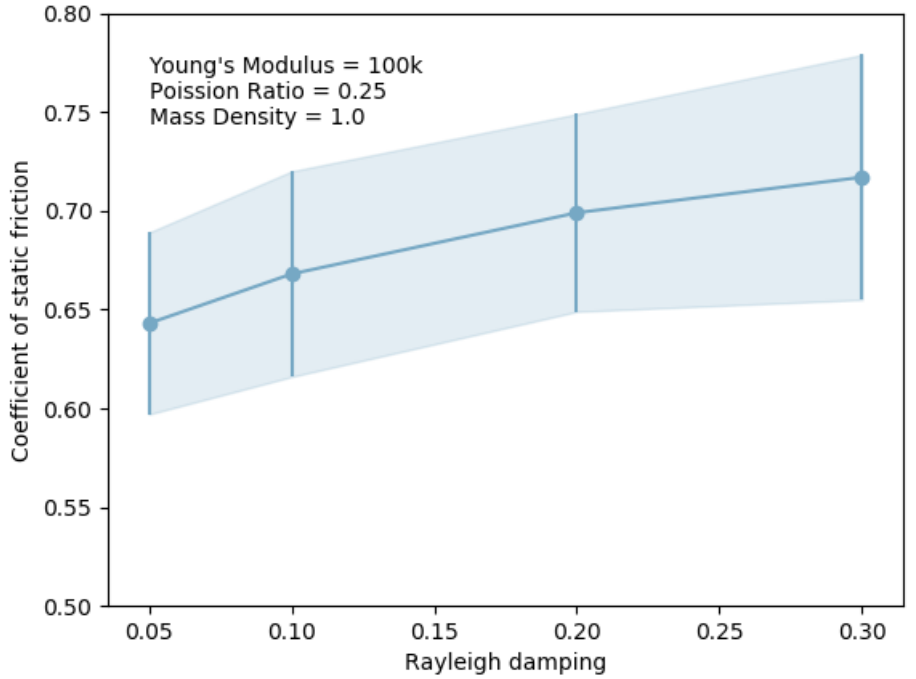

FIGURE 4.2C: **Variation in the coefficient of static friction with Rayleigh damping with associated error bound.**

From figure 4.2 and 4.3, we see that coefficient of static friction is most correlated with Young's modulus and mass density. Poisson ratio has the least effect on the friction coefficient followed by Rayleigh damping. Increasing Young's modulus is equivalent to making the system stiffer and it is difficult for the interlocking triangles on the

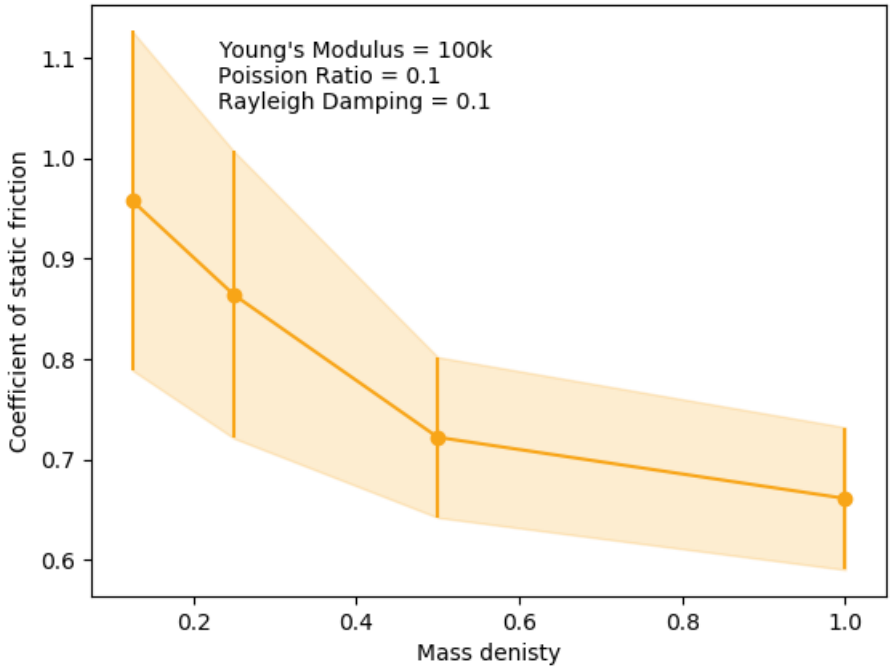

FIGURE 4.2D: **Variation in the coefficient of static friction with mass density with associated error bound.**

surface to deform and slide over each other. This results in an increased coefficient of friction. Counterintuitively, however, increasing mass density reduces the coefficient of friction. The reason we think is that increasing mass density causes increased normal forces which deform or smooth out the rough contact. This causes the coefficient of friction to reduce as relatively lower tangential forces are needed to slide the object due to the smoothing effect.

We also observe that the static coefficient of friction increases marginally with the Rayleigh damping parameter β. In the ideal case, the coefficient of static friction should be independent of Rayleigh damping because the bodies are stationary and have no velocity. However, in practice, just before the soft body starts to move, the vertices gain small velocity causing some additional resistance in the motion of the particles due to damping. The coefficient of static friction is weakly influenced by a change in Poisson ratio. Increasing Poisson ratio increases the resistance to change in volume. In most cases, a change in volume is negligible and so is the resistant force due to volume change. Hence, we only see a small increase in the coefficient of static friction due to increase in Poisson ratio.

In Figure 4.2, we show the error margin in our estimate of friction coefficient. The uncertainty is primarily because of the small number of sample surfaces we draw from our noise distribution. The results were obtained using 12 samples for each data point and increasing the number of samples should result in a lower variance. The second issue is that increasing Young's modulus, Poisson ratio causes the system to become

Young's modulus (E)	Poisson's ratio (ν)	Rayleigh damping (β)	Mass density (ρ)	μ_{mean}	μ_{std}
50 000	0.25	0.2	0.5	0.700	0.076
100 000	0.25	0.2	0.5	0.767	0.062
250 000	0.25	0.2	0.5	0.893	0.093
500 000	0.25	0.2	0.5	0.935	0.110
1000 000	0.25	0.2	0.5	1.005	0.134
100 000	**0.48**	0.1	0.25	0.892	0.150
100 000	**0.25**	0.1	0.25	0.875	0.155
100 000	**0.1**	0.1	0.25	0.864	0.163
100 000	**0.05**	0.1	0.25	0.858	0.159
100 000	**0.48**	0.1	0.5	0.790	0.106
100 000	**0.25**	0.1	0.5	0.751	0.089
100 000	**0.1**	0.1	0.5	0.722	0.080
100 000	**0.05**	0.1	0.5	0.701	0.073
100 000	**0.48**	0.1	1.0	0.690	0.061
100 000	**0.25**	0.1	1.0	0.668	0.053
100 000	**0.1**	0.1	1.0	0.661	0.071
100 000	**0.05**	0.1	1.0	0.654	0.051
100 000	0.25	**0.3**	0.25	0.933	0.144
100 000	0.25	**0.2**	0.25	0.914	0.168
100 000	0.25	**0.1**	0.25	0.875	0.155
100 000	0.25	**0.05**	0.25	0.842	0.185
100 000	0.25	**0.3**	0.5	0.776	0.080
100 000	0.25	**0.2**	0.5	0.767	0.062
100 000	0.25	**0.1**	0.5	0.751	0.071
100 000	0.25	**0.05**	0.5	0.743	0.087
100 000	0.25	**0.3**	1.0	0.717	0.090
100 000	0.25	**0.2**	1.0	0.699	0.044
100 000	0.25	**0.1**	1.0	0.668	0.052
100 000	0.25	**0.05**	1.0	0.643	0.066
100 000	0.48	0.1	**0.125**	0.983	0.189
100 000	0.48	0.1	**0.25**	0.892	0.150
100 000	0.48	0.1	**0.5**	0.790	0.106
100 000	0.48	0.1	**1.0**	0.690	0.061
100 000	0.25	0.1	**0.125**	0.969	0.176
100 000	0.25	0.1	**0.25**	0.875	0.155
100 000	0.25	0.1	**0.5**	0.751	0.089
100 000	0.25	0.1	**1.0**	0.668	0.053
100 000	0.1	0.1	**0.125**	0.958	0.169
100 000	0.1	0.1	**0.25**	0.864	0.163
100 000	0.1	0.1	**0.5**	0.722	0.080
100 000	0.1	0.1	**1.0**	0.661	0.071

TABLE 4.1: **Table of static friction coefficient (μ_{mean}) and the reliability of measurement (μ_{std}) in the 2D case for varying soft body parameters.**

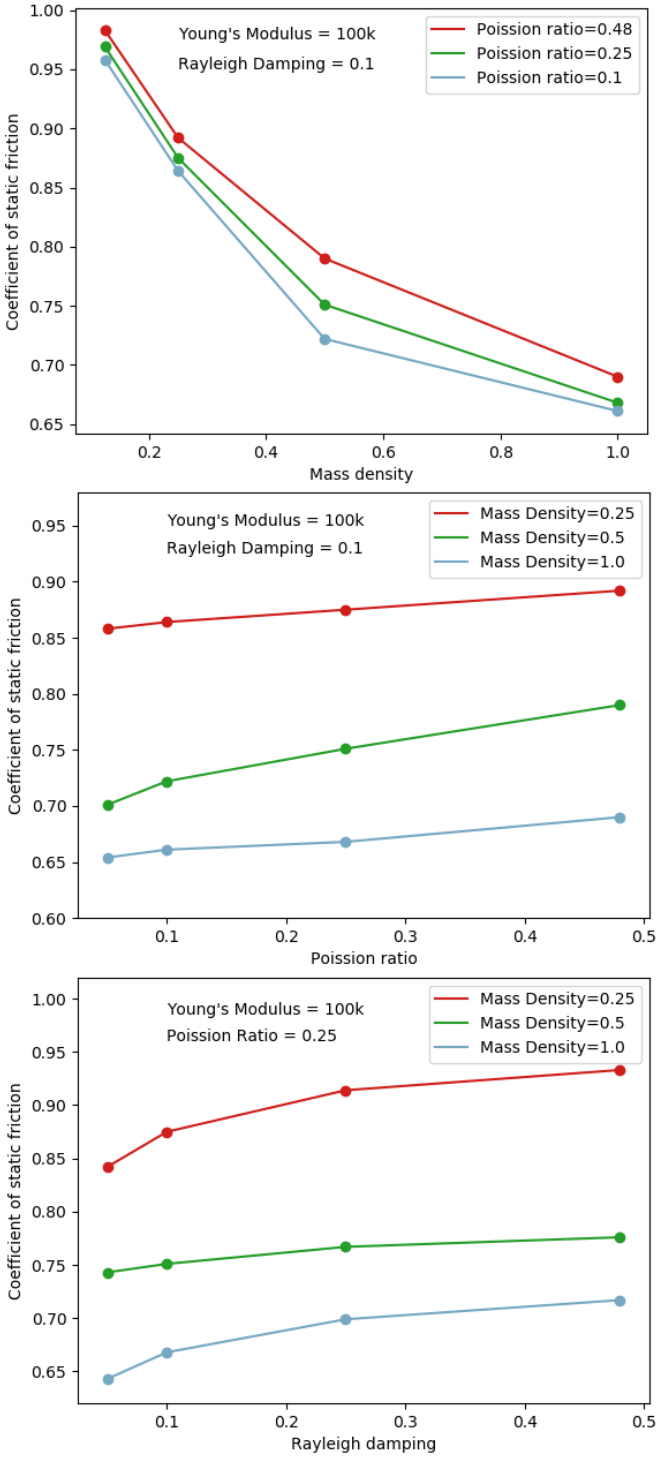

FIGURE 4.3: **Variation in the coefficient of static friction with elastic parameters.**

stiffer and higher number of time steps are required to attain stability. Lower mass density also has a destabilizing effect. The stiffness of an ODE is measured by the ratio of maximum and minimum eigenvalue of the mass-weighted stiffness matrix i.e. $M^{-1}K$. The mass of a particle depends on the size of triangle and density. So, using smaller triangles or smaller density should also make the system stiffer. We think,

increasing the time steps further should lower the instabilities in the system and result in lower error margin.

4.2 Friction in 3D

We investigate dynamic and static friction in the 3D case. The setup for both static and dynamic friction is similar but differs in the way we measure the two coefficients. We try to simulate metal like material with Young Modulus around 100 GPa, mass density of 3000 kg/m^3, Poisson ratio of 0.25. We use uniform noise to generate our surfaces and sample the surface with a pitch of 0.5 cm. So the areal density of sample points on the surface is 4 samples/cm^2. The Rayleigh damping parameters are set to critically damp the largest modes in the system. The larger of the two soft bodies has a radius of 31 cm while the smaller has a radius of 18 cm. The standard deviation of the noise height field is 0.28 cm. The total number of tetrahedrons in the system is around 60 thousand. The total number of tetrahedron - tetrahedron contact pair hovers around 40 thousand with a hash table of size of 1 cm $\times$ 1 cm $\times$ 10 cm. The $z - height$ of the hash table doesn't influence the performance since we limit the collisions to the tetrahedrons on the interacting surface. It takes around 10 to 15 seconds of wall time on a 4 core machine to simulate a time step of 4 μs.

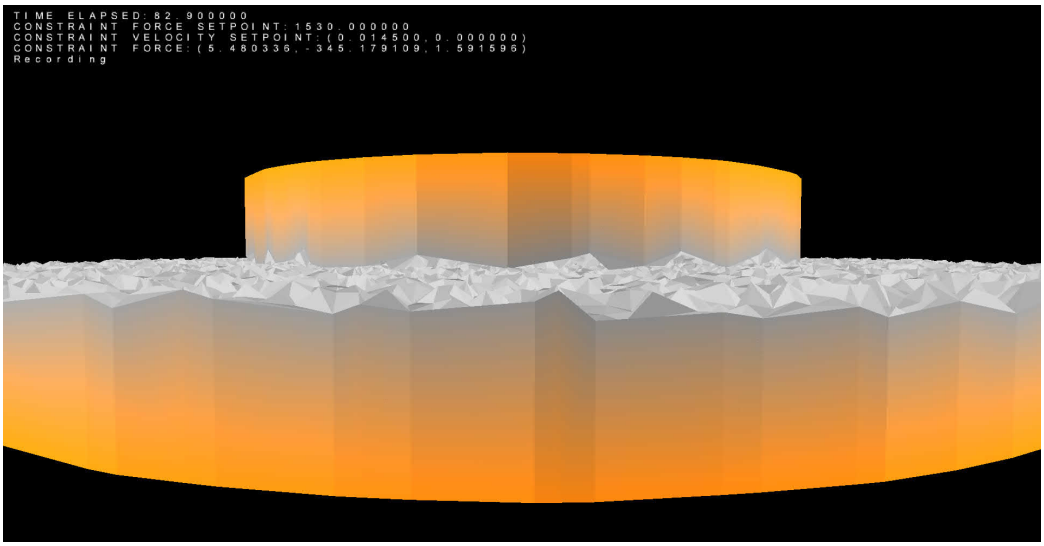

FIGURE 4.4: **An output frame from our 3D simulator.**

For testing friction in 3D case, we use a simple controller which can maintain a constant normal force while dragging the object horizontally. While it is possible to simply drop the object with gravity to simulate desired normal force, the main drawback of this approach is that the object tends to bounce off the surface and takes a while to settle down. The controller uses the component of contact force along the normal

direction as feedback and outputs necessary correction forces that keep the feedback forces within 5 percent margin of the setpoint. The output forces of the controller are applied to the vertices on the top layer of the top object. Vertices on the bottom layer of the bottom object are permanently pinned. Figure 4.4, shows the constrained vertices on top and bottom objects in orange. It should be noted that only one layer of vertices are pinned and the orange shade is due to the interpolation of colors by rasterization system.

4.2.1 Methodology - Dynamic Friction

With dynamic friction, there is a net relative motion between the two soft bodies and we expect some continuous resistive force opposing the relative motion. We apply a normal force of 15 Newton on the smaller of the two soft bodies and drag it over the larger body with a velocity 0.2 meters per second. We collect the horizontal and normal feedback forces from the controller while the object is being dragged. The feedback forces indicate the force necessary to sustain a constant horizontal velocity and also indicate the coefficient of friction.

Results

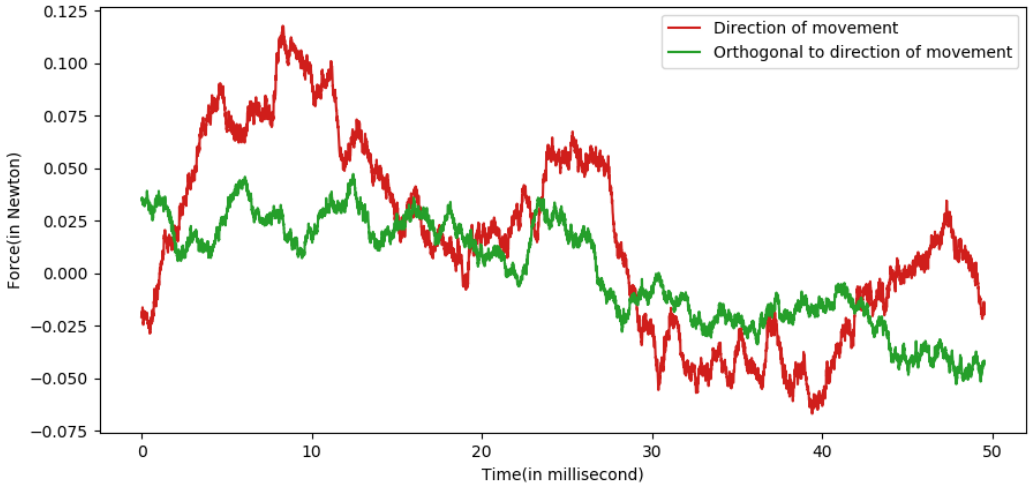

FIGURE 4.5: **The plot of tangential forces over time. The normal force is held constant at 15 Newton with a relative velocity of 0.2 m/s.**

We see in figure 4.5, the horizontal forces are negligible compared to the normal force. With dynamic friction, we expect to see a net positive area under the red curve and zero for the green curve. While the area under the red curve is 1.17×10^{-4}, the area under the green curve is 8.88×10^{-7}. We can define the average coefficient of

friction as follows

$$\mu_{kinetic} = \frac{\int_0^t f_T \, dt}{\int_0^t f_N \, dt}. \tag{4.1}$$

In the above equation, f_T is the force along the direction of motion while f_N is the normal force. We obtain $\mu_{kinetic} = 0.0016$.

Clearly, the dynamic/kinetic coefficient of friction is way too small. In our simulator, the only energy dissipation is due to Rayleigh damping which we think is insufficient to capture the energy loses due to dynamic friction. As discussed in our tribology literature review, in most cases it is essential to introduce additional energy dissipating elements at a local junction of contact. The purpose of the local energy dissipating element is to model the molecular forces of attraction at the point of contact. One such model has been described in Karpenko et al. [20].

4.2.2 Methodology - Static Friction

In the static friction case, there is no relative between the two soft bodies. The two bodies should remain in static equilibrium until the horizontal force unlatches the entangled tetrahedrons and the body starts to move. In our simulator, we slowly increase the horizontal force while maintaining an almost constant normal force. We expect the object to remain stationary until a threshold for horizontal force is reached after which it should start sliding. While processing the collected data, we select the maximum horizontal force and the corresponding normal force to determine the static coefficient of friction. We also ensure that the average velocity of the sliding object is nearly zero (≤ 0.01 m/s) around the instant maximum horizontal force is attained.

Results

As seen from Figure 4.6, the tangential feedback force is maximum around 0.2 ms after which the body is no longer in static equilibrium and accelerates with increasing force. We ran our simulator twice, with different initial starting positions and the results are tabulated as below.

	μ
Test 1	0.081
Test 2	0.096

TABLE 4.2: **The coefficient of static friction obtained by our simulator for metal like material.**

In both tests, we see that our simulator produced a static coefficient of friction which is quite low for metallic surfaces. However, the results show some qualitative

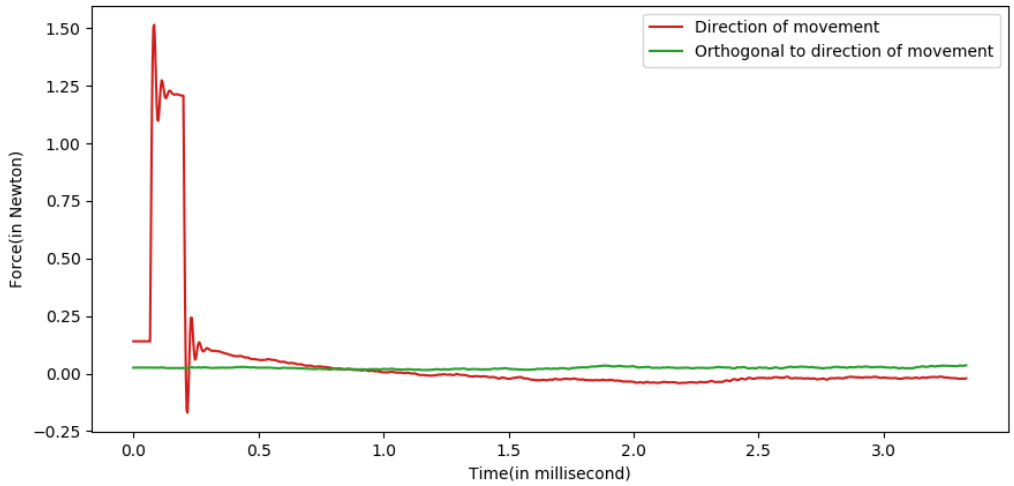

FIGURE 4.6: **The plot of tangential feedback forces over time. The normal force is held constant at 15 Newton while tangential force is increased over time.**

agreement with the theory of static friction as we would expect the friction force to increase up to a point and then decrease.

4.3 Real World Measurements

We collected (Table 4.3) static coefficient of friction and various elastic parameters from multiple sources [34, 42] available on the internet.

Our aim is to find any relationship between the coefficient of static friction and elastic parameters and compare them with the data obtained from our simulator. Our original goal was to generate enough data from our simulator and use a function approximator to summarize the result and predict the friction coefficient for material pairs which are not in our dataset. As we do not have enough data from our simulator, we rely on real measurements for the purpose of prediction.

4.3.1 Methodology

We collected the coefficient of static friction for various material pairs such as metals, plastic, wood, and glass under clean and dry condition. We also obtain the elastic properties namely Young's modulus, Poisson ratio and mass density for all materials in our list. We know that surface roughness plays an important role in determining friction force but no additional information on surface roughness was available at the source. Hence, we assume that the surface roughness of the material pairs in contact is

Material 1				Material 2				μ_{static}
Name	E	ν	ρ	Name	E	ν	ρ	
Steel	200	0.25	8000	Steel	200	0.25	8000	0.78
Copper	133	0.35	8940	Copper	133	0.35	8940	1.21
Aluminum	69	0.33	2700	Aluminum	69	0.33	2700	1.1
Aluminum	69	0.33	2700	Steel	200	0.25	8000	0.61
Brass	115	0.34	8730	Cast Iron	170	0.26	7200	0.4
Cadmium	64	0.31	8650	Cadmium	64	0.31	8650	0.5
Cadmium	64	0.31	8650	Steel	200	0.25	8000	0.6
Cast Iron	170	0.26	7200	Cast Iron	170	0.26	7200	1.1
Chromium	248	0.31	7190	Chromium	248	0.31	7190	0.41
Copper	133	0.35	8940	Cast Iron	170	0.26	7200	1.05
Copper	133	0.35	8940	Copper	133	0.35	8940	1.0
Copper	133	0.35	8940	Steel	200	0.25	8000	0.53
Glass	60	0.25	2400	Glass	60	0.25	2400	0.95
Glass	60	0.25	2400	Steel	200	0.25	8000	0.6
Glass	60	0.25	2400	Nickel	170	0.31	8900	0.78
Graphite	20	0.2	2050	Graphite	20	0.2	2050	0.1
Graphite	20	0.2	2050	Steel	200	0.25	8000	0.1
Nickel	170	0.31	8900	Nickel	170	0.31	8900	0.8
Nickel	170	0.31	8900	Steel	200	0.25	8000	0.7
Nylon	4	0.39	1130	Nylon	4	0.39	1130	0.2
Plexiglass	3.3	0.37	1190	Plexiglass	3.3	0.37	1190	0.8
Plexiglass	3.3	0.37	1190	Steel	200	0.25	8000	0.45
Polystyrene	2.5	0.4	1040	Polystyrene	2.5	0.4	1040	0.5
Polystyrene	2.5	0.4	1040	Steel	200	0.25	8000	0.3
Rubber	0.01	0.47	2300	Asphalt	3	0.35	2500	0.6
Rubber	0.01	0.47	2300	Concrete	17	0.17	2400	0.9
Brass	115	0.34	8730	Steel	200	0.25	8000	0.35
Steel	200	0.25	8000	Cast Iron	170	0.26	7200	0.4
Teflon	0.5	0.47	2200	Steel	200	0.25	8000	0.04
Teflon	0.5	0.47	2200	Teflon	0.5	0.47	2200	0.04
Wood	10	0.35	750	Wood	10	0.35	750	0.3
Wood	10	0.35	750	Copper	133	0.35	8940	0.4
Wood	10	0.35	750	Steel	200	0.25	8000	0.45
Wood	10	0.35	750	Concrete	17	0.17	2400	0.62
Zinc	82.7	0.25	7120	Zinc	82.7	0.25	7120	0.6
Zinc	82.7	0.25	7120	Cast Iron	170	0.26	7200	0.85

E : Young's modulus (in GPa)
ν : Poisson ratio
ρ : Mass density (in kg/m^3)

TABLE 4.3: **Table of static friction coefficient (μ_{static}) obtained from real world measurements. The data in the table is compiled from various sources available online.**

similar for all observations. We then train an $\epsilon - SVR$ [40, 32] regressor using the material properties of the contact pair as our feature vector and coefficient of static friction as the target. The reason we choose SVR is that SVR can approximate non-linear functions and offers greater control for avoiding overfitting. The term overfitting refers to the issue in function approximation literature where the function fails to capture the general trend in the data and behave more like a hash table. The epsilon paramater indicates the maximum permissible error for each data point in the training set. So, a lower value of epsilon is pushes the regressor to fit the training data as closely as possible. The constant $C \geq 0$, determines the trade-off between the flatness of the regession boundary and the amount up to which deviations larger than epsilon are tolerated. So, a higher value of C indicates lower tolerance and higher chances of overfitting the data. Since we only have few tens of samples to train on, overfitting is a major concern. We set the SVR hyperparameters $\epsilon = 0.1$ and $C = 0.5$. The hyperparameters are chosen such that it minimizes the training error while maintaining optimal function complexity. Increasing C or decreasing ϵ have the similar effect of reducing training error and increasing function complexity. To get an idea of the function complexity, we plot the coefficient of friction with respect to elastic parameters and settle on hyperparameters that output friction coefficient within a plausible range with minimal variance.

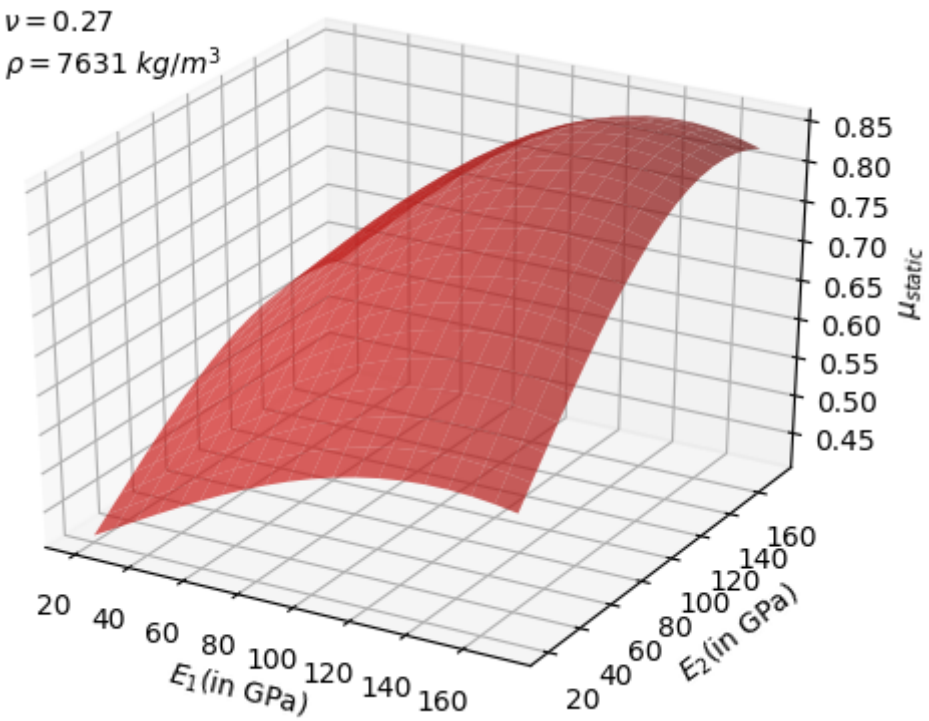

FIGURE 4.7A: **Variation in the coefficient of static friction with Young's modulus (E) of the two materials in contact.**

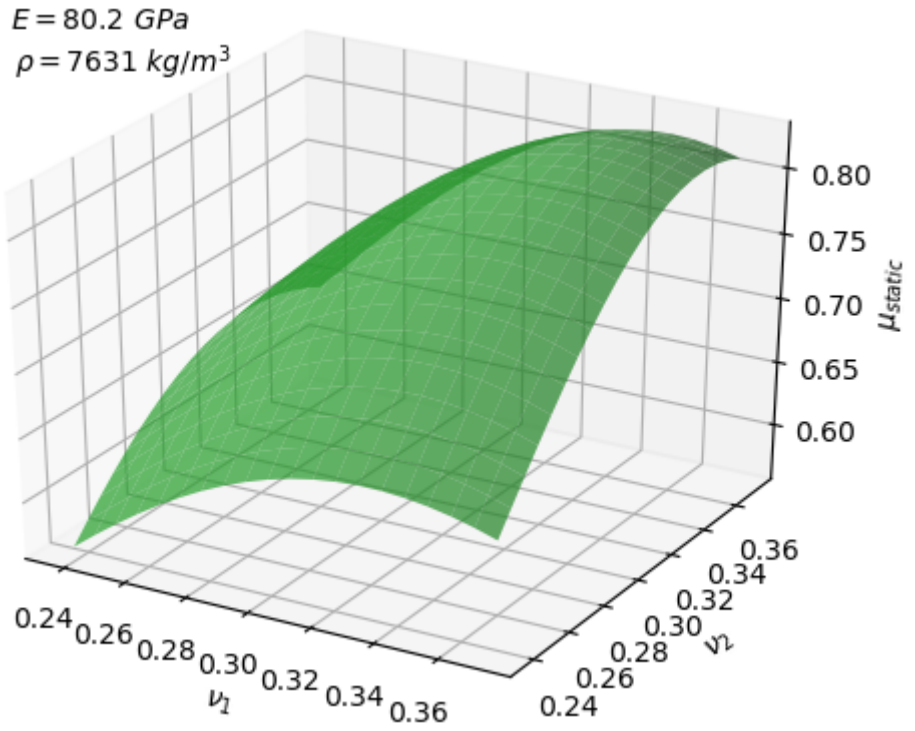

FIGURE 4.7B: **Variation in the coefficient of static friction with Poisson ratio (ν) of the two materials in contact.**

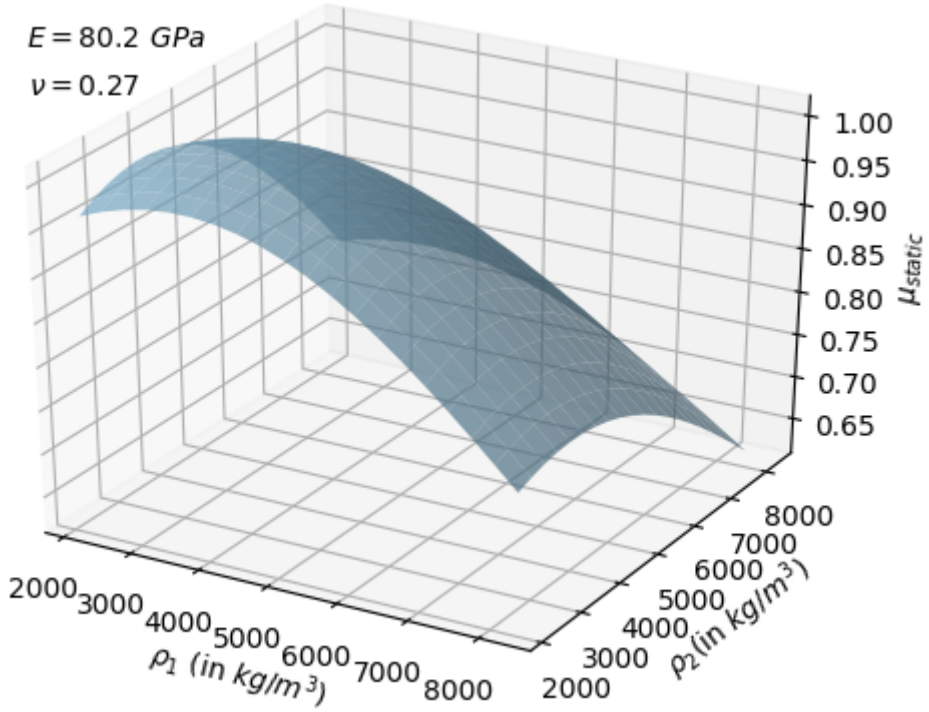

FIGURE 4.7C: **Variation in the coefficient of static friction with mass density (ρ) of the two materials in contact.**

4.3.2 Results

After fitting the regressor, we plot the coefficient of static friction with respect to Young's modulus, Poisson ratio and mass density as shown in Figure 4.7. Comparing Figure 4.7 with Figure 4.3, we see there is a qualitative agreement between the results of our simulator and measurements from real world in the 2D case. Based upon our simulated data, we predicted the coefficient of static friction to increase with Young's modulus and Poisson ratio while decrease with mass density in agreement with the real measurements. The predictions for Poisson ratio from our 2D simulator only showed a weak correlation with static friction while real data suggests otherwise. We think real world materials are much stiffer and for stiff materials, variation in Poisson ratio has more impact on the coefficient of static friction.

4.4 Conclusion

We notice that while we obtained interesting results in the 2D case, the same was not possible in 3D. In 3D, we have a vastly greater number of tetrahedra and contacts. Accurate collision detection and resolution for arbitrary geometry in 3D is challenging in terms of both complexity and computation time. We theorize, since our contacts are not interpenetration free in 3D, the contacts are unable to support extreme force required for deforming stiff materials and produce necessary resistance to stay in equilibrium in the static case. For dynamic case, the coefficient of friction is almost negligible, and we think the reason is that we do not have any energy dissipating elements modeling molecular adhesion. Another issue we have in 3D is the time required for simulating tens of milliseconds may run into days. Our simulator also requires accurate tuning of parameters which control the repulsive collision forces and requires some trial and error to get the correct parameters. Combined with the slowness of our simulator, this can be a huge bottleneck in simulating different surface geometries. On the positive side, we obtained qualitative validation of static friction in 3D case. In 2D case, we showed the variation of the static coefficient with various elastic parameters and results obtained matched qualitatively with data obtained from real-world measurements. In the next chapter, we discuss further improvement based on our current experience as future work.

Chapter 5

Conclusion

In this thesis, we first looked into various techniques used for modeling friction in tribology and graphics literature and their scope in terms of application, robustness, and accuracy. We designed a simulator for capturing the phenomenon of static and dynamic friction and discussed the issues associated with its design. We showed a correlation between the coefficient of static friction and the elastic parameters Young's modulus, Poisson ratio and Rayleigh damping in 2D case. We also accomplished a qualitative verification of the results obtained in the 2D case with real-world measurements. In 3D, we attempted to simulating both static and dynamic friction. We achieved a qualitative validation of the result in the static case. In the dynamic case, our simulator failed to produce any meaningful results due to inaccuracies in the contact model. We discussed various key areas where our simulator is lacking and how they could affect the results obtained. We propose various improvements for our simulator based on our experience as future work in the next section.

5.1 Future Work

It is an engineering challenge to simulate thousands of contacts accurately. While an accurate contact model is very important for simulating friction, steep computation cost remains a major hindrance in implementing such a model. Even with simple contact model, the simulation times are in the order of days for tens of milliseconds of simulation. Such timescales are clearly not useful in terms of practical purposes when our aim is to generate a lot of data. Since this simulator was not designed with scalability in mind, our simulator shows diminishing gains after four CPU cores. We think a better approach is to resolve the contacts accurately without interpretations while distributing the collision computations over multiple CPU cores. Apart from robust and interpretation free contact, it might be important to implement a nominal friction model locally at every contact for simulating molecular adhesion, plastic deformation and wear in the quasi-static dynamic case [29]. We also realized that real materials are extremely stiff and require very small time steps ($10^{-7}s$) with non-implicit integration

schemes. We already know the main computational bottleneck is not in simulating a large number of tetrahedra but in resolving collisions. With larger steps and more accurate collisions, there may be a reduction in compute time due to a reduction in the net time spent on collisions for equivalent simulation time. Since the cost of computing $\mathbf{f_{int}}$ is minimal, it may be worthwhile to see the trade-off between increasing the resolution of the tetrahedral mesh while using larger time steps using implicit schemes with Newton iterations.

Bibliography

[1] M. Anitescu and F. A. Potra. "Formulating Dynamic Multi-Rigid-Body Contact Problems with Friction as Solvable Linear Complementarity Problems". In: *Nonlinear Dynamics* 14.3 (1997), pp. 231–247. ISSN: 1573-269X. DOI: 10.1023/A:1008292328909. URL: https://doi.org/10.1023/A:1008292328909.

[2] J. F. Archard, W. Hirst, and Thomas Edward Allibone. "The wear of metals under unlubricated conditions". In: *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences* 236.1206 (1956), pp. 397–410. DOI: 10.1098/rspa.1956.0144.

[3] David Baraff. "Analytical Methods for Dynamic Simulation of Non-penetrating Rigid Bodies". In: *Computer Graphics*. ACM, 1989, pp. 223–232.

[4] David Baraff. "Coping with Friction for Non-penetrating Rigid Body Simulation". In: *SIGGRAPH Comput. Graph.* 25.4 (July 1991), pp. 31–41. ISSN: 0097-8930. DOI: 10.1145/127719.122722. URL: http://doi.acm.org/10.1145/127719.122722.

[5] David Baraff. "Fast Contact Force Computation for Nonpenetrating Rigid Bodies". In: *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH '94. New York, NY, USA: ACM, 1994, pp. 23–34. ISBN: 0-89791-667-0. DOI: 10.1145/192161.192168. URL: http://doi.acm.org/10.1145/192161.192168.

[6] Florence Bertails-Descoubes et al. "A Nonsmooth Newton Solver for Capturing Exact Coulomb Friction in Fiber Assemblies". In: *ACM Trans. Graph.* 30.1 (Feb. 2011), 6:1–6:14. ISSN: 0730-0301. DOI: 10.1145/1899404.1899410. URL: http://doi.acm.org/10.1145/1899404.1899410.

[7] Bernd Bickel et al. "Design and Fabrication of Materials with Desired Deformation Behavior". In: *ACM Trans. Graph.* 29.4 (July 2010), 63:1–63:10. ISSN: 0730-0301. DOI: 10.1145/1778765.1778800. URL: http://doi.acm.org/10.1145/1778765.1778800.

[8] F.P. Bowden and D. Tabor. "The Friction and Lubrication of Solids". In: *Oxford University Press* (1954).

[9] Robert Bridson. "Fast Poisson Disk Sampling in Arbitrary Dimensions". In: *ACM SIGGRAPH 2007 Sketches*. SIGGRAPH '07. San Diego, California: ACM, 2007. ISBN: 978-1-4503-4726-6. DOI: 10.1145/1278780.1278807. URL: http://doi.acm.org/10.1145/1278780.1278807.

[10] Desai Chen et al. "Spec2Fab: A Reducer-tuner Model for Translating Specifications to 3D Prints". In: *ACM Trans. Graph.* 32.4 (July 2013), 135:1–135:10. ISSN: 0730-0301. DOI: 10.1145/2461912.2461994. URL: http://doi.acm.org/10.1145/2461912.2461994.

[11] Zhili Chen, Renguo Feng, and Huamin Wang. "Modeling Friction and Air Effects Between Cloth and Deformable Bodies". In: *ACM Trans. Graph.* 32.4 (July 2013), 88:1–88:8. ISSN: 0730-0301. DOI: 10.1145/2461912.2461941. URL: http://doi.acm.org/10.1145/2461912.2461941.

[12] Kenny Erleben. "Velocity-based Shock Propagation for Multibody Dynamics Animation". In: *ACM Trans. Graph.* 26.2 (June 2007). ISSN: 0730-0301. DOI: 10.1145/1243980.1243986. URL: http://doi.acm.org/10.1145/1243980.1243986.

[13] H.A. Francis. "Application of spherical indentation mechanics to reversible and irreversible contact between rough surfaces". In: *Wear* 45.2 (1977), pp. 221 –269. ISSN: 0043-1648. DOI: https://doi.org/10.1016/0043-1648(77)90076-X. URL: http://www.sciencedirect.com/science/article/pii/004316487790076X.

[14] N. Gane, P. F. Pfaelzer, and D. Tabor. "Adhesion between Clean Surfaces at Light Loads". In: *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences* 340.1623 (1974), pp. 495–517. ISSN: 00804630. URL: http://www.jstor.org/stable/78768.

[15] J. A. Greenwood, J. B. P. Williamson, and Frank Philip Bowden. "Contact of nominally flat surfaces". In: *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences* 295.1442 (1966), pp. 300–319. DOI: 10.1098/rspa.1966.0242.

[16] Hertz H. and Gesammelte W. "Hertzian Contact Model". In: (1895).

[17] David Harmon et al. "Asynchronous Contact Mechanics". In: *ACM SIGGRAPH 2009 Papers*. SIGGRAPH '09. New Orleans, Louisiana: ACM, 2009, 87:1–87:12. ISBN: 978-1-60558-726-4. DOI: 10.1145/1576246.1531393. URL: http://doi.acm.org/10.1145/1576246.1531393.

[18] Ragnar Holm. "Electric Contacts". In: (1958). DOI: 10.1007/978-3-662-06688-1.

[19] Yifeng Jiang and C. Karen Liu. "Data-Augmented Contact Model for Rigid Body Simulation". In: *CoRR* abs/1803.04019 (2018). arXiv: 1803.04019. URL: http://arxiv.org/abs/1803.04019.

[20] Yu.A. Karpenko and Adnan Akay. "A numerical model of friction between rough surfaces". In: *Tribology International* 34.8 (2001), pp. 531 –545. ISSN: 0301-679X. DOI: https://doi.org/10.1016/S0301-679X(01)00044-5. URL: http://www.sciencedirect.com/science/article/pii/S0301679X01000445.

[21] Danny M. Kaufman, Timothy Edmunds, and Dinesh K. Pai. "Fast Frictional Dynamics for Rigid Bodies". In: *ACM Trans. Graph.* 24.3 (July 2005), pp. 946–956. ISSN: 0730-0301. DOI: 10.1145/1073204.1073295. URL: http://doi.acm.org/10.1145/1073204.1073295.

[22] J. Krim. "Atomic-Scale Origins of Friction". In: *Langmuir* 12.19 (1996), pp. 4564–4566. DOI: 10.1021/la950898j. eprint: https://doi.org/10.1021/la950898j. URL: https://doi.org/10.1021/la950898j.

[23] Paul G. Kry and Dinesh K. Pai. "Interaction Capture and Synthesis". In: *ACM SIGGRAPH 2006 Papers*. SIGGRAPH '06. Boston, Massachusetts: ACM, 2006, pp. 872–880. ISBN: 1-59593-364-6. DOI: 10.1145/1179352.1141969. URL: http://doi.acm.org/10.1145/1179352.1141969.

[24] Ares Lagae et al. "Procedural Noise using Sparse Gabor Convolution". In: *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2009)* 28.3 (2009), pp. 54–64. DOI: 10.1145/1531326.1531360.

[25] J. Li et al. "A GPU-accelerated finite element solver for simulation of soft-body deformation". In: *2013 IEEE International Conference on Information and Automation (ICIA)*. 2013, pp. 631–636. DOI: 10.1109/ICInfA.2013.6720373.

[26] Hammad Mazhar et al. "Using Nesterov's Method to Accelerate Multibody Dynamics with Friction and Contact". In: *ACM Trans. Graph.* 34.3 (May 2015), 32:1–32:14. ISSN: 0730-0301. DOI: 10.1145/2735627. URL: http://doi.acm.org/10.1145/2735627.

[27] Matthew Moore and Jane Wilhelms. "Collision Detection and Response for Computer Animation". In: *SIGGRAPH Comput. Graph.* 22.4 (June 1988), pp. 289–298. ISSN: 0097-8930. DOI: 10.1145/378456.378528. URL: http://doi.acm.org/10.1145/378456.378528.

[28] P. Ranganath Nayak. "Random Process Model of Rough Surfaces". In: *Journal of Lubrication Technology* 93 (July 1971), p. 398. DOI: 10.1115/1.3451608.

[29] J.T. Oden and J.A.C. Martins. "Models and computational methods for dynamic friction phenomena". In: *Computer Methods in Applied Mechanics and Engineering* 52.1 (1985), pp. 527 –634. ISSN: 0045-7825. DOI: https://doi.org/10.1016/0045-7825(85)90009-X. URL: http://www.sciencedirect.com/science/article/pii/004578258590009X.

[30] J A Ogilvy. "Numerical simulation of friction between contacting rough surfaces". In: *Journal of Physics D: Applied Physics* 24.11 (1991), p. 2098. URL: http://stacks.iop.org/0022-3727/24/i=11/a=030.

[31] Simon Pabst, Bernhard Thomaszewski, and Wolfgang Strasser. "Anisotropic Friction for Deformable Surfaces and Solids". In: *Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. SCA '09. New Orleans, Louisiana: ACM, 2009, pp. 149–154. ISBN: 978-1-60558-610-6. DOI: 10.1145/1599470.1599490. URL: http://doi.acm.org/10.1145/1599470.1599490.

[32] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[33] B N J Persson et al. "On the nature of surface roughness with application to contact mechanics, sealing, rubber friction and adhesion". In: *Journal of Physics: Condensed Matter* 17.1 (2005), R1. URL: http://stacks.iop.org/0953-8984/17/i=1/a=R01.

[34] Rob Ramsdale. *Table of friction coefficients*. URL: http://www.engineershandbook.com/Tables/frictioncoefficients.htm.

[35] Alessandro Pietro Rigazzi. "The effects of roughness on the area of contact and on the elastostatic friction". In: *Doctoral Dissertation submitted to the Faculty of Informatics of the Università della Svizzera Italiana* (2004).

[36] Jonathan Richard Shewchuk. "Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator". In: *Applied Computational Geometry: Towards Geometric Engineering*. Ed. by Ming C. Lin and Dinesh Manocha. Vol. 1148. Lecture Notes in Computer Science. From the First ACM Workshop on Applied Computational Geometry. Springer-Verlag, May 1996, pp. 203–222.

[37] Hang Si. "TetGen, a Delaunay-Based Quality Tetrahedral Mesh Generator". In: *ACM Trans. Math. Softw.* 41.2 (Feb. 2015), 11:1–11:36. ISSN: 0098-3500. DOI: 10.1145/2629697. URL: http://doi.acm.org/10.1145/2629697.

[38] Eftychios Sifakis and Jernej Barbic. "FEM Simulation of 3D Deformable Solids: A Practitioner's Guide to Theory, Discretization and Model Reduction". In: *ACM SIGGRAPH 2012 Courses*. SIGGRAPH '12. Los Angeles, California: ACM, 2012, 20:1–20:50. ISBN: 978-1-4503-1678-1. DOI: 10.1145/2343483.2343501. URL: http://doi.acm.org/10.1145/2343483.2343501.

[39] Breannan Smith, Fernando De Goes, and Theodore Kim. "Stable Neo-Hookean Flesh Simulation". In: *ACM Trans. Graph.* 37.2 (Mar. 2018), 12:1–12:15. ISSN: 0730-0301. DOI: 10.1145/3180491. URL: http://doi.acm.org/10.1145/3180491.

[40] Alex J. Smola and Bernhard Schölkopf. "A tutorial on support vector regression". In: *Statistics and Computing* 14.3 (2004), pp. 199–222. ISSN: 1573-1375. DOI: 10.1023/B:STCO.0000035301.49549.88. URL: https://doi.org/10.1023/B:STCO.0000035301.49549.88.

[41] D. E. Stewart and J. C. Trinkle. "An Implicit Time-Stepping Scheme for Rigid Body Dynamics with Inelastic Collisions and Coulomb Friction". In: *International Journal for Numerical Methods in Engineering* 39 (Aug. 1996), pp. 2673–2691. DOI: 10.1002/(SICI)1097-0207(19960815)39:15<2673::AID-NME972>3.0.CO;2-I.

[42] Engineering ToolBox. *Poisson's Ratio for Metals*. URL: https://www.engineeringtoolbox.com/metals-poissons-ratio-d_1268.html (visited on 08/27/2010).