

A Staggered Grid-Based Variational Approach for Modeling Elastic Deformation

by

Brooke Dolny

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Computer Science

Waterloo, Ontario, Canada, 2025

© Brooke Dolny 2025

Author's Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Elastic deformation is often simulated in computer graphics using the Finite Element Method on tetrahedral meshes. However, generating a tetrahedral mesh can be complicated and expensive. When a hierarchy of meshes is needed (for example, with a progressive or multigrid method), generating this set of hierarchical meshes is a time-consuming process. However, in other areas of physics simulation, such as fluid simulation, the use of staggered grids and finite differences is much more common. The application of adaptive or multigrid methods to grid-based simulations is trivial in comparison. By applying variational techniques from fluid simulation [2; 3] to the staggered grid-based elasticity simulation method of Zhu et al. [26], we produce a method that accurately solves the linear elasticity partial differential equations with free boundary conditions solved implicitly.

In this work, we derive a variational formulation for the linear elasticity partial differential equations on a staggered grid. We derive both the static and dynamic forms of the minimization problem, and their subsequent discretizations. Our method only requires an indicator function that specifies the interior and exterior of the solid to be simulated, and does not require any information about the normals of the object’s surface. Furthermore, our method retains the simplicity and sparsity of the basic staggered grid finite difference scheme, but supports non-axis-aligned boundaries without the need for boundary-conforming meshing. We apply our method to several examples of uniform and nonuniform objects under different deformations, demonstrated with both static and quasi-static simulations. We also compare our results with analytical solutions to the linear elasticity partial differential equations to show the accuracy of our method and that our method converges well with grid refinement.

Acknowledgements

Thank you to my supervisor, Professor Christopher Batty, for being the best supervisor I could have asked for. I am immensely grateful to have been a part of your lab. Thank you for meeting with me every week to work through research problems, for your detailed feedback on my work, and for your endless encouragement. I hope we get the opportunity work together again.

Thank you to Professor Justin Wan and Professor Toshiya Hachisuka for being my thesis readers, and for providing me with detailed feedback and interesting questions. I am so grateful to you for lending me your expertise and helping me improve my work.

Thank you to the members of the Computational Motion Group: Rikin, Clara, Ryusuke, Leo, Joel, Nathan, Tina, and Daniel. Your expertise and brilliance are inspiring, and I am so grateful for our discussions on research topics and for the opportunity to learn about your research. Thank you especially to Rikin for encouraging me to continue expanding my knowledge and improving my work, and for the long discussions about my research. I could not have finished this thesis without your help. Thank you also to my friends in the Computer Graphics Lab, especially Ben, Brian, Craig, Ege, Matthew, Saba, Silvia, Stephen, Yiwei, Zack, and Zane, for making the Graphics lab a welcoming and enjoyable environment. I will miss our weekly coffee hour.

Thank you to my friends, who I have too many to name, for being such wonderful and kind people. I am so grateful I got to go on picnics, play Magic, watch movies, go to concerts, talk about TV shows, and spend time with you all. I could not have succeeded with my research without your encouragement and friendship.

Thank you to my family for their support and encouragement, and to my parents, Cole and Nancy, for encouraging me to apply to graduate school. Thank you especially to my brother, Mitchell, for laughing with me about my research difficulties and for joining me in some long debugging sessions.

Thank you to the brilliant people I had the privilege of working with in the past, Professor Derek Rayside, Josh Cameron, Dr. Adrian Butscher, Dr. Massimiliano Meneghin, and Dr. Ahmed Mahmoud, for being my references to the programs I have applied to, for working on problems with me, and for encouraging me to pursue research. Thank you also to the many wonderful teachers and professors I have had over the years for passing on your knowledge and wisdom to me.

Dedication

To my grandmother Rose, the best math teacher I could have asked for.

Table of Contents

Author's Declaration	ii
Abstract	iii
Acknowledgements	iv
Dedication	v
List of Figures	viii
List of Tables	x
1 Introduction	1
2 Related Work	3
2.1 Discretizations	3
2.2 Deformation Models	5
3 Mathematics Background	7
3.1 Elastic Deformation	7
3.2 Linear Elasticity	8
3.3 Spatial Discretization	10
3.4 Boundary Conditions	13

3.5	Dynamics	13
3.6	Geometric Representation	14
4	Variational Approach to Linear Elasticity	15
4.1	Minimization form of the PDE	15
4.2	Static and Quasi-static Problem	16
4.3	Treatment of Boundary Conditions	20
4.3.1	Non-Zero Traction Boundary Conditions	20
4.4	Dynamics	22
5	Results	24
5.1	Convergence Studies	24
5.2	Conditioning	25
5.3	Static and Quasistatic Simulation	29
6	Conclusion	33
6.1	Limitations	33
6.2	Extensions	34
6.2.1	3D Discretization	34
6.2.2	Multigrids and Spatial Adaptive Grids	34
6.2.3	Alternative Elasticity Models	34
	References	35

List of Figures

3.1	Arrangement of the staggered variables on the x and y grids.	12
3.2	Visual representation of the stencils from the discretized PDE on both the staggered x and y grids.	12
4.1	Layout of volume fraction terms relative to the underlying grid. Each $V_{i,j}$ term indicates the centre position of an $h \times h$ control volume over which the volume fractions are estimated.	18
5.1	The numerical error in the static simulation with respect to the analytical solution, visualized in rest configuration of the object. The annulus is embedded in a 2048×2048 grid. The left image has all fixed boundary conditions, and the right image has traction outer boundaries and fixed interior boundaries. The fixed boundary case's error is approximately one order of magnitude lower than the traction boundary case example.	26
5.2	\mathcal{L}_1 and \mathcal{L}_∞ error with respect to the analytical solution for free outer boundary and fixed interior boundary.	26
5.3	\mathcal{L}_1 and \mathcal{L}_∞ Error with respect to the analytical solution for fixed interior and exterior boundaries.	27
5.4	Condition numbers of static simulation with varying material parameters of ν from 0.1 to 0.495 and E from 1 to 10^5 . As ν approaches 0.5, we can see the condition number of the system increases asymptotically, while varying E does not substantially affect the conditioning of the system.	28
5.5	(a)-(d) Quasistatic simulation of an axis-aligned expansion deformation. The bottom and top edges of the rectangle are pulled outwards, causing the free boundaries to be pulled inwards. Fixed boundary points are coloured orange.	29

5.6	(a)-(d) Quasistatic simulation of an axis-aligned expansion deformation. The bottom and top edges of the rectangle are pushed inwards. The left and right free boundaries bulge outward in response to the compression. Fixed boundary points are coloured orange.	29
5.7	(a)-(d) Quasistatic simulation of a simple shear deformation. The bottom and top edges of the rectangle are pulled tangentially in opposite directions. Note that the material bulging slightly on the free left and right sides is an artifact due to the linear elasticity model. Fixed boundary points are coloured orange.	30
5.8	(a)-(d) Rectangular object with fixed ends undergoing deformation due to gravity under increasing gravitational forces. The total gravitational force on each object is approximately 0, 0.42, 0.73, and 1.11. Fixed boundary points are coloured orange.	30
5.9	(a)-(d) Quasistatic simulation of a circular elastic object undergoing continuous deformation. Portions of the edge of the circle are pulled outwards, with free boundary conditions between each segment. Fixed boundary points are coloured orange.	31
5.10	(a)-(d) Quasistatic simulation of a bunny undergoing extension, with fixed boundaries at the top of the ears and feet and free elsewhere. Fixed boundary points are coloured orange.	31
5.11	(a)-(d) Quasistatic simulation of a simple rotation of a square solid with a rotation angle from $\theta = 0$ to $\frac{\pi}{4}$. Artifacts due to the linear elasticity model are visible, such as volume inflation. These artifacts worsen as the angle θ increases. Fixed boundary points are coloured orange.	32

List of Tables

5.1	\mathcal{L}_1 and \mathcal{L}_∞ error for different grid resolutions with respect to analytical solution with free exterior and fixed interior boundary conditions.	27
5.2	\mathcal{L}_1 and \mathcal{L}_∞ error for different grid resolutions with respect to analytical solution with fixed boundary conditions.	28

Chapter 1

Introduction

Simulating elastic solids and deformation is applicable to multiple fields. Elastic solids are solid objects that deform under adequate force, but return to their original size and shape once the forces are removed. Elasticity is the ability of a solid to behave as an elastic solid. How much an object deforms under forces depends on the material, its shape, and the location and magnitude of the forces applied to it. For example, a golf ball will undergo elastic deformation upon being hit by a golf club, but does not noticeably deform by hand. However, other objects, such as rubber o-rings and sheets, can undergo significant elastic deformation by hand. The wide range of behaviour of these materials and objects can make accurately simulating them a difficult problem. In this work, we will only consider static and quasi-static elastic deformation, and not implement plastic (i.e., permanent) or time-dependent deformations.

The results of modeling the deformation of these objects and materials are important for many different applications in computer graphics, engineering, and even biology. In computer graphics, elastic deformation is primarily modeled for physics-based animation to produce realistic or physically plausible animations for use in movies or video games. In mechanical and civil engineering applications, the elasticity equations are used to model the deformation of buildings and bridges under large loads, to locate areas of high stress and strain in the object, and to test designs before construction. In biology, physically accurate modeling is important for modeling surgical simulation, and accurate contact models for how cells deform when undergoing contact and collisions with neighbouring cells. The wide range of applications for physically-based modeling of elastic deformation makes this a highly relevant topic for research.

The most common method for elasticity simulation is the Finite Element Method

(FEM). The Finite Element Method uses a finite subspace to approximate the space of infinite possible deformations, and uses a *polygonal mesh* in combination with a set of *basis functions* to find the best approximation of the deformation in the finite subspace. The mesh elements and basis functions together describe the base shape and the space of possible deformations it can undergo. The Finite Element Method allows for the use of different types of polygons and polyhedra, which allows for more flexibility in the surface representation. For example, it is possible to use a tetrahedral mesh, as shown by O’Brien and Hodgins [17], or cube meshes as shown by Muller et al. [15] and James et al. [9]. Further, by using different basis functions, the accuracy and flexibility of the method can be modified, as shown by Martin et al. [14] with the use of harmonic basis functions.

However, the Finite Element Method is not the only method to discretize the elasticity equations. The use of finite differences to discretize these equations has been done by Zhu et al. [26], where the authors additionally use a multigrid to improve the performance of their solver. In contrast to the discretization procedure with FEM, finite differences directly converts the derivative operators into their respective discrete differences. When compared to Finite Element Method discretization on a cubic mesh, the resulting finite difference equations have a sparser linear system, improving the performance of the solve step [26]. Additionally, in order to use FEM, one typically needs to perform mesh generation to construct an FEM mesh whose elements conform to the surface shape of the object to be simulated.

It is common to solve the partial differential equations governing physical systems using an energy or minimization formulation instead of directly discretizing the equations. This approach is common in fluid dynamics, for example in the papers by Batty and Bridson [3] and Larionov et al. [11], which investigate viscosity of fluids. These two papers are using a “variational finite difference” scheme, where the problem is phrased as an energy minimization but uses finite differences to discretize the necessary derivatives. This contrasts with standard finite differences, which, as explained above, simply replace the derivative operators in the PDE with their discrete operators. In elasticity simulation, this energy minimization perspective can be seen in work by Gast et al. [6], and is also the basis for the incremental potential contact papers by Li et al. [12], where they extend this concept to support robust contact/collision handling.

The goal of this thesis is to propose a variational finite difference method for elasticity simulation, using the energy formulation of the system to implicitly handle free boundary conditions. This discretization simplifies the complexity of discretizing free boundary conditions, and makes the resulting finite difference stencils more comparable to interior degrees of freedom.

Chapter 2

Related Work

Solving the partial differential equations that describe the internal stress and strain on an object is a longstanding problem in both engineering and computer graphics. Though the goals of these research fields differ in their applications, the basic principles of how to simulate these systems remains the same. In this section, we will discuss different models of elasticity in computer graphics, as well as different methods of solving the PDEs from these different models.

2.1 Discretizations

Finite Element Method

The most common method for discretizing elasticity equations in literature today is the Finite Element Method (FEM). The object is divided into tetrahedral volumetric “elements”, together with a combination of *basis functions*, which are then used as the discretized domain to simulate the elasticity PDEs on. One of the first papers in computer graphics to implement this method was the work by Bro-Nielsen and Cotin [4], who apply a volumetric tetrahedral FEM discretization for simulating deformable models in surgery simulation. Since then, many other publications utilize FEM for elasticity simulation, including work by O’Brien and Hodgins [17], and Smith et al. [20].

One benefit of FEM is that it can support not just the regular grid structure of finite differencing, but any other combination of volumetric elements. The use of tetrahedral elements is a common choice for meshes as it allows for a good approximation of the

surface of an object. Embedded finite element methods are also common, which embed a detailed fine-mesh geometry into a coarser FEM simulation mesh to increase apparent simulation detail at lower cost. Longva et al. [13] embed a detailed fine mesh in a coarser mesh, and perform a FEM solve on the coarser (non-conforming) mesh with higher order basis functions. This embedding, combined with higher-order basis functions, results in greater accuracy than a direct coarse simulation but at a lower computational cost than a fine-mesh simulation. Patterson et al. [18] show that they use FEM on a grid structure with an approximation of the boundary using a quadrature, circumventing the need for a mesh-based representation. Because of how common and how many variations there are on FEM, there are several texts that describe this method in more detail, and how it is used to solve these PDEs. In particular, for Finite Element Methods, a good resource is the textbook by Bathe [1].

Boundary Element Method

Another discretization method useful for elasticity is the Boundary Element Method (BEM). The benefit of using BEM for elasticity simulation is that we just need a mesh representation for the surface rather than the entire interior volume. This reduces the number of vertices needed to simulate an object's deformation. One of the earlier papers to use this method was work by James and Pai [10], which uses BEM to simulate real-time elastic deformation. More recently, in work by Sugimoto et al. [22] uses BEM to simulate linear elasticity, but with some additions to the linear model to remove some of the artifacts common with linear elasticity.

Finite Volume Method

The Finite Volume Method (FVM) is a numerical method for solving PDEs by converting volume integrals with a divergence term into surface integrals. The integrated function is then evaluated as fluxes leaving the surface of each finite volume. This makes the method conservative, making it useful to simulate problems involving energy or volume conservation. Teran et al. [23] used FVM to simulate contraction of muscle tissue with a hyperelasticity model, and also showed that the resulting computations are very similar to linear FEM. FVM is also common in fluid dynamics, since volume conservation is one desirable aspect of simulating fluids. One of the earlier applications of FVM in computer graphics was by Feldman et al. [5], who applied it to simulate smoke.

Finite Differences

Finite differences are the simplest method for discretizing PDEs because they straightforwardly replace each derivative operator with a corresponding finite difference estimate. These methods typically result in sparser systems than their FEM counterparts. Finite differences were first used to simulate elastic solids in computer graphics by Terzopoulos et al. [24], to simulate both cloth and sheets as well as three-dimensional elastic objects. More recently, Zhu et al. [26] applied finite differences on a staggered grid, and solved the resulting system via a multigrid scheme. The use of a grid-based structure makes the construction of the multigrid hierarchy trivial in comparison to tetrahedral meshes, and the use of finite differences makes the resulting system sparser.

Variational Methods for Solving Partial Differential Equations

Variational methods for solving PDEs involve using the weak form or the minimization form of the PDE, and solving the resulting minimization problem instead of working with the PDE directly. FEM is solving the weak form, and is one kind of variational method. Batty et al. [2] proposed a variational method that uses the minimization form of the PDE discretized with finite differences to simulate fast liquid-solid coupling. This method is expanded on by Batty and Bridson [3] to simulate viscous liquids with a method that implicitly solves complex boundary conditions on free surfaces. Larionov et al. [11] proposed to solve a variational formulation of the Stokes problem (i.e., solving pressure and viscosity together, rather than separately), allowing for natural treatment of the correct free surface conditions and therefore improved coiling and buckling behavior. While these papers are fluid dynamics papers, damping in elasticity is similar to viscosity in fluids. This idea of the elasticity/viscosity relationship is sometimes called the “Stokes-Rayleigh analogy”, and the similarity in the resulting PDEs makes methods used in fluid simulation relevant to elastic simulation. A minimization form is also used in elastic simulation, for example in work by Gast et al. [6] which uses FEM but formulates the backward Euler solve as a minimization problem to ensure stability at large time steps.

2.2 Deformation Models

To model the behaviour of different elastic solids, we can use different constitutive models, or constitutive laws, for how the internal and external forces relate to the deformation of the solid. The simplest model for elastic deformation is linear elasticity [4]. This model

results in a linear PDE, but is only accurate for small deformations. As a result, the linear elasticity model tends to produce artifacts for large deformations. Additionally, the model produces different deformations on rotated configurations. Despite these artifacts, the linear elasticity deformation model remains popular due to its ease of implementation and is often used as the initial model before extending a proposed method to more complex deformation models. In Chapter 3, we give an overview of the PDE form of linear elasticity. In Chapter 4 we utilize a minimization form of the linear elasticity PDE to simplify free boundary conditions.

Unlike linear elasticity, the co-rotational model, from Müller et al. [16], produces the same deformation in rotated frames (i.e., the resulting force exhibits rotational invariance), but still preserves some of the simplicity of the linear model. This model is nonlinear and, while not entirely physically-based, is an extension of linear elasticity that removes some of the undesirable artifacts seen in linear elasticity. Co-rotational models are still commonly implemented in elastic deformation papers, for example, by Zhu et al. [26]. Co-rotational models have also seen further work over the years, for example by Georgii and Westermann [7] and Stomakhin et al. [21].

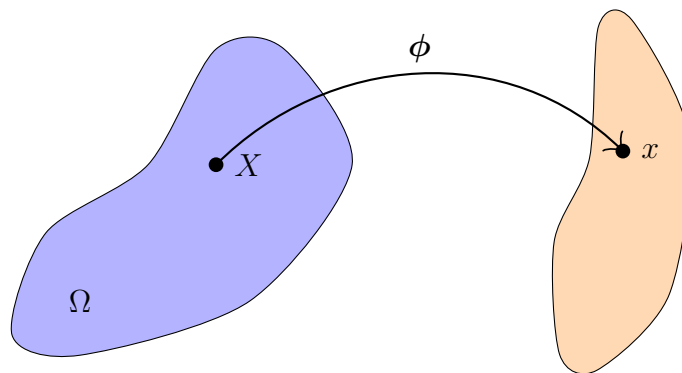
Hyperelastic deformation models are another class of models useful for simulating large elastic deformation of certain classes of material. These models are also nonlinear, and result in more complex systems to solve than linear or co-rotational models. As a result of this increased complexity in the model, they result in more realistic simulation of these hyperelastic materials. An example of this is seen in work from Smith et al. [20], which uses a Neo-hookean model for flesh simulation. The resulting simulation has good volume preservation, and in addition lacks the undesirable artifacts seen with linear and co-rotational models. However, the nonlinearity of these systems means that solving with implicit time-stepping schemes requires solving a nonlinear optimization problems via Newton’s method. This makes these methods more expensive, since they require solving a sequence of linear systems, rather than just one.

Chapter 3

Mathematics Background

3.1 Elastic Deformation

The following is a brief overview of elasticity and the linear elastic model. The course notes from Sifakis [19] provide a more detailed explanation. When simulating elastic deformation, we begin by considering an undeformed elastic object in a coordinate system. This is the *reference configuration* of the elastic object. We represent the elastic object as Ω , and use \mathbf{X} to represent the material points of the elastic object in the reference configuration, such that $\mathbf{X} \in \Omega$. When deforming an object, we can view the deformation as a map applied to all the material points of the elastic object from the reference configuration to the *deformed configuration*. This *deformation map* is represented with $\phi : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ (or in the case of 2D simulation, $\phi : \mathbb{R}^2 \rightarrow \mathbb{R}^2$). We can then map each material point in the reference configuration to its corresponding location in the deformed configuration as $\mathbf{x} = \phi(\mathbf{X})$.



Another important map to consider for the simulation of elastic objects is the *deformation gradient*, which is the Jacobian of the deformation map. This describes the relative change in the deformed position, given a small change in the rest configuration position. In 2D, the deformation gradient is written as:

$$\mathbf{F} = \nabla\phi = \begin{bmatrix} \frac{\partial}{\partial X}\phi_1 & \frac{\partial}{\partial Y}\phi_1 \\ \frac{\partial}{\partial X}\phi_2 & \frac{\partial}{\partial Y}\phi_2 \end{bmatrix} . \quad (3.1)$$

However, because of the number of first and second derivative terms in our model, it is often convenient to represent these derivatives with subscripts after a comma, where 1 and 2 correspond to the X and Y derivatives, respectively. Rewriting the components of Equation 3.1 using this notation, we get

$$F_{ij} = \phi_{i,j} .$$

The deformation gradient is used to define the *strain energy density*, which represents the energy density for a particular elasticity model, i.e., how much energy is stored per unit volume at a particular point, under a given deformation of the shape. We define the strain energy density Ψ as a function of the deformation gradient, $\Psi(\mathbf{F})$, and this energy density can be used to derive the related stress tensor, \mathbf{P} , and the resulting internal forces on the object.

3.2 Linear Elasticity

The relationship between the strain energy Ψ and the deformation gradient \mathbf{F} is referred to as the material's *constitutive model*. The specific constitutive model determines the internal forces that result from the change in the object's shape. Linear elasticity is the simplest constitutive model for elasticity. In this section, we follow the notation and derivation presented by Zhu et al. [26]. There are two material parameters used in the expression for the strain energy and stress in the linear elasticity model. These parameters, μ and λ are called the Lamé parameters, and can be defined in terms of the Young's modulus and Poisson Ratio:

$$\mu = \frac{E}{2 + 2\nu} \quad \lambda = \frac{E\nu}{(1 + \nu)(1 - 2\nu)} . \quad (3.2)$$

The Young's modulus, E , is a measure of how stiff the material is, and the Poisson ratio, ν , is a measure of the compressibility of the material. The Young's modulus is required to be a positive value, and the majority of solids have a Poisson ratio between 0 and 0.5, with a Poisson ratio of 0.5 representing an incompressible solid.

To define the energy density function for linear elasticity, we define the *small strain tensor*, which can be computed from the deformation gradient:

$$\boldsymbol{\epsilon} = \frac{1}{2} (\mathbf{F} + \mathbf{F}^T) - \mathbf{I} . \quad (3.3)$$

This model of the strain is only accurate for small deformations, which is a limitation that holds true for the linear elasticity model as well. We can then define the strain energy density for linear elasticity using Equation 3.3:

$$\Psi(\mathbf{F}) = \mu \boldsymbol{\epsilon} : \boldsymbol{\epsilon} + \frac{\lambda}{2} \text{tr}^2(\boldsymbol{\epsilon}) = \mu \|\boldsymbol{\epsilon}\|_F^2 + \frac{\lambda}{2} \text{tr}^2(\boldsymbol{\epsilon}) , \quad (3.4)$$

where $:$ is the double dot product, $\|A\|_F^2 = \sum_i \sum_j |A_{ij}|^2$ is the Frobenius norm squared, and $\text{tr}(A) = \sum_i A_{ii}$ is the trace of the matrix. The first Piola-Kirchhoff stress tensor, \mathbf{P} , for the linear elasticity model can be derived from Equation 3.4 and is written as follows:

$$\mathbf{P} = 2\mu \boldsymbol{\epsilon} + \lambda \text{tr}(\boldsymbol{\epsilon}) \mathbf{I} . \quad (3.5)$$

Using Equations 3.3 and 3.5, we can write the Piola-Kirchhoff stress tensor for the linear elasticity model component-wise in terms of the deformation map:

$$\begin{aligned} P_{ij} &= 2\mu \epsilon_{ij} + \sum_k \lambda \epsilon_{kk} \delta_{ij} \\ &= 2\mu \left(\frac{1}{2} (\phi_{i,j} + \phi_{j,i}) \right) - 2\mu \delta_{ij} + \sum_k \lambda \left(\frac{1}{2} (\phi_{k,k} + \phi_{k,k}) - \delta_{kk} \right) \delta_{ij} \\ &= \mu (\phi_{i,j} + \phi_{j,i}) - 2\mu \delta_{ij} + \sum_k \lambda (\phi_{k,k} - \delta_{kk}) \delta_{ij} . \end{aligned} \quad (3.6)$$

Deriving the partial differential equation form of the linear elasticity model uses the relationship between the external forces and the first Piola-Kirchhoff stress tensor. The internal elastic forces on the object are

$$\mathbf{f}_{\text{internal}} = \nabla^T \mathbf{P} , \quad (3.7)$$

where ∇ is the divergence operator. This relationship can be equivalently written component-wise as

$$f_i = \sum_j \partial_j P_{ij} .$$

To determine the right hand side of Equation 3.7, we can differentiate with respect to j and simplify to get the expression

$$\begin{aligned}
\sum_j \partial_j P_{ij} &= \sum_j \left[\mu (\phi_{i,jj} + \phi_{j,ij}) + \sum_k \lambda (\phi_{k,kj}) \delta_{ij} \right] \\
&= \sum_j \mu (\phi_{i,jj} + \phi_{j,ij}) + \sum_k \lambda (\phi_{k,ki}) \\
&= \sum_j \mu (\phi_{i,jj} + \phi_{j,ij}) + \sum_j \lambda (\phi_{j,ji}) .
\end{aligned} \tag{3.8}$$

Since $\phi_{j,ij} = \phi_{j,ji}$ by swapping the order of differentiation, Equation 3.8 can be rewritten as the following sum,

$$\sum_j \partial_j P_{ij} = \sum_j \mu \phi_{i,jj} + (\mu + \lambda) \phi_{j,ij} , \tag{3.9}$$

which gives the final relationship between the external forces on the object and the deformation map,

$$f_i = \sum_j \mu \phi_{i,jj} + (\mu + \lambda) \phi_{j,ij} . \tag{3.10}$$

This is a second-order partial differential equation, and it can be written in terms of the following linear operator,

$$\mathcal{L} = \mu \Delta I + (\mu + \lambda) \nabla \nabla^T . \tag{3.11}$$

To maintain the equilibrium between the internal and external forces, we have $\mathbf{f}_{\text{internal}} - \mathbf{f}_{\text{external}} = \mathbf{0}$. Combining this with Equation 3.11 the resulting linear, second order PDE is:

$$\mathcal{L} \phi = \mathbf{f}_{\text{external}} . \tag{3.12}$$

This gives us the final linear PDE form of the elastic model described by Equation 3.4.

3.3 Spatial Discretization

As discussed in Chapter 2, there are several common methods used for discretizing the elasticity PDE. The most common of these discretization methods is using the finite element method (FEM). However, it is possible to use alternative discretizations. For example, in fluid dynamics, it is common to discretize PDEs with finite differences or finite volumes. The benefit of using finite differences over FEM is that it typically produces sparser systems, which makes the subsequent solve steps computationally easier [26].

The discretization approach described below generally follows that of Zhu et al. [26]; our modifications to this method are deferred to Chapter 4. The PDE is discretized using a finite difference method on a staggered grid with central differences. The use of a staggered grid as opposed to a regular grid is to prevent grid-scale oscillations when using central differences, and to produce a slightly narrower stencil [26]. A staggered grid refers to a layout of the degrees of freedom where the different components of the deformation map are not collocated in space, but instead “staggered” as in Figure 3.1. Specifically, the x -coordinates are located halfway between nodal points with the same y -coordinate, and y -coordinates are located halfway between nodal points with the same x -coordinate. This strategy results in the following discrete derivative operators:

$$\begin{aligned}\frac{\partial u}{\partial x} &\approx \frac{1}{h} \left(u \left[x + \frac{h}{2}, y \right] - u \left[x - \frac{h}{2}, y \right] \right) \\ \frac{\partial u}{\partial y} &\approx \frac{1}{h} \left(u \left[x, y + \frac{h}{2} \right] - u \left[x, y - \frac{h}{2} \right] \right) ,\end{aligned}\tag{3.13}$$

where h is the grid spacing. Although the derivations and discretizations in this thesis are done in two dimensions, the ideas generalize to three dimensions well, with additional stencils for the derivative operators in the z direction.

To simplify the subscripts in the discretizations, we write the discretized components of $\phi = (\phi_1, \phi_2)$ as x and y , and the indices of the grid cells are written as $x_{i,j}$ and $y_{i,j}$. For the discretized equations, these comma-separated items do not represent derivatives as in the shorthand notation in Section 3.2, but instead the indices of the particular variable on the staggered grid. Using this notation, we discretize Equation 3.12 using the staggered derivative operators in Equation 3.13, which results in the following equations for the x variables on the staggered grid:

$$\begin{aligned}f_{x_i} = & -\frac{6\mu + 2\lambda}{h^2}x_{i,j} + \frac{2\mu + \lambda}{h^2}(x_{i+1,j} + x_{i-1,j}) + \frac{\mu}{h^2}(x_{i,j+1} + x_{i,j-1}) \\ & + \frac{\mu}{h^2}(y_{i+1,j+1} - y_{i+1,j-1} - y_{i-1,j+1} + y_{i-1,j-1}) .\end{aligned}\tag{3.14}$$

Performing this same procedure for the y variables results in a similar discretization:

$$\begin{aligned}f_{y_i} = & -\frac{6\mu + 2\lambda}{h^2}y_{i,j} + \frac{2\mu + \lambda}{h^2}(y_{i+1,j} + y_{i-1,j}) + \frac{\mu}{h^2}(y_{i,j+1} + y_{i,j-1}) \\ & + \frac{\mu}{h^2}(x_{i+1,j+1} - x_{i+1,j-1} - x_{i-1,j+1} + x_{i-1,j-1}) .\end{aligned}\tag{3.15}$$

The vertex labeling in these equations is shown graphically in Figure 3.1. Equation 3.14 and Equation 3.15 can also be represented as a stencil, which in the two-dimensional case is shown in Figure 3.2.

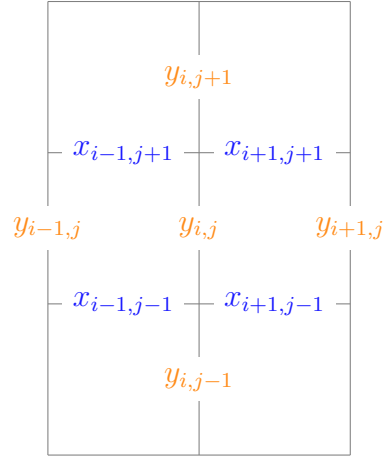
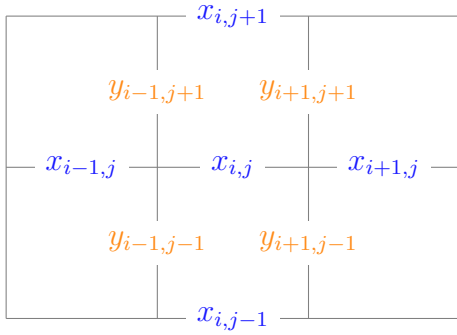


Figure 3.1: Arrangement of the staggered variables on the x and y grids.

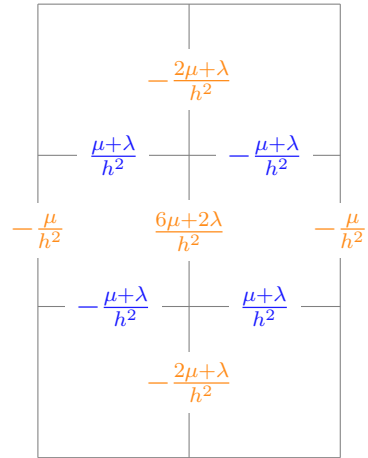
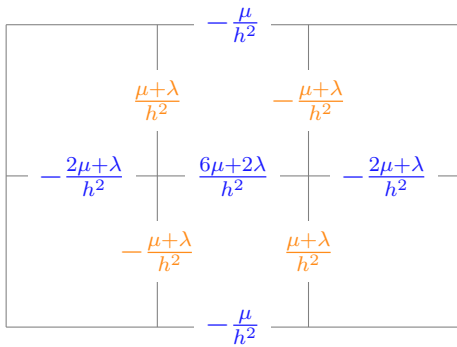


Figure 3.2: Visual representation of the stencils from the discretized PDE on both the staggered x and y grids.

3.4 Boundary Conditions

The boundary conditions used in this thesis are Dirichlet and Neumann boundary conditions. These are also referred to as fixed and traction boundary conditions, respectively. We denote the Dirichlet boundary as Γ_D , and the Neumann boundary as Γ_N . Fixed boundary conditions are where the resulting deformed material point is set to a fixed value, and so can be written as $\phi_i(\mathbf{X}) = c(\mathbf{X})$ for some prescribed function $c(\mathbf{X})$. Traction boundary conditions have the form $\mathbf{P}(\mathbf{X})\mathbf{N} = \mathbf{t}$, where $\mathbf{P}(\mathbf{X})$ is the stress tensor at the material point \mathbf{X} on the boundary, \mathbf{t} is the traction at \mathbf{X} , and \mathbf{N} is the surface normal in the undeformed configuration. Such boundary conditions describe the case where the traction (i.e., the force per unit surface area) along the boundary is known, rather than its position. The case where $\mathbf{t} = \mathbf{0}$ corresponds to a free boundary.

To apply Dirichlet boundary conditions, the known value of c is simply substituted in for the relevant grid variables in the discrete equations above (i.e., in Equations 3.14 and 3.15). To apply Neumann boundary conditions, if the normal \mathbf{N} is strictly axis-aligned, one can likewise directly substitute the value of \mathbf{t} for a corresponding finite difference expression in the discrete equations. However, non-axis-aligned Neumann boundary conditions are more difficult; we propose a solution in Chapter 4.

Given the spatial discretization described in Section 3.2 and the boundary conditions above, we can assemble a linear system of the form

$$\mathbf{L} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f_x \\ f_y \end{bmatrix}, \quad (3.16)$$

where \mathbf{L} is the discrete form of the linear operator in Equation 3.11, x and y are the vectors of unknown staggered positions in the deformed configuration, and f_x and f_y are the vectors of external forces applied to their corresponding material points. This expression is the discrete form of the continuous PDE given by Equation 3.12.

3.5 Dynamics

The discretization above does not include time, and is solving for only the resulting deformation from equilibrium of forces in the static case. However, time-varying effects produce more interesting results that are often sought after in computer graphics. The discretization of the dynamic PDE can be structured to use the same linear operator defined in Equation 3.12, which follows the formulation in Zhu et al. [26].

To implement time-dependent behaviour, we need to discretize the time derivatives of position (ϕ , which is also the deformation map) and velocity (\mathbf{v}):

$$\frac{\partial}{\partial t} \begin{bmatrix} \phi \\ \mathbf{v} \end{bmatrix} = \begin{bmatrix} \mathbf{v} \\ \frac{1}{\rho} (\mathbf{f}_{\text{internal}}(\phi, \mathbf{v}) + \mathbf{f}_{\text{external}}) \end{bmatrix}, \quad (3.17)$$

where $\mathbf{f}_{\text{internal}}(\phi, \mathbf{v})$ is the internal elastic forces, and ρ is the density of the material. Discretizing Equation 3.17 with backward Euler gives us the following equations:

$$\begin{cases} \frac{1}{\delta t} (\phi^{k+1} - \phi^k) = \mathbf{v}^{k+1} \\ \frac{1}{\delta t} (\mathbf{v}^{k+1} - \mathbf{v}^k) = \frac{1}{\rho} (\mathbf{f}_{\text{internal}}(\phi^{k+1}, \mathbf{v}^{k+1}) + \mathbf{f}_{\text{external}}) \end{cases}, \quad (3.18)$$

where δt is the timestep size, k indicates the current timestep, and $k + 1$ is the next time step. The discretization of the internal forces in Equation 3.18 is the discretized linear elastic forces from Equation 3.12. The inclusion of the velocity as a parameter to $\mathbf{f}_{\text{internal}}$ here allows for a damping force to be included in the internal forces. Our model does not use a damping force, which simplifies the expression for the discretized internal elastic forces,

$$\mathbf{f}_{\text{internal}}(\phi^{k+1}, \mathbf{v}^{k+1}) = \mathcal{L}\phi^{k+1}. \quad (3.19)$$

Then, rearranging Equation 3.18, with the internal elastic forces from Equation 3.19, we get:

$$\begin{cases} \phi^{k+1} = \phi^k + \delta t \mathbf{v}^{k+1} \\ \mathbf{v}^{k+1} = \mathbf{v}^k + \frac{\delta t}{\rho} (\mathcal{L}\phi^{k+1} - \mathbf{f}_{\text{external}}) \end{cases}. \quad (3.20)$$

3.6 Geometric Representation

In Chapter 4, we describe a variational approach to discretize Equations 3.12 and 3.20. In the resulting discretization, we use *volume fractions* to approximate the object's surface. Instead of requiring a mesh, or another explicit representation for an object to simulate, we can use an implicit representation of the surface that allows us to test sample points to see if they are inside or outside the object. As a result, we only require an indicator function for a given object in order to simulate elastic deformation.

Chapter 4

Variational Approach to Linear Elasticity

We have summarized the governing equations and staggered discretization in Chapter 3. As we noted, that approach is effective for axis-aligned geometries, where all surface normals are parallel to one of the grid axes. However, for more general, non-axis-aligned geometry, the traction boundary conditions are challenging to discretize. Instead, we adopt a variational perspective on the same equations from Chapter 3 to apply the traction boundary conditions. This alternative formulation simplifies the implementation of traction boundary conditions on the staggered grid. This style of optimization perspective has become increasingly popular in computer graphics (see Gast et al. [6]).

4.1 Minimization form of the PDE

To construct a minimization form of the linear elasticity PDE, there are several contributions to the total energy of the system that must be considered. We begin by considering the expression for the elastic strain energy,

$$\Psi(\mathbf{F}) = \mu \boldsymbol{\epsilon} : \boldsymbol{\epsilon} + \frac{\lambda}{2} \text{tr}^2(\boldsymbol{\epsilon}) = \mu \|\boldsymbol{\epsilon}\|_F^2 + \frac{\lambda}{2} \text{tr}^2(\boldsymbol{\epsilon}), \quad (4.1)$$

where the Lamé parameters λ and μ are constant over the object. This expression for the strain energy can be integrated to give the total strain energy over the object being deformed:

$$\iint_{\Omega} \Psi(\mathbf{F}) dV . \quad (4.2)$$

Equation 4.2 captures the interior stresses in the object. Further, since the homogeneous Neumann boundary conditions are the natural boundary conditions for this energy, they do not need to be explicitly enforced through any additional constraints or equations. In addition to the strain energy, we need to incorporate the work done by the external forces on the elastic object into our minimization problem. Recalling that work is defined as force times distance, it can be expressed by the following integral,

$$\iint_{\Omega} \boldsymbol{\phi} \cdot \mathbf{f}_{\text{external}} dV . \quad (4.3)$$

Next, in order to incorporate non-homogeneous Neumann boundary conditions into this variational approach, we consider the work done by the imposed traction over the boundary of the object,

$$\int_{\partial\Omega} \boldsymbol{\phi} \cdot \mathbf{t} dA = \int_{\partial\Omega} \boldsymbol{\phi} \cdot \mathbf{P}(\mathbf{X}) \mathbf{N} dA , \quad (4.4)$$

where the imposed traction boundary condition is written as $\mathbf{P}(\mathbf{X}) \mathbf{N} = \mathbf{t}$. This is only a surface integral since the boundary conditions are effectively forces applied only to the outer boundary of the solid. For simplicity, we assume that the traction boundary conditions are given in the form of the tensor \mathbf{P} . Finally, the last energy term we need to consider is for time-dependent effects:

$$\frac{1}{2} \iint_{\Omega} \left\| \boldsymbol{\phi}^{k+1} - \hat{\boldsymbol{\phi}}^k \right\|^2 dV , \quad (4.5)$$

where $\hat{\boldsymbol{\phi}}^k = \boldsymbol{\phi}^k + \delta t \mathbf{v}^k$, and $\boldsymbol{\phi}^k$ is the deformation map at the k-th timestep. The final combined optimization problem representing the PDE in Equation 3.20 is therefore,

$$\min_{\boldsymbol{\phi}^{k+1}} \frac{\rho}{2\delta t^2} \iint_{\Omega} \left\| \boldsymbol{\phi}^{k+1} - \hat{\boldsymbol{\phi}}^k \right\|^2 dV + \left[\iint_{\Omega} \boldsymbol{\Psi}(\mathbf{F}) dV + \iint_{\Omega} \boldsymbol{\phi} \cdot \mathbf{f}_{\text{external}} dV + \int_{\partial\Omega} \boldsymbol{\phi} \cdot \mathbf{P}(\mathbf{X}) \mathbf{N} dA \right] . \quad (4.6)$$

If the rightmost terms are discretized using $\boldsymbol{\phi}^{k+1}$, the resulting timestepping method is backwards Euler. If the discretizations of these terms uses $\boldsymbol{\phi}^k$ instead, we get explicit Euler [3].

4.2 Static and Quasi-static Problem

In the static case, we do not need to consider the change in time-dependent terms, since we are simply solving for the equilibrium of forces. In this section, we will also only consider fixed and zero-traction boundary conditions. Quasi-static problems are a series of

static simulations done in sequence and therefore have no time-dependent terms. This is a common approximation for stiffer elastic materials where elastic waves propagate rapidly and may be visually unimportant. This simplifies the minimization problem to:

$$\min_{\phi} \iint_{\Omega} \Psi(\mathbf{F}) + \iint_{\Omega} \phi \cdot \mathbf{f}_{\text{external}} . \quad (4.7)$$

Expanding $\Psi(\mathbf{F})$ in Equation 4.7 using the definition of the small strain tensor in Equation 3.3 gives the expression:

$$\begin{aligned} \Psi(\mathbf{F}) &= \frac{1}{4} \left\| \begin{bmatrix} 2x_1 - 2 & x_2 + y_1 \\ y_1 + x_2 & 2y_2 - 2 \end{bmatrix} \right\|_F^2 + \frac{\lambda}{2} \text{tr}^2 \left(\frac{1}{2} \begin{bmatrix} 2x_1 - 2 & x_2 + y_1 \\ y_1 + x_2 & 2y_2 - 2 \end{bmatrix} \right) \\ &= \frac{\mu}{4} (4(x_1 - 1)^2 + 4(y_2 - 1)^2 + 2(x_2 + y_1)^2) + \frac{\lambda}{2} (x_1 + y_2 - 2)^2 , \end{aligned}$$

where $x_1 = \frac{\partial \phi_1}{\partial X}$ and $x_2 = \frac{\partial \phi_1}{\partial Y}$, and $y_1 = \frac{\partial \phi_2}{\partial X}$ and $y_2 = \frac{\partial \phi_2}{\partial Y}$. Then using this expanded form of $\Psi(\mathbf{F})$, we can expand the integral in Equation 4.7, and then discretize each of these integrals separately:

$$\iint_{\Omega} \Psi(\mathbf{F}) = \frac{\mu}{2} \iint_{\Omega} (2(x_1 - 1)^2 + 2(y_2 - 1)^2 + (x_2 + y_1)^2) + \frac{\lambda}{2} \iint_{\Omega} (x_1 + y_2 - 2)^2 . \quad (4.8)$$

To preserve the sparsity inherent in the standard staggered grid scheme of Chapter 3, we approximate the derivatives here using staggered finite differences on the same grid layout, with a simple piecewise constant, Riemann sum-like strategy to approximate the integrals in Equation 4.8. Approximating each integral with the appropriate sum gives the following expressions:

$$\begin{aligned} \iint_{\Omega} 2(x_1 - 1)^2 &\approx 2 \sum_{i,j} V_{i+\frac{1}{2},j} \left(\frac{1}{h} (x_{i+1,j} - x_{i,j}) - 1 \right)^2 \\ \iint_{\Omega} 2(y_2 - 1)^2 &\approx 2 \sum_{i,j} V_{i,j+\frac{1}{2}} \left(\frac{1}{h} (y_{i,j+1} - y_{i,j}) - 1 \right)^2 \\ \iint_{\Omega} (x_2 + y_1)^2 &\approx \sum_{i,j} V_{i,j+\frac{1}{2}} \left(\frac{1}{h} (x_{i,j+1} - x_{i,j}) + \frac{1}{h} (y_{i+1,j+1} - y_{i-1,j+1}) \right)^2 , \end{aligned} \quad (4.9)$$

where we use $V_{i,j}$ to denote the volume of material in the $h \times h$ area around the i, j sample point, as shown in Figure 4.1. Thus, for cells that are only partially filled with material, these fractional cell volumes implicitly account for the surface of the object.

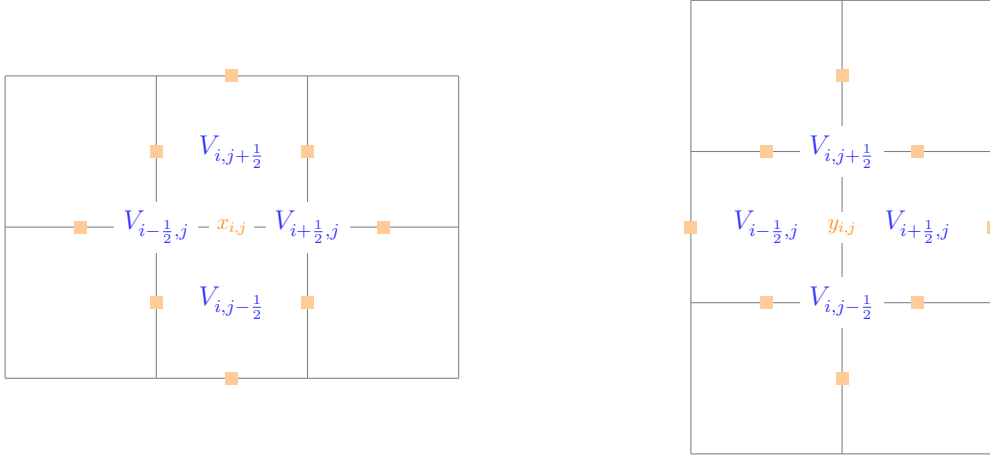


Figure 4.1: Layout of volume fraction terms relative to the underlying grid. Each $V_{i,j}$ term indicates the centre position of an $h \times h$ control volume over which the volume fractions are estimated.

To solve the overall optimization problem, we can observe that it is quadratic; therefore, we can differentiate with respect to the unknown variables $x_{i,j}$ and $y_{i,j}$, and equate the resulting expression(s) to zero to arrive at a linear system whose solution yields the optimizer. Given this strategy, we can now differentiate, one by one, each of the terms of the discrete energy listed in Equation 4.9,

$$\begin{aligned}
& 2 \frac{\partial}{\partial x_{i,j}} \sum_{i,j} V_{i+\frac{1}{2},j} \left(\frac{1}{h} (x_{i+1,j} - x_{i,j}) - 1 \right)^2 \\
& \quad = \frac{4V_{i-\frac{1}{2},j}}{h} (x_{i,j} - x_{i-1,j}) - \frac{4V_{i+\frac{1}{2},j}}{h} (x_{i+1,j} - x_{i,j}) + 4 \left(V_{i+\frac{1}{2},j} - V_{i-\frac{1}{2},j} \right) \\
& \frac{\partial}{\partial x_{i,j}} \sum_{i,j} \frac{V_{i,j+\frac{1}{2}}}{h^2} (x_{i,j+1} - x_{i,j} + y_{i,j+1}^*)^2 \\
& \quad = \frac{2V_{i,j+\frac{1}{2}}}{h^2} (x_{i,j} - x_{i,j-1} + y_{i,j-1}^*) - \frac{2V_{i,j+\frac{1}{2}}}{h^2} (x_{i,j+1} - x_{i,j} + y_{i,j+1}^*) ,
\end{aligned} \tag{4.10}$$

where we define $y_{i,j}^* = y_{i+1,j} - y_{i-1,j}$ for brevity. To discretize Expression 4.3, we begin by approximating the integral with a weighted sum as in Equation 4.9:

$$\iint_{\Omega} \phi \cdot \mathbf{f}_{\text{external}} dV \approx \sum_{i,j} V_{i,j} x_{i,j} f_{x_{i,j}} + \sum_{i,j} V_{i,j} y_{i,j} f_{y_{i,j}} \tag{4.11}$$

Differentiating Equation 4.11 with respect to a particular $x_{i,j}$ and $y_{i,j}$, as in Equation 4.10, gives the following expression:

$$\begin{aligned}\frac{\partial}{\partial x_{i,j}} \sum_{i,j} V_{i,j} x_{i,j} f_{x_{i,j}} &= \sum_{i,j} V_{i,j} f_{x_{i,j}} \\ \frac{\partial}{\partial y_{i,j}} \sum_{i,j} V_{i,j} y_{i,j} f_{y_{i,j}} &= \sum_{i,j} V_{i,j} f_{y_{i,j}}\end{aligned}\tag{4.12}$$

We can now write an expression for a single line of the linear system representing the solution to the static minimization problem in Equation 4.7. This can be written in terms of the expanded derivatives in Equation 4.10, multiplied by their respective coefficients, plus the force terms in Equation 4.12 set equal to zero:

$$\begin{aligned}& \frac{\mu}{2} \left[\frac{4V_{i-\frac{1}{2},j}}{h^2} (x_{i,j} - x_{i-1,j}) - \frac{4V_{i+\frac{1}{2},j}}{h^2} (x_{i+1,j} - x_{i,j}) - \frac{4}{h} (V_{i+\frac{1}{2},j} - V_{i-\frac{1}{2},j}) \right. \\ & \quad \left. + \frac{2V_{i,j-\frac{1}{2}}}{h^2} (x_{i,j} - x_{i,j-1} + y_{i+1,j-1} - y_{i-1,j-1}) - \frac{2V_{i,j+\frac{1}{2}}}{h^2} (x_{i,j+1} - x_{i,j} + y_{i+1,j+1} - y_{i-1,j+1}) \right] \\ & + \frac{\lambda}{2} \left[\frac{2V_{i-\frac{1}{2},j}}{h^2} (x_{i,j} - x_{i-1,j} + y_{i-1,j+1} - y_{i-1,j-1}) - \frac{2V_{i+\frac{1}{2},j}}{h^2} (x_{i+1,j} - x_{i,j} + y_{i+1,j+1} - y_{i+1,j-1}) \right. \\ & \quad \left. - \frac{4}{h} (V_{i-\frac{1}{2},j} - V_{i+\frac{1}{2},j}) \right] = V_{i,j} f_{x_{i,j}} .\end{aligned}\tag{4.13}$$

Similarly, for the y -component of the system, we get the following result:

$$\begin{aligned}& \frac{\mu}{2} \left[\frac{4V_{i,j-\frac{1}{2}}}{h^2} (y_{i,j} - y_{i,j-1}) - \frac{4V_{i,j+\frac{1}{2}}}{h^2} (y_{i,j+1} - y_{i,j}) - \frac{4}{h} (V_{i,j+\frac{1}{2}} - V_{i,j-\frac{1}{2}}) \right. \\ & \quad \left. + \frac{2V_{i-\frac{1}{2},j}}{h^2} (y_{i,j} - y_{i,j-1} + x_{i-1,j+1} - x_{i-1,j-1}) - \frac{2V_{i+\frac{1}{2},j}}{h^2} (y_{i,j+1} - y_{i,j} + x_{i+1,j+1} - x_{i+1,j-1}) \right] \\ & + \frac{\lambda}{2} \left[\frac{2V_{i,j-\frac{1}{2}}}{h^2} (y_{i,j} - y_{i-1,j} + x_{i+1,j-1} - x_{i-1,j-1}) - \frac{2V_{i,j+\frac{1}{2}}}{h^2} (y_{i,j+1} - y_{i,j} + x_{i+1,j+1} - x_{i-1,j+1}) \right. \\ & \quad \left. - \frac{4}{h} (V_{i,j-\frac{1}{2}} - V_{i,j+\frac{1}{2}}) \right] = V_{i,j} f_{y_{i,j}} .\end{aligned}\tag{4.14}$$

Note that, when the volume fraction terms $V_{i,j}$ are set to 1, these equations are identical to the finite difference discretization from Equation 3.14 and 3.15.

4.3 Treatment of Boundary Conditions

In the case of fixed and free boundary conditions, the treatment of boundary conditions under this variational approach is straightforward. For fixed boundary conditions, the value of ϕ is assigned the defined fixed boundary condition $\phi(\mathbf{X}) = c(\mathbf{X})$, and that value is used in the resulting equations, just as we did for standard finite differences in Section 3.4. However, free boundary conditions, where $\mathbf{P}(\mathbf{X})\mathbf{N} = \mathbf{0}$ are now implicitly solved for by the volume fractions in the discretization in Section 4.2. This is because the use of volume fractions means empty cells are ignored, full cells act like the standard elasticity discretization, and partial cells are accounted for with volume fractions. In comparison to the direct finite differences approach in Section 3.4, we are able to trivially support non-axis-aligned normals. As a result, we do not need to modify the discretization shown in Equation 4.13 and Equation 4.14. Thus, the only non-trivial boundary conditions we have left to consider are non-zero traction boundary conditions, which have the form $\mathbf{PN} = \mathbf{t}$ with $\mathbf{t} \neq \mathbf{0}$.

4.3.1 Non-Zero Traction Boundary Conditions

To incorporate non-zero traction boundary conditions into the variational formulation of the linear elasticity problem requires an energy perspective for traction boundary conditions. This is possible by considering the work done by the traction boundary condition over the boundary of the solid (as we saw in Equation 4.3) as:

$$\int_{\partial\Omega} \phi \cdot (\mathbf{P}(\mathbf{X})\mathbf{N}) \, dA . \quad (4.15)$$

Here, the deformation of the vertices is the displacement in the expression, and \mathbf{PN} is the force applied from the traction boundary condition. In the case of zero-traction boundary conditions, the product $\mathbf{PN} = \mathbf{0}$, and so the resulting integral expression is zero and thus is consistent with the zero-traction case.

The boundary integral in Equation 4.15 can be converted into a volume integral by applying the divergence theorem. The reason for this conversion instead of directly discretizing the boundary integrals is to be consistent with the use of volume integrals for the

rest of the discretization, thereby avoiding the possibility of errors due to inconsistencies in the boundary and volume integrals discretizations. Equation 4.15 can thus be written as:

$$\begin{aligned} \int_{\partial\Omega} \boldsymbol{\phi} \cdot (\mathbf{P}\mathbf{N}) \, dA &= \int_{\partial\Omega} (\mathbf{P}^T \boldsymbol{\phi}) \cdot \mathbf{N} \, dA \\ &= \iint_{\Omega} \nabla \cdot (\mathbf{P}\boldsymbol{\phi}) \, dV, \end{aligned} \quad (4.16)$$

since the first Piola-Kirchoff stress tensor is symmetric. To discretize this integral, \mathbf{P} at the material point $\mathbf{X} \in \Gamma_N$ can be rewritten for the associated stencil at i, j as

$$\mathbf{P} = \begin{bmatrix} P_{1,1} & P_{1,2} \\ P_{1,2} & P_{2,2} \end{bmatrix}. \quad (4.17)$$

Then, Equation 4.16 can be expanded as:

$$\iint_{\Omega} \left(\frac{P_{1,1}}{h} (x_{i+1,j} - x_{i,j}) + \frac{P_{1,2}}{h} (y_{i+1,j} - y_{i,j}) + \frac{P_{1,2}}{h} (x_{i,j+1} - x_{i,j}) + \frac{P_{2,2}}{h} (y_{i,j+1} - y_{i,j}) \right) dV \quad (4.18)$$

Discretizing each of these integrals as in Equation 4.9, we get the following summations:

$$\begin{aligned} \iint_{\Omega} \frac{P_{1,1}}{h} (x_{i+1,j} - x_{i,j}) \, dV &\approx \sum_{i,j} \frac{P_{1,1} V_{i+\frac{1}{2},j}}{h} (x_{i+1,j} - x_{i,j}) \\ \iint_{\Omega} \frac{P_{1,2}}{h} (y_{i+1,j} - y_{i,j}) \, dV &\approx \sum_{i,j} \frac{P_{1,2} V_{i+\frac{1}{2},j}}{h} (y_{i+1,j} - y_{i,j}) \\ \iint_{\Omega} \frac{P_{1,2}}{h} (x_{i,j+1} - x_{i,j}) \, dV &\approx \sum_{i,j} \frac{P_{1,2} V_{i,j+\frac{1}{2}}}{h} (x_{i,j+1} - x_{i,j}) \\ \iint_{\Omega} \frac{P_{2,2}}{h} (y_{i,j+1} - y_{i,j}) \, dV &\approx \sum_{i,j} \frac{P_{2,2} V_{i,j+\frac{1}{2}}}{h} (y_{i,j+1} - y_{i,j}) . \end{aligned} \quad (4.19)$$

Differentiating the x expressions with respect to $x_{i,j}$ and the y expressions with respect to

$y_{i,j}$:

$$\begin{aligned}
\frac{\partial}{\partial x_{i,j}} \sum_{i,j} \frac{P_{1,1} V_{i+\frac{1}{2},j}}{h} (x_{i+1,j} - x_{i,j}) &= \frac{P_{1,1}}{h} (V_{i-\frac{1}{2},j} - V_{i+\frac{1}{2},j}) \\
\frac{\partial}{\partial y_{i,j}} \sum_{i,j} \frac{P_{1,2} V_{i+\frac{1}{2},j}}{h} (y_{i+1,j} - y_{i,j}) &= \frac{P_{1,2}}{h} (V_{i-\frac{1}{2},j} - V_{i+\frac{1}{2},j}) \\
\frac{\partial}{\partial x_{i,j}} \sum_{i,j} \frac{P_{1,2} V_{i,j+\frac{1}{2}}}{h} (x_{i,j+1} - x_{i,j}) &= \frac{P_{1,2}}{h} (V_{i,j-\frac{1}{2}} - V_{i,j+\frac{1}{2}}) \\
\frac{\partial}{\partial y_{i,j}} \sum_{i,j} \frac{P_{2,2} V_{i,j+\frac{1}{2}}}{h} (y_{i,j+1} - y_{i,j}) &= \frac{P_{2,2}}{h} (V_{i,j-\frac{1}{2}} - V_{i,j+\frac{1}{2}}) ,
\end{aligned} \tag{4.20}$$

which gives us the final discretized contributions of the traction boundary condition in terms of volume fractions and the prescribed stress along the boundary.

4.4 Dynamics

All results in this thesis are static or quasi-static; however, for completeness, we also derive the dynamic case. The extension from the quasistatic simulation to dynamic simulation involves expanding the first integral term from Equation 4.6 and adding it to the resulting linear solve

$$\frac{1}{2} \iint_{\Omega} \left\| \boldsymbol{\phi}^{k+1} - \hat{\boldsymbol{\phi}}^k \right\|^2 dV .$$

Beginning by separating the x and y components of $\boldsymbol{\phi}$, the two integrals can be considered separately

$$\frac{1}{2} \left[\iint_{\Omega} (x^{k+1} - \hat{x}^k)^2 dV + \iint_{\Omega} (y^{k+1} - \hat{y}^k)^2 dV \right] . \tag{4.21}$$

Thus, it suffices to show the discretization for the time-varying component of the minimization problem for just the x component.

Taking the integrand from Equation 4.21 as the sum of the volume of each cell multiplied by the integrated gives the following summation:

$$\frac{1}{2} \iint_{\Omega} (x^{k+1} - \hat{x}^k)^2 dV \approx \frac{1}{2} \sum_{i,j} V_{i,j} (x_{i,j}^{k+1} - \hat{x}_{i,j}^k)^2 . \tag{4.22}$$

As in Equation 4.10, we differentiate Equation 4.22 with respect to x_i^{k+1} to get the following expression:

$$\frac{1}{2} \frac{\partial}{\partial x_{i,j}} \left[\sum_{i,j} V_{i,j} (x_{i,j}^{k+1} - \hat{x}_{i,j}^k)^2 \right] = \sum_{i,j} V_{i,j} (x_{i,j}^{k+1} - \hat{x}_{i,j}^k) . \quad (4.23)$$

Similarly for the y -expression:

$$\frac{1}{2} \frac{\partial}{\partial y_{i,j}} \left[\sum_{i,j} V_{i,j} (y_{i,j}^{k+1} - \hat{y}_{i,j}^k)^2 \right] = \sum_{i,j} V_{i,j} (y_{i,j}^{k+1} - \hat{y}_{i,j}^k) . \quad (4.24)$$

To compute the velocity vector at the next time step, \mathbf{v}^{k+1} , we use the difference from the previous and next time step positions:

$$\mathbf{v}^{k+1} = \frac{1}{\delta t} (\boldsymbol{\phi}^{k+1} - \boldsymbol{\phi}^k) \quad (4.25)$$

Chapter 5

Results

5.1 Convergence Studies

To verify the convergence of the method, we apply our method to a boundary value problem with a known analytical solution. We use the analytical solution to the linear elasticity PDE by Theillard et al. [25]. To set up the boundary value problem (BVP) for the known analytical solution, we set the Lamé parameters to $\mu = \lambda = 1$, and the external forces on the object to be:

$$\begin{aligned}f_x &= -8\pi^2 \cos(2\pi x) \cos(2\pi y) \\f_y &= -8\pi^2 \sin(2\pi x) \sin(2\pi y) .\end{aligned}$$

The exact solution for this BVP is:

$$\begin{aligned}u &= \cos(2\pi x) \cos(2\pi y) \\v &= \sin(2\pi x) \sin(2\pi y) .\end{aligned}$$

For fixed boundary conditions, it suffices to substitute in the exact solution for their resulting positions in the deformed configuration. For Neumann-type boundary conditions, we also need to specify the stress values associated with the appropriate traction. Using the exact solution to get an expression for the deformation gradient, the resulting stress tensor consistent with the solution of the BVP is:

$$\mathbf{P} = \begin{bmatrix} -4\pi \sin(2\pi x) \cos(2\pi y) - 4 & 0 \\ 0 & 4\pi \sin(2\pi x) \cos(2\pi y) - 4 \end{bmatrix} \quad (5.1)$$

We apply this \mathbf{P} value to enforce the non-zero traction boundaries. We consider two problems, one with an annulus with inner and outer fixed boundaries, and a second annulus

with traction boundary conditions along the outer boundary and fixed boundary conditions along the inner boundary. The annulus has an inner radius of 0.325 and an outer radius of 0.8, and the surrounding domain has a width of two units. The simulation is done with progressive grid refinements from $N = 32$ to $N = 2048$, where $dx = \frac{2}{N}$. The error in the numerical solution relative to points in the rest configuration is shown in Figure 5.1 for the all-Dirichlet boundary value problem.

To test the convergence of our method, we compare the error for simulations at different grid refinements. For the first example, we simulate an annulus with a traction outer boundary with an inner fixed boundary, consistent with the above analytical solution. For this example, we observe an approximately 1.5 order for the \mathcal{L}_1 error, and approximately 1.6 order for the \mathcal{L}_∞ error from the linear regression on the data in Table 5.1, as shown in Figure 5.2. Given we are using a second-order finite difference scheme with a first-order approximation of the boundary with volume fractions, we expected our method to have an error between first and second order.

Repeating this comparison with an analytical solution on fully fixed boundary conditions, we observe consistent second-order error. This can be seen in Figure 5.3, which shows both the \mathcal{L}_1 and \mathcal{L}_∞ error are second order. The error data from this simulation is summarized in Table 5.2. In this simulation, we expect second-order error given that we are just simulating the second-order accurate finite differencing, and volume fractions play no role at Dirichlet boundaries.

5.2 Conditioning

One of the limitations of the discretization used in Section 4.2 is that for near-incompressible solids, the linear system becomes poorly conditioned. This is because as $\nu \rightarrow 0.5$, $\lambda \rightarrow \infty$ from the definition of the material parameters in Equation 3.2. As λ increases, the dominant term of the linear operator in Equation 3.11 becomes the rank-deficient term $\lambda \nabla \nabla^T$, leading to a singular linear system.

The poor conditioning of the system can be seen in the asymptotic increase of the condition numbers in Figure 5.4. To minimize the numerical impact of poor conditioning, and given that we are working with a linear system, we use a direct LU solver. Despite the poor conditioning of the near-incompressible solids used in the simulation, we are able to numerically solve these systems. However, this poor conditioning may still pose an issue with dynamic simulations, or with the iterative solvers necessary for more complex elasticity models.

Error in Analytical Solution wrt the Rest Configuration

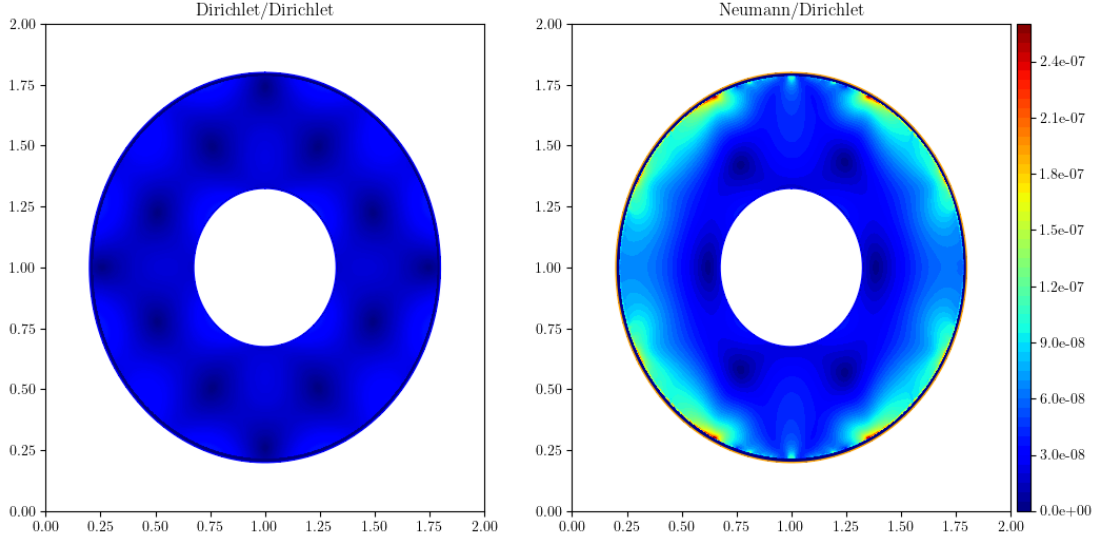


Figure 5.1: The numerical error in the static simulation with respect to the analytical solution, visualized in rest configuration of the object. The annulus is embedded in a 2048×2048 grid. The left image has all fixed boundary conditions, and the right image has traction outer boundaries and fixed interior boundaries. The fixed boundary case's error is approximately one order of magnitude lower than the traction boundary case example.

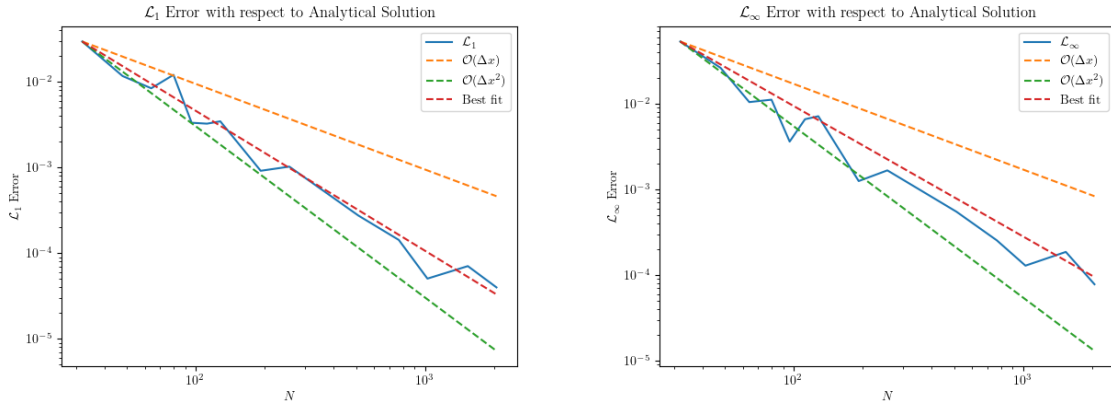


Figure 5.2: \mathcal{L}_1 and \mathcal{L}_∞ error with respect to the analytical solution for free outer boundary and fixed interior boundary.

N	\mathcal{L}_1	\mathcal{L}_∞	\mathcal{L}_1 Order	\mathcal{L}_∞ Order
32	2.97438e-02	5.32801e-02		
48	1.17518e-02	2.62320e-02	2.29	1.75
64	8.46099e-03	1.04549e-02	1.14	3.2
80	1.21437e-02	1.11503e-02	-1.62	-0.289
96	3.34825e-03	3.61626e-03	7.07	6.18
112	3.26758e-03	6.59383e-03	0.158	-3.9
128	3.47968e-03	7.15117e-03	-0.471	-0.608
192	9.20504e-04	1.25111e-03	3.28	4.3
256	1.02894e-03	1.66744e-03	-0.387	-0.999
512	2.76219e-04	5.49338e-04	1.9	1.6
768	1.43497e-04	2.54249e-04	1.62	1.9
1024	5.07250e-05	1.28759e-04	3.61	2.37
1536	7.09758e-05	1.86181e-04	-0.828	-0.91
2048	4.00599e-05	7.80069e-05	1.99	3.02

Table 5.1: \mathcal{L}_1 and \mathcal{L}_∞ error for different grid resolutions with respect to analytical solution with free exterior and fixed interior boundary conditions.

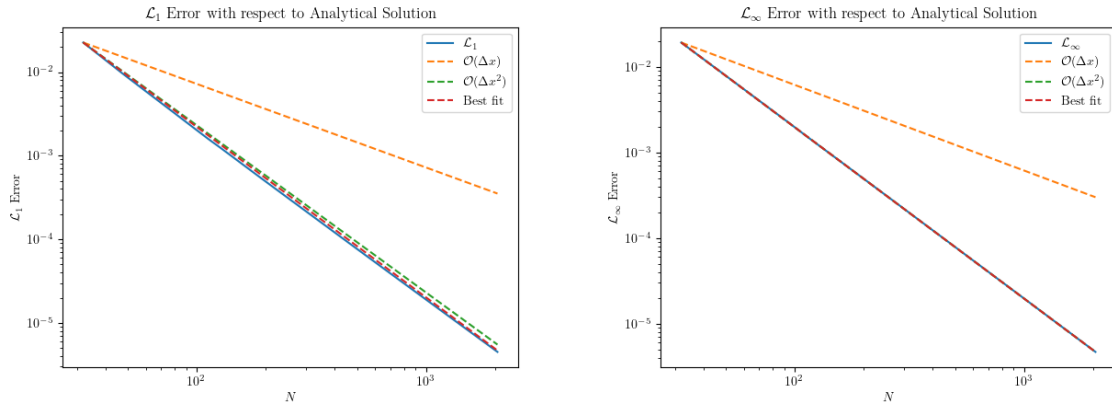


Figure 5.3: \mathcal{L}_1 and \mathcal{L}_∞ Error with respect to the analytical solution for fixed interior and exterior boundaries.

N	\mathcal{L}_1	\mathcal{L}_∞	\mathcal{L}_1 Order	\mathcal{L}_∞ Order
32	2.25809e-02	1.92147e-02		
48	9.39442e-03	8.55514e-03	2.16	2.0
64	5.14979e-03	4.81527e-03	2.09	2.0
80	3.23144e-03	3.08267e-03	2.09	2.0
96	2.20282e-03	2.14108e-03	2.1	2.0
112	1.59608e-03	1.56329e-03	2.09	2.04
128	1.22219e-03	1.19874e-03	2.0	1.99
192	5.32852e-04	5.34258e-04	2.05	1.99
256	2.97383e-04	3.00812e-04	2.03	2.0
512	7.31684e-05	7.51774e-05	2.02	2.0
768	3.23366e-05	3.34143e-05	2.01	2.0
1024	1.81419e-05	1.87890e-05	2.01	2.0
1536	8.03510e-06	8.34482e-06	2.01	2.0
2048	4.50796e-06	4.69023e-06	2.01	2.0

Table 5.2: \mathcal{L}_1 and \mathcal{L}_∞ error for different grid resolutions with respect to analytical solution with fixed boundary conditions.

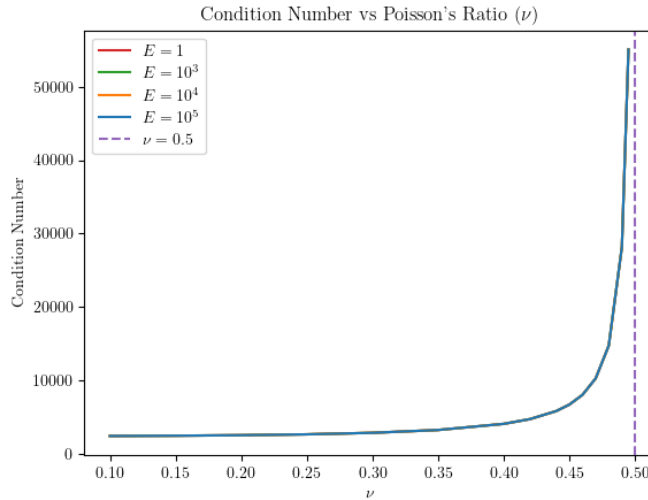


Figure 5.4: Condition numbers of static simulation with varying material parameters of ν from 0.1 to 0.495 and E from 1 to 10^5 . As ν approaches 0.5, we can see the condition number of the system increases asymptotically, while varying E does not substantially affect the conditioning of the system.

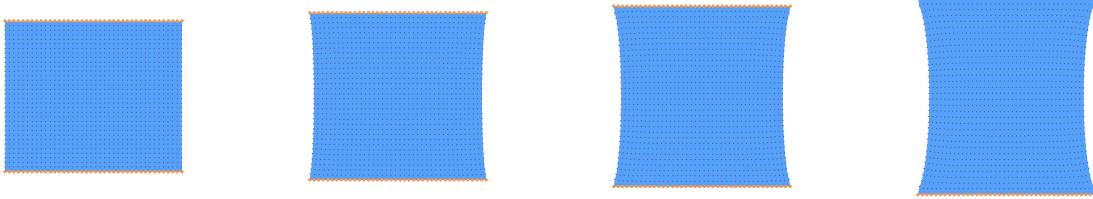


Figure 5.5: (a)-(d) Quasistatic simulation of an axis-aligned expansion deformation. The bottom and top edges of the rectangle are pulled outwards, causing the free boundaries to be pulled inwards. Fixed boundary points are coloured orange.

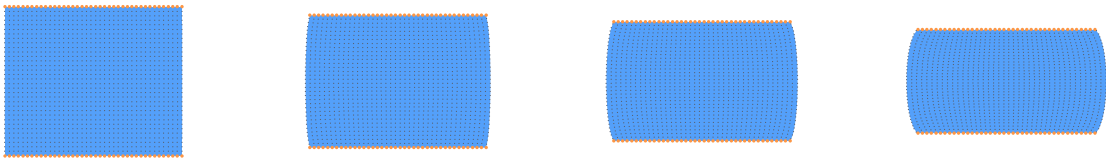


Figure 5.6: (a)-(d) Quasistatic simulation of an axis-aligned expansion deformation. The bottom and top edges of the rectangle are pushed inwards. The left and right free boundaries bulge outward in response to the compression. Fixed boundary points are coloured orange.

5.3 Static and Quasistatic Simulation

To verify we can simulate common deformations, we demonstrate simple shear, compression, and expansion deformations. For all simulations in this section, we use $\nu = 0.3$ and $E = 0.1$ as the material parameters. For expansion, we use free boundaries on the left and right sides of a rectangle, with fixed boundary conditions on the top and bottom edges. To simulate the expansion, we move the fixed boundaries outwards. The results of this deformation can be seen in Figure 5.5. Similarly, to simulate compression we use the same experimental setup as the expansion example, but instead move the top and bottom boundaries inwards, resulting in bulging as the simulation progresses. This compression simulation is shown in Figure 5.6. To simulate a simple shear with our method, we move the top and bottom edges of the object in opposite directions (i.e., the top boundary is

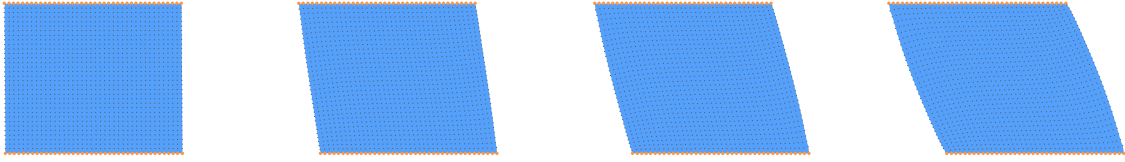


Figure 5.7: (a)-(d) Quasistatic simulation of a simple shear deformation. The bottom and top edges of the rectangle are pulled tangentially in opposite directions. Note that the material bulging slightly on the free left and right sides is an artifact due to the linear elasticity model. Fixed boundary points are coloured orange.

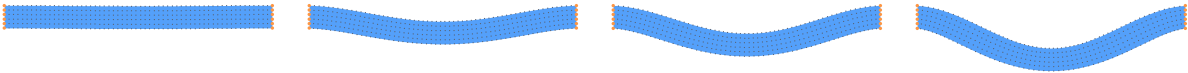


Figure 5.8: (a)-(d) Rectangular object with fixed ends undergoing deformation due to gravity under increasing gravitational forces. The total gravitational force on each object is approximately 0, 0.42, 0.73, and 1.11. Fixed boundary points are coloured orange.

shifted left and the bottom boundary is shifted right). The resulting deformation is shown in Figure 5.7.

To demonstrate the affect of gravity on an elastic object, we simulate a rectangular object with increasing gravitational forces. The boundary conditions for the rectangle are fixed along the left and right edges, with the top and bottom boundaries being free boundaries. The resulting deformation is shown in Figure 5.8.

Deformation with Irregular Geometry

We also test the effectiveness of our method with geometry for which the rest configuration shape does not align with grid edges and instead has smooth curved geometry cutting

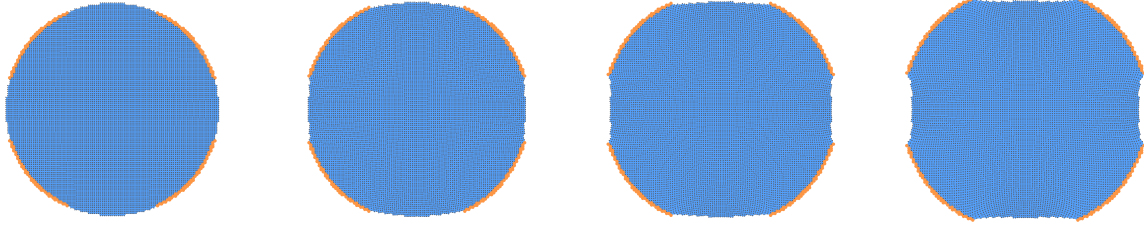


Figure 5.9: (a)-(d) Quasistatic simulation of a circular elastic object undergoing continuous deformation. Portions of the edge of the circle are pulled outwards, with free boundary conditions between each segment. Fixed boundary points are coloured orange.

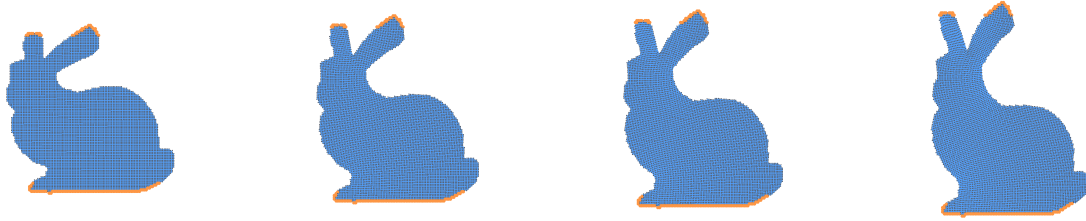


Figure 5.10: (a)-(d) Quasistatic simulation of a bunny undergoing extension, with fixed boundaries at the top of the ears and feet and free elsewhere. Fixed boundary points are coloured orange.

through cells. The purpose is to observe the effectiveness of the volume fractions in handling non-axis-aligned boundaries. We refer to such geometries as irregular.

For an example of quasistatic simulation with irregular geometry, we deform a circle by moving fixed boundary conditions on the corners of a circle outward as the simulation progresses. All other boundaries have zero traction (i.e., are free boundaries) and the simulation is done with no gravitational forces on the objects. The resulting deformation is shown in Figure 5.9.

We can also simulate geometries that are not described by simple analytical expressions. In Figure 5.10, we deform a bunny by stretching it vertically. In this example, we have fixed boundary conditions at the top and bottom of the bunny model, with free boundaries everywhere else, and move the fixed boundary condition values outwards as the simulation progresses. Once again, thanks to our discretization, the free boundaries deform naturally

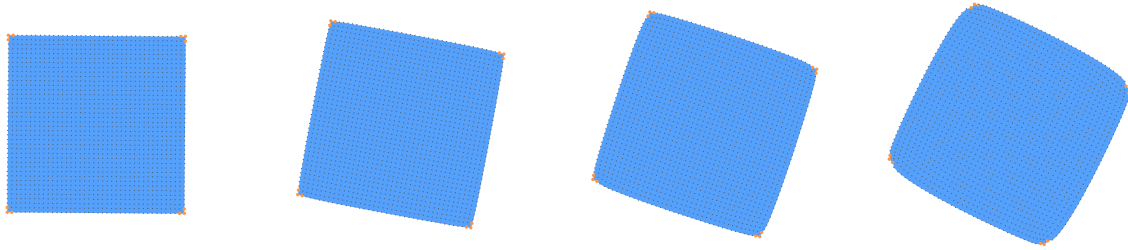


Figure 5.11: (a)-(d) Quasistatic simulation of a simple rotation of a square solid with a rotation angle from $\theta = 0$ to $\frac{\pi}{4}$. Artifacts due to the linear elasticity model are visible, such as volume inflation. These artifacts worsen as the angle θ increases. Fixed boundary points are coloured orange.

without the need for a conforming mesh or explicit computation of the surface contour or normals.

Limitations

One of the main limitations of our method is due to the use of the linear elasticity model. The linear elasticity model is not rotationally invariant (i.e., rotating the object causes a deformation that should not occur), and so simulations with rotations result in visible artifacts [16]. To demonstrate this limitation, we can see considerable volume inflation artifacts in Figure 5.11. Note that, in this simulation, there should be no deformation of the solid as we are performing a uniform rotation. These artifacts are similar to those seen in Figure 5.8 and Figure 5.7, which has larger volume inflation artifacts as the deformation increases.

Chapter 6

Conclusion

In summary, we have demonstrated a variational finite difference method for linear elasticity simulation on a staggered grid, using the energy formulation of the system to implicitly handle free boundary conditions. The result is a staggered grid-based simulation method that can be used to simulate geometries that are difficult to mesh or with non-axis-aligned normals that produces approximately 1.5 order convergence with grid refinement.

6.1 Limitations

Similarly to the pre-augmented system described by Zhu et al. [26] in their staggered grid linear elasticity solver, as the material approaches incompressibility, the system becomes poorly conditioned, as shown in Section 5.2. In prior work, Zhu et al. [26] ensure the stability of the system by using an augmented discretization, and applying such a discretization to the method described in Chapter 4 should minimize the instabilities encountered with a high Poisson’s ratio. This would require modifying our minimization formulation to account for the additional pressure variable they introduce. Given the close relationship between elasticity and viscous flow, one approach would be to take inspiration from the variational formulation by Larionov et al. [11] of the unsteady Stokes equations.

Additionally, the linear elasticity model used for the discretization is not as accurate as other nonlinear models. The main issues with linear elasticity are that it is not rotationally invariant (i.e., the deformation of the object is dependent on its rotation relative to the coordinate system), and it is only accurate for small deformations. Artifacts due to the model become visible upon rotation or large deformation. To remove these artifacts, changing the linear model for a more complex nonlinear model is required.

6.2 Extensions

6.2.1 3D Discretization

The simplest extension for this work is to implement the discretization in three dimensions. The discretization is similar to that given in Chapter 4, but with additional degrees of freedom and derivatives in the z direction. In addition, the volume fractions become sampled volumes instead of two-dimensional areas, but the resulting discretization is otherwise very similar to the two-dimensional case. Further, the dynamics case will be a straightforward generalization given the derivations given in Section 4.4.

6.2.2 Multigrids and Spatial Adaptive Grids

The use of a regular grid makes the use of a multigrid method or adaptive grids a natural extension. Multigrid methods have been used successfully with the staggered grid structure in the work by Zhu et al. [26], which uses multigrid to improve the convergence and performance of their method. In work by Goldade et al. [8], the use of spatially adaptive grids for the viscosity equations (with quadtrees or octrees) can be used to have a high number of degrees of freedom on the surface of the object, but fewer in areas that require less detail. By adapting this adaptive structure to the staggered grid elasticity method in this work, the result would be a simulation that better preserves detail than a coarse simulation, but with improved performance over a simulation done on the finest grid level. Both of these methods should be applicable to this method, with the result being improved performance.

6.2.3 Alternative Elasticity Models

Given that linear elasticity has artifacts for large deformation and is not rotationally invariant, using a nonlinear discretization is a natural extension to remove these artifacts. The simplest nonlinear model that is rotationally invariant is corotational elasticity [16]. However, due to the nonlinear nature of the model, the resulting discretization will be nonlinear as well. This will complicate the minimization problem, as it may not be possible to directly solve via the single linear system based on the differentiation step in Equation 4.9. Additionally, more complex models, such as Neo-Hookean elasticity [20], could be applied to this method as well. Additional work will be required to solve this minimization problem, such as using an iterative solver (e.g., Newton's method), and by adapting our discretization approach to be compatible with these nonlinear constitutive models.

References

- [1] Bathe, K. (2006). *Finite Element Procedures*. Prentice Hall.
- [2] Batty, C., Bertails, F., and Bridson, R. (2007). A fast variational framework for accurate solid-fluid coupling. *ACM Trans. Graph.*, 26(3):100–es.
- [3] Batty, C. and Bridson, R. (2008). Accurate Viscous Free Surfaces for Buckling, Coiling, and Rotating Liquids. *Eurographics Association*, pages 219–228.
- [4] Bro-Nielsen, M. and Cotin, S. (1996). Real-time Volumetric Deformable Models for Surgery Simulation using Finite Elements and Condensation. *Computer Graphics Forum*, 15(3):57–66. eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/1467-8659.1530057>.
- [5] Feldman, B. E., O’Brien, J. F., and Klingner, B. M. (2005). Animating Gases with Hybrid Meshes. *Los Angeles*.
- [6] Gast, T. F., Schroeder, C., Stomakhin, A., Jiang, C., and Teran, J. M. (2015). Optimization Integrator for Large Time Steps. *IEEE Transactions on Visualization and Computer Graphics*, 21(10):1103–1115.
- [7] Georgii, J. and Westermann, R. (2008). *Corotated Finite Elements Made Fast and Stable*. The Eurographics Association.
- [8] Goldade, R., Wang, Y., Aanjaneya, M., and Batty, C. (2019). An adaptive variational finite difference framework for efficient symmetric octree viscosity. *ACM Transactions on Graphics*, 38(4):1–14.
- [9] James, D. L., Barbič, J., and Twigg, C. D. (2004). Squashing cubes: automating deformable model construction for graphics. In *ACM SIGGRAPH 2004 Sketches on - SIGGRAPH '04*, page 38, Los Angeles, California. ACM Press.

- [10] James, D. L. and Pai, D. K. (1999). ArtDefo: accurate real time deformable objects. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques - SIGGRAPH '99*, pages 65–72. ACM Press.
- [11] Larionov, E., Batty, C., and Bridson, R. (2017). Variational stokes: a unified pressure-viscosity solver for accurate viscous liquids. *ACM Transactions on Graphics*, 36(4):1–11. Publisher: Association for Computing Machinery (ACM).
- [12] Li, M., Ferguson, Z., Schneider, T., Langlois, T., Zorin, D., Panozzo, D., Jiang, C., and Kaufman, D. M. (2020). Incremental potential contact: intersection-and inversion-free, large-deformation dynamics. *ACM Transactions on Graphics*, 39(4).
- [13] Longva, A., Löschner, F., Kugelstadt, T., Fernández-Fernández, J. A., and Bender, J. (2020). Higher-order finite elements for embedded simulation. *ACM Transactions on Graphics*, 39(6):1–14.
- [14] Martin, S., Kaufmann, P., Botsch, M., Wicke, M., and Gross, M. (2008). Polyhedral Finite Elements Using Harmonic Basis Functions. *Computer Graphics Forum*, 27(5):1521–1529.
- [15] Muller, M., Teschner, M., and Gross, M. (2004). Physically-based simulation of objects represented by surface meshes. In *Proceedings Computer Graphics International, 2004.*, pages 26–33, Crete, Greece. IEEE.
- [16] Müller, M., Dorsey, J., McMillan, L., Jagnow, R., and Cutler, B. (2002). Stable real-time deformations. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation, SCA '02*, pages 49–54, New York, NY, USA. Association for Computing Machinery.
- [17] O’Brien, J. F. and Hodgins, J. K. (1999). Graphical modeling and animation of brittle fracture. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques, SIGGRAPH '99*, pages 137–146, USA. ACM Press/Addison-Wesley Publishing Co.
- [18] Patterson, T., Mitchell, N., and Sifakis, E. (2012). Simulation of complex nonlinear elastic bodies using lattice deformer. *ACM Transactions on Graphics*, 31(6):1–10.
- [19] Sifakis, E. D. (2012). FEM Simulation of 3D Deformable Solids: A practitioner’s guide to theory, discretization and model reduction. Part One: The classical FEM method and discretization methodology.

- [20] Smith, B., Goes, F. D., and Kim, T. (2018). Stable Neo-Hookean Flesh Simulation. *ACM Transactions on Graphics*, 37(2):1–15.
- [21] Stomakhin, A., Howes, R., Schroeder, C., and Teran, J. M. (2012). Energetically consistent invertible elasticity. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '12, pages 25–32, Goslar, DEU. Eurographics Association.
- [22] Sugimoto, R., Batty, C., and Hachisuka, T. (2022). Surface-Only Dynamic Deformables using a Boundary Element Method. *Computer Graphics Forum*, 41(8):75–86. [_eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.14625](https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.14625).
- [23] Teran, J., Blemker, S., Hing, V. N. T., and Fedkiw, R. (2003). Finite volume methods for the simulation of skeletal muscle. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, SCA '03, pages 68–74, Goslar, DEU. Eurographics Association.
- [24] Terzopoulos, D., Platt, J., Barr, A., and Fleischert, K. (1987). Elastically Deformable Models. *ACM SIGGRAPH Computer Graphics*, 21(4):205–214.
- [25] Theillard, M., Djodom, L. F., Vié, J.-L., and Gibou, F. (2013). A second-order sharp numerical method for solving the linear elasticity equations on irregular domains and adaptive grids – Application to shape optimization. *Journal of Computational Physics*, 233:430–448.
- [26] Zhu, Y., Sifakis, E., Teran, J., and Brandt, A. (2010). An efficient multigrid method for the simulation of high-resolution elastic solids. *ACM Transactions on Graphics*, 29(2):1–18. Publisher: Association for Computing Machinery (ACM).