b
UNIVERSITÄT
BERN

Flesh Simulation in the Field of Animation

Bachelor Thesis

submitted in fulfilment of the requirements for the degree of
Bachelor of Science (B.Sc.)

at the

University of Bern
Institute of Computer Science

1. Prüfer: Prof. Dr. David Bommes
2. Prüfer: Max Mustermann

Eingereicht von: Corina Danja Masanti
Matrikelnummer: 15-128-655
Datum der Abgabe: 01.01.2001

Vorwort

Dies ist ein Vorwort

Abstract

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Contents

Chapter 1

Introduction

"Animation offers a medium of story telling and visual entertainment which can bring pleasure and information to people of all ages everywhere in the world."
- Walt Disney

1.1 Motivation

With steadily increasing computational power, the demand of better results is constantly growing. Especially in the field of animation and simulation we are no longer happy with mediocre results. In the entertainment sector and gaming industry animation studios like Pixar© or Disney© brought us games and movies of highest quality. Both of them have made groundbreaking progress over the years. This is easily observed when we compare today's work with that from ten years ago.

As always, we have different requirements for each use. In some cases we want to exaggerate a movement or a reaction in a certain way. We can for example create a massive explosion in a movie that would not be half as spectacular in the real world.

In other scenarios we want to come as close as possible to reality. For instance, we may want an animated character to move and physically interact with its environment as a real human being would. Otherwise, the human brain would immediately recognize that some things do not add up. The goal here is to bring characters quite literally to life. We

can add small details like visible breathing and small wrinkles to have an even more convincing effect. We aim to create the illusion of a character with personality, thought and emotions. In order to achieve this effect, we need the character to move and react physically correct.

In the paper *Stable Neo-Hookean Flesh Simulation* [SGK18], the authors addressed exactly this problem of making an animated movement of a human-like character look as natural as possible. Consequently this thesis is based heavily on this paper. In the following, I will abbreviate the name of the paper with *SNH-FS*.
But before diving further into the content of the paper, a fundamental background is needed. In order to animate a physical movement, we first need to understand the physics behind it which requires some knowledge in the field of continuum mechanics. Unfortunately, for most of us it has yet to be learned. The goal of this thesis is to give the necessary physical and mathematical background for a regular computer science student to understand the field of animation. In addition I will go deeper into the thematics of the paper *SNH-FS*. I aim to get an understanding of their contribution in the field and implement their proposed energy myself.

1.2 Structure

In the following, I will give a brief overview of the necessary mathematical background and deliver an introduction into the field of continuum mechanics. Next, I will go through the ideas mentioned in the paper and include some calculations and visualisations that serve for a better understanding. Lastly I will give an insight into the process of implementing the energy.

TODO: Adjust according to additions in text.

Chapter 2

Background

As in the paper *SNH-FS*, my goal is to animate human-like characters. More precisely, the focus is on the behaviour of the character's flesh. This chapter serves as an introduction into the mathematical as well as physical background, needed to understand the upcoming calculations and conclusions. At the beginning of this chapter I will define the notation and convention used throughout this whole thesis. Next, I will give insights in the mathematics used and present some of the concepts used in continuum mechanics. However, I will not include each proof explicitly, as there are already many good resources for an interested reader.

2.1 Notation and Convention

At first I will declare the notation used in this thesis to avoid misunderstandings. I will use the common notation used in continuum mechanics taken from the book *Continuum Mechanics* [Spe80]. Additionally I will include some more specific declarations formulated and used by the authors of the paper *SNH-FS*.

2.1.1 General Notation

Scalars are represented by regular, normal-weight variables such as a, whereas tensors and matrices are represented by upper-case bold letters

such as for example $\mathbf{A}$. Vectors will be denoted by bold lower-case variables like $\mathbf{a}$.

2.1.2 Tensor Notation

Furthermore, I will use the tensor notation used in *SNH-FS*. They decided to define the vectorization $\text{vec}(\cdot)$ as column-wise flattening of a matrix into a vector ([SGK18], 12:5) similar to Golub and Van Loan (2012) [GV12].

In order to indicate that I am dealing with a vectorized matrix I will use the symbol $\check{\cdot}$ as shown in the following equation:

$$\mathbf{A} = \begin{bmatrix} a & c \\ b & d \end{bmatrix} \qquad \text{vec}(\mathbf{A}) = \check{\mathbf{a}} = \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}.$$

Additionally, I will have to deal with 4^{th} order tensors in a form of matrix-of-matrices. These matrices are denoted by using blackboard bold:

$$\mathbb{A} = \begin{bmatrix} \begin{bmatrix} a & c \\ b & d \end{bmatrix} & \begin{bmatrix} i & k \\ j & l \end{bmatrix} \\ \begin{bmatrix} e & g \\ f & h \end{bmatrix} & \begin{bmatrix} m & o \\ n & p \end{bmatrix} \end{bmatrix} = \begin{bmatrix} [\mathbf{A}_{00}] & [\mathbf{A}_{01}] \\ [\mathbf{A}_{10}] & [\mathbf{A}_{11}] \end{bmatrix}$$

We can now vectorize $\mathbb{A}$ and get the following result

$$\text{vec}(\mathbb{A}) = \left[\text{vec}\left(\mathbf{A}_{00}\right) \,\middle|\, \text{vec}\left(\mathbf{A}_{10}\right) \,\middle|\, \text{vec}\left(\mathbf{A}_{01}\right) \,\middle|\, \text{vec}\left(\mathbf{A}_{11}\right) \right] = \check{\mathbf{A}}.$$

This term above is equivalent to the subsequent notation

$$\check{\mathbf{A}} = \begin{bmatrix} a & e & i & m \\ b & f & j & n \\ c & g & k & o \\ d & h & l & p \end{bmatrix}.$$

The advantage of this form is that we can write several expressions as a cross product. I will need this property later to simplify complicated expressions and calculations.

2.1.3 Summary

Here is a quick overview of the introduced notation:

a : Scalar

$\mathbf{A}$: Matrix or tensor

$\mathbf{a}$: Vector

$\check{\mathbf{a}}$: Vectorized matrix (also written as vec($\mathbb{A}$))

$\check{\mathbf{A}}$: matrix-of-matrices

TODO: Check if all notations used later are declared.

2.2 Mathematical Background

Since mathematics play an important role in the field of interest, the first step is to build a solid background before diving further into more complex calculations. This section covers all the important concepts used later in the calculations. A basic understanding of linear algebra is assumed.

2.2.1 Matrices

In the upcoming chapters I will mainly be using matrices in the calculations. This subsection lists an overview of the matrix properties used later:

Property 1 (**Positive Definiteness**). *A matrix* $\mathbf{A}$ *is positive definite if* $\mathbf{x}^\intercal\mathbf{A}\mathbf{x}$ *is positive for all of the non-zero values of the column matrix* $\mathbf{x}$.

TODO: Is this necessary?

2.2.2 Singular Value Decomposition

The Singular Value Decomposition (SVD) will play an important role in the formulation of the deformation gradient. It represents the best possible approximation of a given matrix by a matrix of low rank. This approximation can be looked at as a compression of the data given ([LM15], S. 295).

Definition 1 (**Singular Values**). *The singular values of a matrix* $\mathbf{A} \in \mathbb{R}^{m\times n}$ *are the square roots of the eigenvalues of* $\mathbf{A}\mathbf{A}^\intercal$.

The theorem of the singular value decomposition tells us that we can factor every m-by-n matrix into one orthogonal m-by-m, one diagonal m-by-n and one orthogonal n-by-n matrix. More formally:

Theorem 1 (**The SVD Theorem**). *Let* $\mathbf{A} \in \mathbb{R}^{m\times n}$ *be a matrix having r positive singular values,* $m \geq n$. *Then there exist orthogonal matrices* $\mathbf{U} \in \mathbb{R}^{m\times m}$, $\mathbf{V} \in \mathbb{R}^{n\times n}$ *and a diagonal matrix* $\tilde{\mathbf{\Sigma}} \in \mathbb{R}^{m\times n}$ *such that*

$$\mathbf{A} = \mathbf{U}\tilde{\mathbf{\Sigma}}\mathbf{V}^\intercal$$
$$\tilde{\mathbf{\Sigma}} = \begin{bmatrix} \mathbf{\Sigma} & 0 \\ 0 & 0 \end{bmatrix}$$

where $\mathbf{\Sigma} = diag\ (\sigma_1, \sigma_2, \ldots, \sigma_r)$, *and* $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$ *are the positive singular values of* $\mathbf{A}$.

The definition and theorem were taken from [For14] (def. 1 S.113, thm. 1 S.300).

TODO: Check reference. Make paragraphs work together. Add examples.

2.2.3 Polar Decomposition

Theorem 2 (**The Polar Decomposition Theorem**). *Let* $\mathbf{F}$ *be a nonsingular square matrix. Then* $\mathbf{F}$ *can be decomposed uniquely into either of the following two products*

$$\mathbf{F} = \mathbf{RU}, \quad \mathbf{F} = \mathbf{VR}$$

where $\mathbf{R}$ *is orthogonal and* $\mathbf{U}$ *and* $\mathbf{V}$ *are positive definite symmetric matrices.*

The definition and theorem were taken from [Spe80] (thm. 2 S.12).

2.2.4 Frobenius Norm

Definition 2 (**Frobenius Norm**). *Let* $\boldsymbol{A}$ *be a* $(m \times n)$ *matrix in the real or complex space. Then the Frobenius norm is defined as*

$$\|\mathbf{A}\|_F := \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2}.$$

We can further represent the norm with the trace of the matrix, in which $\mathbf{A}^*$ is the conjugate transpose of $\mathbf{A}$. Going from there, we can use the SVD of $\mathbf{A}$ and write the norm with respect to the singular values of $\mathbf{A}$, denoted by σ_i

$$\|\mathbf{A}\|_F = \sqrt{\text{trace}(\mathbf{AA}^*)} = \sqrt{\sum_{i=1}^{\min\{m,n\}}\sigma_i^2}. \tag{2.1}$$

TODO: Complete this section. Add lemmas and theorems used afterwards in the calculations. Possible adjustments may come at the end.

2.3 Continuum Mechanics

In this section, I will give a broad introduction into the field of Continuum Mechanics. In Continuum Mechanics, we are less interested in small particles like atoms or molecules of an object but rather in pieces of matter which are in comparison very large. The reason for this is that the calculation for the behaviour of individual atoms is very difficult for larger systems. We are therefore concerned with the mechanical behavior of solids and fluids on the macroscopic scale and we treat material as uniformly distribeted throughout regions of space. With this approach, it is possible to define quantities such as displacement, density, etc. as continuous functions of position ([Spe80], p. 1).

2.3.1 Deformation

In continuum mechanics the term *strain* is used as a measure of deformation and we denote *stress* as the force per unit area. If we apply a force over an object, the object itself undergoes a deformation. This behavior is intuitively clear. Graphically, we can imagine a deformation with the help of a two dimensional deformation map as shown in Fig. 2.1. The ellipse on the left side represents an object in its rest state. A function ϕ maps this rest state of the ellipse to a deformed state as shown on the right side.

We can imagine, that we map each particle of a chosen object from its rest state to a deformed one. Each particle X of a body can be characterized by a vector $\mathbf{x}$ containing its positional coordinates at time $t_0 = 0$. This is the reference configuration. If the particles are displaced after a time t, we can describe their new coordinate vector with

$$\check{\mathbf{x}} = \phi(\mathbf{x}, t).$$

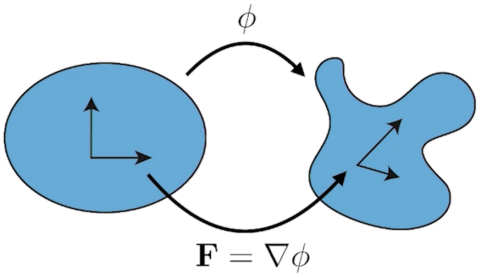


Figure 2.1: Deformation Map [Pix]

If we consider, for example a *Translation*, every particle is displaced by the same distance and direction. Hence, we can calculate $\check{\mathbf{x}}$ with

$$\check{\mathbf{x}} = \mathbf{x} + \mathbf{c}(t)$$

where $\mathbf{c}$ is a vector that only depends on t ([Spe80], p.63).

TODO: Include own image instead of the one here? Other example than translation? Explain strain and stress more?

2.3.2 Deformation Gradient

The next quantity, I will use in the upcoming chapters, is the deformation gradient $\mathbf{F}$. With its help, we can calculate the change in volume and length of an object during a deformation. We can obtain $\mathbf{F}$ by taking the gradient of the function ϕ discussed in the previous section.

As an example, I am taking a simple deformation in 2D. Consider the coloured area in Fig. 2.2. We can stretch this area in the following way:

$$\check{x} = 1.5x + 0y$$
$$\check{y} = 0x + 2y$$

The resulting deformation gradient can

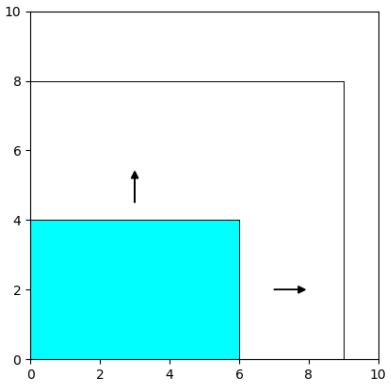


Figure 2.2: Stretching of a rectangle

then be obtained by

$$\mathbf{F} = \begin{bmatrix} \frac{\partial[1.5x+0y]}{\partial x} & \frac{\partial[1.5x+0y]}{\partial y} \\ \frac{\partial[0x+2y]}{\partial x} & \frac{\partial[0x+2y]}{\partial y} \end{bmatrix} = \begin{bmatrix} 1.5 & 0.0 \\ 0.0 & 2.0 \end{bmatrix}.$$

If $\mathbf{F}$ is equal to the identity matrix $\mathbf{I}$, there is no deformation present. This would be the case for rigid body displacements. Since we will be handling deformation in the 3D space, the deformation gradient will have the form of a 3×3 matrix:

$$\mathbf{F} = \left[f_0 \,\middle|\, f_1 \,\middle|\, f_2 \right] = \begin{bmatrix} f_0 & f_3 & f_6 \\ f_1 & f_4 & f_7 \\ f_2 & f_5 & f_8 \end{bmatrix}. \tag{2.2}$$

$\mathbf{F}$ can be factorized and used to calculate other quantities. The next few sections show these factorizations and calculations.

Singular Value Decomposition of F

Using the SVD theorem shown in Thm.1, $\mathbf{F}$ can be written in the form of

$$\mathbf{F} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\intercal} \tag{2.3}$$

in which $\mathbf{\Sigma}$ stands for

$$\mathbf{\Sigma} = \begin{bmatrix} \sigma_0 & 0 & 0 \\ 0 & \sigma_1 & 0 \\ 0 & 0 & \sigma_2 \end{bmatrix}. \tag{2.4}$$

The σ_i denote the singular values of $\mathbf{\Sigma}$. $\mathbf{U}$ and $\mathbf{V}$ are both orthogonal matrices that represent the rotation of $\mathbf{F}$. $\mathbf{\Sigma}$ on the other hand indicates the scaling of each coordinate x_i by the factor σ_i. Unlike the standard convention, where a possible reflection lies in the rotation variants (meaning $\mathbf{U}$ and $\mathbf{V}$), here the reflections are moved to $\mathbf{\Sigma}$ and it is therefore allowed to have a negative entry. This has the effect that $\det(\mathbf{U}) = \det(\mathbf{V}) = 1$.

Polar Decomposition of the Deformation Gradient

With the help of Thm.2 we can decompose the deformation gradient into the form

$$\mathbf{F} = \mathbf{RS} \tag{2.5}$$

where $\mathbf{R}$ is orthogonal and $\mathbf{S}$ is a positive definite symmetric matrix. $\mathbf{R}$ symbolises the rotation (with possible reflection) that $\mathbf{F}$ undergoes, whereas $\mathbf{S}$ contains the scaling along the orthogonal directions of $\mathbf{F}$.

Relative volume change

A useful information about a deformation is the relative volume change of the deformed object. It can be calculated by the determinate of $\mathbf{F}$

$$J = \det(\mathbf{F}). \tag{2.6}$$

For a normal deformation, J is a positive value. A determinant of zero would mean that the object is being deformed into a zero volume state, e.g. a plane or a point. A negative determinant, on the other hand, indicates an inversion.

Cauchy-Green

The right Cauchy-Green tensor $\mathbf{C}$ can be calculated by

$$\mathbf{C} = \mathbf{F}^\intercal\mathbf{F} \tag{2.7}$$

and is a 3×3 matrix for deformations in the 3D space. Using $\mathbf{C}$, we can calculate the first right Cauchy-Green invariant I_C with

$$I_C = \mathrm{tr}(\mathbf{C}). \tag{2.8}$$

This quantity describes the length change of the object after a deformation.

2.3.3 Deformation Energy

The deformation energy or strain energy Ψ is a scalar that defines the stored energy of a object undergoing a deformation. The strain energy density is then the strain energy per unit volume. It equal to the work that has to be done by the stresses in order to alter the strains ([Kor17], p.10). This means that it is an indicator of how much force must be applied in order for an object to be deformed in a certain way. We can use the deformation energy to express the relationship between the stresses and strains. We can illustrate this with a stress-strain curve:

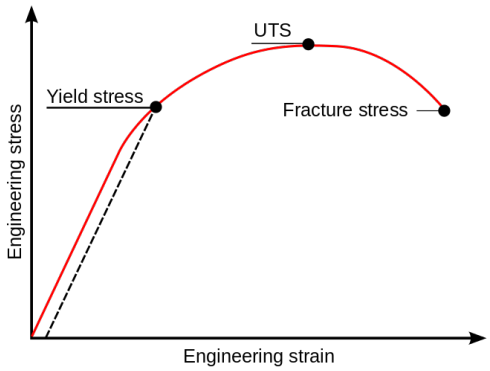


Figure 2.3: Example for a stress-strain curve

Unfortunately, this relationship is not always linear, which makes it harder to find a suitable model. For example if we reach a stress state that produces yielding, we cannot assume that the relationship is linear. If the material reaches the yield point, it can no longer recover into its rest shape due to permanent fractures during the deformation. This point is illustrated in Fig. 2.3 with *Yield stress*. *UTS* in Fig. 2.3 is abbreviated for Ultimate tensile strength and defines the maximal stress an object can bear before breaking. The area under the stress-strain curve corresponds to the strain energy.

In order to get a convincing simulation of high quality, we have to choose an appropriate energy. In the case of modelling deformations on human-like characters an elastic energy needs to be chosen. The key property that makes an energy elastic is that if all the forces that are applied over an object add up to zero, the object must come back into its rest shape. The energy then has to be minimized in order to get the results we want.

2.3.4 Material Constants

When we look at a deformation of an object, we need to consider the material the object consists of. A material can be very stiff like steel or easily deformable like rubber. In order to measure the deformation of a specific material, we need the *Poisson's Ratio* of said material. The poisson's ratio is a material constant that is defined as

$$\sigma = -\frac{\epsilon_{11}}{\epsilon_{22}} \in [-1, 0.5] \tag{2.9}$$

where ϵ_{11} is the lateral and ϵ_{22} the axial strain. The range in which σ lies in starts at -1 and goes up to 0.5 [MR09]. Usually the poisson's ratio of a material is positive. A negative value would mean that the material becomes wider in the cross section when it is stretched. This behaviour is very uncommon in nature. Examples of materials with a negative poisson's ratio are for instance discussed in *Foam structures with a negative Poisson's ratio* [Lak87] or *Advances in negative Poisson's ratio materials* [Lak93].

Here are some examples of the poisson's ratio of various materials:

Material	**Poisson's ratio**
C (graphite)	0.31
Sn (metal)	0.357
Cu	0.355
Zn	0.25
Ag	0.36
Au	0.45
Concrete	0.20–0.37
Titanium (dental alloy)	0.30–0.31
Bronze	0.34
18–8 Stainless steel	0.305
Natural rubber	0.4999
B_2O_3 glass	0.30
GeO_2 glass	0.20

Table 2.1: Different materials with their poisson's ratio taken from [MR09]

In the context of flesh simulations, I am using the poisson's ratio as a characterization for the resistance to volume change of flesh. Biological

tissues such as flesh, fat and muscles have one important property: Volume preservation. As a result the poisson's ratio takes on higher values in the range of 0.45 and 0.5 [SGK18].

The calculation of the poisson's ratio defined in Eq. 2.9 is a challenge. Fortunately, we can make use of the *Lamé Parameters*, the two material specific constants μ and λ. With the help of these two constants we can transform Eq. 2.9 into the following form:

$$\sigma = \frac{\lambda}{2(\lambda + \mu)}. \tag{2.10}$$

This equation allows to calculate the poisson's ratio much easier.

TODO: Topic generally covered, maybe adjust the following: Explain why volume preservation => high poisson's ratio. Explain how to get formula with lamé parameters, explain lamé parameters. Explain more on negative poisson's value?
Further reading: http://silver.neep.wisc.edu/ lakes/PoissonIntro.html

TODO: Add some exaples and visualisations. What exactly is it and what role does it play in a deformation process? Hyperelastic energy, connect with paper.
To include: Piola-Kirchhoff Stress, Cauchy Green invariant, polar decomposition, cauchy green tensor

Chapter 3

Stable Neo-Hookean Flesh Simulation

In this chapter, I will examine further the topic of the paper *SNH-FS*. In the interest of understanding the thought process of the authors, I will include some of their calculations more detailed. In addition, examples and visualisations are presented for a better understanding.

3.1 Deformation Gradient

In the following calculations some properties of the deformation gradient $\mathbf{F}$ covered in chapter 2 will be used. These properties are summarized in table 3.1 for a better overview:

Symbol	**Definition**
$\mathbf{F} = \mathbf{RS}$	Polar decomposition
$J = \det(\mathbf{F})$	Relative volume change
$\mathbf{C} = \mathbf{F}^\intercal\mathbf{F}$	Right Cauchy-Green
$I_C = \mathrm{tr}(\mathbf{C})$	First right Cauchy-Green invariant

Table 3.1: Quantities derived from the Deformation Gradient $\mathbf{F}$

TODO: Check if everything is explained enough in Background.

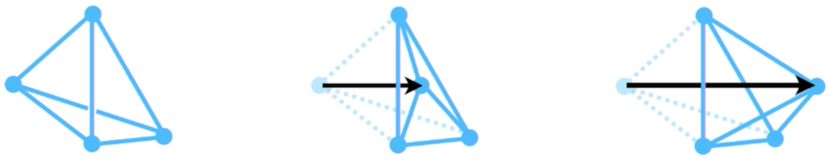
3.2 Energy Formulation

3.2.1 Stability

The core goal of the paper was to model deformations for virtual characters that have human-like features. The focus of the authors was the deformation energy. In order to achieve a convincing result the first step is to specify some requirements. Firstly the energy has to be stable. More precisely, we need a hyperelastic energy that is stable in the following four important ways:

1. Inversion Stability: Given some arbitrary object, it is possible that while deforming the object we can arrive at a zero volume state or even an entire inversion. Take for example the tetrahedron shown in Fig. 3.1a. In Fig. 3.1b we see a deformed state of this tetrahedron where the volume is scaled down to zero and we are left with a simple triangle. In Fig. 3.1c we have an inversion of the object. The deformation energy has to be able to deal with both cases. That means that the energy has to be singularity-free and does not need any filters or threshold ([SGK18], 12:3).

TODO: Explain what last sentence means. Reference correct like this?



(a) Rest state (b) Zero-volume state (c) Inversed state

Figure 3.1: Inversion of a tetrahedron [Pix]

2. Reflection stability: A reflection is a rotation around the coordinate origin. The deformation energy needs to be well behaved regardless of the reflection convention used in the singular value decomposition.

TODO: Explain a bit better what this means. What means well behaved?

3. Rest stability: When deforming an object in a certain way, we apply one or multiple forces to that object. When we achieve rest stability, we ensure that if the sum of all forces is equal to zero, the object must be back in its rest state.

4. Meta-stability under degeneracy: We can crush an object into an arbitrary shape like a plane, line or point. This process is illustrated for a cube in Fig. 3.2. The cube should be able to recover to its actual shape after the deformation.

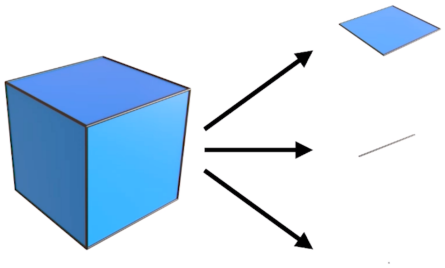


Figure 3.2: Illustration of meta stability [Pix]

Based on these four requirements, we will in the following determine if a deformation energy is suited for our needs.

TODO: Maybe add own images and more sources.

3.2.2 Existing Neo-Hookean Energies

In previous literature, a few energies were proposed that I will analyse in this section. They are listed in table 3.2 below.

Energy	**Author(s)**
$\Psi_{Neo} = \frac{\mu}{2}(I_C - 3) - \mu \log J + \frac{\lambda}{2}(\log J)^2$	e.g. Bonet and Wood 2008
$\Psi_{\mathrm{A}} = \frac{\mu}{2}(I_C - 3) - \mu \log J + \frac{\lambda}{2}(J-1)^2$	Odgen 1997
$\Psi_{\mathrm{B}} = \frac{\mu}{2}\left(J^{-2/3} I_C - 3\right) + \frac{\lambda}{2}(J-1)^2$	Bower 2009
$\Psi_{\mathrm{C}} = \frac{\mu}{2}\left(J^{-2/3} I_C - 3\right) + \frac{\lambda}{2}(J-1)^2$	Wang and Yang 2016

Table 3.2: Summary of proposed energies ([SGK18], 12:3)

Each energy formulation can be split up into a 1D length term and a 3D volume term. The 1D length term penalizes the length changes an object

undergoes, whereas the 3D volume term is a volume-preserving penalty term.

Energy	**1D length term**	**3D volume term**
Ψ_{Neo}	$\frac{\mu}{2}(I_C - 3)$	$-\mu \log J + \frac{\lambda}{2}(\log J)^2$
Ψ_{A}	$\frac{\mu}{2}(I_C - 3)$	$-\mu \log J + \frac{\lambda}{2}(J-1)^2$
Ψ_{B}	$\frac{\mu}{2}\left(J^{-2/3} I_C - 3\right)$	$\frac{\lambda}{2}(J-1)^2$
Ψ_{C}	$\frac{\mu}{2}\left(J^{-2/3} I_C - 3\right)$	$\frac{\lambda}{2}(J-1)^2$

Table 3.3: Energies split up into their 1D length and 3D volume term

1D Length Term

Mooney ([Moo40]) originally proposed the 1D length term

$$\Psi_M = \frac{\mu}{2}(I_C - 3)$$

that is used in Ψ_{Neo} and Ψ_A. If we expand the energy with the singular values of the deformation gradient $\mathbf{F}$, we get the following term:

$$\Psi_M = \frac{\mu}{2}\left(\sigma_0^2 + \sigma_1^2 + \sigma_2^2 - 3\right)$$

This energy reaches its minimum at a zero volume state, meaning $I_C = 0$ which results in $\Psi_M = -3$. Mooney added the hard constraint that $J = 1$, so that the energy is minimized at the volume preserving configuration that is closest to the stretch space origin. Note that the energy is singularity free and well defined under inversion.

The second term

$$\Psi_R = \frac{\mu}{2}\left(J^{-2/3} I_C - 3\right)$$

is used in Ψ_B and Ψ_C and was introduced by Rivlin ([Riv48]). Using the singular values of $\mathbf{F}$, we get the following term:

$$\Psi_R = \frac{\mu}{2}\left(\frac{\sigma_0^2 + \sigma_1^2 + \sigma_2^2}{(\sigma_0\sigma_1\sigma_2)^{\frac{2}{3}}} - 3\right)$$

Unfortunately this term is not singularity free. If either σ_0, σ_1 or σ_2 is equal to zero the result is not defined anymore.

3D Volume Term

The volume term of Ψ_{Neo}, meaning

$$\Psi_{Neo,volume} = -\mu \log J + \frac{\lambda}{2}(\log J)^2$$

results in some numerical problems since the logarithmic function is not defined for $J < 0$ and grows unbounded for $J \to 0$. In conclusion $\Psi_{Neo,volume}$ is not singularity free. The same applies for the 3D volume term of Ψ_A, namely

$$\Psi_{A,volume} = -\mu \log J + \frac{\lambda}{2}(J-1)^2.$$

The term of Ψ_A and Ψ_B which is of the form

$$\Psi_M = \frac{\lambda}{2}(J-1)^2$$

does not have these problems. It is bounded, well defined and invertible. After these observations we combine the robust length with the robust volume term and receive Ψ_D that is defined as

$$\Psi_D = \frac{\mu}{2}(I_C - 3) + \frac{\lambda}{2}(J-1)^2$$

and is singularity free and well defined under inversion. Unfortunately, Ψ_D does not satisfy the requirement of being rest stable which will be discussed in the next section.

TODO: Bonet and Wood 2008 or 1997?

3.2.3 Rest Stabilization

Although Ψ_D meets almost all stated requirements, it is not rest stable. This can be shown with the Piola-Kirchhoff (PK1) stress tensor for Ψ_D

derived by the deformation gradient $\mathbf{F}$:

$$\begin{aligned} P_D(\mathbf{I}) = \frac{\partial \Psi_D}{\partial \mathbf{F}}(\mathbf{I}) &= \frac{\partial \Psi_D}{\partial \mathbf{F}} \left[\frac{\mu}{2} (I_C - 3) + \frac{\lambda}{2}(J-1)^2 \right] \\ &= \frac{\partial \Psi_D}{\partial \mathbf{F}} \frac{\mu}{2} (\mathrm{tr}(\mathbf{I}^\intercal \mathbf{I}) - 3) + \frac{\partial \Psi_D}{\partial \mathbf{F}} \frac{\lambda}{2} (\mathrm{tr}(\mathbf{I}) - 1)^2 \\ &= \mu I + \lambda(\det(\mathbf{I}) - 1) \frac{\partial}{\partial \mathbf{F}} \det(\mathbf{I}) = \mu \mathbf{I} \neq 0 \end{aligned}$$

If the energy had rest stability, $P_D(\mathbf{I})$ would resolve to zero. Unfortunately, this is not the case here. In order to solve that problem, the authors modified $(J-1)^2$ to $(J-\alpha)^2$. Using this modification the energy can written as

$$\Psi_E = \frac{\mu}{2} (I_C - 3) + \frac{\lambda}{2}(J-\alpha)^2.$$

Inserting Ψ_E into the PK1 equation from before, we get

$$\begin{aligned} P_E(\mathbf{F}) &= \frac{\partial \Psi_E}{\partial \mathbf{F}} \left[\frac{\mu}{2} (I_C - 3) + \frac{\lambda}{2}(J-\alpha)^2 \right] \\ &= \mu \mathbf{F} + \lambda(\det(\mathbf{F}) - \alpha) \frac{\partial}{\partial \mathbf{F}} \det(\mathbf{F}). \end{aligned}$$

Solving for an alpha that satisfies $P_E(\mathbf{I}) = 0$ gives us $\alpha = 1 + \frac{\mu}{\lambda}$. Now Ψ_E has to be changed accordingly:

$$\begin{aligned} \Psi_E &= \frac{\mu}{2} (I_C - 3) + \frac{\lambda}{2}(J - 1 - \frac{\mu}{\lambda})^2 \\ &= \frac{\mu}{2} (I_C - 3) - \mu (J-1) + \frac{\lambda}{2}(J-1)^2 + \left(\frac{\mu}{\lambda}\right)^2. \end{aligned}$$

Since constants disappear under differentiation this expression is functionally equivalent to

$$\Psi_E = \frac{\mu}{2} (I_C - 3) - \mu (J-1) + \frac{\lambda}{2}(J-1)^2.$$

Now we can see that Ψ_E looks very similar to Ψ_{Neo}. The difference is that $log(J)$ is replaced with $(J-1)$ in Ψ_E. Keep in mind that $(J-1)$ is

the first term in the taylor approximation of $\log(J)$ at $J = 1$:

$$\begin{aligned}\sum_{n=0}^{\infty} &= \frac{f^{(n)}(1)}{n!}(J-1)^n \\ &= \boldsymbol{(J-1)} - \frac{1}{2}(J-1)^2 + \frac{1}{3}(J-1)^3 - \frac{1}{4}(J-1)^4 + ...\end{aligned}$$

Thus Ψ_E is an approximation of Ψ_{Neo}.

TODO: Add "Zwischenschritte"? Include all calculations? Do we need a bit of a better conclusion of what we did (taylor, neo, sing. free, rest stable)? Check consistency of calculations (when trace when I_C etc.).

3.2.4 Meta-Stability under Degeneracy

The final energy is

$$\Psi_{new} = \frac{\mu}{2}(I_C - 3) + \frac{\lambda}{2}(J-\alpha)^2 - \frac{\mu}{2}\log(I_C + 1). \tag{3.1}$$

With that adjustment the rest stability term can be written as $\alpha = 1 + \frac{\mu}{\lambda} - \left(\frac{\mu}{4}\right)\lambda$.

TODO: How much needs to be explained here?

3.2.5 Lamé Reparametrization

TODO: How much needs to be explained here?

3.3 Energy Analysis

The goal of this chapter is to show that a complete eigenanalysis can be performed on Eq. 3.1.

3.3.1 First Piola-Kirchhoff Stress (PK1)

In order to analyse the energy PK1 can be used for Eq. 3.1:

$$\begin{aligned} P(\mathbf{F}) &= \frac{\partial \Psi_D}{\partial \mathbf{F}} \left[\frac{\mu}{2} (I_C - 3) + \frac{\lambda}{2} (J - \alpha)^2 - \frac{\mu}{2} (I_C + 1) \right] \\ &= \mu \mathbf{F} + \lambda (\det(\mathbf{F}) - \alpha) \frac{\partial}{\partial \mathbf{F}} \det(\mathbf{F}) - \frac{\partial}{\partial \mathbf{F}} \left[\log \left(\operatorname{tr}(\mathbf{F}^\intercal \mathbf{F}) + 1 \right) \right] \\ &= \mu \mathbf{F} + \lambda (\det(\mathbf{F}) - \alpha) \frac{\partial}{\partial \mathbf{F}} \det(\mathbf{F}) - \mu \mathbf{F} \frac{1}{\operatorname{tr}(\mathbf{F}^\intercal \mathbf{F}) + 1} \\ &= \mu \left(1 - \frac{1}{\operatorname{tr}(\mathbf{F}^\intercal \mathbf{F}) + 1} \right) \mathbf{F} + \lambda (J - \alpha) \frac{\partial J}{\partial \mathbf{F}} \end{aligned}$$

with $\alpha = 1 + \frac{\mu}{\lambda} - \left(\frac{\mu}{4} \right) \lambda$. For further use it can be more practical to write $\frac{\partial J}{\partial \mathbf{F}}$ as a result of cross products:

$$\frac{\partial J}{\partial \mathbf{F}} = \left[f_1 \times f_2 \,\middle|\, f_2 \times f_0 \,\middle|\, f_0 \times f_1 \right]. \tag{3.2}$$

TODO: Is it easy to follow? Maybe add some explanations of some steps? Maybe move cross product into background instead of here.

3.3.2 The Energy Hessian Terms

Using the scalar notation for $\mathbf{F}$, the Hessian of the energy can be written as a fourth-order matrix-of-matrices:

$$\frac{\partial^2 \psi}{\partial \mathbf{F}^2} = \frac{\partial P(\mathbf{F})}{\partial \mathbf{F}} = \begin{bmatrix} \left[\frac{\partial P(\mathbf{F})}{\partial f_0} \right] & \left[\frac{\partial P(\mathbf{F})}{\partial f_3} \right] & \left[\frac{\partial P(\mathbf{F})}{\partial f_6} \right] \\ \left[\frac{\partial P(\mathbf{F})}{\partial f_1} \right] & \left[\frac{\partial P(\mathbf{F})}{\partial f_4} \right] & \left[\frac{\partial P(\mathbf{F})}{\partial f_7} \right] \\ \left[\frac{\partial P(\mathbf{F})}{\partial f_2} \right] & \left[\frac{\partial P(\mathbf{F})}{\partial f_5} \right] & \left[\frac{\partial P(\mathbf{F})}{\partial f_8} \right] \end{bmatrix}$$

in which each entry is defined as

$$\frac{\partial P(\mathbf{F})}{\partial f_i} = \frac{\partial}{\partial f_i} \left[\mu \left(1 - \frac{1}{\operatorname{tr}(\mathbf{F}^\intercal \mathbf{F}) + 1} \right) \mathbf{F} + \lambda (J - \alpha) \frac{\partial J}{\partial \mathbf{F}} \right]$$

$$\overset{\text{prod.rule}}{=} \underbrace{\frac{\partial \mathbf{F}}{\partial f_i}\mu\left(1-\frac{1}{\text{tr}(\mathbf{F}^\intercal\mathbf{F})+1}\right)}_{\text{T}_i} + \underbrace{\mathbf{F}\mu\frac{2}{(\text{tr}(\mathbf{F}^\intercal\mathbf{F})+1)^2}f_i}_{\text{M}_i}$$

$$r + \underbrace{\lambda\frac{\partial J}{\partial f_i}\frac{\partial J}{\partial \mathbf{F}}}_{\text{G}_i} + \underbrace{\lambda(J-\alpha)\frac{\partial^2 J}{\partial \mathbf{F}\partial f_i}}_{\text{H}_i} \in \mathbb{R}^3.$$

We can split up this final equation into these four term: T_i (Tikohonov), M_i (Mu), G_i (Volume Gradient), H_i (Volume Hessian). I will examine them separately in the next section.

3.3.3 The Tikhonov, Mu, and Gradient Terms

Tikhonov

The Tikhonov term is a fourth-order matrix-of-matrices

$$\mathbb{T} = \frac{\partial \mathbf{F}}{\partial f_i} = \begin{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} & \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{bmatrix}.$$

If we vectorize $\mathbb{T}$, we get the identity matrix $\mathbf{I} \in \mathbb{R}^{9x9}$ which is of full rank, positive definite and independent of the values in $\mathbf{F}$.

$$\text{vec}(\mathbb{T}) = \check{\mathbf{T}} = \mathbf{I} = \in \mathbb{R}^{9x9}.$$

It serves as a regularizer for the rest of the energy.

Mu

The Mu term has the same structure with different entries

$$\mathbb{M} = \mathbf{F} f_i = \begin{bmatrix} \begin{bmatrix} f_0^2 & f_0 f_3 & f_0 f_6 \\ f_0 f_1 & f_0 f_4 & f_0 f_7 \\ f_0 f_2 & f_0 f_5 & f_0 f_8 \end{bmatrix} & \begin{bmatrix} f_3 f_0 & f_3^2 & f_3 f_6 \\ f_3 f_1 & f_3 f_4 & f_3 f_7 \\ f_3 f_2 & f_3 f_5 & f_3 f_8 \end{bmatrix} & \begin{bmatrix} f_6 f_0 & f_6 f_3 & f_6 f_6 \\ f_6 f_1 & f_6 f_4 & f_6 f_7 \\ f_6 f_2 & f_6 f_5 & f_6 f_8 \end{bmatrix} \\ \begin{bmatrix} f_1 f_0 & f_1 f_3 & f_1 f_6 \\ f_1^2 & f_1 f_4 & f_1 f_7 \\ f_1 f_2 & f_1 f_5 & f_1 f_8 \end{bmatrix} & \begin{bmatrix} f_4 f_0 & f_4 f_3 & f_4 f_6 \\ f_4 f_1 & f_4^2 & f_4 f_7 \\ f_4 f_2 & f_4 f_5 & f_4 f_8 \end{bmatrix} & \begin{bmatrix} f_7 f_0 & f_7 f_3 & f_7 f_6 \\ f_7 f_1 & f_7 f_4 & f_7^2 \\ f_7 f_2 & f_7 f_5 & f_7 f_8 \end{bmatrix} \\ \begin{bmatrix} f_2 f_0 & f_2 f_3 & f_2 f_6 \\ f_2 f_1 & f_2 f_4 & f_2 f_7 \\ f_2^2 & f_2 f_5 & f_2 f_8 \end{bmatrix} & \begin{bmatrix} f_5 f_0 & f_5 f_3 & f_5 f_6 \\ f_5 f_1 & f_5 f_4 & f_5 f_7 \\ f_5 f_2 & f_5^2 & f_5 f_8 \end{bmatrix} & \begin{bmatrix} f_8 f_0 & f_8 f_3 & f_8 f_6 \\ f_8 f_1 & f_8 f_4 & f_8 f_7 \\ f_8 f_2 & f_8 f_5 & f_8^2 \end{bmatrix} \end{bmatrix}.$$

When vectorizing $\mathbb{M}$, the resulting matrix $\check{\mathbf{M}}$ has the squared values of f_i placed on the diagonal

$$\mathrm{vec}(\mathbb{M}) = \check{\mathbf{M}} = \begin{bmatrix} f_0^2 & f_1 f_0 & f_2 f_0 & f_3 f_0 & f_4 f_0 & f_5 f_0 & f_6 f_0 & f_7 f_0 & f_8 f_0 \\ f_0 f_1 & f_1^2 & f_2 f_1 & f_3 f_1 & f_4 f_1 & f_5 f_1 & f_6 f_1 & f_7 f_1 & f_8 f_1 \\ f_0 f_2 & f_1 f_2 & f_2^2 & f_3 f_2 & f_4 f_2 & f_5 f_2 & f_6 f_2 & f_7 f_2 & f_8 f_2 \\ f_0 f_3 & f_1 f_3 & f_2 f_3 & f_3^2 & f_4 f_3 & f_5 f_3 & f_6 f_3 & f_7 f_3 & f_8 f_3 \\ f_0 f_4 & f_1 f_4 & f_2 f_4 & f_3 f_4 & f_4^2 & f_5 f_4 & f_6 f_4 & f_7 f_4 & f_8 f_4 \\ f_0 f_5 & f_1 f_5 & f_2 f_5 & f_3 f_5 & f_4 f_5 & f_5^2 & f_6 f_5 & f_7 f_5 & f_8 f_5 \\ f_0 f_6 & f_1 f_6 & f_2 f_6 & f_3 f_6 & f_4 f_6 & f_5 f_6 & f_6^2 & f_7 f_6 & f_8 f_6 \\ f_0 f_7 & f_1 f_7 & f_2 f_7 & f_3 f_7 & f_4 f_7 & f_5 f_7 & f_6 f_7 & f_7^2 & f_8 f_7 \\ f_0 f_8 & f_1 f_8 & f_2 f_8 & f_3 f_8 & f_4 f_8 & f_5 f_8 & f_6 f_8 & f_7 f_8 & f_8^2 \end{bmatrix}.$$

This structure makes it possible to write $\check{\mathbf{M}}$ as an outer product of $\mathrm{vec}(\mathbf{F})$

$$\check{\mathbf{M}} = \mathrm{vec}(\mathbf{F})\,\mathrm{vec}(\mathbf{F})^\intercal = \check{\mathbf{f}}\check{\mathbf{f}}^\intercal.$$

This matrix has rank one and has a single non-zero eigenvalue. In order to examine the eigenvalues, we can calculate

$$\|\check{\mathbf{f}}\|_2^2 = \sum_{n=0}^{8} |f_n|^2 = \|\mathbf{F}\|_F^2 = \sum_{n=0}^{3} \sigma_i^2 = \left(\sigma_0^2 + \sigma_1^2 + \sigma_2^2\right)$$

in which $\|\cdot\|_F$ stands for the Frobenius norm introduced in Def. 2 and σ_i for the singular values from $\mathbf{\Sigma}$ in the SVD of $\mathbf{F}$ stated in Eq. 2.4. The

equation can be obtained by using Eq. 2.1. The eigenvector is $\check{\mathbf{f}}/\|\check{\mathbf{f}}\|$. The eigenvalue is always non-negative and large if $\mathbf{F}$ contains a large strech.

Gradient

The gradient term also has the same structure as the two terms before with different entries defined by

$$\mathbb{G}(\mathbf{F}) = \frac{\partial J}{\partial \mathbf{F}} \frac{\partial J}{\partial fi}.$$

The vectorized matrix $\check{\mathbf{G}}$ can be written in a similar form as for the Mu term. With the help of the cross product of $\partial J/\partial \mathbf{F}$ stated in Eq. 3.2, $\check{\mathbf{G}}$ can be written as an outer product in the following way:

$$\text{vec}(\mathbb{G}(\mathbf{F})) = \check{\mathbf{G}} = \text{vec}\left(\frac{\partial J}{\partial \mathbf{F}}\right) \text{vec}\left(\frac{\partial J}{\partial \mathbf{F}}\right)^{\intercal} = \check{\mathbf{g}}\check{\mathbf{g}}^{\intercal}.$$

As in the Mu term, we again have a single non-zero, non-negative eigenvalue calculated by

$$\|\check{\mathbf{g}}\|_2^2 = \left\|\frac{\partial J}{\partial \mathbf{F}}\right\|_F^2 = \left[(\sigma_0\sigma_1)^2 + (\sigma_0\sigma_2)^2 + (\sigma_1\sigma_2)^2\right].$$

The eigenvector is $\check{\mathbf{g}}/\|\check{\mathbf{g}}\|$.

TODO: Not all proofs and calculations included yet. Add or leave out some if too confusing.

3.3.4 The Volume Hessian

The volume Hessian term is of the form

$$\mathbb{H} = \frac{\partial^2 J}{\partial \mathbf{F} \partial f_i} = \begin{bmatrix} \frac{\partial}{\partial \mathbf{F}}\left[\frac{\partial J}{\partial f_0}\right] & \frac{\partial}{\partial \mathbf{F}}\left[\frac{\partial J}{\partial f_3}\right] & \frac{\partial}{\partial \mathbf{F}}\left[\frac{\partial J}{\partial f_6}\right] \\ \frac{\partial}{\partial \mathbf{F}}\left[\frac{\partial J}{\partial f_1}\right] & \frac{\partial}{\partial \mathbf{F}}\left[\frac{\partial J}{\partial f_4}\right] & \frac{\partial}{\partial \mathbf{F}}\left[\frac{\partial J}{\partial f_7}\right] \\ \frac{\partial}{\partial \mathbf{F}}\left[\frac{\partial J}{\partial f_2}\right] & \frac{\partial}{\partial \mathbf{F}}\left[\frac{\partial J}{\partial f_5}\right] & \frac{\partial}{\partial \mathbf{F}}\left[\frac{\partial J}{\partial f_8}\right] \end{bmatrix}.$$

Vectorizing $\mathbb{H}$ reveals the structure

$$\text{vec}(\mathbb{H}) = \check{\mathbf{H}} = \begin{bmatrix} 0 & 0 & 0 & 0 & f_8 & 0 & 0 & -f_5 & f_4 \\ 0 & 0 & 0 & -f_8 & 0 & f_6 & f_5 & 0 & -f_3 \\ 0 & 0 & 0 & f_7 & -f_6 & 0 & -f_4 & f_3 & 0 \\ 0 & -f_8 & f_7 & 0 & 0 & 0 & 0 & f_2 & -f_1 \\ f_8 & 0 & -f_6 & 0 & 0 & 0 & -f_2 & 0 & f_0 \\ -f_7 & f_6 & 0 & 0 & 0 & 0 & f_1 & -f_0 & 0 \\ 0 & f_5 & -f_4 & 0 & -f_2 & f_1 & 0 & 0 & 0 \\ -f_5 & 0 & f_3 & f_2 & 0 & -f_0 & 0 & 0 & 0 \\ f_4 & -f_3 & 0 & -f_1 & f_0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

We can write $\check{\mathbf{H}}$ as a cross-product matrix in the form

$$\check{\mathbf{H}} = \begin{bmatrix} 0 & -\widehat{\mathbf{f_2}} & \widehat{\mathbf{f_1}} \\ \widehat{\mathbf{f_2}} & 0 & -\widehat{\mathbf{f_0}} \\ -\widehat{\mathbf{f_1}} & \widehat{\mathbf{f_0}} & 0 \end{bmatrix}$$

where $\widehat{\mathbf{f_1}}$ stands for a cross-product matrix:

$$\widehat{\mathbf{x}} = \begin{bmatrix} 0 & -x_2 & x_1 \\ x_2 & 0 & -x_0 \\ -x_1 & x_0 & 0 \end{bmatrix}.$$

We can easily observe that $\check{\mathbf{H}}$ is a *self-similar* cross-product matrix. This means that the macro structure of the matrix is the same as the micro structure.

TODO: Check each f_1 and $-$ in vectorization of H because it's exhausting.

Volume Hessian Eigenvalues

For determining the eigenvalues of $\check{\mathbf{H}}$, we need to examine the two characteristic polynomials

$$\epsilon^3 - \operatorname{tr}(\mathbf{C})\epsilon - 2J = 0 \tag{3.3}$$

$$\epsilon^3 - \operatorname{tr}(\mathbf{C})\epsilon^2 + \frac{1}{2}\left(\operatorname{tr}^2(\mathbf{C}) - \operatorname{tr}(\mathbf{C}^2)\right)\epsilon - \det(\mathbf{C}) = 0 \tag{3.4}$$

where ϵ denotes the eigenvalues of $\check{\mathbf{H}}$ and $\mathbf{C}$ is taken from Table 3.1. Eq. 3.3 is easier to solve and corresponds to the characteristic polynomial of $\mathbf{C}$. Given its roots ϵ_α, ϵ_β, ϵ_γ, the eigenvalues of $\check{\mathbf{H}}$ are: $\pm\sqrt{\epsilon_\alpha}$, $\pm\sqrt{\epsilon_\beta}$, $\pm\sqrt{\epsilon_\gamma}$. Using the singular values of $\mathbf{F}$, the eigenvalues can be written nicely in the following way:

$$\begin{aligned} \epsilon_3 &= \sigma_0 & \epsilon_6 &= -\sigma_0 \\ \epsilon_4 &= \sigma_1 & \epsilon_7 &= -\sigma_1 \\ \epsilon_5 &= \sigma_2 & \epsilon_8 &= -\sigma_2. \end{aligned}$$

The remaining eigenvalues can be obtained by using Eq. 3.4. This equation represents a depressed cube. A depressed cube is a cubic of the form

$$t^3 + pt + q.$$

Eq. 3.4 can be written in this form with $p = -\operatorname{tr}(\mathbf{C})$ and $q = -2$. Using this knowledge, the roots of Eq. 3.4 can be obtained by

$$\epsilon_k = 2\sqrt{\frac{I_C}{3}}\cos\left[\frac{1}{3}\left(\arccos\left(\frac{3J}{I_C}\sqrt{\frac{3}{I_C}}\right) + 2\pi k\right)\right] \quad k = 0, 1, 2.$$

For further reading about how to solve depressed cubic equations, see e.g. *How to Solve a Cubic Equation, Part 4: The 111 Case* [Bli07].

These are all of the eigenvalues of $\check{\mathbf{H}}$. We know that three of the six eigenvalues (ϵ_3, ... ϵ_8) have to negative or equal to zero since we take the positive and negative roots. In addition, the cosine function for calculating ϵ_0, ϵ_1 and ϵ_2 ensures that one or two of these eigenvalues are also negative. As we have seen in the previous sections, the other terms

of $\frac{\partial^2 \psi}{\partial \mathbf{F}^2}$ all only have nonnegative eigenvalues. So the volume Hessian is the only possible source of negative eigenvalues ([SGK18], 12:6).

In order to investigate the behaviour of the volume Hessian term further, let us look at $J = \det(\mathbf{F})$ a bit more in detail. J is not convex which is problematic since we want to use the Newton method for the optimization process. Fortunately, the other terms of $\frac{\partial^2 \psi}{\partial \mathbf{F}^2}$ serve as an additional regularization.

TODO: include reflection convention? check with background. Regularisation?

Volume Hessian Eigenvectors

$\check{\mathbf{H}}$ can be factorized as $\check{\mathbf{H}} = \check{\mathbf{Q}}\mathbf{\Lambda}\check{\mathbf{H}}^\intercal$ according to its eigendecomposition. The eigenvectors can then be obtained by $\check{\mathbf{Q}}$. In the tensor form signified by $\mathbb{Q}$ each eigenvector is a 3 $\times$3 matrix:

$$\mathbb{Q} = \begin{bmatrix} [\mathbf{Q}_0] & [\mathbf{Q}_3] & [\mathbf{Q}_6] \\ [\mathbf{Q}_1] & [\mathbf{Q}_4] & [\mathbf{Q}_7] \\ [\mathbf{Q}_2] & [\mathbf{Q}_5] & [\mathbf{Q}_8] \end{bmatrix}.$$

The eigenvector corresponding to ϵ_3 can be written as

$$\mathbf{Q}_3 = \frac{1}{\sqrt{2}}\mathbf{U}\mathbf{D}_3\mathbf{V}_\intercal$$

where $\mathbf{U}$ and $\mathbf{V}$ are taken from the SVD of $\mathbf{F}$, $\frac{1}{\sqrt{\acute{\mathrm{L}}}}$

3.3.5 The Complete Eigensystem

TODO: Include what is necessary.

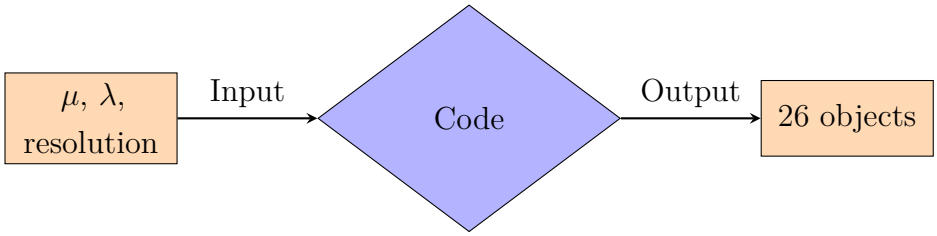
TODO: Eigenvalues of Mu and Gradient term better explain?

3.4 Experiments with the Code

The authors of the paper *SNH-FS* provided the implementation for an application of their formulated energy. In said code, they implemented the stretch test on a cube. Their implementation demands a directory into which the output files should be saved, the two lamé parameters μ and λ and a value for defining the desired resolution as input data. Here is an example for a command with resolution = 10, $\mu = 1.0$, $\lambda = 10.0$ and output directory *output*:

```
1 $ ./tetcli 10 stable_neo_hookean 1.0 10.0 output
```

The algorithm progressively calculates the deformation in 25 steps. The outputs are 26 static objects that show the object in its rest state and the 25 steps of deformation.



TODO: Explain how the code is implemented in simple words and how the energy is taken in account with the poisson's ratio. Do I have to reference code? Explain tetcli and Hexcli.

For starters let us take common values for μ and λ. We first start with $\mu = 1.0$, $\lambda = 10.0$ and a resolution of 10.0. For the poisson's ratio we get the value 0.4545:

$$\sigma = \frac{10.0}{2(10.0 + 1.0)} = 0.4545$$

The images in Fig. 3.3 show the stretch test with $\mu = 1.0$, $\lambda = 10.0$ and a resolution of 10.0 on a tetrahedral and a hexahedral mesh.

TODO: Load into OpenFlipper and screenshot results. Include remaining examples (diff. lamé param. and changes in method) and what went right and what went wrong.

(a) Stretch test on a hexahedral mesh

(b) Stretch test on a tetrahedral mesh

Figure 3.3: Stretch test performed on a cube with (a) a hexahedral mesh and (b) a tetrahedral mesh

3.5 Discussion

Stuff, Taylor approx.

List of Figures

List of Tables

Quelltextverzeichnis

Bibliography

[Bli07] J. F. Blinn. "How to Solve a Cubic Equation, Part 4: The 111 Case". In: *IEEE Computer Graphics and Applications* 27.1 (Jan. 2007), pp. 100–103. ISSN: 1558-1756. DOI: 10.1109/MCG.2007.10.

[For14] William Ford. *Numerical linear algebra with applications: Using MATLAB*. Academic Press, 2014.

[GV12] H Golub Gene and F Van Loan Charles. *Matrix computations. Vol. 3*. 2012.

[Kor17] Alexander M. Korsunsky. "Chapter 2 - Elastic and Inelastic Deformation and Residual Stress". In: *A Teaching Essay on Residual Stresses and Eigenstrains*. Ed. by Alexander M. Korsunsky. Butterworth-Heinemann, 2017, pp. 5–20. ISBN: 978-0-12-810990-8. DOI: https://doi.org/10.1016/B978-0-12-810990-8.00002-1. URL: http://www.sciencedirect.com/science/article/pii/B9780128109908000021.

[Lak87] Roderic Lakes. "Foam structures with a negative Poisson's ratio". In: *Science* 235 (1987), pp. 1038–1041.

[Lak93] Roderic Lakes. "Advances in negative Poisson's ratio materials". In: *Advanced Materials* 5.4 (1993), pp. 293–296.

[LM15] Joerg Liesen and Volker Mehrmann. *Linear algebra*. 1st ed. 2015. Springer International Publishing, Switzerland 2015: Springer, Cham, 2015. ISBN: 978-3-319-24344-3.

[Moo40] Melvin Mooney. "A theory of large elastic deformation". In: *Journal of applied physics* 11.9 (1940), pp. 582–592.

[MR09] P. H. Mott and C. M. Roland. "Limits to Poisson's ratio in isotropic materials". In: *Phys. Rev. B* 80 (13 Oct. 2009), p. 132104. DOI: 10.1103/PhysRevB.80.132104. URL: https://link.aps.org/doi/10.1103/PhysRevB.80.132104.

[Riv48] RS Rivlin. "Large elastic deformations of isotropic materials IV. Further developments of the general theory". In: *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences* 241.835 (1948), pp. 379–397.

[SGK18] 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.

[Spe80] A. J. M. Spencer. *Continuum Mechanics*. 2004th ed. 31 East 2nd Street, Mineola, N.Y. 11501: Dover Publications, Inc., 1980. ISBN: 0-486-43594-6 (pbk.)

Online Sources

[Pix] Pixar. *Deformation Map.* URL: https://dl.acm.org/ft_gateway.cfm?id=3180491&ftid=2009597.

Figure Sources

[Pix] Pixar. *Deformation Map.* URL: https://dl.acm.org/ft_gateway.cfm?id=3180491&ftid=2009597.

Erklärung

gemäss Art. 28 Abs. 2 RSL 05

Name/Vorname: .....

Matrikelnummer: .....

Studiengang: ……

Bachelor ☐ Master ☐ Dissertation ☐

Titel der Arbeit: .....

.....

.....

LeiterIn der Arbeit: .....

.....

Ich erkläre hiermit, dass ich diese Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen benutzt habe. Alle Stellen, die wörtlich oder sinngemäss aus Quellen entnommen wurden, habe ich als solche gekennzeichnet. Mir ist bekannt, dass andernfalls der Senat gemäss Artikel 36 Absatz 1 Buchstabe o des Gesetztes vom 5. September 1996 über die Universität zum Entzug des auf Grund dieser Arbeit verliehenen Titels berechtigt ist.

.....
Ort/Datum

.....
Unterschrift