Discrete Gauß-Codazzi Equations for Efficient Triangle Mesh Processing

Josua Raphael Sassen

Born 27th May 1994 in Bonn, Germany

21th March 2019

Master's Thesis Mathematics

Advisor: Prof. Dr. Martin Rumpf

Second Advisor: Prof. Dr. Jochen Garcke

Institut für Numerische Simulation

Mathematisch-Naturwissenschaftliche Fakultät der

Rheinischen Friedrich-Wilhelms-Universität Bonn

Contents

Nomenclature

The derivatives of a functional $\mathcal{F}\colon \mathbb{R}^n \to \mathbb{R}^m$, $(\xi_1, \dots, \xi_n) \mapsto \mathcal{F}[\xi_1, \dots, \xi_n]$ with respect to the i-th coordinate, respectively ξ_i, are denoted by either $\partial_i \mathcal{F}$ or $\partial_{\xi_i} \mathcal{F}$ depending on the context. The Jacobi matrix at a point ξ is denoted by $D_\xi \mathcal{F}$.

Differential Geometry

$\mathcal{M}$	a generic manifold
$\psi\colon U \subset \mathbb{R}^d \to \mathcal{M}$	local parametrization of a d-dimensional manifold
$T_p\mathcal{M}$	tangent space of $\mathcal{M}$ at point $p \in \mathcal{M}$
g, g_p	Riemannian metric (at $p \in \mathcal{M}$), as well as its matrix representation
$\mathcal{L}$, $\mathcal{E}$	length and path energy functional on $\mathcal{M}$
$\exp_p$, $\log_p$	exponential map and logarithm at p on $\mathcal{M}$
$\mathcal{W}$	approximation of the squared Riemannian distance
L^K, E^K	time-discrete length and path energy functional of order K; minimizers of the energy are called discrete geodesics of order K

Surfaces

$\mathcal{S}$	regular surface in $\mathbb{R}^3$
n	normal field on the surface
S_p, s_ξ	shape operator at $p \in \mathcal{M}$ and its matrix representation
h, h_p	second fundamental form (at $p \in \mathcal{M}$), as well as its matrix representation
$F\colon \Omega \to O(3)$	frame field on surface $\mathcal{S}$, which is parametrized over Ω
P^1, P^2	transition matrices of the frame field

Discrete Surfaces

$\mathcal{S}_h$	connectivity of a triangle mesh
$\mathcal{V}$, $\mathcal{E}$, $\mathcal{F}$	sets of vertices, edges, and faces of a triangle mesh
$\mathcal{V}^*$, $\mathcal{E}^*$	sets of dual vertices and dual edges of a triangle mesh
X	embedding or immersion of a discrete surface, i.e. its nodal positions
$\mathbf{S}$	embedded or immersed discrete surface
X_v, E_e, $T(f)$	embedded vertex, edge, or face respectively
$G\|_f$, $H\|_f$, $S\|_f$	elementwise constant discrete first and second fundamental forms, and shape operator
N_f, N_e	face normal and dihedral angle bisecting edge normal
l_e, θ_e	edge length and dihedral angle of an edge e

Phyiscal Modeling

ϕ, Φ	continuous respectively discrete deformation
$\mathcal{G}[\phi]$, $\mathcal{G}[\Phi]$	continuous respectively elementwise constant discrete distortion tensor
$\mathcal{W}$	generic deformation energy
$\mathcal{W}_{\text{mem}}$, $\mathcal{W}_{\text{bend}}$	generic membrane and bending energy
$\mathcal{W}_{\text{bend}}^{\text{DS}}$	Discrete Shells bending energy for discrete surfaces

Lengths and Angles

z	combined vector of edge lengths and dihedral angles
$\mathcal{Z}$	admissible lengths and angles, also the space of lengths and angles
$Z\colon \mathbb{R}^{3\lvert\mathcal{V}\rvert} \to \mathbb{R}^{2\lvert\mathcal{E}\rvert}$	map yielding the lengths and angles $Z[X]$ of an immersion X
$F\colon \mathcal{F} \to SO(3)$	discrete frame field on $\mathcal{S}_h$
$R_{ij}\colon \mathbb{R}^{2\lvert\mathcal{E}\rvert} \to SO(3)$	induced transition rotation between faces f_i and f_j
$\mathcal{T}_f\colon \mathbb{R}^{2\lvert\mathcal{E}\rvert} \to \mathbb{R}$	triangle inequality map for face f
$\mathcal{I}_v\colon \mathbb{R}^{2\lvert\mathcal{E}\rvert} \to SO(3)$	discrete integrability map for vertex v
$\widetilde{\mathcal{I}}_v\colon \mathbb{R}^{2\lvert\mathcal{E}\rvert} \to \mathbb{R}^3$	$x-y-z$ Euler angles of discrete integrability map for vertex v
$\mathcal{R}\colon \mathbb{R}^{2\lvert\mathcal{E}\rvert} \to \mathbb{R}^{3\lvert\mathcal{V}\rvert}$	generic reconstruction of nodal positions from lengths and angles
$\mathcal{P}_{\mathcal{Z}}\colon \mathbb{R}^{2\lvert\mathcal{E}\rvert} \to \mathbb{R}^{2\lvert\mathcal{E}\rvert}$	generic projection on the space of lengths and angles $\mathcal{Z}$
$\mathcal{W}_q$, $\mathcal{W}_{\text{TS}}$	quadratic and thin shells deformation energy on lengths and angles
$\mathcal{R}_{\mathcal{W}}$, $\mathcal{P}_{\mathcal{Z}}^{\mathcal{W}}$	energy-based reconstruction and projection

Introduction

In modern computer animation, we witness highly complex deformations of character models. Artists are given the opportunity to exploit the availability of tremendous computational power to generate movements nearly indistinguishable from natural ones. In addition, the technical development of motion capturing has created another way to bring natural motions into computer animations. This surpasses professional games and movies, clearly observable since the introduction of Microsoft Kinect. The character models are commonly represented as high-resolution triangle meshes with hundreds of thousands of triangles.

Both, discrete differential geometry and physical modeling play an important role in creating realistic movements. However, to capture deformation paths of shapes in a comprehensive mathematical model, we need to go one step further and introduce *shape spaces*, an idea due to Kendall [Ken84]. Complex geometric objects, such as surfaces or images, are considered points in these potentially infinite dimensional spaces, where we exploit the physical modeling of deformations to define a nonlinear structure on them. These spaces enable rigorously study of the geometry and physics of deformations and mathematical notions such as geodesics and parallel transport have a practical interpretation in the synthesis of animations.

This is part of the larger research area of geometry processing, which involves geometric objects from creation over editing to rendering. In recent years, the fabrication of the resulting shapes by means of modern additive methods has become an increasing focus. In this thesis, we will focus on the processing of triangle meshes, also called *discrete surfaces* under certain conditions. We are interested in the deformation of triangle meshes in dense correspondence, meaning that we assume them to have the same number of vertices and identical connectivity. In this context, the deformation is completely described by the change of nodal positions.

However, when working with nodal positions, problems with rigid body motion invariance arise, thus we want to find degrees of freedom for deformations which are rigid body motions invariant. In the continuous setting, the fundamental theorem of surfaces establishes that immersed surfaces can be modeled rigid body motions invariant by their first and second fundamental forms. They already determine the immersion provided they fulfill the Gauß-Codazzi equations. For discrete surfaces, edge lengths and angles between the normals of neighboring triangles can be considered as counterparts to the fundamental forms which motivates their consideration as primal degrees of freedom for deformations. Beyond their inherent invariance to rigid transformations, they offer other benefits, such as occurring naturally in discrete deformation energies and representing natural deformations in a localized sparse fashion.

The goal of this thesis is to develop a comprehensive understanding of lengths and angles as primal degrees of freedom for deformations of discrete surfaces. This includes their introduction as discrete counterparts to the fundamental forms, adapting deformation energies and investigating different, partially novel, approaches to con-

struct nodal positions from lengths and angles. Furthermore, an in-depth study of the discrete fundamental theorem from [WLT12] reveals conditions on lengths and angles to belong to an immersed surface, akin to the Gauß-Codazzi equations. We will use these conditions to introduce a nonlinear structure on the space of lengths and angles. The resulting toolbox will be used to perform Principal Geodesic Analysis on this new space and build a low-dimensional model of deformations from examples. This model will be used in marker fitting, a problem related to motion capturing.

Structure We will start with two chapters of preliminary notions used throughout this thesis. In the first chapter, we will focus on surfaces, their discrete counterpart and modeling them as midsurface of thin elastic objects. Afterwards, the notions from Riemannian geometry necessary for shape spaces and a time-discrete geodesic calculus will be introduced together with a brief explanation of the application to the space of discrete shells. These two chapters are kept very brief and mainly serve to introduce a consistent notation for the remaining thesis.

Hence, the third chapter will be the first to deal with edge lengths and dihedral angles. In that chapter, we will consider them as discrete counterparts to the fundamental forms, which will include the study of the discrete fundamental theorem from [WLT12]. Also, we will introduce and compare several ways to construct nodal positions for given lengths and angles. Finally, in the fourth chapter, this will be applied to introduce the space of lengths and angles. A simplified version of Principal Geodesic Analysis on this space and the model fitting application will be provided.

Implementation Accompanying this thesis is an implementation of the different elements of the lengths and angles toolbox that will be studied below. The implementation was used to conduct the different numerical experiments throughout this thesis and is based on an C++ framework previously developed in the group of Prof. Dr. Martin Rumpf. Main parts of the existing components are tools for the space of discrete shells developed in the context of [Hee16]. In the course of this thesis, there will be remarks on how to use the introduced notions in implementations, which will be clearly recognizable due to their box with a grey background.

Acknowledgments First, I want to thank my advisor Martin Rumpf for introducing me to this topic and sharing his mathematical insights, for placing an immense amount of trust in my capabilities, and for providing me with all kinds of resources. Also, I am very grateful to Behrend Heeren for all the time he spent on answering my questions, for all our insightful discussions, and for all the helpful remarks about this thesis. Furthermore, I would like to thank Jochen Garcke for supporting me throughout my studies and for taking the role of my second advisor. Moreover, I want to express gratitude to Klaus Hildebrandt for his invaluable insights on the project and for having me as a guest at the TU Delft.

Additionally, I have to thank Alex Dyck and Stefan Simon for proofreading this thesis. I have spent the last year enjoying the atmosphere in Martin Rumpf's group and for this, I want to additionally thank Alexander Effland, Martin Lenz, Marko Rajković, Carole Rossignol, and Gabi Sodoge-Stork.

I am indebted to my parents, Luzia und Johannes Sassen, for their constant support, I could not have gotten so far without it. Last but not least, I have to thank Sandrine Künzel for so many things, among them her everlasting support and encouragement during the work on this thesis.

Chapter 1

Discrete Surfaces and Thin Shells

In this first of two chapters introducing basic notions, we start with the central objects of our studies, surfaces and their discrete approximation. As noted in the introduction, there is a rich set of applications for surfaces as modeling tool be it in mathematical physics or in computer graphics. Independent of this, they have been an object of mathematical studies at least since ancient Greece and hence there is rich theory describing their topology and geometry. Through the advent of modern computers, numerical representations of surfaces have become increasingly important in simulation and graphics. One popular choice to represent surfaces is to approximate them by triangle meshes, under certain conditions also called discrete surfaces. The studies of discrete surfaces have accumulated in the active field of discrete differential geometry, aiming to develop a comprehensive theory for them akin to continuous surfaces. Moreover, discrete surfaces have recently become modeling tools used in additive manufacturing beyond their interpretation as approximations of continuous surfaces. This is reflected for example in publications on discrete developability such as [SGC18, RHSH18] or on related so-called zippables [SPSH18]. Nevertheless, already before this, surfaces have been used in mathematical physics to model thin deformable objects, which led to the rigorous deduction of physically plausible deformation energies.

In this chapter, we will consider these three perspectives by first introducing parametric continuous surfaces in three-dimensional space. Then we will go on to introducing discrete surfaces and briefly describe some basic notions of their geometry without going into much detail at this point. The last section will then be concerning the modeling of thin elastic objects as surfaces and give a concise exposition of continuous and discrete deformation energies.

1.1 A Brief Introduction to Surfaces

In this section, we briefly introduce the necessary properties of embedded surfaces (i.e. two-dimensional manifolds) $\mathcal{S} \subset \mathbb{R}^3$. As it is sufficient for this thesis, we will only consider a parametric description of surfaces as defined in the following. This introduction is kept very brief and is based on [Hee17, dC76], to which we also refer for further reading.

We begin with the first very basic definitions of a surface and its tangent space.

Definition 1.1 (Regular surface)**.** The set $\mathcal{S} \subset \mathbb{R}^3$ is a *regular surface* if for each $p \in \mathcal{S}$ there is an $\varepsilon > 0$, an open set $\Omega \subset \mathbb{R}^2$, and a smooth mapping $\psi\colon \Omega \to \mathbb{R}^3$, such that

(i) $\psi(\Omega) = \mathcal{S} \cap B_\varepsilon(p)$ and $\psi\colon \Omega \to \mathcal{S} \cap B_\varepsilon(p)$ is a homeomorphism.

(ii) The Jacobi matrix $D_\xi \psi \in \mathbb{R}^{3\times 2}$ has rank two for each $\xi \in \Omega$.

Definition 1.2 (Tangent space)**.** Let $\mathcal{S} \subset \mathbb{R}^3$ be a regular surface, and let $p \in \mathcal{S}$. Then we define the *tangent space* at p

$$T_p\mathcal{S} := \{\, \dot\gamma(0) \,|\, \gamma\colon (-1,1) \to \mathcal{S} \text{ smooth},\ \gamma(0) = p\}.$$

Elements of this space are called *tangent vectors*.

Note that $T_p\mathcal{S}$ is a two-dimensional vector space. Let us now consider a parametrization $\psi\colon \Omega \to \mathbb{R}^3$ of S, for which we let $\xi \in \Omega$ and let $p = \psi(\xi) \in \mathcal{S}$. If we then consider a smooth $\alpha\colon (-1,1) \to \Omega$ with $\alpha(0) = \xi$ and $\gamma_\alpha := \psi \circ \alpha$ we have $\dot\gamma_\alpha(0) = D_\xi\psi\dot\alpha(0)$ and get

$$T_p\mathcal{S} = \operatorname{Im} D_\xi\psi = \operatorname{span}\{\partial_{\xi_1}\psi(\xi), \partial_{\xi_2}\psi(\xi)\}\,.$$

We will call this basis of the tangent spaces the *canonical basis* and define

$$V_i = \partial_{\xi_i}\psi(\xi), \quad i = 1,2$$

as handy abbreviations.

Differentiation For a smooth function $\varphi\colon \mathcal{S} \to \mathbb{R}$ and $p \in \mathcal{S}$ we define the differential $D_p\varphi$ as a linear form acting on tangent vectors $V \in T_p\mathcal{S}$ as directional derivative, i.e.

$$D_p\varphi(V) := \frac{d}{dt}\varphi(\gamma(t))\Big|_{t=0}$$

for an arbitrary curve $\gamma\colon (-1,1) \to \mathcal{S}$ with $\gamma(0) = p$ and $\dot\gamma(0) = V$. For a vector-valued deformation $\phi\colon \mathcal{S} \to \mathbb{R}^3$ the definition above holds for each component of $\phi = (\phi_1, \phi_2, \phi_3)$. In particular, $D_p\phi$ defines a linear map between the tangent spaces, i.e.

$$D_p\phi\colon T_p\mathcal{S} \to T_{\phi(p)}\phi(\mathcal{S})\,.$$

1.1.1 Fundamental Forms

We want to measure lengths and angles on the surface, for which we need an inner product at each point.

Remark. In the following, we will introduce the structure of a Riemannian manifold on the surface, which we will be discussed in more generality in Chapter 2.

Definition 1.3 (First fundamental form)**.** The *first fundamental form* in $p \in \mathcal{S}$ is given by

$$g_p\colon T_p\mathcal{S} \times T_p\mathcal{S} \to \mathbb{R}, \quad (U,V) \mapsto \langle U, V\rangle_{\mathbb{R}^3}\,.$$

We see that g_p is a scalar product on $T_p\mathcal{S}$ and thus we can consider its matrix representation in the canonical basis, which by abuse of notation we also denote as g_p. It is given by

$$g_p = (\langle V_i, V_j\rangle_{\mathbb{R}^3})_{ij} \in \mathbb{R}^{2\times 2}.$$

This matrix is invertible because it is positive definite by assumption and we denote the coefficients of the inverse by superscript indices, i.e.

$$g_p^{-1} = (g^{ij})_{ij} \in \mathbb{R}^{2\times 2}.$$

As scalar product, g_p allows us to measure then lengths of tangent vectors and the angle between two of them. We can use this to measure the lengths of curves on the surface.

Definition 1.4 (Length of a curve)**.** Let $\alpha\colon (-1,1) \to \Omega$ and $\gamma_\alpha := \psi \circ \alpha$, then the *length of* γ_α is defined as its integrated velocity

$$\begin{aligned}\mathcal{L}[\gamma_\alpha] &:= \int_{-1}^{1} |\dot\gamma_\alpha(t)|\,\mathrm{d}t = \int_{-1}^{1} \sqrt{\langle D\psi\dot\alpha(t), D\psi\dot\alpha(t)\rangle_{\mathbb{R}^3}}\,\mathrm{d}t \\ &= \int_{-1}^{1} \sqrt{\langle D\psi^T D\psi\dot\alpha(t), \dot\alpha(t)\rangle_{\mathbb{R}^3}}\,\mathrm{d}t\,,\end{aligned}$$

where we write $D\psi = D_{\alpha(t)}\psi$ to shorten the expression.

This way we can also define arbitrary integrals on the surface for some function $\varphi : \mathcal{S} \to \mathbb{R}$ by

$$\int_A \varphi\,\mathrm{d}a := \int_{\psi^{-1}(A)} (\varphi\circ\psi)(\xi)\,\sqrt{\det g_\xi}\,\mathrm{d}\xi\,,$$

Second fundamental form The first fundamental form allows us to study objects that live on the surface. Moreover, we are also interested in studying how the surface $\mathcal{S}$ is bent, i.e. its curvature. To this end, we have to introduce some more definitions, especially another bilinear form called the *second fundamental form*.

Definition 1.5 (Normal field)**.** Let $S^2 \subset \mathbb{R}^3$ be the 2-dimensional unit sphere. The *(unit) normal field* of $\mathcal{S}$ is a mapping $n\colon \mathcal{S} \to S^2$ with $n(p) \perp T_p\mathcal{S}$ for all $p \in \mathcal{S}$. We say that $\mathcal{S}$ is orientable if there is a continuous normal field. In particular, as $\operatorname{rank}(D\psi) = 2$, we will write

$$n(p) = (n\circ\psi)(\xi) = \frac{\partial_1\psi \times \partial_2\psi}{|\partial_1\psi \times \partial_2\psi|}(\xi)\,.$$

Definition 1.6 (Shape operator)**.** Let $\mathcal{S} \subset \mathbb{R}^3$ be a regular and orientable surface, $p \in \mathcal{S}$. The *shape operator* $S_p\colon T_p\mathcal{S} \to T_p\mathcal{S}$ at p is the linear mapping defined via $S_p(U) = D_p n(U)$ for $U \in T_p\mathcal{S}$.

Remark. As $T_{n(p)}S^2 = n(p)^\perp = T_p\mathcal{S}$ the shape operator S_p is indeed an endomorphism on $T_p\mathcal{S}$.

Definition 1.7 (Second fundamental form)**.** Let $\mathcal{S} \subset \mathbb{R}^3$ be a regular and orientable surface, and $p \in \mathcal{S}$. The *second fundamental form* h_p is the bilinear form on $T_p\mathcal{S}$ associated with S_p, i.e.

$$h_p(U,V) := g_p(S_pU, V)\,, \quad U,V \in T_p\mathcal{S}\,.$$

We will exploit this bilinear form later to define different notions of curvature of a surface. Again, we can derive a matrix representation in the canonical basis, which by abuse of notation we also denote as h_p. As before, this is achieved by pulling back h_p to $\mathbb{R}^2$ using $D_\xi\psi$. It is given by

$$h_p = D_\xi(n\circ\psi)^T D_\xi\psi \in \mathbb{R}^{2\times 2},$$

where $\psi(\xi) = p$. In the following, the explicit dependence on the point p of the fundamental forms will typically be dropped when it is clear which point we are considering.

One can see, that the second fundamental form is symmetric, i.e. that S_p is self-adjoint with respect to the first fundamental form g_p. This means that we can diagonalize S_p as s linear map with an orthonormal basis, and thus it has two eigenvalues. These eigenvalues now allow us to define different notions of curvature.

Definition 1.8 (Curvatures)**.** The eigenvalues κ_1 and κ_2 of S_p are called the *principal curvatures* of $\mathcal{S}$ at a point $p \in \mathcal{S}$. The *mean curvature* in p is defined as the sum $H_p = \operatorname{tr} S_p = \kappa_1 + \kappa_2$ and the *Gaussian curvature* in p is defined as the product $K_p = \det S_p = \kappa_1\kappa_2$.

To define a physically-plausible model for the deformations we need to measure the bending of surfaces under deformation. Bending corresponds to a change of the second fundamental form or shape operator, which is captured by the following definitions.

Definition 1.9 (Pulled-back shape operator)**.** The pulled-back shape operator $S_p^*[\phi]\colon T_p\mathcal{S} \to T_p\mathcal{S}$ is given by

$$g_p\left(S_p^*[\phi]\,U,\,V\right) = h_{\phi(p)}\left(D_p\phi\,U,\,D_p\phi\,V\right)\,, \quad \forall\, U, V \in T_p\mathcal{S}\,. \tag{1.1}$$

Definition 1.10 (Relative shape operator)**.** The *relative shape operator* $S_p^{rel}[\phi]$ is defined as the pointwise difference, i.e.

$$S_p^{rel}[\phi]\colon T_p\mathcal{S} \to T_p\mathcal{S}\,, \quad S_p^{rel}[\phi] := S_p - S_p^*[\phi]\,. \tag{1.2}$$

Again, we obtain matrix representations in the canonical basis $s_\xi^*[\phi] \in \mathbb{R}^{2\times 2}$ and $s_\xi^{rel}[\phi] \in \mathbb{R}^{2\times 2}$ of $S_p^*[\phi]$ and $S_p^{rel}[\phi]$, respectively, which are given by

$$s_\xi^*[\phi] = g_\xi^{-1}\tilde{h}_\xi\,, \quad s_\xi^{rel}[\phi] = s_\xi - s_\xi^*[\phi] = g_\xi^{-1}(h_\xi - \tilde{h}_\xi)\,. \tag{1.3}$$

1.1.2 Fundamental Theorem

A natural question one could ask is if there is a criterion when two surfaces will be congruent, i.e. only differ by a rigid transformation. This question will also arise in the physical modeling of surface deformations later on, where will consider energies invariant to rigid transformations of the surface. Thus a more involved question to ask would be if we can parametrize surfaces up to rigid body motions. In this section, we will answer those two question by showing that the fundamental forms determine when two surfaces are congruent and that if we define fundamental forms fulfilling certain compatibility conditions there will be a surface in $\mathbb{R}^3$ admitting them. This will constitute the *fundamental theorem of surfaces*. The exposition below is completely based on [Küh15] and [Pal03]. Again we will work only locally, i.e. assume that we have global parametrizations ψ of our surfaces.

We start with a simple definition.

Definition 1.11 (Standard frame)**.** Let $\mathcal{S}$ be a regular surface with parametrization $\psi\colon \Omega \to \mathbb{R}^3$, then at each coordinate $\xi \in \Omega$ we define the *standard frame* at ξ by

$$F(\xi) = (F_1(\xi), F_2(\xi), F_3(\xi)) := (\partial_1\psi(\xi), \partial_2\psi(\xi), (n \circ \psi)(\xi)). \tag{1.4}$$

This yields a matrix-valued map $F\colon \Omega \to O(3)$ called a *frame field*, where we consider the different components as rows.

Note, that $F(\xi)$ is a basis of $\mathbb{R}^3$, where the first two basis vectors are the canonical basis of $T_{\psi(\xi)}\mathcal{S}$, and thus we can express each vector of $\mathbb{R}^3$ in it. Especially, we can apply this to the derivatives of F itself, i.e. determine coefficients P_{ji}^k such that

$$\partial_k F_j(\xi) = \sum_{i=1}^{2} P_{ji}^k(\xi) F_i(\xi). \tag{1.5}$$

The P^k_{ji} define two matrices P^k, $k = 1, 2$ and we can write this as a pair of equations of matrix-valued functions

$$\partial_1 F = FP^1 \tag{1.6a}$$
$$\partial_2 F = FP^2, \tag{1.6b}$$

which are called the *frame equations* for $\mathcal{S}$.

In the following, we will show that the transition matrices P^k can be calculated from the coefficients of the first and second fundamental form. Then we can consider the frame equations as a coupled pair of first-order partial differential equations for the frame field F, and it will follow from Frobenius' theorem that we can solve these equations and then by another integration recover the surface parametrization ψ.

We start with the first part.

Proposition 1.1 ([Pal03])**.** *Let $\mathcal{S}$ be a regular surface with parametrization $\psi\colon \Omega \to \mathbb{R}^3$, and P^1, P^2 the transition matrices as defined above. Then*

$$P^1 = G^{-1}A^1 := \begin{pmatrix} g^{11} & g^{12} & 0 \\ g^{21} & g^{22} & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \frac{1}{2}\,\partial_{\xi_1} g_{11} & \frac{1}{2}\,\partial_{\xi_2} g_{11} & h_{11} \\ \partial_{\xi_2} g_{12} - \frac{1}{2}\,\partial_{\xi_1} g_{11} & \frac{1}{2}\partial_1 g_{22} & h_{12} \\ -h_{11} & -h_{12} & 0 \end{pmatrix} \tag{1.7a}$$

$$P^2 = G^{-1}A^2 := \begin{pmatrix} g^{11} & g^{12} & 0 \\ g^{21} & g^{22} & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \frac{1}{2}\,\partial_2 g_{11} & \partial_2 g_{12} - \frac{1}{2}\,\partial_1 g_{22} & h_{12} \\ \frac{1}{2}\partial_1 g_{22} & \frac{1}{2}\partial_2 g_{22} & h_{22} \\ -h_{12} & -h_{22} & 0 \end{pmatrix}. \tag{1.7b}$$

Proof. Remember, that $\partial_k F_j = \sum_{i=1}^2 P^k_{ji} F_i$ and thus when we apply $\langle \cdot, F_l \rangle$ we get

$$\langle \partial_k F_j, F_l \rangle = \sum_{i=1}^{2} P^k_{ji} \langle F_i, F_l \rangle. \tag{1.8}$$

If we define matrices $G := (\langle F_i, F_l \rangle)_{il} = \begin{pmatrix} g^{11} & g^{12} & 0 \\ g^{21} & g^{22} & 0 \\ 0 & 0 & 1 \end{pmatrix}$ and $A^k := \langle \partial_k F_j, F_l \rangle_{jl} \in \mathbb{R}^{3\times 3}$ equation (1.8) becomes

$$A^k = GP^k,$$

and have to compute the coefficients of A^k to prove the proposition. For $i = 1, 2$ we get:

$$A^k_{ii} = \langle \partial_k \partial_i \psi, \partial_i \psi \rangle = \frac{1}{2} \partial_k \langle \partial_i \psi, \partial_i \psi \rangle = \frac{1}{2} \partial_k g_{ii}.$$

Next, we note $\partial_1 g_{12} = \partial_1 \langle \partial_1 \psi, \partial_2 \psi \rangle = \langle \partial_1 \partial_1 \psi, \partial_2 \psi \rangle + \langle \partial_1 \psi, \partial_1 \partial_2 \psi \rangle = \langle \partial_1 \partial_1 \psi, \partial_2 \psi \rangle + \frac{1}{2} \partial_2 g_{11} \rangle$, and thus

$$A^1_{21} = \langle \partial_1 \partial_1 \psi, \partial_2 \psi \rangle = \partial_1 g_{12} - \frac{1}{2} \partial_2 g_{11},$$

and in the same way it follows

$$A^2_{12} = \langle \partial_2 \partial_2 \psi, \partial_1 \psi \rangle = \partial_2 g_{12} - \frac{1}{2} \partial_1 g_{22},$$
$$A^1_{12} = \langle \partial_2 \partial_1 \psi, \partial_1 \psi \rangle = \frac{1}{2} \partial_2 g_{11},$$
$$A^2_{12} = \langle \partial_2 \partial_2 \psi, \partial_1 \psi \rangle = \frac{1}{2} \partial_2 g_{11}.$$

For $i = 1, 2$, we get

$$A^k_{i3} = \langle \partial_k(n \circ \psi), \partial_i \psi \rangle = \langle D_p n(\partial_k \psi), \partial_i \psi \rangle = h_{ki},$$

and since F_3 is orthogonal to F_i, we have

$$0 = \partial_k \langle F_3, F_i \rangle = \langle \partial_k F_3, F_i \rangle + \partial_k \langle F_3, \partial_k F_i \rangle,$$

and hence

$$A^k_{3i} = -A^k_{i3}.$$

Finally since $\langle F_3, F_3 \rangle = 1$ it follows $\partial_k \langle F_3, F_3 \rangle = 0$, so

$$A^k_{33} = 0.$$

□

In the following, we will consider the matrix-valued functions G, G^{-1}, A^k, and P^k as being defined by the formulas in the above proposition. From this, we can deduce necessary conditions on the coefficients on the first and second fundamental form to stem from a regular surface.

Corollary 1.1 (Gauß-Codazzi Equations, [Pal03])**.** *If (g_{ij}) and (h_{ij}) are the coefficients of the first and second fundamental forms of a regular surface $\mathcal{S}$ with parametrization $\psi \colon \Omega \to \mathcal{S}$, then the matrix-valued functions P^1 and P^2 defined on Ω by Proposition 1.1 satisfy*

$$\partial_2 P^1 - \partial_1 P^2 = P^1 P^2 - P^2 P^1. \tag{1.9}$$

Proof. We differentiate the first frame equation in (1.6) with respect to ξ_2 and the second with respect to ξ_1 and then by symmetry of second derivatives it follows

$$\partial_2 F P^1 + F \partial_2 P^1 = \partial_1 F P^2 + F \partial_1 P^2.$$

Substituting for $\partial_1 F$ and $\partial_2 F$ their values from the frame equations (1.6) gives

$$F(\partial_2 P^1 - \partial_1 P^2 - (P^1 P^2 - P^2 P^1))$$

and the corollary follows from F being an invertible matrix. □

Next, we want to show that the Gauß-Codazzi equations are in fact sufficient conditions on the coefficients of the first and second fundamental form to stem from a surface, i.e. we will see that just given the coefficients we can construct a surface with those fundamental forms.

Theorem 1.1 (Fundamental Theorem of Surfaces, [Pal03])**.** *Congruent regular surfaces in $\mathbb{R}^3$ have the same first and second fundamental form and conversely, two parametric surfaces with the same first and second fundamental forms are congruent.*

Moreover, if $g \colon \Omega \to \mathbb{R}^{2\times 2}$, and $h \colon \Omega \to \mathbb{R}^{2\times 2}$ are C^2 quadratic forms on an open and connected domain $\Omega \subset \mathbb{R}^2$ fulfilling the Gauß-Codazzi equations, then there exists a regular surface $\psi \colon \Omega \to \mathcal{S} \subset \mathbb{R}^3$ with g and h as (matrix representations of) first and second fundamental form.

Proof. The first point is clear from the definition of the fundamental forms and the invariance of the Euclidean scalar product to rigid body motions.

For the second statement, let $\psi^i \colon \Omega \to \mathcal{S}^i \subset \mathbb{R}^3, i = 1, 2$ be two parametric surfaces such that they have the same first and second fundamental form. After translation, we can assume that both map some coordinates ξ to the origin of $\mathbb{R}^3$. By assumption, we have $(D_\xi \psi^1)^\intercal D_\xi \psi^1 = g_0 = (D_\xi \psi^2)^\intercal D_\xi \psi^2$ and thus after an orthogonal transformation we can assume the standard frames F^1 of $\mathcal{S}^1$ and F^2 of $\mathcal{S}^2$ to agree at ξ. Since ψ^1 and ψ^2 have identical frame equations (1.6), which are integrable by Frobenius' theorem, it follows that F^1 and F^2 agree on all of Ω. In particular, we have $D_{\tilde{\xi}} \psi^1 = D_{\tilde{\xi}} \psi^2$ for all $\tilde{\xi}$, implying that ψ^1 and ψ^2 only differ by a constant, and as they agree at ξ they are identical, proving the second part.

Now, for the third part, note that since g is positive definite for each ξ it is invertible and thus G^{-1} and P^k from Proposition 1.1 are well-defined. Moreover, since the Gauß-Codazzi equations are exactly the integrability conditions of the Frobenius theorem, it follows that we can integrate the frame equations uniquely given arbitrary initial values for F at some $\xi_0 \in \Omega$. For these initial values, we choose a basis $F(\xi_0)$ of $\mathbb{R}^3$ such that $\langle F_i, F_j \rangle = G_{ij}(\xi_0)$, which is possible since g and hence G are positive definite.

Having a frame field F, we need to solve the system $\partial_i \psi = F_i, i = 1, 2$ to get a parametrization $\psi \colon \Omega \to \mathbb{R}^3$. This requires another application of the Frobenius theorem and now the compability condition is

$$\partial_2 F_1 = \partial_1 F_2,$$

which by the frame equations becomes

$$\sum_j P^2_{j1} F_j = \sum_j P^1_{j2} F_j.$$

If we inspect the definitions of P^k in Proposition 1.1, we in fact see that the second column of P^1 agrees with the first of P^2. Hence we can find a unique ψ with $\psi(\xi_0) = 0$ and $\partial_i \psi = F_i$ for $i = 1, 2$.

It remains to show that ψ defines a regular surface in $\mathbb{R}^3$ with g and h as matrix representations of the first and second fundamental form. This means we have to show for all $\xi \in \Omega$

- $\operatorname{rank} D_\xi \psi = 2$,
- $F_3(\xi) \perp T_{\psi(\xi)} \mathcal{S}$,
- $\|F_3(\xi)\| = 1$,
- $D_\xi \psi^\intercal D_\xi \psi = g_{\psi(\xi)}$, and
- $D_\xi F_3^\intercal D_\xi \psi = h_{\psi(\xi)}$.

As first step we prove that $\Phi := (\langle F_i, F_j \rangle)_{ij} = G$, which implies all but the last bullet points of the list. We compute $\partial_1 \Phi_{ij} = \langle \partial_1 F_i, F_j \rangle + \langle F_i, \partial_1 F_j \rangle = (A^1)_{ij} + (A^1)_{ji}$, and thus $\partial_1 \Phi = A^1 + (A^1)^\intercal$. If we look at the definition of A^1 we immediately see that $\partial_1 G = A^1 + (A^1)^\intercal$, thus $\partial_1 \Phi = \partial_1 G$ and by a similar computation $\partial_2 \Phi = \partial_2 G$. Therefore, Φ and G only differ by a constant and since they agree at ξ_0 they are identical.

For the last point, we compute the second fundamental form of $\mathcal{S} = \psi(\Omega)$, again using the frame equations,

$$\begin{aligned}\langle \partial_1 F_3, F_j\rangle = \langle (FP^1)_3, F_j\rangle &= (g^{11}h_{11} + g^{12}h_{12})\langle F_1, F_j\rangle + (g^{12}h_{11} + g^{22}h_{12})\langle F_1, F_j\rangle \\ &= (g^{11}h_{11} + g^{12}h_{12})g_{1j} + (g^{12}h_{11} + g^{22}h_{12})g_{2j} \\ &= h_{11}(g^{11}g_{1j} + g^{12}g_{2j}) + h_{12}(g^{21}g_{1j} + g^{22}g_{2j}) \\ &= h_{11}\delta_{1j} + h_{12}\delta_{2j}.\end{aligned}$$

Hence, $\langle \partial_1 F_3, F_1\rangle = h_{11}$, $\langle \partial_1 F_3, F_2\rangle = h_{12}$ and a similar computation for $\partial_2 F_3$ proves that $D_\xi F_3^\intercal D_\xi \psi = h_{\psi(\xi)}$.

□

1.2 Discrete Surfaces

Next, we will study discrete surfaces, a certain class of polygonal meshes. These are the kind of objects we are actually dealing with in our applications/numerical implementations, hence we want to apply the differential and geometric notions from Section 1.2 to them. However, they required the surface to be sufficiently smooth but polygonal meshes are piecewise affine and thus globally only of class $\mathcal{C}^0$. One aims to introduce discrete equivalents of notions from differential geometry, which allow computing approximations of the properties of a smooth surface. In this section, we will provide a formal definition for discrete surfaces and briefly introduce several geometric notions. As before, this chapter is not meant to provide a complete introduction to *discrete differential geometry (DDG)*, for this, we refer the reader to existing literature such as [Hee17, CdGDS13] on which this section is based.

Topology The connectivity $\mathcal{S}_h$ of a triangle mesh can be represented as a graph structure. We have a finite set of vertices $\mathcal{V} = \{v_1, \dots, v_{|\mathcal{V}|}\}$ and a set of triangle faces $\mathcal{F} = \{f_1, \dots, f_{|\mathcal{F}|}\} \subset \mathcal{V} \times \mathcal{V} \times \mathcal{V}$. From these two we can furthermore deduce a set of edges $\mathcal{E} = \{e_1, \dots, e_{|\mathcal{E}|}\} \subset \mathcal{V} \times \mathcal{V}$. Note, that we could equivalently deduce the set of faces $\mathcal{F}$ from the set of edges $\mathcal{E}$ and all further structural properties of the mesh, such as neighboring relationships or boundary can be derived from them. We will now derive a topology for the mesh solely based on this connectivity. This will yield a language clearly separating the topology of a discrete surface from the geometry induced by an embedding, which will come in handy later on. We begin with introducing the topological notions that correspond to our connectivity.

Definition 1.12 (Abstract simplicial complex)**.** An *abstract simplicial complex* $\mathcal{K}$ consists of a set of vertices $\mathcal{V}$ together with a set Δ of finite non-empty subsets of $\mathcal{V}$, called *simplices*, such that

1. if $\sigma \in \Delta$ and $\emptyset \neq \tau \subset \sigma$ then $\tau \in \Delta$, and
2. for every $v \in \mathcal{V}$ we have $\{v\} \in \Delta$.

We call a $\sigma \in \Delta$ with $p+1$ elements a *p-simplex*. The *dimension* of $\mathcal{K}$ is the largest p such that it contains a p-simplex.

Such an abstract simplicial complex provides a combinatorial way of describing the structure of a simplicial complex. It is basically a construction plan for the 'gluing' of simplices to form a simplicial complex. Note, that giving the connectivity $\mathcal{S}_h =$

$(\mathcal{V}, \mathcal{E}, \mathcal{F})$ of a triangle mesh is equivalent to giving a set of vertices $\mathcal{V}$ and a set of simplices Δ consisting of the vertices, edges and triangles, hence by abuse of notation we refer by $\mathcal{S}_h$ to both.

Remark. In this thesis, we are solely interested in 2-dimensional abstract simplicial complexes with a finite number of vertices (and thus simplices). Especially, we will assume this implicitly in the following when we are talking about them. One could also develop the following notions for infinite(-dimensional) abstract simplicial complexes, but this would require an even more abstract language which is not desirable at this point.

Now that we have the combinatorial structure of a simplicial complex, we will associate to it a topological space.

Definition 1.13 (Geometric realization)**.** Let $\mathcal{K}$ be an abstract simplicial complex with finitely many vertices $\mathcal{V} = \{v_1, \dots, v_{|\mathcal{V}|}\}$. Then we define the *geometric realization* $|\mathcal{K}|$ of $\mathcal{K}$ as the set

$$\bigcup_{\{v_{i_1},\dots,v_{i_K}\}\in\Delta} \mathrm{Conv}(e_{i_1}, \dots, e_{i_K}) \subset \mathbb{R}^{|\mathcal{V}|} \tag{1.10}$$

together with subspace topology given by $\mathbb{R}^{|\mathcal{V}|}$.

In the following, we will by abuse of notion often refer to the connectivity $\mathcal{S}_h$ of a triangle mesh as topological space, by which we mean exactly this geometric realization even though we might not explicitly use the notation $|\mathcal{S}_h|$.

Remark. We could describe this as a functor from the category of abstract simplicial complexes to the category of topological spaces such that abstract p-simplices are sent to geometric p-simplices and the image of an arbitrary abstract simplicial complex is constructed from them. This would give a more abstract viewpoint, which we do not need in this thesis.

Furthermore, if we have a valid triangle mesh, i.e. without self-intersections etc., then it is homeomorphic to the geometric realization of the simplicial complex. Hence, in those cases, we would not need to construct the topological space differently, but we wanted to stress that we can construct it just from the connectivity to more clearly separate it from the geometry of the surface.

Lastly, we have to get rid of unwanted singularities, such as holes or lower dimensional structures, to get to the notion of a discrete surface. Note, that the following definition is equivalent to the one used for example in [DKT08], just derived in a more abstract fashion.

Definition 1.14 (Discrete surface)**.** A *discrete surface* or *two-dimensional discrete manifold* $\mathcal{S}_h$ is an abstract simplicial 2-complex, such that in its geometric realization for each vertex the union of all incident simplices is homeomorphic to a disk or a half-disk if the vertex is on the boundary.

Dual topology Additionally to the primal topology of the surface, we will also consider its dual. In the dual topology, or dual graph, the faces of the surface become the vertices, that is

$$\mathcal{V}^* = \mathcal{F}.$$

The edges in the dual graph describe the neighboring relationships of primal faces and are given by

$$\mathcal{E}^* = \{(k, l) \in \mathcal{F} \times \mathcal{F} \mid k \text{ and } l \text{ share a common edge}\}.$$

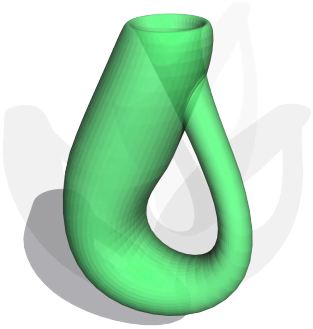
Note, that the dual edges are in one-to-one correspondence with primal interior edges by mapping a dual edge to the common primal edge of the two faces. Thus, they will often be indexed and referred to by their primal counterpart. The dual graph allows us to formalize notions such as iterating over neighboring triangles, which we will need later on when talking about conditions on edge lengths and dihedral angles to be admissible.

Orientation The order of the local indices of nodes within one face determines the orientation of the face and hence of the discrete surface. In the following, only orientable (discrete) surfaces will be considered, and thus we will implicitly assume that indices are ordered consistently.

Geometry So far, we have studied the underlying topology of a triangle mesh, yet in practice, of course, we work with geometric realizations of this topology in three-dimensional space. Hence, we consider these next.

Definition 1.15 (Embedding)**.** Let $\mathcal{S}_h$ be a discrete surface. An *embedding* of $\mathcal{S}_h$ is an injective piecewise linear map $X\colon |\mathcal{S}_h| \to \mathbb{R}^3$ such that it is an homeomorphism onto its image. We call a discrete surface together with an embedding an *embedded discrete surface* **S**.

We will not always be able to only consider embeddings of a discrete surface due to their global injectiveness. Therefore, we also consider immersions of discrete surfaces, which replace this global condition by a local one allowing self-intersection of the surface as long as for each vertex a neighborhood is embedded. This enables considering discrete surfaces such as the triangulated Klein bottle shown in the inset figure.



Definition 1.16 (Immersion)**.** Let $\mathcal{S}_h$ be a discrete surface. An *immersion* of $\mathcal{S}_h$ is a piecewise linear map $X\colon |\mathcal{S}_h| \to \mathbb{R}^3$ such that it is a local injection for each one-ring of faces around a vertex, and thus a local homeomorphism. We call a discrete surface together with an immersion an *immersed discrete surface* **S**.

As the embedding or immersion X is piecewise linear it is uniquely determined by its restriction to the vertices $X|_{\mathcal{V}}$. To simplify formulas, we denote for $v \in \mathcal{V}$ by X_v the image of v and if we have an enumeration of $\mathcal{V}$ we denote for $i \in \mathbb{N}$ by X_i the image of v_i. We denote the image $X(e)$ of an edge $e \in \mathcal{E}$ by E_e, or again if we have an enumeration by E_i the image of e_i. If we consider a face $f = (v_{i_0}, v_{i_1}, v_{i_2}) \in \mathcal{F}$, we denote the embedded triangle by

$$T(f) = \operatorname{Conv}(X_{i_0}, X_{i_1}, X_{i_2}) \subset \mathbb{R}^2.$$

Furthermore, we define $X_j(f) := X(v_{i_j})$ and

$$E_j(f) := X_{j-1}(f) - X_{j+1}(f)$$

for $j \in \{0, 1, 2\}$, where in the last equation the indices are to be read modulo 3.

Parametrization In Section 1.1, we gave formulas for many objects in terms of a (local) parametrization of the surface. Now, for an embedded triangle of a discrete surface, we can easily derive a local parametrization, as well. For this, we consider the unit triangle in $\mathbb{R}^2$

$$\omega := \operatorname{Conv}\left(\begin{pmatrix}0\\0\end{pmatrix}, \begin{pmatrix}1\\0\end{pmatrix}, \begin{pmatrix}0\\1\end{pmatrix}\right) \subset \mathbb{R}^2$$

as reference domain. Then we get our local parametrization as the affine mapping

$$\Psi_f \colon \omega \to T(f), \quad (\xi_1, \xi_2) \mapsto \xi_1 X_1(f) + \xi_2 X_2(f) + (1 - \xi_1 - \xi_2) X_0(f)$$

for the baraycentric coordinates $\xi \in \omega$. We can collect all these local parametrization into one global map on the reference domain $\Omega_h = \omega \times \mathcal{F}$

$$\Psi \colon \Omega_h \to X(\mathcal{S}_h), \quad (\xi, f) \mapsto \Psi_f(\xi)$$

and by abuse of notation call this a global parametrization of the immersed discrete surface. Later on, we will drop the explicit dependence on f and use Ψ whenever possible to simplify our notation.

1.2.1 Fundamental Forms

Now that we have introduced the topology and geometry of discrete surfaces we go on to study their geometry more thoroughly. In Section 1.1, the first and second fundamental form where crucial tools for understanding the geometry of differentiable surfaces. Hence, we will introduce their discrete counterparts in this section.

Discrete first fundamental form Remember that for an regular embedded surface $\mathcal{S} \subset \mathbb{R}^3$ with (local) parametrization ψ we can represent the first fundamental form by $g = D\psi^\intercal D\psi$. Furthermore, the local parametrization Ψ of an immersed discrete surface is affine, thus its derivative is constant on each triangle $f \in \mathcal{F}$ and given by

$$D\Psi|_f = \left(\frac{\partial \Psi_f}{\partial \xi_1}, \frac{\partial \Psi_f}{\partial \xi_2}\right) = \Big[X_1(f) - X_0(f) \,\Big|\, X_2(f) - X_0(f)\Big] \tag{1.11}$$

$$= \Big[E_2(f) \,\Big|\, -E_1(f)\Big] \in \mathbb{R}^{3\times 2}\,. \tag{1.12}$$

From this the definition of a discrete first fundamental form follows canonically.

Definition 1.17 (Discrete first fundamental form)**.** Let $\mathcal{S}_h$ be a discrete surface with immersion X. The elementwise constant *discrete first fundamental form* is given by

$$G|_f := (D\Psi|_f)^\intercal D\Psi|_f = \begin{pmatrix} \|E_2(f)\|^2 & -\langle E_1(f), E_2(f)\rangle \\ -\langle E_1(f), E_2(f)\rangle & \|E_1(f)\|^2 \end{pmatrix} \in \mathbb{R}^{2\times 2} \tag{1.13}$$

for each $f \in \mathcal{F}$.

Discrete second fundamental form We also need a discrete counterpart of the second fundamental form to be able to talk about extrinsic invariants of discrete surfaces. To this end, we will present a triangle-averaged discrete second fundamental form H introduced in [HRWW12]. Combining these, two we can derive a matrix representation of a shape operator living on triangles by setting $S|_f := G|_f^{-1} H|_f \in \mathbb{R}^{2\times 2}$.

First, we need to introduce some basic geometric notions.

Definition 1.18 (Normals)**.** For an immersed discrete surface $\mathbf{S}$, we define the face normal $N_f \in S^2$ on the face $f = (v_{i_0}, v_{i_1}, v_{i_2}) \in \mathcal{F}$ by

$$N_f := \frac{(X_{i_1} - X_{i_0}) \times (X_{i_2} - X_{i_0})}{\|(X_{i_1} - X_{i_0}) \times (X_{i_2} - X_{i_0})\|}. \tag{1.14}$$

Furthermore, for an edge e with adjacent faces f_l and f_r we set

$$N_e := \frac{\alpha_l N_{f_l} + \alpha_r N_{f_r}}{\|\alpha_l N_{f_l} + \alpha_r N_{f_r}\|}, \tag{1.15}$$

with dihedral angle bisecting weights $\alpha_l = \alpha_l = \frac{1}{2}$.

Definition 1.19 (Dihedral angle)**.** For an immersed discrete surface $\mathbf{S}$, we define the dihedral angle of an edge e with adjacent faces f_l and f_r as

$$\theta_e = \angle\left(N_{f_l},\, N_{f_r}\right), \tag{1.16}$$

where, as before, E denotes the embedded edge.

Now, remembering Definition 1.7 and (1.11), we get for the entries of a matrix representation H on a face f

$$\begin{aligned} H_{11} &= \Big\langle dN(E_2), E_2 \Big\rangle_{\mathbb{R}^3}, \quad H_{12} = -\Big\langle dN(E_2), E_1 \Big\rangle_{\mathbb{R}^3}, \\ H_{21} &= -\Big\langle dN(E_1), E_2 \Big\rangle_{\mathbb{R}^3}, \quad H_{22} = \Big\langle dN(E_1), E_1 \Big\rangle_{\mathbb{R}^3}, \end{aligned}$$

where $e_0, e_1, e_2 \in \mathcal{E}$ are the edges of f and $N_0, N_1, N_2 \in S^2$ the corresponding edge normals. Now, one can consider dN as a discrete 1-form in the context of discrete exterior calculus (DEC), which acts on line segments connecting edge midpoints. At this point, we will not explain in more detail what is meant by this and refer to available literature such as [DKT08]. What is important to us is, that one can use the notation of DEC to derive

$$dN(E_k) = 2(N_i - N_j), \tag{1.17}$$

where the notation is such that i, j, and k are complimentary indices of edges in f. This can be used to simplify $H|_f$ to

$$\begin{aligned} H_{11} &= 2\,\langle N_0, E_2\rangle + 2\,\langle N_1, E_0\rangle, \\ H_{12} &= H_{21} = 2\,\langle N_0, E_2\rangle, \\ H_{22} &= 2\,\langle N_0, E_2\rangle + 2\,\langle N_2, E_1\rangle, \end{aligned}$$

where we have used $\langle N_i, E_i\rangle = 0$ and $E_0 + E_1 + E_2 = 0$. Hence, we get the representation

$$H|_f = 2\sum_{i=0}^{2} \langle N_i, E_{i-1}\rangle\, M_i\,,$$

with a basis (M_0, M_1, M_2) of symmetric 2×2-matrices given by

$$M_0 = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}, \quad M_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \quad M_2 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}. \tag{1.18}$$

We can give a more geometric interpretation of this by observing

$$\langle N_i, E_{i-1}\rangle = -2\frac{a_f}{\|E_i\|}\cos\frac{\theta_i + \pi}{2}.$$

All this then accumulates to the following definition.

Definition 1.20 (Discrete second fundamental form, [Hee16])**.** Let $\mathcal{S}_h$ be a discrete surface with immersion X. The elementwise constant triangle-averaged *discrete second fundamental form* is given by

$$H|_f = -4a_f \sum_{i=0}^{2} \frac{\cos\frac{\theta_i+\pi}{2}}{\|E_i\|} M_i \tag{1.19}$$

for each $f \in \mathcal{F}$, where (M_0, M_1, M_2) is the basis of symmetric 2×2-matrices from (1.18).

Note, that this is one possible choice of discrete second fundamental form, proposed by Wardetzky in unpublished work and elaborated by Heeren in [Hee16], which is particularly suitable for our applications while other choices do exist. Lastly, we can use both triangle-based discrete fundamental forms to define a discrete notion of the shape operator.

Definition 1.21 (Discrete shape operator, [Hee16])**.** Let $\mathcal{S}_h$ be a discrete surface with immersion X. The elementwise constant triangle-averaged *discrete shape operator* is given by

$$S|_f = G|_f^{-1} H|_f \tag{1.20}$$

for each $f \in \mathcal{F}$.

Remark (Curvature)*.* In the continuous case, we have derived different notions of the surface's curvature from the eigenvalues of the shape operator. We can define corresponding discrete notions of curvature the same way using the triangle-averaged shape operator. Especially, we can define a triangle-averaged mean curvature $\operatorname{tr} S|_f$ with explicit formula

$$\operatorname{tr} S|_f = -\sum_{i=0}^{2} \frac{\cos\frac{\theta_i+\pi}{2}}{a_f} \|E_i\|. \tag{1.21}$$

1.3 Thin Shell Modeling

Our goal is to study deformations of discrete surfaces and their applications. In many of those, for instance, animation movies, the discrete surfaces represent complex shapes such as the skin of characters and the deformations are supposed to model natural motions. To create such, we need a physically plausible model for the deformation behavior of discrete surfaces.

We will go one step back to continuous surfaces and present the model developed within this context by Heeren et al. in [HRWW12, HRS$^+$14, Hee16]. It starts with the assumption that the surfaces represent *thin shells*, three-dimensional solids with a high ratio from width to thickness. To derive two-dimensional models from this one starts with three-dimensional elasticity and investigates deformation energies of solids $\Omega_\delta \subset \mathbb{R}^3$ with δ being a tiny but finite thickness of the material, and then considers the limit $\delta \to 0$ based on the notion of Γ-convergence. Qualitative insights from the mathematical rigorous study of this limit in [LR95, LR96, FJM02, FJGM03] were then used to define a generic thin shell model.

Elasticity We begin with introducing our notation for three-dimensional elasticity, which is based on [Hee17]. Let $\mathcal{O} \subset \mathbb{R}^3$ be a solid object with boundary and $\phi \in W^{1,2}(\mathcal{O}; \mathbb{R}^3)$ a potentially large and nonlinear deformation. We assume that ϕ is

orientation preserving, i.e. $\det D\phi(x) > 0$ for all $x \in \mathcal{O}$, and injective. Especially this implies that it is a homeomorphism onto its image. We then postulate that we have an *hyperelastic* material with a *deformation energy* given as integral of an elastic energy density W, i.e.

$$\mathcal{W}[\phi, \mathcal{O}] = \int_{\mathcal{O}} W(D\phi)\mathrm{d}x. \tag{1.22}$$

A fundamental axiom of continuum mechanics is the *frame indifference* of W, i.e. $W(D\phi) = W(Q^\intercal D\phi Q)$ for all $Q \in SO(3)$, furthermore we assume $\mathcal{O}$ to be *isotropic*, i.e. $W(D\phi) = W(D\phi Q)$ for all $Q \in SO(3)$. From these two assumptions it follows by the Rivlin-Ericksen-Theorem [RE55] that the energy density only depends on $\|D\phi\|_F, \|\operatorname{cof} D\phi\|_F$, and $\det D\phi$. Additionally, we assume that isometries ϕ are local minimizers with $W(D\phi) = 0$, this holds especially for all rigid body motions, and that $W(D\phi) \to \infty$ for $\det D\phi$, hence $W(D\phi) = \infty$ for $\det D\phi \leq 0$. These assumptions will be upheld for the rest of this section.

A particular choice for a nonlinear energy density we will consider was introduced in [Wir10] as

$$W(D\phi) = \frac{\mu}{2}\|D\phi\|_F^2 + \frac{\lambda}{4}(\det D\phi)^2 - \left(\mu + \frac{\lambda}{2}\right)\log\det D\phi - \mu - \frac{\lambda}{4}, \tag{1.23}$$

which is a concrete instance of a Mooney-Rivling model [Cia88] and fulfills all our assumptions. Furthermore, we are also interested in a generic isotropic material described by the *St. Venant-Kirchhoff* density

$$W^{\mathrm{StVK}}(F) = \frac{\lambda}{8}(\operatorname{tr}(F^TF - \mathrm{Id}))^2 + \frac{\mu}{4}\operatorname{tr}(F^TF - \mathrm{Id})^2. \tag{1.24}$$

In both cases μ and λ are positive material constants.

We can use this to define a notion of distance between different shapes.

Definition 1.22 (Elastic dissimilarity measure)**.** Let $\mathcal{O}_A$ and $\mathcal{O}_B$ be two solid objects/shapes. Then we define the *elastic dissimilarity measure* for a given energy density W as

$$\mathrm{d}^2_{\mathrm{elast}}(\mathcal{O}_A, \mathcal{O}_B) := \min_{\phi\colon \phi(\mathcal{O}_A)=\mathcal{O}_B} \int_{\mathcal{O}_A} W(D\phi)\mathrm{d}x. \tag{1.25}$$

1.3.1 Membrane and Bending Energies

Now, that we have established basic notions, we turn towards the model of Heeren et al., which consists of a membrane and a bending part, that we will present separately. In both cases, they used qualitative insights into analytical models to develop a physically-sound deformation energy. Especially they took into consideration on which properties of the surface the limit depends. The qualitative parts of the results, such as physical constants in the limit, are not relevant, as they will be chosen individually in different applications.

Membrane model For the membrane model, we need the right *Cauchy-Green strain tensor* $C[\phi] = D\phi^\intercal D\phi$, which is a pointwise linear operator describing the infinitesimal change of lengths on the surface under the deformation ϕ. A two-dimensional representation of $C[\phi] \in \mathbb{R}^{3\times3}$ by a distortion tensor $\mathcal{G}[\phi] \in \mathbb{R}^{2\times2}$ was derived in [LDRS05, CLR04]. In particular, we can write

$$\mathcal{G}[\phi] = g^{-1}g_\phi \tag{1.26}$$

with g and g_ϕ denoting the first fundamental form of the undeformed and deformed configuration, respectively.

Then Heeren et al. derived a membrane shell energy $\mathcal{W}_{\text{mem}}$ from the results of Le Dret and Raoult in [LR95, LR96]. It is supposed to measure the tangential stretching and shearing induced by a deformation ϕ of the surface $\mathcal{S}$.

Definition 1.23 (Membrane energy)**.** Let $\mathcal{S}$ be a regular surface and ϕ a deformation. Then the *membrane energy* is given by

$$\mathcal{W}_{\text{mem}}[\mathcal{S}, \phi] = \int_{\mathcal{S}} W_{\text{mem}}(\mathcal{G}[\phi]) \mathrm{d}a, \tag{1.27}$$

with density (1.23).

Bending energy For the bending part of the model, Heeren et al. used results from Friesecke et al. [FJM02, FJGM03], who investigated the energy for isometric deformations of thin shells in the Γ-limit. This led to the following generic bending shell energy, which measures the change of the second fundamental form for isometric deformations.

Definition 1.24 (Bending energy)**.** Let $\mathcal{S}$ be a regular surface and ϕ a deformation. Then the *bending energy* is given by

$$\mathcal{W}_{\text{bend}}[\mathcal{S}, \phi] = \int_{\mathcal{S}} W_{\text{bend}}(S_\phi^{rel}) \mathrm{d}a\,, \tag{1.28}$$

where in general we make use of the density

$$W_{\text{bend}}(A) = \alpha(\operatorname{tr} A)^2 + (1-\alpha)\,\|A\|_F^2\,, \quad \alpha \in \{0, 1\}\,. \tag{1.29}$$

Recall that the matrix representation of the relative shape operator in the parameter domain Ω was defined in Section 1.1.1 as

$$s_\xi^{rel}[\phi] = s_\xi - s_\xi^*[\phi] = g_\xi^{-1}(h_\xi - \tilde{h}_\xi)\,.$$

Based on this it was verified in [Hee16] that for $\alpha = 0$ one gets

$$\mathcal{W}_{\text{bend}}[\mathcal{S}, \phi] = \int_{\mathcal{S}} \|S_\phi^{rel}\|_F^2 \,\mathrm{d}a = \int_\Omega \operatorname{tr}\left(s_\xi^{rel}[\phi]^2\right) \sqrt{\det g}\,\mathrm{d}\xi\,, \tag{1.30}$$

and for $\alpha = 1$ one gets

$$\mathcal{W}_{\text{bend}}[\mathcal{S}, \phi] = \int_{\mathcal{S}} \left(\operatorname{tr} S_\phi^{rel}\right)^2 \mathrm{d}a = \int_\Omega \left(\operatorname{tr} s_\xi^{rel}[\phi]\right)^2 \sqrt{\det g}\,\mathrm{d}\xi\,. \tag{1.31}$$

Full elastic model Given a surface $\mathcal{S} \in \mathbb{R}^3$ representing a physical shell and a deformation $\phi\colon \mathcal{S} \to \mathbb{R}^3$, we consider the following generic elastic deformation energy

$$\mathcal{W}_{\mathcal{S}}[\phi] = \int_{\mathcal{S}} W_{\text{mem}}(\mathcal{G}[\phi]) + \eta W_{\text{bend}}(S_\phi^{rel})\,\mathrm{d}a, \tag{1.32}$$

where the bending weight η represents the squared thickness of the shell. Note that $\mathcal{W}_{\mathcal{S}}[\phi]$ is invariant with respect to rigid body motions, which is a desirable property from the standpoint of modeling, but will cause us some headaches later on when working with numerical schemes. With this energy, we can also define an elastic dissimilarity measure for surfaces as in Definition 1.22.

1.3.2 Discrete Deformations Energies

In the following, we will consider (discrete) deformations Φ of immersed discrete surfaces and present the discretization of the above model, which was also derived by Heeren et al. in [HRWW12, Hee16]. As before, we assume that the deformations are homeomorphisms onto their image, and hence the topology of the surface remains unchanged. Before, we have seen that the topology of a discrete surface is induced by its connectivity and in the following, we will restrict us to immersed discrete surfaces which share the same connectivity. Then studying discrete deformations comes to studying the differences between immersions of a discrete surface $\mathcal{S}_h$. We formalize this in the following two definitions.

Definition 1.25 (Dense correspondence)**.** We say that two immersed discrete surfaces $\mathbf{S}$ and $\tilde{\mathbf{S}}$ are in *dense correspondence* or in 1-to-1-correspondence if they share the same connectivity $\mathcal{S}_h$.

Definition 1.26 (Discrete deformations)**.** Let $\mathbf{S}$ and $\tilde{\mathbf{S}}$ be two immersed discrete surfaces in dense correspondence, i.e. we have a discrete surface $\mathcal{S}_h$ with two immersions $X\colon \mathcal{S}_h \to \mathbb{R}^3$ and $\tilde{X}\colon \mathcal{S}_h \to \mathbb{R}^3$. Then a *discrete deformation* $\Phi\colon \mathbf{S} \to \tilde{\mathbf{S}}$ is the unique piecewise affine map defined by its nodal values $\Phi(X(v)) = \tilde{X}(v)$ for all $v \in \mathcal{V}$.

Remark. Restricting ourselves to the study of immersed discrete surfaces in dense correspondence guarantees the well-definedness of an elastic dissimilarity measure (cf. (1.25)).

In the following, we will sometimes refer to connectivity properties of $\tilde{\mathbf{S}}$ with a tilde (e.g. $\tilde{f}$) even though they are the same as for $\mathbf{S}$. This allows us to easier refer to properties of the immersed surface, e.g. we can write $T(\tilde{f})$ for the embedded triangle of the deformed surface when actually the embedded property T differs rather than the connectivity f.

Discrete membrane model In Section 1.2.1, we have introduced an elementwise constant first fundamental form (cf. Definition 1.17) and now this is combined with the membrane model (1.27). Hence, to describe tangential distortions induced by Φ, we consider the elementwise constant discrete distortion tensor

$$\mathcal{G}[\Phi]|_f = (G|_f)^{-1} G^\Phi|_f \in \mathbb{R}^{2\times 2}. \tag{1.33}$$

Here, we denote by G^Φ the discrete first fundamental form on $\tilde{\mathbf{S}}$, and will continue to use this notation throughout this section. Putting this into the membrane model (1.27), one arrives at the following

Definition 1.27 (Discrete membrane energy, [HRWW12])**.**

$$\mathcal{W}_{\text{mem}}[\mathbf{S}, \tilde{\mathbf{S}}] := \int_{\mathbf{S}} W_{\text{mem}}(\mathcal{G}[\Phi])\mathrm{d}a = \sum_{f\in\mathcal{F}} a_f \cdot W_{\text{mem}}(\mathcal{G}[\Phi]|_f)\,, \quad \tilde{\mathbf{S}} = \Phi(\mathbf{S})\,. \tag{1.34}$$

In this energy, one can continue to use the energy density (1.23), where $\operatorname{tr} G[\Phi]|_f$ controls the change of edge lengths and $\det G[\Phi]|_f$ controls the local change of triangle area. Moreover, let us note that the use of (1.23) prevents the degeneration of triangles.

Discrete bending model Next, we will connect the triangle-averaged discrete second fundamental form from Definition 1.20 and the corresponding shape operator with the bending model (1.28) using the density (1.29). By choosing $\alpha = 1$, we can derive a discrete version of the Willmore energy.

Definition 1.28 (Discrete bending energy, [HRWW12])**.**

$$\mathcal{W}_{\text{bend}}[\mathbf{S}, \tilde{\mathbf{S}}] := \sum_{f \in \mathcal{F}} a_f \cdot \left(\operatorname{tr}(S_f - S_f^{\Phi}) \right)^2, \quad \tilde{\mathbf{S}} = \Phi(\mathbf{S}) \,. \tag{1.35}$$

Heeren has shown in [Hee16] that one can derive the following *Discrete Shells* bending model by simplifying the above bending energy, which is a different path than in the original publication [GHDS03].

Definition 1.29 (*Discrete Shells* bending energy, [GHDS03, Hee16])**.**

$$\mathcal{W}_{\text{bend}}^{\text{DS}}[\mathbf{S}, \tilde{\mathbf{S}}] := \sum_{e \in \mathcal{E}} \frac{(\theta_e - \tilde{\theta}_e)^2}{d_e} l_e^2, \,, \quad \tilde{\mathbf{S}} = \Phi(\mathbf{S}) \,, \tag{1.36}$$

where $d_e = \frac{1}{3}(a_f + a_{f'})$ for the two faces f, f' adjacent to $e \in \mathcal{E}$.

Discrete dissimilarity measure Finally, we are able to combine the membrane and bending part and arrive at a discrete deformation energy, which at the same time can also be considered as a dissimilarity measure as in (1.25).

Definition 1.30 (Discrete deformation energy, [HRWW12])**.** Let $\mathbf{S}$ and $\tilde{\mathbf{S}}$ be two immersed discrete surfaces in dense correspondence and let Φ be their unique affine deformation with $\tilde{\mathbf{S}} = \Phi(\mathbf{S})$. The *discrete deformation energy* $\mathcal{W}[\mathbf{S}, \tilde{\mathbf{S}}] = \mathcal{W}_{\mathbf{S}}[\Phi]$ is defined by

$$\mathcal{W}[\mathbf{S}, \tilde{\mathbf{S}}] = \mathcal{W}_{\text{mem}}[\mathbf{S}, \tilde{\mathbf{S}}] + \eta \mathcal{W}_{\text{bend}}^{\text{DS}}[\mathbf{S}, \tilde{\mathbf{S}}], \tag{1.37}$$

where the *bending weight* η represents the squared thickness of the shell. As the deformation is unique, we call this also the *discrete dissimilarity measure*.

Remark. We have associated the physical model of thin shells with discrete surfaces, hence we will also call them *discrete shells*.

Chapter 2

Riemannian Shape Spaces

Now that we have a grip on (discrete) surfaces and physical modeling, we are turning our attention towards mathematical modeling of collections of deformations as they occur for example in computer graphics. Kendall [Ken84] developed in this context the notion of a *shape space* in which points are complex geometric objects on their own. This is related to the concept of moduli spaces in pure mathematics, of which the projective space and Grassmannians are the perhaps most well-known examples. The first ingredient is to equip the shape space with a mathematical structure going beyond that of a set. In many instances, such a structure will not be that of a linear vector space, for instance in nodal positions linear interpolation will often lead to strange results. Hence, we turn to nonlinear structures and then equipping the shape space with the structure of a Riemannian manifold is a common approach. Through the local metric, one can induce a sense of distance which is based on geometrical or physical plausible models of deformations. Then many problems from computer graphics can be modeled as mathematical concepts from Riemannian geometry. Interpolation of two shapes will correspond to geodesics, extrapolation of movements to the exponential map, detail or pose transfer can be modeled as parallel transport, and keyframe interpolation can be achieved by Riemannian splines. Moreover, one can study the structure of the shape space itself to gain insights into the deformation of surfaces and use tools from Riemannian statistics to develop data-driven approaches. To apply this to real world data, one needs a discretization of the continuous concepts, which can be computed numerically. In this context, Rumpf and Wirth developed a variational time discretization of geodesic calculus together with a mathematically rigorous theory containing the convergence to continuous counterparts. This discretization has been applied to different problems in computer graphics and other areas and will pose the framework for the analysis of deformations in this thesis.

To this end, we will introduce very briefly the necessary concepts from Riemannian geometry in this chapter. From this, we will turn to the time-discretization, where we will present how to discretize the different notions but will omit the rigorous theory behind it completely. Lastly, how this can be applied to discrete surfaces together with the physical modeling of thin shells will be discussed.

2.1 Riemannian Manifolds in a Nutshell

As we have explained, our goal is to model the shape space as a *Riemannian manifold*, hence we will introduce the needed concepts from Riemannian geometry to read the following chapters. We do not aim to give an even remotely comprehensive introduction

into this very broad and interesting topic. Instead, we refer the reader to standard textbooks such as [dC92] and [Lan95] for a more detailed treatment. This section is also based on them as well as on [Hee17].

Remark (Dimension). In this section, everything will be presented on finite dimensional manifolds, although the geodesic calculus is set up in a way that it also works on infinite dimensional manifolds. Nevertheless, considering the second case would require more care and unnecessarily complicate this overview and the application to discrete shells we are interested in leads to a finite dimensional manifold anyway. If one is interested in more details on infinite dimensional manifolds, we refer to [Lan95].

We begin with the basic definitions introducing our objects of study.

Definition 2.1 (Differentiable manifold)**.** A *differentiable manifold* $\mathcal{M}$ of dimension $d < \infty$ is a set together with a family of injective maps $\psi_\alpha \colon U_\alpha \subset \mathbb{R}^d \to \mathcal{M}$ of open sets U_α of $\mathbb{R}^d$ into $\mathcal{M}$ such that

1. $\bigcup_\alpha \psi_\alpha(U_\alpha) = \mathcal{M}$, and
2. for any pair α, β with $\psi_\alpha(U_\alpha) \cap \psi_\beta(U_\beta) = W \neq \emptyset$, the sets $\psi_\alpha^{-1}(W)$ and $\psi_\beta^{-1}(W)$ are open in $\mathbb{R}^d$ and the map $\psi_\beta^{-1} \circ \psi_\alpha$ is differentiable.

The pair (U_α, ψ_α) with $p \in \psi_\alpha(U_\alpha)$ is called a *parametrization* of $\mathcal{M}$ at p.

Remark. Typically one assumes that the differentiable structure in Definition 2.1 is maximal with respect to the given conditions. This is can, in general, be achieved by extending the structure by parametrizations compatible with condition (2).

To simplify notation, we are going to assume in the following that we have indeed a global surjective parametrization $\psi \colon U \subset \mathbb{R}^d \twoheadrightarrow \mathcal{M}$.

Definition 2.2 (Tangent space)**.** The *tangent space* $T_p\mathcal{M}$ of $\mathcal{M}$ at $p \in \mathcal{M}$ is defined as

$$T_p\mathcal{M} = \{\dot\gamma(0) \mid \gamma \colon (-\varepsilon, \varepsilon) \to \mathcal{M} \text{ is a smooth curve with } \gamma(0) = p, \varepsilon > 0\}.$$

If $\psi \colon U \subset \mathbb{R}^d \to \mathcal{M}$ is a parametrization with $\psi(\xi) = p$ for some $\xi \in U$, then $V_i(\xi) := \partial_{\xi_i}\psi$ for $i = 1, \cdots, d$ is a basis of $T_p\mathcal{M}$, called the *canonical basis*.

This defines the topology of our manifold, and we are going to add geometric structure to it next.

Definition 2.3 (Riemannian manifold)**.** Let $\mathcal{M}$ be a d-dimensional differentiable manifold. A *Riemannian metric* on $\mathcal{M}$ is a family of bilinear, symmetric and positive-definite forms $g_p \colon T_p\mathcal{M} \times T_p\mathcal{M} \to \mathbb{R}$ smoothly varying with $p \in \mathcal{M}$, in the sense that for a parametrization $\psi \colon U \subset \mathbb{R}^d \to \mathcal{M}$ the map $\xi \mapsto g_{ij}(\xi) := g_{\psi(\xi)}(V_i(\xi), V_j(\xi))$ is a smooth function on U. A manifold equipped with a Riemannian metric is called a *Riemannian manifold*.

As $(g_{ij})_{ij}$ is an invertible matrix in $\mathbb{R}^{d\times d}$ we have an inverse $g^{-1} \in \mathbb{R}^{d\times \mathrm{d}}$, which we denote by $(g^{kl})_{kl}$, i.e. $g_{ij}g^{jk} = \delta_{ik}$.

Remark. Remember that in Section 1.1 we have studied embedded surfaces in $\mathbb{R}^3$ together with their first fundamental form. If we now compare this to our definitions above, we realize that those are in fact two-dimensional Riemannian manifolds with a metric induced by the embedding into $\mathbb{R}^3$. This also provides an intuition for the metric as ability to measure local lengths and angles on those higher-dimensional manifolds.

Paths and geodesics As for surfaces, one can define the length of a smooth path $y\colon [0,1] \to \mathcal{M}$ on a Riemannian manifold $(\mathcal{M}, g)$ in terms of the metric as

$$\mathcal{L}[(y(t))_{t\in[0,1]}] = \int_0^1 \sqrt{g_{y(t)}(\dot{y}(t), \dot{y}(t))}\mathrm{d}t\,. \tag{2.1}$$

Note that the path length is invariant to reparametrization. The path energy is defined as

$$\mathcal{E}[(y(t))_{t\in[0,1]}] = \int_0^1 g_{y(t)}(\dot{y}(t), \dot{y}(t))\mathrm{d}t\,, \tag{2.2}$$

which is not independent of the parametrization. By Cauchy-Schwarz inequality, one can directly see that

$$\mathcal{L}[(y(t))_{t\in[0,1]}] \leq \sqrt{\mathcal{E}[(y(t))_{t\in[0,1]}]} \tag{2.3}$$

and equality holds if and only if $g_{y(t)}(\dot{y}(t), \dot{y}(t)) = \text{const}$.

Definition 2.4 (Geodesic path)**.** For y_A, $y_B \in \mathcal{M}$ a minimizer of the path energy among all paths $y\colon [0,1] \to \mathcal{M}$ with $y(0) = y_A$ and $y(1) = y_B$ is denoted as *geodesic path* connecting y_A and y_B.

Remark. Readers familiar with Riemannian geometry might know a different definition of geodesics using the covariant derivative. The definition above is in fact equivalent to this definition, which one can see by investigating the Euler-Lagrange equations.

Rumpf and Wirth have shown in [RW15] that a minimizer of $\mathcal{E}$ exists and is unique under suitable assumptions. Moreover, based on this definition one can introduce the exponential map, which "shoots" geodesics in prescribed directions.

Definition 2.5 (Exponential map)**.** Let $y(t) = y(t, p, V)\colon I \to \mathcal{M}$, $0 \in I$, be the solution of $\frac{D}{dt}\dot{y}(t) = 0$ for initial data $y(0) = p$ and $\dot{y}(0) = V$. The (geometric) exponential map $\exp_p\colon T_p\mathcal{M} \to \mathcal{M}$ is defined as $\exp_p(V) = y(1, p, V)$.

One can show that the exponential map is locally bijection, i.e. there exists a $\delta > 0$, such that $\exp_p\colon B_\delta(0) \to \exp_p(B_\delta(0))$ is a bijection, which follows from the local uniqueness of geodesics. The image $U_p := \exp_p(B_\delta(0))$ is called a *normal neighborhood* of p

Definition 2.6 (Logarithm)**.** The inverse operator of the exponential map is called the (geometric) logarithm $\log_p\colon U_p \to T_p\mathcal{M}$, where U_p denotes the normal neighborhood of p.

Proofs for the well-definedness of all these objects (even in the infinite-dimensional case) can, for instance, be found in [RW15].

2.2 Time-discrete Geodesic Calculus

Now that we have introduced the various notions of Riemannian geometry, which are elements of what we call geodesic or geometric calculus, we want to be able to compute them in practice. Instead of using numerical methods for ordinary differential equations, we will present the elements of a variational time discretization of geodesic calculus, which was introduced by Rumpf and Wirth in a sequence of papers [Wir10, WBRS09, RW13, RW15]. The resulting time-discrete geodesic calculus is well suited for the application to shape space, as demonstrated for example in [HRWW12, HRS$^+$14, HRS$^+$16, BER15, MRSS15]. In this section, only the definitions and some basic intuition of the different discrete notions will be presented. For more details and especially for detailed results on convergence, we refer to [RW15].

Metric The continuous geodesic calculus presented in Section 2.1 was developed starting from a Riemannian metric, allowing to locally measure lengths and angles. In contrast, the discrete geodesic calculus we will see below is completely based on the notion of a (squared) Riemannian distance. The Riemannian distance is naturally induced by the metric as the minimal length of a curve connecting two points. Conversely, given the Riemannian distance dist, one can recover the Riemannian metric at some point $p \in \mathcal{M}$ by

$$g_p(V,W) = \frac{1}{2}\,\partial_2^2 \operatorname{dist}^2(p,p)(V,W)\,, \quad V,W \in T_p\mathcal{M}\,. \tag{2.4}$$

Coming up with a notion of distance is in many applications much easier than defining an inner product on tangent vectors. For example, in the context of physical shape spaces defining a Riemannian metric would be quite difficult whereas distances can be constructed by dissimilarity measures such as (1.25). Nevertheless, computing a precise evaluation of such a distance might still be challenging in practice, because it requires solving an optimization problem. Therefore, one introduces an approximation $\mathcal{W}$ of the squared Riemannian distance, which is easy to evaluate, and the discrete geodesic calculus will be based on this approximation. Precisely, one assumes there is a smooth functional $\mathcal{W}\colon \mathcal{M}\times\mathcal{M}\to\mathbb{R}$ such that for $y,\tilde{y}\in\mathcal{M}$

$$\mathcal{W}[y,\tilde{y}] = \operatorname{dist}^2(y,\tilde{y}) + O(\operatorname{dist}^3(y,\tilde{y}))\,. \tag{2.5}$$

Note that $\mathcal{W}$ is not required to fulfill the axioms of a metric, and thus is easier to define in practice. The next theorem shows, that the condition (2.5) implies consistency with the metric as in (2.4) and is even necessary given a certain smoothness. This means for g smooth enough, a valid approximation is, for instance, given by $\mathcal{W}[y,\tilde{y}] = \frac{1}{2}g_y(\tilde{y}-y,\tilde{y}-y)$.

Theorem 2.1 (Consistency conditions, [RW15])**.** *If $\mathcal{W}$ is twice Gâteaux-differentiable on $\mathcal{M}\times\mathcal{M}$ with bounded second Gâteaux derivative, then $\mathcal{W}[y,\tilde{y}] = \operatorname{dist}^2(y,\tilde{y}) + O(\operatorname{dist}^3(y,\tilde{y}))$ for $\tilde{y}$ close to $y\in\mathcal{M}$ implies*

$$\mathcal{W}[y,y] = 0\,, \quad \partial_2\mathcal{W}[y,y](V) = 0\,, \quad \partial_2^2\mathcal{W}[y,y](V,W) = 2g_y(V,W)$$

for any $V,W\in T_y\mathcal{M}$. Furthermore, $\partial_1\mathcal{W}[y,y](V)=0$ and

$$\partial_1^2\mathcal{W}[y,y](V,W) = -\partial_1\partial_2\mathcal{W}[y,y](V,W) = -\partial_2\partial_1\mathcal{W}[y,y](V,W) = \partial_2^2\mathcal{W}[y,y](V,W)\,.$$

If $\mathcal{W}$ is even three times Fréchet-differentiable, the implication becomes an equivalence.

Time-discrete geodesics As noted in the beginning, the central building block of the time-discretization is the notion of a discrete geodesic. To this end, we denote an ordered set of points $Y^K = (y_0,\dots,y_K)\subset\mathcal{M}$ as a time-discrete K-path. Usually, one thinks of such a discrete path as an uniform sampling of a smooth curve $\gamma\colon[0,1]\to\mathcal{M}$ by $y_k = \gamma(k\tau)$ for $k=0,\dots,K$ where $\tau = K^{-1}$ and $K\in\mathbb{N}$ is the sample size.

To relate the continuous path energy (2.2) to the Riemannian distance, one considers the following two estimates

$$\mathcal{L}[(y(t))_{t\in[0,1]}] \geq \sum_{k=1}^{K}\operatorname{dist}(y_{k-1},y_k)\,, \qquad \mathcal{E}[(y(t))_{t\in[0,1]}] \geq \frac{1}{\tau}\sum_{k=1}^{K}\operatorname{dist}^2(y_{k-1},y_k)\,, \tag{2.6}$$

where equality holds for geodesics paths due to the constant speed property. The first estimate simply follows from the definition of the length and the distance, whereas

the second is an application of the Cauchy-Schwarz inequality. This second estimate suggests that the sum on the right-hand side might be a reasonable approximation of $\mathcal{E}$, which motivates the following definition.

Definition 2.7 (Discrete length and energy, [RW15])**.** For a discrete K-path $Y^K = (y_0, \dots, y_K)$ with $y_k \in \mathcal{M}$ for $k = 0, \dots, K$, the *discrete length* L^K and the *discrete energy* E^K are given by

$$L^K[Y^K] = \sum_{k=1}^{K} \sqrt{\mathcal{W}[y_{k-1}, y_k]}\,, \qquad E^K[Y^K] = K \sum_{k=1}^{K} \mathcal{W}[y_{k-1}, y_k]\,. \tag{2.7}$$

Then a *discrete geodesic* (of order K) is defined as a minimizer of $E^K[Y^K]$ for fixed end points y_0, y_K.

Rumpf and Wirth showed in [RW15] that discrete geodesics exist and are locally unique, that the discrete path energy Γ-converges to the continuous path energy, and that minimizers converge as well. Furthermore, they showed that the points along the discrete geodesic are *equidistributed* in terms of the Riemannian distance.

They used this definition of a discrete geodesic as the foundation to develop a rich time-discrete geodesic calculus including discrete counterparts for the logarithm and exponential map. Therefore are time-discrete geodesics the first objects to study when one wishes to use this discrete calculus, which we will adhere to later on.

2.3 The Space of Discrete Shells

Our objective is to study the deformations of discrete shells and derive useful applications from this, for instance in animation. As mentioned at the beginning of the chapter, we do this by considering a so-called shape space, i.e. a space containing shapes as points. In our case, the shapes will be discrete surfaces, and we will the equip the shape space with the structure of a Riemannian manifold based on the physically sound deformation energies from Section 1.3. This structure was introduced by Heeren et al. in [HRWW12, HRS$^+$14, Hee16] and allows us to phrase different problems which are relevant in computer graphics via the geometric notions we have seen in Section 2.1. Applying the time-discretization from Section 2.2 allows us to compute numerical solutions to these problems.

As in Section 1.3, we will only consider discrete surfaces in dense correspondence (cf. Definition 1.25). Hence, we are given the connectivity $\mathcal{S}_h$ of a triangle mesh and consider different immersions of it into three-dimensional space.

Definition 2.8 (Shape space of discrete shells, [Hee16])**.** Given the connectivity $\mathcal{S}_h$ of a triangle mesh, the *shape space of discrete shells* $\mathcal{M}[\mathcal{S}_h]$ is given by all immersions of $\mathcal{S}_h$ into $\mathbb{R}^3$, i.e. all immersed discrete surfaces $\mathbf{S}$ in dense correspondence with $\mathcal{S}_h$, modulo rigid body motions.

Definition 2.9 (Riemannian shape space, [Hee16])**.** Given the connectivity $\mathcal{S}_h$ of a triangle mesh, the *Riemanninan shape space of discrete shells* $(\mathcal{M}[\mathcal{S}_h], g)$ is given by the shape space of discrete shells $\mathcal{M}[\mathcal{S}_h]$ together with the Riemannian metric g induced by the discrete deformation energy from Definition 1.30.

Remark. That the discrete deformation energy, in fact, induces a Riemannian metric on the shape space was proven in [HRS$^+$14]. At this point, we simply assume that the above notion is indeed well-defined.

In the following, we are assuming that we work with a fixed connectivity $\mathcal{S}_h$ and thus will drop the explicit dependence on it and simply write $\mathcal{M}$ for the shape space. As the immersion of a discrete surface is uniquely determined by the nodal positions, we can also identify $\mathcal{M}$ with $\mathbb{R}^{3|\mathcal{V}|}/SE(3)$.

Remark on implementation. In practice, of course one cannot directly work with $\mathbb{R}^{3|\mathcal{V}|}/SE(3)$ and hence solves the resulting problems in $\mathbb{R}^{3|\mathcal{V}|}$, while taking care of the rigid body motions in the algorithms. This need to deal with rigid body motions complicates numerical treatments. Especially working with the tangent space, e.g. in the context of a discrete logarithm or exponential map, is numerical cumbersome because of it.

Computing a time-discrete geodesic interpolating between y_0 and y_K, which is the central building block of the time-discrete geodesic calculus, amounts to solving the optimization problem

$$\underset{y_1,\dots,y_{K-1}}{\text{minimize}} \quad K \sum_{i=0}^{K-1} \mathcal{W}\left[y_i, y_{i+1}\right].$$

This is a highly nonlinear and high dimensional problem, and is typically solved using Newton's method. As the minimizer of this problem is only unique up to rigid body motions, one has to take care of this in the implementation because otherwise, the Hessian of the functional would be singular. This can be done for example by adding a small regularization term to the Hessian, fixing at least three nodal positions not on a line, or adding a suitable constraint. For more details on these problems, see the work of Heeren et al. in [HRWW12, HRS$^+$14, Hee16].

To tackle the high dimensionality of the optimization problems, often multiresolution schemes are employed. In the existing framework, this is achieved by coarsening all inputs simultaneously by applying an iterative edge collapse approach based on minimizing the quadric error metric [GH05] computed groupwise [MG03] to preserve the dense correspondence. The coarse solution will then be prolongated using an approach from [BSPG06]. For more details on this procedure, we refer to [Hee16].

Chapter 3

Lengths and Angles

In this chapter, our goal is to develop efficient tools for the computation and analysis of deformations of discrete shells. As we assume dense correspondence (cf. Definition 1.25), we want to derive a parameterization of the immersions of a discrete surface allowing for efficient computations. One important point is that it has to be rigid body motions invariant, i.e. if two immersed discrete surfaces only differ by a rigid transformation they should have the same "coordinates". In Section 1.1.2, we have seen that for smooth surfaces, such a parameterization is given by first and second fundamental forms fulfilling the Gauß-Codazzi equations. Furthermore, in Section 1.3 these fundamental forms turned out to be the right properties for physically-sound deformation energies. Altogether, this motivates to study our discrete notions of first and second fundamental form as coordinates for immersed discrete surfaces.

In [WDAH10], Winkler et al. proposed to use edge lengths and dihedral angles as primary degrees of freedom to interpolate between different shapes and employed a hierarchical shape matching technique to construct the nodal positions that best match the interpolated points. Previously, lengths and angles were used as properties of the mesh to compute deformations in the Discrete Shells model [GHDS03], which Fröhlich et al. realized and connected in [FB11]. They developed a computationally efficient scheme allowing interpolation and shape editing based on formulating the reconstruction of nodal positions as a nonlinear least-squares problem, which is solved using the Gauß-Newton method and a multi-resolution scheme. It was also used in [HRS$^+$16, Hee16] to compute Riemannian splines in the space of discrete shells. In this thesis, their approach is present in form of the quadratic deformation energy and the energy-based reconstruction.

Not all combinations of edge lengths and dihedral angles are admissible, meaning that we can construct nodal positions admitting them. In [WLT12], Wang et al. derived conditions, akin to the Gauß-Codazzi equations, for them to be. Their work is based on [LSLCO05], which proved similar conditions for a different vertex-based mesh representation and [CDS10] introducing certain connections between tangent spaces on discrete surfaces. The conditions on length and angles will play an important role in this thesis and Section 3.2 is entirely devoted to explaining the results from [WLT12]. An unexpected application of the transition matrices that were investigated in this context is given by Schonsheck et al. in [SDL18] for generalizing Convolutional Neural Networks to discrete surfaces.

Comparing the number of lengths and dihedral angles to the number of nodal degrees of freedom quickly reveals that the former overparameterize immersions. Chern et al. used this in [CKPS18] to create discrete immersions for given edge lengths, while Amenta et al. investigated only the dihedral angles as a scale-invariant description of

discrete surfaces in [AR18]. A similar approach was used by Baek et al. in [BLL15] to create almost isometric interpolations by varying the dihedral angles and keeping the edge lengths as fixed as possible. Another way to get a scale-invariant description is to use interior angles instead of edge lengths, which is pursued in [LWLL14], however without investigating the admissibility of configurations.

There are certainly numerous other local descriptions of meshes [BS08] and other reconstruction problems for surfaces [BEKB15], but as our focus is lengths and angles, we will keep it at those two exemplary references.

In this chapter, we will first argue for the use of edge lengths and dihedral angles (i.e. the angle between normals of neighboring triangles) as equivalent notions to the discrete fundamental forms. Then, we are going to present in detail a corresponding fundamental theorem for discrete surfaces from [WLT12] introducing conditions reminiscent of the Gauß-Codazzi equations. Furthermore, we will study deformation energies on length and angles. Based on these two, different approaches for the reconstruction of an immersion from given lengths and angles will be discussed.

3.1 Discrete Fundamental Forms

Let us recall the discrete first fundamental form from Definition 1.17, which was elementwise constant and given by

$$G|_f = \begin{pmatrix} \|E_2(f)\|^2 & -\langle E_1(f), E_2(f)\rangle \\ -\langle E_1(f), E_2(f)\rangle & \|E_1(f)\|^2 \end{pmatrix} \tag{3.1}$$

for each $f \in \mathcal{F}$. We directly see that the entries of the diagonal are simply edge lengths of the triangle $T(f)$. Additionally, the off-diagonal entries are given by Euclidean scalar products of edge vectors and from linear algebra we recall that for two vectors $v, w \in \mathbb{R}^3$ we have $\langle v, w\rangle = \|v\|\|w\|\cos(\gamma)$, where γ is the angle between v and w. Hence, in our case, we need to compute the interior angles of a triangle from its edge lengths which is possible due to a classic result from trigonometry.

Lemma 3.1 (Law of cosines)**.** *Assume we have a triangle whose sides have lengths a, b, and c. Then the angle opposite to c is given by*

$$\gamma_c = \arccos\left(\frac{a^2 + b^2 - c^2}{2ab}\right). \tag{3.2}$$

We, therefore, see we can describe our discrete first fundamental form solely in terms of edge lengths. In Section 1.3, we also needed the determinant of $G|_f$, which is given by the triangle's squared area. Let us note, that this can also be computed directly from edge lengths by another classic result from trigonometry, which will come in handy later on.

Lemma 3.2 (Heron's formula)**.** *Assume we have a triangle whose sides have lengths a, b, and c. Then its area is given by*

$$A = \sqrt{s(s-a)(s-b)(s-c)}, \tag{3.3}$$

where s is its semiperimeter, i.e. $s = \frac{a+b+c}{2}$.

Remark. Essentially, we used above that the geometry of a Euclidean triangle is completely determined by its edge lengths.

Next, we recall the triangle-averaged discrete second fundamental form, which is elementwise constant and given in Definition 1.20 by

$$H|_f = -4a_f \sum_{i=0}^{2} \frac{\cos\frac{\theta_i+\pi}{2}}{\|E_i\|} M_i \tag{3.4}$$

for each $f \in \mathcal{F}$, where (M_0, M_1, M_2) is the basis of symmetric 2×2-matrices from (1.18). Again, we immediately see that we need the edge lengths to define the matrix, as well as the triangle's area, which can be deduced from them. Additionally, we need the dihedral angles (i.e. the angle between adjacent face normals, cf. Definition 1.19) of the immersed discrete surface.

Altogether, we can fully describe our discrete fundamental forms by lengths and angles, and we will treat them as such in the following.

Definition 3.1 (Lengths and Angles)**.** For an immersion $X \in \mathbb{R}^{3|\mathcal{V}|}$ of a discrete surface $\mathcal{S}_h$. we denote by $(l_e(X))_{e\in\mathcal{E}}$ its edge lengths and by $(\theta_e(X))_{e\in\mathcal{E}}$ its dihedral angles. We combine them in a single vector and thus obtain a map

$$\begin{aligned} Z\colon \mathbb{R}^{3|\mathcal{V}|} &\to \mathbb{R}^{|\mathcal{E}|} \times \mathbb{R}^{|\mathcal{E}|} \\ X &\mapsto ((l_e(X))_{e\in\mathcal{E}}, (\theta_e(X))_{e\in\mathcal{E}}). \end{aligned} \tag{3.5}$$

In the following, we will denote by $z = (l_e, \theta_e)_{e\in\mathcal{E}}$ a set of lengths and angles in $\mathbb{R}^{2|\mathcal{E}|}$, regardless of the fact whether it belongs to an immersion of our discrete surface or not, and by $Z[X]$ the lengths and angles of a specific immersion. Note, that the map Z is well-defined only on vertex positions in $\mathbb{R}^{3|\mathcal{V}|}$ which do not lead to degenerated triangles, i.e. belong to an immersion as in Definition 1.16. Otherwise, the dihedral angles would not be well-defined. Then, restricted to this open domain, Z is a smooth map.

Remark on implementation. When implementing the law of cosines and Heron's formula one has to pay attention to numerical stability. The formulas given in the statements above are in fact numerically unstable, as they would lead to problems with elimination for slim triangles.

Furthermore, we can interpret them as maps from edge lengths to interior angles $\mathbb{R}^{|\mathcal{E}|} \to \mathbb{R}^{3|\mathcal{F}|}$ and to triangle areas $\mathbb{R}^{|\mathcal{E}|} \to \mathbb{R}^{|\mathcal{F}|}$ respectively. These maps are smooth on their domain of definition, i.e. away from edge lengths violating the triangle inequality.

We present numerical stable formulations of the two formulas from [Kah14] as well as derivatives in Section A.1.

3.2 Admissible Lengths and Angles

In Section 1.1.2, we have seen that not all bilinear forms are fundamental forms of an immersed surfaces, but rather they have to fulfill conditions called the Gauß-Codazzi equations. The same holds for elements in $\mathbb{R}^{2|\mathcal{E}|}$ that are actually given as edge lengths and dihedral angles of an immersion of our discrete surface.

Definition 3.2 (Admissible lengths and angles)**.** Let $\mathcal{S}_h$ be a discrete surface. Then we define the *admissible lengths and angles* $\mathcal{Z}$ as the subset of $\mathbb{R}^{2|\mathcal{E}|}$ consisting of those lengths and angles for which an immersion admitting them exists, i.e.

$$\mathcal{Z} := \{z \in \mathbb{R}^{2|\mathcal{E}|} \mid \exists X \in \mathbb{R}^{3|\mathcal{V}|}\colon Z[X] = z\}.$$

While this description as the image of Z yields a valid definition of $\mathcal{Z}$ it does not offer much of an insight into the structure of $\mathcal{Z}$. Moreover, using it as a constraint in optimization problems to ensure the admissibility of solutions is cumbersome because it requires keeping track of the corresponding nodal positions. Our goal is now to develop a more thorough understanding by describing $\mathcal{Z}$ through a set of conditions on lengths and angles to be admissible, reminiscent of the Gauß-Codazzi equations. The proof that these conditions fully describe $\mathcal{Z}$ will lead to the fundamental theorem for discrete surfaces introduced by [WLT12].

This understanding will also be useful for applications later on. For example, through the conditions we can consider $\mathcal{Z}$ as a submanifold of $\mathbb{R}^{2|\mathcal{E}|}$ and thus compute its tangent space $T_z\mathcal{Z}$. This will be exploited this later in the context of Principal Geodesic Analysis (PGA). Besides, we can use the conditions more conveniently as constraints for minimization problems in lengths and angles ensuring the admissibility of the results, which will lead to more natural results and simplify reconstruction of an immersion. Lastly, proving the sufficiency of the conditions will provide a way to reconstruct the immersion, i.e. nodal positions, from admissible lengths and angles, which will be discussed in more detail in Section 3.4.2.

This section is largely presenting the work of Wang et al. in [WLT12]. We provide a more comprehensive derivation of the transition rotations and discrete integrability conditions below, while the definitions themselves are directly adopted from [WLT12]. Afterwards, the central theorems and their proofs given in [WLT12] will be presented and at the same time, we are going to provide a rigorous definition of accumulated translations not present there. Furthermore, we will report our experience with using the different conditions in numerical implementations, especially as constraints for optimization problems.

3.2.1 Triangle Inequality

The first condition is quite simple. We have to ensure that the lengths actually describe valid Euclidean triangles. This is guaranteed by the triangle inequality, which reads as condition on all three edge lengths a, b, c of a triangle

$$a + b + c - 2\max(a, b, c) > 0.$$

This can be used directly as a constraint on lengths and angles, formalized in the following definition.

Definition 3.3 (Triangle inequality constraints)**.** For a face $f \in \mathcal{F}$ with edges e_i, e_j, and e_k, we define the *triangle inequality map* on lengths and angles z to be

$$\mathcal{T}_f(z) = l_i + l_j + l_k - 2\max(l_i, l_j, l_k). \tag{3.6}$$

We say that lengths and angles z fulfill the *triangle inequality constraints* if

$$\mathcal{T}_f(z) > 0 \text{ for all } f \in \mathcal{F}. \tag{T}$$

Remark on implementation. In practice, using the triangle inequality map as defined above might not be feasible because it is discontinuous at points where two edges of a triangle have the same length. When using it in constraints for optimization problems we need its derivatives and thus might get into problems. Therefore, it is also relevant for practical implementations to consider a vector-

valued triangle inequality map

$$\widetilde{\mathcal{T}}_f(z) = \begin{pmatrix} l_i + l_j - l_k \\ l_j + l_k - l_i \\ l_k + l_i - l_j \end{pmatrix}, \tag{3.7}$$

such that (T) has to hold component-wise. Using these separate inequalities has the advantage that they are all differentiable and thus on their own far easier to treat when optimizing. On the other hand, this increases the number of constraints by going from one per triangle to three per triangle. This, of course, leads to different numerical challenges when solving the constraint optimization problem.

Furthermore, enforcing the triangle inequalities explicitly is not always necessary. We will introduce another condition on lengths and angles next, which is only well-defined when the triangle inequalities are fulfilled and will be formally defined and implemented such that it is violated when one of the triangle inequalities is violated. Hence, requiring this second condition will also entail the fulfillment of the triangle inequalities. Also, the nonlinear energy we will introduce in Section 3.3.2 converges to infinity for edge lengths getting close to violating the triangle inequality. Hence, when using it in the objective of our optimization problem this leads to an inbuilt penalty. Nevertheless, in some cases, it might be helpful to explicitly require the triangle inequalities to somewhat guide the optimization method.

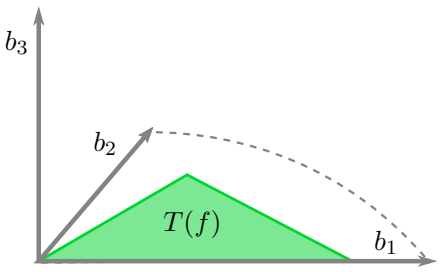
3.2.2 Discrete Integrability Conditions

Next, we will derive the conditions reminiscent of the Gauß-Codazzi equations. They ensure that we can integrate the local change of geometry induced by the lengths and angles to get the immersed discrete surface. By this, we mean that constructing the immersion starting from a single triangle is well-defined in the sense that it is invariant with respect to the order of construction. Therefore, they are called *discrete integrability conditions*. To formulate them, we have to introduce the notion of *frames* on triangles and *transition rotations*, which describe their connection.

Frames In the following, we will work with face normals and thus localize tangent spaces on the triangles of an immersed discrete surface. For a triangle $f \in \mathcal{F}$, where again by abuse of notation we identify it with its immersion, we denote by $N_f \in S^2$ its normal and by $T_f\mathbf{S}$ its tangent plane, i.e. the plane spanned by its edge vectors.

Definition 3.4 (Discrete frame)**.** Given a discrete surface $\mathcal{S}_h$ with immersion $X \in \mathbb{R}^{3|\mathcal{V}|}$, we define a discrete frame on a face f to be an orthonormal basis $F_f = (b_1, b_2, b_3)$ with $b_3 = N_f$, hence (b_1, b_2) forms an orthonormal basis of the tangent plane $T_f\mathbf{S}$.

The notion of discrete frames provides a formal way to talk about the triangles' orientation of the immersed surface as orthonormal bases. This will allow us in the following to describe the local change of geometry using changes of bases and hence matrices.



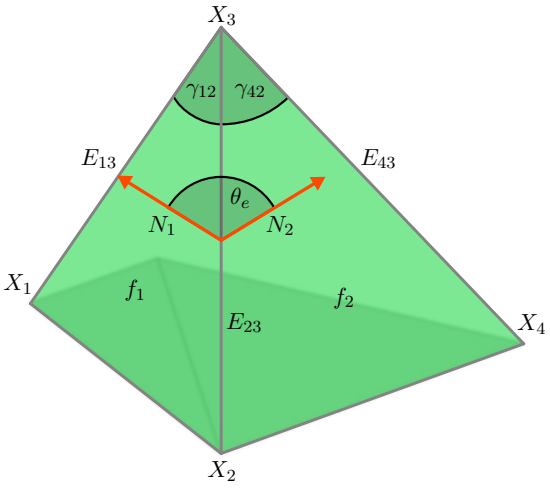
On a single immersed triangle, there naturally exist uncountably many discrete frames through the different choices for the orthonormal basis (b_1, b_2) of the tangent plane $T_f\mathbf{S}$. For all of them, the third vector of the basis is fixed, and hence the first two can only be

changed by a rotation around it, precisely in the plane of the triangle. Thus, the choice of a discrete frame on a triangle can be completely characterized by the angle of its first basis vector b_1 to one of the edges. This leads us the notion of a *standard discrete frame*, which is also depicted in the inset sketch.

Definition 3.5 (Standard discrete frame)**.** Considering a face f with edges e_1, e_2, and e_3, where we assume the local indices to be ordered consistently, we define the standard discrete frame to be $F_f = \left(\frac{E_1}{\|E_1\|}, \frac{E_1 \times N_f}{\|E_1 \times N_f\|}, N_f\right)$.

Remark. If we choose discrete frames for all faces this leads to a map $F\colon \mathcal{F} \to SO(3)$, which we call a *discrete frame field*.

Transition rotations Now, we introduce a relation between the frames of adjacent triangles on which the integrability conditions will be built. It describes the change of the frames induced by the dihedral angles through change of basis matrices, but can also be explained by the reconstruction of a triangle from an adjacent triangle. We first consider the second explanation.



Let us look at two faces f_1 and f_2 with vertices v_1, v_2, v_3 and v_2, v_3, v_4 respectively, i.e. $e = (v_2v_3)$ is the common edge of the two. As before, we denote by $(X_i)_{i\in\{1,2,3,4\}}$ the coordinates of immersed vertices, by $E_{ij} = X_i - X_j$ the corresponding edge vectors, and by N_1 resp. N_2 the normals of the faces. Our goal is now to reconstruct E_{43} and N_2 from E_{13} and N_1. For notational simplicity, we assume that E_{43} and E_{13} have both unit length, hence we only have to describe a rotation that maps them onto each other. One quickly observes in the inset sketch, that the rotation $\tilde{R} := R_{N_2}(\gamma_{42})R_E(\theta_e)R_{N_1}(\gamma_{21})$ maps E_{13} to E_{43} and N_1 to N_2, where γ_{ij} is the angle opposite of E_{ij}.

We can rewrite this in terms of frames by defining the local frames $\tilde{F}_1 = \left(E_{13}, \frac{E_{13}\times N_1}{\|E_{13}\times N_1\|}, N_1\right)$ and $\tilde{F}_2 = \left(E_{43}, \frac{E_{43}\times N_2}{\|E_{43}\times N_2\|}, N_2\right)$. Then one sees that the above rotation $\tilde{R}$ yields the change of basis in the sense that $\tilde{F}_2 = \tilde{R}\tilde{F}_1$. If we consider arbitrary frames F_1 and F_2 one can compute $F_2F_1^{-1} = R_{N_2}(\gamma_{e,F_2})R_E(\theta_e)R_{N_1}(\gamma_{F_1,e})$, where γ_{e,F_2} is the angle between the common edge E and the first vector of F_2, and $\gamma_{F_1,e}$ is the angle between the first vector of F_1 and E. This leads us to the following definition.

Definition 3.6 (Extrinsic transition rotation)**.** Given an immersed discrete surface $\mathbf{S}$ with frames F_f for all faces $f \in \mathcal{F}$, we define for an edge e with adjacent faces f_i and f_j the *extrinsic transition rotation* by

$$R^E_{ij} = R_{N_j}(\gamma_{e,F_j})R_E(\theta_e)R_{N_i}(\gamma_{F_i,e}), \tag{3.8}$$

where γ_{e,F_j} and $\gamma_{F_i,e}$ denote the angles between the immersed edge E and the first vector of the frames F_{f_i} resp. F_{f_j}, N_i and N_j denote the normals of f_i resp. f_j.

Hence, the extrinsic transition rotations describe the change of basis between frames of adjacent triangles. The problem is, that it includes rotations around the normals and edges, hence cannot be formulated directly in lengths and angles. Therefore, we derive an intrinsic formulation, where we try to find a matrix R such that $F_2 = F_1 R$. It encodes how the coefficients have to change if we want to express a vector given in F_2 by coefficients in F_1. Let us take a closer look at this. First, we introduce intermediate frames $\tilde{F}_1 = \left(E, \frac{E \times N_1}{\|E \times N_1\|}, N_1\right)$ and $\tilde{F}_2 = \left(E, \frac{E \times N_2}{\|E \times N_2\|}, N_2\right)$. Then

$$\begin{aligned} R &= F_1^{-1} F_2 \\ &= F_1^{-1} \tilde{F}_1 \tilde{F}_1^{-1} \tilde{F}_2 \tilde{F}_2^{-1} F_2. \end{aligned}$$

Now, as both F_1 and $\tilde{F}_1$ are orthogonal, and $F_1^{-1}\tilde{F}_1 z = z$ we see that $F_1^{-1}\tilde{F}_1$ is a rotation around the z-axis. Furthermore, by $F_1 \tilde{F}_1^{-1} E = F_{11}$ and the fact that $\operatorname{tr} F_1 \tilde{F}_1^{-1} = \operatorname{tr} F_1^{-1} \tilde{F}_1$, we deduce $F_1^{-1}\tilde{F}_1 = R_z(\gamma_{e,F_1})$. In the same fashion, we obtain $\tilde{F}_1^{-1}\tilde{F}_2 = R_x(-\theta_e)$ and $\tilde{F}_2^{-1} F_2 = R_z(\gamma_{F_2,e})$ and therefore

$$R = R_z(\gamma_{e,F_1}) R_x(-\theta_e) R_z(\gamma_{F_2,e}),$$

which justifies the following definition.

Definition 3.7 (Intrinsic transition rotation)**.** Given an immersed discrete surface $\mathbf{S}$ with frames F_f for all faces $f \in \mathcal{F}$, we define for an edge e with adjacent faces f_i and f_j the *intrinsic transition rotation* by

$$R_{ij}^I = R_z(\gamma_{e,F_i}) R_x(-\theta_e) R_z(\gamma_{F_j,e}), \tag{3.9}$$

where γ_{e,F_j} and $\gamma_{F_i,e}$ denote the angles between the immersed edge E and the first vector of the frames F_{f_i} resp. F_{f_j}.

If the frames F_f are standard discrete frames, the angles γ are given by the inner angles of the triangles. Hence, they can be computed from edge lengths using the law of cosines and thus the transition rotations R_{ij}^I can be completely determined from the length and angles of the immersed surface. This allows us to define transition rotations for lengths and angles without knowing the immersion, as long as they fulfill the triangle inequality, formalized as follows.

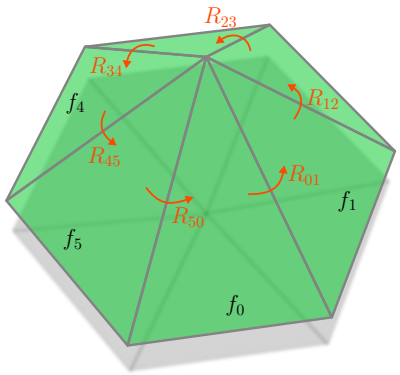
Definition 3.8 (Induced transition rotation)**.** Given a discrete surface $\mathcal{S}_h$ and an element $z \in \mathbb{R}^{2|\mathcal{E}|}$ such that $\mathcal{T}_f(z) > 0$ holds for all $f \in \mathcal{F}$, we define for an edge e with adjacent faces f_i and f_j the *induced transition rotation* by

$$R_{ij}(z) = R_z(\gamma_{e,i}) R_x(-\theta_e) R_z(\gamma_{j,e}), \tag{3.10}$$

where $\gamma_{e,j}$ and $\gamma_{i,e}$ denote the interior angle between the common edge e and the first edge of f_i resp. f_j, determined from the edge lengths of z.

Conditions Starting from this description of how the frames of adjacent triangles are related to each other, we can derive a necessary condition for lengths and angles to be admissible. Let us first consider the extrinsic point of view. Thereby, we consider an interior vertex v which is the center of a n-loop of faces $f_0, \dots, f_{n-1}$ with frames $F_0, \dots, F_{n-1}$. As above, to foster intuition we can think of them as given by the normal and one of the edges. We have seen that we can reconstruct them from each other using the extrinsic transition rotations introduced above by $F_j = R_{ij}^E F_i$.

As this amounts to reconstructing a triangle from its neighbor, if we apply this sequentially along the loop, we will get back to the original triangle, formally $F_0 = \prod_{i=0}^{n-1} R^E_{n-i-1,(n-i) \bmod n} F_0$ and hence $\prod_{i=0}^{n-1} R^E_{n-i-1,(n-i) \bmod n} = \mathrm{Id}$. Equivalently, we can formulate this using intrinsic rotation matrices as $F_0 = F_0 \prod_{i=0}^{n-1} R^I_{i,(i+1) \bmod n}$ and $\prod_{i=0}^{n-1} R^I_{i,(i+1) \bmod n} = \mathrm{Id}$. Now, assume that we have an element $z \in \mathbb{R}^{2|\mathcal{E}|}$ and consider its induced transition rotations $R_{ij}(z)$. Then for z to be admissible those transitions have to fulfill the equality as well.



Definition 3.9 (Discrete integrability conditions)**.** Given a discrete surface $\mathcal{S}_h$, for each interior vertex v we define the *discrete integrability map* on lengths and angles $z \in \mathbb{R}^{2|\mathcal{E}|}$ as

$$\mathcal{I}_v(z) = \begin{cases} \prod_{i=0}^{n-1} R_{i,(i+1) \bmod n}(z) & \text{if } \mathcal{T}_f(z) > 0 \text{ for all } f \in \mathcal{N}_v \\ \mathrm{diag}(\infty, \infty, \infty) & \text{else} \end{cases} \tag{3.11}$$

for the n-loop of faces $\mathcal{N}_v = \{f_0, \dots, f_{n-1}\}$ around v. We say that lengths and angles z satisfy the *discrete integrability conditions* if

$$\mathcal{I}_v(z) = \mathrm{Id} \quad \text{for each } v \in \mathcal{V}. \tag{I}$$

Remark. Note that for an immersed discrete surface the conditions are independent of the choice of local frames. Another choice for the discrete frame on the triangle f_j would only differ by a rotation around the normal and thus would affect both $R_z(\gamma_{e,F_j})$ and $R_z(\gamma_{F_j,e'})$ canceling the effect in the product.

Furthermore, we have extended the definition of the discrete integrability map to edge lengths not fulfilling the triangle inequality such that (I) implies (T), as previously mentioned.

Remark on implementation. The first thing to remark with respect to implementing the notions described in this section is that one has to pay attention to the signs of the angles, for instance in (3.10). The sign of the dihedral angle depends on a notion of left and right of an edge. Moreover, the angle between two edge vectors depends on their relative orientation, which can be misleading when drawing a triangle. Hence, the signs of all the angles depend on the stored orientation of the edges in the implementation.

Now, using condition (I) as a constraint in an optimization problem would lead to nine scalar constraints per vertex. For discrete surfaces with a high resolution, and thus numerous vertices, this would lead to a massive amount of constraints, rendering the numerical treatment of such problems expensive. We can exploit the structure of the integrability map to reduce the number of constraints. Note that all matrices multiplied in (3.11) are rotations and thus their product $\mathcal{I}_v(z)$ is also a rotation matrix. Hence, we have $\mathcal{I}_v(z) \in SO(3)$, and $SO(3)$ can be parameterized locally using Euler angles. This will give us three scalar conditions per vertex and thus reduce the number of constraints significantly. We explicitly choose Euler

angles in $x-y-z$ orientation, which are given for a matrix $Q \in SO(3)$ as

$$(\arcsin Q_{31},\, \operatorname{atan2}(Q_{32}, Q_{33}),\, \operatorname{atan2}(Q_{21}, Q_{11})), \tag{3.12}$$

where $\operatorname{atan2}(y, x)$ is the angle in the Euclidean plane between the positive x-axis and the ray to the point (x, y). Note, that this is only a local parameterization and it is important to choose the right orientation of Euler angles. Otherwise would the resulting map not be differentiable at admissible lengths and angles. Requiring that $\mathcal{I}_v(z) = \mathrm{Id}$ is the same as saying that all of the above Euler angles are zero. To this end, we introduce the modified integrability map

$$\widetilde{\mathcal{I}}_v(z) = (\arcsin \mathcal{I}_v(z)_{31},\, \operatorname{atan2}(\mathcal{I}_v(z)_{32}, \mathcal{I}_v(z)_{33}),\, \operatorname{atan2}(\mathcal{I}_v(z)_{21}, \mathcal{I}_v(z)_{11})), \tag{3.13}$$

which leads to the modified integrability condition

$$\widetilde{\mathcal{I}}_v(z) = 0 \text{ for each } v \in \mathcal{V}. \tag{$\tilde{\text{I}}$}$$

This replaces (I) in implementations.

Finally, we have remarked above that the integrability condition does not depend on the choice of frames on the triangles. Especially, we can also make different choices for validating the condition on different vertices. This allows us to depart in the implementation from the fixed choice of standard frames used in Definition 3.8 to check (I) more efficiently by reducing the number of needed rotations. For this approach, let us consider the vertex $v \in \mathcal{V}$ with the n-loop of faces $f_0, \ldots, f_{n-1}$ around it. Then there are also n edges connected to v and we create a one-to-one correspondence between faces and edges such that the edge belongs to the face. This can be thought of as "always picking the left/right edge". Based on this, we choose the frames $F_0, \ldots, F_{n-1}$ such that the chosen edge is always the first basis vector. Then the corresponding transition rotations simplify to $R^I_{ij} = R_x(\theta_e)R_z(\gamma_{j,v})$, where $\gamma_{j,v}$ is the interior angle at v in f_j. By this, the product of transition rotations $\mathcal{I}_v(z)$ becomes an alternating product of rotations about the interior and dihedral angles. The reduced number of rotations leads to shorter computational times and also simplifies the computation of derivatives of $\mathcal{I}_v$. These are needed if we use (I) as a constraint for minimization problems and are given in Section A.2.

3.2.3 Fundamental Theorem

So far, we have introduced necessary conditions for lengths and angles of a simply connected discrete surface to be admissible, meaning that we have shown

$$\mathcal{Z} \subseteq \left\{ z \in \mathbb{R}^{2|\mathcal{E}|} \mid \mathcal{T}_f(z) > 0 \text{ for all } f \in \mathcal{F},\ \mathcal{I}_v(z) = \mathrm{Id} \text{ for all } v \in \mathcal{V} \right\}. \tag{3.14}$$

In this section, we will prove first, following [WLT12], that this is actually an equality. Furthermore, we will generalize the result, also following [WLT12], to general discrete surfaces, which will constitute a *discrete fundamental theorem of surfaces*.

Theorem 3.1 (Local constructability, [WLT12])**.** *Given a discrete surface $\mathcal{S}_h$ with disk-like topology, if $z \in \mathbb{R}^{2|\mathcal{E}|}$ satisfies the discrete integrability conditions and the triangle inequalities, then there exists an immersion $X \in \mathbb{R}^{3|\mathcal{V}|}$ of the surface in three-dimensional Euclidean space admitting z as edge lengths and dihedral angles.*

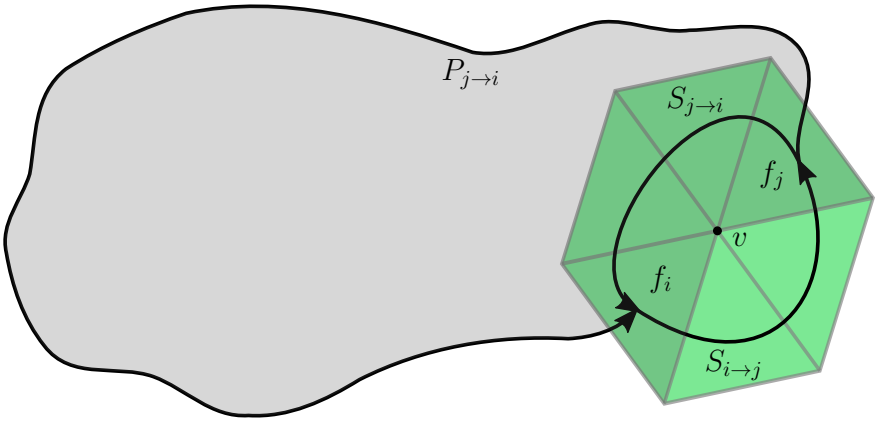


Figure 3.1: Domain (gray overlay) and dual paths (black) resulting from removing a single vertex v (black dot).

Remark. By disk-like topology, we mean that $\mathcal{S}_h$ as topological space is homeomorphic to the closed disk. Most importantly, this implies that it is simply connected.

Proof. We start with constructing a consistent frame field F on the surface. Based on our explanations above, we can construct a frame on the face f_j from the neighboring face f_i by

$$F_j = F_i R_{ij}(z). \tag{3.15}$$

Beginning with a seed frame F_0 and iterating the above gives frames for all faces. We have to show that this is well-defined, meaning that if two faces f_i and f_j are connected by two dual paths, then constructing F_j by repeatedly applying (3.15) along each of the paths produces the same result. The two paths together form a dual cycle, which without loss of generality we can assume to not be self-intersecting because otherwise, the result would follow from proving it for each sub-cycle. We proceed by induction on the number of vertices k enclosed by the dual loop. If $k = 1$, the dual cycle encloses a single vertex, meaning it is a loop of triangles as in Definition 3.9 and thus the accumulated rotation will be the identity. This implies that the construction of frames agrees for both paths.

For larger k, due to the disk-like topology of our surface, we can always find an enclosed vertex v, such that removing it leaves the remaining region simply connected (cf. Figure 3.1). We denote the common segment of the dual loop around v and the original loop by $S_{i\to j}$ and the remainder of the vertex loop by $S_{j\to i}$. The other part of the original loop is denoted by $P_{j\to i}$, see also Figure 3.1. By R_{Γ^*} we denote the multiplication of transition rotations along a dual curve Γ^*, hence

$$R_{\Gamma^*} := \prod_{(k,l)\in\Gamma^*} R_{kl}(z).$$

Then we have

$$R_{P_{j\to i}} R_{S_{i\to j}} = \left(R_{P_{j\to i}} R_{S_{j\to i}}^{-1}\right)\left(R_{S_{j\to i}} R_{S_{i\to j}}\right) = \mathrm{Id}, \tag{3.16}$$

because the first part is the identity by the induction hypothesis and the second by the discrete integrability condition for v. Hence, we have shown that we can construct a consistent frame field from a seed frame and z.

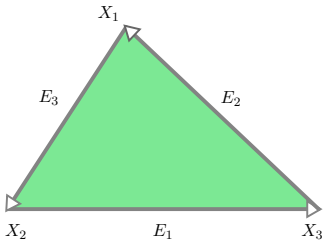
Next, we have to construct consistent vertex positions, for which we again start with the position of a seed vertex v_0. First, we will show that F_j being consistent with

R_{ij}, i.e. (3.15) holds, implies that the edge vector E_e of the common edge e is the same if we construct it from either face. The edge lengths of z induce local coordinates

$$a_e^i = \begin{pmatrix} \cos\gamma_{e,F_i} \\ \sin\gamma_{e,F_i} \\ 0 \end{pmatrix}, \quad a_e^j = \begin{pmatrix} \cos\gamma_{e,F_j} \\ \sin\gamma_{e,F_j} \\ 0 \end{pmatrix}, \tag{3.17}$$

for the edge vector in the respective frames, i.e. we want $F_i a_e^i = E_e = F_j a_e^j$, where the γs are defined as in Definition 3.8. This is consistent with R_{ij} in the sense that $R_{ij} a_e^j = a_e^i$ and therefore $F_j a_e^j = F_i R_{ij} a_e^j = F_i a_e^i$. Hence, we get consistent edge vectors from the frames and local coordinates induced by z.

Furthermore, for a single triangle with edges e_1, e_2, e_3 it holds that $a_{e_1} + a_{e_2} + a_{e_3} = 0$, because if we add the three edge vectors of a triangle we get back to the starting vertex and the edge lengths fulfill the triangle inequality. See also the inset sketch. By induction, using the simply connectedness as above, this follows for each primal edge cycle and thus can the nodal positions be constructed consistently without any harmful translation. □



Global theorem The last element of the proof above was to show that we do not acquire any inconsistent translations when we integrate the change of geometry induced by lengths and angles. This argument appeared to be rather unnecessary considering that it simply followed from the fact that all edge lengths induce Euclidean triangles. However, in the general case of non-simply connected surfaces, we need to introduce an explicit condition for this, whose necessity we will also illustrate in an example later on. To this end, we will formalize the notion of translation accumulated when integrating along a path below.

Definition 3.10 (Accumulated translation)**.** Let $\mathcal{S}_h$ be a discrete surface, $z \in \mathbb{R}^{2|\mathcal{E}|}$ fulfilling (I), and $\Gamma = (e_1, \dots, e_J)$ be a path of primal edges. Then, for each edge e_i we choose a face f_i such that $e_i \in f_i$ and denote for each $i > 1$ by $R_{e_i \to e_{i-1}}$ the accumulated induced transition rotation along the shorter of the possible dual paths from f_i to f_{i-1} around their common vertex. We define the *accumulated translation* along the path of primal edges as

$$\sum_{j=1}^{K} R_{e_2 \to e_1} \cdots R_{e_j \to e_{j-1}} a_{e_j}, \tag{3.18}$$

where a_{e_j} are the induced local coordinates of E_j in f_j, defined as in the proof of Theorem 3.1.

Remark. The above definition is only well-posed if its accumulated transition rotation is equal for both paths around a common interior vertex, which is the case as z is required to fulfill the integrability condition (I). The choice of face for the interior edges in the definition does not matter as the corresponding edge vectors will agree by the same argument, see the proof of Theorem 3.1.

Furthermore, the definition is made independently of an initial frame, as this only affects the direction of the resulting translation and for our purposes only its magnitude is relevant.

Lastly, let us note once again that computing R_{ij} and also a_{e_j} requires fixing a specific choice of orientation for frames in the plane of the triangles, for instance always picking standard frames. However, this does not affect the accumulated translation as long as everything is chosen consistently.

Equipped with this definition, we can now present the central theorem and its proof from [WLT12].

Theorem 3.2 (Discrete fundamental theorem of surfaces, [WLT12])**.** *Given a discrete surface $\mathcal{S}_h$ with genus g, if for a $z \in \mathbb{R}^{2|\mathcal{E}|}$*

(i) the discrete integrability condition is fulfilled for all but one arbitrary vertex,

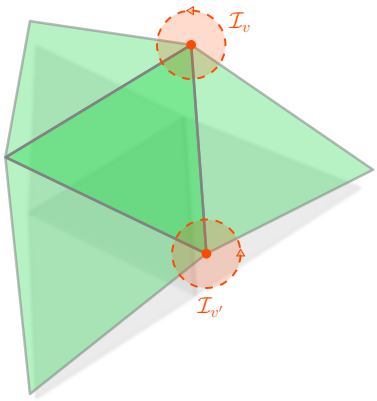
(ii) the discrete integrability condition is fulfilled for every element of a dual homology basis (non-contractible loops), and

(iii) the accumulated translation along every loop in a primal homology basis is zero,

then we can create an immersion of $\mathcal{S}_h$ in three-dimensional space admitting z as lengths and angles, unique up to rigid body motions.

Proof. For a simply connected surface with boundary, this is Theorem 3.1.

For a closed surface with genus zero, removing one triangle allows to apply Theorem 3.1 and reconstruct all vertex positions. The three additional dihedral angles appearing when adding the triangle back in are consistent with z as they are completely determined by the integrability conditions for any two of the triangle's three vertices.



For a not simply connected surface, for example, a closed surface of higher genus, we need to adjust the proof of Theorem 3.1 taking into account the non-contractible primal and dual loops. We do this by adding them to the start of the induction. In case of frame construction, this leads to the second, additional condition on the transition rotations, because now each dual loop can be broken down into loops around vertices and the non-contractible loops in a dual homology basis. Before, we had no explicit condition on translations, because they were automatically consistent for a single triangle and this sufficed as start of the induction. However, in the case of not simply connected surfaces, we will get non-contractible primal edge loops that are not reducible to cycles of edges of a single triangle. Hence, we have to explicitly enforce that the accumulated translation is zero for each element in a basis of non-contractible primal edge loops, which is the third condition. If we put all this together, we have extended the proof of Theorem 3.1 to the general case. □

Example 3.1. We want to illustrate the additional conditions' necessity by the simple example of a ring, shown in Figure 3.2. First, we see that fulfilling the integrability condition for each interior vertex is an empty condition, as there are none. Still, ensuring that the frames match when reconstructing them along the ring is clearly necessary to have consistent dihedral angles, which is achieved by the second condition of the global theorem. Nevertheless, this still does not suffice, as we could accumulate unwanted translations when reconstructing the vertex positions around the ring. This

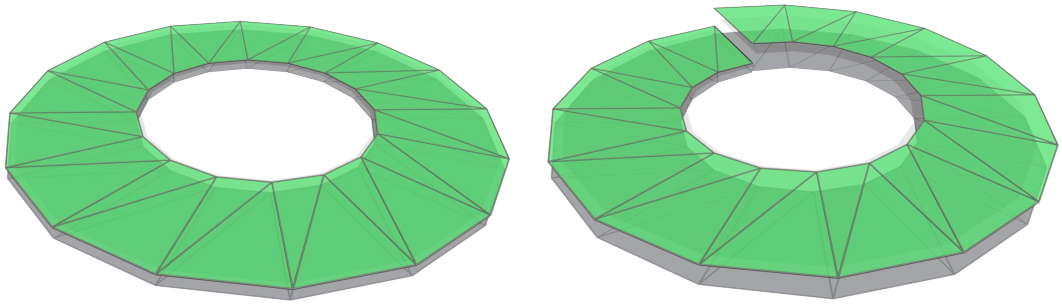


Figure 3.2: Example showing possible inconsistency occurring when the accumulated translation is not controlled for non-contractible primal cycles.

phenomenon is illustrated in the right half of Figure 3.2, where we can certainly find a dihedral angle for the edge at the jump, such that the frames match, i.e. the integrability condition is fulfilled, and also the induced edge vectors in both adjacent triangles agree. The accumulated translation along the edge loop around the hole, however, makes it impossible to find an immersion admitting this dihedral angle because it would lead to the seen gap. Hence, we have to explicitly require that the accumulated translation along this loop is zero, which is exactly the third condition of the global theorem.

Remark on implementation. The basis of the dual non-contractible cycles can be computed using the *tree-cotree decomposition* of Eppstein [Epp03], and a corresponding basis of the primal non-contractible loops can be derived from it. In our current implementation, only the conditions for simply connected surfaces are included.

3.3 Deformation Energies on Lengths and Angles

In Section 1.3, we have presented a deformation energy on discrete surfaces in dense correspondence which we used in the context of a variational time-discretization of geodesic calculus in Section 2.2. Our goal is to use lengths and angles as a parameterization of the immersions of a discrete surface to allow for efficient computations in shape space. This raises the need for deformations energies $\mathcal{W}$ on lengths and angles.

3.3.1 Quadratic Energy

The simplest way to define a deformation energy on lengths and angles would be to just use the squared euclidean distance $\|z-\tilde{z}\|^2$. Building on this idea, we can consider the case of a weighted quadratic energy.

Definition 3.11 (Quadratic deformation energy)**.** For a discrete surface $\mathcal{S}_h$, we define the quadratic deformation energy on lengths and angles $z, \tilde{z} \in \mathbb{R}^{2|\mathcal{E}|}$ as

$$\mathcal{W}_q[z,\tilde{z}] = \sum_{e\in\mathcal{E}} w_{s,e}\|l_e - \tilde{l}_e\|^2 + \eta \sum_{e\in\mathcal{E}} w_{b,e}\|\theta_e - \tilde{\theta}_e\|^2, \tag{3.19}$$

where the $w_{s,e}$ and $w_{b,e}$ are weights tailored to the application.

The second term measures the difference in the bending between triangles. In fact, it is the same approximation used in the Discrete Shells energy for nodal positions and we call it the *bending term*, which makes η the *bending weight*. The first term measures the stretching of the triangles, hence the failure of the deformation to be an isometry and is zero for isometric deformations. Therefore, we call it the *membrane term*.

The edge specific weights $w_{s,e}$ and $w_{b,e}$ make the energy robust to remeshing and can be chosen in different ways. The choice introduced along with this quadratic model in [FB11] is

$$w_{s,e} = \frac{1}{\bar{l}_e^2}, \quad w_{b,e} = \frac{\bar{l}_e^2}{\bar{d}_e}, \tag{3.20}$$

and another choice, used in [HRS+16], is

$$w_{s,e} = \frac{\bar{d}_e}{\bar{l}_e^2}, \quad w_{b,e} = \frac{\bar{l}_e^2}{\bar{d}_e}. \tag{3.21}$$

In both cases, we use the lengths $\bar{l}_e$ and edge-associated areas $\bar{d}_e$ (cf. Definition 1.29) of some reference configuration $\bar{X}$. In practice, this reference configuration may be chosen differently for each problem we are solving to improve the quality of the results, even though it then does not induce a consistent metric anymore.

This constant choice of weights leads to an energy quadratic in the lengths and angles, which we consider an approximation of the thin shells deformation energy introduced in Definition 1.30. From a computational point of view, this is, of course, a benefit leading to faster and more robust computations. On the other hand, this means that all the variables are independent of each other, which is a drawback from a physical point of view. One instance of this drawback is that there is no penalization of the degradation of triangles and compression, which might lead to very short edges and very small interior angles.

3.3.2 Thin Shells Energy

In Section 1.3, we have derived a physically sound model for the deformation behavior of discrete surfaces. They were built on the discrete fundamental forms and can thus be adapted easily to lengths and angles as primary degrees of freedom. This will be done in this section, where we implicitly assume that all considered sets of lengths and angles $z \in \mathbb{R}^{2|\mathcal{E}|}$ fulfill at least the triangle inequality to make the needed results from trigonometry well-defined.

Membrane energy The membrane part penalized the deviation from being an isometry by considering the discrete distortion tensor $\mathcal{G}[X, \tilde{X}]|_f = (G|_f)^{-1}\widetilde{G}|_f$ and applying a nonlinear energy density to it, which has a global minimum at the identity. At the beginning of this chapter, in Section 3.1, we discovered that we can express the discrete first fundamental form solely in terms of edge lengths. Now, we can combine this to derive a physically sound membrane deformation model for lengths and angles. To achieve this, we express the trace and determinant of $\mathcal{G}[X, \tilde{X}]|_f$ in terms of edge lengths, interior angles, and triangle areas, which can all be derived from edge lengths. The determinant takes the form,

$$\det \mathcal{G}[X, \tilde{X}]|_f = (\det G|_f)^{-1} \det \widetilde{G}|_f = a_f^{-2}\, \tilde{a}_f^2, \tag{3.22}$$

whereas the trace is not as straightforward. First, we recall the inverse of the discrete fundamental form, given by Cramer's rule as

$$(G|_f)^{-1} = \frac{1}{\det G_f} \begin{pmatrix} \|E_1\|^2 & \langle E_1, E_2\rangle \\ \langle E_1, E_2\rangle & \|E_2\|^2 \end{pmatrix},$$

which implies

$$\operatorname{tr} \mathcal{G}[X, \tilde{X}]|_f = \frac{1}{4a_f^2} \left(\|E_1\|^2 \|\widetilde{E}_2\|^2 - 2\langle E_1, E_2\rangle \langle \widetilde{E}_1, \widetilde{E}_2\rangle + \|\widetilde{E}_1\|^2 \|E_2\|^2 \right).$$

As the computation of all terms only requires edge lengths, we define $\mathcal{G}[z, \tilde{z}]|_f$ as the distortion tensor induced by lengths and angles, which agrees for admissible ones with above's definition. This allows us to define the membrane energy.

Definition 3.12 (Membrane energy)**.** For a discrete surface $\mathcal{S}_h$, we define the membrane energy on lengths and angles $z, \tilde{z} \in \mathbb{R}^{2|\mathcal{E}|}$ as

$$\mathcal{W}_{\text{mem}}[z, \tilde{z}] = \sum_{f \in \mathcal{F}} a_f \cdot W_{\text{mem}}(\mathcal{G}[z, \tilde{z}]|_f), \tag{3.23}$$

where

$$W_{mem}(A) := \frac{\mu}{2} \operatorname{tr} A + \frac{\lambda}{4} \det A - \left(\mu + \frac{\lambda}{2}\right) \log \det A - \mu - \frac{\lambda}{4},$$

for positive material constants μ and λ.

Bending energy We have seen that the bending model in Section 1.3 was based on the discrete shape operator, which is defined using the first and second fundamental form and thus can be described using only lengths and angles. Now, the discrete bending energy from Definition 1.28 could be completely reformulated using lengths and angles. Yet, we will only consider the approximation through the Discrete Shells bending energy (cf. Definition 1.29), because it is the energy we are interested in for practical applications. To express this energy in lengths and angles requires no further calculations, and as before we replace its primary variables with lengths and angles.

Definition 3.13 (*Discrete Shells* bending energy)**.** For a discrete surface $\mathcal{S}_h$, we define the Discrete Shells bending energy on lengths and angles $z, \tilde{z} \in \mathbb{R}^{2|\mathcal{E}|}$ as

$$\mathcal{W}_{\text{bend}}[z, \tilde{z}] = \sum_{e \in \mathcal{E}} \frac{(\theta_e - \tilde{\theta}_e)^2}{d_e} l_e^2, \tag{3.24}$$

where $d_e = \frac{1}{3}(a_f + a_{f'})$ for the two faces f, f' adjacent to $e \in \mathcal{E}$.

Deformation energy As before, we combine the membrane and bending energy in a weighted sum.

Definition 3.14 (Discrete deformation energy)**.** Let $\mathcal{S}_h$ be a discrete surface and let $z, \tilde{z} \in \mathbb{R}^{2|\mathcal{E}|}$ be two sets of lengths and angles. The *thin shells energy on lengths and angles* is defined by

$$\mathcal{W}_{\text{TS}}[z, \tilde{z}] = \mathcal{W}_{\text{mem}}[z, \tilde{z}] + \eta \mathcal{W}_{\text{bend}}[z, \tilde{z}], \tag{3.25}$$

where the *bending weight* η represents the squared thickness of the shell.

3.4 Reconstruction

In many applications, we are eventually not interested in the lengths and angles themselves but the corresponding immersion. Thus, we need to able to reconstruct it in terms of nodal positions. This means we are looking for a map $\mathcal{R}\colon \mathcal{Z} \to \mathbb{R}^{3|\mathcal{V}|}$ such that $z \circ \mathcal{R}$ is the identity, where $Z\colon \mathbb{R}^{3|\mathcal{V}|} \to \mathcal{Z}$ is the map to lengths and angles, as before. Note, that we cannot expect $\mathcal{R} \circ Z \equiv \mathrm{Id}$, as lengths and angles describe the immersion only up to rigid body motions. In particular, this means that we aim for $\mathcal{R} \circ Z(X) = QX$, where $Q \in SE(3)$ is a rigid body motion acting on the immersion. This is in fact already implied by $Z \circ \mathcal{R} = \mathrm{Id}$ and the discrete fundamental theorem we have seen.

Furthermore, we want to reconstruct immersions from elements $z \in \mathbb{R}^{2|\mathcal{E}|}$ even if they are not admissible and thus, there will not be an immersion admitting those lengths and angles. Formally, we want to extend our map $\mathcal{R}$ to the ambient space, i.e. $\mathcal{R}\colon \mathbb{R}^{2|\mathcal{E}|} \to \mathbb{R}^{3|\mathcal{V}|}$, such that $\mathcal{R}|_{\mathcal{Z}}$ is still a right inverse of Z.

We will introduce two classes of methods to define and compute such a reconstruction. The first one is based on computing a deformation of a given immersion and thus solving an optimization problem in the nodal positions, whereas the second one is based on the constructive nature of the proof of Theorem 3.1. Afterwards, we will compare both in numerical experiments on different examples.

3.4.1 Energy-based Reconstruction

The underlying idea of this approach is to search for the element of the admissible lengths and angles $\mathcal{Z}$ closest to a target $z^* = (l^*, \theta^*) \in \mathbb{R}^{2|\mathcal{E}|}$. Hence, we want to compute the element $z \in \mathcal{Z}$ which minimizes $\mathrm{dist}^2(z^*, z)$, i.e. for which the squared distance $\mathrm{dist}^2(z^*, \mathcal{Z})$ is obtained. To this end, we will use the description of $\mathcal{Z}$ using nodal positions. Recall that by $Z(X)$ we denote the lengths $l(X)$ and dihedral angles $\theta(X)$ of the immersed discrete surface with immersion/nodal positions $X \in \mathbb{R}^{3|\mathcal{V}|}$.

Now, the above's distance has to be elaborated. As introduced in Section 3.3, we choose some deformation energy $\mathcal{W}$ on lengths and angles as approximation of a notion of squared distance. Usually, we pick the same energy used to compute the deformations. This enables us to extend our physically motivated setup for $\mathcal{Z}$ to obtain a dissimilarity measure on arbitrary elements of $\mathbb{R}^{2|\mathcal{E}|}$. Additionally, we parameterize $\mathcal{Z}$ by nodal positions, hence replace $z \in \mathcal{Z}$ by $Z(X)$ where $X \in \mathbb{R}^{3|\mathcal{V}|}$. This leads to the resulting minimization problem

$$\underset{X \in \mathbb{R}^{3|\mathcal{V}|}}{\text{minimize}} \quad \mathcal{W}\left[z^*, Z(X)\right] \tag{3.26}$$

and we obtain a reconstruction for z^* as its solution.

The solution to the above problems is not unique, as Z is invariant to rigid body motions. Hence, in practice, we need a way to deal with them and obtain a particular minimizer as we have already discussed at the end of Section 2.3. To obtain a map to $\mathbb{R}^{3|\mathcal{V}|}$, we introduce this in terms of an abstract operator $\mathcal{C}$ returning an element from the equivalence class of immersions solving (3.26).

Definition 3.15. Let $\mathcal{S}_h$ be an immersed discrete surface, let $z^* \in \mathbb{R}^{2|\mathcal{E}|}$, and let $\mathcal{W}$ be a deformation energy on lengths and angles. Then we define the *energy-based reconstruction map* as

$$\mathcal{R}_{\mathcal{W}}[z^*] := \mathcal{C}\left[\underset{X \in \mathbb{R}^{3|\mathcal{V}|}}{\arg\min}\, \mathcal{W}\left[z^*, Z(X)\right]\right], \tag{3.27}$$

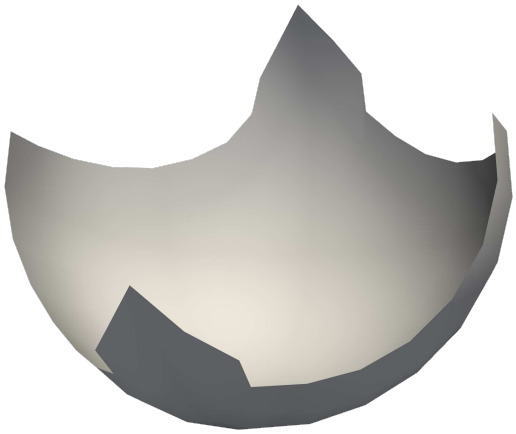


Figure 3.3: Energy-based reconstruction with $\mathcal{W}_q$ ($\eta = 1$) of lengths and angles of a plate leading to degenerated triangles, even though the lengths and angles fulfill the triangle inequalties.

where $\mathcal{C}$ is an operator choosing a particular minimizer.

Remark. Posing the reconstruction as a minimization problem certainly raises questions about the well-definedness, i.e. about existence and uniqueness of minimizers (beyond rigid body motions), which we will not cover in this thesis.

The main benefit of this method is that it easily allows treating target lengths and angles z^* which are not admissible. As long as they do not lead to an infinite value of the energy, we can simply plug them into the problem and solve it. One has to keep in mind that in this case, the resulting lengths and angles $Z(X)$ may be far away from z^* in the sense that the residual energy $\mathcal{W}[z^*, Z(X)]$ of the minimization problem might be large. Overall, we need to solve a high-dimensional and nonlinear optimization problem as (3.26) is formulated in nodal positions. In general, this might be a difficult and costly undertaking, which poses a potential disadvantage of this method.

We obtain a special variant of this method when choosing $\mathcal{W}$ to be the quadratic deformation energy from Definition 3.11. This is the way the method was first introduced by [FB11] and used for example in [HRS$^+$16]. This offers benefits from a computational point-of-view (see below), while on the other hand using the quadratic deformation energy might lead to unwanted non-physical behavior. Especially, if the target lengths and angles z^* are far away from $\mathcal{Z}$, artifacts may occur as seen in Figure 3.3, where the lacking penalization of compression leads to degenerate triangles. Nevertheless, in many instances, it leads to pleasant results in relatively short time.

Remark on implementation. The choice of the quadratic energy above turns the minimization problem (3.26) into a nonlinear least-squares problem. Indeed, this is beneficial from a computational point-of-view, as such problems can be treated efficiently by the Gauß-Newton method [NW06]. If we have an admissible target z^* and the initialization of the Gauß-Newton method is close to the solution, it usually converges in only a few iterations. We have also implemented the Hessian of the problem and tried using Newton's method without any approximation of the Hessian. In our experiments, this usually performed worse than using the Gauß-Newton approximation, which indicates that it seems to be an efficient

preconditioning for this problem. For more experiments and especially timings of this method, we refer to Section 3.4.3.

3.4.2 Frame-based Reconstruction

This method performs the reconstruction of nodal positions using frames and transition rotations, which were described in Section 3.2.2. The idea of the method was introduced in [LSLCO05] and the application to lengths and angles as described in this section was introduced in [WLT12]. First, we will look at a direct way to reconstruct nodal position based on the constructive proof of Theorem 3.2, and at a novel adaptive modification of this algorithm to apply it to non-admissible lengths and angles. Afterwards, we will briefly point at a relaxation which leads to a linear least-squares problem.

Direct reconstruction Assume, we are given an admissible target $z^* \in \mathcal{Z}$, consisting of edge lengths l^* and dihedral angles θ^*. Furthermore, as the reconstruction from lengths and angles is only defined up to rigid body motions, we assume that we are given the position X_0 of one vertex v_0 and the orientation of an adjacent triangle f_0 in the form of a frame F_0. Now, denote by f_{i_0}, f_{i_1}, and f_{i_2} the neighboring triangles of f_0. From z^*, we determine the induced transition rotations R_{0i_0}, R_{0i_1}, and R_{0i_2} (cf. Definition 3.8) and thus determine frames F_{i_0}, F_{i_1}, and F_{i_2} on the triangles by

$$F_j = F_0 R_{0j}. \tag{3.28}$$

Repeating this for all neighboring triangles of f_{i_0}, f_{i_1}, and f_{i_2} and continuing iteratively, we can construct frames F_f for all faces $f \in \mathcal{F}$. This is well-defined, in the sense that if we have two paths connecting a triangle f to f_0, then the frames constructed along the two paths agree by Theorem 3.2.

Given frames F_f for all faces $f \in \mathcal{F}$ and hence the orientation of all triangles, we next reconstruct the nodal positions. To this end, recall that two frames for the same embedded triangle only differ by a rotation around the normal and hence we have an additional degree of freedom. We get rid of it by assuming that the given frame F_0 and thus the others are in fact standard discrete frames. Given one nodal position of a triangle, our goal is to construct the remaining two positions using the given frame and edge lengths. If we repeat this iteratively, starting with v_0 and f_0, we can then reconstruct all nodal positions. For this, consider a face f with vertices $(v_i)_{i\in\{1,2,3\}}$, nodal positions $(X_i)_{i\in\{1,2,3\}}$, edge vectors $E_j = X_{j-1} - X_{j+1}$ (indices modulo 3), discrete frame F, and target edge lengths $(l_i^*)_{i\in\{1,2,3\}}$. As we have chosen F to be a standard discrete frame, we obtain that

$$E_1 = l_1^* F \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \; E_2 = l_2^* F \begin{pmatrix} -\cos\gamma_3 \\ \sin\gamma_3 \\ 0 \end{pmatrix}, \; E_3 = l_3^* F \begin{pmatrix} -\cos\gamma_2 \\ -\sin\gamma_2 \\ 0 \end{pmatrix}, \tag{3.29}$$

as visualized in Figure 3.4. Combining these two steps, we obtain Algorithm 1.

As one can imagine, the direct frame-based reconstruction with a spanning tree of the dual graph built by breadth-first search is very sensitive to violations of the integrability. In this case, the violation means that reconstructing nodal positions of a face along two different paths connecting it to the initial face f_0 would lead to different results. Hence, we get inconsistent nodal positions for the triangle leading to visual artifacts. Moreover, the errors occurring when walking over such a violation propagate

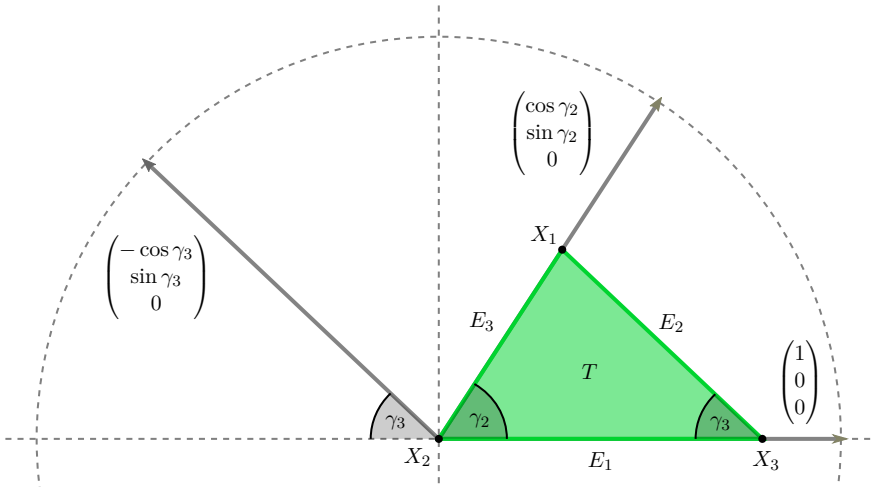


Figure 3.4: Vertex reconstruction from identity frame $F = \mathrm{Id}$ and vertex X_2 at the origin

Algorithm 1 Direct frame-based reconstruction

Input: Discrete surface $\mathcal{S}_h = (\mathcal{V}, \mathcal{E}, \mathcal{F})$, admissible target edge lengths l^* and dihedral angles θ^*, position of first vertex X_0, frame F_0 for first face f_0

Output: Nodal positions X with $l(X) = l^*$ and $\theta(X) = \theta^*$

1: Traverse the dual graph of $\mathcal{S}_h$ breadth-first starting at f_0
2: **for all** dual edges (f_i, f_j) **do**
3: Construct induced transition rotation R_{ij}
4: Construct frame $F_j = F_i R_{ij}$
5: Compute interior angles $\gamma_1, \gamma_2, \gamma_3$ of f_j
6: Determine edge vectors E_1, E_2, E_3 according to (3.29)
7: Determine nodal positions for f_j from known vertices and E_1, E_2, E_3

to all following reconstructions and are even amplified. We will see examples for this later in the numerical experiments.

Adaptive spanning trees Next, we want to change the order of reconstruction in Algorithm 1 to make it more robust against violations of the integrability. In examples, we observed that this violation is highly localized, as the deformation itself is highly localized, see Section 3.4.3. Furthermore, we do not need to reconstruct the orientation of all triangles to reconstruct the nodal positions, as some will directly follow from already constructed vertices. Thus, an instinctive idea is to build a spanning tree which is adapted to this localization and tries to avoid the locations of the violation during reconstruction as much as possible. To this end, we introduce a novel modification of the frame-based reconstruction which we call *adaptive frame-based reconstruction*.

First, we introduce edge weights $w \in \mathbb{R}^{|\mathcal{E}|}$ for the dual graph based on the integrability. Each dual edge corresponds to a primal edge $e = (vv') \in \mathcal{E}$ and we can assign a value of integrability w_e by averaging the violation of integrability at the two adjacent vertices v and v'. For this, we need a scalar version of this integrability violation. Remember, that $\mathcal{I}_v(z)$ is always a rotation matrix and thus its distance to the identity is measured by its rotation angle. This angle can, in turn, be computed from the trace of $\mathcal{I}_v(z)$, which is three for the identity and less otherwise. Therefore, we define our

Algorithm 2 Adaptive frame-based reconstruction

Input: Discrete surface $\mathcal{S}_h = (\mathcal{V}, \mathcal{E}, \mathcal{F})$, target edge lengths l^* and dihedral angles θ^*, position X_0 of first vertex, frame F_0 for first face f_0

Output: Nodal positions X with $l(X) \approx l^*$ and $\theta(X) \approx \theta^*$

1: Evaluate the discrete integrability map $\mathcal{I}_v(z)$ for each vertex $v \in \mathcal{V}$

2: Define edge weights $w_e = \frac{|\operatorname{tr} \mathcal{I}_v(z^*)-3|+|\operatorname{tr} \mathcal{I}_{v'}(z^*)-3|}{2}$ for $e = (vv') \in \mathcal{E}$

Alternative 1:

3: Construct a minimal spanning tree of the dual graph with edge weights w

Alternative 2:

3: Construct a shortest path tree of the dual graph with edge weights w

4: Iterate as in Algorithm 1

edge weights to be

$$w_e := \frac{|\operatorname{tr} \mathcal{I}_v(z^*) - 3|+|\operatorname{tr} \mathcal{I}_{v'}(z^*) - 3|}{2}. \tag{3.30}$$

Now, we use these weights to build a spanning tree adapted to the problem. The first variant is to construct a minimal spanning tree (MST) of the dual graph, which is built such that the sum of all edge weights in the tree is minimal. For instance, this can be done by Prim's algorithm and provides a way to traverse the dual graph while avoiding unnecessarily large violations of the integrability. Another variant is to construct a shortest path tree (SPT), which is built such that the path distance from the root to any other vertex in the tree is the shortest path distance in the whole dual graph. For instance, this can be achieved by Dijkstra's algorithm and provides a way to traverse the dual graph such that for each face the sum of integrability violation along the dual path used for its reconstruction is minimal. Then, we use this spanning tree as before, which leads to Algorithm 2.

Remark (Preassembled spanning tree). When executing Algorithm 2 the main part of its runtime is needed for the construction of the minimal spanning tree. Hence, if we plan to reconstruct numerous immersions of a discrete surface with a very high resolution (i.e. many vertices) it would be desirable to use a preassembled spanning tree. Of course, this preassembled tree has to be reasonable for all sets of lengths and angles we plan to reconstruct. We will now present a simple approach for this based on sampling.

Assume that we only want to reconstruct lengths and angles from a compact subset $U \subset \mathbb{R}^{2|\mathcal{E}|}$. A particularly interesting instance of this would be an affine subspace with bounds for the coefficients of the different coordinate directions. Then, we draw finitely many samples $(z_i)_{i=1}^{n_s} \subset U$ and compute for each of them the edge weights $(w_e^{(i)})_{i=1}^{n_s}$ from above. Those samples can be distributed uniformly over all of U or can be only a set of extremal points, such as the points with maximal coefficients in a single direction in the affine subspace case. Those edge weights are combined to a single weight vector by taking the maximum, i.e. $\tilde{w}_e = \max\left\{w_e^{(i)} \mid i \in \{1, \dots, n_s\}\right\}$. Now, we can construct a minimal spanning tree with respect to these weights and use it for the reconstruction as in Algorithm 2.

Remark on implementation. In practice, we only once need to construct more than one nodal position for a face f in Algorithm 1, namely for the first triangle. For all other triangles we need to construct at most one nodal position, and often

even none as the positions are already determined.

Furthermore, the traversal of the dual spanning tree for reconstructing faces and nodal positions could be parallelized to speed up both algorithms. So far, we have not done this in our implementation.

In the formal exposition above, we have assumed that the triangle inequality is not violated for the target $z^* \in \mathbb{R}^{2|\mathcal{E}|}$ because otherwise, the results from trigonometry are not well-defined anymore. However, in practice, sometimes we might have targets which do violate the triangle inequality and we still want to apply our algorithm to them. We incorporate this in all applications by setting the interior angles γ of a triangle $f \in \mathcal{F}$ with $\mathcal{T}_f(z^*) \leq 0$ to zero, as they cannot be determined from the edge lengths. By our definition of $\mathcal{I}$ and edge weights w, those triangles will be automatically considered as late as possible in the adaptive algorithm.

The overall runtime of Algorithm 1 and Algorithm 2 depends mainly on the implementation of the dual spanning tree. In the non-adaptive case, the breadth-first search has a worst-case time complexity of $\mathcal{O}(|\mathcal{F}| + |\mathcal{E}|)$ when implemented using priority queues. Thus, the overall algorithm also has a worst-case time complexity of $\mathcal{O}(|\mathcal{F}| + |\mathcal{E}|)$ as the time complexity of the reconstruction operation in Algorithm 1 is constant for each face. In the adaptive case, Prim's algorithm and Dijkstra's algorithm have both a worst-case time complexity of $\mathcal{O}(|\mathcal{E}| + |\mathcal{F}| \log |\mathcal{F}|)$ when implemented using Fibonacci heaps. In both cases, this leads to an overall time complexity of $\mathcal{O}(|\mathcal{F}| + |\mathcal{E}| + |\mathcal{F}| \log |\mathcal{F}|)$ for Algorithm 2.

Least-squares Relaxation Wang et al. proposed in [WLT12] a different way to handle non-admissible target lengths and angles $z^* \in \mathbb{R}^{2|\mathcal{E}|}$, when we cannot apply Theorem 3.2 anymore. In their approach, they relax the equalities in (3.28) and (3.29) from the direct reconstruction. They do so by only requiring them in a least-squares sense, i.e. for the frames, they introduce the quadratic mismatch energy

$$E_{\mathcal{F}}(F) = \frac{1}{2} \sum_{(i,j) \in \mathcal{E}^*} w_{ij,\mathcal{F}} \|F_j - F_i R_{ij}(z^*)\|_F^2 \tag{3.31}$$

and for edge vectors, they introduce

$$E_{\mathcal{E}}(X, F) = \frac{1}{2} \sum_{(i,j) \in \mathcal{E}} \sum_{f_k \supset (i,j)} w_{ij,\mathcal{E}} \|E_{ij} - F_k a_{ij}^k(z^*)\|^2, \tag{3.32}$$

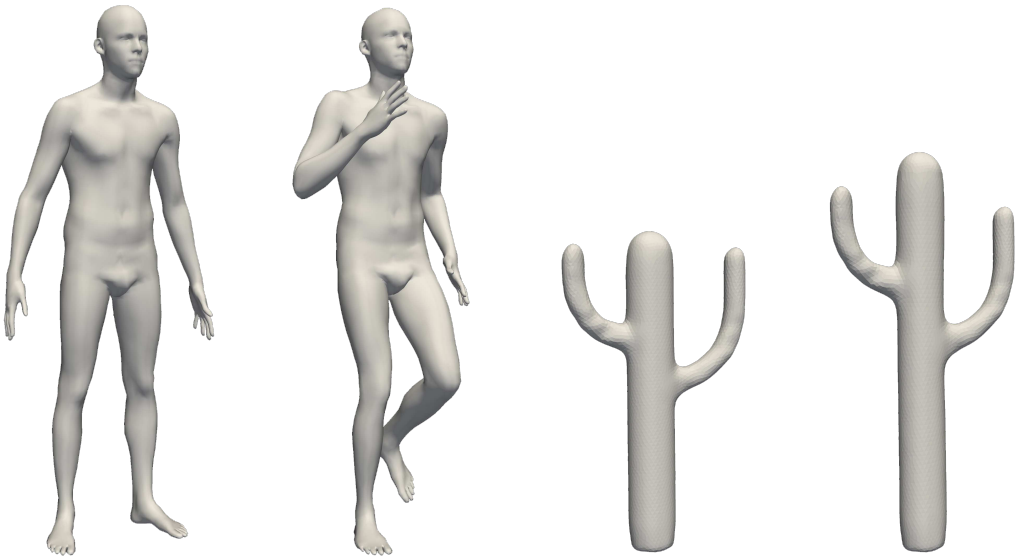
where the a_{ij}^k are the corresponding vectors from (3.29). The weights w serve again to increase robustness against remeshing and can, for example, be chosen as the cotan-weights [WLT12].

The nodal positions can be reconstructed from this in a two-step approach as in [LSLCO05]. First, the frames F are constructed by minimizing $E_{\mathcal{F}}$, which means finding the set of frames whose change over the discrete surface deviates from the prescribed transition rotations as little as possible. Then, the nodal positions X are reconstructed by minimizing $E_{\mathcal{E}}(\,\cdot\,, F)$, i.e. for given frames, they find the nodal positions which best fit to the ones prescribed from the different frames.

In [WLT12], these two steps were combined into a single step by introducing a composite energy

$$E_C(X, F) = w E_{\mathcal{F}}(F) + E_{\mathcal{E}}(X, F), \tag{3.33}$$

where w is relative weighting factor. By summing the two energies, they introduce a trade-off between meeting the prescribed transitions rotations and having vertex



(a) Two poses from the Dyna dataset (b) Two poses of the cactus

Figure 3.5: Input data of the two examples.

positions consistent with the edge lengths and the different frames. In both approaches, minimizing the energies means solving linear least-squares problems, which amounts to assembling and solving linear systems.

During the course of this thesis, we have not yet explored this relaxation approach. Especially, we have not evaluated its performance in comparison to our new adaptive approach. It could also provide a possible way to post-process the output of the adaptive approach to cheaply smooth out the remaining jumps. This would certainly be a relevant task for future work on extending our lengths and angles toolbox.

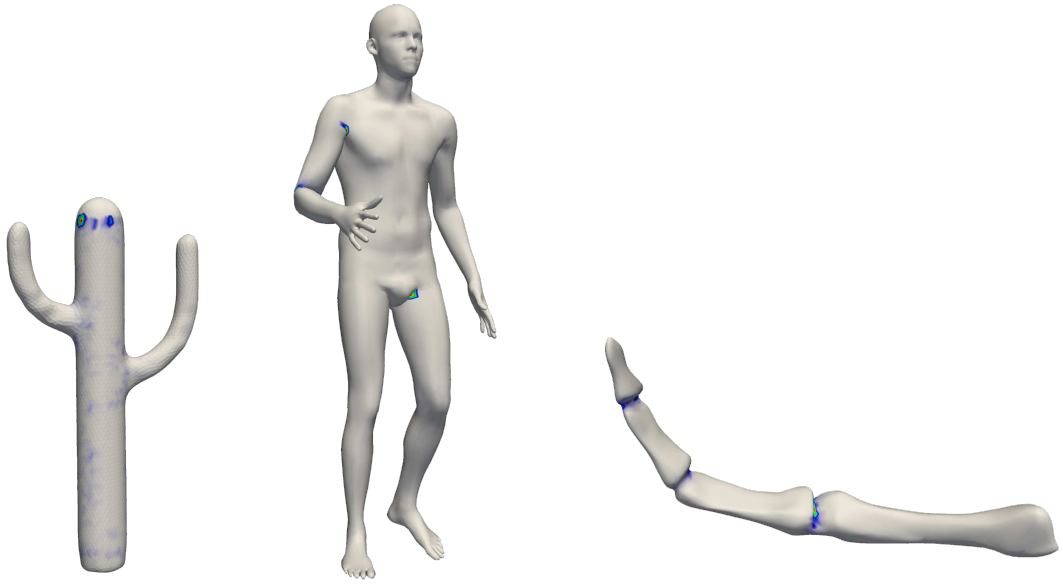
3.4.3 Numerical Experiments

Now, with several methods to nodal positions from length and angles at hand, we will compare them on different examples in numerical experiments.

Examples We consider three different examples from different datasets for our numerical experiments.

The first example consists of two poses of a human body model from the Dyna dataset [PMRMB15], seen in Figure 3.5a, which contains numerous human body shapes in dense correspondence with 6890 vertices each. The two poses feature the human once in a resting state and once in a running motion. We use the lengths and angles of the second immersion (the running one) as an admissible target, and the average of the lengths and angles $\frac{Z[X^1]+Z[X^2]}{2}$ of the two immersions X^1, X^2 as a non-admissible target. This leads to deformations with a lot of bending (i.e. change of dihedral angles) but only a small change in the edge lengths.

The cactus shape [BS08] with 5261 vertices of which we consider two different immersions is our second example, seen in Figure 3.5b. Again, we use the lengths and angles of the second immersion as an admissible and the average as a non-admissible target. It serves as an example for a rather smooth shape. The deformation is highly non-isometric but only exhibits minimal bending.



(a) Cactus (b) DYNA (c) Finger

Figure 3.6: Visualization of the violation of integrability on the three different examples using a rainbow color scale. Note, that the color scale does not match for the different examples.

The last example consists of two poses of a skeleton finger model from the Large Geometric Models Archive at the Georgia Institute of Technology, also used in [KMP07]. With 2046 vertices, it is the discrete surface with the fewest number of vertices among our examples. For the finger, we show only the average as a non-admissible target, because the second immersion did not offer additional insightful results.

The violation of integrability of the three resulting non-admissible targets is shown in Figure 3.6.

Parameters In the Dyna example, we used the quadratic energy $\mathcal{W}_q$ in the energy-based reconstruction with bending weight $\eta = 10^{-3}$ and the resulting optimization problem (3.27) was solved using the Gauß-Newton method. To fix the rigid body motions, the nodal positions of a single triangle on the front of the torso were fixed (cf. Figure 3.7e), which was also used as the starting point for the frame-based approaches.

In the Cactus example, in opposition to above, we used $\eta = 10^{-1}$. To fix the rigid body motions, the nodal positions of a single triangle on the bottom of the cactus were fixed (cf. Figure 3.8e), which was also used as the starting point for the frame-based approaches. For the smoothing of the results from the frame-based approaches, we used a single iteration of the Gauß-Newton method in the energy-based approach with the same parameters.

In the skeleton finger example, the same material parameters as in the cactus example were used in the energy-based reconstruction. To take care of the rigid body motions, the nodal positions of a single triangle on the base joint were fixed, which also served as the starting point for the frame-based approaches.

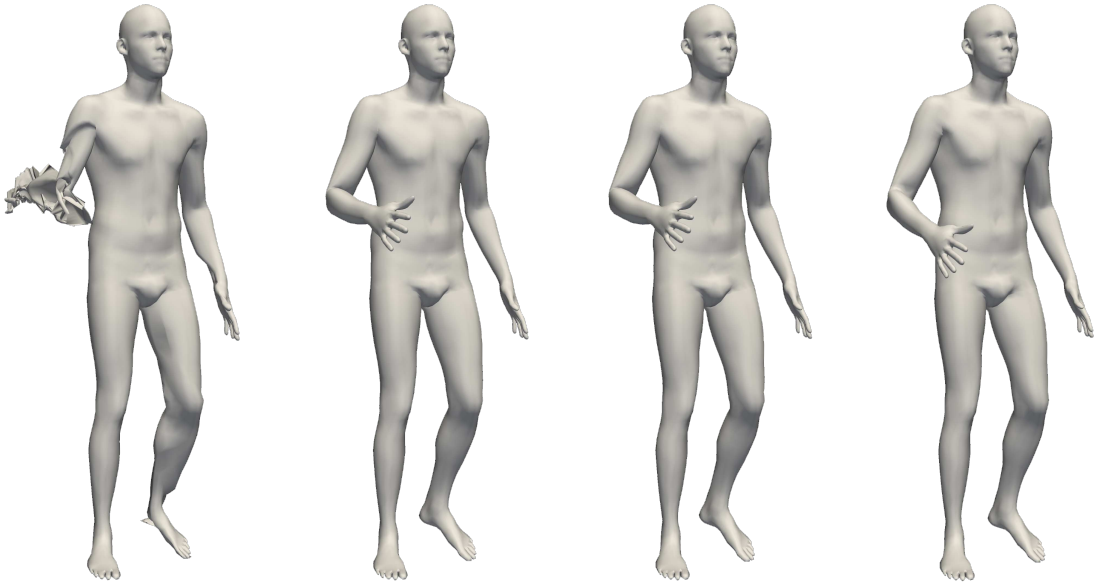
Results on admissible targets On the admissible example targets, the non-adaptive frame-based approach (Algorithm 1) worked without any problems and led to a reproduction of lengths and angles up to numerical errors. The energy-based approach with

Gauß-Newton converged for the cactus example to numerical precision within 5 iterations. For the Dyna example, however, it did not fully converge but stopped after approximately 20 iterations without finding an admissible step size. This behavior cannot be easily fixed by changing the material parameter η, as an increase worsens the convergence behavior and a decrease leads to unwanted crumpling of the surface. However, for the parameters given above, there occur no visual artifacts. Typically, one would use a multiresolution scheme to improve the convergence, as explained in [FB11], but as this was not our focus, we have not employed it within these experiments.

Results on non-admissible targets As we can see in Figure 3.7, the energy-based reconstruction with $\mathcal{W}_q$ works well on the non-admissible Dyna example for which it takes about 19 iterations, whereas the non-adaptive frame-based approach fails miserably. This is due to the violated integrability in both armpits, and at the left hip (cf. Figure 3.6b), which leads to clearly visible artifacts in both arms and the left leg, hence essentially all moving body parts. However, in Figure 3.6b we also see that the violation of integrability is highly localized, which allows the adaptive method to perform very well with both, a minimal spanning and a shortest path tree. Here, changing the way the adaptive spanning tree is built only slightly changes the results. We can see in Figure 3.7, that the adaptive methods create a different order for the reconstruction of frames and vertices, which avoids the violations of integrability, whereas the simple spanning tree built by breadth-first search simply walks over them. This leads to an increased depth of the dual spanning tree, meaning that the maximal number of reconstruction steps used to construct a triangle is larger for the adaptive spanning trees than the breadth-first tree. Furthermore, the results from the adaptive approach exhibit small differences compared to the reconstruction from the energy-based approach. We should point out, the energy-based result matches the target lengths and angles more precisely, which also leads to a visible difference in the right arm's pose. However, the convergence behavior of the latter is again dependent on the material parameters as explained above.

In the cactus example, seen in Figure 3.8, the energy-based approach works well and takes about 6 iterations for the shown result. Interestingly, we see that in this example the adaptive frame-based approach with both kinds of spanning trees leads to clearly visible jumps on the body of the cactus, whereas the non-adaptive reconstruction only exhibits some crumpling at the top of the cactus. A possible explanation is that the violation of integrability is not so strongly localized, even though, it is smaller than in the Dyna example. In Figure 3.6a, we can see that there are some hot spots at the top, but there is also some violation all over the body. Then, the adaptivity leads to spanning trees which are rather deep and there are triangles on the cactus, where two reconstruction paths meet. The jumps occur exactly at those points, which can be seen in Figure 3.8. Hence, in all cases, we smoothed the results using a single Gauß-Newton iteration of the energy-based reconstruction, which removed the artifacts entirely in the case of a breadth-first and minimal spanning tree. For the case of a shortest path tree, it at least reduced them visibly and using more iterations would remove them completely.

In the skeleton finger example, only the non-adaptive frame-based approach produced artifacts, which can be seen in Figure 3.9a, such as sharp edges on the tip of the finger. In this rendering, the results were colored by the mismatch to the target lengths and angles measured using $\mathcal{W}_q$, which shows that this mismatch is the largest at the artifacts. This localization of the mismatch at the visual artifacts could, in fact, be observed for all examples in this section. All other algorithms produced visu-



(a) Using Algorithm 1

(b) Using Algorithm 2 with MST

(c) Using Algorithm 2 with SPT

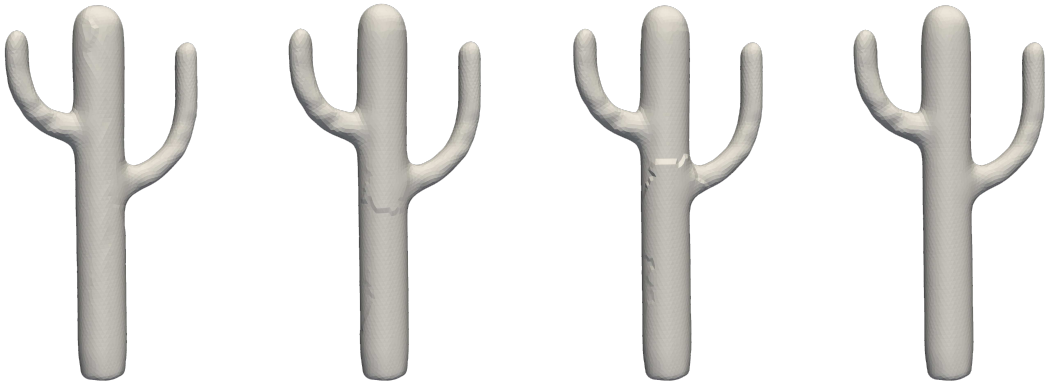
(d) Using (3.27) with $\mathcal{W}_q$

(e) Breadth-first search

(f) Minimal Spanning Tree

(g) Shortest Path Tree

Figure 3.7: Reconstruction of the non-admissible average of lengths and angles of the two Dyna poses (a – d). Number of reconstruction steps leading to a triangle (e – g), indicated by color from blue (few steps) to red (many steps). Note, that the color scale does not match for the different examples. The maximal number of steps is 138 for BFS, 305 for the MST, and 232 for the SPT.

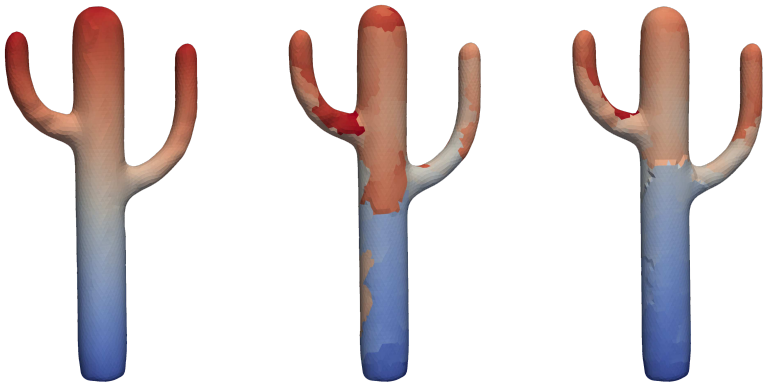


(a) Using Algorithm 1

(b) Using Algorithm 2 with MST

(c) Using Algorithm 2 with SPT

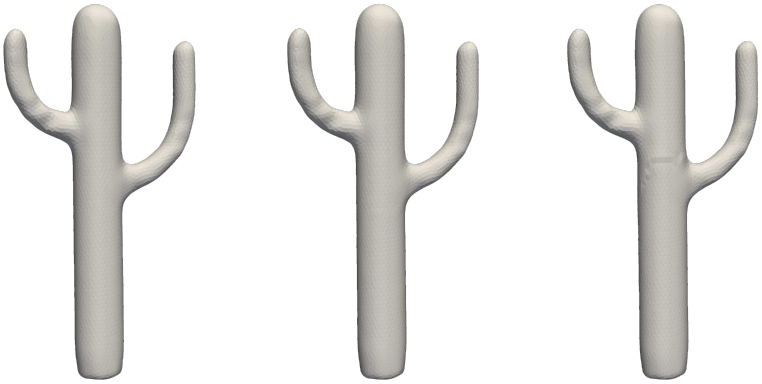
(d) Using (3.27) with $\mathcal{W}_q$



(e) Breadth-first search

(f) Minimal Spanning Tree

(g) Shortest Path Tree

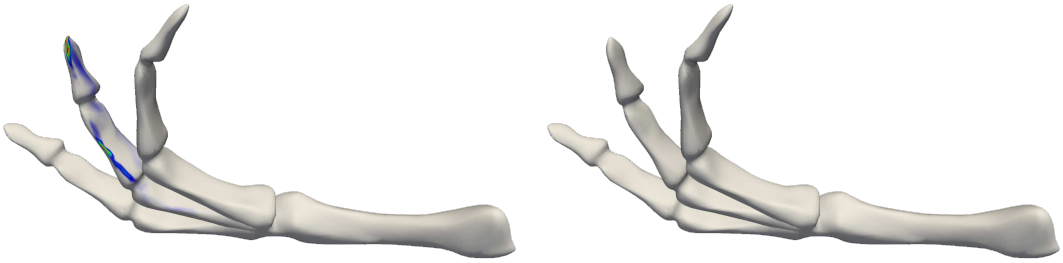


(h) BFS after smoothing

(i) MST after smoothing

(j) SPT after smoothing

Figure 3.8: Reconstruction of the non-admissible average of lengths and angles of the two cactus poses (a – d). Number of reconstruction steps leading to a triangle (e – g), indicated by color from blue (few steps) to red (many steps). Note, that the color scale does not match for the different examples. The maximal number of steps is 180 for BFS, 409 for the MST, and 297 for the SPT. Lastly, the results after smoothing the reconstruction (h – j).



(a) Using Algorithm 1 (b) Using Algorithm 2 with SPT

Figure 3.9: Reconstructed average of the lengths and angles of two finger skeleton poses. The color indicates the difference of the lengths and angles of the reconstructed shape to the prescribed target measured using $\mathcal{W}_q$.

Example				BFS		MST		SPT	
Model	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{E}\rvert$	$\lvert\mathcal{F}\rvert$	Tree	Traversal	Tree	Traversal	Tree	Traversal
Cactus	5.2k	16k	11k	0.9	7.9	9.6	4.3	9.0	4.6
Dyna	6.9k	21k	12k	0.9	9.6	12.2	5.1	11.9	5.3
Finger	2k	6.4k	4k	0.3	2.7	3.6	2.1	3.3	2.0

Table 3.1: Runtimes (in ms) of the two identified steps in the frame-based algorithms.

ally pleasing results, and we only illustrate the result of the adaptive approach with a shortest path tree in Figure 3.9b.

Timings As the last step of this section, we present the time it takes to compute the immersions of our examples.

The frame-based reconstruction algorithms consist of two parts mainly contributing to the runtime. First, establishing the dual spanning tree, which, in the case of adaptive variants, also entails the computation of the violation of integrability. The second component arises from traversing the dual spanning tree and actually reconstructing the nodal positions, which means that we have to compute the interior angles and the transition rotations.

On the other hand, the energy-based reconstruction, which performed is using the Gauß-Newton method, consists of four steps in each iteration constituting most of the runtime. First, there is the computation of the gradient, and following the computation of the Gauß-Newton approximation of the Hessian by squaring the gradient. Next, we need to compute the descent direction by solving a linear system, for which we use the UMFPACK solver from the SuiteSparse package [Dav04] in our current implementation. Finally, determining the step size is the last major step, which we have implemented by a simple backtracking line search accepting the first step leading to a decrease of the energy. We also tried out more involved line search approaches based on the Armijo or Wolfe conditions, but this did not lead to improved performance.

Overall, the time measurements show that for the adaptive frame-based reconstruction Algorithm 2, the computational effort for constructing the dual spanning tree is larger than for the actual reconstruction. This indicates that preassembling the tree whenever possible is important to achieve interactive rates, as proposed before. Nevertheless, for large-scale models, it is also necessary to reduce the cost of the actual reconstruction, for example by parallelization. Furthermore, if we want to use

Example	Total	Gradient	Hessian	Linear solve	Step size
Cactus	310	15 (5 %)	44 (14 %)	240 (77 %)	2 (1 %)
Dyna	328	19 (6 %)	49 (15 %)	265 (81 %)	8 (2 %)
Finger	90	5 (6 %)	13 (14 %)	74 (81 %)	2 (2 %)
Average		6 %	15 %	80 %	2 %

Table 3.2: Runtimes (in ms) of the identified steps in a Gauß-Newton iteration, along with their share of the overall runtime of an iteration.

Gauß-Newton as a possible way to smoothen the reconstruction while retaining the possibility to obtain interactive rates, we should first try to reduce the time it takes to solve the linear system.

Chapter 4

The Space of Lengths and Angles

So far, we have introduced lengths and angles as a discrete counterpart to the fundamental forms and presented a way to describe the ones belonging to immersions $\mathcal{Z}$. This yields natural degrees of freedom for our study of deformations of discrete surfaces, as they are rigid body motions invariant, appear naturally in our discrete deformation energies, and also provide a natural encoding of the locality of deformations. By the latter we mean, that lengths and angles change the most at edges naturally associated to a deformation, such as the edges at joints in human shapes, whereas the nodal displacement is the largest at points, whose alteration is a result of the deformation, exemplified by the tip of a finger. Due to their rigid body motion invariance, it is more practical to use model order reduction techniques, such as subspace methods in lengths and angles than in nodal positions. The reason is that such methods construct linear subspaces which requires special care in the presence of rigid body motions. Thus, lengths and angles describe natural degrees of freedom for the Riemannian shape space introduced in Chapter 2.

To this end, in this section, the structure of a Riemannian manifold will be introduced on $\mathcal{Z}$ and discrete geodesics will be considered as a key element of time-discrete geodesic calculus on it. Afterwards, we will present a computationally efficient approximation of this time-discrete geodesic calculus by working in the ambient space and projecting back onto $\mathcal{Z}$. This procedure was introduced in [FB11] and used for example in [HRS$^+$16]. Ultimately, we will study Principal Geodesic Analysis (PGA) on $\mathcal{Z}$ as an efficient way to conduct statistical analysis in the shape space of discrete shells, which was, in fact, the starting point of this thesis.

4.1 Nonlinear Structure

To allow the notion of $\mathcal{Z}$ being a nonlinear space, in the following, we are going to equip it with the structure of a manifold. This enables us to study and exploit its structure for computing deformations of discrete shells. In this chapter, we will assume $\mathcal{S}_h$ to be a closed simply connected discrete surface, to reduce our exposition to the local integrability conditions.

Remember, that in this case, we showed

$$\mathcal{Z} = \left\{ z \in \mathbb{R}^{2|\mathcal{E}|} \mid \mathcal{T}_f(z) > 0 \text{ for all } f \in \mathcal{F},\ \mathcal{I}_v(z) = \mathrm{Id} \text{ for all } v \in \mathcal{V} \right\}. \tag{4.1}$$

First, we note that the triangle inequalities $\mathcal{T}_f(z) > 0$ define an open subset of $\mathbb{R}^{2|\mathcal{E}|}$ and thus are not of interest for describing the nonlinear structure of $\mathcal{Z}$. However, for our following deduction to be well-defined, we need to restrict ourselves to the subset

fulfilling these inequalities. We will not state this explicitly but use $\mathbb{R}^{2|\mathcal{E}|}$ to simplify our notation, as all following notions need to hold only locally anyway.

Hence, we return to the integrability condition and its corresponding map $\mathcal{I}\colon \mathbb{R}^{2|\mathcal{E}|} \to \left(\mathbb{R}^{3\times 3}\right)^{|\mathcal{V}|}$, which in fact only maps to rotation matrices due to its structure and therefore, we have $\mathcal{I}\colon \mathbb{R}^{2|\mathcal{E}|} \to SO(3)^{|\mathcal{V}|}$. Now, we want to use this map as an implicit description of $\mathcal{Z}$ as a submanifold of $\mathbb{R}^{2|\mathcal{E}|}$, and thus go to a chart of $SO(3)$ about the identity. In fact, we already did this when explaining in Section 3.2.2 how to use the integrability conditions in implementations. There we used certain Euler angles to define a smooth integrability map $\widetilde{\mathcal{I}}\colon \mathbb{R}^{2|\mathcal{E}|} \to \mathbb{R}^{3|\mathcal{V}|}$ and this is exactly composing $\mathcal{I}$ with a chart about the identity for each vertex. Now, one would typically require the differential of $\widetilde{\mathcal{I}}$ to have full rank for each $z \in \mathcal{Z}$, but we have already seen that some vertices are redundant in Theorem 3.2, hence we cannot expect this to hold. However, we state the following conjecture, which would allow us to use $\widetilde{\mathcal{I}}$ as implicit description since it would induce a consistent dimension by the implicit function theorem.

Conjecture 1. *Let $\mathcal{S}_h$ be a discrete surface and let $\mathcal{Z} \subset \mathbb{R}^{2|\mathcal{E}|}$ be the set of its admissible lengths and angles. Then, for every $z \in \mathcal{Z}$, we have*

$$\operatorname{rank} D_z\widetilde{\mathcal{I}} = 3|\mathcal{V}| - 6. \tag{4.2}$$

In our numerical experiments, this was true in all instances. Yet, as of writing this thesis, rigorously proving the conjecture is still an open problem.

We will, however, assume that it holds, which allows the following definition.

Definition 4.1. Let $\mathcal{S}_h$ be a discrete surface. We call its admissible lengths and angles $\mathcal{Z} \subset \mathbb{R}^{2|\mathcal{E}|}$ together with the (sub)manifold structure induced on it by the implicit description through $\widetilde{\mathcal{I}}$ the *space of lengths and angles*.

Remark. Other names for this space include *discrete Gauß-Codazzi submanifold*, alluding to the resemblance of the discrete integrability conditions to the Gauß-Codazzi equations in the continuous case.

Remark (Dimension). Based on Conjecture 1, we can determine the dimension of $\mathcal{Z}$ as

$$\dim \mathcal{Z} = 3|\mathcal{V}| - 6 = |\mathcal{E}|. \tag{4.3}$$

Note, however, that this does not imply that the dihedral angles of a closed discrete surface are already determined by its edge lengths. A (local) parametrization of $\mathcal{Z}$ would need a different choice of coordinates.

Beyond the question of the definition being well-posed, there are more open questions regarding the structure of $\mathcal{Z}$, for example, whether it is connected or not. These might be interesting problems to tackle in future theoretical work.

Tangent space As we have introduced the structure of a submanifold of $\mathbb{R}^{2|\mathcal{E}|}$ on $\mathcal{Z}$ given by an implicit description, we can explicitly describe the tangent space at a point z as the kernel of the differential of $\widetilde{\mathcal{I}}$, i.e.

$$T_z\mathcal{Z} = \{v \in \mathbb{R}^{2|\mathcal{E}|} \mid D_z\widetilde{\mathcal{I}} \cdot v = 0\} =: \ker D_z\widetilde{\mathcal{I}}. \tag{4.4}$$

This explicit description of the tangent space, along with the possibility to numerically compute it (see below), is a very helpful tool in shape analysis. When working with nodal positions, the tangent space of $\mathcal{M}[\mathcal{S}_h]$ at an immersion X is a set of equivalence

classes, due to the need to factor out rigid body motions. This renders the numerical treatment of $\mathcal{M}[\mathcal{S}_h]$'s tangent bundle cumbersome and entails the development of complicated nonlinear tools in nodal positions as supplements. For the case of Principal Geodesic Analysis, this was done in [HZRS18], which lead to nonlinear, physically plausible modes of shape variation and motivated our study of lengths and angles to develop more efficient numerical schemes for this purpose.

Remark on implementation. Given the differential $A := D_z\widetilde{\mathcal{I}}$ of $\widetilde{\mathcal{I}}$ at z, numerically computing its kernel can be achieved in different ways. We can directly compute an orthonormal basis of it by computing the Singular Value Decomposition (SVD) of $D_z\widetilde{\mathcal{I}}$ and taking the right-singular vectors corresponding to the singular value zero. The same can be achieved by a rank-revealing QR decomposition.

If due to time or memory constraints, it is not feasible to compute a basis this way, we can compute the orthonormal projection of a vector $v \in \mathbb{R}^{2|\mathcal{E}|}$ onto the tangent space by

$$(\mathrm{Id} - A^\intercal (AA^\intercal)^{-1} A)v,$$

where the linear system $(AA^\intercal)^{-1}x = Av$ can be solved using a Cholesky decomposition. This allows efficiently computing the projection for various vectors.

In our implementation, we used the SVD from the Eigen library [GJO10] for small examples and the projection approach with the CHOLMOD solver from the SuiteSparse package [CDHR08] for larger examples.

Riemannian structure Next, we want to equip $\mathcal{Z}$ with a Riemannian structure, i.e. a smoothly varying scalar product g_z on $T_z\mathcal{Z}$. As before, we use a metric induced by a deformation energy $\mathcal{W}$ by taking the Euclidean Hessian of $\mathcal{W}$ and projecting it onto the tangent space $T_z\mathcal{Z}$.

Definition 4.2. Let $\mathcal{S}_h$ be a discrete surface, let $\mathcal{W}$ be a deformation energy on lengths and angles. Then we define the Riemannian metric induced by $\mathcal{W}$ as

$$g_z(v, w) := \frac{1}{2} v^\intercal \, \partial_2^2 \, \mathcal{W}[z, z] \, w, \tag{4.5}$$

for tangent vectors $v, w \in T_z\mathcal{Z}$.

This is consistent with the definition of Riemannian Hessians for submanifolds, as the part of it depending on the Riemannian structure of $\mathcal{Z}$ vanishes for zero gradients of the function extended to ambient space, which is the case for $\partial_2 \mathcal{W}[z, z]$.

This definition raises the question of the non-degeneracy (and positive-definiteness) of the Hessian for the two deformation energies we have introduced before. In the case of the quadratic energy $\mathcal{W}_q$, this is straightforward to see, as its Hessian is simply a diagonal matrix with the weights $w_{s,e}$ and $w_{b,e}$ as entries. For the discrete thin shells energy, this follows from the result on the non-degeneracy of its Hessian up to rigid body motions in the case of nodal positions as primal variables [HRS$^+$14, Thm. 2].

Time-discrete Geodesic Calculus We have seen before, in Section 2.2, that the central element of our time-discrete geodesic calculus is the notion of a discrete geodesic. Recall, that for fixed end points $z_0, z_K \in \mathcal{Z}$ it is defined as a minimizer $(z_1, \ldots, z_{K-1}) \in$

$\mathcal{Z}$ of $E^K[(z_0, \dots, z_K)]$. Hence, computing a discrete geodesic in the space of lengths and angles means solving the constrained optimization problem

$$\begin{aligned} \underset{z_1,\dots,z_{K-1}}{\text{minimize}} \quad & K \sum_{i=0}^{K-1} \mathcal{W}\left[z_i, z_{i+1}\right] \\ \text{subject to} \quad & \mathcal{T}_f(z_i) > 0 \text{ for each } i \in \{1, \dots, K-1\},\ f \in \mathcal{F} \\ & \widetilde{\mathcal{I}}_v(z_i) = 0 \text{ for each } i \in \{1, \dots, K-1\},\ v \in \mathcal{V}, \end{aligned} \tag{4.6}$$

which is a highly nonlinear problem, that is difficult to solve in general.

Remark on implementation. To solve (4.6), one needs to employ a robust numerical solver for inequality constrained, large-scale optimization problems. In our current implementation, we use the Ipopt software package [WB06] for this purpose, which is an interior-point filter line-search algorithm. This works quite well for smaller problems, such as the plate example shown throughout this chapter. However, for larger problems, we have observed that the Hessian of the Lagrangian is indefinite, which the algorithm tries to counter by regularization, but still fails to properly converge. This indicates, that we need to implement a trust-region method to solve constrained optimization problems in $\mathcal{Z}$ for discrete surfaces with a higher resolution, which is an important task for future work on the implementation.

4.2 Linear Approximation

In this section, we will present an approximation of time-discrete geodesic calculus on the space of lengths and angles originally introduced in [FB11]. It was, for example, was used in [HRS+16] to drastically speed up the computation of Riemannian splines in the space of discrete shells. Instead of enforcing to stay on $\mathcal{Z}$, we exploit that lengths and angles are rigid body motions invariant coordinates for surfaces and thus can be treated more easily in a linear approximation. Hence, we can work in ambient space $\mathbb{R}^{2|\mathcal{E}|}$ and then project the results back onto $\mathcal{Z}$ using a projection operator $\mathcal{P}_{\mathcal{Z}}$, which we will define in the following. This leads to a very computationally efficient scheme for computing notions from time-discrete geodesic calculus, especially if we combine it with the quadratic deformation energy from Definition 3.11.

Projection The projection operator $\mathcal{P}_{\mathcal{Z}}$ is typically realized using the reconstruction algorithms from Section 3.4, as these automatically yield the immersed surface $\mathbf{S}$ belonging to the projected lengths and angles. In the following, we will focus on the energy-based reconstruction from Definition 3.15, which is primarily used in the implementation and has the nicest interpretation in the geometric context of $\mathcal{Z}$. Recall, we parameterized $\mathcal{Z}$ using nodal positions and minimized the difference in energy to a given target. We adopt this approach to define a projection operator.

Definition 4.3 (Energy-based projection)**.** Let $\tilde{z} \in \mathbb{R}^{2|\mathcal{E}|}$. Then, we define the energy-based projection onto $\mathcal{Z}$ by

$$\mathcal{P}_{\mathcal{Z}}^{\mathcal{W}}[\tilde{z}] := Z\left[\underset{X \in \mathbb{R}^{3|\mathcal{V}|}}{\arg\min}\, \widehat{\mathcal{W}}\left[\tilde{z}, Z[X]\right]\right]. \tag{4.7}$$

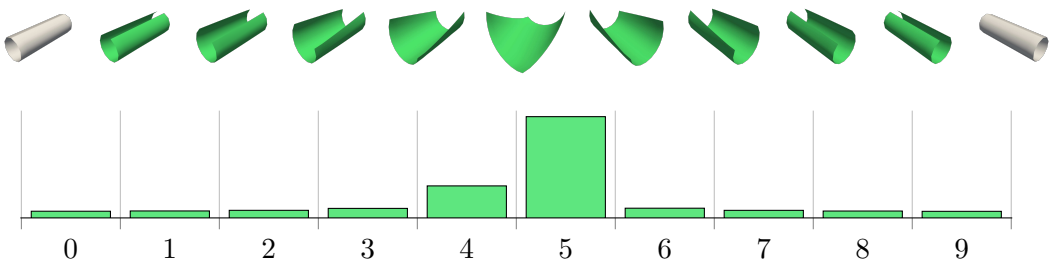


Figure 4.1: The approximation of a discrete geodesic between two plates rolled up in different directions computed using the quadratic energy $\mathcal{W}_q$, and the energy-based projection. Additionally, an diagram showing the energy between two steps.

Note, that this definition allows the nice reformulation to $\mathcal{P}_{\mathcal{Z}}^{\mathcal{W}}[\tilde{z}] = \arg\min_{z\in\mathcal{Z}} \widehat{\mathcal{W}}[\tilde{z}, z]$, up to the issues of existence and uniqueness of minimizers discussed above. Such an interpretation is not available for other reconstruction operators, nevertheless, we can generalize our definition to them.

Definition 4.4 (Projection by reconstruction)**.** Let $\tilde{z} \in \mathbb{R}^{2|\mathcal{E}|}$, and let $\mathcal{R}\colon \mathbb{R}^{2|\mathcal{E}|} \to \mathbb{R}^{3|\mathcal{V}|}$ be an reconstruction operator extended to ambient space. Then, we define the corresponding projection by

$$\mathcal{P}_{\mathcal{Z}}^{\mathcal{R}}[\tilde{z}] := Z\left[\mathcal{R}[\tilde{z}]\right]. \tag{4.8}$$

Note, that in both cases we have, at least in theory, $\mathcal{P}_{\mathcal{Z}}[z] = z$ for $z \in \mathcal{Z}$, which means that we have indeed defined projections. Moreover, let us stress the distinction between a projection, which yields elements in $\mathbb{R}^{2|\mathcal{E}|}$, and a reconstruction, which yields nodal positions in $\mathbb{R}^{3|\mathcal{V}|}$, even though they are closely related in our setup above.

Time-discrete geodesic Again, we focus on the time-discrete geodesic as the central building block, which also explains exemplary how to adapt other computations. In the approximation, it is computed by first solving the unconstrained minimization problem

$$\underset{\tilde{z}_1,\dots,\tilde{z}_{K-1}\in\mathbb{R}^{2|\mathcal{E}|}}{\text{minimize}} \quad K\sum_{i=0}^{K-1} \mathcal{W}\left[\tilde{z}_i, \tilde{z}_{i+1}\right], \tag{4.9}$$

for fixed end points $z_0, z_K \in \mathcal{Z}$ and afterwards applying the projection to get the approximate discrete geodesic $(z_1, \dots, z_{K-1}) \in \mathcal{Z}$ by

$$z_k = \mathcal{P}_{\mathcal{Z}}[\tilde{z}_k], \text{ for } k = 1, \dots, K-1. \tag{4.10}$$

This projection can be computed in parallel for each time step, which can speed-up the computation significantly.

The downside of this approach is that the resulting approximations may have artifacts on the one hand but also more subtle shortcomings such as not being equidistant with respect to the energy on $\mathcal{Z}$. This may lead to visible jumps in the animation, as seen in Figure 4.1, whereas the discrete geodesic computed by solving (4.6) is equidistant (cf. Figure 4.2).

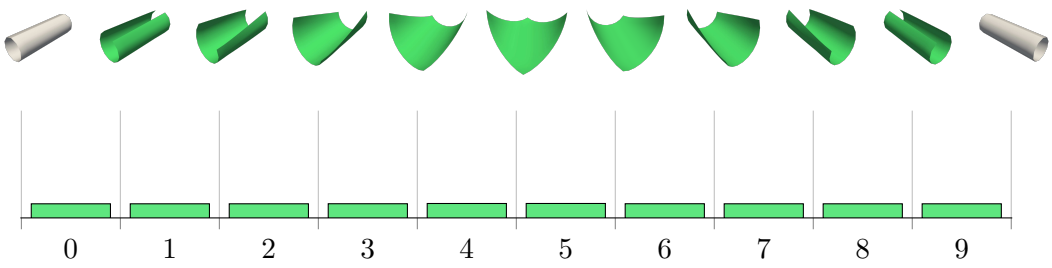


Figure 4.2: Discrete geodesic between two plates rolled up in different directions computed by solving (4.6) with the quadratic energy $\mathcal{W}_q$, and the energy profile with the same scale as in Figure 4.1.

Special case: Quadratic energy We obtain an even more efficient scheme by using the quadratic energy $\mathcal{W}_q$. In this case, the problem computing the geodesic (4.9) becomes a (weighted) quadratic optimization problem, whose unique solution is given by equidistant linear interpolation of the endpoints. Furthermore, as we have explained before, the projection can be implemented efficiently using the Gauss-Newton method. The downside of this approach is the possibility of artifacts due to missing penalization of compression of triangles, as we have seen for example in Figure 3.3. This typically becomes even more apparent when computing extrapolations that go beyond a certain magnitude. However, as one can witness in [HRS+16], this linear approximation already leads to visually pleasing results.

4.3 Principal Geodesic Analysis

As the last element of this thesis, we will study an application of statistical shape analysis to the space of lengths and angles $\mathcal{Z}$. Broadly speaking, our goal is to extract a model for *natural* deformations of the discrete surface from a set of example deformations. To provide an instructive example, we consider the shape of a human as a discrete surface and want to extract possible movements, e.g. only at the joints of that human. Principal Geodesic Analysis (PGA) [FJLP04] has been used previously on shape spaces [FJLP04, TWC+09, FB12, ZHRS15, vTAMZ18, HZRS18] to construct such models as low-dimensional submanifolds. When working with the space of discrete shells, one quickly faces the problem of rigid body motion invariance requiring adaption towards it, which increases computational cost. On the contrary, lengths and angles are inherently rigid body motion invariant and thus provide a potential way to simplify this application. In this section, we will briefly recapitulate Principal Geodesic Analysis, explain its application to the space of lengths and angles, look at an approximated version in the sense of the previous section, and consider an application of the resulting low-dimensional submanifold to model fitting via soft constraints.

Principal Component Analysis Let us briefly recall Principal Component Analysis (PCA) as a tool from the domain of linear statistics. For this, assume we are given samples $z^1, \dots, z^N \subset \mathbb{R}^{2|\mathcal{E}|}$. First, we determine their Euclidean mean $\bar{z} = \frac{1}{N}\sum_{i=1}^{N} z^i$ and center the samples by setting $\tilde{z}^i = z^i - \bar{z}$ for $i = 1, \dots, N$, which we also arrange in the data matrix $\tilde{Z} = (\tilde{z}^1, \dots, \tilde{z}^N) \in R^{N \times 2|\mathcal{E}|}$. Now, in geometric terms, we want to compute the J-dimensional subspace of $\mathbb{R}^{2|\mathcal{E}|}$, which approximates the samples as good as possible. If we think of the subspace in terms of an orthonormal basis

$U = (u_1, \dots, u_J) \in \mathbb{R}^{J \times 2|\mathcal{E}|}$, this can be expressed as the optimization problem

$$\begin{aligned} \underset{U \in \mathbb{R}^{J\times 2|\mathcal{E}|}}{\text{minimize}} \quad & \sum_{i=1}^{N} \left\| \tilde{z}^i - \sum_{j=1}^{J} \langle u_j, \tilde{z}^i \rangle \, u_j \right\|_2^2 . \\ \text{subject to} \quad & U^\intercal U = \mathrm{Id} \end{aligned} \tag{4.11}$$

It turns out [HV05] that this problem can be solved by computing the first J eigenpairs (λ_i, ϕ_i) in order of descending eigenvalue of the Gram's matrix $\tilde{Z}^\intercal \tilde{Z}$ and setting $u_i = \lambda_i^{-1} \tilde{Z} \phi_i$ for $i = 1, \dots, J$.

In statistics, the orthonormal basis $u_1, \dots, u_J$ is interpreted as the main directions of the variance in the data and in physics as principal axis of angular momentum, which also is the origin of the name Principal Component Analysis. It is a very important tool for dimensionality reduction, be it to reveal statistical structure in data or as a reduced order modeling technique to speed up complicated problems. In this case, it is also known as Proper Orthogonal Decomposition (POD) [HV05, vRESH16].

Principal Geodesic Analysis We will generalize the different notions necessary for PCA to nonlinear Riemannian manifolds. This generalization will be directly phrased using the space of lengths and angles $\mathcal{Z}$ and thus yield the general recipe on how to apply it in this context.

First off, is the notion of a mean, which we replace by a version using the squared distance, called the Fréchet mean [Fré48], or Riemannian center of mass [GK73], given as

$$\bar{z} := \arg\min_{z \in \mathcal{Z}} \sum_{i=1}^{N} \operatorname{dist}^2(z^i, z), \tag{4.12}$$

which might not be unique in general.

The notion of a J-dimensional subspace $U^J = \operatorname{span}(u_1, \dots, u_J)$ of $\mathbb{R}^{2|\mathcal{E}|}$ will be replaced by the notion of a J-dimensional submanifold $\mathcal{Z}^J \subset \mathcal{Z}$, and hence the task will become to determine the best-approximating submanifold $\mathcal{Z}^J$ of our data containing $\bar{z}$. We will compute an (approximate) solution of this problem by parametrizing submanifolds through subspaces of the tangent space $T_{\bar{z}}\mathcal{Z}$ at the mean, which yield submanifolds by application of the exponential map.

To compute such a subspace, we need tangent vectors corresponding to our input lengths and angles. These will be computed using the logarithm, which we have introduced in Section 2.1 as the inverse of the exponential, and thus we define

$$v^i := \log_{\bar{z}} z^i \in T_{\bar{z}}\mathcal{Z}, \text{ for } i = 1, \dots, N. \tag{4.13}$$

Therefore, we have reduced our problem to computing the optimal subspace of $T_{\bar{z}}\mathcal{Z}$ for the tangent vectors v^i. This can be done by applying linear PCA with respect to the Riemannian scalar product, i.e. as solution U^J of

$$\begin{aligned} \underset{U \in R^{J\times 2|\mathcal{E}|}}{\text{minimize}} \quad & \sum_{i=1}^{N} \left\| v^i - \sum_{j=1}^{J} g_{\bar{z}}(u_j, v^i) \, u_j \right\|_{T_{\bar{z}}\mathcal{Z}}^2 \\ \text{subject to} \quad & g_{\bar{z}}(u_k, u_l) = \delta_{kl} \text{ for all } k, l \in \{1, \dots, J\}. \end{aligned} \tag{4.14}$$

The resulting subspace U^J is in fact a subspace of the tangent space as its basis vectors $u_1, \dots, u_J$ are given as linear combinations of the tangent vectors v^i. From this, we

get our low-dimensional submanifold by applying the exponential map

$$\mathcal{Z}^J := \left\{ \exp \sum_{j=1}^{J} \alpha_j u_j \mid \alpha \in \mathbb{R}^J \right\}. \tag{4.15}$$

For a more detailed introduction to this technique, we refer to [FJLP04].

Remark (Discretization). A discretization of the Principal Geodesic Analysis on the space of lengths and angles can be realized using the tools from time-discrete geodesic calculus introduced in Section 2.2. Nevertheless, in this thesis, we will instead consider a simplified version of it.

Approximation This simplified version will be achieved by linearizing the different steps explained above and using the projection as we have done it in Section 4.2. In the following, we will introduce approximations that will be considered as definitions afterwards, thus by abuse of notation, we will denote them by equations.

Again, we will start with the mean, where we replace the Fréchet mean from before by the projection of the linear mean of the input length and angles, i.e. we set

$$\bar{z} = \mathcal{P}_{\mathcal{Z}} \left[\frac{1}{N} \sum_{i=1}^{N} z^i \right], \tag{4.16}$$

which is sometimes referred to as the *extrinsic mean* on $\mathcal{Z}$.

We replace the application of the logarithm from before by a simple linear projection onto the tangent space

$$v^i = \mathcal{P}^{\text{lin}}_{T_{\bar{z}}\mathcal{Z}}[z^i - \bar{z}] \in T_{\bar{z}}\mathcal{Z}, \text{ for } i = 1, \dots, N. \tag{4.17}$$

Furthermore, we also consider the variation, where we completely ignore the tangent space and set

$$v^i = z^i - \bar{z}, \text{ for } i = 1, \dots, N. \tag{4.18}$$

In the both cases, (4.14) remains untouched, which also means that in the latter case we essentially perform linear PCA in the ambient space $\mathbb{R}^{2|\mathcal{E}|}$, however with respect to the Riemannian metric g.

Finally, we obtain our submanifold by projecting elements from our subspace onto $\mathcal{Z}$, meaning that we have

$$\mathcal{Z}^J = \left\{ \mathcal{P}_{\mathcal{Z}}[\bar{z} + \sum_{j=1}^{J} \alpha_j u_j] \mid \alpha \in \mathbb{R}^J \right\}. \tag{4.19}$$

Overall, this leads to a computationally efficient scheme for computing a low-dimensional subspace/-manifold of lengths and angles. In the following, by abuse to notation, we will refer to the matrix and the subspace spanned by the rows both as U^J

4.3.1 Application: Model Fitting via Soft Constraints

As explained at the beginning, our motivation to construct a low-dimensional submanifold is to generate a model for natural deformations based on examples. The model can be used to analyze the deformations present in the data or, as we will proceed now, use it as a so-called *template model*. In this context, we are given external conditions and want to compute a deformation which is natural according to our model and

adheres to the external conditions as good as possible. This is a typical application for low-dimensional models in computer graphics, which can either be entirely handcrafted, such as skeleton models [HH95, LCF00], or be computed from example data [ASK+05, PMRMB15, HZRS18].

Marker We consider a special case of external conditions that we call *markers* or *landmarks*, where we are given a set of positions in three-dimensional space $\mathcal{X} = \{x_1, \dots, x_M\} \subset \mathbb{R}^3$, which indicate the wanted pose of our shape. Such *marker positions* are for example obtained using motion capture systems. Furthermore, we will assume that we already know correspondences between the vertices of our discrete surface and these markers, meaning that we have indices $i_1, \dots, i_M$ such that the marker position x_m corresponds to the position of the vertex v_{i_m}. Obtaining such correspondences is a nontrivial problem itself, which we choose to disregard here as we focus on the deformations. To tie these marker positions to the surface, or more correctly its embedding, we consider a *soft constraint* approach, where we aim to minimize the distances to the markers, which is opposite to a hard constraint approach setting them as fixed boundary conditions. Thus, we obtain the energy

$$\mathcal{D}_{\mathcal{X}}[X] := \sum_{m=1}^{M} \|X_{i_m} - x_m\|^2_{\mathbb{R}^3}. \tag{4.20}$$

Physically, this corresponds to applying forces to the different vertices in direction of the markers with their magnitude given by the distance to the marker.

Prior Now that we know the external conditions, we introduce a suitable way to incorporate our low-dimensional submanifold/-space of $\mathcal{Z}$ into the problem. Given our submanifold, we want to describe the distance between it and an embedding X, which we do by minimizing the energy $\mathcal{W}[Z[X], z]$ over all $z \in \mathcal{Z}^J$. In our simplified approach, we replace the submanifold by the subspace and thus obtain

$$\mathcal{F}[X] := \operatorname*{arg\,min}_{z \in \bar{z} + U^J} \mathcal{W}\left[Z[X],\ z\right]. \tag{4.21}$$

In statistics, one would call $\mathcal{F}$ a *prior* for our problem limiting the admissible solutions.

Note that in the case of the quadratic energy $\mathcal{W}_q$, the solution to the minimization problem of $\mathcal{F}$ is given by the orthogonal projection of $Z[X]$ onto $\bar{z} + U^J$ with respect to the (constant) Hessian of the energy. Precisely, we have

$$\mathcal{F}[X] := \mathcal{W}_q\left[Z[X],\ \bar{z} + U^J (U^J)^\intercal M (z[X] - \bar{z})\right], \tag{4.22}$$

where $M := \partial_2^2 \mathcal{W}_q[\bar{z}, \bar{z}]$. However, often we still include $z \in \bar{z} + U^J$ in our overall optimization as the orthogonal projection would lead to a densely populated Hessian, which is an issue for meshes with large resolutions.

Overall problem We combine the data term $\mathcal{D}$ with the prior $\mathcal{F}$ to obtain our overall problem

$$\begin{aligned} &\text{minimize} && \mathcal{J}_{\mathcal{X}}[X,\, z] := \mathcal{D}_{\mathcal{X}}[X] + \gamma\, \mathcal{W}\left[Z[X],\ z\right] \\ &\text{subject to} && X \in \mathbb{R}^{3|\mathcal{V}|},\ z \in \bar{z} + U^J, \end{aligned} \tag{4.23}$$

where γ is called the *prior weight*. This problem resembles the variational problems occurring in mathematical physics, where the term $\mathcal{D}$ corresponds to the external forces

integrated over the surface and the prior $\mathcal{F}$ to the internal energy dissipation. In other terms, this means that deformations on the manifold $\mathcal{Z}^J$ do not cost any energy and thus, in this context, we sometimes call $\mathcal{Z}^J$ the *rest manifold*.

Furthermore, the problem can be considered an extension of the one introduced in [FB11] by considering soft instead of hard constraints and by using a low-dimensional model built from the example data instead of using the examples directly.

In case of a time series of marker positions $(\mathcal{X}_t)_t$, one could explicitly add another term guaranteeing the smoothness of the results by penalizing the difference to the previous step. Nevertheless, in our implementation, we took a more implicit and simple approach by taking the previous solution as initialization of the optimization method.

Remark on implementation. To solve (4.23), we consider both an alternating and a joint optimization approach. In the alternating approach, we fix either X or z and optimize with respect to the other one. In case of the quadratic energy, optimizing with respect to z results in computing the orthogonal projection of $Z[X]$ onto $\bar{z} + U^J$ as remarked before, which means it can be done using two matrix multiplications, or one if we preassemble $U^J(U^J)^\intercal$. Then, optimizing with respect to X is essentially our energy-based reconstruction from before with soft constraints for which we can also use the Gauss-Newton approximation of the Hessian to obtain a computationally efficient approach.

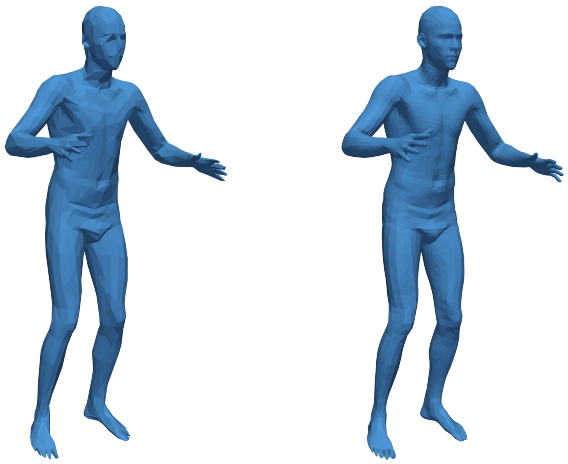
Typically, we initially employ multiple iterations of the alternating approach and then use the result as initialization for a joint optimization of $\mathcal{J}_\mathcal{X}[X, z]$ with respect to X and z. For this, we also found the Gauss-Newton approximation of the Hessian of the prior term to be beneficial in terms of runtimes and robustness. The improved initialization from the alternating approach also improved the convergence behavior of the joint optimization.

4.3.2 Numerical Experiments

We will apply above's variation of Principal Geodesic Analysis, along with the corresponding model fitting approach, to human body shapes and marker positions mimicking dance movements.

Data As in our experiments on the reconstruction of embeddings in Section 3.4, we used a selection of data from the Dyna dataset [PMRMB15]. From the shape 50009 (a human male), we used a selection of 29 poses that were also used in [HZRS18], which were fitted to 41 markers in a sequence from the Carnegie Mellon University Motion Capture Database. Originally, the shapes had 6890 vertices each, but as (4.23) is an optimization problem in nodal positions, we employed the multi-resolution scheme explained in Section 2.3 to reduce the dimension by 80 percent to 1378 vertices, which is still well-capable of capturing the deformation of interest. The coarse mesh is displayed in Figure 4.3 alongside its fine counterpart.

Parameters We used the quadratic energy $\mathcal{W}_q$ with $\eta = 0.005$ on the coarse mesh. Both, in the case of PGA, i.e. with projecting on the tangent space, and in the case of plain PCA in $\mathbb{R}^{2|\mathcal{E}|}$, we constructed a $J = 10$ dimensional subspace. For the marker fitting, we used a prior weight of $\gamma = 5.5$ in all cases.



(a) Coarse resolution (b) Original resolution

Figure 4.3: Projected linear mean of input data, computed in coarse and prolongated to original resolution.

Results on PGA In Figure 4.4, we show the first five principal modes constructed from the Dyna dataset using the simplified PGA (with linear projection on the tangent space) and PCA (in ambient space, without projection). The nodal positions were reconstructed along a line segment $z_j(t) := \bar{z} + t\,u_j$ at the mean in direction of the principal components in lengths and angles. In the case of PGA, we used a maximal factor of $t = \pm 10$ and in case of PCA a factor of $t = \pm 2.5$.

First off, we see that in both cases the principal modes represent motions an actual human could perform, which indicates that they indeed yield a useful model. The modes from PGA appear to be less dynamic than the ones constructed using PCA. This indicates that the linear projection onto the tangent space might lose information about the deformations present in the dataset. Furthermore, in some PGA modes, one can observe an undesirable deflation or inflation of the moving parts, see for example the hand in the fifth PGA mode in Figure 4.4.

Moreover, in the mode from PGA, the violation of integrability, especially the violation of the triangle inequality, is more severe and less localized, which hinders an application of the adaptive frame-based reconstruction. We have, however, observed that if we use the full-resolution data in the model construction, the frame-based reconstruction of the PGA modes is possible, as the higher number of faces allows to avoid the violations more easily.

Results on Marker Fitting First off, we see that with the PGA and PCA subspaces, the results improve comparing to only using the mean as prior, even though using PGA the improvements are small. Without the subspaces, the motions look rather stiff, which could be reduced by decreasing the bending or the prior weight, but this leads to unwanted deformations such as a compression of the leg or torso.

Next, we observe that the use of our simplified PGA approach decreases the quality of the results compared to the PCA approach. With the PGA subspace, the optimization failed more often to find a deformation suitable for the marker position and got stuck. This indicates that using the tangent space without the logarithm and exponential map might not be a good approximation of our space $\mathcal{Z}$ due to the strong deformations we are considering. Thus, the simple linear projection onto the tangent

Figure 4.4: First five modes of PCA (green) and simplified PGA (orange) in lengths and angles on Dyna.

space seems to lose information on deformations given in the examples.

Compared to the results of [HZRS18], we see that our approach is less capable generalizing to movements requiring deformations different from the ones observed in the dataset. For example, there are marker positions which require the shape to raise the left leg behind their body, whereas in the training set, there are barely any examples in which the left hip is moved backwards substantially. The completely nonlinear method in nodal positions from [HZRS18] is able to produce visually pleasing results, whereas our method produces results appearing a little stiff, nevertheless natural.[1]

Furthermore, we tried to use the whole Dyna dataset consisting of approximately 3,300 shapes for the model in the training of the model. However, we found that with the same dimensionality of the subspace, this decreased the quality of the results, which we assume to have multiple reasons. First, there are a lot of shapes in the dataset with rather small deformations, because the deformations were created capturing real human's movements. Furthermore, due to the scanning procedure, there are some shapes with artifacts included which could decrease the quality of the results, because, in general, PCA is sensitive to outliers. Lastly, numerous motions in the dataset were meant to show soft tissue movement, such as "jiggling", which are therefore not helpful for our task.

We also observed some numerical difficulties in our experiments due to triangles (nearly) violating the triangle inequality, This leads to exploding gradients of the quadratic energy on nodal positions, as the gradient of dihedral angles with respect to nodal positions grows rapidly. To avoid this unwanted compression of triangles, it might be interesting to consider the discrete shells energy on lengths and angles we have discussed before because it strongly penalizes this violation in the energy. Another way to reduce these issues could be to consider a different kind of coarsening

[1] Essentially we are saying that our dancer is less skilled after the same amount of training than the dancer from [HZRS18].

Example			Mean	Tangent space		PCA	
Model	$\lvert\mathcal{V}\rvert$	N		Decomposition	Solve	Gram's	EVD
Dyna	1.4k	29	1.3 s	9.2s	0.7 s	1.8 ms	2.5 ms
Dyna	1.4k	3288	1.3 s	9.2s	65 s	4.6 s	65 s
Dyna	6.9k	29	3.6 s	860s	8.5 s	5.8 ms	8.7 ms
Dyna	6.9k	3288	2.7 s	860s	900 s	16.4 s	65 s

Table 4.1: Runtimes of the identified steps in creating the subspace.

Example	Mean	Alternating	Joint	Prolongation	Total
Dyna (Mean)	557	–	–	20	577
Dyna (PCA, $J = 10$)	–	1630	1397	20	3047
Dyna (PGA, $J = 10$)	–	1403	1861	20	3284

Table 4.2: Average runtimes (in ms) of the identified steps in a model fitting step.

which takes them into account.

Lastly, let us note that the lengths and angles $z \in \bar{z} + U^J$ resulting from our optimization often do not exhibit the strong locality of the integrability violation we have seen in the separate modes or in Section 3.4.3. This prohibits the use of the adaptive frame-based reconstruction approach from Section 3.4.2 in the model fitting application.

Timings The creation of the subspace models contains three major steps that consume most of the runtime. The first step is computing the mean, which essentially means reconstructing a single set of lengths and angles. In the case of the simplified PGA, we have to compute the linear projection onto the tangent space, for which we use the approach without computing an explicit basis as explained before. Hence, this step has two substeps. First, we factorize $(D_{\bar{z}}\widetilde{\mathcal{I}})(D_{\bar{z}}\widetilde{\mathcal{I}})^\intercal$ using the Cholesky decomposition and then compute the projection of each input shape, which includes solving the factorized linear system. As the last step in both cases, we have to solve the PCA problem, which includes computing the Gram's matrix $\tilde{Z}^\intercal\tilde{Z}$ and its eigendecomposition. We always compute the full eigendecomposition, hence its runtime is independent of the subspace dimension J.

There are two or three steps in the model fitting application, depending on whether we use a subspace or only the mean as prior. In both cases, we have the prolongation of the result from the coarse to the fine mesh. This result is computed by solving a single optimization of (4.23) with fixed $z = \bar{z}$. In case of a subspace prior, we first run multiple iterations of the alternating optimization and then the joint optimization.

Discussion Overall, we have introduced a data-driven approach for creating a template model of surface deformations using PGA or PCA in the space of lengths and angles. We have used this template model in a simple model fitting approach with marker positions acting as soft constraints or forces on the surfaces. This yielded naturally looking deformations of human shapes when fitting to markers of a dance sequence. Compared to other methods, however, it exhibited some shortcomings in the capability to generalize to deformations not present enough in the training data.

We expect that the template model would benefit from implementing more nonlinear tools in the space of lengths and angles. Using the logarithm and exponential-map

might allow for strong deformations and potentially close the gap in generalization capability to [HZRS18]. Also, adaptations of PCA might result in improved models, for example, Sparse PCA which enforces a certain sparsity of the modes, because as we have explained before, this sparsity is a natural concept for our applications.

Furthermore, it might be helpful for the model fitting application to consider different data terms than the one used above which required exact correspondences between vertices and markers. These are currently created manually and are thus not always optimal. Therefore, it would be interesting to remove the need for strict correspondences and instead, for example, consider the distance of the marker to a patch of the surface. In general, this is related to the question of how to couple the lengths and angles model with extrinsic notions such as forces acting on the surface.

To achieve realtime capability with this approach, especially if we were to add more nonlinearity, it would be interesting to investigate reduced order modeling approaches to speed up the computations. Subspace methods as for example used in [vRESH16] are potentially easier to apply to lengths and angles as they are rigid body motion invariant. This would allow us to handle the nonlinearity in lengths and angles efficiently, but for the model fitting application, we still need to work in nodal positions. Thus, it would be interesting to develop a coupled reduced approach for both $\mathcal{Z}$ and nodal positions.

Figure 4.5: Model fitted to shown markers using only the mean (blue), the PCA subspace (green), and the simplified PGA subspace (orange) as prior.

Chapter 5

Conclusion

Summary

In this thesis, we studied edge lengths and dihedral angles as primal degrees of freedom for deformations of a discrete surface and eventually used this approach to perform statistical analysis on those deformations. The reconstruction of immersions from lengths and angles and the application to statistical analysis were investigated numerically on an example dataset of human shapes.

Describing edge lengths and dihedral angles as discrete counterparts to the first and second fundamental form allowed us to develop a comprehensive understanding of them. Following [WLT12], we extensively studied which lengths and angles are actually realizable by an immersion of the discrete surface in $\mathbb{R}^3$. This led to a description of the set of admissible lengths and angles $\mathcal{Z}$ by a set of equations, which are akin to the Gauß-Codazzi equations of continuous surfaces, as they ensure that we can integrate the local change of geometry induced by the lengths and angles to obtain an immersion. Together, this constituted the discrete fundamental theorem of surfaces from [WLT12]. Moreover, we adopted the deformation energies previously used on nodal positions to generate physically plausible deformations. This turned out to be a straightforward task, which highlights that lengths and angles are in fact natural degrees of freedom for our applications.

Eventually, we were typically interested in the immersion corresponding to a set of lengths and angles, which led to our investigation of various approaches for constructing them. The proof of the aforementioned discrete fundamental theorem yielded a way to construct the immersion iteratively starting from a single triangle if the target lengths and angles are admissible. However, we observed that this algorithm leads to severe visual artifacts if the lengths and angles violate the integrability, even if this violation is often highly localized. Based on the latter observation, we developed a novel variation of this algorithm which adapts the order of integration to avoid the vertices with violated integrability as much as possible. Nevertheless, sometimes the constructed immersion still needs to be post-processed or the violation is not localized prohibiting the application of the above algorithm. For these cases, we also considered an energy-based reconstruction introduced in [FB11]. The methods were compared in numerical experiments on different examples.

With this comprehensive toolbox for lengths and angles at hand, we introduced a manifold structure on the admissible lengths and angles through the conditions given in the discrete fundamental theorem. This description of the nonlinear structure has the advantage that we are capable of explicitly computing the tangent space and use

it in applications, which was previously not feasible as it required dealing with equivalence classes. In this context, we introduced a simplified version of Principal Geodesic Analysis on the space of lengths and angles and used the resulting low-dimensional models in an application to the problem of fitting a model to sparse marker positions.

Outlook

This thesis offers a profound amount of possibilities for future work. First of all, only the integrability conditions for simply connected surfaces are present in our current implementation, which should be extended to the general case.

Furthermore, we currently post-process the results of the adaptive frame-based reconstruction, if necessary, by a small number of Gauß-Newton iterations. While this works well, it is undesirable from the perspective of runtimes. Hence, to enable the use of the reconstruction in interactive applications, we need to speed up the post-processing. A first step could be the replacement of the linear solver in the implementation of the Gauß-Newton method as it takes up most of the runtime. Another idea would be to investigate the least-squares relaxation proposed in [WLT12].

In light of our model fitting, we suspected that it would benefit from using the nonlinear tools of geodesic calculus on the space of lengths and angles. Currently, they are only available for discrete surfaces with a fairly small number of vertices, while for surfaces with a higher resolution we run into problems with the line-search optimization due to indefinite Hessian matrices. Thus, an important implementation task would be to realize a trust-region solver for constrained problems in our framework.

Such a solver would also allow inspecting the application of lengths and angles to problems in which constraints on geometric properties of the surface arise. One example is constraints on the Gaussian curvature of the surface, which is an intrinsic property and can thus be computed from the edge lengths only. Such constraints occur when dealing with developable surfaces, which have constant zero Gaussian curvature, or hyperbolic surfaces with a constant negative Gaussian curvature.

In Section 4.3.1, we explained that the marker positions essentially act as forces on the surfaces. This raised the modeling question of how to couple lengths and angles with extrinsic notions, such as mentioned forces. A first idea is to use the frame-based reconstruction with a fixed spanning tree, possibly together with a rigid registration, and consider it a differentiable map. To use this in an optimization problem would require computing the derivatives of this map, which is a difficult undertaking but might be a feasible application of modern automatic differentiation frameworks.

While our approach already improves the runtimes compared to [HZRS18], there is still a considerable gap to real-time capability. To close it, especially if we were to add more nonlinear problems in $\mathcal{Z}$, it is necessary to investigate reduced order modeling techniques. For computations entirely in lengths and angles, subspace approaches might be a viable tool as our degrees of freedom are rigid body motion invariant. However, if we consider applications with coupled nodal positions, such as the model fitting, this would require an approach for the nodal positions as well. Hence, it would be interesting to develop a combined approach.

Finally, there are also open theoretic problems on the space of lengths and angles. The first is proving that our manifold structure is actually well-defined, meaning that we have a constant dimension, which is the content of Conjecture 1. Furthermore, it would be interesting to develop a better understanding of the structure of $\mathcal{Z}$, which might be helpful for future optimization problems and algorithms.

Bibliography

[AR18] Nina Amenta and Carlos Rojas. Dihedral Rigidity and Deformation. *Canadian Computational Geometry Conference*, 2018.

[ASK+05] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. SCAPE: Shape Completion and Animation of People. *ACM Transactions on Graphics*, 24(3):408–416, 2005.

[BEKB15] Davide Boscaini, Davide Eynard, Drosos Kourounis, and Michael M. Bronstein. Shape-from-Operator: Recovering Shapes from Intrinsic Operators. In *Computer Graphics Forum*, volume 34, pages 265–274. John Wiley & Sons, Ltd (10.1111), May 2015.

[BER15] Benjamin Berkels, Alexander Effland, and Martin Rumpf. Time Discrete Geodesic Paths in the Space of Images. *SIAM Journal on Imaging Sciences*, 8(3):1457–1488, 2015.

[BLL15] Seung Yeob Baek, Jeonghun Lim, and Kunwoo Lee. Isometric shape interpolation. *Computers and Graphics (Pergamon)*, 46:257–263, feb 2015.

[BS08] Mario Botsch and Olga Sorkine. On Linear Variational Surface Deformation Methods. *IEEE Transactions On Visualization and Computer Graphics*, 14(1):213–230, 2008.

[BSPG06] Mario Botsch, Robert W Sumner, Mark Pauly, and Markus Gross. Deformation Transfer for Detail-Preserving Surface Editing. In *Vision, Modeling and Visualization*, pages 357–364, 2006.

[CdGDS13] Keenan Crane, Fernando de Goes, Mathieu Desbrun, and Peter Schröder. Digital Geometry Processing with Discrete Exterior Calculus. In *ACM SIGGRAPH 2013 courses*, SIGGRAPH '13, New York, NY, USA, 2013. ACM.

[CDHR08] Yanqing Chen, Timothy A Davis, William W Hager, and Sivasankaran Rajamanickam. Algorithm 887: CHOLMOD, Supernodal Sparse Cholesky Factorization and Update/Downdate. *ACM Transactions on Mathematical Software*, 35(3):22:1–22:14, 2008.

[CDS10] Keenan Crane, Mathieu Desbrun, and Peter Schröder. Trivial Connections on Discrete Surfaces. *Computer Graphics Forum (SGP)*, 29(5):1525–1533, 2010.

[Cia88] Philippe G Ciarlet. *Mathematical elasticity. Vol. I*, volume 20 of *Studies in Mathematics and its Applications*. North-Holland Publishing Co., Amsterdam, 1988.

[CKPS18] Albert Chern, Felix Knöppel, Ulrich Pinkall, and Peter Schröder. Shape from metric. *ACM Transactions on Graphics*, 37(4):1–17, jul 2018.

[CLR04] Ulrich Clarenz, Nathan Litke, and Martin Rumpf. Axioms and variational problems in surface parameterization. *Computer Aided Geometric Design*, 21(8):727–749, 2004.

[Dav04] Timothy A Davis. Algorithm 832: UMFPACK V4.3—an Unsymmetric-pattern Multifrontal Method. *ACM Transactions on Mathematical Software*, 30(2):196–199, 2004.

[dC76] Manfredo Perdigão do Carmo. *Differential geometry of curves and surfaces*. Prentice Hall, 1976.

[dC92] Manfredo Perdigão do Carmo. *Riemannian Geometry*. Mathematics (Boston, Mass.). Birkhäuser, 1992.

[DKT08] Mathieu Desbrun, Eva Kanso, and Yiying Tong. Discrete differential forms for computational modeling. In *Oberwolfach Seminars: Discrete differential geometry*, volume 38 of *Oberwolfach Seminars*, pages 287–324. Birkhäuser, Basel, 2008.

[Epp03] David Eppstein. Dynamic Generators of Topologically Embedded Graphs. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '03, pages 599–608, Philadelphia, PA, USA, 2003. Society for Industrial and Applied Mathematics.

[FB11] Stefan Fröhlich and Mario Botsch. Example-Driven Deformations Based on Discrete Shells. *Computer Graphics Forum*, 30(8):2246–2257, 2011.

[FB12] Oren Freifeld and Michael J Black. Lie Bodies: A Manifold Representation of 3D Human Shape. In *European Conference on Computer Vision (ECCV)*, Part I, LNCS 7572, pages 1–14. Springer-Verlag, 2012.

[FJGM03] Gero Friesecke, Richard D James, Maria Giovanna Mora, and Stefan Müller. Derivation of nonlinear bending theory for shells from three-dimensional nonlinear elasticity by Gamma-convergence. *Comptes Rendus Mathematique*, 336(8):697–702, 2003.

[FJLP04] Fletcher P.T., Joshi S., Lu C., and Stephen M Pizer. Principal Geodesic Analysis for the Study of Nonlinear Statistics of Shape. *IEEE Transactions on Medical Imaging*, 23(8):995–1005, 2004.

[FJM02] Gero Friesecke, Richard D James, and Stefan Müller. A theorem on geometric rigidity and the derivation of nonlinear plate theory from three-dimensional elasticity. *Communications on Pure and Applied Mathematics*, 55(11):1461–1506, 2002.

[Fré48] Maurice Fréchet. Les éléments aléatoires de nature quelconque dans un espace distancié. In *Annales de l'institut Henri Poincaré*, volume 10, pages 215–310, 1948.

[GH05] Michael Garland and Paul S Heckbert. Surface simplification using quadric error metrics. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '97, pages 209–216, New York, NY, USA, 2005. ACM Press/Addison-Wesley Publishing Co.

[GHDS03] Eitan Grinspun, Anil N Hirani, Mathieu Desbrun, and Peter Schröder. Discrete Shells. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '03, pages 62–67, Aire-la-Ville, Switzerland, Switzerland, 2003. Eurographics Association.

[GJO10] Gaël Guennebaud, Benoît Jacob, and Others. Eigen v3. http://eigen.tuxfamily.org, 2010.

[GK73] Karsten Grove and Hermann Karcher. How to conjugate C1-close group actions. *Mathematische Zeitschrift*, 132(1):11–20, 1973.

[Hee16] Behrend Heeren. *Numerical Methods in Shape Spaces and Optimal Branching Patterns.* PhD thesis, Rheinischen Friedrich-Wilhelms-Universität Bonn, 2016.

[Hee17] Behrend Heeren. Selected Topics in Science and Technology: Geometry Processing and Discrete Shells. `https://ins.uni-bonn.de/teachings/ss-2017-175-v5e2-selected-topics/`, 2017.

[HH95] Tony Heap and David Hogg. Extending the Point Distribution Model using polar coordinates. In Václav Hlaváč and Radim Šára, editors, *Computer Analysis of Images and Patterns*, pages 130–137, Berlin, Heidelberg, 1995. Springer Berlin Heidelberg.

[HRS+14] Behrend Heeren, Martin Rumpf, Peter Schröder, Max Wardetzky, and Benedikt Wirth. Exploring the Geometry of the Space of Shells. *Computer Graphics Forum*, 33(5):247–256, 2014.

[HRS+16] Behrend Heeren, Martin Rumpf, Peter Schröder, Max Wardetzky, and Benedikt Wirth. Splines in the Space of Shells. *Computer Graphics Forum*, 35(5):111–120, 2016.

[HRWW12] Behrend Heeren, Martin Rumpf, Max Wardetzky, and Benedikt Wirth. Time-Discrete Geodesics in the Space of Shells. *Computer Graphics Forum*, 31(5):1755–1764, 2012.

[HV05] Michael Hinze and Stefan Volkwein. Proper Orthogonal Decomposition Surrogate Models for Nonlinear Dynamical Systems: Error Estimates and Suboptimal Control. In Peter Benner, Danny C Sorensen, and Volker Mehrmann, editors, *Dimension Reduction of Large-Scale Systems*, pages 261–306, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

[HZRS18] Behrend Heeren, Chao Zhang, Martin Rumpf, and William Smith. Principal Geodesic Analysis in the Space of Discrete Shells. *Computer Graphics Forum*, 37(5):173–184, 2018.

[Kah14] William Kahan. Miscalculating Area and Angles of a Needle-like Triangle (from Lecture Notes for Introductory Numerical Analysis Classes). `http://http.cs.berkeley.edu/~wkahan/Triangle.pdf`, 2014.

[Ken84] David G Kendall. Shape Manifolds, Procrustean Metrics, and Complex Projective Spaces. *Bulletin of the London Mathematical Society*, 16(2):81–121, 1984.

[KMP07] Martin Kilian, Niloy J Mitra, and Helmut Pottmann. Geometric modeling in shape space. *ACM Transactions on Graphics*, 26(3):64, 2007.

[Küh15] Wolfgang Kühnel. *Differential Geometry: Curves – Surfaces – Manifolds*. Student Mathematical Library. American Mathematical Society, 2015.

[Lan95] Serge Lang. *Differential and Riemannian Manifolds*, volume 160 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, third edition, 1995.

[LCF00] J P Lewis, Matt Cordner, and Nickson Fong. Pose Space Deformation: A Unified Approach to Shape Interpolation and Skeleton-driven Deformation. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '00, pages 165–172, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co.

[LDRS05] Nathan Litke, Marc Droske, Martin Rumpf, and Peter Schröder. An Image Processing Approach to Surface Matching. In *Proceedings of the Third Eurographics Symposium on Geometry Processing*, SGP '05, Aire-la-Ville, Switzerland, Switzerland, 2005. Eurographics Association.

[LR95] Hervé Le Dret and Annie Raoult. The nonlinear membrane model as a variational limit of nonlinear three-dimensional elasticity. *Journal de Mathématiques Pures et Appliquées*, 73(6):549–578, 1995.

[LR96] Hervé Le Dret and Annie Raoult. The Membrane Shell Model in Nonlinear Elasticity: A Variational Asymptotic Derivation. *Journal of Nonlinear Science*, 6(1):59–84, 1996.

[LSLCO05] Yaron Lipman, Olga Sorkine, David Levin, and Daniel Cohen-Or. Linear rotation-invariant coordinates for meshes. *ACM Transactions on Graphics*, 24(3):479, 2005.

[LWLL14] Zheng Liu, Wei-ming Wang, Xiu-ping Liu, and Li-gang Liu. Scale-aware shape manipulation. *Journal of Zhejiang University SCIENCE C*, 15(9):764–775, sep 2014.

[MG03] Alex Mohr and Michael Gleicher. Deformation sensitive decimation. Technical report, University of Wisconsin, Madison, 2003.

[MRSS15] Jan Maas, Martin Rumpf, Carola Schönlieb, and Stefan Simon. A generalized Model for Optimal Transport of Images including Dissipation and Density Modulation. *ESAIM: Mathematical Modelling and Numerical Analysis*, 49(6):1745–1769, 2015.

[NW06] Jorge Nocedal and Stephen J Wright. *Numerical Optimization*. Springer, New York, NY, USA, second edition, 2006.

[Pal03] Richard Palais. Lecture notes on the Differential Geometry of Curves and Surfaces. `http://vmm.math.uci.edu/math32/`, 2003.

[PMRMB15] Gerard Pons-Moll, Javier Romero, Naureen Mahmood, and Michael J Black. Dyna: A Model of Dynamic Human Shape in Motion. *ACM Transactions on Graphics*, 34(4):120:1–120:14, 2015.

[PP12] Kaare Brandt Petersen and Michael Syskind Pedersen. The Matrix Cookbook. `http://www2.imm.dtu.dk/pubdb/p.php?3274`, November 2012.

[RE55] Ronald Samuel Rivlin and Jerald Laverne Ericksen. Stress-deformation relations for isotropic materials. *Journal of Rational Mechanics and Analysis*, 4:323–425, 1955.

[RHSH18] Michael Rabinovich, Tim Hoffmann, and Olga Sorkine-Hornung. Discrete Geodesic Nets for Modeling Developable Surfaces. *ACM Transactions on Graphics*, 37(2), 2018.

[RW13] Martin Rumpf and Benedikt Wirth. Discrete Geodesic Calculus in Shape Space and Applications in the Space of Viscous Fluidic Objects. *SIAM Journal on Imaging Sciences*, 6(4):2581–2602, 2013.

[RW15] Martin Rumpf and Benedikt Wirth. Variational time discretization of geodesic calculus. *IMA Journal of Numerical Analysis*, 35(3):1011–1046, 2015.

[SDL18] Stefan C. Schonsheck, Bin Dong, and Rongjie Lai. Parallel Transport Convolution: A New Tool for Convolutional Neural Networks on Manifolds. arXiv:1805.07857, 2018.

[SGC18] Oded Stein, Eitan Grinspun, and Keenan Crane. Developability of triangle meshes. *ACM Transactions on Graphics*, 37(4):1–14, 2018.

[SPSH18] Christian Schüller, Roi Poranne, and Olga Sorkine-Hornung. Shape Representation by Zippables. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH)*, 37(4), 2018.

[TWC+09] M Tournier, X Wu, N Courty, E Arnaud, and L Revéret. Motion Compression using Principal Geodesics Analysis. *Computer Graphics Forum*, 28(2):355–364, 2009.

[vRESH16] Philipp von Radziewsky, Elmar Eisemann, Hans-Peter Seidel, and Klaus Hildebrandt. Optimized subspaces for deformation-based modeling and shape interpolation. *Computers & Graphics*, 58:128–138, 2016.

[vTAMZ18] Christoph von Tycowicz, Felix Ambellan, Anirban Mukhopadhyay, and Stefan Zachow. An efficient Riemannian statistical shape model using differential coordinates: With application to the classification of data from the Osteoarthritis Initiative. *Medical Image Analysis*, 43:1–9, 2018.

[WB06] Andreas Wächter and Lorenz T Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25–57, 2006.

[WBRS09] Benedikt Wirth, Leah Bar, Martin Rumpf, and Guillermo Sapiro. Geodesics in Shape Space via Variational Time Discretization. In *Proc. of International Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition*, volume 5681 of *Lecture Notes in Computer Science*, pages 288–302, 2009.

[WDAH10] T. Winkler, J. Drieseberg, M. Alexa, and K. Hormann. Multi-scale geometry interpolation. *Computer Graphics Forum*, 29(2):309–318, 2010.

[Wir10] Benedikt Wirth. *Variational Methods in Shape Space*. PhD thesis, Rheinische Friedrich-Wilhelms-Universität Bonn, 2010.

[WLT12] Yuanzhen Wang, Beibei Liu, and Yiying Tong. Linear surface reconstruction from discrete fundamental forms on triangle meshes. *Computer Graphics Forum*, 31(8):2277–2287, 2012.

[Wol] Wolfram Research Inc. Mathematica, Version 11.0.

[ZHRS15] Chao Zhang, Behrend Heeren, Martin Rumpf, and William A.P. Smith. Shell PCA: Statistical shape modelling in shell space. In *Proceedings of the IEEE International Conference on Computer Vision*, volume 2015 Inter, pages 1671–1679. IEEE, 2015.

Appendix A

Derivatives

A.1 Trigonometric Formulas

Heron's formula

In Section 3.1, we introduced Heron's formula to compute the area of a triangle from its edge lengths a, b, c as

$$A(a,b,c) := \sqrt{s(s-a)(s-b)(s-c)}, \tag{A.1}$$

where $s = \frac{a+b+c}{2}$.

Stable formula Due to rounding errors, the formula (A.1) is inaccurate for needle-like triangles, i.e. those for which two edge lengths add up to only a little more than the third. In this case, s can be almost as big as c in floating-point arithmetic and thus $(s-a)$ and $(s-b)$ can be inaccurate [Kah14]. To circumvent this, we first sort the edge lengths such that $a \geq b \geq c$ and then set

$$A(a,b,c) = \frac{1}{4}\sqrt{(a+(b+c))(c-(a-b))(c+(a-b))(a+(b-c))}, \tag{A.2}$$

where the parenthesis indicate the necessary order of evaluation.

Derivatives For the first partial derivatives, we get

$$\begin{aligned}
\partial_a A(a,b,c) &= \frac{-a(a^2-b^2-c^2)}{8A(a,b,c)}, \\
\partial_b A(a,b,c) &= \frac{-b(a^2-b^2-c^2)}{8A(a,b,c)}, \\
\partial_c A(a,b,c) &= \frac{c(a^2+b^2-c^2)}{8A(a,b,c)}.
\end{aligned}$$

Law of cosines

We also introduced a formula for the interior angle of a triangle opposite to an edge with length c as

$$\gamma_c(a,b,c) := \arccos\left(\frac{a^2+b^2-c^2}{2ab}\right). \tag{A.4}$$

Stable formula Again, for needle-like triangles this formula suffers from numerical inaccuracies and we thus define a more stable one in the following way, adopted from [Kah14]. First, we define a helper variable μ by

$$\mu(a,b,c) := \begin{cases} c-(a-b) & b \geq c \geq 0 \\ b-(a-c) & c > b \geq 0 \\ \infty & \text{else} \end{cases}, \tag{A.5}$$

where in the last case a, b, c do not define valid edge lengths of a triangle. Now, we compute the interior angle by

$$\gamma_c(a,b,c) := 2\arctan\left(\sqrt{\frac{((a-b)+c)\mu(a,b,c)}{(a+(b+c))((a-c)+b)}}\right), \tag{A.6}$$

where for the special case that the denominator is zero and x positive, we define $\arctan(x/0) = \frac{\pi}{2}$.

Derivatives For the first partial derivatives, we get

$$\begin{aligned} \partial_a \gamma_c(a,b,c) &= \frac{-a^2+b^2-c^2}{4a\,A(a,b,c)}, \\ \partial_b \gamma_c(a,b,c) &= \frac{a^2-b^2-c^2}{4b\,A(a,b,c)}, \\ \partial_c \gamma_c(a,b,c) &= \frac{2c}{4\,A(a,b,c)}, \end{aligned}$$

assuming that a, b, and c are positive.

A.2 Discrete Integrability Conditions

In Section 3.2.2, we discussed the conditions for lengths and angles to be admissible and used them later in Chapter 4 to define the space of lengths and angles and formulate optimization problems in it. For this, we need the derivatives of the integrability conditions which we will derive now. Recall, the integrability map was given as $\mathcal{I}(z) = (\mathcal{I}_v(z))_{v\in\mathcal{V}} \colon \mathbb{R}^{2|\mathcal{E}|} \to (\mathbb{R}^{3\times 3})^{|\mathcal{V}|}$, where the individual components were defined as

$$\mathcal{I}_v(z) = \prod_{i=0}^{n-1} R_{i,(i+1) \bmod n}(z) \tag{A.8}$$

for the n-ring of faces $\mathcal{N}_v = \{f_0, \ldots, f_{n-1}\}$ around v and induced transition rotations $R_{ij}(z)$. We will ignore the triangle inequalities, as they define an open subset of $\mathbb{R}^{2|\mathcal{E}|}$ and thus we can calculate the derivatives at points fulfilling them. Furthermore, recall that we made a specific choice of frames for each triangle which simplified the induced transition rotations to

$$R_{ij}(z) := R_x(\theta_e) R_z(\gamma_{j,v}), \tag{A.9}$$

where e is the common edge of triangles f_i and f_j and $\gamma_{j,v}$ is the interior angle in f_j at v. It is given by the edge lengths as

$$\gamma_{j,v} = \arccos\left(\frac{l_a^2 + l_b^2 - l_c^2}{2 l_a l_b}\right), \tag{A.10}$$

where $c \in \mathcal{E}$ is the edge opposite to $\gamma_{j,v}$ and $a, b \in \mathcal{E}$ are the other two edges of f_j. Lastly, we also considered the concatenation with Euler angles

$$\widetilde{\mathcal{I}}_v(z) = (\arcsin \mathcal{I}_v(z)_{31},\, \operatorname{atan2}(\mathcal{I}_v(z)_{32}, \mathcal{I}_v(z)_{33}),\, \operatorname{atan2}(\mathcal{I}_v(z)_{21}, \mathcal{I}_v(z)_{11})), \tag{A.11}$$

which will require an additional application of the chain rule.

Structure

What makes it tractable to compute the derivatives explicitly is the structure of the problem. For sure, we can compute the derivative for each component $\mathcal{I}_v \colon \mathbb{R}^{2|\mathcal{E}|} \to \mathbb{R}^{3\times 3}$ separately, which yields a sparse tensor-valued differential $D_z\mathcal{I}_v \in \mathbb{R}^{3\times 3\times 2|\mathcal{E}|}$. We denote our vector of lengths and angles by $z = (l_1, \dots, l_{|\mathcal{E}|}, \theta_1, \dots, \theta_{|\mathcal{E}|}) = (z_1, \dots, z_{2|\mathcal{E}|})$ and thus the differential consists of partial derivatives

$$D_z\mathcal{I}_v = \left(\partial_{z_1}\mathcal{I}_v(z), \dots, \partial_{z_{2|\mathcal{E}|}}\mathcal{I}_v(z)\right). \tag{A.12}$$

Moreover, we can also assemble a second-order differential from partial second derivatives by

$$D_z\mathcal{I}_v = \left(\partial_{z_l}\partial_{z_k}\mathcal{I}_v(z)\right)_{kl} \in \mathbb{R}^{3\times 3\times 2|\mathcal{E}|\times 2|\mathcal{E}|}. \tag{A.13}$$

For each partial derivative, we can apply the product rule for matrices (cf. [PP12]) to get

$$\begin{aligned}
\partial_{z_k}\mathcal{I}_v(z) &= \partial_{z_k}\left(\prod_{i=0}^{n-1} R_{i,(i+1) \bmod n}(z)\right) \\
&= \sum_{j=0}^{n-1} R_{01}(z) \cdot \ldots \cdot \partial_{z_k} R_{j,j+1}(z) \cdot \ldots \cdot R_{n-1,0}(z),
\end{aligned}$$

where we will compute the individual local first derivatives $\partial_{z_k} R_{j,j+1}(z)$ explicitly, later. The partial second derivatives are given by another application of the product rule and consists of mixed first-order and second-order terms

$$\begin{aligned}
\partial_{z_l}\partial_{z_k}\mathcal{I}_v(z) &= \partial_{z_k}\left(\prod_{i=0}^{n-1} R_{i,(i+1) \bmod n}(z)\right) \\
&= \sum_{j=0}^{n-1} R_{01}(z) \cdot \ldots \cdot \partial_{z_l}\partial_{z_k} R_{j,j+1}(z) \cdot \ldots \cdot R_{n-1,0}(z) \\
&\quad + \sum_{\substack{i,j=0 \\ i\neq j}}^{n-1} R_{01}(z) \cdot \ldots \cdot \partial_{z_l} R_{i,i+1}(z) \cdot \ldots \cdot \partial_{z_k} R_{j,j+1}(z) \cdot \ldots \cdot R_{n-1,0}(z),
\end{aligned}$$

where we will again compute the local second derivatives below.

Eventually, by the chain rule, we obtain

$$\begin{aligned}
\partial_{z_k}\widetilde{\mathcal{I}}_v(z)_1 &= \frac{(\partial_{z_k}\mathcal{I}_v(z))_{31}}{\sqrt{1-\mathcal{I}_v(z)_{31}^2}}, \\
\partial_{z_k}\widetilde{\mathcal{I}}_v(z)_2 &= \frac{\mathcal{I}_v(z)_{32}\ \partial_{z_k}\mathcal{I}_v(z)_{33} - \partial_{z_k}\mathcal{I}_v(z)_{32}\ \mathcal{I}_v(z)_{33}}{\mathcal{I}_v(z)_{32}^2 + \mathcal{I}_v(z)_{33}^2}, \\
\partial_{z_k}\widetilde{\mathcal{I}}_v(z)_3 &= \frac{\mathcal{I}_v(z)_{21}\ \partial_{z_k}\mathcal{I}_v(z)_{11} - \partial_{z_k}\mathcal{I}_v(z)_{21}\ \mathcal{I}_v(z)_{11}}{\mathcal{I}_v(z)_{21}^2 + \mathcal{I}_v(z)_{11}^2},
\end{aligned}$$

and thus for the second derivatives

$$\partial_{z_l}\partial_{z_k}\widetilde{\mathcal{I}}_v(z)_1 = \frac{\mathcal{I}_v(z)_{31}\ \partial_{z_k}\mathcal{I}_v(z)_{31}\ \partial_{z_l}\mathcal{I}_v(z)_{31} + (1-\mathcal{I}_v(z)_{31}^2)\ \partial_{z_l}\partial_{z_k}\mathcal{I}_v(z)_{31}}{(1-\mathcal{I}_v(z)_{31}^2)^{3/2}},$$

and

$$\begin{aligned}\partial_{z_l}\partial_{z_k}\widetilde{\mathcal{I}}_v(z)_2 = &\frac{\partial_{z_k}\mathcal{I}_v(z)_{32}\ \partial_{z_l}\mathcal{I}_v(z)_{33} - \partial_{z_l}\mathcal{I}_v(z)_{32}\ \partial_{z_k}\mathcal{I}_v(z)_{33}}{\mathcal{I}_v(z)_{32}^2+\mathcal{I}_v(z)_{33}^2}\\ &+\frac{\partial_{z_l}\partial_{z_k}\mathcal{I}_v(z)_{32}\ \mathcal{I}_v(z)_{33} - \mathcal{I}_v(z)_{32}\ \partial_{z_l}\partial_{z_k}\mathcal{I}_v(z)_{33}}{\mathcal{I}_v(z)_{32}^2+\mathcal{I}_v(z)_{33}^2}\\ &-\frac{(2\,\mathcal{I}_v(z)_{32}\ \partial_{z_l}\mathcal{I}_v(z)_{32} + 2\,\mathcal{I}_v(z)_{33}\ \partial_{z_l}\mathcal{I}_v(z)_{33})}{\mathcal{I}_v(z)_{32}^2+\mathcal{I}_v(z)_{33}^2}\\ &\cdot\frac{(\partial_{z_k}\mathcal{I}_v(z)_{32}\ \mathcal{I}_v(z)_{33} - \mathcal{I}_v(z)_{32}\ \partial_{z_k}\mathcal{I}_v(z)_{33})}{\mathcal{I}_v(z)_{32}^2+\mathcal{I}_v(z)_{33}^2},\end{aligned}$$

and

$$\begin{aligned}\partial_{z_l}\partial_{z_k}\widetilde{\mathcal{I}}_v(z)_3 = &\frac{\partial_{z_k}\mathcal{I}_v(z)_{21}\ \partial_{z_l}\mathcal{I}_v(z)_{11} - \partial_{z_l}\mathcal{I}_v(z)_{21}\ \partial_{z_k}\mathcal{I}_v(z)_{11}}{\mathcal{I}_v(z)_{21}^2+\mathcal{I}_v(z)_{11}^2}\\ &+\frac{\partial_{z_l}\partial_{z_k}\mathcal{I}_v(z)_{21}\ \mathcal{I}_v(z)_{11} - \mathcal{I}_v(z)_{21}\ \partial_{z_l}\partial_{z_k}\mathcal{I}_v(z)_{11}}{\mathcal{I}_v(z)_{21}^2+\mathcal{I}_v(z)_{11}^2}\\ &-\frac{(2\,\mathcal{I}_v(z)_{21}\ \partial_{z_l}\mathcal{I}_v(z)_{21} + 2\,\mathcal{I}_v(z)_{11}\ \partial_{z_l}\mathcal{I}_v(z)_{11})}{\mathcal{I}_v(z)_{21}^2+\mathcal{I}_v(z)_{11}^2}\\ &\cdot\frac{(\partial_{z_k}\mathcal{I}_v(z)_{21}\ \mathcal{I}_v(z)_{11} - \mathcal{I}_v(z)_{21}\ \partial_{z_k}\mathcal{I}_v(z)_{11})}{\mathcal{I}_v(z)_{21}^2+\mathcal{I}_v(z)_{11}^2}.\end{aligned}$$

Local first derivatives

As noted above, each induced transition rotation depends on one dihedral angle and on the three edge lengths of a triangle. We simplify our notation for the computation of the partial derivatives and define

$$\hat{R}(\theta,a,b,c) := R_x(\theta)R_z\left(\arccos\left(\frac{a^2+b^2-c^2}{2ab}\right)\right). \tag{A.19}$$

Then, we compute the following derivatives using Mathematica [Wol]

$$\partial_\theta\hat{R}(\theta,a,b,c) = \begin{pmatrix} 0 & 0 & 0\\ -\sqrt{1-\frac{(a^2+b^2-c^2)^2}{4a^2b^2}}\sin(\theta) & -\frac{\left(a^2+b^2-c^2\right)\sin(\theta)}{2ab} & -\cos(\theta)\\ \sqrt{1-\frac{(a^2+b^2-c^2)^2}{4a^2b^2}}\cos(\theta) & \frac{\left(a^2+b^2-c^2\right)\cos(\theta)}{2ab} & -\sin(\theta)\end{pmatrix},$$

and

$$\partial_a\hat{R}(\theta,a,b,c) = \begin{pmatrix} \frac{a^2-b^2+c^2}{2a^2b} & \frac{a^4-\left(b^2-c^2\right)^2}{8a^2bA(a,b,c)} & 0\\ \frac{\left(\left(b^2-c^2\right)^2-a^4\right)\cos(\theta)}{8a^2bA(a,b,c)} & \frac{\left(a^2-b^2+c^2\right)\cos(\theta)}{2a^2b} & 0\\ \frac{\left(\left(b^2-c^2\right)^2-a^4\right)\sin(\theta)}{8a^2bA(a,b,c)} & \frac{\left(a^2-b^2+c^2\right)\sin(\theta)}{2a^2b} & 0\end{pmatrix},$$

and

$$\partial_b\hat{R}(\theta,a,b,c) = \begin{pmatrix} \frac{-a^2+b^2+c^2}{2ab^2} & \frac{b^4-\left(a^2-c^2\right)^2}{8ab^2\,A(a,b,c)} & 0\\ \frac{\left(\left(a^2-c^2\right)^2-b^4\right)\cos(\theta)}{8ab^2\,A(a,b,c)} & \frac{\left(-a^2+b^2+c^2\right)\cos(\theta)}{2ab^2} & 0\\ \frac{\left(\left(a^2-c^2\right)^2-b^4\right)\sin(\theta)}{8ab^2\,A(a,b,c)} & \frac{\left(-a^2+b^2+c^2\right)\sin(\theta)}{2ab^2} & 0\end{pmatrix},$$

and lastly

$$\partial_c \hat{R}(\theta,a,b,c) = \begin{pmatrix} -\frac{c}{ab} & \frac{c^3-\left(a^2+b^2\right)c}{4abA(a,b,c)} & 0 \\ \frac{c\left(a^2+b^2-c^2\right)\cos(\theta)}{4abA(a,b,c)} & -\frac{c\cos(\theta)}{ab} & 0 \\ \frac{c\left(a^2+b^2-c^2\right)\sin(\theta)}{4abA(a,b,c)} & -\frac{c\sin(\theta)}{ab} & 0 \end{pmatrix}.$$

Local second derivatives

For the second derivatives, we have

$$\partial_\theta\partial_\theta \hat{R}(\theta,a,b,c) = \begin{pmatrix} 0 & 0 & 0 \\ -\sqrt{1-\frac{\left(a^2+b^2-c^2\right)^2}{4a^2b^2}}\cos(\theta) & -\frac{\left(a^2+b^2-c^2\right)\cos(\theta)}{2ab} & \sin(\theta) \\ -\sqrt{1-\frac{\left(a^2+b^2-c^2\right)^2}{4a^2b^2}}\sin(\theta) & -\frac{\left(a^2+b^2-c^2\right)\sin(\theta)}{2ab} & -\cos(\theta) \end{pmatrix},$$

$$\partial_a\partial_\theta \hat{R}(\theta,a,b,c) = \begin{pmatrix} 0 & 0 & 0 \\ \frac{\left(a^4-\left(b^2-c^2\right)^2\right)\sin(\theta)}{8a^2bA(a,b,c)} & -\frac{\left(a^2-b^2+c^2\right)\sin(\theta)}{2a^2b} & 0 \\ \frac{\left(\left(b^2-c^2\right)^2-a^4\right)\cos(\theta)}{8a^2bA(a,b,c)} & \frac{\left(a^2-b^2+c^2\right)\cos(\theta)}{2a^2b} & 0 \end{pmatrix},$$

$$\partial_b\partial_\theta \hat{R}(\theta,a,b,c) = \begin{pmatrix} 0 & 0 & 0 \\ -\frac{\left(\left(a^2-c^2\right)^2-b^4\right)\sin(\theta)}{8ab^2A(a,b,c)} & \frac{\left(a^2-b^2-c^2\right)\sin(\theta)}{2ab^2} & 0 \\ \frac{\left(\left(a^2-c^2\right)^2-b^4\right)\cos(\theta)}{8ab^2A(a,b,c)} & \frac{\left(-a^2+b^2+c^2\right)\cos(\theta)}{2ab^2} & 0 \end{pmatrix},$$

$$\partial_c\partial_\theta \hat{R}(\theta,a,b,c) = \begin{pmatrix} 0 & 0 & 0 \\ -\frac{c\left(a^2+b^2-c^2\right)\sin(\theta)}{4abA(a,b,c)} & \frac{c\sin(\theta)}{ab} & 0 \\ \frac{c\left(a^2+b^2-c^2\right)\cos(\theta)}{4abA(a,b,c)} & -\frac{c\cos(\theta)}{ab} & 0 \end{pmatrix},$$

$$\partial_a\partial_a \hat{R}(\theta,a,b,c) = \begin{pmatrix} \frac{(b-c)(b+c)}{a^3b} & -\frac{X_1(a,b,c)}{64a^3bA(a,b,c)^3} & 0 \\ \frac{X_1(a,b,c)\cos(\theta)}{64a^3bA(a,b,c)^3} & \frac{(b-c)(b+c)\cos(\theta)}{a^3b} & 0 \\ \frac{X_1(a,b,c)\sin(\theta)}{64a^3bA(a,b,c)^3} & \frac{(b-c)(b+c)\sin(\theta)}{a^3b} & 0 \end{pmatrix},$$

$$\partial_b\partial_a \hat{R}(\theta,a,b,c) = \begin{pmatrix} -\frac{a^2+b^2+c^2}{2a^2b^2} & -\frac{X_2(a,b,c)}{128a^2b^2A(a,b,c)^3} & 0 \\ \frac{X_2(a,b,c)\cos(\theta)}{128a^2b^2A(a,b,c)^3} & -\frac{\left(a^2+b^2+c^2\right)\cos(\theta)}{2a^2b^2} & 0 \\ \frac{X_2(a,b,c)\sin(\theta)}{128a^2b^2A(a,b,c)^3} & -\frac{\left(a^2+b^2+c^2\right)\sin(\theta)}{2a^2b^2} & 0 \end{pmatrix},$$

$$\partial_c\partial_a \hat{R}(\theta,a,b,c) = \begin{pmatrix} \frac{c}{a^2b} & -\frac{X_3(a,b,c)}{64a^2bA(a,b,c)^3} & 0 \\ \frac{X_3(a,b,c)\cos(\theta)}{64a^2bA(a,b,c)^3} & \frac{c\cos(\theta)}{a^2b} & 0 \\ \frac{X_3(a,b,c)\sin(\theta)}{64a^2bA(a,b,c)^3} & \frac{c\sin(\theta)}{a^2b} & 0 \end{pmatrix},$$

$$\partial_b\partial_b \hat{R}(\theta,a,b,c) = \begin{pmatrix} \frac{(a-c)(a+c)}{ab^3} & -\frac{X_4(a,b,c)}{64ab^3A(a,b,c)^3} & 0 \\ \frac{X_4(a,b,c)\cos(\theta)}{64ab^3A(a,b,c)^3} & \frac{(a-c)(a+c)\cos(\theta)}{ab^3} & 0 \\ \frac{X_4(a,b,c)\sin(\theta)}{64ab^3A(a,b,c)^3} & \frac{(a-c)(a+c)\sin(\theta)}{ab^3} & 0 \end{pmatrix},$$

$$\partial_c\partial_b \hat{R}(\theta,a,b,c) = \begin{pmatrix} \frac{c}{ab^2} & -\frac{X_5(a,b,c)}{64ab^2A(a,b,c)^3} & 0 \\ \frac{X_5(a,b,c)\cos(\theta)}{64ab^2A(a,b,c)^3} & \frac{c\cos(\theta)}{ab^2} & 0 \\ \frac{X_5(a,b,c)\sin(\theta)}{64ab^2A(a,b,c)^3} & \frac{c\sin(\theta)}{ab^2} & 0 \end{pmatrix}, \text{ and}$$

$$\partial_c\partial_c\hat{R}(\theta,a,b,c) = \begin{pmatrix} -\frac{1}{ab} & -\frac{c^6-3\left(a^2+b^2\right)c^4+3\left(a^2-b^2\right)^2c^2-\left(a^2-b^2\right)^2\left(a^2+b^2\right)}{64abA(a,b,c)^3} & 0 \\ -\frac{X_7(a,b,c)\cos(\theta)}{64abA(a,b,c)^3} & -\frac{\cos(\theta)}{ab} & 0 \\ -\frac{X_7(a,b,c)\sin(\theta)}{64abA(a,b,c)^3} & -\frac{\sin(\theta)}{ab} & 0 \end{pmatrix},$$

where

$$\begin{aligned}
X_1(a,b,c) &:= \left(b^2+c^2\right)a^6 - 3\left(b^2-c^2\right)^2a^4 + 3\left(b^2-c^2\right)^2\left(b^2+c^2\right)a^2 - \left(b^2-c^2\right)^4 \\
X_2(a,b,c) &:= c^8 - 2\left(a^2+b^2\right)c^6 - 8a^2b^2c^4 + 2\left(a^2+b^2\right)^3c^2 - \left(a^2-b^2\right)^4 \\
X_3(a,b,c) &:= c\left(a^6 + 3(b-c)(b+c)a^4 + \left(-5b^4+2c^2b^2+3c^4\right)a^2 + \left(b^2-c^2\right)^3\right) \\
X_4(a,b,c) &:= c^8 - \left(4a^2+3b^2\right)c^6 + 3\left(2a^4+b^2a^2+b^4\right)c^4 \\
&\qquad - \left(4a^6-3b^2a^4+6b^4a^2+b^6\right)c^2 + a^2\left(a^2-b^2\right)^3 \\
X_5(a,b,c) &:= c\left(a^6 - \left(5b^2+3c^2\right)a^4 + \left(3b^4+2c^2b^2+3c^4\right)a^2 + \left(b^2-c^2\right)^3\right) \\
X_7(a,b,c) &:= a^6 - \left(b^2+3c^2\right)a^4 + \left(-b^4+6c^2b^2+3c^4\right)a^2 + \left(b^2-c^2\right)^3,
\end{aligned}$$

and $A(a,b,c)$ is the area as before.