

Riemannian Calculus and Shape Optimization on the Space of Discrete Surfaces

Dissertation
zur
Erlangung des Doktorgrades (Dr. rer. nat.)
der
Mathematisch-Naturwissenschaftlichen Fakultät
der
Rheinischen Friedrich-Wilhelms-Universität Bonn

vorgelegt von
Josua Raphael Sassen
aus Bonn

Bonn, Februar 2023

Angefertigt mit Genehmigung der Mathematisch-Naturwissenschaftlichen Fakultät
der Rheinischen Friedrich-Wilhelms-Universität Bonn

1. Gutachter: Prof. Dr. Martin Rumpf
2. Gutachter: Prof. Dr. Max Wardetzky

Tag der Promotion: 25.05.2023
Erscheinungsjahr: 2023

Summary

This thesis makes contributions to shape spaces and shape optimization of discrete surfaces. Discrete surfaces are triangle meshes with certain regularity and are ubiquitous in many applications in computer graphics, geometric design, and computational anatomy to name only a few.

We consider the space of all immersions of a fixed discrete surface. Heeren *et al.* [HRS+14] used a physically-inspired metric derived from an elastic shell model to introduce the structure of a Riemannian manifold on this space resulting in the shape space of discrete shells. Rumpf and Wirth [RW15] introduced the time-discrete geodesic calculus used for this shape space.

In this thesis, we show that this space can be represented as an implicit submanifold by considering the *Nonlinear Rotation-Invariant Coordinates* (NRIC) consisting of the vector stacking edge lengths and dihedral angles. To this end, we exploit discrete integrability conditions akin to the famous Gauß–Codazzi equations introduced by Wang, Liu, and Tong [WLT12], leading to the implicit description. Furthermore, we also show that NRIC are helpful to phrase and numerically solve geometric variational problems on discrete surfaces, especially if they involve isometry constraints.

Based on the implicit representation, we propose a new method to construct submanifolds from shape datasets. We augment Principal Geodesic Analysis, an algorithm from Riemannian statistics, by a sparsity-inducing regularization leading to *Sparse Principal Geodesic Analysis*. The resulting sparse nonlinear modes of variation span a shape submanifold we can equip with a product structure based on decoupling modes. We propose two efficient parametrizations of this submanifold: One based on a grid-based multilinear interpolation of the Riemannian exponential map on individual factors combined affinely and one based on neural networks with an architecture and training procedure incorporating the product structure. These parametrizations are useful for (near) real-time applications in computer graphics, as demonstrated by numerical examples.

Finally, we introduce a modification of the metric that moves surfaces with self-intersections infinitely far away from ones without. We will do so by leveraging the tangent-point energy, a repulsive nonlocal curvature energy for which Yu *et al.* [YBSC21] recently introduced an efficient discretization and numerical tools for its application in computer graphics. Our principal insight is that the graph of this energy over the space of discrete shells yields a metric as desired while retaining the appeal that geodesics provide physically-sound interpolations. This leads to the *space of repulsive shells*. We also propose numerical methods to compute geodesics and exponential maps on it.

Additionally, we consider two shape optimization problems on discrete surfaces outside the realm of shape spaces. First, we introduce an approach to *stochastic bilevel shape optimization* that models, figuratively speaking, a design engineer optimizing shape parameters, a test engineer devising worst-cased load scenarios, and stochastic manufacturing inaccuracies. We develop numerical algorithms to compute solutions of the corresponding mathematical model and apply it to optimize thickness distributions on a discrete shell modeling roof-type structures.

Furthermore, we consider a *phase-field model for surface segmentation* into equally sized parts with minimal interface length and distortion when conformally mapping the segments to the plane. To compute the distortion, we adopt the approach of Sharp and Crane [SC18] and introduce a diffuse version of the Yamabe equation used as a PDE-constraint. Finally, we discretize the resulting variational problems using finite elements and standard algorithms for nonlinear optimization.

Acknowledgements

First, I would like to thank all the people directly involved in preparing this thesis and my doctoral studies: *Martin Rumpf* for being an exceptional advisor, for introducing me to this area of research, for sharing his mathematical insights, for providing an encouraging and inspiring environment, for placing an immense amount of trust in my capabilities, for repeatedly throwing me in at the deep end, for providing me with all kinds of resources, and for giving me advice about and investing in my further career. *Max Wardetzky* for being co-reviewer of this thesis and for providing us with data and advice for paper folding examples included in Chapter 4. *Tim Laux* for chairing my doctoral committee even when it was detrimental to his own plans and for always being open for fun conversations. *Matthias Hullin* for being the ‘non-mathematical’ member of my doctoral committee.

The work presented in this thesis is the result of intense collaboration with many people I would also like to thank: *Klaus Hildebrandt* for the magnificent collaboration already lasting through multiple projects starting with my Master’s thesis, for his invaluable insights on these projects, for having me as a guest at the TU Delft, and for all his helpful advice going beyond our projects. *Keenan Crane* for introducing me to some truly repulsive stuff, for always being open to discussing my ideas and asking for feedback on his’, for consistently pushing for better results, for his support of my future career ambitions, and finally for hosting me at the Carnegie Mellon University. *Benedikt Wirth* for his sharp mathematical insights and for fruitful discussions during the quest to understand what our network-based parametrizations are doing. *Henrik Schumacher* for making stuff repulsive and for starting many entertaining conversations by going off-topic. *Miri Ben-Chen* and *Omri Azencot* for accompanying us on our excursion through the jungle of functional maps and for being excited about our results. *Johanna Burtscheidt*, *Matthias Claus*, *Sergio Conti*, and *Rüdiger Schultz* for teaching me about bilevel optimization and for the following patient and productive collaboration.

During the last almost five years, I had the luxury of being part of a tremendous research environment, which is due to many additional people: *Behrend Heeren* for being the perfect first office-mate, for helping me learn to navigate the world of research, for all our insightful discussions, and especially for all the relaxing coffee breaks. I hope I succeeded in paying this forward to the next generation. *Florine Hartwig* and *Christoph Smoch* for making the return to the office after working from home so refreshing as my incredible officemates and for all the fun we have in Bonn and while traveling together. Also, Christoph for always being ready to dream about phase-fields with me and Florine for being a trusted confidant on our joint projects and beyond. *Stefan Simon* for all our discussions on implementation and for our coffee breaks. *Marko Rajković* and *Jorge Justiniano* for proof-reading part of this thesis and for navigating the little pitfalls of doctoral studies and writing a thesis together. *Martin Lenz* for all his technical assistance, for proof-reading part of this thesis, and especially for lively lunch breaks. *Carole Rossignol* for always having my back when it comes to administrative things. *Rohan Sawhney*, *Nicole Feng*, and especially *Mark Gillespie* for welcoming me at CMU and quickly integrating me into their circles. *Kai Echelmeyer*, *Janos Meny*, and *Yannick Kees* for trusting me to be their mentor while working on their Master’s thesis which allowed me to grow personally and for the ensuing rewarding collaborations.

Pursuing a PhD also requires a strong support system outside the university and for this I am grateful to: *Renate, Roland, Steffen, and Sören Künzel*, as well as, *Palina Golovanova* for becoming like a second family and for many, many memorable Settlers of Catan matches. *Tornado, Zorro*, and *Coco* for bringing out my caring side. *Alexander, Franziska, and Sharon Sassen* for accompanying me on my way growing up and for doing stupid things together. My parents *Luzia and Johannes Sassen* for their endless support, for enabling my education, for understanding when I forget to call, and for exposing me early to both art and mathematics, which I believe shows in this thesis. *Sandrine Künzel* for basically everything, for constantly making me a better person, for being proud of me even when I am not, for enduring the hardships working on this thesis brought with it and still encouraging me, for every time we laugh together, for being my best friend always on my side, and for every time she improves my day just by being there.

Funding. My work was funded by the *German Research Foundation* (DFG) through the project “Geodesic Paths in Shape Space” (DFG project 212212052), which was part of the National Research Network (NFN) *Geometry + Simulation* of the Austrian FWF, and the *Collaborative Research Centre (SFB) 1060* (DFG project 211504053). Furthermore, it was funded by the *German Israeli Foundation* on Scientific Research and Development (GIF) through the project “A Functional Map Approach to Shape Spaces.” Finally, I am thankful for the material and immaterial support by the *Bonn International Graduate School of Mathematics*—part of the Cluster of Excellence *Hausdorff Center for Mathematics* (DFG project 390685813)—and especially for enabling my travels on many occasions helping me to be an active member of the research community.

Contents

1	Introduction	1
1.1	Related Work	6
1.1.1	Shape Spaces	6
1.1.2	Shape Optimization	9
1.2	Collaborations and Publications	12
2	Surfaces and Elasticity	13
2.1	Differential Geometry of Surfaces	13
2.1.1	Fundamental Forms	14
2.2	Discrete Surfaces	17
2.2.1	Fundamental Forms	19
2.3	From Nonlinear Elasticity to Discrete Shells	21
2.3.1	Membrane and Bending Energies	22
2.3.2	Discrete Deformations Energies	23
I	Shape Spaces	25
3	Shape Spaces and Time-Discrete Geodesic Calculus	27
3.1	Riemannian Manifolds	27
3.2	Time-Discrete Geodesic Calculus	29
3.3	The Space of Discrete Shells	31
4	Nonlinear Rotation-Invariant Coordinates	33
4.1	Discrete Gauß–Codazzi Equations	34
4.1.1	Discrete Integrability Conditions	35
4.2	Manifold Structure and Geodesic Calculus	39
4.3	Geometric Optimization Problems	41
4.3.1	Elastic Energies	41
4.3.2	Algorithmic Approach	43
4.4	Experiments and Applications	46
4.5	Conclusion and Outlook	51
5	Sparse Principal Geodesic Analysis	53
5.1	Principal Geodesic Analysis	55
5.2	Sparsity and Decoupling	57
5.3	Submanifold Decomposition and Deformation Synthesis	58
5.3.1	Submanifold Approximation	58
5.3.2	Efficient Deformation Synthesis	59
5.4	Numerical Implementation	61

5.4.1	Quadratic Matrix Programming	62
5.5	Experiments and Applications	66
5.6	Parametrization via Composite Networks	72
5.6.1	Experiments and Applications	74
5.7	Conclusion and Outlook	79
6	The Space of Repulsive Shells	81
6.1	Tangent-Point Energy	82
6.1.1	Definition and Properties	83
6.1.2	Discretization	84
6.2	Shape Space Augmentation	87
6.2.1	Time-Discretization	89
6.3	Implementation and Experiments	91
6.3.1	Numerical Optimization	91
6.3.2	Shape Space Results	93
6.3.3	Elastic Deformations	95
6.4	Conclusion and Outlook	97
II	Shape Optimization	101
7	Stochastic Bilevel Shape Optimization	103
7.1	A Glimpse of Bilevel Optimization	104
7.2	Bilevel Problem Formulation and Analysis	107
7.2.1	Deterministic Problem	107
7.2.2	Stochastic Problem	109
7.3	Application to Discrete Shells	112
7.4	Experimental Results	116
7.5	Conclusion and Outlook	121
8	A Phase-field Approach to Surface Segmentation	123
8.1	Phase-fields on Euclidean Domains	125
8.2	Phase-fields on Surfaces	129
8.2.1	Hierarchical Approach	129
8.3	A Diffuse Yamabe Equation	131
8.4	Discretization, Implementation, and Experiments	135
8.4.1	Discretization	135
8.4.2	Results and Applications	138
8.5	Conclusion and Outlook	141
	Bibliography	143

Nomenclature

The derivatives of a functional $\mathcal{F}: \mathbb{R}^n \rightarrow \mathbb{R}^m, (\xi_1, \dots, \xi_n) \mapsto \mathcal{F}[\xi_1, \dots, \xi_n]$ 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
$\psi: U \subset \mathbb{R}^d \rightarrow \mathcal{M}$	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
$\text{Exp}_p^K, \text{Log}_p^K$	time-discrete exponential map and logarithm of order K at p on \mathcal{M}
dist_g	Riemannian distance on \mathcal{M} with respect to metric g

Surfaces

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
$F: \Omega \rightarrow O(3)$	frame field on surface 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
$\mathbf{V}, \mathbf{E}, \mathbf{T}$	sets of vertices, edges, and faces of a triangle mesh
X	embedding or immersion of a discrete surface, i.e. its nodal positions
\mathbf{S}	embedded or immersed discrete surface
$X_v, E_e, T(\tau)$	embedded vertex, edge, or face respectively
$G _\tau, H _\tau, S _\tau$	elementwise constant discrete first and second fundamental forms, and shape operator
N_τ	face normal at $\tau \in \mathbf{T}$
l_e, θ_e	edge length and dihedral angle of an edge e

Physical 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
$\mathcal{M}[\mathcal{S}_h]$	admissible lengths and angles for discrete surface \mathcal{S}_h , also the NRIC manifold
$Z: \mathbb{R}^{3 V } \rightarrow \mathbb{R}^{2 E }$	map yielding the lengths and angles $Z[X]$ of an immersion X
$F: \mathbf{T} \rightarrow SO(3)$	discrete frame field on \mathcal{S}_h
$R_{ij}: \mathbb{R}^{2 E } \rightarrow SO(3)$	induced transition rotation between faces τ_i and τ_j
$\mathcal{T}: \mathbb{R}^{2 E } \rightarrow \mathbb{R}^{3 T }$	triangle inequality map
$\mathcal{I}: \mathbb{R}^{2 E } \rightarrow SO(3)^{ V }$	discrete integrability map
$\mathcal{Q}: \mathbb{R}^{2 E } \rightarrow \mathbb{R}^{3 V }$	quaternion formulation of discrete integrability map
\mathcal{W}_q	quadratic deformation energy on lengths and angles

Riemannian Statistics

\bar{z}	Riemannian center of mass
\mathcal{M}^J	J -dimensional (data approximating) submanifold
$\mathcal{M}[\mathcal{U}^J]$	J -dimensional submanifold parametrized over subspace $\mathcal{U}^J \subset T_{\bar{z}}\mathcal{M}$ via the exponential map
$\mathbf{Z}_\sigma[\cdot], z_\sigma[\cdot]$	approximate grid-based parametrization of SPGA manifold before resp. after projection onto \mathcal{M}
$\Phi^\zeta(\cdot)$	approximate neural network-based parametrization of SPGA manifold
$\text{MLP}_\rho^\zeta[N_1, \dots, N_T]$	a fully-connected network with layer sizes N_1, \dots, N_T , nonlinear activation function $\rho: \mathbb{R} \rightarrow \mathbb{R}$ after each layer, and parameters ζ

Repulsive Shells

\mathcal{T}^α	tangent-point energy with integrability parameter α
\mathcal{T}_h^α	naïve discretization of tangent-point energy on discrete surface \mathcal{S}_h
$\mathcal{T}_{h,\eta}^\alpha$	fast multipole-based adaptive discretization of tangent-point energy
\mathcal{V}	generic potential energy $\mathcal{V}: \mathcal{M} \rightarrow \mathbb{R}_{\geq 0}$ on (\mathcal{M}, g)
$g^\mathcal{V}$	pull-back metric from graph of \mathcal{V} over \mathcal{M}

Bilevel Optimization

j, g, \mathcal{F}	objective, constraint, and admissible set of follower
J, G, \mathcal{U}	objective, constraint, and admissible set of leader
Ψ, ψ	lower level solution set mapping and optimal value mapping
Φ_o, Φ_p	optimistic resp. pessimistic objective
$(\Omega, \mathcal{B}, \mathbb{P})$	probability space
Υ	random variable on $(\Omega, \mathcal{B}, \mathbb{P})$
u	design parameters / triangle thicknesses
f	forces
$y[u, f]$	elastic displacement

Phase-fields

u	phase-field
$\mathcal{P}_\varepsilon[\cdot]$	Modica–Mortola functional
$\mathcal{A}[\cdot]$	area approximation for positive phase
$\mathcal{C}_\varepsilon[\cdot]$	connectedness functional / constraint for positive phase
$\chi(u)$	diffuse indicator function of positive phase
s	logarithmic conformal scale factor
$\mathcal{Y}_\varepsilon[u, \cdot]$	diffuse Yamabe functional

Chapter 1

Introduction

Processing geometric shapes lies at the heart of many applied domains. In computer graphics one considers—for example—their animation, in engineering their optimal design and elastic simulation, in biomedicine their statistical analysis, and this list could be continued endlessly. Therefore, powerful digital and mathematical tools for working with geometry are paramount for advances in these fields. The development of such tools is the core objective of *geometry processing*—an emergent area of research.

In many applications, shapes are represented by surfaces, i.e. curved two-dimensional objects in three-dimensional space. Surfaces have been the object of intense mathematical studies since at least ancient Greece. Over time, a rich theory for their topology and geometry has been developed, which we will consider through the modern language of Riemannian geometry. This language also extends to higher dimensional objects as we will discuss later. For computation and visualization, numerical representations of surfaces on modern computers are essential. One extremely popular choice is to describe them by triangle meshes called *discrete surfaces* under certain conditions. The study of discrete surfaces is a central element of the active field of discrete differential geometry with the goal to develop a comprehensive theory—similar to continuous surface theory—and computational tools for them. We will provide some necessary insights into these efforts in Chapter 2.

This thesis makes contributions to mathematical geometry and especially surface processing. From a bird's perspective, we will consider questions concerning *shape spaces* and *shape optimization*. This is reflected by structuring the thesis into two corresponding parts. A common theme will be devising nonlinear variational problems modeling the problem in question and subsequently developing effective numerical tools for their solution.

Shape Spaces. In applications such as animation or shape analysis, we are interested in processing multiple shapes at once and, hence, in a mathematical model for collections of shapes yielding flexible numerical tools. Kendall [Ken84] proposed to consider Riemannian shape spaces in this context, i.e. possibly infinite-dimensional Riemannian manifolds where points are geometric objects such as surfaces. This is closely related to the concept of moduli spaces as studied in pure mathematics of which the Grassmannians—spaces of linear subspaces—are among the most well-known examples. In the following, we will focus on Riemannian shape spaces and thus mostly refer to them simply as shape spaces. They have found usage in a lot of areas of applied mathematical research such as computational anatomy, computer graphics, shape optimization, and image processing.

In this thesis, we will consider a shape space of immersions of a discrete surface—often referred to only as the *space of discrete shells*. That means we will consider a triangle mesh with fixed connectivity, i.e. a fixed set of faces and edges. Then the space consists of all possibilities to realize this mesh in three-dimensional space such that no triangle degenerates. These realizations are described by the nodal positions of the mesh and are called *discrete immersions*. Crucially, this means that the

topology of the shape is fixed and in this shape space only its geometry varies. This assumption is also called dense correspondence and simplifies, for example, the computation of deformations by avoiding the necessity to compute correspondences.

On this space, we need a Riemannian structure to be useful in our applications of interest. That means we need to assign a cost to (infinitesimal) motions on the shape space leading to a Riemannian metric. One approach would be to consider the immersion as elements of some function space and then the norm of this function space—e.g. some kind of Sobolev norm—yields a geometric way to obtain a Riemannian metric. However, we will follow the idea of Heeren *et al.* [HRWW12] and consider a physically-inspired metric. Given a discrete elastic shell energy penalizing tangential and bending distortions, we consider the combined elastic energies of small consecutive steps along a path on the shape space. By letting the size of these steps go to zero, we obtain a viscous model Riemannian metric. This is also called Rayleigh’s paradigm. The resulting Riemannian shape space with this metric is called the *space of discrete shells*.

It is useful for many applications in computer graphics. For example, shape interpolation can be phrased as constructing geodesics—i.e. locally shortest paths, shape extrapolation as an application of the Riemannian exponential map—shooting geodesics, and detail or pose transfer as parallel transport. Moreover, it can be used to construct so-called deformation priors for editing tasks as we will see later on. To apply this to actual data, one needs a discretization in time for these geometric operations. To this end, we rely on the variational time-discretization of geodesic calculus introduced by Rumpf and Wirth [RW15], which we will discuss in more detail in Chapter 3. It leads to nonlinear variational problems that can be solved efficiently with numerical optimization algorithms and comes with a rigorous convergence theory under time refinement. Especially, already solutions for coarse time steps are often predictive of finer ones.

While this leads mostly to plausible and visually pleasing interpolation, it is not enough to get an accurate model for the movement of shapes. For example, it does not represent the skeleton of humanoid models. This limits its usage in editing applications because a point on the shape space is not guaranteed to be a plausible deformation of a humanoid even if its distance to plausible ones is small. This has spawned a recent interest in data-driven approaches to exploit existing databases of shape poses to alleviate this issues. Typically, they aim at constructing subspaces of the shape space from data such that they only contain plausible deformations of the shape. The prototypical algorithm for this is called *Principal Geodesic Analysis* (PGA) and was introduced by Fletcher *et al.* [FLPJ04]. Its adaptation to the space of discrete shells by Heeren *et al.* [HZRS18] yields high quality results for editing tasks. However, it is computationally very expensive which restricts its applicability in practical scenarios. Furthermore, the invariance of the approach to rigid body motions makes its formulation in nodal positions very cumbersome. Hence, one central goal of this thesis is to develop more efficient approaches to the construction of data approximating submanifolds of the space of discrete shells. Our ambition is to establish methods suitable for real-time applications.

Since working in nodal positions is cumbersome as explained above, the first step towards this is to find an alternative representation of the space of discrete shells. In Chapter 4, we will introduce the so-called *Nonlinear Rotation-Invariant Coordinates* (NRIC) for this purpose which consist of the vector stacking edge lengths and dihedral angles. They are inherently rigid body motion-invariant and have a natural relation to elastic distortions. Using discrete integrability conditions introduced by Wang, Liu, and Tong [WLT12], we will see that the space of immersions of a discrete surface can be represented as an implicit submanifold of an

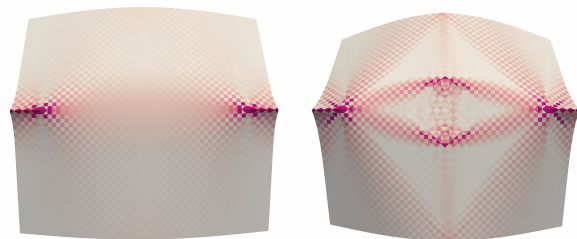


Figure 1.1: *NRIC can be used to simulate isometric paper folding (right) where nodal positions lead to inaccurate results (left). See also Figure 4.6.*

high-dimensional Euclidean space. This simplifies the description and computation of many geometric notions. Furthermore, we will also see that NRIC are very useful to phrase and numerically solve geometric variational problems on discrete surfaces, especially if they involve isometry constraints. Such a problem involving the isometric folding of a sheet of paper can be seen in Figure 1.1.



Figure 1.2: Incremental mesh editing example using our approach. See also Figure 5.1.

Based on this representation, we will propose a new method to construct submanifolds of the space of discrete shells in Chapter 5. The principal objective is that we have an efficient parametrization of the submanifold suitable for (near) real-time applications. We recognize that we need additional structure on the submanifold to achieve this goal. Thus, we will construct the submanifold to have a certain product structure that will stem from the sparsity and decoupling of nonlinear modes of shape variation spanning the submanifold. Here, sparsity means that the modes of variation only affect a small region of the mesh, where ‘small’ will be crucially measured in NRIC. We will construct these sparse modes using a modification of PGA. Then we will introduce two efficient approximate parametrizations of the submanifold, useful for mesh editing as shown in Figure 1.2. The first will be the combination of a grid-based multilinear interpolation of the exponential map-based parametrization of the individual factor submanifolds and an affine combination of the results. This will already provide an efficient approximate parametrization but will be limited in the number of dimensions it can handle. Hence, we will also adopt structural insights from this to construct an efficient parametrization based on neural networks.

Finally, we will address one other problem restricting the plausibility of deformations: self-intersections. All methods we have discussed so far—even the data-driven ones—are not able to prevent them. Mathematically speaking, they only guarantee *immersions* and not *embeddings*. Even more, there are only few shape interpolation techniques able to prevent self-intersections, which rely on diffeomorphisms of ambient space requiring expensive discretizations. Hence, in Chapter 6,

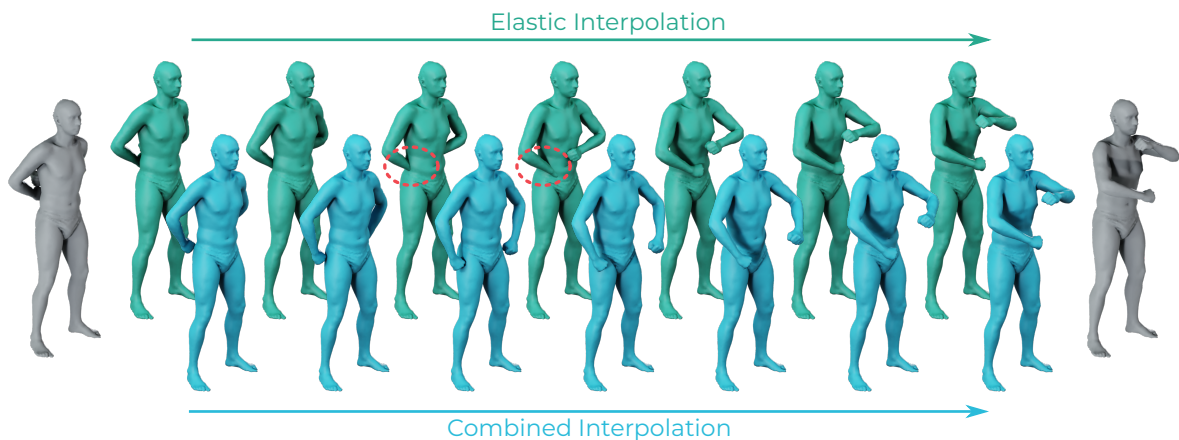


Figure 1.3: Geodesic on humanoid shape. The interpolation on the space of discrete shells (green) leads to intersections especially visible at the timesteps in the center (marked by red ellipses). Geodesics with respect to our metric incorporating the tangent-point energy (blue) avoid these self-intersections.

we will propose a modification of the metric on the space of discrete shells that moves shapes with self-intersections infinitely far away from ones without. We will do so by leveraging the tangent-point energy, a repulsive energy for which recently an efficient discretization and numerical tools for its application in computer graphics have been introduced. Our principal insight will be that the graph of the tangent-point energy over the space of discrete shells yields such a Riemannian metric while retaining the appeal that geodesics provide physically-sound interpolations between complex surfaces. This will lead to the *space of repulsive shells* and a geodesic on this space is shown in Figure 1.3.

Shape Optimization. Variational problems offer a powerful way to describe and design shapes by characterizing them as minimizers of an energy subject to constraints. These energies can be used to describe all kinds of desired properties, for example, regarding the topology, the size, the complexity, the mechanical properties of the shape or even its elastic behavior under mechanical load. Also the variational problems arising from the shape space problems sketched above could be regarded as shape optimization problems in a broad sense. However, in the second part of this thesis, we will consider two shape optimization problems outside the framework of shape spaces and more closely related to geometric design.

Indeed, elastic behavior under load will be of primary concern in the problem discussed in Chapter 7. Hence, we deal with a so-called elastic shape optimization problem. However, we will not consider some predetermined finite set of load scenarios, i.e. sets of forces applied to the shape. Instead, we will consider the determination of a worst-case scenario. This will lead to a hierarchical optimization problem, i.e. the optimization of the shape depends on the optimization of the force which in turn depends on the optimization of the deformation. Since we will consider a linearized model of elasticity for deformations such that the last level has

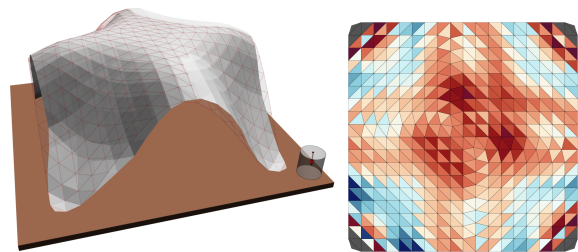


Figure 1.4: *Our bilevel elastic shape optimization approach produces symmetry-breaking material configurations (right) ‘tricking’ the follower to choose a worst-case force benevolent for the leader (left). See also Figure 7.2.*

a closed form solution, we can consider our problem as a so-called *bilevel optimization problem*. These problems have been introduced in and extensively investigated for economy-driven decision making. Recently, they found their way also to other fields such as machine learning. They can be seen as the interplay of two decision makers—a leader and a follower—whose decisions depend on another in a certain sense. This also found some application in elastic shape optimization. Our approach will differ from this in that we will consider a so-called pessimistic problem where the lower level is not the computation of deformations. This means it is more challenging to analyze and treat numerically yet are more realistic. In our scenario, the leader can be imagined as a design engineer controlling design variables of the elastic object and the follower as a test engineer trying to break it. Furthermore, we will also consider a stochastic extension of the problem modeling manufacturing inaccuracies. We will develop numerical algorithms to compute solutions of these problems and apply it as a proof-of-concept to the optimization of thickness distributions on a discrete shell modeling roof-type structures—an example of which can be seen in Figure 1.4.

The final problem we will consider in this thesis concerns the segmentation of surfaces. That means we will consider the optimization of the shape of segments on a (discrete) surface. Our goal is a hierarchical segmentation of the surface into equally sized parts with minimal length of interfaces between segments. Furthermore, we will also ask for an elastic property of the segments: When conformally mapping the segments to the plane—i.e. pressing them flat—we want the incurred area distortion to be minimal. This kind of segmentation is useful for many downstream

applications in computer graphics and other PDE-based methods. The particular example application we consider as proof-of-concept is texture mapping. In Chapter 8, we will use phase-fields

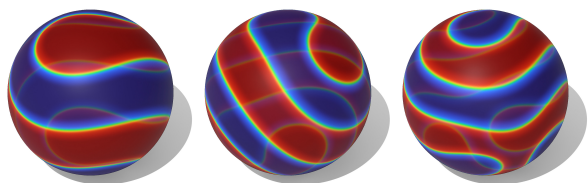


Figure 1.5: *‘Peeling’ a sphere using our phase-field-based surface segmentation. See also Figure 8.6.*

to describe the segments variationally. Phase-fields represent segments as areas of constant material—described by a scalar value—with a smooth transition between them. They have been widely used in image segmentation, elastic shape optimization, and the analysis of free boundary problems among other areas. However, so far, they have found no use in the computer graphics or geometry processing community. Thus, one can understand our method also

as a proof-of-concept application of phase-fields in computer graphics. To compute the distortion of conformal flattening the segments, we adopt the approach of Sharp and Crane [SC18] and introduce a diffuse version of the Yamabe equation used as a PDE-constraint. The resulting diffuse formulation of the surface segmentation problem leads to variational problems that are straightforward to discretize with finite elements and treat with standard algorithms for nonlinear optimization. An exemplary solution can be seen in Figure 1.5.

Structure. As explained above, this thesis is structured into two parts: one on shape spaces and one on shape optimization. Before going into them, we introduce useful background on surfaces, their discrete counterparts, and shell elasticity in Chapter 2. In the part on shape spaces, we will first provide some additional background in Chapter 3. Then we will proceed to introduce the Non-linear Rotation-Invariant Coordinates in Chapter 4, which we will use in Chapter 5 to introduce our efficient submanifold construction while the problem of self-intersections will be addressed in Chapter 6. In the second part on shape optimization, we directly start with the bilevel elastic shape optimization in Chapter 7 and conclude with the phase-field-based segmentation of surfaces in Chapter 8. The Chapters 4, 6, 7, and 8 are mostly self-contained except for the background chapters which might not be necessary for readers familiar with the field. Chapter 5 relies to some extent on the NRIC introduced in Chapter 4.

1.1 Related Work

1.1.1 Shape Spaces

In a variety of applied fields, shape spaces have become an important paradigm to devise methods for processing geometric data. In this thesis, we will be particularly concerned with the space of immersions of a discrete surface. Nevertheless, we will provide a summary of related work looking beyond this particular shape space below. The definition of a shape space can sometimes just arise from the way the geometry of a shape is represented. Since these representations are also important for other problems, we will discuss them first before passing to shape spaces. Afterwards, we will discuss works outlining how shape spaces can be used in ways relevant to the remainder of the thesis.

Shape Coordinates. As explained above, the representation of the immersion of a surface—often called shape coordinates—can have a dramatic effect on the effectiveness of algorithms. Therefore, it has been an intense subject of study and we summarize some important contributions here. We distinguish between coordinates that depend linearly and nonlinearly on the nodal positions.

The most important class of linear coordinates are so-called differential coordinates that use discrete differential operators on the discrete surface to define the coordinates. For example, Yu *et al.* [YZX+04] and Sumner and Popović [SP04] consider the gradient of immersions which entails solving a Poisson problem to reconstruct nodal positions. Similarly, Sorkine *et al.* [SCL+04] and Lipman *et al.* [LSC+04] applied the Laplace–Beltrami operator to nodal positions to obtain so-called Laplace coordinates. They also considered the explicit treatment of rigid body motions and further global transformation. Since these coordinates depend linearly on nodal positions, linear least-squares problem are used to compute the immersion that best matches given differential coordinates. This makes it computationally efficient, however, shape editing using linear coordinates often yields unnatural and distorted shapes when larger deformations are involved. This phenomenon and linear shape coordinates in general are also discussed in the review by Botsch and Sorkine [BS08].

For continuous surfaces, the fundamental theorem of surfaces states that two immersions of a surface to \mathbb{R}^3 differ by a rigid motion if and only if the first and second fundamental forms agree. Furthermore, it provides integrability conditions on these fundamental forms—called the Gauß–Codazzi equations—that guarantee the existence of corresponding immersions. Thus fundamental forms can be seen as nonlinear shape coordinates for continuous surfaces. This motivates using discrete analogs to the fundamental forms as coordinates for immersions of discrete surfaces. To this end, Lipman *et al.* [LSLC05] used discrete frame fields anchored at vertices to define discrete fundamental forms on triangle meshes. They linearized the ensuing nonlinear problem of reconstruction immersions at some reference shape to enable efficient mesh editing applications. Due to the linearization, the quality of results typically deteriorates for strong metric distortions. Consequently, Kircher and Garland [KG08], Baran *et al.* [BVG09], Hasler *et al.* [HSS+09], Gao *et al.* [GLL+16], and Ambellan, Zachow, and von Tycowicz [AZv19a] extended these ideas and proposed various ways to construct discrete frame fields on triangle meshes along with corresponding approximate metric distortions and transition rotations approximating the derivative of the surface normals. Many also linearize the reconstruction problem and none provide discrete equivalents to the integrability conditions. In contrast, Wang, Liu, and Tong [WLT12] showed that edge lengths and dihedral angles can also be understood as equivalents of the first and second fundamental form for discrete frame fields anchored at triangles. They develop discrete integrability conditions—akin to the Gauß–Codazzi equations—that guarantee the existence of an immersion admitting given edge lengths and dihedral angles. Already before, for example in [WDAH10; FB11], edge lengths and dihedral angles have been used as shape coordinates in geometry processing problems. We will pick up this line of work in Chapter 4 and show that discrete integrability conditions can be used to effectively phrase geomet-

ric variational problems in edge lengths and dihedral angles and how they can be used to represent the space of discrete shells as an implicit submanifold. More recently, Ambellan, Zachow, and von Tycowicz [AZv19a] used the same transition rotations as Wang, Liu, and Tong [WLT12] and a matrix representation of metric distortion as shape coordinates for a shape space construction applied to medical shape analysis.

Shape Spaces. Kendall [Ken84] introduced the concept of Riemannian shape spaces as means to analyze collection of shapes. Since then, shape spaces have proven useful for applications in areas such as computer graphics [KMP07; HRWW12; WLX+18] and vision [XJKS14], medical imaging [KKG+11; SKSC14; KXSC16; BKR18], computational biology [LKSM14], and computational anatomy [MTY06; KKD+11]. The textbook by Younes [You10] gives some overview of this subject and we will discuss here some lines of work related to this thesis.

For the shape space of immersions of continuous surfaces, Kurtek *et al.* [KKG+11; KKD+11] introduced a metric based on local area distortions of a parametrization and a corresponding explicit construction to make this metric invariant to reparametrization and rigid body motions. Bauer, Harms, and Michor [BHM11; BHM12c; BHM12b; BHM12a] generalized weighted L^2 and Sobolev metrics from work on spaces of curves to the space of surfaces. Concretely, it builds on work for the space of curves summarized in [MM07], which also observed the importance of higher-order terms in the metric. In [BHM20], this construction was extended to fractional Sobolev metrics. An alternative approach to construct a metric on the spaces of curves and surfaces was pursued by Fuchs *et al.* [FJSY09], who used an linear elasticity energy on enclosed volumes to define such a metric.

The first construction for a metric on the space of immersions of a discrete surface was given by Kilian, Mitra, and Pottmann [KMP07]. They introduced a metric measuring the change of edge lengths and show applications of geodesics and the exponential map in computer graphics. However, this (pseudo-)metric has a significant kernel due to the lack of a term penalizing bending deformations, which leads to wrinkling effects and necessitated an additional regularization. Liu *et al.* [LSDM10] proposed a metric measuring the change of edge length and edge direction on arbitrary simplicial complexes which does not have this problem. In an alternative approach, Heeren *et al.* [HRWW12; HRS+14] introduced a physically-based metric reflecting viscous dissipation defined from a discrete shell elasticity model via Rayleigh's paradigm. This leads to the shape space of discrete shells, which will be the primary shape of interest in this thesis. To this end, we will discuss their approach in more detail in Chapter 3. Tycowicz *et al.* [TAMZ18] and Ambellan, Zachow, and von Tycowicz [AZv19b] introduced shape spaces of discrete surfaces by equipping gradient-based shape coordinates as explained above with a Lie group-structure and Ambellan, Zachow, and von Tycowicz [AZv19a] took a similar approach for the combination of transition rotations and metric distortions.

Other examples of shape spaces include spaces of images. Riemannian structures for this include the flow of diffeomorphism model [DGM98] and its extension to allow image intensity changes, the metamorphosis model [MY01; TY05a; TY05b]. The flow of diffeomorphism has recently also been extended by Miller, Tward, and Trouvé [MTT22] to so-called image varifolds representing functional biological data using measures. For general measures, optimal transport theory can also be used to introduced a Riemannian structure as, for example, discussed in [PPKC10]. Wirth *et al.* [WBRS09] considered a space of volumetric objects with a metric based on viscous dissipation. For applications in shape optimization, Herzog and Loayza-Romero [HL22] equipped the space of planar triangulations with a complete Riemannian metric using an approach similar to the one we will discuss in Chapter 6.

To compute geodesics on shape spaces, one needs to solve the so-called geodesic equation which is a system of ordinary differential equation (ODE). This requires numerical integration techniques as for example in [BMTY05] or a closed form solution which is rarely available. Wirth *et al.* [WBRS09] introduced a variational time-discretization as used for Hamiltonian mechanical systems

an alternative approach to compute geodesics. They discretize the path energy in time and minimize this discretization subject to given boundary points. This was extended to a comprehensive time-discrete geodesic calculus on shape spaces by Rumpf and Wirth [RW15], who also developed a corresponding convergence analysis for vanishing time steps. This time-discretization has been successfully applied to spaces of images [BER15], measures [MRSS15], viscous volumetric objects [RW13], and—most relevant for this thesis—discrete surfaces [HRS+14]. We will give a brief introduction to it in Section 3.2.

Shape Interpolation. Shape interpolation, also called shape blending or morphing, is an important problem in geometry processing used for applications such as deformation transfer [BVG09; YGL+18], motion processing [PKC+16], example-based methods for shape editing [FB11], inverse kinematics [SP04; Wam16], and material design [MTGG11].

Many approaches to shape interpolation derive from the works on shape coordinates and shape spaces we have discussed above. When working with linear shape coordinates, typically nonlinear interpolation schemes are employed. For example, the gradient-domain approach of Xu *et al.* [XZWB06] extracts the rotational components from deformation gradients via polar decomposition and applies nonlinear blending operations to these components. While this helps to compensate linearization artifacts, estimating local rotations that resolve large deformations is an arduous task. Therefore, Kircher and Garland [KG08] and Gao *et al.* [GLL+16] introduced improved nonlinear blending operations for the rotational components. Also the shape space approaches by Kilian, Mitra, and Pottmann [KMP07] and Heeren *et al.* [HRWW12] yield shape interpolation schemes via the computation of geodesics. Based on the space of discrete shells, Heeren *et al.* [HRS+16] also introduced an approach to compute keyframe interpolations using Riemannian splines. Brandt, von Tycowicz, and Hildebrandt [BvH16] derived a discrete curve shortening flow in shape space introduced by Heeren *et al.* [HRS+14] and use it for efficiently processing animations of deformable objects. In contrast, for nonlinear shape coordinates, often linear blending schemes are used to compensate for the cost of nonlinear reconstruction. For example, Winkler *et al.* [WDAH10] introduced a scheme for shape interpolation by linearly blending edge lengths and dihedral angles. They use a multi-scale shape matching algorithm for constructing the immersions of interpolating shapes. Alternatively, Fröhlich and Botsch [FB11] modeled the process of finding the shape that best matches the blended lengths and angles as a nonlinear least-squares optimization problem and solve it using a multi-resolution Gauß–Newton scheme. A related approach by Wuhler *et al.* [WBS+10] blends edge lengths and the normal vectors of two example shapes and constructs the intermediate shapes using a mesh traversal algorithm based on a minimal spanning tree with dihedral angle differences as weights. Other shape interpolations rely on discretizations of ambient space. For example, Eisenberger and Cremers [EC20] and Eisenberger, Löhner, and Cremers [ELC19] used eigenfunctions of differential operators for this purpose.

Shape Statistics. Statistical shape modeling is a rich field as can, for example, be witnessed by the review of Brunton *et al.* [BSBW14]. Hence, we will restrict here to discussing some relevant work on statistical methods on Riemannian manifolds and a particular line of work dealing with sparse modes of shape variation related to our approach in Chapter 5.

Regarding statistics on Riemannian manifold, the Riemannian center of mass [Kar77] of the data points yields a zeroth order analysis. Pennec [Pen06] extended this to covariance measures and higher order moments of data on Riemannian manifolds via parametrizations over the tangent space. In a similar direction, Principal Geodesic Analysis (PGA) was developed by Fletcher *et al.* [FLPJ04] as a generalization of Principal Component Analysis (PCA) to Riemannian manifolds. It constructs data approximating submanifolds and has become the prototypical method for Riemannian statistics. We will discuss it in more detail in Section 5.1. For example, Freifeld and Black [FB12] used PGA on a Lie group representation of a triangle mesh to develop a statistical model of

humanoid shapes. Close to our approach in Chapter 5, Heeren *et al.* [HZRS18] developed a variation of PGA for the space of discrete shells that includes careful handling of the rigid body motion invariance. We will reformulate this in our Nonlinear Rotation-Invariant Coordinates (NRIC), which will simplify the formulation and will allow us to consider the sparsity of variations

Regarding sparse modes of variation, Neumann *et al.* [NVW+13] included a sparsity inducing term in the computation of the principal components resulting in so-called Sparse Localized Deformation Components (SPLOCS). However, they performed the modified PCA on nodal positions which leads to linearization artifacts for larger deformations. To counteract these artifacts, Huang *et al.* [HYZ+14] represented the components in gradient-based shape coordinates and treated rotational components nonlinearly. Gao *et al.* [GLY+21] adapted it to the coordinates proposed in [GLL+16]. SPLOCS was formulated in the NRIC representation of meshes [WLZH17; LLW+19], which results in an improved analysis for articulated shapes. In contrast to our approach, the NRIC representation is treated as a linear space limiting its approximation quality for deformations outside the dataset. To mitigate this issue, Wang *et al.* [WLZ+21] introduced a combination of spaces for different parts of the mesh and a neural-network based correction of approximation error. Recently, Tan *et al.* [TZY+21] also applied this sparsity concept to autoencoders.

Model Reduction. Model order reduction refers to techniques that reduce the computational cost of numerical simulation. As such, a wide array of approaches has been developed for various applications. For example, various techniques for PDE-constrained optimization are discussed in [BSV14]. Here, we will only discuss a few works that motivated the study of submanifold construction as presented later in the thesis.

Tycowicz *et al.* [TSSH13] constructed a model reduction approach for the efficient simulation of nonlinear elastic models. They construct a linear subspace by modal analysis and efficient approximation of elastic energies and their derivatives on this space by a scheme similar to [AKJ08]. This was extended by Tycowicz *et al.* [TSSH15] to shape space problems particularly focusing on efficient computation of means. Afterwards, Radziewsky *et al.* [RESH16] proposed a different construction for the linear subspace approximating given example solutions. Furthermore, Brandt, von Tycowicz, and Hildebrandt [BvH16] used a similar construction in their discrete curve shortening flow to achieve efficient computations. Recently, also neural network-based autoencoders are increasingly used for model order reduction. One example of this related to the previously discussed works is [FMD+19].

Also the Riemannian submanifold constructions above can be used as model order reduction techniques if the submanifold has an efficient parametrization. Many of the constructions are parametrized via the Riemannian exponential methods whose computation often entails nonlinear problems. Hence, their applicability for effective model reduction is limited. Our submanifold construction in Chapter 4 will allow for such an efficient parametrization by imposing additional structure on the submanifold.

1.1.2 Shape Optimization

Shape optimization is a vast field of mathematical and applied studies and we will discuss here a selection of works particularly relevant to this thesis. Azegami [Aze20] provides an introduction to shape optimization problems in general considering mathematical and applied perspectives.

Bilevel Shape Optimization. Bilevel optimization originates from work by Stackelberg [Sta34] on market structures. The first mathematical formalization of this idea was developed by Bracken and McGill [BM73] and has since been mainly used in economy-driven decision making and, recently, an increasing number of other areas. Among the many applications of bilevel optimization, there are some works that study applications in the context of elastic shape optimization. For example,

Herskovits *et al.* [HLDS00] investigated the design of an elastic object, where contact to a rigid obstacle supporting the object is only possible at certain parts of the domain. Kočvara and Outrata [KO95; Koč97] considered a model to optimize the design parameters of a truss structure where the follower problem amounts to computing the displacement under loads. Zuo [Zuo15] investigated an optimistic bilevel problem for car design, where design parameters are optimized on both levels: on the lower level, the mass distribution along the body frame and, on the upper level, the shape of shell segments of the hull.

The stochastic extension of bilevel optimization was first formulated in its optimistic version by Patriksson and Wynter [PW99]. They were further studied, for example, by Burtscheidt, Claus, and Dempe [BCD20], who consider stochastic linear bilevel problems primarily in the optimistic regime but their results also extend to the pessimistic regime. Christiansen, Patriksson, and Wynter [CPW01] and Martínez-Frutos *et al.* [MHKP18] applied this to stochastic optimization of elastic structures using a truss respectively a level set model. However, all these applications to elastic shape optimization are either a so-called optimistic formulation or consider the problem of determining displacements under given forces as the lower-level problem using a linear elastic model, which means that it has a unique solution. In contrast, the follower will play the role of a test engineer in Chapter 7 and optimize the forces applied to an elastic structure, which leads to a more involved optimization problem.

For a more detailed overview of possible applications of bilevel optimization and its stochastic extensions, we refer to the review by Sinha, Malo, and Deb [SMD18].

Phase-fields. The diffuse model of phase-fields has a long history ranging back to late 19th century (cf. [Row79]) with important contributions by Cahn, Hilliard, and Allen [CH58; AC72] in the middle of the 20th century, who developed the model in a modern language. They can be used to describe the segmentation via a smooth function that intuitively models the mixture of two materials. Modica and Mortola [MM77] rigorously investigated the behavior of these models for vanishing interface width, which results in a Γ -convergence result of the so-called Modica–Mortola-functional to the interface length. Due to the Γ -convergence result, phase-fields yield an effective approximation of the perimeter, which has been used extensively for image segmentation [BCM04], approximating motion by mean curvature [DF20], shape optimization [BC03], simulation of material processes [Che03], and many other applications.

Phase-fields—or comparable diffuse interface models—on surfaces have already been used, for example, by Hornung, Rumpf, and Simon [HRS20] who used phase-fields to diffusely represent hard and soft material on the surface and optimize their distribution. Dziuk and Elliott [DE13] considered the Allen–Cahn equation for the approximation of motion by mean curvature using phase-fields on surfaces as one example for a surface finite element method. Du, Ju, and Tian [DJT11] and Elliott and Ranner [ER15] studied the Cahn–Hilliard equation on (evolving) surfaces and its numerical approximation using finite elements. Zayer *et al.* [ZMSS19] and Stadlbauer *et al.* [SMS+20] used a diffuse representation to compute Voronoi-type diagrams by growing segments from given start points. Even more general, Bertozzi and Flenner [BF12] used phase-field models on arbitrary graphs and combine perimeter minimization with data-based fitting terms to address classification problems and developed a comprehensive theory for this in follow-up work.

Mesh Segmentation. Most approaches to mesh segmentation are concerned with splitting or cutting the mesh along its edges. However, computing exact minimizers for even basic task such as finding the shortest cuts to obtain a topological disk [EH04] or a partition into equally-sized convex parts with minimal interface [CDST97] are NP-hard. Hence, approximate solutions and algorithms to compute them are investigated.

With respect to atlas generation and related applications, curvature-based objectives have been used to compute segmentations. For example, Yamauchi *et al.* [YGZS05] optimized for an even dis-

tribution of Gauß curvature among the segments while Julius, Kraevoy, and Sheffer [JKS05] optimize the fit of conics to segments. Both use their objectives as proxy for developability. In contrast, Lien and Amato [LA07] and Mamou and Ghorbel [MG09] computed decomposition into approximately convex parts by measuring the distance between them and their convex hull.

A very popular approach to the hierarchical segmentation of meshes was introduced by Garland, Willmott, and Heckbert [GWH01], who hierarchically cluster the faces of a mesh by edge contractions in the dual graph using the planarity of resulting clusters as primary criterion. Attene, Falcidieno, and Spagnuolo [AFS06] extended this by taking into account how well clusters can be approximated by certain geometric primitives (e.g. planes or cylinders). Furthermore, Katz and Tal [KT03] used a probabilistic formulation and combine this with a recursive approach. Lai *et al.* [LZHM06] used a similar approach, but add semantic objectives based on the change in texture and use a Lloyd's-type clustering algorithm.

Cutting and Flattening. For problems such as texture mapping, parametrizations of the surface with low area distortion are of great interest. This motivates the study of algorithms to cut and flatten—i.e. parametrize—surfaces. Many of the curvature-based segmentation methods listed above draw their inspiration from this problem. Zhou *et al.* [ZSGS04] used nonlinear dimensionality reduction methods based on spectral analysis as initialization for a nonlinear optimization of the charts. Other approaches focus on placing special vertices, e.g. cone singularities, and then connecting them with cuts. To reduce the distortion, some place them via a greedy ansatz [SSP08; BGB08; ZYCF20], while others follow a global shape optimization technique [SSC18]. Departing from greedy approaches and other heuristics, Poranne *et al.* [PTH+17] and Li *et al.* [LKK+19] computed patches that minimize the associated distortion. To this end, they explicitly compute the parametrization of different patches and simultaneously optimize their distortion and boundaries. Going further, Sharp and Crane [SC18] used the Yamabe equation to compute the lowest possible distortion and thus circumvent the need to compute parametrizations during the optimization. To optimize over surface patches to be flattened, they use a level set approach to describe the shape of the patches and solve the shape optimization problem of minimal distortion. In particular, they allow cuts not aligned with triangle edges. We will follow a similar approach using phase-fields.

1.2 Collaborations and Publications

The work presented in this thesis is the result of many fruitful collaborations and has been partially published in a set of joint publications:

The results on nonlinear shape coordinates presented in Chapter 4 are joint work with Behrend Heeren, Klaus Hildebrandt (TU Delft), and Martin Rumpf. They were published in [SHHR20] while some preliminary results were already published in [SHHR19]. I contributed to developing a representation of the space of discrete surfaces as an implicit submanifold via the discrete integrability conditions and investigated as well as implemented efficient numerical methods for variational problems phrased in this representation.

The work on the construction of data approximating submanifolds of the space of discrete shells is the result of a collaboration with Klaus Hildebrandt (TU Delft), and Martin Rumpf published in [SHR20]. The network-based parametrization is additionally the result of joint work with also Benedikt Wirth (University of Münster) and published in [SHWR23]. I contributed to the modification of Principal Geodesic Analysis to obtain sparse modes, the approach to equipping the resulting submanifold with a product structure, the grid-based approximation of the exponential map on these factors, and the numerical methods for the approach. Furthermore, I worked on the structural observations for the neural network-based parametrization and its implementation along with conducting the corresponding numerical experiments.

The work on constructing a metric for the space of discrete shells guaranteeing embeddings as presented in Chapter 6 is joint work with Keenan Crane (Carnegie Mellon University), Martin Rumpf, and Henrik Schumacher (TU Chemnitz) that was not published so far. Preliminary results appeared in the non-peer-reviewed report [Sas22]. I contributed to the construction of the metric, the application of the variational time-discretization to the resulting shape space, and the formulation and implementation of numerical optimization methods for the resulting nonlinear problems. The employed discretization of the tangent-point energy was primarily developed by H. Schumacher and thus we will only give an overview of this for the sake of completeness.

The application of bilevel optimization in the context of elastic shape optimization presented in Chapter 7 is the result of joint work with Johanna Burtscheid (University of Duisburg-Essen), Matthias Claus (University of Duisburg-Essen), Sergio Conti, Martin Rumpf, and Rüdiger Schultz (University of Duisburg-Essen) published in [BCC+21]. I contributed to the mathematical modeling of the bilevel shape optimization and its application to discrete shells, developed and implemented the numerical optimization approach, and performed the numerical experiments using it. The existence results for the bilevel problems are primarily the work of J. Burtscheid, M. Claus, S. Conti, and R. Schultz and as such only a summary of them will be presented in this thesis.

The investigation of a phase-field model for surface segmentation in Chapter 8 is joint work with Janos Meny and Martin Rumpf that was published in [MRS21]. I contributed to the mathematical modeling of the segmentation problem, the diffuse adaption of the Yamabe equation, and to their discretization. Notably, J. Meny implemented the method and performed the numerical experiments. Nevertheless, we will report on them in this thesis to give practical insights into the method.

Moreover, I contributed to two further publications, that are not considered in this thesis:

- Sandrine H. Künzel, Moritz Lindner, Josua Sassen, Philipp T. Möller, Lukas Goerdt, Matthias Schmid, Steffen Schmitz-Valckenberg, Frank G. Holz, Monika Fleckenstein, and Maximilian Pfau. “Association of Reading Performance in Geographic Atrophy Secondary to Age-Related Macular Degeneration With Visual Function and Structural Biomarkers”. In: *JAMA Ophthalmology* 139.11 (2021), pp. 1191–1199. ISSN: 2168-6165. DOI: 10.1001/jamaophth.2021.3826
- Florine Hartwig, Josua Sassen, Omri Azencot, Martin Rumpf, and Mirela Ben-Chen. “An Elastic Basis for Spectral Shape Correspondence”. under review. 2023

Chapter 2

Surfaces and Elasticity

In this chapter, we will provide the necessary foundations in surface theory, discrete differential geometry, and elasticity necessary for the remainder of the thesis. The goal is that, at the end of the chapter, the reader is equipped with all the notation and key ideas from these fields that will be used afterwards. To this end, these notions will be presented in a very condensed fashion and accompanied with pointers to more comprehensive introductions. A similar chapter was also part of [Sas19].

2.1 Differential Geometry of Surfaces

In this section, we will 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; Car76], to which we also refer for further reading. We begin with the basic definitions of a surface and its tangent space.

Definition 2.1 (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 $\psi: \Omega \rightarrow \mathbb{R}^3$, such that

- (i) $\psi(\Omega) = \mathcal{S} \cap B_\varepsilon(p)$ and $\psi: \Omega \rightarrow \mathcal{S} \cap B_\varepsilon(p)$ 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 2.2 (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

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

Elements of this space are called *tangent vectors*.

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

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

This basis of the tangent space is called the *canonical basis*.

Differentiation. For a smooth function $\varphi: \mathcal{S} \rightarrow \mathbb{R}$ 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.

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

for an arbitrary curve $\gamma: (-1, 1) \rightarrow \mathcal{S}$ with $\gamma(0) = p$ and $\dot{\gamma}(0) = V$. For a vector-valued deformation $\phi: \mathcal{S} \rightarrow \mathbb{R}^3$ 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.

$$D_p\phi: T_p\mathcal{S} \rightarrow T_{\phi(p)}\phi(\mathcal{S}).$$

2.1.1 Fundamental Forms

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 3.

Definition 2.3 (First fundamental form). The *first fundamental form* in $p \in \mathcal{S}$ is given by

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

We see that g_p is a scalar product on the linear space $T_p\mathcal{S}$ 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

$$g_p = \left(\langle \partial_{\xi_i} \psi(\xi), \partial_{\xi_j} \psi(\xi) \rangle_{\mathbb{R}^3} \right)_{ij} \in \mathbb{R}^{2 \times 2},$$

where $p = \psi(\xi)$.

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

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

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 $\varphi: \mathcal{S} \rightarrow \mathbb{R}$ by

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

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 2.5 (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 $n: \mathcal{S} \rightarrow S^2$ 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 $\text{rank}(D\psi) = 2$, 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 2.6 (Shape operator). Let $\mathcal{S} \subset \mathbb{R}^3$ be a regular and orientable surface, $p \in \mathcal{S}$. The *shape operator* $S_p: T_p\mathcal{S} \rightarrow T_p\mathcal{S}$ 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 2.7 (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.

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

We will exploit this bilinear form 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 a linear map with an orthonormal basis and thus it has two real eigenvalues. These eigenvalues now allow us to define different notions of curvature.

Definition 2.8 (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 $H_p := \text{tr } S_p = \kappa_1 + \kappa_2$ and the *Gauß' 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 2.9 (Pulled-back shape operator). The pulled-back shape operator $S_p^*[\phi]: T_p\mathcal{S} \rightarrow T_p\mathcal{S}$ is given by

$$g_p \left(S_p^*[\phi] U, V \right) = h_{\phi(p)} (D_p \phi U, D_p \phi V), \quad \forall U, V \in T_p\mathcal{S}. \quad (2.1)$$

Definition 2.10 (Relative shape operator). The *relative shape operator* $S_p^{\text{rel}}[\phi]$ is defined as the point-wise difference, i.e.

$$S_p^{\text{rel}}[\phi]: T_p\mathcal{S} \rightarrow T_p\mathcal{S}, \quad S_p^{\text{rel}}[\phi] := S_p - S_p^*[\phi]. \quad (2.2)$$

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

$$s_\xi^*[\phi] = g_\xi^{-1} \tilde{h}_\xi, \quad s_\xi^{\text{rel}}[\phi] = s_\xi - s_\xi^*[\phi] = g_\xi^{-1}(h_\xi - \tilde{h}_\xi). \quad (2.3)$$

In the physical modeling of surface deformations later on, we will ask for a criterion when two surfaces will be congruent, i.e. only differ by a rigid transformation. Furthermore, we will also desire a way to parametrize surfaces up to rigid body motions. In this section, we will see 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 2.11 (Standard frame). Let \mathcal{S} be a regular surface with parametrization $\psi: \Omega \rightarrow \mathbb{R}^3$, then at each coordinate $\xi \in \Omega$ we define the *standard frame* at ξ by

$$F(\xi) = (F_1(\xi), F_2(\xi), F_3(\xi)) := (\partial_1 \psi(\xi), \partial_2 \psi(\xi), (n \circ \psi)(\xi)). \quad (2.4)$$

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

Note, that $F(\xi)$ is a basis of \mathbb{R}^3 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

$$\partial_k F_j(\xi) = \sum_{i=1}^2 P_{ji}^k(\xi) F_i(\xi). \quad (2.5)$$

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

$$\partial_1 F = F P^1 \quad (2.6a)$$

$$\partial_2 F = F P^2, \quad (2.6b)$$

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

In the following, we will see 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 2.12 ([Pal03]). Let \mathcal{S} be a regular surface with parametrization $\psi: \Omega \rightarrow \mathbb{R}^3$, and P^1, P^2 the transition matrices as defined above. Then

$$P^1 = G^{-1} A^1 := \begin{pmatrix} g^{11} & g^{12} & 0 \\ g^{21} & g^{22} & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \frac{1}{2} \partial_{\xi_1} g_{11} & \frac{1}{2} \partial_{\xi_2} g_{11} & h_{11} \\ \partial_{\xi_2} g_{12} - \frac{1}{2} \partial_{\xi_1} g_{11} & \frac{1}{2} \partial_1 g_{22} & h_{12} \\ -h_{11} & -h_{12} & 0 \end{pmatrix} \quad (2.7a)$$

$$P^2 = G^{-1} A^2 := \begin{pmatrix} g^{11} & g^{12} & 0 \\ g^{21} & g^{22} & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \frac{1}{2} \partial_2 g_{11} & \partial_2 g_{12} - \frac{1}{2} \partial_1 g_{22} & h_{12} \\ \frac{1}{2} \partial_1 g_{22} & \frac{1}{2} \partial_2 g_{22} & h_{22} \\ -h_{12} & -h_{22} & 0 \end{pmatrix}. \quad (2.7b)$$

The proof essentially requires a careful calculation. 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 proposition above. From this, we can deduce necessary conditions on the coefficients on the first and second fundamental form to stem from a regular surface.

Corollary 2.13 (Gauß–Codazzi Equations, [Pal03]). *If (g_{ij}) and (h_{ij}) are the coefficients of the first and second fundamental forms of a regular surface S with parametrization $\psi: \Omega \rightarrow S$, then the matrix-valued functions P^1 and P^2 defined on Ω by Proposition 2.12 satisfy*

$$\partial_2 P^1 - \partial_1 P^2 = P^1 P^2 - P^2 P^1. \quad (2.8)$$

This again follows from a straightforward calculation. Next, we see 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. Hence, one has to show that just given the coefficients we can construct a surface with those fundamental forms unique up to rigid body motions.

Theorem 2.14 (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 $g: \Omega \rightarrow \mathbb{R}^{2 \times 2}$, and $h: \Omega \rightarrow \mathbb{R}^{2 \times 2}$ 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 $\psi: \Omega \rightarrow S \subset \mathbb{R}^3$ with g and h as (matrix representations of) first and second fundamental form.

The principal idea of the proof is that the Gauß–Codazzi equations are exactly the integrability conditions of the Frobenius’ theorem, which allows us to integrate the local change of geometry encoded by the fundamental forms to obtain a regular surface.

2.2 Discrete Surfaces

In this section, we will study discrete surfaces, a certain class of polygonal meshes. These are the objects of central interest in this thesis and we want to transfer the differential and geometric notions from continuous surfaces to them. However, they required the surface to be sufficiently smooth while polygonal meshes are piecewise affine. In *discrete differential geometry* (DDG), one aims to introduce discrete equivalents to these notion allowing to compute approximations of a smooth surface’s properties. 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 DDG. For this, we refer the reader to existing literature such as [Hee17; CdDS13] 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 $\mathbf{V} = \{v_1, \dots, v_{|\mathbf{V}|}\}$ and a set of triangle faces $\mathbf{T} = \{\tau_1, \dots, \tau_{|\mathbf{T}|}\} \subset \mathbf{V} \times \mathbf{V} \times \mathbf{V}$. From these two, we can furthermore deduce a set of edges $\mathbf{E} = \{e_1, \dots, e_{|\mathbf{E}|}\} \subset \mathbf{V} \times \mathbf{V}$. Note, that we could equivalently deduce the set of faces \mathbf{T} from the set of edges \mathbf{E} and all further structural properties of the mesh can be derived from them. We will now derive a topology for the mesh solely based on this connectivity. We begin with introducing the topological notions that correspond to our connectivity.

Definition 2.15 (Abstract simplicial complex). *An abstract simplicial complex \mathcal{K} consists of a set of vertices \mathbf{V} together with a set Δ of finite non-empty subsets of \mathbf{V} , called *simplices*, such that*

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

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

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 $\mathcal{S}_h = (\mathbf{V}, \mathbf{E}, \mathbf{T})$ of a triangle mesh is equivalent to giving a set of vertices \mathbf{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. Now that we have the combinatorial structure of a simplicial complex, we will associate to it a topological space.

Definition 2.16 (Geometric realization). Let \mathcal{K} be an abstract simplicial complex with finitely many vertices $\mathbf{V} = \{v_1, \dots, v_{|\mathbf{V}|}\}$. Then we define the *geometric realization* $|\mathcal{K}|$ of \mathcal{K} as the set

$$\bigcup_{\{v_{i_1}, \dots, v_{i_K}\} \in \Delta} \text{Conv}(e_{i_1}, \dots, e_{i_K}) \subset \mathbb{R}^{|\mathbf{V}|} \quad (2.9)$$

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

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

Lastly, we have ensure sufficient regularity of the topological space to get to the notion of a discrete surface. Note, that the following definition is equivalent to the one used for example by Desbrun, Kanso, and Tong [DKT08], just derived in a more abstract fashion.

Definition 2.17 (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.

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

We will not always be able to only consider embeddings of a discrete surface due to their global injectiveness. In fact, ensuring embeddedness will be one of the principal goals of Chapter 6. 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.

Definition 2.19 (Immersion). Let \mathcal{S}_h be a discrete surface. An *immersion* of \mathcal{S}_h is a piecewise linear map $X: |\mathcal{S}_h| \rightarrow \mathbb{R}^3$ 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* \mathbf{S} .

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

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

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

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

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

In this thesis, we will also use other maps defined on the different elements of such a mesh, which can be written as vectors. For example, a map $w: \mathbf{V} \rightarrow \mathbb{R}^k$ assigning each vertex a value in this section corresponds to a vector $\mathbb{R}^{k|\mathbf{V}|}$ and similarly for functions defined on edges and faces. We denote evaluations $w(v)$ of such a map also via indexing to simplify notation, i.e. $w_v := w(v) \in \mathbb{R}^k$. This notation will allow us to directly write down products of such maps with matrices, which correspond to linear operators from the function point-of-view.

Parametrization. In Section 2.1, 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

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

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

$$\Psi_\tau: \omega \rightarrow T(\tau), \quad (\xi_1, \xi_2) \mapsto \xi_1 X_1(\tau) + \xi_2 X_2(\tau) + (1 - \xi_1 - \xi_2) X_0(\tau)$$

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

$$\Psi: \Omega_h \rightarrow X(\mathcal{S}_h), \quad (\xi, \tau) \mapsto \Psi_\tau(\xi)$$

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

2.2.1 Fundamental Forms

In Section 2.1, 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 $g = D\psi^\top D\psi$. Furthermore, the local parametrization Ψ of an immersed discrete surface is affine, thus its derivative is constant on each triangle $\tau \in \mathbf{T}$ and given by

$$D\Psi|_\tau = \left(\frac{\partial \Psi_\tau}{\partial \xi_1}, \frac{\partial \Psi_\tau}{\partial \xi_2} \right) = \left[X_1(\tau) - X_0(\tau) \mid X_2(\tau) - X_0(\tau) \right] \quad (2.10)$$

$$= \left[E_2(\tau) \mid -E_1(\tau) \right] \in \mathbb{R}^{3 \times 2}. \quad (2.11)$$

From this the definition of a discrete first fundamental form follows as

Definition 2.20 (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

$$G|_\tau := (D\Psi|_\tau)^\top D\Psi|_\tau = \begin{pmatrix} \|E_2(\tau)\|^2 & -\langle E_1(\tau), E_2(\tau) \rangle \\ -\langle E_1(\tau), E_2(\tau) \rangle & \|E_1(\tau)\|^2 \end{pmatrix} \in \mathbb{R}^{2 \times 2} \quad (2.12)$$

for each $\tau \in \mathbf{T}$.

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 $S|_\tau := G|_\tau^{-1} H|_\tau \in \mathbb{R}^{2 \times 2}$.

First, we need to introduce some basic geometric notions.

Definition 2.21 (Normals). For an immersed discrete surface \mathbf{S} , we define the face normal $N_\tau \in S^2$ on the face $\tau = (v_{i_0}, v_{i_1}, v_{i_2}) \in \mathbf{T}$ by

$$N_\tau := \frac{(X_{i_1} - X_{i_0}) \times (X_{i_2} - X_{i_0})}{\|(X_{i_1} - X_{i_0}) \times (X_{i_2} - X_{i_0})\|}. \quad (2.13)$$

Furthermore, for a vertex v , we define a vertex normal as

$$N_v := \frac{\sum_{\tau \in \mathbf{N}_v} N_\tau}{\|\sum_{\tau \in \mathbf{N}_v} N_\tau\|}, \quad (2.14)$$

for the faces adjacent to the vertex \mathbf{N}_v .

Definition 2.22 (Dihedral angle). For an immersed discrete surface \mathbf{S} , we define the dihedral angle of an edge e with adjacent faces τ_l and τ_r as

$$\theta_e := \begin{cases} \arccos(N_{\tau_l} \cdot N_{\tau_r}) & \text{if } \langle N_{\tau_l} \times N_{\tau_r}, E \rangle_{\mathbb{R}^3} > 0 \\ -\arccos(N_{\tau_l} \cdot N_{\tau_r}) & \text{else,} \end{cases} \quad (2.15)$$

where, as before, E denotes the embedded edge shared by the triangles.

Now, combining Definition 2.7 and (2.10) with observations from discrete exterior calculus (cf. [DKT08]) one can establish the following definition of a triangle-averaged second fundamental form.

Definition 2.23 (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

$$H|_\tau = -4a_\tau \sum_{i=0}^2 \frac{\cos \frac{\theta_i + \pi}{2}}{\|E_i\|} M_i \quad (2.16)$$

for each $\tau \in \mathbf{T}$, where (M_0, M_1, M_2) is the basis of symmetric 2×2 -matrices given by

$$M_0 = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix}, \quad M_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \quad M_2 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}. \quad (2.17)$$

This is one possible choice of discrete second fundamental form, proposed by Wardetzky in unpublished work and elaborated by Heeren [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 2.24 (Discrete shape operator, [Hee16]). Let S_h be a discrete surface with immersion X . The elementwise constant triangle-averaged *discrete shape operator* is given by

$$S|_\tau = G|_\tau^{-1} H|_\tau \quad (2.18)$$

for each $\tau \in \mathbf{T}$.

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 $\text{tr } S|_\tau$ with explicit formula

$$\text{tr } S|_\tau = - \sum_{i=0}^2 \frac{\cos \frac{\theta_i + \pi}{2}}{a_\tau} \|E_i\|. \quad (2.19)$$

However, we typically do not use the determinant of the triangle-averaged discrete shape operator as Gauß curvature, as it is more naturally associated with vertices. That means for some area A_v associated with some vertex $v \in \mathbf{V}$ one defines an *integrated* Gauß curvature by the so-called *angle-defect*, i.e.

$$\int_{A_v} K_h(x) da := 2\pi - \sum_{\tau: v \in \tau} \gamma_{\tau, v}, \quad (2.20)$$

where $\gamma_{\tau, v}$ denotes the interior triangle angle in τ at vertex v . One verifies immediately a discrete Gauß–Bonnet theorem, by computing

$$\begin{aligned} \int_{S_h} K_h(x) da &= \sum_{v \in \mathbf{V}} \int_{A_v} K_h(x) da = 2\pi |\mathbf{V}| - \sum_{v \in \mathbf{V}} \sum_{\tau: v \in \tau} \gamma_\tau \\ &= 2\pi (|\mathbf{V}| - \frac{1}{2} |\mathbf{T}|) = 2\pi (|\mathbf{V}| + |\mathbf{T}| - |\mathbf{E}|), \end{aligned}$$

which justifies the vertex-based definition of the Gauß curvature, see also [CM03].

2.3 From Nonlinear Elasticity to Discrete Shells

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.* [HRWW12; HRS+14] and Heeren [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 $\delta \rightarrow 0$ based on the notion of Γ -convergence. Qualitative insights from the mathematical rigorous study of this limit in [LR95; LR96; FJM02; FJMM03] were then used to define a generic thin shell model.

Elasticity. We begin with introducing our notation for three-dimensional elasticity 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 non-linear deformation. We assume that ϕ is orientation preserving—i.e. $\det D\phi(x) > 0$ for all $x \in \mathcal{O}$ —and

injective. We then postulate that we have a *hyperelastic* material with a *deformation energy* given as integral of an elastic energy density W , i.e.

$$\mathcal{W}[\phi, \mathcal{O}] = \int_{\mathcal{O}} W(D\phi) dx. \quad (2.21)$$

A fundamental axiom of continuum mechanics is the *frame indifference* of W , i.e. we have $W(D\phi) = W(Q^T D\phi Q)$ 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 $\|D\phi\|_F$, $\|\text{cof } D\phi\|_F$, 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 $W(D\phi) \rightarrow \infty$ for $\det D\phi \leq 0$, hence $W(D\phi) = \infty$ for $\det D\phi \leq 0$.

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

$$W(D\phi) = \frac{\mu}{2} \|D\phi\|_F^2 + \frac{\lambda}{4} (\det D\phi)^2 - \left(\mu + \frac{\lambda}{2}\right) \log \det D\phi - \mu - \frac{\lambda}{4}, \quad (2.22)$$

which is a concrete instance of a Mooney-Rivling model [Cia90] and fulfills all our assumptions.

2.3.1 Membrane and Bending Energies

Now, that we have established basic notions, we turn towards the model of Heeren *et al.* [HRWW12], 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.

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

$$\mathcal{G}[\phi] = g^{-1} g_\phi \quad (2.23)$$

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}_{mem} from the results of Le Dret and Raoult [LR95; LR96]. It is supposed to measure the tangential stretching and shearing induced by a deformation ϕ of the surface \mathcal{S} .

Definition 2.25 (Membrane energy). Let \mathcal{S} be a regular surface and ϕ a deformation. Then the *membrane energy* is given by

$$\mathcal{W}_{\text{mem}}[\mathcal{S}, \phi] = \delta \int_{\mathcal{S}} W_{\text{mem}}(\mathcal{G}[\phi]) da, \quad (2.24)$$

with density (2.22).

Bending energy. For the bending part of the model, Heeren *et al.* used results from Friesecke, James, and Müller [FJM02] and Friesecke *et al.* [FJMM03], 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 2.26 (Bending energy). Let \mathcal{S} be a regular surface and ϕ a deformation. Then the *bending energy* is given by

$$\mathcal{W}_{\text{bend}}[\mathcal{S}, \phi] = \delta^3 \int_{\mathcal{S}} W_{\text{bend}}(S_{\phi}^{\text{rel}}) da, \quad (2.25)$$

where in general we make use of the density

$$W_{\text{bend}}(A) = \alpha (\text{tr } A)^2 + (1 - \alpha) \|A\|_F^2, \quad \alpha \in \{0, 1\}. \quad (2.26)$$

Recall, that the matrix representation of the relative shape operator in the parameter domain Ω was defined in Section 2.1.1 as

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

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

$$\mathcal{W}_{\text{bend}}[\mathcal{S}, \phi] = \delta^3 \int_{\mathcal{S}} \|S_{\phi}^{\text{rel}}\|_F^2 da = \delta^3 \int_{\Omega} \text{tr} \left(s_{\xi}^{\text{rel}}[\phi]^2 \right) \sqrt{\det g} d\xi, \quad (2.27)$$

and for $\alpha = 1$ one gets

$$\mathcal{W}_{\text{bend}}[\mathcal{S}, \phi] = \delta^3 \int_{\mathcal{S}} \left(\text{tr } S_{\phi}^{\text{rel}} \right)^2 da = \delta^3 \int_{\Omega} \left(\text{tr } s_{\xi}^{\text{rel}}[\phi] \right)^2 \sqrt{\det g} d\xi. \quad (2.28)$$

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

$$\mathcal{W}_{\mathcal{S}}[\phi] = \int_{\mathcal{S}} \delta W_{\text{mem}}(\mathcal{G}[\phi]) + \delta^3 W_{\text{bend}}(S_{\phi}^{\text{rel}}) da, \quad (2.29)$$

where the weight δ represents the thickness of the shell. Note that $\mathcal{W}_{\mathcal{S}}[\phi]$ is invariant with respect to rigid body motions.

2.3.2 Discrete Deformations Energies

In the following, we will consider deformations Φ of immersed discrete surfaces and present a discretization of the above model 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 2.27 (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 2.28 (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 $X: \mathcal{S}_h \rightarrow \mathbb{R}^3$ and $\tilde{X}: \mathcal{S}_h \rightarrow \mathbb{R}^3$. Then a *discrete deformation* $\Phi: \mathbf{S} \rightarrow \tilde{\mathbf{S}}$ is the unique piecewise affine map defined by its nodal values $\Phi(X(v)) = \tilde{X}(v)$ for all $v \in \mathbf{V}$.

In the following, we will sometimes refer to connectivity properties of $\tilde{\mathbf{S}}$ with a tilde (e.g. $\tilde{\tau}$) 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 $T(\tilde{\tau})$ for the embedded triangle of the deformed surface when actually the embedded property T differs rather than the connectivity τ .

Discrete membrane model. In Section 2.2.1, we have introduced an elementwise constant first fundamental form (cf. Definition 2.20) and now this is combined with the membrane model (2.24). Hence, to describe tangential distortions induced by Φ , we consider the elementwise constant discrete distortion tensor

$$\mathcal{G}[\Phi]|_\tau = (G|_\tau)^{-1} G^\Phi|_\tau \in \mathbb{R}^{2 \times 2}. \quad (2.30)$$

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 (2.24), one arrives at the following

Definition 2.29 (Discrete membrane energy, [HRWW12]).

$$\mathcal{W}_{\text{mem}}[\mathbf{S}, \tilde{\mathbf{S}}] := \delta \int_{\mathbf{S}} W_{\text{mem}}(\mathcal{G}[\Phi]) da = \delta \sum_{\tau \in \mathbf{T}} a_\tau \cdot W_{\text{mem}}(\mathcal{G}[\Phi]|_\tau), \quad \tilde{\mathbf{S}} = \Phi(\mathbf{S}). \quad (2.31)$$

In this energy, one can continue to use the energy density (2.22), where $\text{tr} G[\Phi]|_\tau$ controls the change of edge lengths and $\det G[\Phi]|_\tau$ controls the local change of triangle area.

Discrete bending model. Next, we will connect the triangle-averaged discrete second fundamental form from Definition 2.23 and the corresponding shape operator with the bending model (2.25) using the density (2.26). By choosing $\alpha = 1$, we can derive a discrete version of the Willmore energy.

Definition 2.30 (Discrete bending energy, [HRWW12]).

$$\mathcal{W}_{\text{bend}}[\mathbf{S}, \tilde{\mathbf{S}}] := \delta^3 \sum_{\tau \in \mathbf{T}} a_\tau \cdot \left(\text{tr}(S_\tau - S_\tau^\Phi) \right)^2, \quad \tilde{\mathbf{S}} = \Phi(\mathbf{S}). \quad (2.32)$$

Heeren [Hee16] showed 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 2.31 (*Discrete Shells* bending energy, [GHDS03; Hee16]).

$$\mathcal{W}_{\text{bend}}^{\text{DS}}[\mathbf{S}, \tilde{\mathbf{S}}] := \delta^3 \sum_{e \in \mathbf{E}} \frac{(\theta_e - \tilde{\theta}_e)^2}{d_e} l_e^2, \quad \tilde{\mathbf{S}} = \Phi(\mathbf{S}), \quad (2.33)$$

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

Gladbach and Olbermann [GO21] recently investigated the convergence of the Discrete Shells energy (2.33) and related energies under spatial refinement for planar reference configurations, i.e. $\theta = 0$. They were able to prove Γ -convergence of the energies to the Willmore energy for strong assumption on the mesh including a Delaunay condition.

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 since deformations are unique to the assumption of dense correspondence.

Definition 2.32 (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

$$\mathcal{W}[\mathbf{S}, \tilde{\mathbf{S}}] = \mathcal{W}_{\text{mem}}[\mathbf{S}, \tilde{\mathbf{S}}] + \mathcal{W}_{\text{bend}}^{\text{DS}}[\mathbf{S}, \tilde{\mathbf{S}}], \quad (2.34)$$

where the *bending weight* δ in the different contributions represents the thickness of the shell. As the deformation is unique, we call this also the *discrete dissimilarity measure*.

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

Part I

Shape Spaces

Chapter 3

Shape Spaces and Time-Discrete Geodesic Calculus

We begin the first part of the thesis with introducing the necessary background on Riemannian geometry. Before, we already discussed the geometry of surfaces and the first section of this chapter can be seen as a generalization of this to manifold of arbitrary dimension. As demonstrated, for example, by Kilian, Mitra, and Pottmann [KMP07], many of the notions we will see in that chapter are useful for applications. Hence, we will discuss afterwards a variational time discretization due to Rumpf and Wirth [RW15] that enables the numerical approximation of these notions. Finally, we will introduce the *space of discrete shells* as central object of study in the remainder of the first part. As in Chapter 2, we will introduce these concepts only briefly and provide references to more exhaustive literature. A similar chapter was also part of [Sas19].

3.1 Riemannian Manifolds

Our goal is to model the shape space as a *Riemannian manifold*, hence we will briefly introduce the needed concepts from Riemannian geometry to read the following chapters. We refer the reader to standard textbooks such as [Car92] and [Lan95] for a more detailed treatment. This section is also based on them as well as on [Hee17].

In this section, everything will be presented on finite dimensional manifolds since the space of discrete surfaces will be a finite dimensional manifold. The geodesic calculus is set up in a way that it also works on infinite dimensional manifolds. Nevertheless, considering this case would require more care and we refer to [Lan95] for more details on infinite dimensional manifolds.

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

Definition 3.1 (Differentiable manifold). A *differentiable manifold* \mathcal{M} of dimension $d < \infty$ is a set together with a family of injective maps $\psi_\alpha: U_\alpha \subset \mathbb{R}^d \rightarrow \mathcal{M}$ 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 .

Typically one assumes that the differentiable structure in Definition 3.1 is maximal with respect to the given conditions. This can 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 $\psi: U \subset \mathbb{R}^d \rightarrow \mathcal{M}$.

Definition 3.2 (Tangent space). The *tangent space* $T_p\mathcal{M}$ of \mathcal{M} at $p \in \mathcal{M}$ is defined as

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

If $\psi: U \subset \mathbb{R}^d \rightarrow \mathcal{M}$ is a parametrization with $\psi(\xi) = p$ for some $\xi \in U$, then $\partial_{\xi_i}\psi$ for $i = 1, \dots, d$ 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 3.3 (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 $g_p: T_p\mathcal{M} \times T_p\mathcal{M} \rightarrow \mathbb{R}$ smoothly varying with $p \in \mathcal{M}$, in the sense that for a parametrization $\psi: U \subset \mathbb{R}^d \rightarrow \mathcal{M}$ the map $\xi \mapsto g_{ij}(\xi) := g_{\psi(\xi)}(\partial_{\xi_i}\psi, \partial_{\xi_j}\psi)$ 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 $g^{-1} \in \mathbb{R}^{d \times d}$, which we denote by $(g^{kl})_{kl}$, i.e. $g_{ij}g^{jk} = \delta_{ik}$. In Section 2.1, we have studied embedded surfaces in \mathbb{R}^3 together with their first fundamental form. Those are in fact two-dimensional Riemannian manifolds with a metric as defined above 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 $y: [0, 1] \rightarrow \mathcal{M}$ on a Riemannian manifold (\mathcal{M}, g) in terms of the metric as

$$\mathcal{L}[(y(t))_{t \in [0,1]}] = \int_0^1 \sqrt{g_{y(t)}(\dot{y}(t), \dot{y}(t))} dt. \quad (3.1)$$

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

$$\mathcal{E}[(y(t))_{t \in [0,1]}] = \int_0^1 g_{y(t)}(\dot{y}(t), \dot{y}(t)) dt, \quad (3.2)$$

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

$$\mathcal{L}[(y(t))_{t \in [0,1]}] \leq \sqrt{\mathcal{E}[(y(t))_{t \in [0,1]}]} \quad (3.3)$$

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

Definition 3.4 (Geodesic path). For $y_A, y_B \in \mathcal{M}$ a minimizer of the path energy among all paths $y: [0, 1] \rightarrow \mathcal{M}$ with $y(0) = y_A$ and $y(1) = y_B$ is denoted as *geodesic path* connecting y_A and y_B .

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 which leads to the usual geodesic equations. Rumpf and Wirth [RW15] showed 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 3.5 (Exponential map). Let $y(t) = y(t, p, V): I \rightarrow \mathcal{M}$, $0 \in I$, be the solution of $\frac{D}{dt}\dot{y}(t) = 0$ for initial data $y(0) = p$ and $\dot{y}(0) = V$. The Riemannian exponential map $\exp_p: T_p\mathcal{M} \rightarrow \mathcal{M}$ 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 $\eta > 0$, such that $\exp_p: B_\eta(0) \rightarrow \exp_p(B_\eta(0))$ is a bijection, which follows from the local uniqueness of geodesics. The image $U_p := \exp_p(B_\eta(0))$ is called a *normal neighborhood* of p

Definition 3.6 (Logarithm). The inverse operator of the exponential map is called the Riemannian logarithm $\log_p: U_p \rightarrow T_p\mathcal{M}$, 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].

3.2 Time-Discrete Geodesic Calculus

Now that we have introduced notions geodesic calculus on Riemannian manifold, we want to be able to compute them in practice. Instead of using numerical integration schemes for systems of ordinary differential equations resulting the geodesic equations, we will present the elements of a variational time discretization of geodesic calculus introduced by Rumpf and Wirth in a sequence of papers [WBR09; RW13; RW15]. The resulting time-discrete geodesic calculus is tailored-made for the application to shape spaces and thus yields effective numerical methods as demonstrated, for example, in [HRS+14; BER15; MRSS15]. In this section, we will only present the central definitions and some basic intuition of the different time-discrete notions. For more details and especially for detailed results on convergence, we refer to [RW15].

Metric. The continuous geodesic calculus presented in Section 3.1 was developed starting from a Riemannian metric, allowing to locally measure lengths and angles. In contrast, the discrete geodesic calculus below is 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_g , we can recover the metric at some point $p \in \mathcal{M}$ by

$$g_p(V, W) = \frac{1}{2} \partial_{22}^2 \text{dist}_g^2(p, p)(V, W), \quad V, W \in T_p\mathcal{M}. \quad (3.4)$$

Coming up with a notion of distance—i.e. quantifying how different two objects are—is often much easier than defining an inner product on tangent vectors.

For example, in the context of physical shape spaces distances can be constructed by dissimilarity measures as the discrete elastic energy from Section 2.3.1. However, these do not necessarily fulfill the axioms of a distance. 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 $\mathcal{W}: \mathcal{M} \times \mathcal{M} \rightarrow \mathbb{R}$ such that for $y, \tilde{y} \in \mathcal{M}$

$$\mathcal{W}[y, \tilde{y}] = \text{dist}_g^2(y, \tilde{y}) + O(\text{dist}_g^3(y, \tilde{y})). \quad (3.5)$$

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 (3.5) implies consistency with the metric as in (3.4) and is even necessary given a certain smoothness. This means for g smooth enough, a valid approximation is, for instance, given by $\mathcal{W}[y, \tilde{y}] = \frac{1}{2} g_y(\tilde{y} - y, \tilde{y} - y)$, which is of course not a practical choice since it requires access to the metric.

Theorem 3.7 (Consistency conditions, [RW15]). *If \mathcal{W} is twice Gâteaux-differentiable on $\mathcal{M} \times \mathcal{M}$ with bounded second Gâteaux derivative, then $\mathcal{W}[y, \tilde{y}] = \text{dist}_g^2(y, \tilde{y}) + O(\text{dist}_g^3(y, \tilde{y}))$ for \tilde{y} close to $y \in \mathcal{M}$ implies*

$$\mathcal{W}[y, y] = 0, \quad \partial_2 \mathcal{W}[y, y](V) = 0, \quad \partial_{22}^2 \mathcal{W}[y, y](V, W) = 2g_y(V, W)$$

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

$$\partial_{11}^2 \mathcal{W}[y, y](V, W) = -\partial_{12}^2 \mathcal{W}[y, y](V, W) = -\partial_{21}^2 \mathcal{W}[y, y](V, W) = \partial_{22}^2 \mathcal{W}[y, y](V, W).$$

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

Time-discrete geodesics. 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 $(y_0, \dots, y_K) \subset \mathcal{M}$ as a time-discrete K -path. Usually, one thinks of such a discrete path as an uniform sampling of a smooth curve $\gamma: [0, 1] \rightarrow \mathcal{M}$ by $y_k = \gamma(k\tau)$ for $k = 0, \dots, K$ where $\tau = K^{-1}$ and $K \in \mathbb{N}$ is the number of samples.

To relate the continuous path energy (3.2) to the Riemannian distance, one considers the following two estimates

$$\mathcal{L}[(y(t))_{t \in [0,1]}] \geq \sum_{k=1}^K \text{dist}_g(y_{k-1}, y_k), \quad \mathcal{E}[(y(t))_{t \in [0,1]}] \geq \frac{1}{\tau} \sum_{k=1}^K \text{dist}_g^2(y_{k-1}, y_k), \quad (3.6)$$

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 together with the statement of equality suggests that the sum on the right-hand side might be a reasonable approximation of \mathcal{E} motivating the following

Definition 3.8 (Discrete length and energy, [RW15]). For a discrete K -path (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

$$L^K[y_0, \dots, y_K] = \sum_{k=1}^K \sqrt{\mathcal{W}[y_{k-1}, y_k]}, \quad E^K[y_0, \dots, y_K] = K \sum_{k=1}^K \mathcal{W}[y_{k-1}, y_k]. \quad (3.7)$$

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

Rumpf and Wirth [RW15] showed 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.

Logarithm and Exponential. Based on Definition 3.8, we will now go on to introduce discrete counterparts to the notions of logarithm and exponential map. Let $p, q \in \mathcal{M}$ such that there is a unique geodesic $y: [0, 1] \rightarrow \mathcal{M}$ connecting them. Then, by Definition 3.6, we have $\log_p(q) = \dot{y}(0) \in T_p\mathcal{M}$. Using first-order difference quotients in time we get the approximation $\log_p(q) = \tau^{-1}(y(\tau) - y(0)) + \mathcal{O}(\tau)$, which motivates the following

Definition 3.9 (Discrete logarithm). Let $p, q \in \mathcal{M}$ and suppose the discrete geodesic (y_0, \dots, y_K) is the unique minimizer of the discrete path energy (3.7) with $y_0 = p$ and $y_K = q$. Then we define the *discrete logarithm* $\text{Log}_p^K(q) = K(y_1 - y_0)$.

Next, we introduce a discrete exponential map $\text{Exp}_{y_0}^k$ in a way such that it is compatible with the discrete logarithm, i.e. such that for a discrete geodesic (y_0, \dots, y_K) , we have that $\text{Exp}_{y_0}^k(\frac{V}{K}) = y_k$ where $V = \text{Log}_{y_0}^K(y_K) = K(y_1 - y_0)$. For this, we start with the definition of $\text{Exp}_{y_0}^k$ as third point $y_2 \in \mathcal{M}$, such that (y_0, y_1, y_2) is a time-discrete geodesic for $K = 2$ and continue by recursion.

Definition 3.10 (Discrete exponential map). For given points $y_0, y_1 \in \mathcal{M}$, $V = K(y_1 - y_0)$, we define $\text{Exp}_{y_0}^K(V) := y_K$, where for $k \in \{1, \dots, K-1\}$ the y_{k+1} are defined recursively as solutions of the nonlinear problem

$$y_{k+1} \in \mathcal{M} \quad \text{such that} \quad \partial_2 \mathcal{W}[y_{k-1}, y_k] + \partial_1 \mathcal{W}[y_k, y_{k+1}] = 0.$$

Rumpf and Wirth [RW15] showed the convergence of the discrete logarithm and exponential map to their continuous counterparts based on the convergence of discrete geodesics.

3.3 The Space of Discrete Shells

Our objective is to study deformations of discrete shells and derive useful applications from this, for instance, in shape editing. As mentioned in the introduction, we do this by considering a shape space, i.e. a space containing shapes as points. In our case, the shapes will be immersions of a discrete surface and we will equip the space with a Riemannian metric based on the physically sound deformation energies from Section 2.3.1. This structure was introduced by Heeren et al. in [HRWW12; HRS+14; Hee16]. As observed by Kilian, Mitra, and Pottmann [KMP07], this space allows us to phrase different problems relevant in computer graphics via the geometric notions we have seen in Section 3.1. Applying the time-discretization from Section 3.2, we can compute numerical approximation of these notions.

As in Section 2.3.1, we will only consider discrete surfaces in dense correspondence (cf. Definition 2.27). Hence, we are given the connectivity \mathcal{S}_h of a triangle mesh and consider different immersions of it into three-dimensional space.

Definition 3.11 (Space of discrete surfaces, [Hee16]). Given the connectivity \mathcal{S}_h of a triangle mesh, the *shape space of discrete surfaces* $\mathcal{M}[\mathcal{S}_h]$ 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 3.12 (Space of discrete shells, [Hee16]). Given the connectivity \mathcal{S}_h of a triangle mesh, the *Riemannian 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 2.32.

That the discrete deformation energy, in fact, induces a Riemannian metric on the shape space was proven by Heeren *et al.* [HRS+14]. 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 its nodal positions, we can also identify \mathcal{M} with an open subset of $\mathbb{R}^{3|V|} / SE(3)$. However, we will also introduce a representation of \mathcal{M} as an implicit submanifold in the next chapter.

Chapter 4

Nonlinear Rotation-Invariant Coordinates

In the previous chapters, we discussed elastic energies on discrete surfaces and how to use them to introduce the structure of a Riemannian manifold on the space of all immersions of a given discrete surface. Actually computing Riemannian operations boils down to solving geometric variational problems originating from the time-discretization of geodesic calculus. These variational problems are often quite challenging to solve numerically as they are inherently nonlinear. Furthermore, the elastic energy is rigid body motion invariant and thus these need to be controlled via an appropriate constraint to obtain a well-posed problem, which can be difficult in practice. Moreover, we already noted that the space of immersions can be represented as an open subset of the quotient space $\mathbb{R}^{3|V|}/SE(3)$. In this representation, also the tangent space at an immersion consists of equivalence classes of infinitesimal deformations under the action of the Lie algebra associated with $SE(3)$. This makes actually using the tangent space for computations very cumbersome. Thus, our aim was to find an alternative representation of a discrete surface's geometry which is inherently rigid body motion invariant and well-suited to be used as degrees of freedom for numerical optimization problems.

To this end, we propose to use a representation that we call the *Nonlinear Rotation-Invariant Coordinates (NRIC)*. They describe the geometry of a discrete surface by the vector stacking all edge lengths and dihedral angles instead of nodal positions. These coordinates are naturally rigid body motion invariant and can be seen as a discrete equivalent of the first and second fundamental form. Then—by virtue of integrability conditions akin to the Gauß–Codazzi equations introduced by Wang, Liu, and Tong [WLT12]—we will see that the space of immersions can in fact be represented by a implicit submanifold of $\mathbb{R}^{2|E|}$ which immediately greatly facilitates working in its tangent space. Beyond their inherent invariance to rigid transformations, these coordinates offer additional benefits, such as their natural occurrence in discrete deformation energies and their representation of natural modes of deformation in a localized sparse fashion. The latter property will be of central importance in the next chapter.

Prior work on shape interpolation by Winkler *et al.* [WDAH10] and Fröhlich and Botsch [FB11] showed that linear blending of the NRIC for a set of shapes already yields interesting nonlinear deformations. However, since, in general, nodal positions that realize given edge lengths and dihedral angles may not exist, these methods rely on optimization in the space of nodal positions. We will instead consider edge lengths and dihedral angles as primary degrees of freedom for geometric variational problems and introduce a general approach to solve such problems numerically. Through practical examples, we will see that this approach is also useful for variational problems beyond the time-discrete geodesic calculus. In particular, problems involving near-isometric deformations are typically ill-conditioned when phrased in nodal positions due to the combination of high stretching

and low bending resistance. In contrast, NRIC will offer a natural way to phrase and efficiently solve these problems.

The remainder of this chapter is split into four sections. We will begin in Section 4.1 by introducing the necessary background on edge lengths and dihedral angles and a discrete equivalent of the fundamental theorem of surfaces. In Section 4.2, we will use this to represent the space of immersions as an implicit submanifold. We will introduce our general setup for phrasing and numerically solving variational problems in NRIC in Section 4.3. Finally, in Section 4.4, we will demonstrate results obtained by minimization in NRIC and sketch how this might be useful in applications.

Remark. This chapter is a continuation and extension of the work from the author’s Master’s thesis [Sas19]. As such, it is the result of joint work with Behrend Heeren, Klaus Hildebrandt, and Martin Rumpf published in [SHHR20]. The adaption of the discrete Riemannian logarithm and exponential maps was published as part of [SHR20], which will be discussed in more detail in the next chapter.

4.1 Discrete Gauß–Codazzi Equations

We begin the chapter with recalling the characterization of admissible edge lengths and dihedral angles—i.e. such that an immersion admitting them exists—by Wang, Liu, and Tong [WLT12]. To this end, we first describe edge lengths and dihedral angles as discrete equivalents of first and second fundamental form. Then we will present a line of argumentation mirroring the one in Section 2.1.1 and leading to conditions reminiscent of the Gauß–Codazzi equations. This section is also a condensed version of an equivalent chapter in [Sas19]. Here, we will restrict ourselves to simply-connected discrete surfaces and details on the general case can be found in the aforementioned references.

Discrete Fundamental Forms. To describe edge lengths as an equivalent to the discrete first fundamental form we use that the geometry of a Euclidean triangle is completely determined by its edge lengths. To this end, let us recall the discrete first fundamental form from Definition 2.20, which was elementwise constant and given by

$$G|_{\tau} = \begin{pmatrix} \|E_2(\tau)\|^2 & -\langle E_1(\tau), E_2(\tau) \rangle \\ -\langle E_1(\tau), E_2(\tau) \rangle & \|E_1(\tau)\|^2 \end{pmatrix} \quad (4.1)$$

for each $\tau \in \mathbf{T}$. We directly see that the entries of the diagonal are simply squared edge lengths of the triangle $T(\tau)$. 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 law of cosines. We, therefore, see we can describe our discrete first fundamental form solely in terms of edge lengths. In Section 2.3, we also needed the determinant of $G|_{\tau}$, which is given by the triangle’s squared area. Let us note, that this can also be computed directly from edge lengths by Heron’s formula—another classic result from trigonometry—which will come in handy later on.

Next, we recall the triangle-averaged discrete second fundamental form, which is elementwise constant and given in Definition 2.23 by

$$H|_{\tau} = -4a_{\tau} \sum_{i=0}^2 \frac{\cos \frac{\theta_i + \pi}{2}}{\|E_i\|} M_i \quad (4.2)$$

for each $\tau \in \mathbf{T}$, where (M_0, M_1, M_2) is the basis of symmetric 2×2 -matrices from (2.17). Again, we immediately see that we need the edge lengths to define the matrix, as well as the triangle’s area. Additionally, we need the dihedral angles (i.e. the angle between adjacent face normals, cf. Definition 2.22) 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. For an immersion $X \in \mathbb{R}^{3|V|}$ of a discrete surface \mathcal{S}_h , we denote by $(l_e(X))_{e \in E}$ its edge lengths and by $(\theta_e(X))_{e \in E}$ its dihedral angles. We combine them in a single vector and thus obtain a map

$$\begin{aligned} Z: \mathbb{R}^{3|V|} &\rightarrow \mathbb{R}^{|E|} \times \mathbb{R}^{|E|} \\ X &\mapsto ((l_e(X))_{e \in E}, (\theta_e(X))_{e \in E}). \end{aligned} \tag{4.3}$$

In the following, we will denote by $z = (l_e, \theta_e)_{e \in E}$ a set of lengths and angles in $\mathbb{R}^{2|E|}$, 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. As we will consider $z \in \mathbb{R}^{2|E|}$ as coordinates describing the geometry of a surface in optimization problems, we also call them the *Nonlinear Rotation-Invariant Coordinates (NRIC)*.

The map Z is well-defined only on vertex positions in $\mathbb{R}^{3|V|}$ which do not lead to degenerated triangles, i.e. belong to an immersion as in Definition 2.19. Otherwise, the dihedral angles would not be well-defined. Then, restricted to this open domain, Z is a smooth map.

4.1.1 Discrete Integrability Conditions

In Section 2.1.1, we have seen that not all bilinear forms are fundamental forms of an immersed surfaces, but rather they have to fulfill the Gauß–Codazzi equations. The same holds for elements in $\mathbb{R}^{2|E|}$ that are actually given as edge lengths and dihedral angles of an immersion of our discrete surface as proven in [WLT12] and summarized below.

Triangle Inequality. The first condition is that we have to ensure the lengths actually describe valid Euclidean triangles. This is guaranteed by the triangle inequality, formalized in the following definition.

Definition 4.1 (Triangle inequality constraints). For a face $\tau \in \mathbf{T}$ with edges e_i, e_j , and e_k , we define the *triangle inequality map* on lengths and angles z to be

$$\mathcal{T}_\tau(z) = (l_i + l_j - l_k, \quad l_j + l_k - l_i, \quad l_k + l_i - l_j) \in \mathbb{R}^3. \tag{4.4}$$

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

$$\mathcal{T}_\tau(z) > 0 \text{ for all } \tau \in \mathbf{T}, \tag{T}$$

where the inequality is to be understood componentwise.

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 $\tau \in \mathbf{T}$, where again by abuse of notation we identify it with its immersion, we denote by $N_\tau \in S^2$ its normal and by $T_\tau \mathbf{S}$ its tangent plane, i.e. the plane spanned by its edge vectors.

Definition 4.2 (Discrete frame). Given a discrete surface \mathcal{S}_h with immersion $X \in \mathbb{R}^{3|V|}$, we define a discrete frame on a face τ to be an orthonormal basis $F_\tau = (b_1, b_2, b_3)$ with $b_3 = N_\tau$, hence (b_1, b_2) forms an orthonormal basis of the tangent plane $T_\tau \mathbf{S}$.

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 $T_\tau \mathbf{S}$ differing by a rotation around the normal. 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 to the following notion of a *standard discrete frame*.

Definition 4.3 (Standard discrete frame). Considering a face τ 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 $F_\tau = \left(\frac{E_1}{\|E_1\|}, \frac{E_1 \times N_\tau}{\|E_1 \times N_\tau\|}, N_\tau \right)$.

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 τ_1 and τ_2 with vertices v_1, v_2, v_3 and v_2, v_3, v_4 respectively, i.e. $e = (v_2 v_3)$ 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. Let F_1 and F_2 be frames on τ_1 and τ_2 respectively. 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 $\text{tr} F_1 \tilde{F}_1^{-1} = \text{tr} F_1^{-1} \tilde{F}_1$, 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 4.4 (Transition rotation). Given an immersed discrete surface \mathbf{S} with frames F_τ for all faces $\tau \in \mathbf{T}$, we define for an edge e with adjacent faces τ_i and τ_j the *transition rotation* by

$$R_{ij} = R_z(\gamma_{e, F_i}) R_x(-\theta_e) R_z(\gamma_{F_j, e}), \quad (4.5)$$

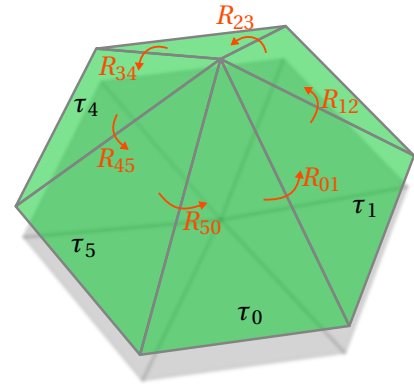
where γ_{e, F_j} and $\gamma_{F_i, e}$ denote the angles between the immersed edge E and the first vector of the frames F_{τ_i} resp. F_{τ_j} .

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} can be completely determined from the length and angles of the immersed surface. This allows us to define transition rotations for NRIC without knowing the immersion, as long as they fulfill the triangle inequality, formalized as follows. We define for an edge e with adjacent faces τ_i and τ_j the *induced transition rotation* by

$$R_{ij}(z) = R_z(\gamma_{e, i}) R_x(-\theta_e) R_z(\gamma_{j, e}), \quad (4.6)$$

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

The 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. Thereby, we consider an interior vertex v which is the center of a n -loop of faces $\tau_0, \dots, \tau_{n-1}$ 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 $F_j = F_i R_{ij}$. 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, i.e. $F_0 = F_0 \prod_{i=0}^{n-1} R_{i,(i+1) \bmod n}$, and hence $\prod_{i=0}^{n-1} R_{i,(i+1) \bmod n} = \text{Id}$. Now, assume that we have an element $z \in \mathbb{R}^{2|\mathbb{E}|}$ 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 4.5 (Discrete integrability conditions). Given a discrete surface S_h , for each interior vertex v , we define the *discrete integrability map* on NRIC $z \in \mathbb{R}^{2|\mathbb{E}|}$ with $\mathcal{T}(z) > 0$ as

$$\mathcal{I}_v(z) = \prod_{i=0}^{n-1} R_{i,(i+1) \bmod n}(z) \quad (4.7)$$

for the n -loop of faces $\mathbf{N}_v = \{\tau_0, \dots, \tau_{n-1}\}$ around v . We say that lengths and angles z satisfy the *discrete integrability conditions* if

$$\mathcal{I}_v(z) = \text{Id} \text{ for each } v \in \mathbf{V}_0. \quad (\text{I})$$

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 τ_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. 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 to check (I) more efficiently by reducing the number of needed rotations. For this approach, let us consider the vertex $v \in \mathbf{V}$ with the n -loop of faces $\tau_0, \dots, \tau_{n-1}$ 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. Based on this, we choose the frames F_0, \dots, F_{n-1} such that the chosen edge is always the first basis vector. The corresponding transition rotations simplify to $R_{ij}(z) = R_x(\theta_e) R_z(\gamma_{j,v})$, where $\gamma_{j,v}$ is the interior angle at v in τ_j . By this, the product of transition rotations $\mathcal{I}_v(z)$ becomes an alternating product of rotations about the interior and dihedral angles.

Fundamental Theorem. Finally, we one can show that these conditions are indeed necessary and sufficient for z to admissible for a simply-connected discrete surface.

Theorem 4.6 (Discrete fundamental theorem for simply-connected surfaces, [WLT12]). *Given a simply-connected discrete surface S_h , if $z \in \mathbb{R}^{2|\mathbb{E}|}$ satisfies the discrete integrability conditions and the triangle inequalities, then there exists an immersion $X \in \mathbb{R}^{3|\mathbf{V}|}$ of the surface in three-dimensional Euclidean space admitting z as edge lengths and dihedral angles.*

The proof amounts to showing that constructing the immersions of all triangles from a given triangle is independent of the chosen path. To this end, one considers the loop resulting from two different reconstruction paths and applies the integrability condition (I) recursively to show that this loop leads to the identity. The proof was originally introduced by Wang, Liu, and Tong [WLT12] and a more explicit version matching the notation here can be found in [Sas19].

However, in the formulation above, $\mathcal{I}(z)$ consists of matrices from $SO(3)$ which is impractical for implementations. Thus we aim at reformulating this using a suitable parametrization of $SO(3)$. In [Sas19], we have used Euler angles for this. In the following, we will take a different route based on quaternions.

Quaternions. To this end, let us first briefly recall the necessary basics of quaternions and their relation to spatial rotation. For a detailed treatment, we refer to standard textbooks such as [Han07]. Quaternions can be understood as an extension of the complex numbers and are generally represented as $q = a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k}$, where $a, b, c, d \in \mathbb{R}$ and $\mathbf{i}, \mathbf{j}, \mathbf{k}$ are the so-called *quaternion units*. These units fulfill the fundamental identity $\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{i}\mathbf{j}\mathbf{k} = -1$, from which the general multiplication of quaternions can be defined via distributive and associative law and thus quaternions form a noncommutative division ring \mathbb{H} . In this context, a is called the real part of q and b, c , and d the vector part, for which we also write $\text{vec}(q) = (b, c, d) \in \mathbb{R}^3$. Unit quaternions are those for which the product with their conjugate $\bar{q} := a - b\mathbf{i} - c\mathbf{j} - d\mathbf{k}$ is one, i.e. $q\bar{q} = a^2 + b^2 + c^2 + d^2 = 1$. Points in three-dimensional space $p \in \mathbb{R}^3$ can be identified with quaternions having vanishing real part, i.e. we write $p = p_1\mathbf{i} + p_2\mathbf{j} + p_3\mathbf{k}$. Now, given a rotation Q around the unit vector $u \in \mathbb{R}^3$ by angle $\varphi \in [0, 2\pi)$ we can define a corresponding *unit* quaternion

$$q(u, \varphi) := \cos \frac{\varphi}{2} + (u_1\mathbf{i} + u_2\mathbf{j} + u_3\mathbf{k}) \sin \frac{\varphi}{2}.$$

Then one can verify that for any $p \in \mathbb{R}^3$, the conjugation qpq^{-1} with q results in the rotated point Qp . The quaternion $-q(u, \varphi)$ would lead to the same rotation, thus the quaternions form a double covering of $SO(3)$. Furthermore, investigating this conjugation one realizes that the composition of two rotations given as unit quaternions $q_1, q_2 \in \mathbb{H}$ is given by their product q_1q_2 and hence this correspondence is a homomorphism between $SO(3)$ and the unit quaternions.

Reformulation. Turning to the reformulation of the integrability conditions (I), recall that we needed rotations around the 0th and 2nd basis vector in \mathbb{R}^3 for which we now introduce the corresponding quaternions

$$q_0(\varphi) := \cos \frac{\varphi}{2} + \mathbf{i} \sin \frac{\varphi}{2}, \quad q_2(\varphi) := \cos \frac{\varphi}{2} + \mathbf{k} \sin \frac{\varphi}{2} \quad \text{for } \varphi \in [0, 2\pi). \quad (4.8)$$

Then we identify the simplified transition rotation $R_{ij} = R_0(\theta_i)R_2(\gamma_j)$ from before with the quaternion

$$q_{ij} := q_0(\theta_i)q_2(\gamma_j), \quad (4.9)$$

where again $\tau_0, \dots, \tau_{n_v-1}$ and e_0, \dots, e_{n_v-1} are the n_v -loops of faces resp. edges connected to v , γ_j is the interior angle at v in τ_j with $j = i + 1$ modulo n_v , and θ_i is the dihedral angle at e_i . To finally reformulate the condition (I), we need to deal with the ambiguity introduced by the double covering, i.e. that the identity rotation is represented by $q = \pm 1$. However, we see that in both cases the vector part $\text{vec}(q) \in \mathbb{R}^3$ is zero, which is indeed for unit quaternions already a sufficient condition to be plus or minus one. Then we use this alternative characterization of the identity rotation to formulate the *quaternion integrability conditions* as

$$\mathcal{Q}_v(z) := \text{vec} \left(\prod_{i=0}^{n_v-1} q_{i, (i+1) \bmod n_v}(z) \right) \stackrel{!}{=} 0 \quad (\text{I}_q) \quad (4.10)$$

for the n_v -loop of faces around all interior vertices $v \in \mathbf{V}_0$.

Reconstruction of an Immersion. The constructive proof of Theorem 4.6 also provides us with an algorithm to reconstruct immersions $X \in \mathbb{R}^{3|V|}$ from NRIC $z \in \mathbb{R}^{2|E|}$ fulfilling the integrability conditions. In this context, the position and orientation of the immersion are specified via fixing one vertex and the orientation of one triangle. For a $z \in \mathbb{R}^{2|E|}$ that does not satisfy the integrability conditions, we can compute the immersion of the closest admissible point. This means we search for the solution of the nonlinear least-squares problem $\min_{X \in \mathbb{R}^{3|V|}} \|Z(X) - z\|^2$ with some appropriate norm on $\mathbb{R}^{2|E|}$. Fröhlich and Botsch [FB11] first used this approach with a weighted Euclidean norm, which we will discuss in Section 4.3.1. A detailed explanation and comparison of these reconstruction algorithms can be found in [Sas19, Chapter 3] or [SHHR20, Section 7].

4.2 Manifold Structure and Geodesic Calculus

At the end of Section 3.3, we remarked that the space of all immersions of a discrete surface \mathcal{S}_h modulo rigid body motions $\mathcal{M}[\mathcal{S}_h]$ can be represented as an open subset of $\mathbb{R}^{3|V|}/SE(3)$. This has also been the representation primarily used in numerical computations, i.e. to work with nodal positions and fix the rigid body motions via some constraint. However, handling rigid body motions can be cumbersome in practice, especially when considering infinitesimal variations, i.e. tangent vectors. Hence, we will introduce an alternative representation of $\mathcal{M}[\mathcal{S}_h]$ as implicit submanifold of $\mathbb{R}^{2|E|}$ based on the integrability conditions introduced in the previous section. In the following, we will refer to the aforementioned representation using nodal positions as $\mathcal{N}[\mathcal{S}_h]$ to better discriminate it from our new representation. Furthermore, we will drop the explicit dependence on \mathcal{S}_h in our formulas since we will work with a fixed discrete surface throughout the section.

The representation we choose is—not surprisingly—the set of all admissible lengths and angles, i.e. we define

$$\mathcal{M} := Z(\mathcal{N}) = \{z \in \mathbb{R}^{2|E|} \mid \exists X \in \mathcal{N} : Z(X) = z\}. \quad (4.10)$$

From Theorem 4.6, we know that we can it alternatively characterize by

$$\mathcal{M} = \{z \in \mathbb{R}^{2|E|} \mid \mathcal{T}(z) > 0, \mathcal{Q}(z) = 0\}, \quad (4.11)$$

which indeed describes \mathcal{M} as an implicit submanifold of $\mathbb{R}^{2|E|}$. In the following, we will describe the remaining geometric structure on \mathcal{M} and discuss some adaptations to the time-discrete geodesics calculus we have seen so far.

Tangent space. The implicit formulation (4.11) consists of the triangle inequalities defining an open convex polytope and of the nonlinear integrability conditions defining a lower-dimensional, differential structure on \mathcal{M} . Therefore, we can derive an implicit description of its tangent space solely based on \mathcal{Q} . In detail, for $z \in \mathcal{M}$ the tangent space is given by

$$T_z \mathcal{M} = \ker D\mathcal{Q}(z) := \{w \in \mathbb{R}^{2|E|} \mid D\mathcal{Q}(z)w = 0\}, \quad (4.12)$$

where $D\mathcal{Q}(z)$ is a matrix in $\mathbb{R}^{3|V_0| \times 2|E|}$.

The partial derivatives of \mathcal{Q}_v are given by the chain rule as

$$\partial_{z_k} \mathcal{Q}_v(z) = \text{vec} \left(\sum_{i=0}^{n_v-1} q_{01}(z) \dots \partial_{z_k} q_{i,i+1}(z) \dots q_{n_v-1,0}(z) \right), \quad (4.13)$$

where the partial derivatives of a quaternion-valued map are to be understood componentwise as for vector-valued maps. The gradient of \mathcal{Q}_v can be computed with $O(n_v)$ cost and is sparse. It has only $O(n_v)$ non vanishing entries, i.e. $\partial_{\theta_e} \mathcal{Q}_v \equiv 0$ if v is not a vertex of the edge e and $\partial_{l_e} \mathcal{Q}_v \equiv 0$ if the edge e is not an edge of a triangle with vertex v .

The direct access to this tangent space is useful in various computational applications. For example, in Chapter 5, on the one hand we will see that it allows the straightforward computation of a PGA on the shape space of discrete shells, while the other hand it will also be a crucial building block to enable the computation of appropriate sparse deformation modes. Furthermore, access to the tangent space allows us to numerically investigate the infinitesimal rigidity of surfaces, which we will explain in more detail in Section 4.4. To incorporate the tangent space into our algorithms, we can, for example, simply use the equation $DQ(z)w = 0$ as a constraint in an optimization setup, which we will see later in Chapter 5. Alternatively, we can use the orthogonal projection onto it, which is given by the linear operator $\text{Id} - DQ^T (DQDQ^T)^{-1} DQ$ where the involved linear system can be solved with direct approaches such as the Cholesky decomposition. If we want to study the tangent space itself, we can compute a basis for it by computing the Singular Value Decomposition (SVD) of $DQ(z)$ and using the right-singular vectors corresponding to the singular value zero or by computing a rank-revealing QR decomposition of it.

Riemannian Metric. Given an appropriate elastic energy for NRIC we can define a metric on tangent vectors of the NRIC manifold following Rayleigh's paradigm as explained in Section 3.2. Concretely, we define the metric as half of the Hessian of the elastic energy, i.e. $g_z: \mathbb{R}^{2|\mathbb{E}|} \times \mathbb{R}^{2|\mathbb{E}|} \rightarrow \mathbb{R}$ with $g_z = \frac{1}{2} \text{Hess} \mathcal{W}[z, \cdot]$ restricted to $T_z \mathcal{M} \times T_z \mathcal{M}$. To this end, we will discuss in Section 4.3 the adaption of the discrete elastic energy introduced in Section 2.3.1 and a quadratic approximation introduced in [FB11].

Discrete Geodesic Calculus. Having access to the tangent space and tangent vectors has also implications for the time-discrete geodesic calculus introduced in Section 3.2. There we assumed that we do not work with tangent vectors, instead we used the difference of two points on the manifold as approximation and based the whole calculus on this approximation. Here, we present a simple adaption of this calculus, concretely for the time-discrete exponential and logarithm, via the projection onto the manifold and its tangent space, respectively.

To this end, we define the discrete logarithm at some discrete shell z_0 as

$$\text{Log}_{z_0}^K(z_K) = v_0$$

where $v_0 = K \mathbf{P}_{T_{z_0} \mathcal{M}}(z_1 - z_0)$ is a the initial velocity of the discrete geodesic (z_0, z_1, \dots, z_K) based on time step size $\tau = \frac{1}{K}$. Here, $\mathbf{P}_{T_{z_0} \mathcal{M}}$ is the orthogonal projection onto the tangent space $T_{z_0} \mathcal{M}$ with respect to the metric g_{z_0} , which is extended to $\mathbb{R}^{2|\mathbb{E}|}$ per its definition. The corresponding exponential map $\text{Exp}_{z_0}^K(v_0)$ for $v_0 \in T_{z_0} \mathcal{M}$, as the inverse mapping, computes a discrete geodesic (z_0, z_1, \dots, z_K) for given discrete, initial velocity v_0 . In fact, we first compute z_1 for given z_0 and v_0 via orthogonal projection of $z_0 + \frac{v_0}{K}$ onto \mathcal{M} with respect to g_{z_0} , i.e. $z_1 := \mathbf{P}_{\mathcal{M}}[z_0 + \frac{v_0}{K}]$. The projection $\mathbf{P}_{\mathcal{M}}$ can be computed using a Gauß–Newton scheme [FB11]. Then we recursively compute z_{k+1} solving $\partial_2 \mathcal{W}[z_{k-1}, z_k] + \partial_1 \mathcal{W}[z_k, z_{k+1}] = 0$ in a least squares sense, i.e.

$$z_{k+1} = \arg \min_{z \in \mathcal{M}} \|\partial_2 \mathcal{W}[z_{k-1}, z_k] + \partial_1 \mathcal{W}[z_k, z]\|_2^2.$$

For sufficiently large K , a solution to $\partial_2 \mathcal{W}[z_{k-1}, z_k] + \partial_1 \mathcal{W}[z_k, z] = 0$ under the constraint $z \in \mathcal{M}$ exists and then agrees with z_{k+1} . After $K - 1$ steps this yields the requested discrete geodesic and the discrete exponential, i.e.

$$\text{Exp}_{z_0}^K(v_0) := z_K.$$

As a sanity check for this construction, we note the following convergence result, which easily follows from the results in [RW15] and the regularity of the projection onto the manifold for large enough K , cf. [LS21].

Proposition 4.7. *Let $\mathcal{M} \subset \mathbb{R}^d$ be an embedded Riemannian manifold. Then for a tangent vector $v \in T_z\mathcal{M}$, we have $\text{Exp}_z^K(v) \rightarrow \exp_z v$ for $K \rightarrow \infty$. Correspondingly, for a point $\tilde{z} \in \mathcal{M}$, we have $\text{Log}_z^K(\tilde{z}) \rightarrow \log_z \tilde{z}$.*

A numerical investigation of this convergence can be found in Section 4.4.

4.3 Geometric Optimization Problems

The quest for geometrically optimal, discrete surfaces often leads to variational problems. However, in many applications, the corresponding objective functional can naturally be formulated in our coordinates, thus on the NRIC manifold (4.11), and its first and second variation can be computed easily. To this end, one aims at solving a constrained optimization problem, i.e. given an objective functional $\mathcal{E} : \mathbb{R}^{2|\mathbb{E}|} \rightarrow \mathbb{R}$ the task is to

$$\begin{aligned} & \underset{z \in \mathbb{R}^{2|\mathbb{E}|}}{\text{minimize}} && \mathcal{E}(z) \\ & \text{subject to} && \mathcal{Q}_\nu(z) = 0 \text{ for each } \nu \in \mathbf{V}_0, \\ & && \mathcal{T}_\tau(z) > 0 \text{ for each } \tau \in \mathbf{T}. \end{aligned} \tag{OPT}$$

Due to non-convexity of the objective, in general, there is no guarantee for a unique, global minimizer for the optimization problem.

4.3.1 Elastic Energies

One central ingredient in many of the examples we will consider in the next section is an elastic deformation energy \mathcal{W} between different NRIC. To this end, we will explain how the membrane and bending energy discussed in Section 2.3.1 can be reformulated in NRIC. Both are then again added to obtain the elastic energy

$$\mathcal{W}[z, \tilde{z}] = \mathcal{W}_{\text{mem}}[z, \tilde{z}] + \mathcal{W}_{\text{bend}}[z, \tilde{z}], \tag{4.14}$$

where the components scale quadratically, respectively cubically in the thickness δ . This reformulation will be a straightforward undertaking which underlines our claim that NRIC are a natural choice for computing deformations. In particular, we will see that the local injectivity constraints inbuilt in this energy allow us to replace the triangle inequalities and thus reduce the number of constraints. For comparison reasons, we will finally consider a simple quadratic deformation energy as it has been used in [FB11].

Membrane energy. Consider two immersions $X, \tilde{X} \in \mathbb{R}^{3|\mathbf{V}|}$ with corresponding NRIC $z, \tilde{z} \in \mathbb{R}^{2|\mathbb{E}|}$. To reformulate the membrane energy, we have to express the trace and determinant of the discrete distortion tensor $\mathcal{G}[X, \tilde{X}]|_\tau = (G|_\tau)^{-1} \tilde{G}|_\tau$ for a face $\tau \in \mathbf{T}$ in edge lengths, interior angles, and triangle areas, which can all be derived from edge lengths. The determinant takes the form,

$$\det \mathcal{G}[X, \tilde{X}]|_\tau = (\det G|_\tau)^{-1} \det \tilde{G}|_\tau = a_\tau^{-2} \tilde{a}_\tau^2. \tag{4.15}$$

Concerning the trace, recall the inverse of the discrete fundamental form, given by Cramer's rule as

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

which implies

$$\text{tr } \mathcal{G}[X, \tilde{X}]|_\tau = \frac{1}{4a_\tau^2} (\|E_1\|^2 \|\tilde{E}_2\|^2 - 2\langle E_1, E_2 \rangle \langle \tilde{E}_1, \tilde{E}_2 \rangle + \|\tilde{E}_1\|^2 \|E_2\|^2).$$

As the computation of all terms only requires edge lengths, we also write $\mathcal{G}[z, \bar{z}]|_\tau$ for the distortion tensor. Hence, we define the membrane energy on NRIC $z, \bar{z} \in \mathbb{R}^{2|\mathbf{E}|}$ as

$$\mathcal{W}_{\text{mem}}[z, \bar{z}] = \delta \sum_{\tau \in \mathbf{T}} a_\tau \cdot W_{\text{mem}}(\mathcal{G}[z, \bar{z}]|_\tau), \quad (4.16)$$

with the same energy density

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

as before, where μ and λ are positive material constants and a_τ is the area of τ computed from edge lengths by Heron's formula.

Bending energy. Next, we adapt the bending energy introduced in Definition 2.31. One directly sees that expressing this energy in lengths and angles requires no further calculations, and as before we replace its primary variables by NRIC. That is we define the *Discrete Shells bending energy* on NRIC $z, \bar{z} \in \mathbb{R}^{2|\mathbf{E}|}$ as

$$\mathcal{W}_{\text{bend}}[z, \bar{z}] := \delta^3 \sum_{e \in \mathbf{E}} \frac{(\theta_e - \bar{\theta}_e)^2}{d_e} l_e^2, \quad (4.17)$$

where $d_e = \frac{1}{3}(a_\tau + a_{\tau'})$ for the two faces τ and τ' adjacent to $e \in \mathbf{E}$, as before computed by Heron's formula.

Relationship with triangle inequalities. One essential property of the membrane energy is that it allows us to control local injectivity via the built-in penalization of volume shrinkage, i.e. we have $W_{\text{mem}}(\mathcal{G}[z, \bar{z}]|_\tau) \rightarrow \infty$ for $\bar{a}_\tau \rightarrow 0$. To see this, we recognize that $\det \mathcal{G}[z, \bar{z}]|_\tau = (\det G|_\tau)^{-1} \det \tilde{G}|_\tau = a_\tau^{-2} \bar{a}_\tau^2$ and hence $-\log \det \mathcal{G}[z, \bar{z}]|_\tau \rightarrow \infty$ when \bar{a}_τ goes to zero. This control over the local injectivity also has consequences for the consideration of the triangle inequalities. Because of it, we also have that the energy diverges, i.e. $W_{\text{mem}}(\mathcal{G}[z, \bar{z}]|_\tau) \rightarrow \infty$ if one of the components of $\mathcal{T}_\tau(\bar{l})$ approaches zero meaning that we get close to violating one of the triangle inequalities. Especially, we set $W_{\text{mem}}(\mathcal{G}[z, \bar{z}]|_\tau) = \infty$ if $\mathcal{T}_\tau(\bar{l}) > 0$ does not hold. This allows us to characterize the NRIC manifold \mathcal{M} by

$$\mathcal{M} = \{z \in \mathbb{R}^{2|\mathbf{E}|} \mid \mathcal{W}[z^*, z] < \infty \text{ for a fixed } z^* \in \mathcal{M}, \mathcal{Q}(z) = 0\}, \quad (4.18)$$

avoiding the explicit dependence on the triangle inequalities (T) we had before. Note, however, that the integrability conditions (I_q) are still necessary as finite energy does not guarantee their attainment. The characterization (4.18) will be helpful later on to devise efficient numerical schemes for solving variational problems on \mathcal{M} .

Quadratic model. Previously, Fröhlich and Botsch [FB11] used a quadratic deformation model for NRIC, i.e. they considered the weighted quadratic energy

$$\mathcal{W}_q[z, z^*] = \delta \sum_{e \in \mathbf{E}} \alpha_e \|l_e - l_e^*\|^2 + \delta^3 \sum_{e \in \mathbf{E}} \beta_e \|\theta_e - \theta_e^*\|^2. \quad (4.19)$$

In fact, almost the same model has been used by Grinspun *et al.* [GHDS03] to define the *Discrete Shells* energy for physical simulations based on nodal positions. The weights $\alpha = (\alpha_e)_e$ and $\beta = (\beta_e)_e$ can be chosen in different ways. Typically, they are computed from edge lengths $l_e = l_e(\bar{z})$ and areas $d_e = d_e(\bar{z})$ associated with edges and defined on some representative reference configuration $\bar{z} \in \mathbb{R}^{2|\mathbf{E}|}$. For example, Grinspun *et al.* [GHDS03] and Fröhlich and Botsch [FB11] set in a related context $\alpha_e = l_e^{-2}$ and $\beta_e = l_e^2 d_e^{-1}$, whereas Heeren *et al.* [HRS+16] have chosen $\alpha_e = d_e l_e^{-2}$, for $e \in \mathbf{E}$. Here the (physical) parameter $\delta > 0$ trades the impact on length variations off against angle variations and can be considered as the thickness of the material as before. This quadratic energy has no

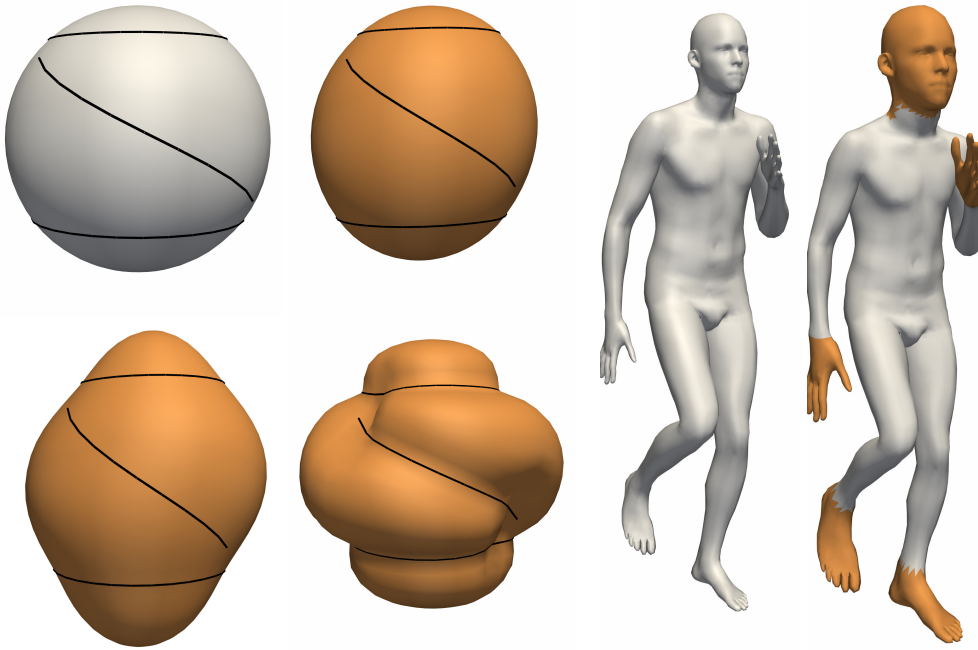


Figure 4.1: Left: Unit sphere (grey) with black constraint curves to be shortened by means of equality constraints on edge lengths along with different results for varying bending parameter $\delta = 10^{-\{0,1,2\}}$ in the nonlinear objective (4.14). Right: Same experiment but with (orange) constraint areas where the target edge lengths were increased by 30%.

inbuilt control over the local injectivity of the deformation and hence does not allow a characterization without explicit dependence on the triangle inequalities as in (4.18). We found that in many of our examples this decreased the numerical accessibility and increased the needed number of iterations and runtimes. Nevertheless, as demonstrated by Fröhlich and Botsch [FB11], it often leads to natural-looking deformations and we will consider it in some of our examples.

We also generalize this quadratic energy to introduce a family of weighted L^p -norms on $R^{2|\mathbf{E}|}$. Consider, as above, edge lengths $\bar{l}_e = l_e(\bar{z})$ and areas $\bar{d}_e = d_e(\bar{z})$ defined on some representative reference configuration $\bar{z} \in \mathbb{R}^{2|\mathbf{E}|}$. Then we define

$$\|u\|_p^p := \sum_{e \in \mathbf{E}} \left(\frac{1}{\bar{l}_e} |l_e| \right)^p + \sum_{e \in \mathbf{E}} \left(\frac{\bar{l}_e}{\sqrt{\bar{d}_e}} |\theta_e| \right)^p, \quad u = (l, \theta) \in \mathbb{R}^{2|\mathbf{E}|}. \quad (4.20)$$

For $p = 2$, this agrees with the quadratic energy from above with the weights used by Fröhlich and Botsch [FB11]. These norms will come in handy in Chapter 5, where we will use the L^1 -norm as sparsity-inducing term.

4.3.2 Algorithmic Approach

Below, we specify the details of our approach for the numerical minimization of (OPT) before turning to concrete examples in the next section.

Ensuring triangle inequalities. One crucial problem we encounter trying to solve (OPT) are the triangle inequalities which lead to $3|\mathbf{T}|$ inequality constraints causing the problem to be computationally expensive. Therefore, we aim for an approach to deal with them efficiently rooted in our geometric setup. We achieve this by a modified line search. First, recall that the set $\mathcal{M}_{\mathcal{T}} =$

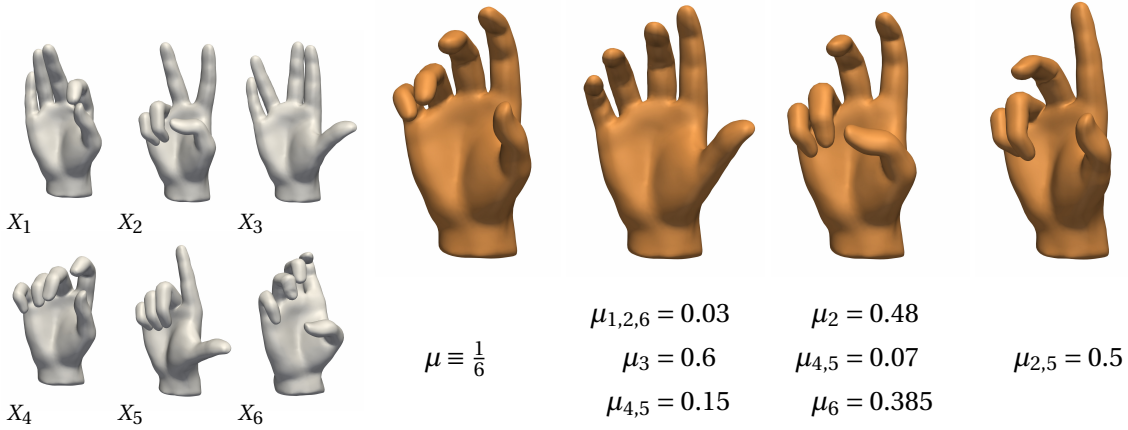


Figure 4.2: Reconstruction of nodal positions from elastic averages of six hand poses (grey) with different (convex) weights $\mu \in \mathbb{R}^6$ computed as minimizer of (4.25) on the NRIC manifold.

$\{z \in \mathbb{R}^{2|\mathbb{E}|} \mid \mathcal{T}(z) > 0\}$ defines an open connected subset of $\mathbb{R}^{2|\mathbb{E}|}$. Therefore, if we start with an initial point z^0 fulfilling the triangle inequalities we only have to ensure that every iterate remains in the set. Hence, in a line search method—where we search for a new iterate z^{k+1} along a direction d^k —we have to restrict this search to $\mathcal{M}_{\mathcal{T}}$. We accomplish this using backtracking, i.e. reducing the stepsize β^k until $z^{k+1} = z^k + \beta^k d^k \in \mathcal{M}_{\mathcal{T}}$ holds. In implementations, this can easily be achieved by setting $\mathcal{Q}_v(z) = \infty$ if $\mathcal{T}_\tau(z) \not> 0$ for any face τ adjacent to $v \in \mathbf{V}_0$.

We can obtain an even more natural approach when we work with the nonlinear membrane energy \mathcal{W}_{mem} . Recall that in Section 4.3.1 we introduced the characterization (4.18) of \mathcal{M} without explicit dependence on the triangle inequalities by exploiting the growth of W_{mem} for triangles with vanishing area. This now readily fits into our modified line search approach. In fact, if we compare our nonlinear energy to interior point methods [NW06, Chapter 19] we see that the logarithmic penalty in the energy takes the role of a barrier term which ensures that we stay in the admissible set $\mathcal{M}_{\mathcal{T}}$.

Overall, we see that in both cases we can treat the inequality constraints in the line search and hence apply algorithms for equality-constrained optimization with a considerably lower number of constraints. Note, that this approach can be adapted for trust-region methods by limiting the size of the trust-region appropriately.

Augmented Lagrange. Next, we describe our approach to solving these equality-constrained problems based on the augmented Lagrange method. First, let us briefly recall the Lagrangian formulation of our problem. In fact, this means we seek for a saddle point of the Lagrangian

$$L(z, \lambda) = \mathcal{E}(z) - \mathcal{Q}(z) \cdot \lambda \quad (4.21)$$

with $z \in \mathbb{R}^{2|\mathbb{E}|}$ and Lagrange multiplier $\lambda \in \mathbb{R}^{3|\mathbb{V}_0|}$. The necessary condition for a saddle point $(z, \lambda) \in \mathbb{R}^{2|\mathbb{E}|} \times \mathbb{R}^{3|\mathbb{V}_0|}$ is

$$DL(z, \lambda) = (D_z L(z, \lambda), D_\lambda L(z, \lambda))^T = (D_z \mathcal{E}(z) - D_z \mathcal{Q}(z) \cdot \lambda, -\mathcal{Q}(z))^T = 0, \quad (4.22)$$

where D_z and D_λ denote the Jacobian with respect to z and λ , respectively.

Instead of directly applying Newton's method to this equation, we consider the augmented Lagrange method [Hes75; NW06]. It is a combination of the Lagrangian approach with the quadratic penalty method where we construct a series of unconstrained optimization problems in z to approximate the solution of (OPT). For the sake of completeness, we briefly recall it here. The augmented

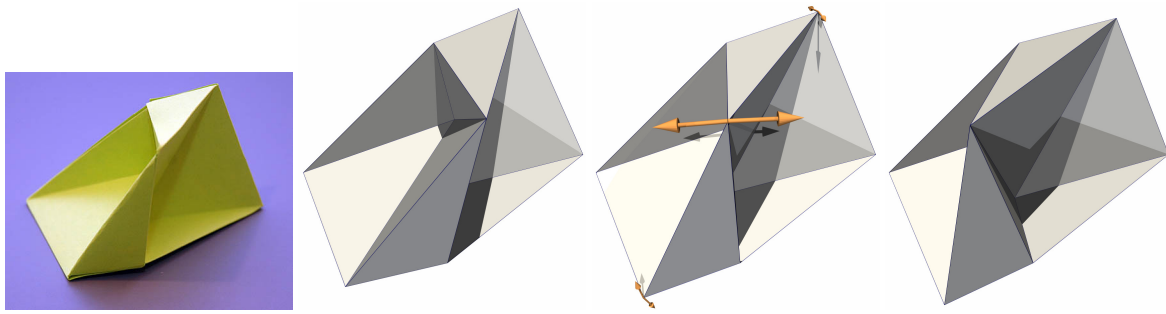


Figure 4.3: *The tangent space reveals an infinitesimal isometric variation at the classical Steffen's polyhedron (middle). Indeed, extrapolating in this positive (left) resp. negative (right) direction (solely in the θ component) allows for isometric deformations. The extrapolation is implemented via an incremental addition of the infinitesimal isometric variation coupled with a back projection onto \mathcal{M} . The photo of Steffen's polyhedron (far left) was generously provided by Laszlo Bardos from cutout-foldup.com.*

Lagrangian is defined by

$$L(z, \lambda, \mu) = \mathcal{E}(z) - \mathcal{Q}(z) \cdot \lambda + \frac{\mu}{2} \|\mathcal{Q}(z)\|_2^2, \quad (4.23)$$

and a sequence (z^k, λ^k, μ^k) of approximate solutions, approximate Lagrangian multipliers, and penalty parameters is generated by alternating between minimizing $L(\cdot, \lambda^k, \mu^k)$ to obtain z^{k+1} and computing updates to λ^k and μ^k . Hereby, the penalty parameter μ is increased until we reach sufficient attainment of the equality constraints, which is quantified by their l^∞ -norm being below a user-specified threshold. On the other hand, λ is updated by an increasingly accurate estimation of the correct multipliers λ^* solving (4.22). This can be accomplished in various ways; one popular way which we choose to follow is to set $\lambda^{k+1} = \lambda^k - \mu^k \mathcal{Q}(z^{k+1})$. Though we cannot expect the augmented Lagrange method to converge for arbitrary initial data, under reasonable assumptions, one can prove that the sequence λ^k obtained this way converges to λ^* , which significantly improves convergences compared to the quadratic penalty method, see for example [NW06]. We want to remark that though our problem (OPT) involves strict inequality constraints, the local convergence theory for the augmented Lagrange method given in [NW06, Chapter 17] still applies. The triangle inequality constraints define an open set and thus (OPT) can be seen as an equality-constrained problem over an open set. As [NW06, Theorem 17.5 & 17.6] are only concerned with local minimizers and provide local results, they still hold if the problem is only defined on an open set after possibly modifying constants describing local neighborhoods.

Unconstrained Optimization. Using the augmented Lagrange method leads to a series of unconstrained optimization problems. They are typically non-convex, i.e. we encounter indefinite Hessians $D_z^2 L$ of the Lagrangian. This means that a simple Newton's method with line search might not be an efficient and robust approach as we are not guaranteed to obtain a descent direction. To rectify this, we choose a simple adaption suggested in [NW06, Section 3.4]. First, we determine a shift τ^k such that the matrix $D_z^2 L(z^k, \lambda^k, \mu^k) + \tau^k \text{Id}$ is positive definite. This is achieved by starting with an initial estimate and then increasing τ^k until a Cholesky decomposition succeeds. Then, a descent direction is obtained by solving the linear system

$$\left(D_z^2 L(z^k, \lambda^k, \mu^k) + \tau^k \text{Id} \right) d^k = -D_z L(z^k, \lambda^k, \mu^k). \quad (4.24)$$

Along this direction we perform an Armijo-type backtracking line search. Note again, that the local convergence theory for Newton-type methods is still valid even though we minimize over an open set defined by the strict triangle inequalities, cf. [Ber99, Chapter 1].

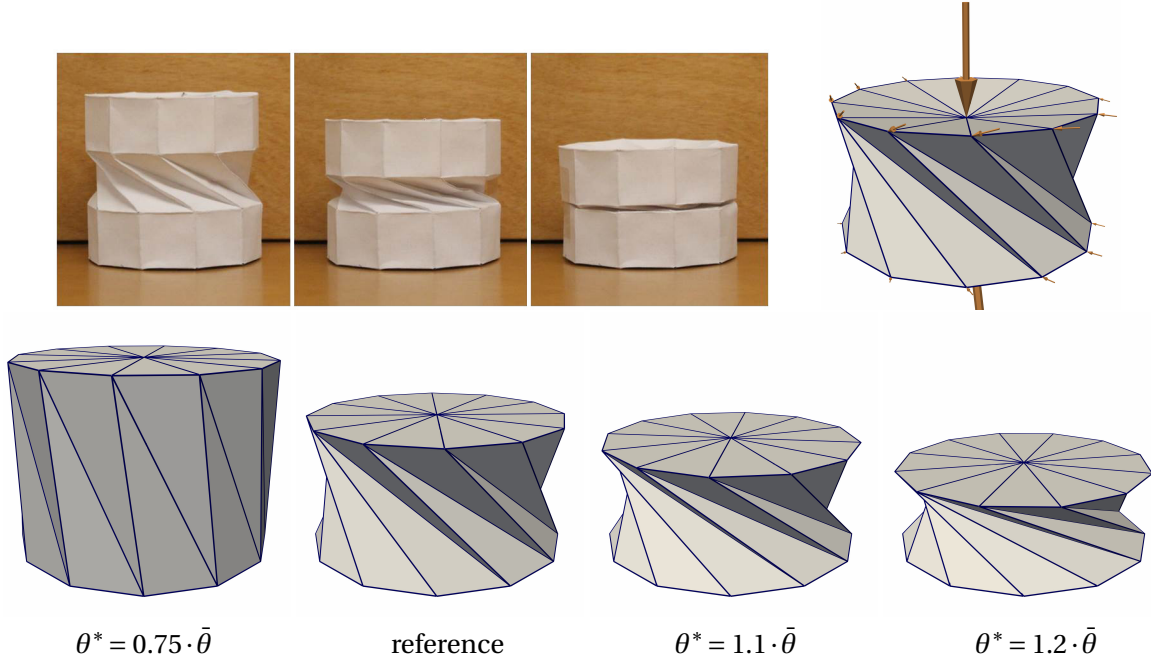


Figure 4.4: Top: Almost isometric compression of an Origami cylinder as depicted in [BVGW16] (left, with permission) along with the only infinitesimal isometric variation (right, indicated by arrows). Bottom: Optimizing (4.19) on NRIC manifold with $\delta = 0$ and hard constraints on target angles θ^* along the upper horizontal edges (relative to the reference angle $\bar{\theta} = 2.257$) leads to non-isometric deformations with as small as possible edge length distortion (from left to right 4%, 0.3%, and 0.4% average change of edge length).

To compute the descent direction as above, we need the gradient and the Hessian of our constraint functionals \mathcal{Q} . We already evaluated $D_z \mathcal{Q} \in \mathbb{R}^{3|\mathbb{V}_0| \times 2|\mathbb{E}|}$ in (4.13) and compute for the Hessian of \mathcal{Q}

$$D_z^2 \mathcal{Q} \cdot \lambda = \left(\sum_{v \in \mathbb{V}_0} \partial_{z_l} \partial_{z_k} \mathcal{Q}_v \cdot \lambda_v \right)_{l,k=1,\dots,2|\mathbb{E}|}$$

the components as

$$\begin{aligned} \partial_{z_l} \partial_{z_k} \mathcal{Q}_v(z) = & \text{vec} \left(\sum_{j=0}^{n-1} q_{01}(z) \dots \partial_{z_l} \partial_{z_k} q_{j,j+1}(z) \dots q_{n-1,0}(z) \right) \\ & + \text{vec} \left(\sum_{\substack{i,j=0 \\ i \neq j}}^{n-1} q_{01}(z) \dots \partial_{z_l} q_{i,i+1}(z) \dots \partial_{z_k} q_{j,j+1}(z) \dots q_{n-1,0}(z) \right), \end{aligned}$$

which can also be evaluated with $O(n_v)$ cost.

4.4 Experiments and Applications

Finally, we report on a range of numerical experiments on the proposed optimization framework. This way, we study its qualitative and quantitative properties and demonstrate its potential usefulness for various applications in geometry processing. To this end, we pick up the generic variational problem (OPT) introduced in Section 4.3 together with the proposed augmented Lagrange method.

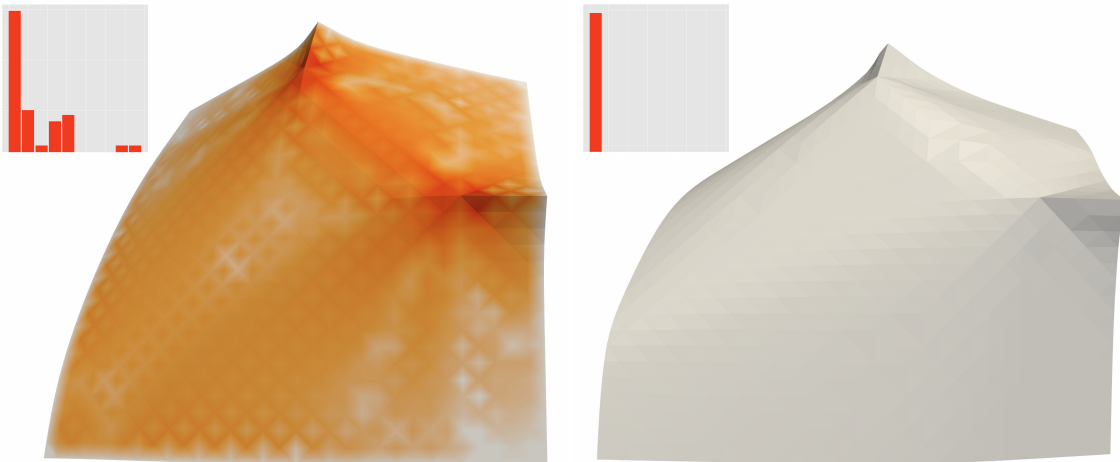


Figure 4.5: Paper folding with local constraints for dihedral angle: simulation in vertex space (left) leads to infinitesimal isometry violations whereas the result in NRIC is completely isometric (right). The absolute value of discrete Gauss curvature (as angle defect) is shown using the color map $0 \rightarrow 0.03$, which is zero everywhere on the right. Furthermore, the corresponding histograms are plotted aside the surfaces.

In the following, we discuss different objective functionals \mathcal{E} in (4.23) and depending on the application additional constraints. Note, however, that the *constraint functional* \mathcal{Q} in (4.23), which describes the NRIC manifold implicitly via (4.11), remains unchanged.

Constriction and Inflation. A simple example of an objective functional \mathcal{E} is given by the dissimilarity to some given z^* on the linear space $\mathbb{R}^{2|E|}$ measured by the deformation energy, i.e. $\mathcal{E}(z) = \mathcal{W}[z^*, z]$, where \mathcal{W} is an elastic deformation energy as discussed in the previous section. For example, in Figure 4.1, we have used the nonlinear energy defined in (4.14) along with coordinate constraints on a certain subset of edge lengths to simulate a “constriction” of a sphere along curves or creating cartoon-like characters by inflating for instance hands and feet (cf. [KNP14]).

Elastic averages. Let $X_1, \dots, X_n \in \mathcal{N}$ be a set of example shapes sharing the same connectivity. Frequently, one is interested in a mean or average shape, cf. [TSSH15]. Given an elastic deformation energy, a so-called weighted elastic average is defined to be the minimizer of a weighted sum of elastic energies for deformations from the input shapes to the free shape. This can be translated directly to our NRIC manifold, i.e. for a given elastic deformation energy \mathcal{W} on \mathcal{M} and convex weights $\mu \in \mathbb{R}^n$ we define the weighted elastic NRIC average as a solution of (OPT) with

$$\mathcal{E}(z) = \sum_{i=1}^n \mu_i \mathcal{W}[Z(X_i), z]. \quad (4.25)$$

In Figure 4.2, we show (the reconstructions of) weighted elastic NRIC averages for a set of six hand shapes and different weights μ_1, \dots, μ_6 . Here, we have used the nonlinear deformation energy (4.14) in (4.25).

Infinitesimal rigidity. With the tangent space at hand, one can verify the *infinitesimal rigidity* of a discrete surface with NRIC $z \in \mathcal{M}$. Infinitesimal rigidity is defined as the non-existence of an infinitesimal isometric variation $w \in T_z \mathcal{M}$ with $P_l w = 0$ and $w \neq 0$, where P_l is the projection onto the length component, i.e. $P_l(l, \theta) = l$, see for example Figure 4.3. Its absence—i.e. the existence of an infinitesimal isometric variation—is a necessary condition for a shape to *not* be *rigid*, where rigidity

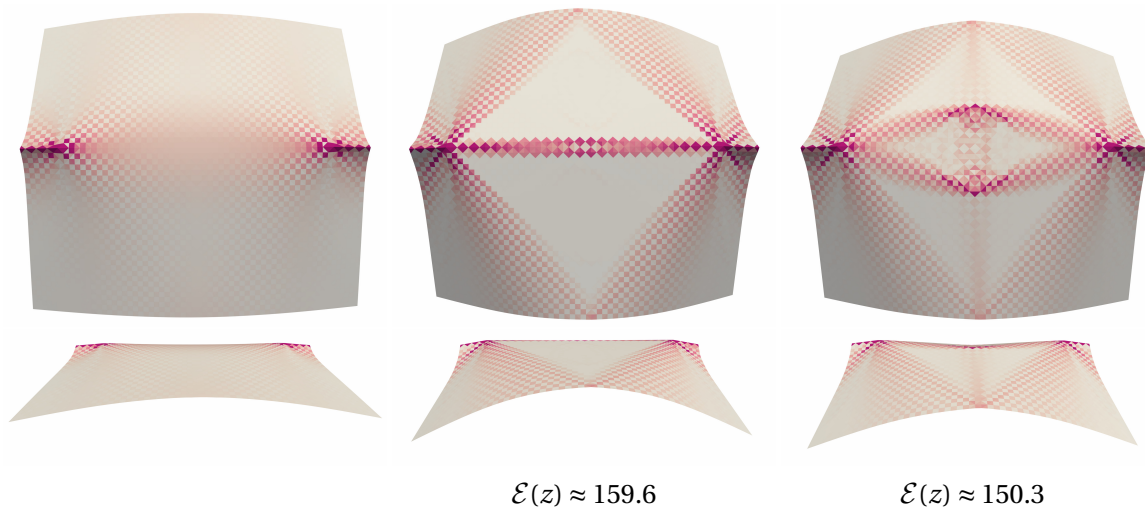


Figure 4.6: *Once more paper folding with local constraints for dihedral angle: simulation in vertex space (left) leads to infinitesimal isometry violations whereas the results in NRIC are completely isometric (middle, right). The result in the middle shows a local minimum obtained by the augmented Lagrange method when increasing the penalty parameter μ (too) aggressively exhibiting higher deformation energy (shown below) than the right result where the penalty was increased more conservatively. Triangle-averaged mean curvature is shown as color map $0 \leq \text{color} \leq 0.005$. Flat triangular regions can only be seen in the NRIC simulations.*

is defined as the non-existence of a continuous one-parameter family of isometric deformations. Or in other words, infinitesimal rigidity implies rigidity. Note, however, that this is surely not a sufficient condition, which we can also observe in Figure 4.4.

We verify infinitesimal rigidity by computing if the kernel of $DQ(z)$ has a non-trivial intersection with the θ subspace of $\mathbb{R}^{2|E|}$, namely the kernel $\ker P_l$. This intersection is given by $\ker \begin{pmatrix} B_{T_z \mathcal{M}} \\ B_\theta \end{pmatrix}$, where $B_{T_z \mathcal{M}}$ is a matrix whose columns form an orthonormal basis of $T_z \mathcal{M}$ and B_θ is the canonical basis of $\ker P_l$. We compute a singular value decomposition (SVD) of this matrix and evaluate the smallest singular value λ_0 . If $\lambda_0 = 0$, then there exists an infinitesimal isometric variation. Otherwise, the singular value provides a quantitative measure for the lack of such an infinitesimal isometric variation.

Isometric deformations via additional constraints. Interesting applications can be described by considering (OPT) along with the simple objective $\mathcal{E}(z) = \mathcal{W}[z^*, z]$ and additional, simple coordinate constraints. For example, in Figure 4.1, we have seen experiments where we posed length constraints $l_i = l_i^*$ for $i \in I$ on the coordinates $z = (l, \theta)$ for some index set $I \subset E$ and prescribed target lengths l^* . Similarly, we obtain an elegant way to simulate the *isometric* folding of a (flat) sheet of paper given in NRIC as $z^* = (l^*, \theta^*)$ where $\theta^* = 0$. To this end, we pose the length constraints $l_e = l_e^*$ for all $e \in E$ along with $\theta_i = \text{const.} \neq 0$ if $i \in I$ for some index set $I \subset E$. Note, that under these length constraints the nonlinear and quadratic energy approach agree if we compute the weights in (4.19) from the reference z^* . For example, in Figure 4.5, we impose the constraint $\theta_i = \pi/2$ for the edges on two short line segments on two neighboring sides of the sheet. Since all edge lengths are fixed and all other dihedral angles are degrees of freedom for the minimization of (4.14) on \mathcal{M} , we obtain a perfect isometric deformation as indicated by the vanishing discrete Gauß curvature (Figure 4.5, right). In comparison, vertex-based methods as [GHDS03] or [HRWW12] do not achieve a perfect isometry—even when computed with a very high membrane stiffness (Figure 4.5, left). For the optimization in nodal positions, we used the energy $X \mapsto \mathcal{W}[z^*, Z(X)]$ with a shell thickness parameter $\delta = 10^{-3}$. In fact, when further reducing δ we observed numerical instabilities. This is

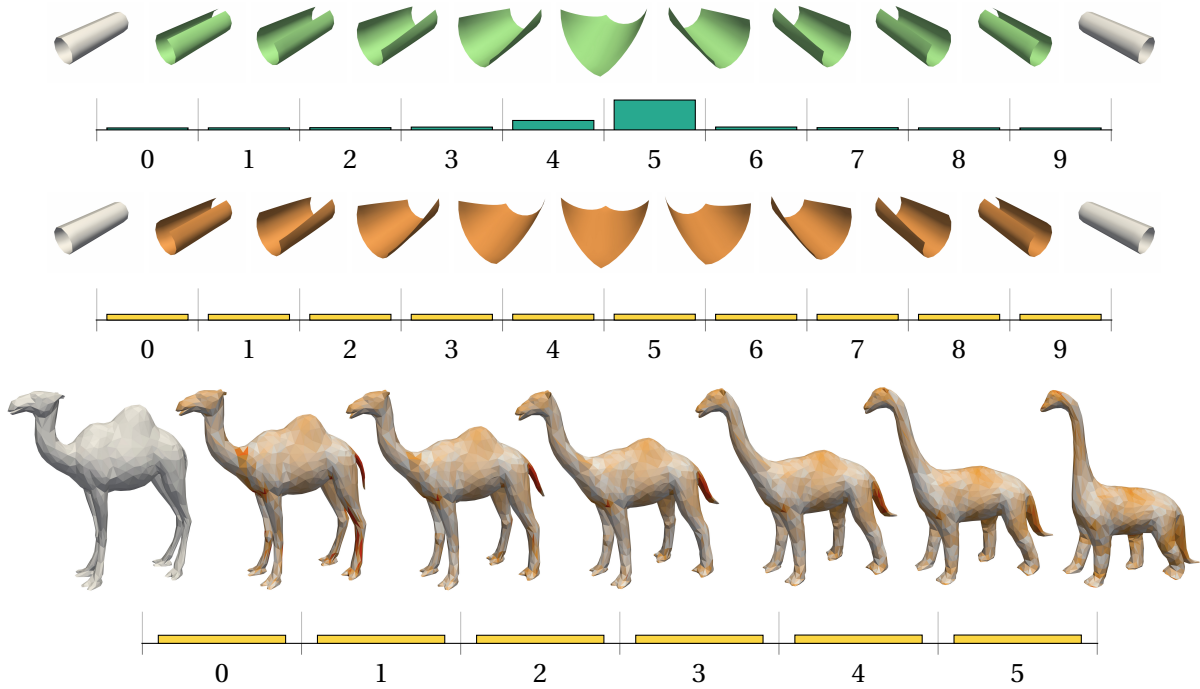


Figure 4.7: *Bottom: discrete geodesic in NRIC with input data from [AR18] and membrane distortion as colormap (0 \rightarrow ≥ 1); above: Linear interpolation in ambient space $\mathbb{R}^{2|E|}$ as in [FB11] with energy distribution (green) vs. geodesic interpolation on \mathcal{M} with constant energy distribution (orange).*

due to the fact that isometric deformations induce bending distortions only but optimizing bending energies in terms of nodal positions is a highly nonlinear singular perturbation problem that quickly triggers numerical issues. Conversely, the corresponding bending energy in NRIC is quadratic.

Besides vanishing Gauß curvature, pure isometric deformations exhibit further characteristics, as illustrated in Figure 4.6. In this example, we have a very similar setup as in Figure 4.5 but we pose the angle constraints on two opposite sides. First, let us point out that we observed convergence of the augmented Lagrange method described above to different local minima when using different parameters for the increase of the penalty parameter μ . We show two different local minima in Figure 4.6 where we obtained the lowest energy value when increasing μ conservatively (shown on the right). Now, since the NRIC results are perfectly isometric and rather smooth deformations of the flat sheet one can indeed observe effects predicted analytically by the Hartman-Nirenberg theorem [HN59; Hor11]. Loosely speaking, isometric deformations of a flat sheet can locally be described either as flat patches or segments of straight lines (rulings) going to the boundary. In the middle and right columns of Figure 4.6 one can easily identify flat triangular regions as well as a cone-like bundle of straight lines propagating towards the boundary. These structures are not reflected by the vertex-based numerical minimizer already discussed above (Figure 4.6, far left).

Geodesics. So far, we have only considered static examples where a single shape was optimized subject to external forces or boundary conditions. However, one can easily generalize (OPT) to optimize for multiple shapes simultaneously, for instance, to simulate a kinematic behavior. Hence, we consider the computation of time-discretized geodesics on the NRIC manifold. Recall from Section 3.2, that with the Riemannian structure introduced in Section 4.2 the time-discrete path energy

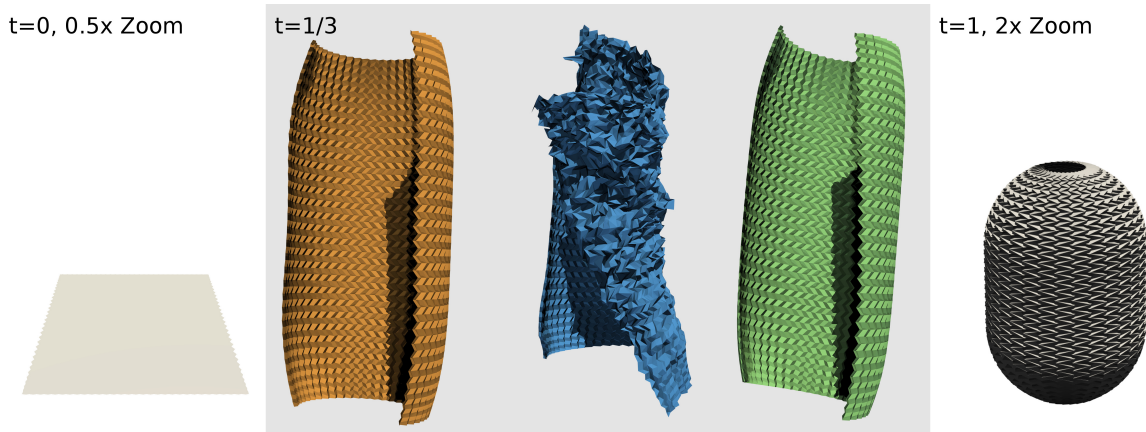


Figure 4.8: Intermediate shapes at $t = 1/3$ of a discrete geodesic between two perfectly isometric end shapes (grey) taken from [DVTM16] obtained via NRIC optimization (orange) and vertex-based methods as in [HRWW12] (blue) resp. [FB11] (green). Note that we preserve the isometry due to our hard length constraints. In contrast, the vertex-based methods get either stuck in local minima (blue) or reveal artifacts such as unnatural asymmetries (green).

is given by

$$E[z_0, \dots, z_K] = K \sum_{k=1}^K \mathcal{W}[z_{k-1}, z_k], \quad (4.26)$$

for discrete paths (γ_0, \dots, z_K) with K steps. In particular, discrete geodesics—i.e. minimizers of (4.26)—obey a constant speed property, which means there is a uniform energy distribution of $\mathcal{W}[z_{k-1}, z_k] \approx \text{const.}$ along the curve.

The path energy in (4.26) can be considered as an objective functional in (OPT). Note, however, that this increases the number of free variables substantially. In Figure 4.7, we show different time-discrete geodesics in NRIC where we use the quadratic deformation energy (4.19) in (4.26). In particular, we compare for end shapes being two oppositely bent plates our NRIC geodesic (orange) to the linear interpolation (green) in the embedding space $\mathbb{R}^{2|E|}$, which corresponds to a naive transfer of the projection approach by [FB11]. As indicated by the histogram plots, the discrete constant speed property can only be obtained for the NRIC formulation.

Furthermore, we can combine the computation of time-discrete geodesics with further constraints on the coordinates, e.g. to simulate isometric deformation paths. For example, in Figure 4.8, we compare the computation of (almost) *isometric* geodesic paths between perfectly isometric end shapes taken from [DVTM16] to vertex-based methods. A similar example is shown in Figure 4.9, where the first input shape $z_0 = (l^*, \theta^*)$ describes a hyperbolic monkey saddle and the second input shape is given by a reflection $z_K = (l^*, -\theta^*)$ of the saddle. Figure 4.9 demonstrates that our approach is able to realize a perfectly isometric deformation path (orange) by enforcing $l_k = l^*$ for all $0 < k < K$, whereas vertex based optimization methods fail.

Time discretization. In Section 4.2, we introduced a modified time-discrete Riemannian exponential map and logarithm for the NRIC manifold. The discretization is controlled by the number of steps K in a discrete geodesic and converges for $K \rightarrow \infty$ to the corresponding continuous notions. We demonstrate this convergence empirically for the SCAPE dataset in Figure 4.10.

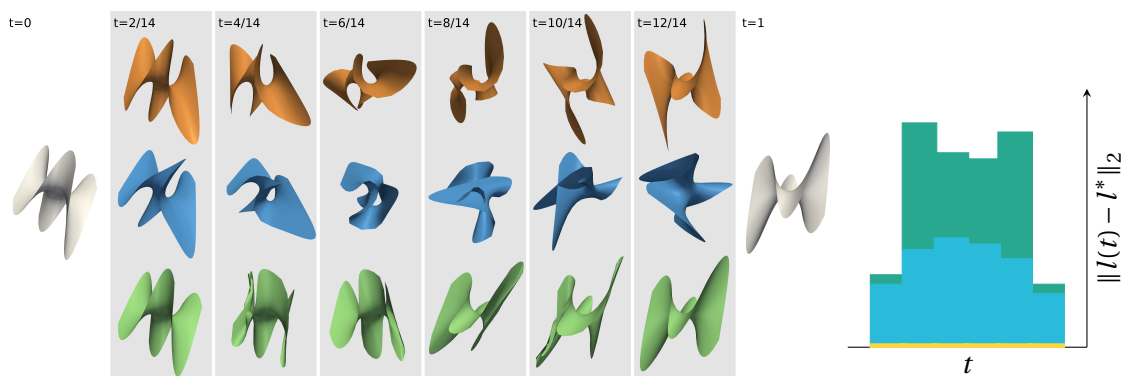


Figure 4.9: *Isometric geodesic paths. Left input shape (l^*, θ^*) as hyperbolic monkey saddle, right input shape is the reflection $(l^*, -\theta^*)$. Comparison of discrete geodesics computed in NRIC (orange, perfectly isometric) and by methods based on nodal positions (i.e. [HRWW12] (blue) and [FB11]). The latter approaches are not able to resolve pure isometric geodesics as indicated by a histogram of varying lengths on the right.*

4.5 Conclusion and Outlook

In this chapter, we have introduced a framework to phrase and handle geometric variational problems on discrete surfaces in NRIC—the vector stacking edge lengths and dihedral angles. After introducing the necessary background on discrete integrability conditions akin to the Gauß–Codazzi equations originally introduced by Wang, Liu, and Tong [WLT12], we observed that they allow us to represent the space of immersions of a discrete surface as an implicit submanifold. We demonstrated that the readily available tangent space of this submanifold can be used to numerically study infinitesimal isometric variations. Furthermore, we developed an approach to numerically solve general optimization problems formulated in NRIC. Through a range of numerical examples, we showed that this is particularly useful for the simulation of isometric deformations.

There are still a few gaps in our theoretical understanding of the discrete integrability conditions. First, it would be worthwhile to prove that the NRIC manifold is indeed a manifold, i.e. that the differential of the integrability conditions has constant rank. This was true in all of our numerical experiments but remains to be shown in general. Furthermore, it would be interesting to establish a sense of convergence of the discrete integrability conditions to their continuous counterpart—the Gauß–Codazzi equations. This would round off our understanding of the space of immersions.

From a practical point-of-view, it would be interesting to extend the NRIC framework with additional functionals and constraints on the geometry of the surface, such as ones describing its developability. Furthermore, while NRIC are by construction well-suited to describe local geometric properties, formulating non-local properties or points is challenging. One idea to realize them would be to construct the immersion—or its necessary parts—using the method from the proof of Theorem 4.6 and differentiate it. This could be a suitable task for modern automatic differentiation software. Nevertheless, this would introduce highly nonlinear and nonlocal terms to the optimization potentially limiting the performance of our method. This introduces the challenge of devising different ways to combine NRIC-based modeling with point constraints. Implementation-wise, we can currently only handle simply connected surfaces. An extension to higher-genus surfaces would require to include integrability conditions along non-contractible paths that generate the fundamental group. This would lead to more global constraints in our optimization problems. Typical examples of surfaces in geometric modeling have only a small number of generators of the homology group. However, this necessity of complicated constraints is a general limitation of our method compared to nodal positions.

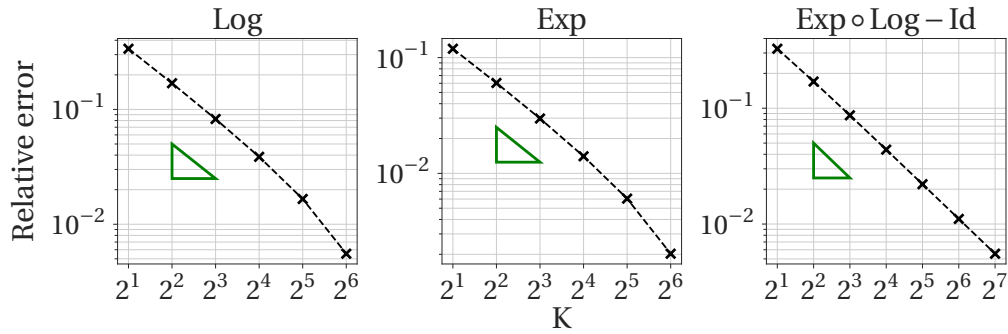


Figure 4.10: Convergence of time-discrete exponential and logarithm for $K \rightarrow \infty$ on the SCAPE dataset. For the logarithm and exponential, we show the relative error in the weighted L^2 -norm using $K_{\max} = 128$ as pseudo ground truth. Furthermore, due to the tangent space projection the discrete exponential does not coincide with the inverse of the discrete logarithm. But, we observe that it is an approximation with decreasing relative error in the weighted L^2 -norm for $K \rightarrow \infty$.

Finally, a potential next step would be to find a parametrization of the NRIC manifold. For closed surfaces, it has dimension $3|\mathbf{V}| - 6 \ll 2|\mathbf{E}|$ and thus could such a parametrization drastically speed-up computations. Amenta and Rojas [AR18] and Chern *et al.* [CKPS18] observed that in some cases only edge lengths respectively dihedral angles can be used for this. However, this does not work in general. An alternative approach would be to parametrize only a submanifold of \mathcal{M} that we restrict our problems to. This is the object of study in the next chapter, where we derive the submanifold of interest from data.

Chapter 5

Sparse Principal Geodesic Analysis

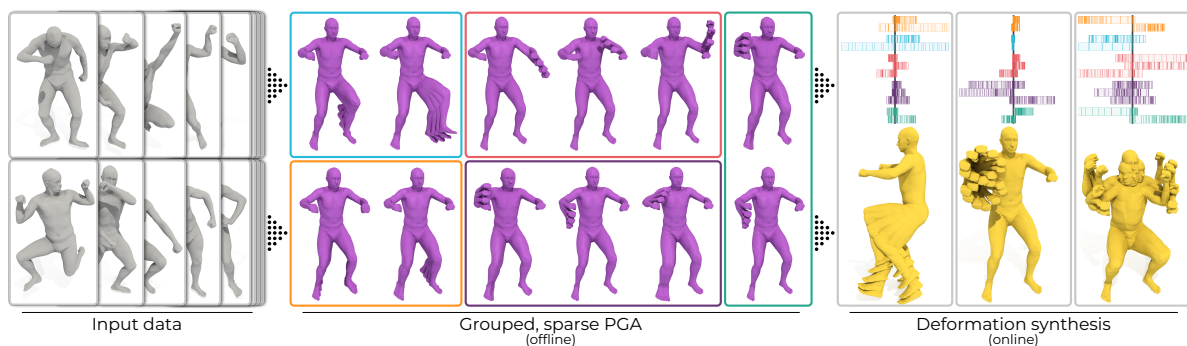


Figure 5.1: Sparse principal geodesic analysis for data-driven nonlinear deformation synthesis. In a precomputation phase, sparse nonlinear deformation modes (middle) are extracted from a dataset of poses of a non-rigid shape (left) using the shape space of discrete shells and Nonlinear Rotation-Invariant Coordinates. Then, the modes are grouped by the overlap of their support (colored frames) and this is used to create an efficient parametrization of the corresponding data submanifold. In the online phase, this parametrization is used to synthesize nonlinear deformations (right) at near-realtime rates.

One of the more advanced applications of shape spaces is the computation of shape statistics. This begins with computing nonlinear means of input data and continues with computing principal modes of variation. The latter can be described using data approximating submanifolds of the shape space. To accurately capture shape variability, the nonlinear structure of the shape space is often indispensable. Linear approaches such as Principal Component Analysis (PCA) are not well-suited to describe strongly nonlinear deformations effectively and fail to represent invariances such as rigid body motions. Instead, using Principal Geodesic Analysis (PGA)—a generalization of PCA to manifolds and the prototypical algorithm to construct data approximating submanifolds—on the space of discrete shells leads to convincing results [HZRS18].

However, these shape space-based methods typically require the solution of nonlinear (variational) problems, which is computationally expensive. This is specifically a limitation if it is required in the parametrization of the resulting submanifold. For example, in the case of PGA, this submanifold is parametrized using the Riemannian exponential map, which for the space of discrete shells requires solving a nonlinear equation. This parametrization is a crucial ingredient for most applications based on the constructed submanifold. Hence, if we do not have a closed form of the exponential map available it is not a feasible approach if fast results are required—this was also observed by Heeren *et al.* [HZRS18]. Our goal in this chapter is therefore to devise a submanifold construction that equips it with additional structure enabling a more efficient (approximate) parametrization.

A crucial ingredient for this will be the notion of sparsity of nonlinear variations. To obtain a sensible regularization inducing such a sparsity, we will rely on the Nonlinear Rotation-Invariant Coordinates (NRIC) discussed in the previous chapter. Their natural connection to elastic distortion allows us to use conventional notions of spatial sparsity, which would not be reasonable in nodal positions. Furthermore, using the NRIC manifold as a representation of the space of discrete shells will significantly facilitate the formulation of our approach due to their inherent invariance to rigid body motions. Finally, they will also provide our data approximating submanifold with a benevolent extrinsic geometry—since, recall, working linearly in $\mathbb{R}^{2|E|}$ already often yields acceptable results—which will be crucial for the efficient parametrizations.

Our approach will be based on PGA. We will modify the problem of computing the approximating submanifold—which is linearized by passing to the tangent space at a mean shape—to include a sparsity-inducing regularization. This will not only lead to sparse modes of variation but also to a group-wise decoupling of modes, i.e. them having mostly disjoint support in NRIC. Based on this decoupling, we will construct two different approximate parametrizations. The first will approximate the exponential map on individual groups by a grid-based multilinear interpolation and combine the results affinely. This will already provide an efficient method useful for shape editing and interpolation tasks and generate vital structural insights. However, since it is limited in the number of dimensions it can handle, we will also introduce a second parametrization based on neural networks incorporating the product structure of our data approximating submanifold. Our method is visually outlined in Figure 5.1.

Previous methods concerned with including sparsity or localization into data-driven reduced modeling of deformations—such as [NVW+13; HYZ+14; WLZH17]—concentrated on linear approaches. For example, Neumann *et al.* [NVW+13] used a sparsity- and localization-inducing term on nodal positions yet the linearity of the construction severely limited the size of possible deformations. Follow-up work proposed to alleviate this issue by using other representations. Notably, Wang *et al.* [WLZH17] and Liu *et al.* [LLW+19] also used edge lengths and dihedral angles for this purpose. However, they did not consider the nonlinearity of the NRIC manifold which leads to unnatural results (see e.g. Figure 5.14). In contrast, we use this nonlinearity to obtain a less complicated regularization and an effective generalization to new deformations. Looking at machine learning terminology, our approach could be considered as an attempt to ‘disentanglement learning’, i.e. the identification of individual factors in data. Indeed, one of the potential directions of future work we will propose in Section 5.7 will be to extend our line of work to use its insights for the construction of autoencoders.

The remainder of this chapter will be structured as follows: We begin with a recapitulation of PGA and with applying it to the NRIC manifold in Section 5.1. Then we will propose our modification to obtain sparse modes in Section 5.2. For the resulting sparse modes, we will discuss the decoupling and grid-based interpolation in Section 5.3. In Section 5.4, we will discuss the necessary numerical tools to realize our method. We will use this for experimental validation and some proof-of-concept applications in Section 5.5. Then, in Section 5.6, we will discuss the neural network-based parametrization and its experimental validation. Finally, we will summarize and provide an outlook to potential future work in Section 5.7.

Remark. The bulk of this chapter is the result of joint work with Klaus Hildebrandt and Martin Rumpf and as such an extension of the publication [SHR20]. The neural network-based parametrization in Section 5.6 is additionally the result of joint work with also Benedikt Wirth to be published in [SHWR23].

5.1 Principal Geodesic Analysis

Principal Geodesic Analysis (PGA) is a technique introduced by Fletcher *et al.* [FLPJ04] to construct data approximating submanifolds. It can be understood as a generalization of Principal Component Analysis (PCA) to the nonlinear manifold setting. The primary goal is to construct a low-dimensional submanifold such that given data points on the manifold have minimal distance to this submanifold. To do so, PGA considers submanifolds parametrized by the Riemannian exponential map on the tangent space at a mean point and linearizes the approximation by passing to this tangent space. PGA was originally introduced for a shape space with medical application and has since then been applied to many other shape spaces and general manifolds. Below, we will recapitulate the approach in more detail and in the same breath apply it to the NRIC manifold. A schematic overview of the resulting time-discrete PGA is given in Algorithm 1.

To construct the submanifold, we are given N input shapes $z^1, \dots, z^N \in \mathcal{M}$. As explained above, we want to pass to a tangent space and thus linearize the approximation problem. To this end, we need to choose a suitable point $\bar{z} \in \mathcal{M}$ whose tangent space we will use. For this, we consider the Riemannian center-of-mass¹, which is the Fréchet mean with respect to the Riemannian distance, i.e. given by

$$\bar{z} := \operatorname{argmin}_{z \in \mathcal{M}} \sum_{n=1}^N \operatorname{dist}_g^2(z^n, z). \quad (5.1)$$

We choose this point because working in its tangent space leads to a similar size of error for the approximation of manifold operations by linear ones in the tangent space for all input shapes.

Then we consider *geodesic submanifolds* at \bar{z} , that is J -dimensional submanifolds $\mathcal{M}^J \subset \mathcal{M}$ with $\bar{z} \in \mathcal{M}^J$ such that every geodesic in \mathcal{M}^J between \bar{z} and an arbitrary $z \in \mathcal{M}^J$ is also a geodesic in the ambient manifold \mathcal{M} . These are generalizations of affine subspaces of Euclidean space. In fact, geodesic submanifolds of \mathbb{R}^d at the origin are exactly linear subspaces. Geodesic submanifolds at \bar{z} can be parametrized locally around \bar{z} over a J -dimensional linear subspace $\mathcal{U}^J \subset T_{\bar{z}}\mathcal{M}$ by the Riemannian exponential map. In the following, we assume that all input points are close enough to \bar{z} such that the Riemannian logarithm (and exponential map) at \bar{z} are defined for them. Thus, we have reduced the problem to finding the basis $u_1, \dots, u_J \in T_{\bar{z}}\mathcal{M}$ of such a subspace.

We want to construct \mathcal{M}^J such that the distances $\operatorname{dist}_g(z^n, \mathcal{M}^J)$ of the input shapes to it are minimal. Since evaluating such distances is computationally expensive in general, we aim to replace them by an approximation in $T_{\bar{z}}\mathcal{M}$. This can be done by considering the approximation $\operatorname{dist}_g^2(z^n, z) \approx \|\log_{\bar{z}} z^n - \log_{\bar{z}} z\|_g^2$, which is an equality if $z = \bar{z}$. Now, assume again that \mathcal{M}^J is parametrized over a subspace $\mathcal{U}^J \subset T_{\bar{z}}\mathcal{M}$, which also forms the tangent space of \mathcal{M}^J at \bar{z} , i.e. $T_{\bar{z}}\mathcal{M}^J = \mathcal{U}^J$. Then, since we assume \mathcal{M}^J to be a geodesic submanifold at \bar{z} , we have $\log_{\bar{z}} z \in \mathcal{U}^J$ and can thus estimate

$$\operatorname{dist}_g^2(z^n, \mathcal{M}^J) = \min_{z \in \mathcal{M}^J} \operatorname{dist}_g^2(z^n, z) \approx \min_{z \in \mathcal{M}^J} \|\log_{\bar{z}} z^n - \log_{\bar{z}} z\|_g^2 = \|\log_{\bar{z}} z^n - \mathbf{P}_{\mathcal{U}^J}(\log_{\bar{z}} z^n)\|_g^2, \quad (5.2)$$

where $\mathbf{P}_{\mathcal{U}^J}$ is the orthogonal projection onto \mathcal{U}^J with respect to $g_{\bar{z}}$. Hence, we have to find a subspace \mathcal{U}^J such that the distances of the input shapes' logarithms to this space is minimal. This is, of course,

Algorithm 1 Time-discrete PGA in NRIC

Input: Input shapes $z^1, \dots, z^N \in \mathcal{M}$

Output: J -dim. approximating submanifold \mathcal{M}^J

- 1: Compute the elastic mean \bar{z} of the input shapes z^1, \dots, z^N .
 - 2: Compute $v^n = \operatorname{Log}_{\bar{z}}^K(z^n)$ for $n = 1, \dots, N$.
 - 3: Perform a PCA of the v^n with $g_{\bar{z}}$ as the underlying scalar product to compute the J principal modes of variation $u_1, \dots, u_J \in T_{\bar{z}}\mathcal{M}$.
 - 4: Set $\mathcal{M}^J = \operatorname{Exp}_{\bar{z}}^K(\operatorname{span}\{u_1, \dots, u_J\})$.
-

¹Sometimes also called the Karcher mean, cf. [Kar14].

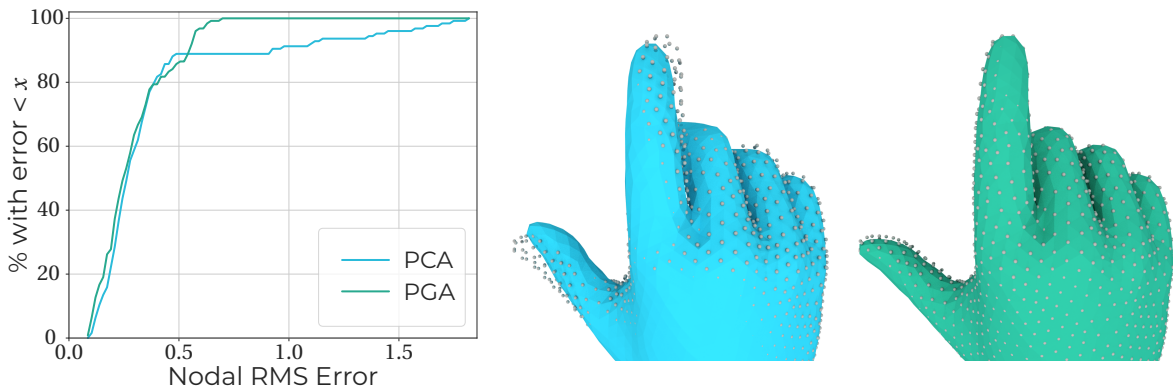


Figure 5.2: Comparison of nonlinear PGA (green) with linear PCA (blue) in NRIC. The histogram shows the distribution of errors in terms of mismatch of nodal position for reconstruction of the input shapes. Moreover, we compare for a selected input shape (vertices shown as gray point cloud) the approximation using PCA and PGA respectively. In both cases, we used $J = 8$ components and both models achieved a linear approximation quality of about 98%.

exactly the problem statement of a PCA of these logarithms and we can phrase this variationally as

$$\begin{aligned} & \underset{\substack{U \in \mathbb{R}^{2|\mathbb{E}| \times J} \\ W \in \mathbb{R}^{J \times N}}}{\text{minimize}} && \|V - UW\|_g^2 \\ & \text{subject to} && g_{\bar{z}}(u_i, u_j) = \delta_{ij} \text{ for } i, j \in \{1, \dots, J\}, \end{aligned} \quad (5.3)$$

where $V \in \mathbb{R}^{2|\mathbb{E}| \times N}$ is the matrix containing the v^i as columns and the columns u_j of U form an orthonormal basis of \mathcal{U}^J . The approximation error is measured in the Frobenius norm $\|\cdot\|_g$ weighted according to $g_{\bar{z}}$. Since all v^i lie in the tangent space, the u_j minimizing (5.3) will do so as well and hence there is no need for an explicit tangent space constraint.

As explained above, we parametrize \mathcal{M}^J via the exponential map, i.e. we set

$$\mathcal{M}^J = \mathcal{M}[\mathcal{U}^J] := \exp_{\bar{z}}(\mathcal{U}^J) \subset \mathcal{M} \quad (5.4)$$

as submanifold approximating the data. Moreover, for each u_j , the curve $t \rightarrow \exp_{\bar{z}}(t u_j)$ can be interpreted as the j th nonlinear mode of shape variation of the input data. In the following sections, we will mostly drop the explicit dependence on the dimension J and instead write \mathcal{U} for the linear subspace of $T_{\bar{z}}\mathcal{M}$ and $\mathcal{M}[\mathcal{U}]$ for the corresponding submanifold.

To perform all these computation in practice, we replace the logarithm and exponential maps with their time-discretizations introduced in Section 4.2. For the Riemannian center-of-mass, we replace the squared distances in (5.1) by discrete path energies. During our experiments, we realized that already $K = 1$ yields a sufficient approximation and thus we set

$$\bar{z} \approx \underset{z \in \mathcal{M}}{\text{argmin}} \sum_{n=1}^N \mathcal{W}(z^n, z), \quad (5.5)$$

which is also called the *elastic mean* (cf. [RW09]). Performing the computation on the NRIC manifold instead of in nodal positions makes them a lot more straightforward. Indeed, the approach to PGA on the space of discrete shells proposed by Heeren *et al.* [HZRS18] required careful and costly handling of (local) nonlinear variations to parametrize the manifold instead of just evaluating the exponential map. We can circumvent this in our approach, because we have access to the shape space's tangent space.

In comparison to linear PCA in $\mathbb{R}^{2|\mathbb{E}|}$, i.e. using the (projected) linear mean and linear differences, the nonlinearity allows us to better capture the articulation of the input shapes. This means for the

same linear approximation quality, i.e. $\frac{\|V-UW\|^2}{\|V\|^2}$, the actual approximation of the input shapes z^n is more accurate using nonlinear PGA and the exponential map than using linear PCA and orthogonal projection on \mathcal{M} . A comparison of this PGA and a PCA in NRIC coordinates is given in Figure 5.2.

However, PGA requires the computation of numerous evaluation of the Riemannian logarithm and exponential map. This makes it costly if these evaluations are computationally expensive, which is, in general, the case for our application. This is especially important for the exponential map since it is used to parametrize the constructed submanifold and thus required for all its applications. For example, this makes the method established so far not feasible for realtime shape edition applications. The computational costs of the logarithms are typically not as important as they are performed in a pre-computation phase. Hence, in the following sections, we will modify the submanifold construction by modifying (5.3) and introduce approaches to efficiently parametrize the resulting submanifolds.

5.2 Sparsity and Decoupling

By definition, conventional PGA outputs an orthonormal set of dominant deformation modes. However, to be able to approximate deformations with only a few modes and be orthogonal, these modes typically have a large support which largely overlaps for different modes. In this section, we endeavor to replace these modes by sparse ones. On the one hand, this often allows to easier assign modes a semantic meaning. For example, we would expect modes related to the movement of different joints or muscle groups on humanoid shapes. On the other hand, we expect—and will see in the next sections—that sparse modes with disjoint support facilitate efficient parametrization schemes for data approximating submanifolds.

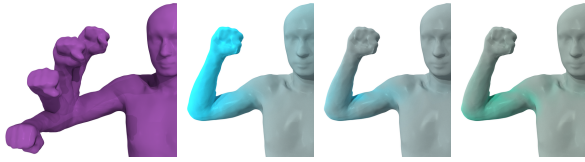


Figure 5.3: Support of deformation (left) in nodal positions (middle left), edge lengths (middle right) and dihedral angles (right). Change of nodal positions and absolute length change are shown on the same scale as color map 0 \rightarrow ≥ 0.05 , while change of dihedral angle is shown by 0 \rightarrow 1.3.

To compute sparse deformation modes, we first need an appropriate notion of support and sparsity. We could consider the change of nodal positions for this task. However, the support of deformations in nodal positions often disagrees with the places of elastic distortions and also semantically plausible notions of localization. For example, in Figure 5.3, we see this phenomenon for the movement of the forearm of a humanoid model. In contrast, the support of deformations in NRIC naturally agrees with locations of elastic distortions (cf. Section 4.3.1). Moreover, it frequently matches our semantic

expectations—for example, in Figure 5.3. Thus, the usage of NRIC will form a crucial ingredient to our approach below. Nevertheless, we expect it to generalize to other shape spaces where a sensible sparsity regularization—e.g. through appropriate spatial coordinates—is available.

Using NRIC, we follow the common approach of considering a sparsity-inducing norm. However, using an unweighted l^1 -norm on tangent vectors would be problematic as they consist of edge lengths and dihedral angles varying on vastly different scales. To this end, we consider the family of weighted L^p -norms from Section 4.3.1. With this L^1 -norm at hand, we generalize the PGA to a Sparse PGA (SPGA) as follows.

$$\begin{aligned}
 & \underset{\substack{U \in \mathbb{R}^{2|E| \times J} \\ W \in \mathbb{R}^{J \times N}}}{\text{minimize}} && \|V - UW\|_g^2 + \lambda \sum_{j=1}^J \|u_j\|_1 \\
 & \text{subject to} && u_j \in T_{\bar{z}}\mathcal{M} \text{ for } j \in \{1, \dots, J\} \\
 & && \max_{i \in \mathbb{R}^N} |w_{ji}| \leq 1 \text{ for } j \in \{1, \dots, J\}.
 \end{aligned} \tag{5.6}$$

We dropped the orthonormality constraints as we do not aim for an orthonormal basis anymore but favor sparsity. To ensure that the magnitude of the coordinates of the u_j does not simply shrink to achieve a small L^1 -norm while the weights grow accordingly, we introduce a bound on the magnitude of the weights' entries. Different from standard PGA, we have to impose that the modes u_j are indeed tangent vectors in $T_{\bar{z}}\mathcal{M}$, which ensures that they yield admissible, infinitesimal deformations of the mean \bar{z} . Dropping this constraint, which can be written equivalently as $D_z Q(\bar{z})u_j = 0$, one would leave the NRIC manifold \mathcal{M} and there is no guarantee that u_j represents a geometrically admissible variation of the underlying triangular mesh. For example, in Figure 5.4, we added just the L^1 -regularization to a linear PCA in NRIC. As we can see, this led to modes of variation with non-matching support between edge lengths and dihedral angles. Hence, their usability for approximation and the synthesis of new shapes is quite limited since it precludes extrapolation via the exponential map.

In contrast to, for example, Neumann *et al.* [NVW+13], we do not include a term that explicitly enforces a connected, localized support of the deformation modes. This is a natural byproduct of the combination of data approximation, sparsity, and admissibility—i.e. the tangent space constraint—in our approach. For the same reason, SPGA modes tend to (group-wise) decouple which we will see in more detail in the next section.

Finally, we can—again by virtue of the exponential map—define a submanifold

$$\mathcal{M}[\mathcal{U}] = \{\text{Exp}_{\bar{z}}^K(v) \mid v \in \mathcal{U}\}$$

of the NRIC manifold \mathcal{M} containing nonlinear deformations.

5.3 Submanifold Decomposition and Deformation Synthesis

In this section, we will propose an approximation of $\mathcal{M}[\mathcal{U}]$ with \mathcal{U} originating from the sparse modes computed in (5.6) that can be computed efficiently. We will do this in two steps: First, we will introduce an approximate direct sum structure on $\mathcal{M}[\mathcal{U}]$ by grouping sparse modes and then we will efficiently approximate the exponential map on the individual summands using a multilinear interpolation. An alternative approach that uses neural networks to approximate the exponential maps and the direct sum structure will be introduced in Section 5.6

5.3.1 Submanifold Approximation

In many applications, the sparse modes (e.g. see Figure 5.5) can at least partially be split into subsets with pairwise decoupled support. For example, one expects distinct, localized articulations for certain joints and muscle groups on humanoid models. Hence, the supports of corresponding groups of sparse modes representing these articulations are expected to be well separated. Furthermore, as pointed out in Figure 5.7, the support of these tangent vectors in $T_{\bar{z}}\mathcal{M} \subset \mathbb{R}^{2|\mathbb{E}|}$ is only very moderately extended under application of the exponential $\text{Exp}_{\bar{z}}^K$. Given such a strong separation of supports of

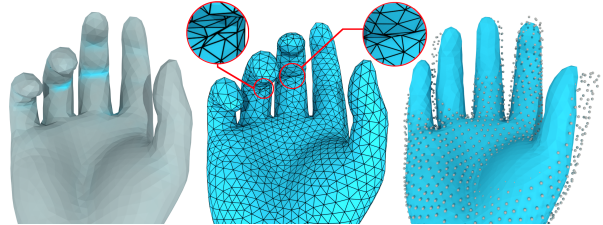


Figure 5.4: Comparison to sparse linear PCA with Euclidean metric in $\mathbb{R}^{2|\mathbb{E}|}$. The dihedral angle support of a sparse PCA mode which has no length component at all is shown on the left. This leads to mesh artifacts already in case of short time extrapolation as seen in the middle picture. Furthermore, this is prohibitive also for shape approximation, which can be seen in the last picture showing the failure to approximate an input shape easily handled even by the linear PCA in $\mathbb{R}^{2|\mathbb{E}|}$.

two subspaces \mathcal{U}_a and \mathcal{U}_b one observes that

$$\text{Exp}_{\bar{z}}^K(u_a + u_b) - \bar{z} \approx (\text{Exp}_{\bar{z}}^K(u_a) - \bar{z}) + (\text{Exp}_{\bar{z}}^K(u_b) - \bar{z})$$

for $u_a \in \mathcal{U}_a$ and $u_b \in \mathcal{U}_b$. This is certainly not the case for subspaces corresponding to modes with overlapping supports, which is illustrated in Figure 5.6 using the inset modes. The defect of a potentially non-exact separation of supports after applying the exponential map can be measured using

$$\mathcal{D}^{\text{Exp}}(u_a, u_b) := \frac{(|\text{Exp}_{\bar{z}}^K(u_a) - \bar{z}|, |\text{Exp}_{\bar{z}}^K(u_b) - \bar{z}|)_2}{\|\text{Exp}_{\bar{z}}^K(u_a) - \bar{z}\|_2 \|\text{Exp}_{\bar{z}}^K(u_b) - \bar{z}\|_2},$$

where $(\cdot, \cdot)_2$ is the scalar product associated with the weighted L^2 -norm introduced in Section 4.3.1, see also Figure 5.7. Before, we have already argued that the support of the tangent vectors is only moderately extended under the exponential map. The same then holds for the coupling of pairs of tangent vectors, which allows us to measure the separation of modes and induced submanifolds $\text{Exp}_{\bar{z}}^K(\mathcal{U}_a)$ and $\text{Exp}_{\bar{z}}^K(\mathcal{U}_b)$ using

$$\mathcal{D}(u_a, u_b) := \frac{(|u_a|, |u_b|)_2}{\|u_a\|_2 \|u_b\|_2},$$

which can also be seen when comparing Figure 5.8 and Figure 5.7.

This observation motivates the following splitting approach. Let us suppose that the space $\mathcal{U} \subset T_{\bar{z}}\mathcal{M}$ spanned by the sparse principal modes u_1, \dots, u_j can be written as the direct sum of L subspaces \mathcal{U}_l for $l = 1, \dots, L$, i.e.

$$\mathcal{U} = \mathcal{U}_1 \oplus \mathcal{U}_2 \oplus \dots \oplus \mathcal{U}_L,$$

where each of these subspaces is spanned by a subset of the principal modes. In fact, as discussed above, we suppose that due to our sparse PGA approach the subspaces \mathcal{U}_j can be chosen to have — at least approximately—pairwise disjoint supports. For each vector $v \in \mathcal{U}$, we have a corresponding decomposition $v = v_1 + v_2 + \dots + v_L$ with $v_l \in \mathcal{U}_l$. Now, we consider a superposition of exponential maps via summation of NRIC coordinates

$$\mathbf{Z}[v] = \bar{z} + \sum_{l=1}^L (z_l - \bar{z}) \quad \text{with } z_l = \text{Exp}_{\bar{z}}^K v_l.$$

Finally, $\mathbf{Z}[v]$ is not necessarily on \mathcal{M} and we have to project back onto \mathcal{M} and obtain

$$z[v] = \mathbf{P}_{\mathcal{M}} \mathbf{Z}[v].$$

This defines the submanifold

$$\mathcal{M}[\mathcal{U}_1, \dots, \mathcal{U}_L] = \{z[v] \mid v \in \mathcal{U}\} \subset \mathcal{M}$$

which approximates the original SPGA submanifold $\mathcal{M}[\mathcal{U}]$.

5.3.2 Efficient Deformation Synthesis

So far, we introduced an approximation of the submanifold by splitting the application of the exponential map into applications on smaller subspaces. It remains to show how these exponential

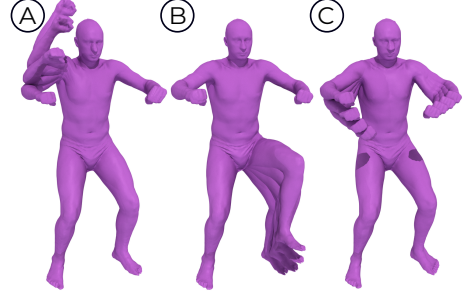


Figure 5.5: Three sparse modes A,B,C for the human model (used in Figure 5.6 and Figure 5.10).

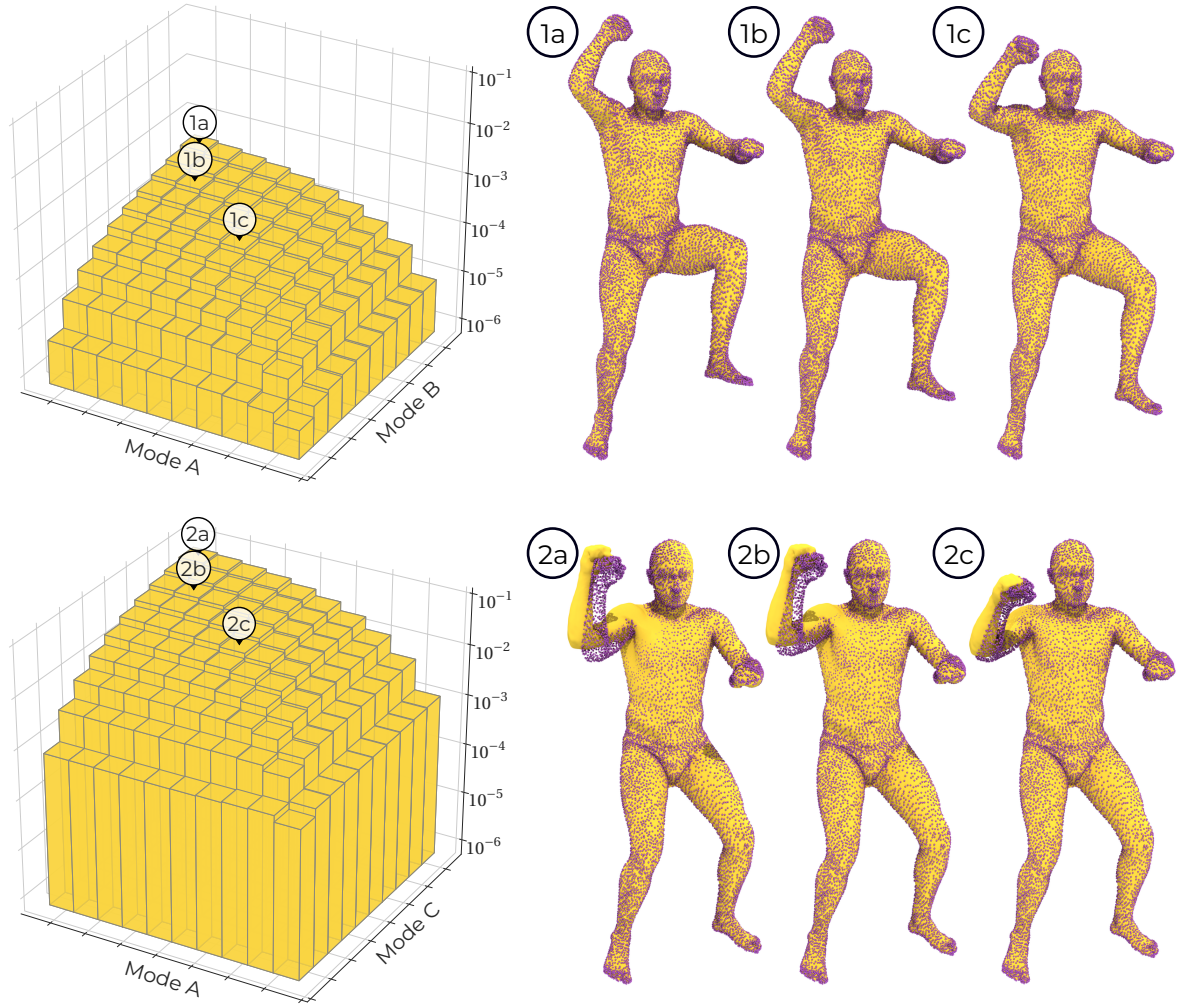


Figure 5.6: Comparison of splitting error for two separated modes (top row) and two overlapping modes (bottom) (cf. Figure 5.5). We consider a deformation synthesis $z[v]$, which treats the modes separately and show the relative approximation error in the energy, i.e. $\mathcal{W}[\text{Exp}_{\bar{z}}^K(v), z[v]] / \mathcal{W}[\bar{z}, \text{Exp}_{\bar{z}}^K(v)]$, on periodically distributed samples as bar plot along with three examples (approximation in yellow, correct extrapolation as purple point cloud).

maps and the needed projection can be approximated efficiently to enable the fast synthesis of non-linearly deformed shapes on the submanifold. Here, we propose an approach based on sampling in a preprocessing phase and multilinear interpolation in the online phase. Later, in Section 5.6, we will propose an alternative approach based on neural networks.

We first consider a single subspace \mathcal{U}_l of dimension d with basis u_1^l, \dots, u_d^l . In this subspace, we take a lattice $\{\sum_{n=1}^d \alpha_n \sigma u_n^l \mid \alpha \in \mathbb{Z}^d\}$ with mesh size $\sigma \in \mathbb{R}$. For $\alpha \in \mathbb{Z}^d$, we denote by $v^\alpha = \sum_{n=1}^d \alpha_n \sigma u_n^l$ the corresponding lattice point. Now, an arbitrary $v = \sum_{n=1}^d w_n(v) u_n^l \in \mathcal{U}_l$ lies in a cell of the lattice with nodes $v^{\alpha(v)+\beta}$ for $\beta \in \{0, 1\}^d$, where $\alpha(v) = \lfloor \frac{w(v)}{\sigma} \rfloor \in \mathbb{Z}^d$ elementwise. In two dimensions, this would mean that $\alpha(v)$ identifies the lattice point to the lower-left of v . To approximate $\text{Exp}_{\bar{z}}^K(v)$, we consider a piecewise multilinear interpolation $\mathbf{Z}_{\sigma, l}$ of the values at lattice points, i.e.

$$\mathbf{Z}_{\sigma, l}[v] := \sum_{\beta \in \{0, 1\}^d} w_\beta(v) \text{Exp}_{\bar{z}}^K(v^{\alpha(v)+\beta})$$

where the $w_\beta(v)$ are the multilinear interpolation weights (cf. [WZ88]) determined such that $v =$

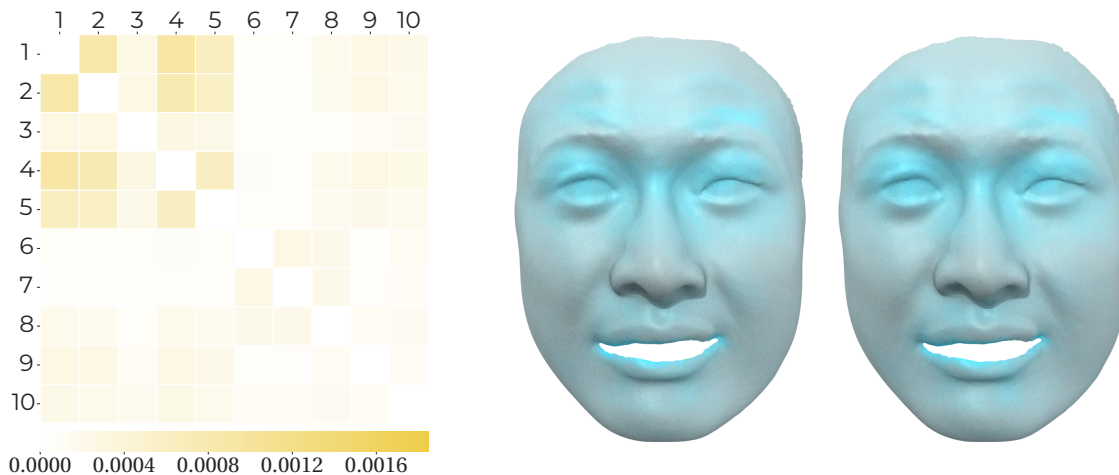


Figure 5.7: $\mathcal{D}^{\text{Exp}}(u_i, u_j)$ is plotted as heatmap on the left for $i, j \in \{1, \dots, J\}$. On the right, we show the extrapolation $\text{Exp}_{\bar{z}}^K(u)$ of a randomly sampled mode $u \in \mathcal{U}$, once colored according to the absolute tangential mode $|u|$ (left) and once colored according to the absolute difference $|\text{Exp}_{\bar{z}}^K(u) - \bar{z}|$ (right) both on the same color scale underlining the close similarity of the support of $u \in T_{\bar{z}}\mathcal{M} \subset \mathbb{R}^{2|\mathcal{E}|}$ and $\text{Exp}_{\bar{z}}^K(u) - \bar{z} \in \mathbb{R}^{2|\mathcal{E}|}$.

$\sum_{\beta \in \{0,1\}^d} w_\beta(v) v^{\alpha(v)+\beta}$. In the offline phase, exponentials of lattice points are computed for a suitable finite subset of \mathbb{Z}^d and stored in lookup tables, which can be used to cheaply evaluate the piecewise multilinear interpolation in the online phase.

Considering the whole subspace $\mathcal{U} = \mathcal{U}_1 \oplus \dots \oplus \mathcal{U}_L$, we use for general $v \in \mathcal{U}$ the splitting $v = v_1 + \dots + v_L$ and repeat the interpolation above for each subspace. Then, we evaluate the superposition of these interpolations

$$\mathbf{Z}_\sigma[v] = \bar{z} + \sum_{l=1}^L (\mathbf{Z}_{\sigma,l}[v_l] - \bar{z})$$

for the actual synthesis of nonlinear deformations. As above for $\mathbf{Z}[v]$ the superposition $\mathbf{Z}_\sigma[v]$ is in general not in \mathcal{M} and we define the submanifold $\mathcal{M}^\sigma[\mathcal{U}_1, \dots, \mathcal{U}_L] = \{z_\sigma[u] \mid u \in \mathcal{U}\}$ for

$$z_\sigma[u] = \mathbf{P}_{\mathcal{M}} \mathbf{Z}_\sigma[u]$$

which again is expected to approximate the submanifold $\mathcal{M}[\mathcal{U}]$. The necessity of the projection is demonstrated in Figure 5.9.

Surely, when dealing with the submanifolds $\text{Exp}_{\bar{z}}^K(\mathcal{U}_l)$ in our computational setup, we are interested only in compact subsets of the subspace \mathcal{U}_l representing plausible deformations. For the multilinear interpolation of precomputed samples on a rectangular grid to be a feasible option, we have to restrict to subspaces \mathcal{U}_l of low dimensionality between one and four. In many applications, it turned out that this is sufficient to achieve practicable results. Nevertheless, it is a limitation of the approach that we will address in Section 5.6.

A comparison of the exact submanifold $\mathcal{M}[\mathcal{U}_l] = \text{Exp}_{\bar{z}}^K(\mathcal{U}_l)$ and the approximation $\mathcal{M}^\sigma[\mathcal{U}_l]$ for a single two dimensional subspace \mathcal{U}_l is shown in Figure 5.10. The efficiency of this approach based on multilinear interpolation of precomputed samples is evaluated in the next section and in particular in Table 5.4.

5.4 Numerical Implementation

Having described the variational problems underlying our approach, we turn to their numerical solution in this section. The approach to the variational problems in NRIC and nodal positions

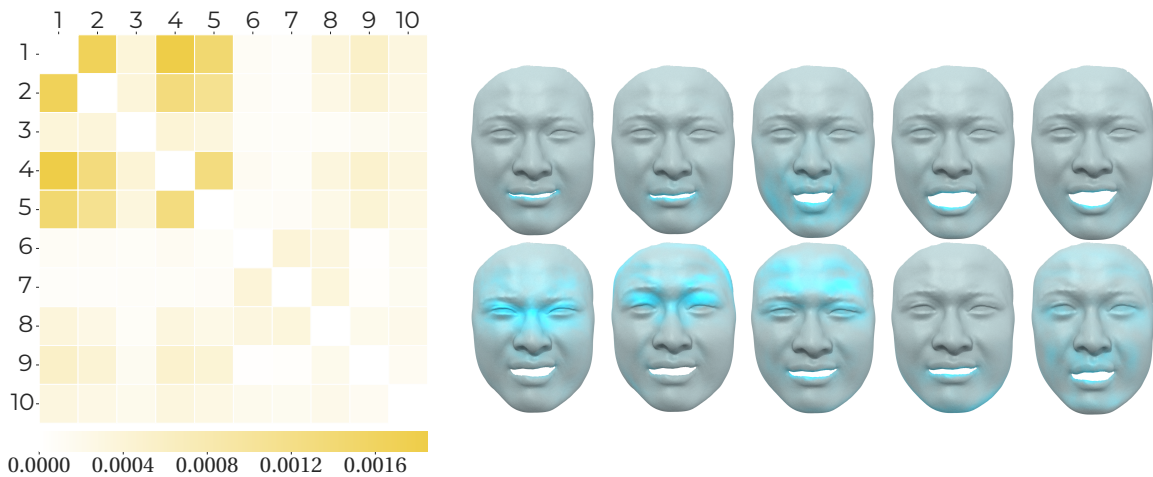


Figure 5.8: Coupling of tangential modes on a face data set [ZSCS04] exhibiting non-isometric deformations. The heatmap shows the coupling between the different tangential modes, i.e. $\mathcal{D}(u_i, u_j)$ for $i, j \in \{1, \dots, J\}$. It is accompanied by the extrapolated modes colored according to the absolute tangential modes $|u_i|$ (weighted as in the L^1 -norm), each on a different scale to improve the visual appearance of the modes.

resulting from time-discrete geodesic calculus corresponds to the approaches from Chapter 3 and Chapter 4 and we only briefly describe a multi-resolution scheme to make these methods more efficient below. What remains is the solution of the SPGA problem (5.6) on tangential modes, for which we will introduce an approach based on quadratic matrix programming.

Multi-Resolution Scheme. We use a multi-resolution approach to enable fast computations while retaining the possibility to produce high-quality deformations. In this approach, the input shapes are coarsened by one of the following two methods: For some examples, we used an iterative edge collapse approach based on minimizing the quadric error metric [GH97] computed in groups [MG03] to preserve the dense correspondence between input shapes. For other examples, we remeshed a reference input shape using OpenFlipper [MK12] and computed coarse representations of the other input shapes with the same approach as in the prolongation described next. In both cases, the coarse results were prolonged to the fine level using a representation of the fine mesh vertices in terms of intrinsic positions and normal displacement with respect to the coarse mesh, which is computed on a reference shape similar to [KMP07]. See Table 5.4 for a comparison of original and coarse resolutions of the different input data.

Software. We have implemented our method in C++ with the Geometric Optimization And Simulation Toolbox (GOAST) [HS+20], where we use the Eigen library [GJ+10] for numerical linear algebra and CHOLMOD [CDHR08] and UMFPACK [Dav04] from the SuiteSparse collection as direct linear solvers. For the quadratic problem (5.19), we use MOSEK [MOS20] as efficient off-the-shelf solver.

5.4.1 Quadratic Matrix Programming

We solve (5.6) by alternatingly solving for U while keeping W fixed and vice versa. When U is fixed, solving for W becomes a straightforward quadratic optimization problem. However, solving for U is more difficult due to the regularization term and the tangent space constraint.

To solve for the sparse modes efficiently, we will first rephrase (5.6) for fixed weights W as a nonlinear matrix optimization problem. Then, we will vectorize it to obtain a nonlinear optimization

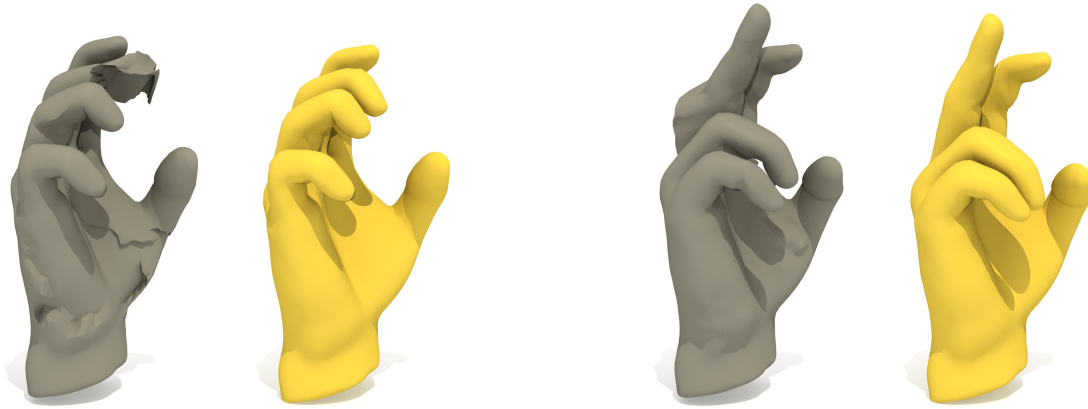


Figure 5.9: Two comparisons of projected shapes $z_\sigma[v]$ (yellow) with an adaptive frame-based reconstruction from [SHHR20] of $Z_\sigma[v]$ without projecting it on the manifold (gray). This shows that the projection is required.

problem in a more common formulation. Finally, we will relax the L^1 -penalty term as proposed by Tibshirani [Tib96] and Brandt and Hildebrandt [BH17] to obtain a quadratic problem. This mirrors an approach commonly used for *quadratic matrix programming* (QMP) problems [Bec07] and we will also briefly show an alternative approach that first relaxes the L^1 -penalty in (5.6) to obtain a QMP problem.

Matrix optimization. Let us begin by reformulating the matrix optimization problem to prepare the relaxation. We denote by $G \in \mathbb{R}^{2|\mathbb{E}| \times 2|\mathbb{E}|}$ the matrix representation of the metric $g_{\bar{z}}$, i.e. $G = \partial_{\bar{z}}^2 \mathcal{W}[\bar{z}, \bar{z}]$. By the definition of the (weighted) Frobenius norm, we have

$$\begin{aligned} \|V - UW\|_g^2 &= \text{tr}((V - UW)^T G (V - UW)) \\ &= \text{tr}(W^T U^T G U W) - 2 \text{tr}(V^T G U W) + \text{tr}(V^T G V) \\ &= \text{tr}(W^T U^T G U W) - 2 \text{tr}(W V^T G U) + \text{tr}(V^T G V). \end{aligned}$$

To rephrase the regularization term, we denote by $M \in \mathbb{R}^{J \times 2|\mathbb{E}|}$ the matrix whose rows are all equal to the weight vector of the L^1 -norm from Section 4.3.1. This allows the following reformulation

$$\sum_{j=1}^J \|u_j\|_1 = \text{tr}(M^T |U|), \quad |U| := (|u_{kj}|)_{\substack{k=1, \dots, 2|\mathbb{E}| \\ j=1, \dots, J}}.$$

We express the tangent space constraint using the quaternion integrability operator \mathcal{Q} , i.e.

$$D_z \mathcal{Q}(\bar{z}) U = 0 \in \mathbb{R}^{3|\mathbb{V}_0| \times J}$$

Together, this yields for a given W the problem

$$\begin{aligned} \underset{U \in \mathbb{R}^{2|\mathbb{E}| \times J}}{\text{minimize}} \quad & \text{tr}(W^T U^T G U W) - 2 \text{tr}((G V W^T)^T U) + \lambda \text{tr}(M^T |U|) + \text{tr}(V^T G V) \\ \text{subject to} \quad & D_z \mathcal{Q}(\bar{z}) U = 0. \end{aligned} \tag{5.7}$$

While this problem already closely resembles a QMP problem, the choice of the L^1 -term still renders it nonlinear.

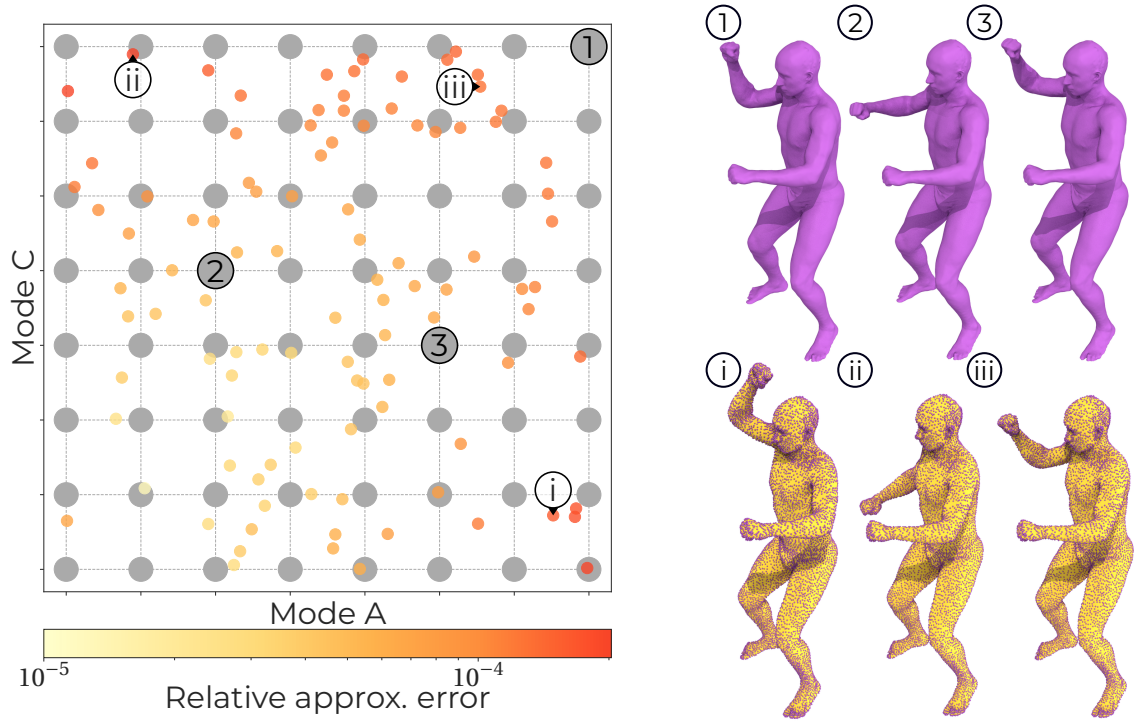


Figure 5.10: Approximation of the exponential map by multilinear interpolation of lattice points for two strongly coupled modes (cf. Figure 5.5). On the left, we show the lattice (grey) in the two-dimensional subspace along with randomly sampled points colored according to the logarithm of their relative approximation error in the energy, i.e. $\mathcal{W}[\text{Exp}_{\bar{z}}^K(v), z_\sigma[v]]/\mathcal{W}[\bar{z}, \text{Exp}_{\bar{z}}^K(v)]$. On the right, selected extrapolated lattice points are shown in purple and we highlight the quality of the approximation of shapes (purple point clouds) by our deformation synthesis approach (yellow) for samples with relatively high approximation error.

Vectorization. Next, we vectorize the problem, i.e. we replace the matrix as primary degree of freedom by its vectorization. By vectorization, we refer to the map

$$\begin{aligned} \text{vec}: \mathbb{R}^{n \times m} &\rightarrow \mathbb{R}^{nm} \\ A &\mapsto (a_{1,1}, \dots, a_{n,1}, a_{1,2}, \dots, a_{n,2}, \dots, a_{n,m})^T, \end{aligned}$$

i.e. concatenating the columns of a matrix to a single vector, which is an isomorphism. Hence, we aim at an optimization problem in $\mathbb{R}^{2|E|J}$ such that its solution are exactly the vectorizations of solutions of (5.7). To this end, we reformulate all terms appearing in (5.7) in the vectorization of U . The essential formulas can be found in [Bec07] and we only apply them to our problem here.

For the quadratic term, we obtain

$$\text{tr}(W^T U^T M U W) = \text{vec}(U W)^T (\text{Id}_N \otimes M) \text{vec}(U W) \quad (5.8)$$

$$= \text{vec}(U)^T (W^T \otimes \text{Id}_{2|E|})^T (\text{Id}_N \otimes M) (W^T \otimes \text{Id}_{2|E|}) \text{vec}(U) \quad (5.9)$$

$$= \text{vec}(U)^T (W W^T \otimes M) \text{vec}(U), \quad (5.10)$$

while we get for the linear term

$$\text{tr}(W V^T M U) = \text{vec}(M V W^T)^T \text{vec}(U). \quad (5.11)$$

Now, for the nonlinear term note that taking absolute values of the entries of a matrix/vector and vectorization commute, i.e. $\text{vec}(|A|) = |\text{vec}(A)|$. This yields

$$\text{tr}(M^T |U|) = \text{vec}(M)^T |\text{vec}(U)|. \quad (5.12)$$

Finally, we use the following equivalence for the constraint

$$D_z \mathcal{Q}(\bar{z}) \cdot U = 0 \in \mathbb{R}^{3|V| \times J} \iff (\text{Id}_J \otimes D_z \mathcal{Q}(\bar{z})) \text{vec}(U) = 0 \in \mathbb{R}^{3|V|J}. \quad (5.13)$$

Combining all this, we arrive at the following vector optimization problem where we use a boldface \mathbf{u} to denote that we refer to a vectorization and not a single basis vector,

$$\begin{aligned} & \underset{\mathbf{u} \in \mathbb{R}^{2|E|J}}{\text{minimize}} && \mathbf{u}^T (W W^T \otimes M) \mathbf{u} - 2 \text{vec}(M V W^T)^T \mathbf{u} + \lambda \text{vec}(M)^T |\mathbf{u}| + \text{tr}(V^T M V) \\ & \text{subject to} && (\text{Id}_J \otimes D_z \mathcal{Q}(\bar{z})) \mathbf{u} = 0. \end{aligned} \quad (5.14)$$

Relaxation. Finally, we relax the L^1 -penalty term as in [BH17; Tib96] to obtain the quadratic problem. To this end, we introduce non-negative variables, i.e.

$$\mathbf{u} = \mathbf{u}^+ - \mathbf{u}^-, \quad \mathbf{u}^+, \mathbf{u}^- \geq 0, \quad (5.15)$$

and define the combined variable

$$\tilde{\mathbf{u}} = \begin{pmatrix} \mathbf{u}^+ \\ \mathbf{u}^- \end{pmatrix} \in \mathbb{R}^{4|E|J}. \quad (5.16)$$

Then, by the componentwise triangle inequality, we obtain the following linear upper bound for the L^1 -term

$$\text{vec}(M)^T |\mathbf{u}| \leq \text{vec}(M)^T (\mathbf{u}^+ + \mathbf{u}^-) = \left(\begin{pmatrix} 1 \\ 1 \end{pmatrix} \otimes \text{vec}(M) \right)^T \tilde{\mathbf{u}} \quad (5.17)$$

Reformulating the complete vectorized problem (5.14) in $\tilde{\mathbf{u}}$ leads to

$$\begin{aligned} & \underset{\tilde{\mathbf{u}} \in \mathbb{R}^{4|E|J}}{\text{minimize}} && \tilde{\mathbf{u}}^T (E \otimes W W^T \otimes M) \tilde{\mathbf{u}} + \left(\begin{pmatrix} 1 \\ 1 \end{pmatrix} \otimes \text{vec}(M) - \begin{pmatrix} 2 \\ -2 \end{pmatrix} \otimes \text{vec}(M V W^T) \right)^T \tilde{\mathbf{u}} + \text{tr}(V^T M V) \\ & \text{subject to} && \begin{pmatrix} 1 & -1 \end{pmatrix} \otimes (\text{Id}_J \otimes D_z \mathcal{Q}(\bar{z})) \tilde{\mathbf{u}} = 0 \quad \text{and} \quad \tilde{\mathbf{u}} \geq 0 \end{aligned} \quad (5.18)$$

with $E := \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}$. The above triangle inequality is an equality, i.e. $|\mathbf{u}| = \mathbf{u}^+ + \mathbf{u}^-$, if $\mathbf{u}^+ \odot \mathbf{u}^- = 0$.

Hence, the solutions $\mathbf{u}^+, \mathbf{u}^-$ of (5.18) have disjoint support because, otherwise, we could move values from one vector to the other without changing the value of the objective term while decreasing the regularization term. Thus, $\mathbf{u} = \mathbf{u}^+ - \mathbf{u}^-$ is indeed the solution of the original problem. To solve the relaxed problem, one can utilize standard numerical software for quadratic programming.

Relaxing the matrix problem. Alternatively, we could have switched the order of vectorization and relaxation and, thus, relaxed the L^1 -penalty in the matrix formulation. To this end, we introduce non-negative variables $U^+, U^- \in \mathbb{R}^{2|E| \times J}$ with

$$U = U^+ - U^-, \quad U^+, U^- \geq 0,$$

This way, using the componentwise triangle inequality $|U| \leq U^+ + U^-$ we obtain the estimate

$$\text{tr}(M^T |U|) \leq \text{tr}(M^T (U^+ + U^-))$$

for the L^1 -term. Analogously, adapting the other terms in (5.7) we finally arrive at the relaxed quadratic matrix programming problem

$$\begin{aligned} & \underset{\substack{U^+ \in \mathbb{R}^{2|E| \times J} \\ U^- \in \mathbb{R}^{2|E| \times J}}}{\text{minimize}} && \text{tr} \left(W^T \begin{pmatrix} U^+ \\ U^- \end{pmatrix}^T \begin{pmatrix} G & -G \\ -G & G \end{pmatrix} \begin{pmatrix} U^+ \\ U^- \end{pmatrix} W \right) + \text{tr} \left(\begin{pmatrix} \lambda M + 2G V W^T \\ \lambda M - 2G V W^T \end{pmatrix}^T \begin{pmatrix} U^+ \\ U^- \end{pmatrix} \right) + \text{tr}(V^T G V) \\ & \text{subject to} && D_z \mathcal{Q}(\bar{z}) U^+ - D_z \mathcal{Q}(\bar{z}) U^- = 0, \quad U^+ \geq 0, \quad U^- \geq 0. \end{aligned} \quad (5.19)$$

By vectorizing this QMP problem, we again arrive at (5.18).

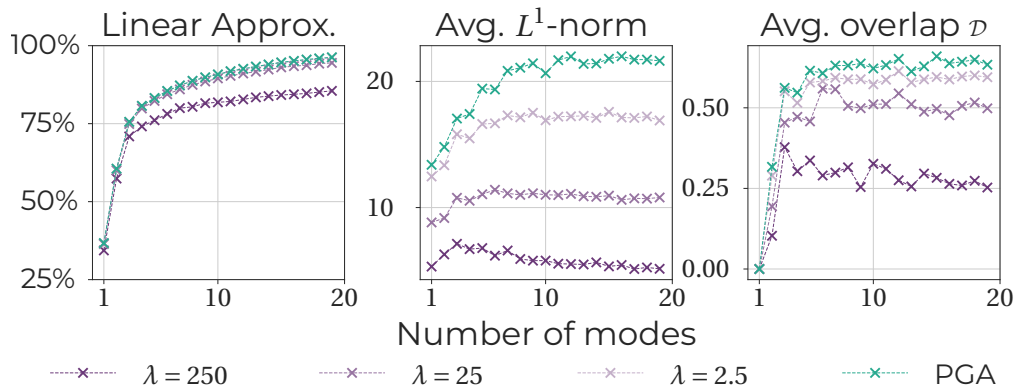


Figure 5.11: Comparison of multiple choices of J and λ on the Face dataset. The linear approximation quality is measured as $\|V - UW\|^2 / \|V\|^2$ and weighted L^1 -norm and overlap are computed as described before. Noticeably larger values for λ led to vanishing modes. In this case, we chose $J = 10$ and $\lambda = 250$.

5.5 Experiments and Applications

After introducing our submanifold construction and the numerical tools necessary for it, we will now consider a range of experiments to illuminate its practical properties. Furthermore, we will discuss potential applications in geometric modeling.

Datasets. We evaluated our approach on four different datasets. The first dataset consists of 71 humanoid shapes from Angelov *et al.* [ASK+05] called SCAPE. For the second dataset, we considered six hand shapes from Yeh *et al.* [YLSL11] and upsampled them by computing 126 weighted elastic averages of these six shapes. Furthermore, we took a set of 384 face meshes from Zhang *et al.* [ZSCS04] originating from a 3D-scanned time series of a person expressing different emotions. Finally, we also considered a set of 50 meshes of a horse in a galloping sequence from Sumner and Popović [SP04]. The corresponding mesh sizes of the input data and coarsened versions are listed in Table 5.4.

Selection of number of modes and sparsity. In the computation of modes, we have two parameters which are currently chosen manually based on heuristics: The number of modes J and the sparsity weight λ in (5.6). This is linked to a trade-off between the approximation quality of the model, the sparsity and overlap of support of the modes, and the size of the model. In Figure 5.11, we compare these quantitatively for a number of choices and also to PGA. For the approximation, not only the average overlap is important but also the resulting subspace dimensions because the grid-based interpolation is only feasible for up to four dimensions. Thus, a higher number of modes might lead to subspace violating this constraint and is thus not necessarily beneficial.

Selection of coupling spaces. The selection of the subspace decomposition $\mathcal{U} = \mathcal{U}_1 \oplus \dots \oplus \mathcal{U}_L$ —i.e. the grouping of the sparse tangential modes based on their coupling—is an important step to obtaining high-quality results with our proposed method as exemplarily pointed out in Figure 5.6. In our implementation, we employ spectral clustering [SM00] of a set of modes using the sparse tangential coupling $\mathcal{D}(\cdot, \cdot)$ as the underlying similarity measure. To this end, we use the coupling matrix of the sparse modes, cf. Figure 5.8, as affinity or similarity matrix. The number of clusters is then determined such that the resulting dimensionality of the subspaces \mathcal{U}_l is in the range 1–4, see also Figure 5.12, and thus small enough for the lattice generation described in Section 5.3.2 to be computationally feasible. The decomposition into subspaces of different dimensions used in our

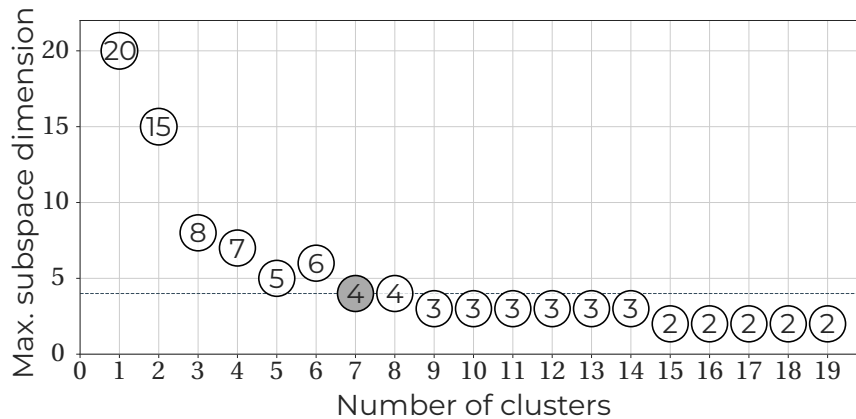


Figure 5.12: Comparison of maximal subspace dimension for different numbers of clusters on the Horse dataset. The selected number highlighted in gray.

examples is listed in Table 5.1. Examples of groups of modes extracted from the SCAPE data set are already shown in Figure 5.1. The examples illustrate that modes that move the same part of the body, for example the left leg in the leftmost groups, can be in different groups. The reason is that the distortions induced by the modes are located in different areas of the body, for example, in the hip and knee regions for the leftmost groups.

Example	J	1D	2D	3D	4D
SCAPE	40	0	5	6	3
Hands	12	0	1	2	1
Faces	10	4	1	0	1
Horse	20	0	2	4	1

Table 5.1: Distribution of the dimensionality of subspaces \mathcal{U}_i used in the different examples.

Comparison to ADMM. Another method commonly used to compute sparse deformation components is the alternating direction method of multipliers (ADMM), cf. [NVW+13; HYZ+14; WLZH17]. We have also investigated this approach to solve for the sparse modes. To deal with the tangent space constraint in (5.6), one needs to add a third term to the commonly used ADMM problem representing this constraint as a convex indicator function. To see this, consider the general optimization problem

$$\begin{aligned} & \underset{\mathbf{u} \in \mathbb{R}^{2|\mathbb{E}|J}}{\text{minimize}} && \|\mathbf{W}\mathbf{u} - \mathbf{v}\|_{\mathbf{G}}^2 + \lambda \|\mathbf{u}\|_1 \\ & \text{subject to} && \mathbf{A}\mathbf{u} = \mathbf{0}, \end{aligned} \quad (5.20)$$

where in our case $\mathbf{u} \in \mathbb{R}^{2|\mathbb{E}|J}$ are the vectorized tangent modes, $\mathbf{v} = \text{vec}(V) \in \mathbb{R}^{2|\mathbb{E}|N}$ the vectorized logarithms, $\mathbf{W} = \text{Id}_{2|\mathbb{E}|} \otimes W \in \mathbb{R}^{2|\mathbb{E}|N \times 2|\mathbb{E}|J}$ the fixed weights as applied to vectorized modes, $\|\cdot\|_1$ the weighted L^1 -norm, $\mathbf{G} = \text{Id}_N \otimes G \in \mathbb{R}^{2|\mathbb{E}|N \times 2|\mathbb{E}|N}$ the vectorized inner product, and $\mathbf{A} = \text{Id}_J \otimes D_z \mathcal{Q}(\bar{z}) \in \mathbb{R}^{3|V|J \times 2|\mathbb{E}|J}$ the vectorized tangent space constraints. Then we rewrite it as

$$\begin{aligned} & \underset{\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathbb{R}^{2|\mathbb{E}|J}}{\text{minimize}} && \|\mathbf{W}\mathbf{x} - \mathbf{v}\|_{\mathbf{G}}^2 + \iota_{\mathbf{A}}(\mathbf{y}) + \lambda \|\mathbf{z}\|_1 \\ & \text{subject to} && \mathbf{x} - \mathbf{z} = \mathbf{y} - \mathbf{z} = \mathbf{0}, \end{aligned} \quad (5.21)$$

where $\iota_{\mathbf{A}}(\mathbf{y}) = 0$ if $\mathbf{A}\mathbf{y} = \mathbf{0}$ and $\iota_{\mathbf{A}}(\mathbf{y}) = \infty$ else. Then—as usual for ADMM—one alternately evaluates the proximal mappings for each term, i.e. in each iteration k of the algorithm the following updates

are computed:

$$\begin{aligned}\mathbf{x}^{k+1} &= \arg \min_{\mathbf{x} \in \mathbb{R}^{2|E|}} \|\mathbf{W}\mathbf{x} - \mathbf{v}\|_{\mathbf{G}}^2 + \frac{\eta}{2} \|\mathbf{x} - \mathbf{z}^k + \gamma_{\mathbf{x}}^k\|_2^2, \\ \mathbf{y}^{k+1} &= \arg \min_{\mathbf{y} \in \mathbb{R}^{2|E|}} \iota_{\mathbf{A}}(\mathbf{y}) + \frac{\eta}{2} \|\mathbf{y} - \mathbf{z}^k + \gamma_{\mathbf{y}}^k\|_2^2, \\ \mathbf{z}^{k+1} &= \arg \min_{\mathbf{z} \in \mathbb{R}^{2|E|}} \lambda \|\mathbf{z}\|_1 + \frac{\eta}{2} \|\mathbf{z} - \frac{1}{2}(\mathbf{x}^{k+1} + \mathbf{y}^{k+1}) - \frac{1}{2}(\gamma_{\mathbf{x}}^k + \gamma_{\mathbf{y}}^k)\|_2^2, \\ \gamma_{\mathbf{x}}^{k+1} &= \gamma_{\mathbf{x}}^k + \mathbf{x}^{k+1} - \mathbf{z}^{k+1}, \\ \gamma_{\mathbf{y}}^{k+1} &= \gamma_{\mathbf{y}}^k + \mathbf{y}^{k+1} - \mathbf{z}^{k+1},\end{aligned}$$

where $\gamma_{\mathbf{x}}^{k+1}$ and $\gamma_{\mathbf{y}}^{k+1}$ are estimates of the Lagrange multipliers for the constraints $\mathbf{x} - \mathbf{z} = 0$ and $\mathbf{y} - \mathbf{z} = 0$, respectively, and $\eta > 0$ is a stepsize parameter. The three minimization problems for \mathbf{x}^{k+1} , \mathbf{y}^{k+1} , and \mathbf{z}^{k+1} have a closed form solution. This approach can be seen as an application of the Augmented Lagrangian method to (5.21) and a more detailed explanation of this can, for example, be found in [BPC+11]. However, we observed a slower convergence of this approach, i.e. compared to our approach outlined in Section 5.4.1 the objective value was larger after the same number of outer iterations while still requiring more time per outer iteration. Let us note that this picture might change for larger problems. We provide objective values and runtimes for the two methods on our examples in Table 5.2.

	SCAPE		Hands		Faces	
Solver	Obj.	Time	Obj.	Time	Obj.	Time
QMP	2618	1553s	575	205s	28765	127s
ADMM	2755	2092s	629	382s	31114	408s

Table 5.2: Objective values after ten outer iterations and runtime per outer iteration for QMP and ADMM. For ADMM, we choose the penalty parameters as $\eta = 10$ after comparing multiple options.

Comparison PGA to SPGA. In comparison to PGA, our SPGA computes sparse modes at the expense of approximation accurateness. Hence, to achieve the same approximation quality with SPGA as with PGA one will typically need more modes (cf. Figure 5.11). The sparsity, however, is crucial for the submanifold approximation in Section 5.3.1, as otherwise the resulting error would be large (cf. Figure 5.6). When working with PGA, we cannot use our efficient deformation synthesis and instead have to directly use the nonlinear exponential map. In editing applications, when additional handle constraints come into play, this implies that we have to evaluate derivatives of the discrete exponential and hence use methods from PDE-constrained optimization, see [HZRS18]. This results in online runtimes which are far from interactive rates (Table 5.3). In contrast, our approximation built on SPGA allows for interactive rates but requires more runtime in the offline phase for the quadratic optimization and sampling of the exponential map.

Deformation synthesis. A central aim of our method is the fast synthesis of high-quality and large nonlinear deformations in articulated motion, i.e. the evaluation of our discrete approximation z_{σ} of the exponential map for varying subspace coordinates $\alpha \in \mathbb{R}^J$. We demonstrate this by considering curves $C: \mathbb{R} \rightarrow \mathcal{U}$ in the subspace and their counterpart $c: \mathbb{R} \rightarrow \mathcal{M}^{\sigma}$ on the manifold \mathcal{M} obtained via our deformation synthesis, i.e. $c(t) := z_{\sigma}[C(t)]$. Such curves can, for example, be used to obtain smooth deformations interpolating given key poses on the manifold. In Figure 5.13, we show such an interpolating curve based on a Catmull–Rom spline $t \mapsto C(t)$ [CR74], for a set of hand poses. Recall, that we follow Fröhlich and Botsch [FB11] to compute the projection onto the manifold and

Example	PGA		SPGA	
	Off-	Online	Off-	Online
Figure 5.13 (Interpolation)	3min	3s	90min	60ms
Figure 5.14 (Fitting)	2min	919s*	660min	10s (100ms)

Table 5.3: Comparison of runtimes for PGA and SPGA. In the interpolation case, we report the average time to evaluate the spline in Figure 5.13 at 120 evenly spaced points. In the fitting case, the total amount of time to produce the results in Figure 5.14 is reported and for SPGA, in brackets, also the time for incremental editing steps. * as reported in [HZRS18]

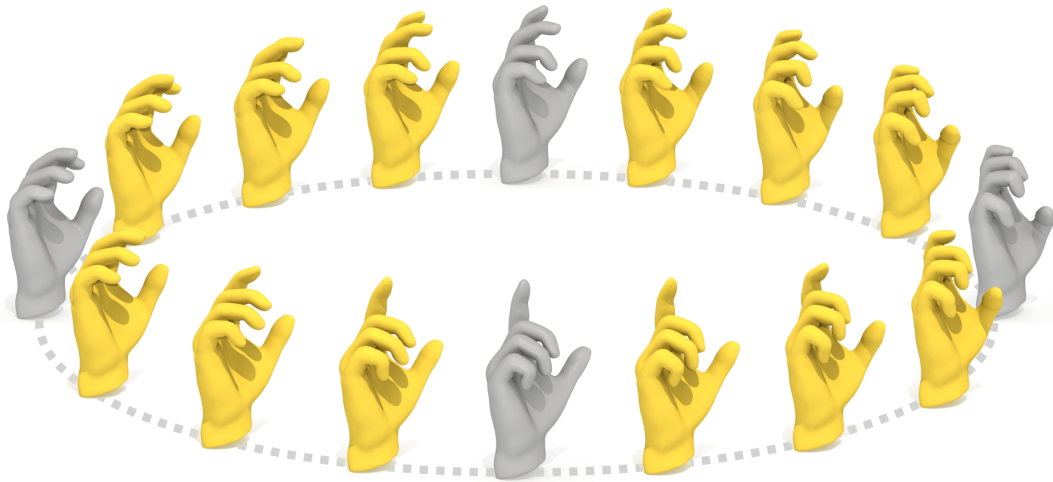


Figure 5.13: An interpolating deformation path computed using our method. The interpolation is obtained by fitting a periodic Catmull–Rom spline through the subspace coordinates of the input shapes (shown in grey) and evaluating our discrete parametrization for evenly spaced points on the spline (yellow).

formulate it as nonlinear least squares problem in nodal positions, i.e. $\min_{X \in \mathbb{R}^{3|V|}} \|(l, \theta)(X) - \mathbf{Z}_\sigma[u]\|_g^2$. This directly gives us the nodal positions needed for visualizations and can be solved efficiently using the Gauß–Newton method. Then, evaluating the curve $t \mapsto z_\sigma[C(t)]$ along consecutive points can be done in (near) realtime in these examples as each step only needs very few Gauß–Newton iterations to compute the projection \mathbf{P}_M due to the good initialization from the previous step. See also the paragraph *Timings* below.

Mesh editing. The deformation synthesis restricted to the submanifold $\mathcal{M}^\sigma[\mathcal{U}_1, \dots, \mathcal{U}_L]$ can also be used as a so-called deformation prior for mesh editing—i.e. in a regularization term encouraging reasonable deformations. In this case, we assume the editing information is given as M sparse handle positions $x_1, \dots, x_M \in \mathbb{R}^3$ corresponding to the positions of vertices $v_1, \dots, v_M \in \mathbf{V}$. Then we perform the editing via a quadratic penalty method, i.e. we consider a series of optimization problems of the form

$$\underset{\substack{X \in \mathbb{R}^{3|V|} \\ u \in \mathcal{U}}}{\text{minimize}} \quad \|(l, \theta)(X) - \mathbf{Z}_\sigma[u]\|_g^2 + \gamma \sum_{m=1}^M \|X_{v_m} - x_m\|_{\mathbb{R}^3}^2 \quad (5.22)$$

for increasing γ , each again solved with Gauß–Newton. We choose the starting value for γ and increase it by a fixed factor until the mesh fulfills the handle positions up to a pre-defined tolerance.

In Figure 5.14, we compare this mesh editing approach to other state-of-the-art, data-driven approaches. To compare our approach to [LLW+19], we replicated their approach without localization,

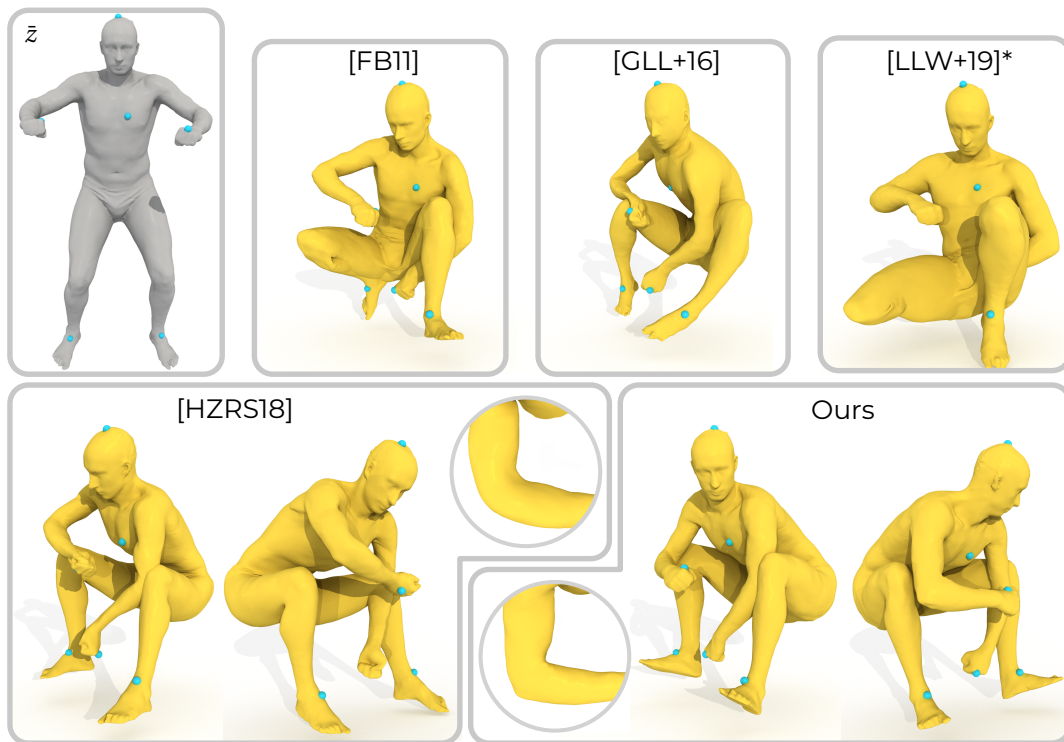


Figure 5.14: Handle editing comparison on SCAPE. Top row: Rest pose, results using the methods in [FB11], [GLL+16], and an adapted version of [LLW+19] (see main text for more information); Bottom row: results from [HZRS18] and ours along with close ups of the right elbows.

i.e. using a spatially constant l_1 -term in the mode computation, and with a fixed penalty parameter (see the right-most example in the top row). The other two methods, [FB11] and [GLL+16], are applied as described in the original publications. Our method delivers plausible deformations on-par with [HZRS18], while being nearly two orders of magnitude faster. Some details are even superior, for example, the bending of the right arm, shown in the close-ups, appears more natural in our results with the sharp elbow contour being better preserved.

Using the superposition of nonlinear subspaces represented by \mathbf{z}_σ instead of the linear combinations of modes in $\mathbb{R}^{2|E|}$ allows our method to generalize better for edits requiring deformations far from the input data set. For example, in Figure 5.15, we consider the challenging example of dragging the right hind leg outwards of a horse shape, while the input data only consists of a galloping sequence. Here, besides the penalty approach used in Figure 5.14, we used hard constraints for the vertex positions at the handles. We compare the editing via a linear basis and hard constraints in $\mathbb{R}^{2|E|}$ (left) with editing based on the large deformation interpolation $z_\sigma[\cdot]$, both for hard constraints (middle), and using the quadratic penalty method (5.22) (right). The linear approach leads to unnatural bending similar to the one reported by Liu *et al.* [LLW+19, Figure 17], where the latter two approaches yield far more natural results.

Timings. In Table 5.4, we list detailed timings of all components of our method. For the quadratic problem, we report runtimes with enabled parallelization, while it was disabled for all other timings. The offline phase was performed on a workstation with two AMD EPYC 7601 CPUs, the high number of cores allows to efficiently parallelize the offline phase by computing many logarithms or exponential maps in parallel. All other results were computed on a laptop with an Intel Core i7-9750H CPU.

In the preprocessing phase, we used 25 outer iterations, i.e. solving once for the weights and modes, in each of the examples. Precomputing the exponential for lattice points (cf. Section 5.3.2)

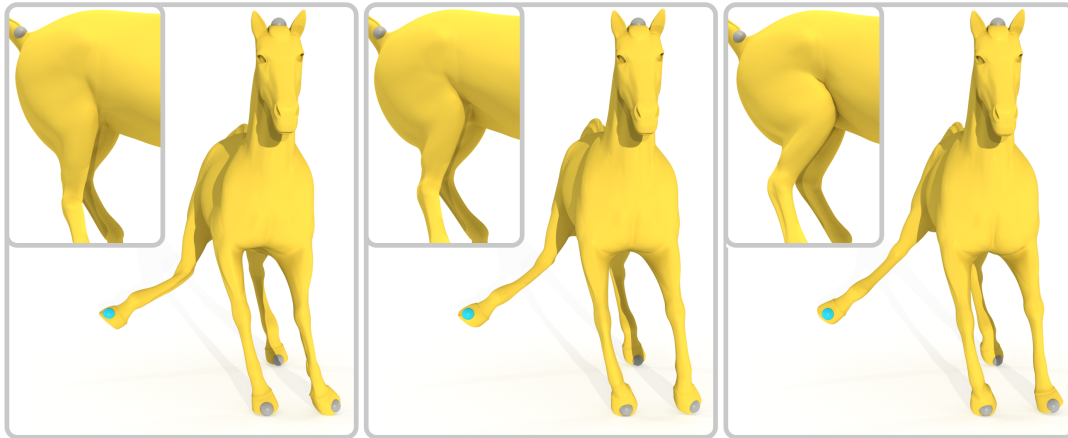


Figure 5.15: Challenging handle editing example on the horse as suggested in [LLW+19]. Left to right: editing with linear interpolation in $\mathbb{R}^{2|E|}$ and hard constraints; editing with nonlinear interpolation and hard constraints; editing with nonlinear interpolation and soft constraints, i.e. via the quadratic penalty method (5.22).

Data & Parameters					Preprocessing				Online		
Dataset	N	$ V $	$ V_c $	J	Mean	Log	QMP	Exp	\mathbf{Z}_σ -eval.	GN iter (proj)	GN iter (edit)
SCAPE	71	12.5k	1.3k	40	105s	30s	1553s	3s	0.5ms	18ms	100ms
Hands	126*	6.1k	1.9k	12	230s	90s	205s	6s	0.3ms	25ms	–
Faces	384	24k	2.1k	10	420s	70s	127s	6s	0.3ms	35ms	–
Horse	50	8.5k	1.3k	20	100s	50s	375s	4s	0.3ms	16ms	60ms

Table 5.4: Overview of used datasets with corresponding preprocessing and online timings listing from left to right: the data set, the number of input shapes N , the number of vertices of the full mesh $|V|$ and the coarse mesh $|V_c|$, the number of modes J , the timings in the preprocessing phase to compute the mean \bar{z} , a discrete logarithm $\text{Log}_{\bar{z}}$, one solution of the quadratic matrix programming problem, a discrete exponential map $\text{Exp}_{\bar{z}}^K$, and the timings in the online phase to compute the multilinear interpolations \mathbf{Z}_σ , a Gauß–Newton iteration used for the projection on the manifold \mathcal{M} , and a Gauß–Newton iteration required in the mesh editing context.

* obtained from six shapes by computing weighted elastic averages

required about 3000 (horse), 4000 (hands), 20000 (human), and 30000 (face) evaluations for the respective examples.

In the online phase, evaluating our deformation synthesis z_σ for a random $v \in \mathcal{U}$ using the mean shape as initialization needed in all examples at most ten Gauß–Newton iterations. For larger, instant changes, such as in Figures 5.14 and 5.15, we need multiple increases of the penalty parameter in (5.22) and thus a larger number of iterations. Concretely, the computation for the result in Figure 5.14 took about ten seconds while for Figure 5.15 it took about five seconds. In applications with incremental changes, we only need a very small number of Gauß–Newton iterations. For example, each step of a spline for the shapes in Figure 5.13 evaluated at 120 evenly spaced points required two to three iterations, while for the incremental handle editing (shown in Figures 5.1 and 1.2) even one was sufficient. This makes the method applicable for interactive applications.

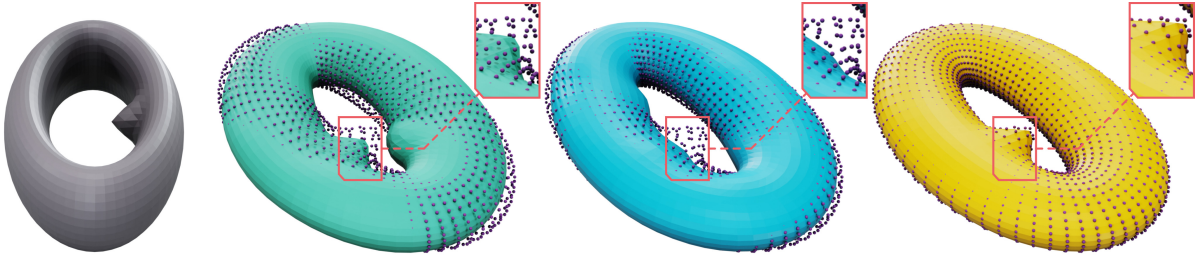


Figure 5.16: Approximation of the Freaky Torus. We show the reference shape \bar{z} in grey, the approximation of $\text{Exp}_{\bar{z}}^K v$ by affine combination of exact factors in green, by a monolithic network in blue, and by our composite network in yellow. The correct vertex positions of $\text{Exp}_{\bar{z}}^K v$ are shown as purple points. These purple points should ideally lie on the shaded surface, which would indicate a good fit. Indeed, for the approximation by our composite network, this is mostly the case, while for the other two approaches the approximation does not match the point cloud in many places.

5.6 Parametrization via Composite Networks

The (approximate) parametrization of the SPGA manifold $\mathcal{M}[\mathcal{U}]$ via the affine combination of multilinear interpolations of the exponential map on factor manifolds yielded an efficient tool for nonlinear synthesis of deformations in a data-driven model. However, it has the crucial limitation that the individual subspaces \mathcal{U}_l have to be of limited dimensionality ($m_l \leq 4$) for it to be computationally feasible. This can become a problem, for example, if we require higher dimensional SPGA manifolds for humanoid datasets where also the face and hands are articulated. Such models and datasets are becoming increasingly more popular with one recent example being SMPL-X [PCG+19]. Hence, in this section, we will turn to neural networks as an alternative method to parametrize such manifolds constructed from data. However, training a single fully connected network to approximate the exponential map and thus parametrize the SPGA manifold does not perform satisfactorily as we will see in Section 5.6.1. Instead, we will use our structural insights from Section 5.3 to create a network architecture that yields an efficient and accurate parametrization.

Neural Networks. Before we detail this architecture, we first introduce our notation for neural networks. We indicate functions that are implemented as neural networks by the superscript ζ as in φ^ζ . This ζ represents the network parameters and is the same for all occurring networks, with the implicit convention that different networks depend on different subsets of these parameters.

We denote by $\text{MLP}_\rho^\zeta[N_1, \dots, N_T]$ a fully-connected network with layer sizes N_1, \dots, N_T , non-linear activation function $\rho: \mathbb{R} \rightarrow \mathbb{R}$ after each layer, and parameters ζ . That means its evaluation $\text{MLP}_\rho^\zeta[N_1, \dots, N_T](z^1) = z^T$ is defined recursively through $z^{t+1} := \rho(W^{t+1}z^t + b^{t+1})$ for $t \in \{1, \dots, T-1\}$ with weights $W^{t+1} \in \mathbb{R}^{N_{t+1} \times N_t}$ and biases $b^{t+1} \in \mathbb{R}^{N_{t+1}}$ —all stored in the parameters ζ . For graph convolutional networks, we adapt the approach by Kipf and Welling [KW17] to data $z_e^1 \in \mathbb{R}^{N_0}$ on edges $e \in \mathbf{E}$: The t th layer is given by

$$z_e^{t+1} = \rho \left(W_1^{t+1} z_e^t + W_2^{t+1} \sum_{\tilde{e} \in \mathcal{N}(e)} z_{\tilde{e}}^t + b^{t+1} \right) \quad (5.23)$$

where $W_1^{t+1} \in \mathbb{R}^{N_{t+1} \times N_t}$ and $W_2^{t+1} \in \mathbb{R}^{N_{t+1} \times N_t}$ are small matrices with learnable parameters and $b^{t+1} \in \mathbb{R}^{N_{t+1}}$ is the bias. Note, that in this case the sizes of the layers N_t refers to the data dimension per edge. We define the neighborhood of an edge in a triangle mesh as $\mathcal{N}(e) := \{\tilde{e} \in \mathbf{E} \mid e \text{ and } \tilde{e} \text{ share a vertex}\}$. We used the Exponential Linear Unit [CUH16] as activation function in all our experiments, i.e.

$$\rho(x) = \begin{cases} x & x > 0 \\ (e^x - 1) & x \leq 0. \end{cases} \quad (5.24)$$

Structural Assumptions. We want to learn an efficient parametrization $\Phi^c: \mathbb{R}^J \rightarrow \mathcal{M}^J$ of a J -dimensional Riemannian data manifold \mathcal{M}^J by exploiting a specific product structure. One can think of the parametrization as the decoder part of an autoencoder. The assumptions below directly originate from our observations on the structure of SPGA manifolds in Section 5.3. Nevertheless, we will formulate them in an abstract language as we expect them to also apply in other scenarios, which we will briefly explain afterwards. However, we will focus on the situation in which Φ should encode the Riemannian exponential map at some point $\bar{z} \in \mathcal{M}^J$. Then, the structure of \mathcal{M}^J we want to exploit has three components:

- (1) **Correlation.** We assume a structural correlation between the different coordinates, e.g. graph-neighbour relations for triangular meshes or pixel-neighbour relations for image data, so that convolutional networks can be applied. In our SPGA scenario, we achieve this by using NRIC.
- (2) **Factorization.** We assume that the manifold can be smoothly approximated by a product of much lower-dimensional manifolds $\mathcal{M}_1, \mathcal{M}_2, \dots, \mathcal{M}_L$ we will parametrize separately. This way, we exploit that the necessary network size as well as the required training effort decrease with smaller manifold dimension: For m -dimensional manifolds the network size should scale at least linearly in m , while the required training set and thus also training time will scale exponentially in m .
- (3) **Combination.** It is not sufficient that the single factor manifolds are easy to parametrize since a generic point on \mathcal{M} has components in all factors. Thus, we assume that the direct sum of all factors

$$\mathcal{M}_1 \oplus_{\bar{z}} \dots \oplus_{\bar{z}} \mathcal{M}_L := \{\bar{z} + (z_1 - \bar{z}) + \dots + (z_L - \bar{z}) \mid z_1 \in \mathcal{M}_1, \dots, z_L \in \mathcal{M}_L\}$$

for some $z \in \mathcal{M}$ already approximates the actual manifold \mathcal{M} with a (possibly large, but) smooth approximation error. This will ensure that a suitable map from $(z_1 - z, \dots, z_L - z)$ to \mathcal{M} can be efficiently learned. We have seen that this and the previous condition are reasonable for SPGA manifolds of articulated shapes in Section 5.3.

Note that while condition (1) essentially has to hold for any approach employing neural networks, conditions (2) and (3) are specific to our approach. Condition (2) expresses that the *intrinsic geometry* of the data manifold \mathcal{M} has a simplifying structure, while condition (3) is about the *extrinsic geometry* of how \mathcal{M} is embedded in \mathbb{R}^n —only both conditions together characterize a structure that can efficiently be learned.

The toy shape manifold \mathcal{M} of tori from Figure 5.16 has the flat metric of $S^1 \times S^1 \times \mathbb{T}^2$ —the factors representing the orientation of the longitudinal and latitudinal ellipsoidal cross-sections as well as a bump position, see Figure 5.18—and thus satisfies condition (2). Condition (3) holds since each torus is represented as NRIC: For instance, the creation of the bump (which corresponds to changing the position in \mathbb{T}^2) is well described by simply adding fixed numbers in the right places of the edge length vector $l(X)$. However, the direct sum is indeed only an approximation. Otherwise, there would be no error in Figure 5.16 (green), but one clearly sees a remnant of the bump from the reference shape.

This shows that we can expect these structural assumptions to hold for other data manifolds beyond those constructed using SPGA. We think that they are plausible in two rough scenarios: The first is different factor manifolds corresponding to different spatial regions. For instance, images may sometimes be partitioned into different regions that can vary more or less independently of each other. Also SPGA manifolds fall into this scenario. The second scenario comprises variations at different length scales that are independent of each other, for example, geometric texture on small length scales versus global shape variations at large length scales. The toy shape manifold of tori from above falls into this category. In our computational experiments in Section 5.6.1, we will focus on examples of the first kind constructed via SPGA.

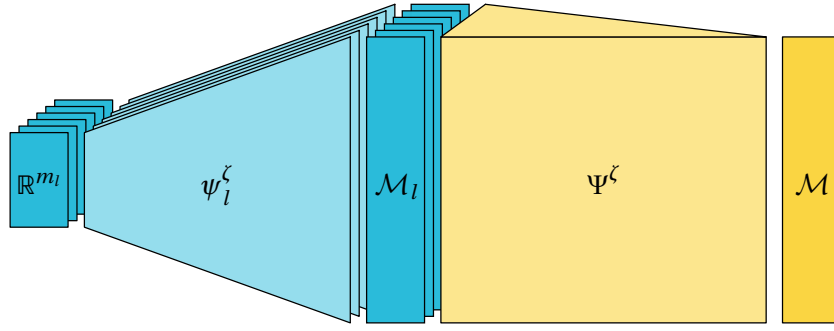


Figure 5.17: Structure of composite network.

Network Representation and Training. Before we discuss the previous conditions, let us describe how we exploit them in our network architecture and training procedure. To learn the parametrization $\Phi^\zeta: \mathbb{R}^J \rightarrow \mathcal{M}$, we decompose it as

$$\Phi^\zeta = \Psi^\zeta \circ (\psi_1^\zeta, \dots, \psi_L^\zeta),$$

where $\psi_l^\zeta: \mathbb{R}^{m_l} \rightarrow \mathbb{R}^{2|\mathcal{E}|}$ is the parametrization of the m_l -dimensional factor manifold \mathcal{M}_l and $\Psi^\zeta: (\mathbb{R}^{2|\mathcal{E}|})^L \rightarrow \mathbb{R}^n$ is the combination of the single factors, behaving approximately like $(\bar{z}_1, \dots, \bar{z}_L) \mapsto \bar{z} + (z_1 - \bar{z}) + \dots + (z_L - \bar{z})$, see Figure 5.17.

The maps ψ_l^ζ will be fully connected neural networks. Because they approximate the smooth exponential map on rather low-dimensional spaces, they are expected to achieve a high approximation quality that is typically stable under variation of the concrete network architecture (number and size of layers). For the maps Ψ^ζ , we exploit that they operate on a structured domain, i.e. a mesh, and thus they will be convolutional neural networks. This allows for efficient training and storage even though they operate on high-dimensional data. One could alternatively also use fully connected networks for Ψ^ζ , but the observed quality of the results was similar despite significantly increased memory requirements to train, store, and evaluate.

These networks are trained separately: To train the map ψ_l^ζ , we consider a basis $u_1^l, \dots, u_{m_l}^l$ of the Riemannian logarithm $\mathcal{U}_l \subset T_{\bar{z}}\mathcal{M}$ of \mathcal{M}_l and corresponding parametrization $\omega_l(a^l) = \sum_{i=1}^{m_l} a_i^l u_i^l$ for a coefficient vector $a^l = (a_1^l, \dots, a_{m_l}^l)$. In the case of SPGA manifolds, the basis arises by the grouping of SPGA modes as in Section 5.3. We then consider a set of random samples $S^l \subset \mathbb{R}^{m_l}$ as training data (for instance normally distributed or uniformly on a ball) and minimize the loss function

$$\mathcal{J}_l(\zeta) = \frac{1}{|S^l|} \sum_{a^l \in S^l} \|\text{Exp}_{\bar{z}}^K(\omega_l(a^l)) - \psi_l^\zeta(a^l)\|_2^2$$

using the weighted L^2 -norm $\|\cdot\|_2$ from Section 4.3.1. To train the map Ψ^ζ , we subsequently consider a random training set $S \subset \mathbb{R}^{m_1} \times \dots \times \mathbb{R}^{m_L} = \mathbb{R}^J$ and minimize the loss function

$$\mathcal{J}(\zeta) = \frac{1}{|S|} \sum_{(a^1, \dots, a^L) \in S} \|\text{Exp}_{\bar{z}}^K(\omega_1(a^1) + \dots + \omega_L(a^L)) - \Psi^\zeta(\psi_1^\zeta(a^1), \dots, \psi_L^\zeta(a^L))\|_2^2,$$

where we may or may not keep the maps ψ_l^ζ fixed.

5.6.1 Experiments and Applications

In this section, we present experimental results on the aforementioned synthetic shape manifold of deformed tori and manifolds extracted via SPGA. We will compare our method to approaches based on Section 5.3.2 and a straightforward approximation of the parametrization by a single fully

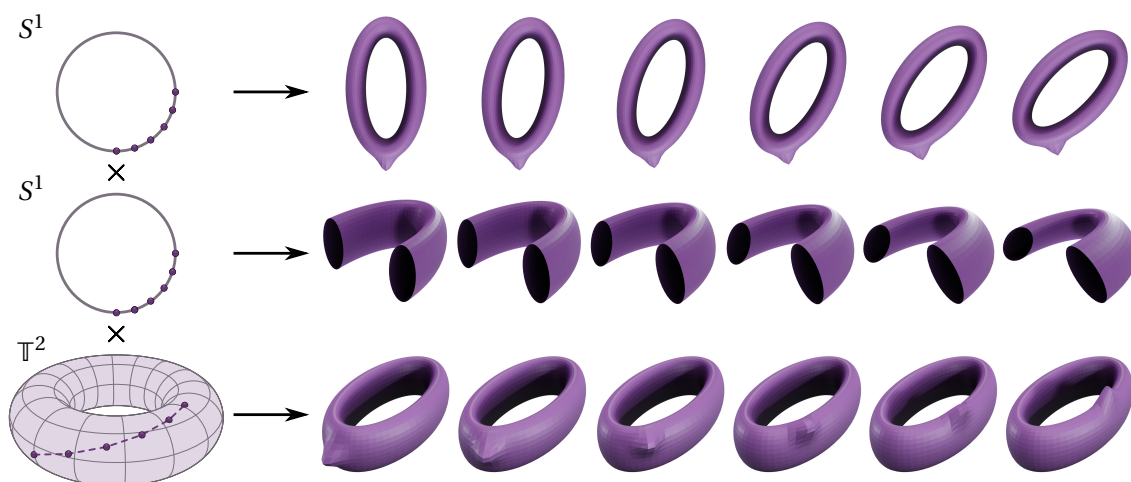


Figure 5.18: *Factors of the Freaky Torus.* We visualize our synthetic shape space by demonstrating the effect of moving along the individual factors to the final shape. In the first row, we see how the first factor, an S^1 , controls the deformation of the latitudinal cross-section. We show the deformed torus from the top. In the second row, we see how the next factor, another S^1 , controls the deformation of the longitudinal cross-section. Here, we show the torus cut in half to better highlight the cross-section's shape. In the last row, we show how the third factor, a two-dimensional flat torus \mathbb{T}^2 , controls the position of a bump on the deformed torus.

connected network. To quantify the approximation quality of different approaches, we use the coefficient of determination R^2 . For approximations \tilde{z}_i of NRIC z_i with mean \bar{z} , it is defined as

$$R^2(z, \tilde{z}) = 1 - \frac{\sum_i \|\tilde{z}_i - z_i\|_2^2}{\sum_i \|z_i - \bar{z}\|_2^2}.$$

From a statistical point of view, it quantifies the proportion of variation of the data that is explainable by a given model. This means an R^2 of one is optimal, the smaller it is the worse is the approximation, and a negative R^2 means that the model is worse than simply using the mean.

Training & Implementation. We used Adam [KB15] as descent method for training all networks, where the initial learning rate was 10^{-3} and was reduced by a factor of 10 every time the loss did not decrease for multiple iterations. For regularization, we used batch normalization after each layer and a moderate dropout regularization ($p = 0.1$) after each convolutional layer. We implemented the neural networks in PyTorch [PGM+19] using the PyTorch Geometric library [FL19]. The tools for the NRIC manifold were implemented in C++ as in Section 5.4 and we also used the multi-resolution scheme described there.

Synthetic Data: Freaky Torus. For the *Freaky Torus* dataset, we construct a synthetic shape space with factors $S^1 \times S^1 \times \mathbb{T}^2$, where \mathbb{T}^2 refers to the flat 2-dimensional torus. It is realized in NRIC by (i) deforming the two cross-sectional circles of a torus to ellipses of fixed aspect ratio and orientation controlled by the first two S^1 factors and (ii) growing a bump in normal direction whose position is controlled by the last \mathbb{T}^2 factor. The factors are visualized in Figure 5.18. These torus deformations are applied to a regular mesh of a regular torus embedded in \mathbb{R}^3 , and the deformed meshes' NRIC are extracted to obtain our datapoints. We used a mesh with 2048 vertices and uniformly drew 1000 samples from $S^1 \times S^1 \times \mathbb{T}^2$.

Figure 5.16 shows that a single fully connected network struggles with approximating the high frequency detail of the bump, while our composite network is able to handle this well. This can also

be observed in the approximation quality quantified using the R^2 . The composite network achieves an R^2 of 0.99 and the monolithic network one of 0.95. This difference may sound small, however, this is because the bump is a detail and the error is dominated by the overall shape of the torus.

Application: SPGA Manifolds. Furthermore, we report the results of applying our method to shape manifolds whose approximate product structure is found with the help of SPGA. To this end, we repeat three of the examples discussed in Section 5.5 and consider one new dataset. The repeated examples are the humanoid SCAPE dataset, the dataset of face meshes, and the dataset of hand meshes. For the new example, we examine a humanoid dataset based on SMPL-X [PCG+19], where we consider their *expressive hands and faces* (EHF) dataset—containing 100 shapes—and 49 additional shapes from the SMPL+H dataset, which feature more expressive arm and leg movements, adding to a total of 149 input shapes.

Based on this data, we compute the sparse tangent modes and factor the resulting data manifolds as in Section 5.5. We report the chosen number J of included modes in Table 5.5, where we used the same number as before for the repeated examples. For the hand and face examples, we choose the same number of factors L , while for the SCAPE example we decreased the number to account for the possibility to handle higher-dimensional factors with the network-based approximation. Our choices are again documented in Table 5.5. Each cluster then spans exactly one of the factor manifolds and the range of their dimensions is also reported in said table.

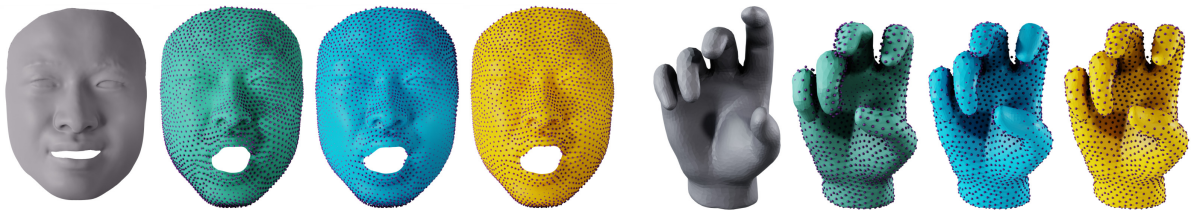


Figure 5.19: Examples from the face and hands datasets. We use the same colors as in Figure 5.16.

Data Generation. Next, we sample the exponential map on the space spanned by the SPGA modes to generate the training (and test) data. To this end, we consider the hypercube in \mathbb{R}^J given by the minimal and maximal coefficients of projections of the input data onto the SPGA subspace. Then we draw our parametrization coefficient samples $S \subset \mathbb{R}^J$ uniformly from this hypercube. To create the samples S^l for the factor manifolds, we simply take the corresponding subcomponents of coefficient vectors from S . The corresponding shapes were then computed by evaluating the discrete exponential map for each of them. Overall, we sampled approximately $|S| = 4000$ points for each of the considered examples. The dataset was split randomly into a training (80%) and a test (20%) set, with the training set being used for the descent method of the loss functionals and the test set being used to evaluate the performance of the networks.

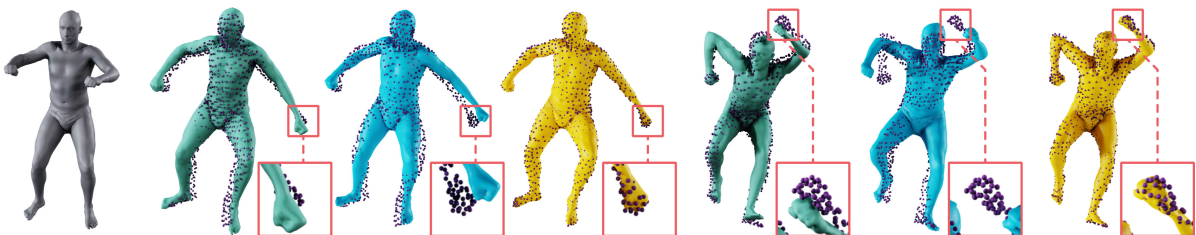


Figure 5.20: Two examples from the SCAPE dataset. We use the same colors as in Figure 5.16.

Example	J	L	m_l	Affine	Monolithic	Composite	Affine+Grid (Section 5.3.2)
SMPL+X	80	10	3 – 24	0.78	0.85	0.93	—
SCAPE	40	6	5 – 9	0.77	0.60	0.91	—
Hands	12	4	2 – 4	0.88	0.95	0.98	0.80
Faces	10	6	1 – 4	0.96	0.95	0.99	0.95

Table 5.5: Approximation quality R^2 on SPGA examples.

Comparison to Affine Approximation. Recall, we proposed a scheme based on multilinear interpolation of precomputed exponentials for each of the factor manifolds and subsequent affine combination of the results to approximate the exponential map and parametrize \mathcal{M}^J in Section 5.3.2. A natural question is how our network-based approach compares to this.

For the humanoid examples, it was not possible to precompute Riemannian exponentials on a regular grid for all factor manifolds due to their high dimensionality. Hence, instead of multilinearly interpolating precomputed exponentials, we simply compute the exponentials within each factor manifold exactly before combining them affinely. We dub this method simply ‘affine combination’ and present its results in Figure 5.20 and Table 5.5; it is computationally heavy, but yields an upper bound on the quality of the method from Section 5.3.2. The limitation does not apply to our network-based approach, which for example allows us to learn an efficient parametrization for the SMPL+X dataset, where the expressive movements of hand and face require a higher-dimensional data manifold.

In all examples, our composite network approach achieves higher approximation accuracy than the ‘affine combination’. This shows that the network Ψ^ζ is able to correct the approximation errors of the direct sum structure. For the lower-dimensional examples, this difference is not as pronounced since their sparse modes have a better support separation.

Furthermore, storing our network-based approximation requires less memory than the grid-based approach. For example, on the SCAPE dataset storing grids with approx. 20000 samples requires about 1.7 GB of storage, while our networks only require 0.6 GB (without optimizing for a small memory footprint).

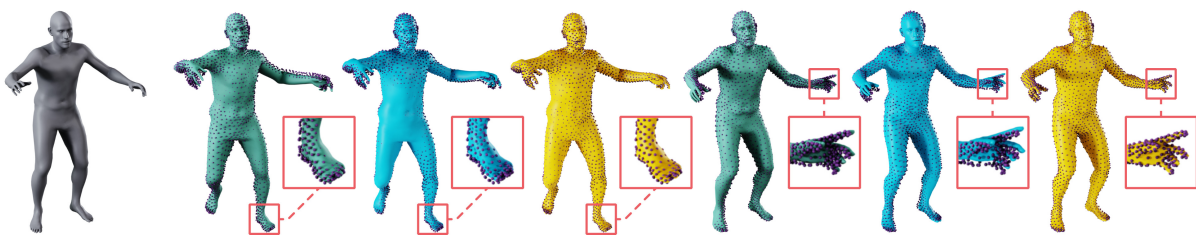


Figure 5.21: Two examples from the SMPL-X dataset. We use the same colors as in Figure 5.16.

Comparison to Monolithic Network. Another obvious question is whether our composite approach shows any benefit over training a simple, single network. For evaluation, we also trained one fully connected network $\tilde{\Phi}^\zeta: \mathbb{R}^J \rightarrow \mathcal{M}^J$ to approximate the parametrization at once, dubbed ‘monolithic’ approach. The corresponding approximation qualities are reported in Table 5.5 under the heading *Monolithic*. One sees that for the lower-dimensional examples this monolithic approach achieves an approximation quality close to the one of our composite network. However, for the higher-dimensional, humanoid examples the approximation quality of the monolithic approach is noticeably lower.

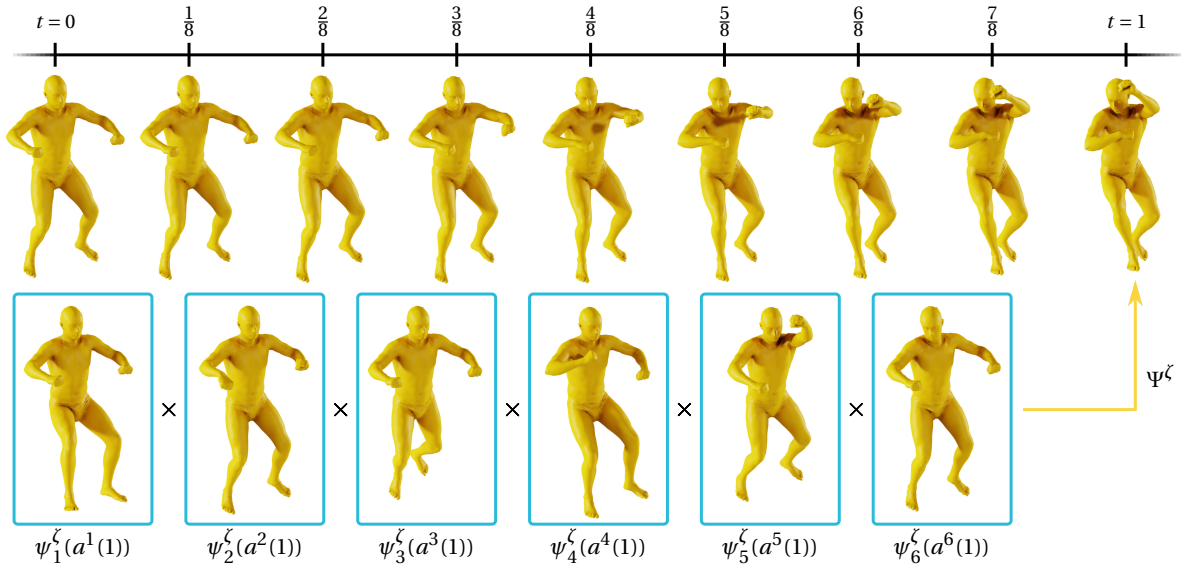


Figure 5.22: For two given shapes with latent coordinates $a(0)$ and $a(1)$, we compute interpolating NRIC $z(t)$ using our composite network. In the top row, we see the surfaces reconstructed from these NRIC for intermediate time steps exhibiting smooth deformations. Below, we also show the elements $\psi_j^\zeta(a^j(1))$ from the factor manifolds \mathcal{M}_j which lead to the final shape by applying the combination network Ψ^ζ . These individual factors lead primarily to deformations of the legs for ψ_1 and ψ_3 , of the arms for ψ_4 and ψ_5 , of the wrists for ψ_2 , and of the head for ψ_6 .

Animation. One possible application of our composite network is again the efficient animation of shapes. As before, we can consider shape interpolation and extrapolation problems, which correspond to the evaluation of the Riemannian logarithm and exponential map. For the case of shape interpolation, we are given two shapes by their latent coordinates $a(0) \in \mathbb{R}^J$ and $a(1) \in \mathbb{R}^J$ respectively. Then, the latent coordinates of intermediate shapes are obtained by linear interpolation, i.e. we define $a(t) := t a(1) + (1 - t) a(0)$ for $t \in [0, 1]$. By evaluating our composite network on these coordinates, we obtain the approximate NRIC $z(t) := \Psi^\zeta(\psi_1^\zeta(a^1(t)), \dots, \psi_J^\zeta(a^J(t)))$ of these shapes, where $a^j(t) \in \mathbb{R}^{m_j}$ are the factorized coordinates as before. This leads to a smooth interpolation between shapes as demonstrated in Figure 5.22.

Number of Samples. We observed that our composite network can also be trained with smaller amounts of samples than we used above. For example on the SCAPE dataset, if we only use 20% of the data as training set (about 800 samples) then we still achieve an R^2 of 0.86. Even if we use a mere 5% (200 samples) we still reach an R^2 of 0.77. We observed a similar behavior on the SMPL+X dataset with an R^2 of 0.85 at 20% training data and of 0.74 at 5% training data.

Runtimes. Our network-based approach enables runtime efficient approximation of the exponential map. For example, on the SCAPE dataset, we used $K = 16$ time steps to evaluate the time-discrete exponential map when generating the training samples. The computation for each such evaluation required around 8 seconds. In contrast, evaluating the networks takes about 10 milliseconds. To render the result, we have to reconstruct the nodal positions of the triangle mesh from the NRIC, for which we use the nonlinear least-squares method from Fröhlich and Botsch [FB11] as explained in Section 4.1. This requires again a small number (e.g. 2 to 3 in Figure 5.22) of Gauß–Newton iterations taking about 20ms each. Overall the performance is comparable to the approach grid-based approach from before.

5.7 Conclusion and Outlook

In this chapter, we have introduced a new method to construct submanifolds of the shape space of discrete shells approximating given data that can be parametrized efficiently. We picked up the approach from Principal Geodesic Analysis to describe such manifolds using the exponential map on the tangent space at a mean shape. However, we modified the problem to compute the basis of the subspace over which the submanifold is parametrized by including a sparsity-inducing regularization. For this to make sense, it was crucial that we used NRIC as primary coordinates for the shape space. In the experiments, we saw that the sparse modes often can be assigned some semantic meaning such as moving a joint or muscle group on a humanoid model. Furthermore, we saw that the modes tend to group-wise decouple—i.e. have mostly disjoint support—and we used this to introduce a product structure on the constructed submanifold. We then proceeded to show two ways how this structure can be exploited to create an efficient parametrization. First, we considered a grid-based multilinear interpolation on each factor together with an affine combination—relying again on NRIC for this. This yielded an efficient and useful approximate parametrization but was limited in the number of dimensions it could handle due to the needed grids. Hence, we proposed a neural network-based alternative parametrization second. We built the product structure of the submanifold into the network architecture and its training to dramatically improve its performance. This allows us to better approximate the parametrization of the submanifold while still being useful for realtime shape interpolation and editing. Overall, this means we have constructed the first methods to use Riemannian operators on the space of discrete shells in realtime applications.

In the future, it would be interesting to gain a better theoretical understanding of our decoupling resp. product structure in the language of Riemannian geometry. To this end, it seems reasonable to connect it to the de Rham decomposition theorem, which allows to identify product structures in the tangent space and has, for example, been recently used by Pfau *et al.* [PHBR20] to do so in a manifold learning context. An important tool for this would be the time-discrete covariant derivative from Effland *et al.* [EHRW22]. This might also allow us to generalize our construction to other shape spaces.

Identifying product-like structure in data is the principal goal of ‘disentanglement learning’—an increasingly popular direction of machine learning. An interesting extension of our work in this direction would be to also construct an encoder network counterpart for our composite network decoder built on our structural insights. This would allow to generalize the approach to other shape spaces where no equivalent to SPGA is available. Furthermore, it might even allow us to apply it to scenarios where we do not have or know an underlying shape space.

To further improve the performance of our method—especially to reduce the effort necessary for the projection onto the NRIC manifold—it would be worthwhile to devise a method to build the constraint of mapping to the manifold into the network. Since explicitly integrating this into architecture seems arduous, an interesting first idea would be to include a penalty or constraint based on the discrete integrability condition from the previous chapter into the training.

Finally, when passing to an autoencoder setup, we will obtain immersion of data manifold into relatively low-dimensional Euclidean latent spaces. It would be interesting to obtain implicit representations or even parametrizations of these latent manifolds. This would enable formulation (time-discrete) geodesic operators on these, which could be useful too in many applications. For example, it could allow finding a joint manifold structure on a space of remeshed discrete surfaces.

Chapter 6

The Space of Repulsive Shells

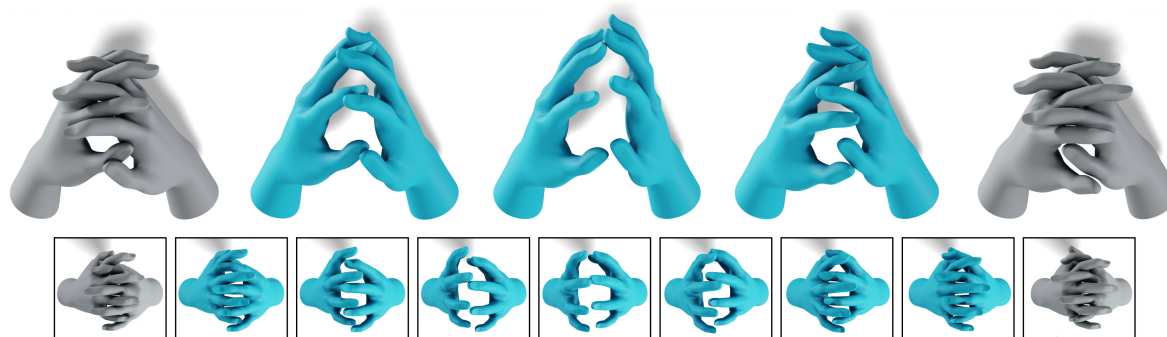


Figure 6.1: *Intersection-free interpolation between clasping hands. Our proposed method computes interpolations between surfaces as geodesics on a Riemannian shape space, where the metric guarantees the avoidance of self-intersections. Here, we show the fixed end shapes in gray and the interpolated shapes in blue from two perspectives.*

All the methods for the shape space of discrete shells presented so far have one crucial limitation: They do not guarantee that the resulting shapes are free of self-intersections. The membrane energy guarantees that all surfaces are *immersions* by moving discrete surfaces with collapsed triangles infinitely far away due to the logarithmic term in the energy density. However, nothing ensures that we obtain *embeddings*, which can lead to implausible results as we have, for example, seen in Figure 5.14. Our goal in this chapter is to rectify this problem by constructing a modification of the metric that also moves discrete surfaces with self-intersections infinitely far away. This will allow us to perform shape interpolation and extrapolation avoiding intersections leading to natural looking results, e.g., in Figure 6.1.

Previous methods that guarantee intersection-free shape interpolation focused on ambient diffeomorphisms to describe deformations. This requires a sufficiently highly resolved discretization of ambient space, which becomes computationally intractable for initial shapes close to contact as in Figure 6.1. Instead, we continue to consider triangle meshes. By modifying the metric of the shape space—instead of, for example, adding penalties to the resulting variational problems—we can carry over all the tools and applications from the space of discrete shells discussed so far.

To this end, we turn to recent work on preventing self-intersections in geometric variational problems. Yu *et al.* [YBSC21] showed that the tangent-point energy provides a suitable regularization in this context and developed efficient numerical methods for it. The tangent-point energy is a geometric curvature energy originally proposed by Buck and Orloff [BO95] to study knots with geometric analysis and was later generalized to higher-dimensional manifolds by Banavar *et al.* [BGMM03].

It is repulsive, which means that it pushes away points that are close in ambient space but distance along the surface, i.e. close to intersection. Especially, it is only finite for embedded surfaces and goes to infinity if a surface approaches self-intersections. By considering the graph of the tangent-point energy over the space of shells, we can indeed construct a metric that moves discrete surfaces with self-intersections infinitely far away. Thus, in this chapter, we introduce the *space of repulsive shells*.

In Section 6.1, we will introduce the tangent-point energy and an efficient and truly repulsive discretization. Afterwards, we will present our modified shape space including the variational time-discretization of geodesic calculus in Section 6.2. Finally, in Section 6.3, we will describe the numerical methods necessary to solve the variational problems originating from this time-discretization and show results obtained using our approach. Furthermore, in this last section, we will also show some additional experiments on combining the elastic discrete shells energy with the tangent-point energy beyond the shape space framework.

Remark. This chapter is the result of joint work with Keenan Crane, Martin Rumpf, and Henrik Schumacher to be published in [CSSR23]. Preliminary results appeared in the non-peer-reviewed report [Sas22]. Notably, Henrik Schumacher primarily developed and implemented the fast multipole approximation and the adaptive discretization introduced in Section 6.1.

6.1 Tangent-Point Energy

The tangent-point energy—as the name suggests—is a geometric energy that uses the tangent spaces and points on a manifold to generate a self-avoiding behavior. It originates from the study of *geometric curvature energies*. These are self-avoidance energies for lower-dimensional subsets—e.g. submanifolds—of Euclidean space defined via the geometry of these sets. They originate from geometric knot theory which studies knots and their representation by space curves. In this context, the purpose of these energies is to provide insights into knots via variational problems, e.g. find minimizers as particularly nice representations of knots, or derive topological information about knots via energy bounds. For these applications, it is crucial that the used functionals provide an infinite energy barrier to self-intersection to have control over the knot class. This makes the energies highly singular, nonlinear, and nonlocal and thus challenging to study.

Interest in such energies for curves resp. knots arose in the 1990's. One of the first examples of a geometric curvature energy was the Möbius energy that regularizes a Coulomb-type repulsive energy by a suitable term based on intrinsic distances to obtain an energy that is finite and repulsive on smoothly embedded curves. Starting from this, a range of further energies have been introduced, such as ropelength, the (integral) Menger curvature, and the tangent-point energy, which we will see in more detail below. More recently, e.g. in [BGMM03], these energies have been generalized to higher-dimensional subsets such as surfaces.

So far, regularizing and topological effects as well as the connection to knot theory of these energies have been investigated. Recently, the regularity theory has been extended to the fractional Sobolev setting [Bla13]. This also led to efficient numerical minimization schemes based on Sobolev gradient descent, e.g., in [RS21]. These schemes can yield crucial insights into the energy landscape of the energies, which are currently also a subject of interest, for example, in the form of characterizing critical points. Through the work of Yu, Schumacher, and Crane [YSC21] the tangent-point energy has recently also made its way to computer graphics. The fact that it is not invariant to scaling or Möbius transformations, that it does not require intrinsic distances on the surfaces, and availability of efficient numerical tools for its minimization make it a prime candidate for our efforts. Below, we will introduce it in more detail and summarize some important results about its regularizing effects. More details on this and the history and analysis of geometric curvature energies in general can be found in the recent survey [SM22].

6.1.1 Definition and Properties

In the following, we will introduce the tangent-point energy and its properties on a submanifold of arbitrary dimension and not necessarily a surface. To this end, let $\mathcal{S} \subset \mathbb{R}^d$ be a closed, n -dimensional, immersed submanifold of class C^1 . For two points $p, q \in \mathcal{S}$, the *tangent-point radius* $R_{\text{tp}}(p, q)$ is defined as the radius of the smallest sphere through p and q that is also tangent to \mathcal{S} at p . In formulas, this radius can be expressed as

$$R_{\text{tp}}(p, q) := \frac{|p - q|^2}{2|P(p)(p - q)|},$$

where $P(p)$ denotes the orthogonal projection onto the normal space of \mathcal{S} at p . For example, for hypersurfaces, this projector is given as the outer product of the normal vector $n(p)$ with itself, i.e. $P(p) = n(p)n(p)^T$.

How does this radius relate to how well-embedded a surface/manifold is? To answer this, we consider the *reach* of a manifold \mathcal{S} , which is defined as the largest distance such that all points closer to \mathcal{S} have a unique closest point on \mathcal{S} . Note that if \mathcal{S} is not embedded, e.g., has self-intersections, then $\text{reach}(\mathcal{S}) = 0$. Physically, the reach of a surface can also be thought of as half the thickness of an object whose boundary is described by it. More generally, the reach is related to the curvature of the manifold and a large reach entails the absence of small oscillations. Coming back to the tangent-point radius, Federer [Fed59] showed the following identity between the reach and the tangent-point radius:

$$\text{reach}(\mathcal{S}) = \inf_{p, q \in \mathcal{S}, p \neq q} R_{\text{tp}}(p, q).$$

Thus the function R_{tp} can be used to measure how well-embedded \mathcal{S} is. A natural approach to exploit this for variational problems would be to use the inverse of the reach as a barrier functional against self-intersections, which would correspond to an L^∞ -norm of the tangent-point radius. However, the lack of L^∞ -norm turns (numerically) minimizing this functional into an arduous task. Instead, we consider the L^p -norm of the tangent-point radius and arrive at

Definition 6.1 (Tangent-Point Energy). Let $\mathcal{S} \subset \mathbb{R}^d$ be a closed, n -dimensional, embedded submanifold of class C^1 . The *tangent-point energy* (TPE) for integrability parameter $\alpha \geq 1$ is defined as

$$\mathcal{T}^\alpha[\mathcal{S}] := \left\| \frac{1}{2R_{\text{tp}}} \right\|_{L^\alpha(\mathcal{S} \times \mathcal{S})}^\alpha = \int_{\mathcal{S}} \int_{\mathcal{S}} \frac{|P(p)(p - q)|^\alpha}{|p - q|^{2\alpha}} d\mathcal{H}^n(p) d\mathcal{H}^n(q), \quad (6.1)$$

where \mathcal{H}^n denotes the n -dimensional Hausdorff measure on \mathbb{R}^d .

Remark. We could have introduced the tangent-point energy as well as the following results for a larger class of manifolds than immersed submanifolds of class C^1 . However, for this thesis, considering the entire class is not necessary and thus we omit its technical description.

Properties. Our goal is to use the tangent-point energy to define a metric on the space of surfaces which guarantees that operators from (time-discrete) geodesic calculus lead to surfaces without self-intersections. Hence, we will summarize here the central properties of the tangent-point energy that pave the road towards this goal. The first realization is that the feasible set $\{\mathcal{T}^\alpha < \infty\}$ indeed only consists of embeddings for a sufficiently large integrability parameter α . This is the content of

Theorem 6.2. Let $\mathcal{S} \subset \mathbb{R}^d$ be a closed, n -dimensional, immersed submanifold of class C^1 such that $\mathcal{T}^\alpha[\mathcal{S}] < \infty$ for some $\alpha > 2n$. Then \mathcal{S} is already an embedded submanifold of class $C^{1,\mu}$, where $\mu = 1 - \frac{2n}{\alpha}$.

It was conjectured in [BGMM03] and rigorously proven in [SM13, Theorem 1.4]. The regularity aspect can be further improved to

Theorem 6.3. *Let $S \subset \mathbb{R}^d$ be a closed, n -dimensional, immersed submanifold of class C^1 such that $\mathcal{T}^\alpha[S] < \infty$ for some $\alpha > 2n$. Then S is an embedded $W^{2-\frac{n}{\alpha}, \alpha}$ submanifold.*

This was proven in [Bla13, Theorem 1.1]. Here, $W^{m+s, \alpha}(\mathbb{R}^d)$ for $m \in \mathbb{N}$, $s \in (0, 1)$, and $\alpha \in [1, \infty)$ denotes the *Sobolev-Slobodeckij space* defined as

$$W^{m+s, \alpha}(\mathbb{R}^d) := \{f \in W^{m, \alpha}(\mathbb{R}^d) \mid \|f\|_{m+s, \alpha}^\alpha < \infty\}$$

with the Sobolev-Slobodeckij norm

$$\|f\|_{m+s, \alpha}^\alpha := \|f\|_{W^{m, \alpha}(\mathbb{R}^d)}^\alpha + \sum_{|\theta|=m} \int_{\mathbb{R}^d} \int_{\mathbb{R}^d} \frac{|D^\theta f(x) - D^\theta f(y)|^\alpha}{|x-y|^{d+s\alpha}} dy dx,$$

where $\theta \in \mathbb{N}^d$ in the usual multi-index notation. These spaces are a type of fractional Sobolev spaces originating from generalizing Hölder continuity to Lebesgue spaces. An embedded $W^{m+s, \alpha}$ submanifold is then a submanifold which is locally the graph of a $W^{m+s, \alpha}$ function. A comprehensive introduction to fractional Sobolev spaces can be found in [NPV12]. The regularity result Theorem 6.3 led to the development of Sobolev preconditioning strategies for the optimization of the tangent-point energy in [YSC21; YBSC21].

Finally, we also need that the tangent-point energy is indeed a barrier against self-intersections, meaning that it prevents a converging sequence of embedded manifolds from approaching a non-embedded surface. This is formalized in

Theorem 6.4. *Let $S_j \subset \mathbb{R}^d$ be closed, n -dimensional, embedded submanifolds of class C^1 with $\mathcal{T}^\alpha[S_j] < C$ for all $j \in \mathbb{N}$, where $C > 0$. If the S_j converge to a compact set $S \subset \mathbb{R}^d$ with respect to the Hausdorff-metric as $j \rightarrow \infty$, then $S \subset \mathbb{R}^d$ is a closed, n -dimensional, embedded submanifold with $\mathcal{T}^\alpha[S] < C$.*

The rigorous version of this theorem along with its proof can be found in [KSM18]. It also includes statement of lower semicontinuity and compactness for such sequences, which are then used to prove the existence of minimizers of \mathcal{T}^α under a bounded-diameter constraint within each isotopy class. Hence, we got an even stronger statement that allows us to additionally obtain bounds on the tangent-point energy of limiting surfaces. All this motivates and justifies to include the tangent-point energy into the structure of our shape spaces to guarantee avoidance of self-intersections. To do this, we first study its discretization to obtain a repulsive energy on discrete surfaces.

6.1.2 Discretization

We will now consider an immersed discrete surface $\mathbf{S} \subset \mathbb{R}^3$ as defined in Definition 2.19. This means different to the previous part of this section we will not consider the case of arbitrary submanifolds. However, generalizing the discretization of general simplicial submanifolds is straight forward. Recall, that \mathbf{T} denotes the set of triangles and a_τ the area of the face τ immersed via the piece-wise affine immersion $X: |\mathcal{S}_h| \rightarrow \mathbb{R}^3$. Then the orthogonal projector $P(\tau)$ onto the normal space of τ is given by $P(\tau) = N_\tau N_\tau^T$, where N_τ is the normal of the immersed face $X(\tau)$ and we denote its barycenter by $\bar{X}(\tau)$. With this notation in place, we can introduce a first naïve and yet prototypical discretization of the tangent-point energy via midpoint quadrature:

$$\mathcal{T}_h^\alpha[\mathbf{S}] := \sum_{\tau \in \mathbf{T}} \sum_{\substack{\tau' \in \mathbf{T} \\ \tau' \neq \tau}} \frac{|P(\tau)(\bar{X}(\tau) - \bar{X}(\tau'))|^\alpha}{|\bar{X}(\tau) - \bar{X}(\tau')|^{2\alpha}} a_\tau a_{\tau'} \quad (6.2)$$

Whenever we fix the connectivity \mathcal{S}_h of the discrete surface, we will write $\mathcal{T}_h^\alpha[X]$ for the evaluation on discrete immersions X of this surface.

The consistency of this discretization has been experimentally verified in [YBSC21]. However, it has two significant drawbacks:

- (a) The evaluation of \mathcal{T}_h^α scales quadratically in the number of faces $|\mathbf{T}|$ and thus quartically in the mesh resolution h .
- (b) \mathcal{T}_h^α is not truly repulsive, i.e. even with finite energy discrete surfaces can still intersect away from triangle barycenters.

These two drawbacks also play against each other, since we have to make h small to get a strong enough repulsion to prevent intersections but then the evaluation becomes dramatically more expensive. Hence, in the remainder of this section, we will consider remedies for these shortcomings. First, we will describe a fast multipole approximation of the energies—and its derivatives—that allows evaluating it much more efficiently. Second, we will introduce an adaptive discretization that automatically refines triangles close to contact to make the energy truly repulsive.

Fast Multipole Approximation. Typically, many terms of the discretized tangent-point energy (6.2) contribute only little to the energy since the involved faces are far apart. Hence, the energy dominated by only a small number of triangle pairs and we only need to compute the contribution accurately for those and can approximate it for the others without losing too much overall accuracy. This is the principal idea of the fast multipole approximation of the discrete tangent-point energy as described below.

To construct this approximation, we begin with splitting the mesh hierarchically into clusters $\mathbf{C} \subset \mathcal{P}(\mathbf{T})$, i.e. subsets of faces, where the energy will be only evaluated on pairs of leaf clusters. For a finite set of clusters \mathbf{C} , we introduce a hierarchical ordering $<$ by defining $U' < U$ for two clusters $U', U \in \mathbf{C}$ if $U' \subsetneq U$ and there exist no other cluster $U'' \in \mathbf{C}$ such that $U' \subsetneq U'' \subsetneq U$. If $U' < U$, we call U the parent cluster of U' and U' the child cluster of U . We call \mathbf{C} a *cluster tree* of \mathcal{S}_h if

1. $\mathbf{T} \in \mathbf{C}$ and it is the only cluster without a parent called the root of the tree,
2. the intersection of two clusters sharing the same parent is empty, and
3. each cluster $V \in \mathbf{C}$ with children is identical to the union of all its children, i.e. $V = \bigcup_{U' < V} U'$.

A cluster of a tree without children is called a leaf cluster and we denote the set of all leaf clusters by $L(\mathbf{C})$. In our application, we construct a cluster tree for an immersed mesh simply by splitting bounding boxes along their longest edge. That is for a given cluster, we consider its axis-aligned bounding box—i.e. the smallest axis-parallel cuboid containing all embedded vertices of the faces of the cluster—and split it in half along its longest axis. This yields two children containing the faces whose barycenters are left respectively right of the split. Hence, we obtain a binary cluster tree, where we continue until each leaf of the tree contains exactly one triangle. This is a rather simple construction but it proved effective in our experiments and is efficient to generate, which is important since we have to rebuild it for each evaluation.

We want to speed-up the evaluation of (6.2), hence we next consider pairs of clusters and build a *block cluster tree* $\mathbf{B} \subset \mathcal{P}(\mathbf{T}) \times \mathcal{P}(\mathbf{T})$. This tree is constructed recursively: Starting from the root (\mathbf{T}, \mathbf{T}) , an admissibility criterion—detailed below—is checked for the current cluster pair. If it is admissible it is kept as a leaf node, this means we can well approximate the contribution of this pair using one term. If it is not admissible, we need a better approximation and all possible pairs of child clusters of the two original clusters are added to the tree and checked for admissibility and split again if necessary. To determine the admissibility of a pair of clusters (U, V) , we check if

$$\max\{\text{diam}(U), \text{diam}(V)\} \leq \theta \text{dist}(\text{conv}(U), \text{conv}(V)), \quad (6.3)$$

for some $\eta \geq 0$, where $\text{diam}(U)$ denotes the diameter of the smallest sphere containing all embedded faces of U and $\text{conv}(U)$ their convex hull. This estimate allows controlling the error of the midpoint

quadrature of (6.1) and is typically called the multipole acceptance criterion (MAC). In practice, checking the MAC of the clusters U and V is too expensive to be performed repeatedly. Instead, we check the MAC of their bounding boxes because if they satisfy it then U and V do so as well. The parameter η allows a trade-off between accuracy and computational cost of the approximation. For $\eta = 0$, the criterion is only fulfilled for empty clusters. Thus the block cluster tree is built to its maximal depth and approximation below agrees with the original discretization since the leaf consists of all pairs of triangles. With increasing η the computational cost of the evaluation decreases as well as the approximation accuracy. In our experiments, already $\eta = \frac{1}{4}$ was sufficient to achieve satisfactory energy approximations. Finally, we denote by $LBC_\eta(\mathbf{S})$ the leafs of the block cluster tree constructed with the MAC (6.3).

These leafs form a cover of the discrete surface $\mathcal{S}_h \times \mathcal{S}_h$ such that all pairs satisfy the MAC or consist only of single triangles. Thus, we approximate the tangent-point energy by summing over all leafs, i.e.

$$\mathcal{T}_{h,\eta}^\alpha[\mathbf{S}] := \sum_{(U,V) \in LBC_\eta(\mathbf{S})} \frac{|\bar{P}(U)(\bar{X}(U) - \bar{X}(V))|^\alpha}{|\bar{X}(U) - \bar{X}(V)|^{2\alpha}} a_U a_V, \quad (6.4)$$

where a_U is the area of the cluster U immersed via the piece-wise affine immersion X , $\bar{X}(U)$ its barycenter, and $\bar{P}(U)$ the average of the orthogonal projectors onto the normals of the triangles contained in U . As for \mathcal{T}_h^α , we also write $\mathcal{T}_{h,\eta}^\alpha[X]$ whenever the connectivity is fixed. The approximation $\mathcal{T}_{h,\eta}^\alpha$ is not differentiable with respect to the immersion X since modifying X also modifies the block cluster tree. Hence, we approximate the derivative also using the multipole method to approximate $D\mathcal{T}_h^\alpha[\mathbf{S}]$, which means differentiating $\mathcal{T}_{h,\eta}^\alpha$ while keeping the block cluster tree fixed.

Adaptive Discretization. So far, neither the naive discretization of the tangent-point energy nor its approximation by the fast multipole method prevent self-intersections. To explain how this works, let us first take a more detailed look at the problem. Consider two non-neighboring triangles $\tau, \tau' \in \mathbf{T}$ whose immersions intersect but their barycenters do not agree. Then their discrete energy contribution $|\bar{X}(\tau) - \bar{X}(\tau')|^{-2\alpha} |P(\tau)(\bar{X}(\tau) - \bar{X}(\tau'))|^\alpha$ will be finite although possibly large. Even worse, it might even be zero if the barycenters lie in the plane of the respectively other triangle. The underlying issue is that we only use one quadrature point to approximate the integral of a (nearly) singular function. We will approach this problem by extending the block cluster tree beyond pairs of single triangles by refinement.

In other words, we use an adaptive quadrature that automatically introduces additional quadrature points on triangles close to contact. Formally, whenever a pair of triangles is a leaf of the block cluster tree and violates the MAC, we subdivide the triangles—using a standard 1-to-4 subdivision—and extend the block cluster tree using the resulting pairs. We do not actually refine the mesh since all information is already present and we only need the normal and barycenter of the new (virtual) triangles. Therefore, everything can be computed implicitly when evaluating the energy and thus requires only a small amount of memory. We repeat this until all leafs of the block cluster tree fulfill the MAC. This means that if two triangles intersect the block cluster tree will be extended to infinite depth. Thus, we have to exclude neighboring triangles from this refinement.

This strategy has the advantage that it is easy to implement given an implementation of the fast multipole approximation of \mathcal{T}_h . It has the downside that it becomes computationally very expensive if many subdivisions are required. To this end, we do not evaluate the MAC on axis-aligned bounding boxes of the triangles—potentially leading to unnecessary subdivisions—and instead compute exact minimal distances between triangles. This also allows us to set the energy to infinity whenever two triangles intersect and stop the evaluation. Moreover, we also use a parameter $\eta_{\text{near}} > \eta$ in the evaluation of the MAC for the refinement since a very accurate evaluation of the energy per (virtual) triangle is not necessary as long as the number of quadrature points increases when two triangles approach each other. In practice, we chose $\eta_{\text{near}} = 10$. Overall, we have constructed a discretization

that is efficient to evaluate for most meshes—by virtue of the fast multipole approximation—and truly repulsive in that it approaches infinity if the mesh approaches self-intersection—due to the adaptive refinement. With this at hand, we return to the shape framework next and discuss how to integrate the (discretized) tangent-point energy into it.

6.2 Shape Space Augmentation

Our goal is to introduce a Riemannian shape space, whose metric combines the discrete shell metric with the tangent-point energy. For shapes far from self intersection, it should resemble the space of viscous shells with a metric measuring viscous dissipation due to infinitesimal membrane distortion and bending. When approaching self contact, a component of the metric reflecting the infinitesimal increase of the tangent-point energy should become increasingly important. This concept would allow us to use the rich set of tools from Riemannian geometry, while guaranteeing self-intersection free shapes. For example, geodesics on this new space would lead to an intersection-free interpolation between shapes.

A key requirement is to incorporate the avoidance of self contact directly into the metric. In fact, following instead the naïve idea to simply add the penalty \mathcal{T}_h^α to the path energy in (3.7) one faces undesirable behavior: For start and end shapes that are in somewhat close contact, one would be perfectly happy with a trajectory that maintains a similar level of “closeness” throughout and thereby only moderate modulations of the actual geometry. Instead, the tangent-point energy as a penalty would try to push the intermediate shapes far away from itself at the expense of large surface deformations (Figure 6.2).

A first attempt for a Riemannian metric taking into account the preference for non self contact would be to choose the Hessian of the sum $\mathcal{W}[X, \cdot] + \mathcal{T}_h^\alpha[\cdot]$. However, \mathcal{T}_h^α is not convex and therefore this Hessian is not necessarily positive-definite. Hence, it is not admissible as a metric and in particular does not provide a meaningful notion of length and path energy for interpolating trajectories $c: [0, 1] \rightarrow \mathcal{M}$ in shape space.

Our below approach avoids these shortcomings. In the following, we will formulate it for a general potential energy $\mathcal{V}: \mathcal{M} \rightarrow \mathbb{R}_{\geq 0} \cup \{\infty\}$, for which the tangent-point energy will be the motivating example. In general, we assume that \mathcal{V} is continuous, especially $\lim_{x \rightarrow y} \mathcal{V}[x] = \infty$ if $\mathcal{V}[y] = \infty$. Then, we consider the feasible set

$$\mathcal{M}_+ := \{x \in \mathcal{M} \mid \mathcal{V}[x] < \infty\}. \quad (6.5)$$

We assume that \mathcal{M}_+ is an open submanifold of \mathcal{M} and that the potential energy is continuously differentiable on \mathcal{M}_+ . In case of the continuous tangent-point energy, \mathcal{M}_+ would be the space of embeddings and thus indeed an open submanifold of the space of immersions. The same approach we detail here could be applied to continuous and discrete shells alike.

Now, we want to modify the metric on \mathcal{M}_+ such that it includes a barrier at the boundary of \mathcal{M}_+ . To this end, we consider the graph of the potential energy \mathcal{V} over \mathcal{M}_+ . That is, we define

$$\mathcal{M}^\mathcal{V} := \{(x, \mathcal{V}[x]) \mid x \in \mathcal{M}_+\} \subset \mathcal{M}_+ \times \mathbb{R}_{\geq 0}. \quad (6.6)$$

We can picture $\mathcal{M}^\mathcal{V}$ as a mountain range over the underlying shape space, where the height of mountains corresponds to the amount of potential energy a given shape carries. However, let us emphasize that the base manifold \mathcal{M} is already nonlinear.



Figure 6.2: Using the time-integrated tangent-point energy for interpolation leads to unwanted deformations along the trajectory.

We can parametrize $\mathcal{M}^\mathcal{V}$ using $\Phi: \mathcal{M}_+ \rightarrow \mathcal{M}_+ \times \mathbb{R}_{\geq 0}$, $\Phi(x) = (x, \mathcal{V}[x])$. Since \mathcal{V} is continuously differentiable, the parametrization Φ is also differentiable and has an injective differential given at $x \in \mathcal{M}_+$ by

$$D_x \Phi u = (u, D_x \mathcal{V} u) \quad \text{for } u \in T_x \mathcal{M}_+.$$

In particular, Φ is a diffeomorphism of \mathcal{M}_+ onto $\mathcal{M}^\mathcal{V}$ —showing that $\mathcal{M}^\mathcal{V}$ is indeed a manifold—and we see that the tangent space of the graph manifold at $(x, \mathcal{V}[x])$ is given by

$$T_{(x, \mathcal{V}[x])} \mathcal{M}^\mathcal{V} = D_x \Phi(T_x \mathcal{M}) = \{(u, D_x \mathcal{V} u) \mid u \in T_x \mathcal{M}\}. \quad (6.7)$$

Thus, the dimension of $\mathcal{M}^\mathcal{V}$ equals the dimension of \mathcal{M} .

Next, we introduce a metric on $\mathcal{M}^\mathcal{V}$ to turn it into a Riemannian manifold. Let g be the metric on \mathcal{M} , in our case the viscous shell metric from Section 3.3. Then, the product metric \tilde{g} on $\mathcal{M}^\mathcal{V}$ induced via Φ is given by

$$\tilde{g}_{(x,r)}((u, s), (v, t)) := g_x(u, v) + \beta^2 s t,$$

for $(x, r) \in \mathcal{M}^\mathcal{V}$ and $(u, s), (v, t) \in T_{(x,r)} \mathcal{M}^\mathcal{V}$, where $\beta > 0$ is a weighting parameter for the contribution of the potential energy.

Our goal was to introduce a modified metric on \mathcal{M} . Hence, we now pull \tilde{g} back along the parametrization Φ to obtain a new metric on the original shape space \mathcal{M} :

$$\begin{aligned} g_x^\mathcal{V}(u, v) &:= \tilde{g}_{\Phi(x)}(D_x \Phi u, D_x \Phi v) \\ &= g_x(u, v) + \beta^2 (D_x \mathcal{V} u)(D_x \mathcal{V} v). \end{aligned} \quad (6.8)$$

In our case, this metric combines the existing metric based on viscous dissipation with the differential of the tangent-point energy.

Does this space achieve our goal? To answer this, we first observe the following

Lemma 6.5. *Let $x, y \in \mathcal{M}_+$ be two feasible points. As before, denote by $\text{dist}_{\tilde{g}}$ the Riemannian distance with respect to a metric \tilde{g} . Then*

1. $\text{dist}_{g^\mathcal{V}}(x, y) \geq \text{dist}_g(x, y)$,
2. $\text{dist}_{g^\mathcal{V}}(x, y) \geq \beta |\mathcal{V}[y] - \mathcal{V}[x]|$, and
3. *let $y \rightarrow z \in \mathcal{M}$ with $\mathcal{V}[z] = \infty$, then $\text{dist}(x, y) \rightarrow \infty$.*

Proof. Denote by \mathcal{C} the set of continuously differentiable curves $c: [0, 1] \rightarrow \mathcal{M}_+$ satisfying $c(0) = x$ and $c(1) = y$. We use the definition of the Riemannian distance and see

$$\begin{aligned} \text{dist}_{g^\mathcal{V}}(x, y) &:= \inf_{c \in \mathcal{C}} \int_0^1 \sqrt{g(\dot{c}(t), \dot{c}(t)) + \beta^2 (D_{c(t)} \mathcal{V} \dot{c}(t))^2} dt \\ &\geq \inf_{c \in \mathcal{C}} \int_0^1 \sqrt{g(\dot{c}(t), \dot{c}(t))} dt = \text{dist}_g(x, y), \end{aligned}$$

and similarly

$$\begin{aligned} \text{dist}_{g^\mathcal{V}}(x, y) &\geq \inf_{c \in \mathcal{C}} \int_0^1 \sqrt{\beta^2 (D_{c(t)} \mathcal{V} \dot{c}(t))^2} dt \\ &= \inf_{c \in \mathcal{C}} \int_0^1 \beta |D_{c(t)} \mathcal{V} \dot{c}(t)| dt \geq \beta |\mathcal{V}[y] - \mathcal{V}[x]|. \end{aligned}$$

From the last estimate, it immediately follows that $\lim_{y \rightarrow z} \text{dist}_{g^\mathcal{V}}(x, y) = \infty$ if $\mathcal{V}[z] = \infty$. \square

The last point, saying that points with infinite potential energy are pushed to infinite Riemannian distance implies that any geodesic in $(\mathcal{M}_+, g^\mathcal{V})$ will only consist of points with finite potential energy. In our applications, this means consisting of shapes without self-intersection. Furthermore, we can extend $\text{dist}_{g^\mathcal{V}}$ continuously from \mathcal{M}_+ to \mathcal{M} by setting it to be infinite whenever one of the arguments does not lie in \mathcal{M}_+ . When we will talk about the time-discretization of geodesic calculus on the augmented space and numerical methods for it, we will only require this distance and thus consider \mathcal{M} for notational convenience. Overall, we arrive at

Definition 6.6 (The space of repulsive shells). Let $(\mathcal{M}, g^{\mathcal{V}\nu})$ be the space of discrete shells and let \mathcal{T}_h^α be the discretized tangent-point energy with feasible manifold \mathcal{M}_+ . Then we call $(\mathcal{M}_+, g^\mathcal{V})$, where $g_x^\mathcal{V}(u, v) = g_x^{\mathcal{V}\nu}(u, v) + \beta^2(D_x \mathcal{T}_h^\alpha u)(D_x \mathcal{T}_h^\alpha v)$, the space of repulsive (discrete) shells with repulsion weighting parameter $\beta > 0$.

The idea to modify the metric as in our approach has also been studied, for example, by Gordon in 1973, who proved the following

Theorem 6.7 ([Gor73]). *Let (\mathcal{M}, g) be a connected, finite-dimensional Riemannian manifold of class C^3 and let $\mathcal{V}: \mathcal{M} \rightarrow \mathbb{R}$ be a proper function on \mathcal{M} of class C^3 . Then \mathcal{M} is complete with respect to the metric $g_x^\mathcal{V}(u, v) = g_x(u, v) + (D_x \mathcal{V} u)(D_x \mathcal{V} v)$.*

Remark. A continuous function f is called proper if $f^{-1}(K)$ is compact whenever K is compact.

However, this result does not directly apply to our situation. First of, its proof is only valid in finite dimensions since it relies on the Hopf–Rinow theorem. This means we could only apply it to the spatially discrete case. There we rely on an adaptive refinement (cf. Section 6.1) to achieve a truly repulsive energy—i.e. to actually achieve that the feasible manifold \mathcal{M}_+ only consists of embeddings—which is not C^3 . Hence, to apply this theorem, we would have to construct a sufficiently smooth approximation. For a related model in two dimensions, this has been investigated in [HL22] and we leave it for our model as potential future work.

Moving towards the time-discretization, recall that we have seen that geodesics can be defined variationally as minimizers of the path energy. For a path $c: [0, 1] \rightarrow \mathcal{M}$, the path energy with respect to the augmented metric $g^\mathcal{V}$ is given by

$$\begin{aligned} \mathcal{E}[c] &:= \int_0^1 g_{c(t)}^\mathcal{V}(\dot{c}(t), \dot{c}(t)) dt \\ &= \int_0^1 g_{c(t)}(\dot{c}(t), \dot{c}(t)) + \beta^2(D_{c(t)} \mathcal{V} \dot{c}(t))^2 dt. \end{aligned} \quad (6.9)$$

For the curve length $\mathcal{L}[c] := \int_0^1 \sqrt{g_{c(t)}^\mathcal{V}(\dot{c}(t), \dot{c}(t))} dt$, we again obtain by the Cauchy–Schwarz inequality $\mathcal{L}[c] \leq \sqrt{\mathcal{E}[c]}$ and equality holds if and only if the speed along the curve is constant, i.e. if $g_{c(t)}^\mathcal{V}(\dot{c}(t), \dot{c}(t)) \equiv \mathcal{L}[c]^2$. Thus, as before, minimizers of the path energy are indeed constant speed and length minimizing geodesics. With the path energy at hand, we can use all the tools introduced in Chapter 3 on our adapted shape space yielding shapes free of self-intersection.

6.2.1 Time-Discretization

By discretizing this path energy, we can define discrete geodesic interpolation and extrapolation following Section 3.2. To compute geodesics numerically, we first define for a discrete path $(x_0, \dots, x_K) \in \mathcal{M}^{K+1}$ the discrete path energy

$$E^K[x_0, \dots, x_K] := K \sum_{k=1}^K \mathcal{W}[x_{k-1}, x_k] + \beta^2 |\mathcal{V}[x_{k-1}] - \mathcal{V}[x_k]|^2, \quad (6.10)$$

where the first component of every summand reflects the discrete viscous shell energy and the second component discretizes the time-continuous second component of the energy in (6.9). This second component can also be seen as a finite difference approximation and provides a lower bound for the change of potential energy along the continuous path. Discrete geodesics are then defined as minimizers of this energy.

Consistency of the discrete and the continuous path energy follows from the general theory of time-discrete geodesic calculus developed by Rumpf and Wirth [RW15] and presented here in Section 3.2. In that respect, we note the following essential identity (cf. [RW15, Lemma 4.7]):

$$\begin{aligned} & \frac{1}{2} \partial_{22}^2 \left(\mathcal{W}[x, y] + \beta^2 (\mathcal{V}[x] - \mathcal{V}[y])^2 \right) \Big|_{y=x}(u, v) \\ &= \frac{1}{2} \partial_{22}^2 \mathcal{W}[x, y] \Big|_{y=x}(u, v) + \beta^2 \left((\mathcal{V}[x] - \mathcal{V}[y]) D_y^2 \mathcal{V}(u, v) + (D_y \mathcal{V} u)(D_y \mathcal{V} v) \right) \Big|_{y=x} \\ &= g_x^{\mathcal{V}\mathcal{V}}(u, v) + \beta^2 (D_x \mathcal{V} u)(D_x \mathcal{V} v) = g_x^{\mathcal{V}}(u, v). \end{aligned}$$

In fact, together with the smoothness of \mathcal{W} and \mathcal{V} , this identity implies the Γ -convergence of the discrete path energy (6.10) (extended to time continuous paths) to the continuous path energy (6.9) (cf. [RW15, Theorem 4.9]). This, in turn, implies the convergence of discrete geodesics to their continuous counterparts for $K \rightarrow \infty$ (cf. [RW15, Theorem 4.10]). In our application, this typically means that already coarse discrete geodesics, i.e. with a small number of time steps, are predictive of finer ones.

To perform extrapolation on our augmented shape space, we use the discrete counterpart of the time continuous exponential map and proceed iteratively as follows. At an initial point $x_0 \in \mathcal{M} \subseteq \mathbb{R}^{3|V|}$, we consider some initial velocity $v \in \mathbb{R}^{3|V|}$ and define $x_1 = x_0 + \frac{1}{K} v$. Next, given x_{k-1} and x_k , we consider the discrete path energy

$$E^2[x_{k-1}, x_k, x_{k+1}] = 2 \left(\mathcal{W}[x_{k-1}, x_k] + \mathcal{W}[x_k, x_{k+1}] + \beta^2 (\mathcal{V}[x_k] - \mathcal{V}[x_{k-1}])^2 + \beta^2 (\mathcal{V}[x_k] - \mathcal{V}[x_{k+1}])^2 \right)$$

for $K = 2$ steps. We search for x_{k+1} such that corresponding Euler–Lagrange equation, i.e. the necessary condition for x_k to be a minimizer of this energy for fixed x_{k-1} and x_{k+1} , is fulfilled. It is given by

$$0 \stackrel{!}{=} \partial_2 \mathcal{W}[x_{k-1}, x_k] + \partial_1 \mathcal{W}[x_k, x_{k+1}] - 2\beta^2 (\mathcal{V}[x_{k-1}] - 2\mathcal{V}[x_k] + \mathcal{V}[x_{k+1}]) D_{x_k} \mathcal{V} =: F_k[x_{k+1}]. \quad (6.11)$$

In fact, this is the k th equation of the set of Euler–Lagrange equations characterizing a discrete geodesic $(x_0, \dots, x_{k-1}, x_k, x_{k+1}, \dots, x_K)$ as a minimizer of the discrete path energy (6.10). Thus, given x_0 and x_1 , we consecutively solve these equations for $k = 1, \dots, K-1$. Thereby, in the k th step, we extend the already computed discrete geodesic $(x_0, \dots, x_{k-1}, x_k)$ by the next extrapolated point x_{k+1} . Finally, we define the discrete exponential map as before as $\text{Exp}_{x_0}^K v := x_K$. Following [RW15, Theorem 5.9], this discrete exponential map converges to the continuous one. In the k th step of this discrete geodesic extrapolation scheme, we have to compute the root of the function F_k . To compute these roots, we need the Jacobian

$$DF_k[x_{k+1}] = \partial_{21}^2 \mathcal{W}[x_k, x_{k+1}] - 2\beta^2 D_{x_k} \mathcal{V} \otimes D_{x_{k+1}} \mathcal{V}$$

which is a rank-one update of the derivative of the corresponding Euler–Lagrange operator on the viscous shape space. By construction, a discrete geodesic extrapolated this way solves the full system of Euler–Lagrange equations for a minimizer of the discrete path energy. Thus, if variationally defined discrete geodesics are unique then they coincide with extrapolated discrete geodesics starting from the first two points x_0 and x_1 with $v = \frac{1}{K}(x_1 - x_0)$. The inverse of the discrete exponential map is the discrete logarithm defined as $\text{Log}_{x_0}^K(x_K) := K(x_1 - x_0)$, where (x_0, x_1, \dots, x_K) is the unique minimizer of the discrete path energy for fixed x_0 and x_K .

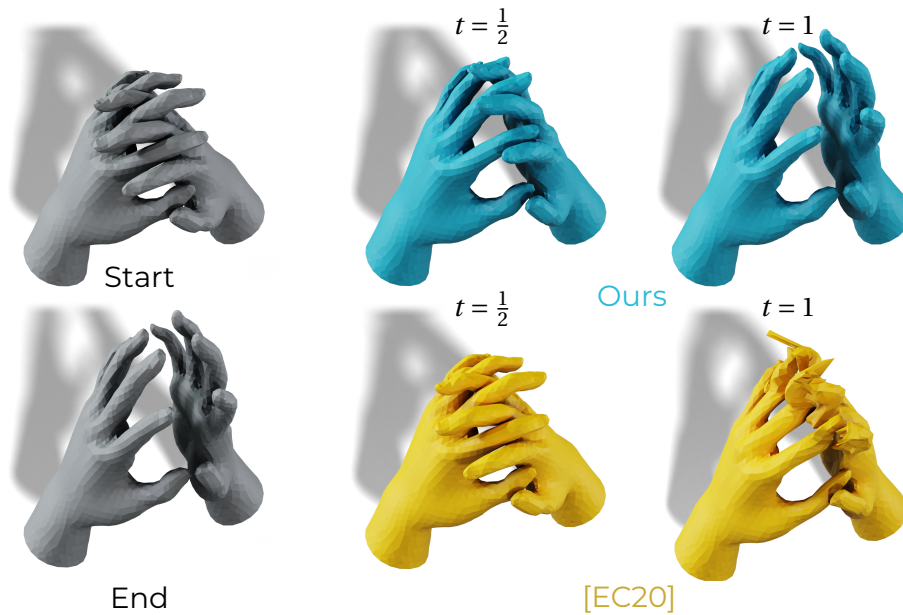


Figure 6.3: Methods based on ambient diffeomorphisms (here [EC20]) struggle with initial shapes that are close due to insufficient spatial resolution.

6.3 Implementation and Experiments

With the time-discrete model for the space of repulsive shells at hand, we now turn to the numerical optimization of the resulting variational problems. Furthermore, we will apply the approach to various inter- and extrapolation task. We will also investigate some other potential applications of the combination of elastic and repulsive energies in the variational computation of deformations beyond the shape space framework.

6.3.1 Numerical Optimization

Computing interpolating geodesics and evaluating the exponential maps leads to challenging numerical problems. So far, we have explained how to handle the non-locality of the tangent-point energy to evaluate it and its derivatives efficiently. Below, we will explain our algorithmic approaches to solving the nonlinear optimization problems. Since we focus on discrete shells in the remainder of the chapter, we will use discrete immersions $X \in \mathbb{R}^{3|V|}$ as variables in the optimization. This means we consider discrete paths $\mathbf{X} = (X_0, \dots, X_K) \in \mathbb{R}^{3(K-1)|V|}$.

Geodesics. To compute discrete geodesics, we use a trust-region method to minimize the discrete path energy (6.10). In such a method, we build a quadratic model of the objective function at the current iterate and then minimize this model in a region of limited size around this iterate, the trust-region. We use the Hessian of the discrete elastic path energy for the quadratic model combined with a Gauß–Newton approximation of the potential energy part, which can be seen as a non-linear least squares term. Concretely, this means we solve in each iteration $i \in \{1, \dots, i_{\max}\}$ the trust-region subproblem

$$\min_{p \in \mathbb{R}^{3(K-1)|V|}} E^K[\mathbf{X}^i] + DE^K[\mathbf{X}^i]^T p + \frac{1}{2} p^T B^i p \quad \text{s.t. } \|p\| \leq \Delta^i, \quad (6.12)$$

to obtain the new iterate $\mathbf{X}^{i+1} \in \mathbb{R}^{3(K-1)|V|}$ from the previous $\mathbf{X}^i \in \mathbb{R}^{3(K-1)|V|}$, where we refer to the elements of the discrete path in vectorized form. The quadratic form $B^i \in \mathbb{R}^{3(K-1)|V| \times 3(K-1)|V|}$ is given

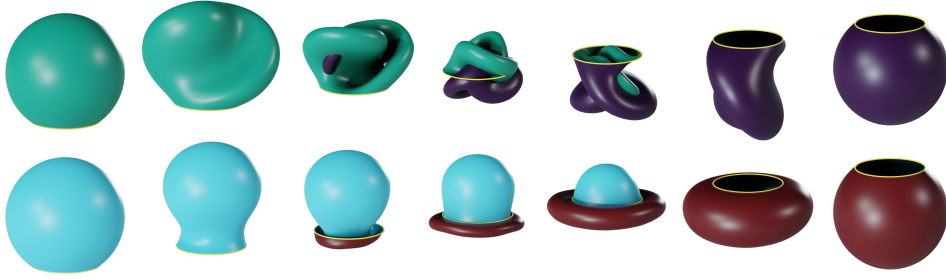


Figure 6.4: *Inversion of cut-open sphere. Top: Geodesic on space of discrete shells (green to purple). Bottom: Geodesic on space of repulsive discrete shells (blue to red). The fixed boundary of the surface is shown as a yellow curve.*

by $B^i = D^2 E_{\mathcal{V}}^K[\mathbf{X}^i] + DE_{\mathcal{V}}^K[\mathbf{X}^i]^T DE_{\mathcal{V}}^K[\mathbf{X}^i]$, where

$$E_{\mathcal{W}}^K[X_0, \dots, X_K] := K \sum_{k=1}^K \mathcal{W}[X_{k-1}, X_k], \text{ and}$$

$$E_{\mathcal{V}}^K[X_0, \dots, X_K] := K \sum_{k=1}^K \beta^2 |\mathcal{V}[X_{k-1}] - \mathcal{V}[X_k]|^2.$$

For the adaption of the trust-region size, we follow the approach described in [NW06, Alg. 4.1], which amounts to increasing or shrinking the trust-region based on the difference between the decrease predicted by the quadratic model and the actual decrease of energy.

To solve the quadratic—potentially indefinite—trust-region problem, we use Steihaug’s Conjugate Gradient (CG) method [Ste83]. It is an iterative method to find (approximate) solutions, extending the conventional CG method to handle indefinite problems with constraints on the norm of the solution. The iterative nature fits well for our problem as we can avoid assembling the dense matrices of the Gauß–Newton approximation, i.e. we only need to compute multiplications with B^i .

To make this efficient, we need to apply a preconditioner as is usual for CG methods. In the context of trust-region methods, this corresponds to considering a different metric for the subproblem, i.e. for our problem we consider

$$\min_{p \in \mathbb{R}^{3(K-1)|\mathcal{V}|}} E^K[\mathbf{X}^i] + DE^K[\mathbf{X}^i]^T p + \frac{1}{2} p^T B^i p \quad \text{s.t. } \|p\|_{P^i} \leq \Delta^i, \quad (6.13)$$

with $\|p\|_A := p^T A p$. In the algorithm, this boils down to one multiplication with the inverse of the preconditioner P^i per iteration similar to the elementary CG method. We use the L^2 -in-time elastic metric for this, i.e., the block-diagonal matrix $P^i \in \mathbb{R}^{3(K-1)|\mathcal{V}| \times 3(K-1)|\mathcal{V}|}$ with blocks $P_k^i = \partial_{22}^2 \mathcal{W}[X_k^i, X_k^i] \in \mathbb{R}^{3|\mathcal{V}| \times 3|\mathcal{V}|}$ for $k = 1, \dots, K-1$ on the main diagonal. Its inverse regularizes the discrete elastic path energy, which typically dominates the overall energy. While it is not the ideal preconditioner for the elastic part—which would be H^1 -in-time—it is easier to assemble and cheaper to invert as we can easily exploit its block diagonal structure.

Exponential Map. As explained above, we have to recursively solve (6.11), i.e., find roots of a non-linear function $F_k: \mathbb{R}^{3|\mathcal{V}|} \rightarrow \mathbb{R}^{3|\mathcal{V}|}$, to compute the discrete exponential map. For this, we use Newton’s method with line search using the Armijo acceptance condition. This means we have to invert the derivative $DF_k[X]$ in each iteration. As noted before, $DF_k[X]$ is the rank-one update of a mixed Hessian of our elastic energy. Hence, we can apply the Sherman–Morrison formula [SM50] and see that

$$DF_k[X]^{-1} = \partial_{21}^2 \mathcal{W}[X_k, X]^{-1} - \frac{\partial_{21}^2 \mathcal{W}[X_k, X]^{-1} (D_{X_k} \mathcal{V}) (D_X \mathcal{V})^T \partial_{21}^2 \mathcal{W}[X_k, X]^{-1}}{(D_X \mathcal{V})^T \partial_{21}^2 \mathcal{W}[X_k, X]^{-1} (D_{X_k} \mathcal{V})}.$$

This allows us to efficiently compute the update directions in Newton’s method.

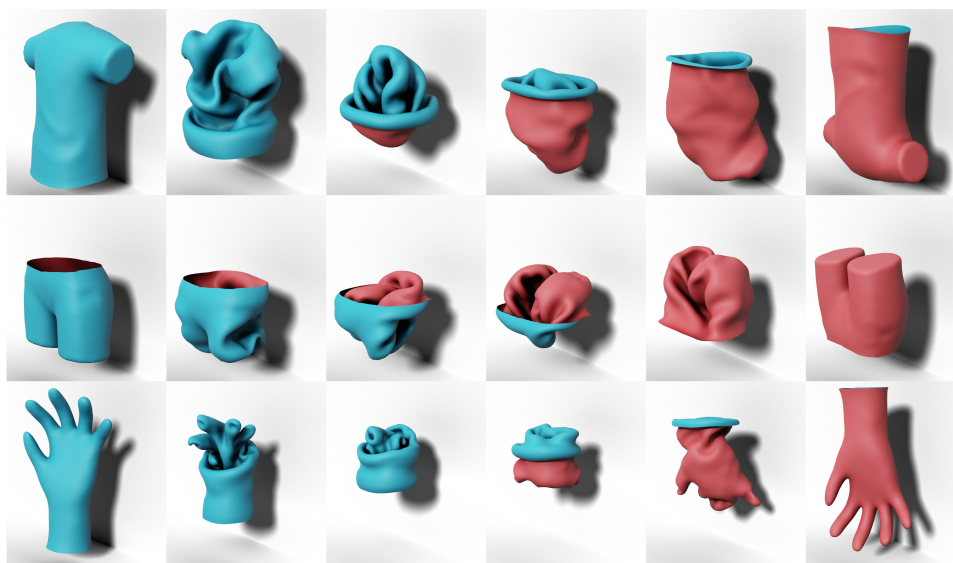


Figure 6.5: Turning clothes inside out. Geodesic interpolation between meshes of various clothes in original and flipped configuration (far left and right, respectively).

Multi-Resolution Scheme. We use the same multi-resolution approach as in Section 5.4 to speed-up computations while retaining the possibility to produce high-quality deformations. To this end, it is essential that we use the adaptive discretization of the tangent-point energy introduced in Section 6.1 to guarantee the prevention of self-intersections. The prolongation might introduce small intersections to the fine result if the coarse result is close to itself. Hence, if these are undesirable one has to apply some extra care, although we found that it does not affect the qualitative appearance of deformations in our experiments. We did not perform any further optimization after the prolongation.

6.3.2 Shape Space Results

In what follows, we show a variety of examples and applications for the interpolation and extrapolation on the Riemannian manifold of repulsive shells. The aim is to demonstrate central properties of our approach and illuminate how it might be useful for applications. Geodesic interpolation and extrapolation via the exponential map are fundamental tools on Riemannian shape spaces and can be used for various tasks in computer graphics. As we have seen in Chapter 3, one can derive more tools such as parallel transport for detail transfer [HRS+14], computation of Riemannian center of mass or nonlinear statistics of shapes based on these two. The presented approach allows to use all these tools while guaranteeing surfaces without self-intersection. As a proof of concept, here, we will demonstrate the application to shape interpolation and extrapolation.

Interpolation. Already Figure 6.1 shows that geodesics on the space of repulsive shells lead to intersection-free interpolation of shells. Furthermore, the path energy (6.9) propagates maintaining a constant level of tangent-point energy along the geodesic if both end shapes are on a similar level. In practice, this leads to interpolations that remain close to contact along the entire trajectory as is also visible in Figure 1.3. In this figure, we also see that an geodesic interpolation only using the viscous metric would lead to clear self-intersections along the trajectory.

Previous methods to ensure intersection-free interpolation typically rely on the discretization of ambient diffeomorphisms. This is a natural approach since self-intersection of surfaces is avoided if these surfaces are physical interfaces between bulk phases and the deformation of this bulk material

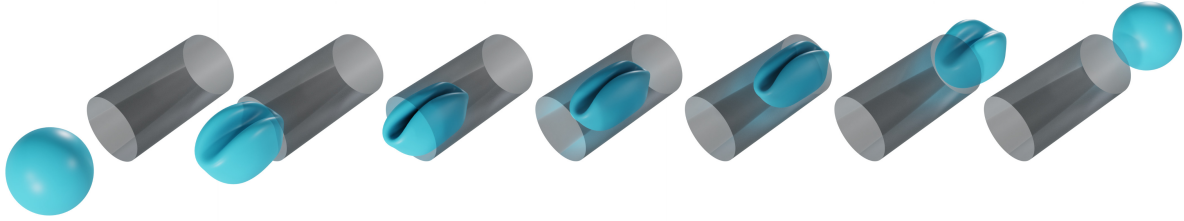


Figure 6.6: *Interpolation of translated sphere with obstacle. In the computations, there is a wall around the tunnel preventing the sphere from going around it, which is not shown in the rendering to better expose the deformation of the sphere.*

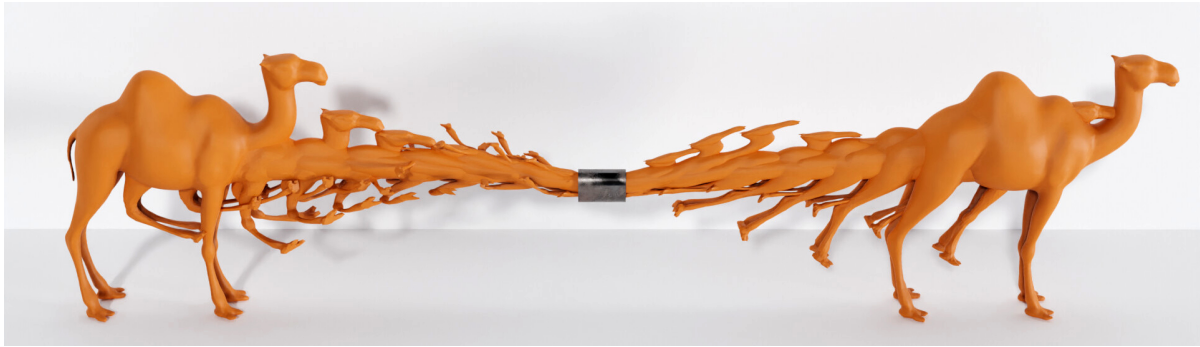


Figure 6.7: *A camel “walks” through the eye of a needle (cf. Matthew 19:23–24). Similar to Figure 6.6, there is a wall around the tunnel in the computations.*

either behaves elastic with a strong penalty on compression, or is controlled by a diffeomorphic flow. However, to work properly for shapes that are already in close contact a sufficiently refined discretization of ambient maps is indispensable, which is often times computationally infeasible. For example, we show a result using the method from [EC20]—which uses a low-rank approximation of ambient maps using a basis derived from the Laplace basis—at the highest resolution possible on a modern GPU with 40GB RAM compared to our solution in Figure 6.3.

One can also phrase shape inversion as an interpolation problem. Given a surface, one creates the end shape for a geodesic interpolation by flipping the sign of all dihedral angles and constructing the corresponding embedding using the method described in Chapter 4. As before, geodesic interpolation without the repulsive term in the path energy would typically lead to self-intersections. This is also the case for many other methods for this problem, e.g. [CKPS18], which only guarantee immersions. In contrast, a geodesic on our augmented shape space leads to a shape inversion free of intersections. We demonstrate this for the example of a cut-open sphere in Figure 6.4 and for a set of clothes turning inside out in Figure 6.5.

Kinetic Interpolation. So far, we have considered interpolations of static shapes, i.e. where we assume that there are no rigid body motions. In fact, because our path energy is invariant to rigid body motions, this was also a necessity to obtain a well-posed problem. We can add further terms to our path energy penalizing translations and rotations to allow for kinetic interpolations. For this purpose, we add to the discrete model

$$E_{\text{trans}}[X_0, \dots, X_K] = K \sum_{k=1}^K \|\bar{X}_k - \bar{X}_{k-1}\|^2, \quad (6.14)$$

which penalizes movement of the barycenters \bar{X} and

$$E_{\text{rot}}[X_0, \dots, X_K] = K \sum_{i=1}^K \left\| \sum_{v \in \mathbf{V}} (X_k(v) - X_{k-1}(v)) \times X_{k-1}(v) \right\|^2, \quad (6.15)$$



Figure 6.8: *Extrapolation on the space of repulsive shells. The tangent-point component of the metric leads to an evading behavior.*

which penalizes the angular momentum of the deformations of consecutive shapes. Recall that $X_k(v) \in \mathbb{R}^3$ is the position of the vertex v for the immersion $X_k \in \mathbb{R}^{3|V|}$.

With this, we can consider the interpolation of objects in motion. For example, in Figure 6.6, we interpolate between two translated spheres, one left and one right of a barrier with a hole in it. The barrier is incorporated into the energy via a tangent-point barrier energy (cf. [YBSC21]) as another potential following the approach described in Section 6.2. This leads to an interpolation of the sphere avoiding contact with the barrier and with itself. A similar setup makes a camel “walk” through the eye of a needle in Figure 6.7.

Extrapolation. Now, let us consider shape extrapolation as the second core application. Using the exponential map on the space of repulsive shells leads to self-contact avoiding behavior as seen in Figure 6.8. Note that the shape maintains a certain distance to itself, which stems from the fact that we use an energy modeling repulsion instead of contact. As before, the extrapolation with the exponential map coincides with the corresponding geodesic interpolation, cf. Figure 6.9 and Figure 6.1. Incorporating the tangent-point energy into the metric not only leads to the prevention of deformations that would lead to collisions, but also modifies the qualitative character of some extrapolations. For example, in Figure 6.10, it keeps the two cylinders close such that they twist around each other.

6.3.3 Elastic Deformations

The combination of elastic and tangent-point energy is also useful for applications beyond the shape space framework discussed so far. It models deformations of a shell repelling itself, which especially means that it avoids intersection, i.e. a guarantee to compute an embedding. In the following, we will consider the variational problem

$$\min_X \mathcal{W}[\tilde{X}, X] + \beta \mathcal{T}_{h,\eta}^\alpha[X], \quad (6.16)$$

where $\tilde{X} \in \mathbb{R}^{3|V|}$ is some given reference configuration, $\beta > 0$ a weight for the tangent-point energy, and we will add further terms based on the specific application.

Isometric Embeddings. One interesting application is the computation of an embedding for a given first fundamental form. The existence of such an embedding for an arbitrary metric is a famous question in mathematics. The Nash-Kuiper theorem [Nas54; Kui55] answers it to the affirmative in the class of C^1 -regular embeddings. However, for higher regularity, the question is still largely open. In fact, computing such embeddings is interesting for applications in natural sciences, engineering, and design. For example, low bending embeddings of a given hyperbolic metric might explain growth patterns of “frilly” plants in nature, see for example [YSS+21].

For a triangle mesh, as discussed in Section 2.2.1, the first fundamental form is piecewise constant and determined by the mesh’s edge lengths. Hence, we aim to compute an embedding of a triangle mesh with given edge lengths and minimal bending, i.e. minimal dihedral angles. We will



Figure 6.9: *Extrapolation of clasping hands. We show in gray the initial two shapes for the extrapolation derived from the example in Figure 6.1 and the result after $K = 18$ steps in blue.*

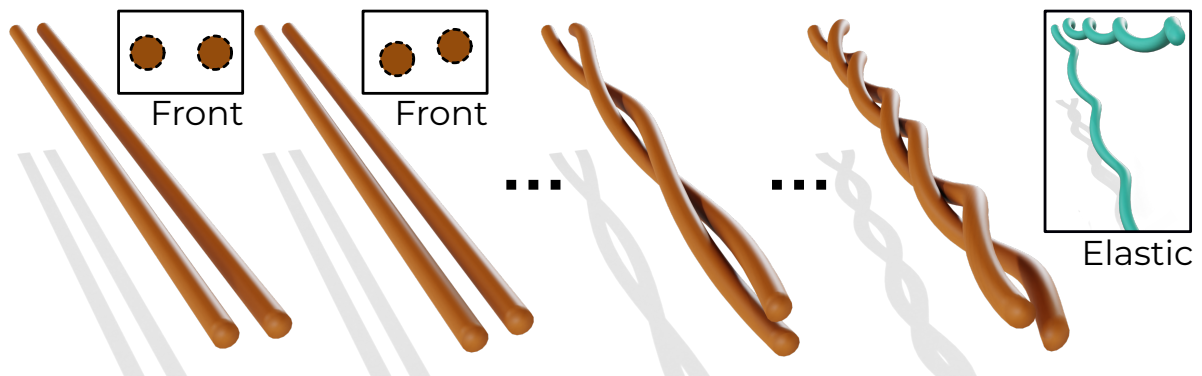


Figure 6.10: *Twisting two cylinders via extrapolation. The tangent-point component of the metric prevents that the two cylinders pass through each other and gain increasing distance. This would be the case for extrapolation using only the viscous metric, which is shown in green in the inset figure.*

relax the problem and include the prescription of edge lengths via the membrane energy with a high weight, which serves as a penalty term. However, the elastic energy of course only guarantees immersions and we are interested in embeddings. To this end, we use the tangent-point energy to ensure that we do compute embeddings. Furthermore, the tangent-point energy can also be viewed as a term regularizing the global curvature of the shape. We show the results of this approach for the aforementioned problem of computing an embedding of a part of the hyperbolic disk in Figure 6.11.

Fixing Intersections. So far, we applied our method to scenarios where we assumed the input data to be free of self-intersections. Yet, in many geometry processing task one wants to also work with data that *does* contain self-intersections. To this end, Yu *et al.* [YBSC21] observed that considering the tangent-point energy in the sub-repulsive regime, i.e. for an exponent $\alpha < 4$, still discourages overlap even though it does not have an infinite energy barrier to self-intersections. Thus decreasing it by taking a few descent steps could be used to remove unwanted self-intersections. However, only considering the sub-repulsive tangent-point energy leads to unwanted deformations all over the shape, cf. Figure 6.12 (middle). Hence, we instead consider the combination of elastic energy with sub-repulsive tangent-point energy as in (6.16), which improves the preservation of the overall shape, cf. Figure 6.12 (right).

Packing. Another application we consider is packing objects into tight spaces. To this end, we will add a barrier term to (6.16) that forces the surface to stay inside a given geometry. For example, we consider the barrier term $E_{\text{box}}[X] := \sum_{v \in V} \sum_{d=1}^3 (l_d - X(v)_d)^{-2} + (X(v)_d - u_d)^{-2}$ representing

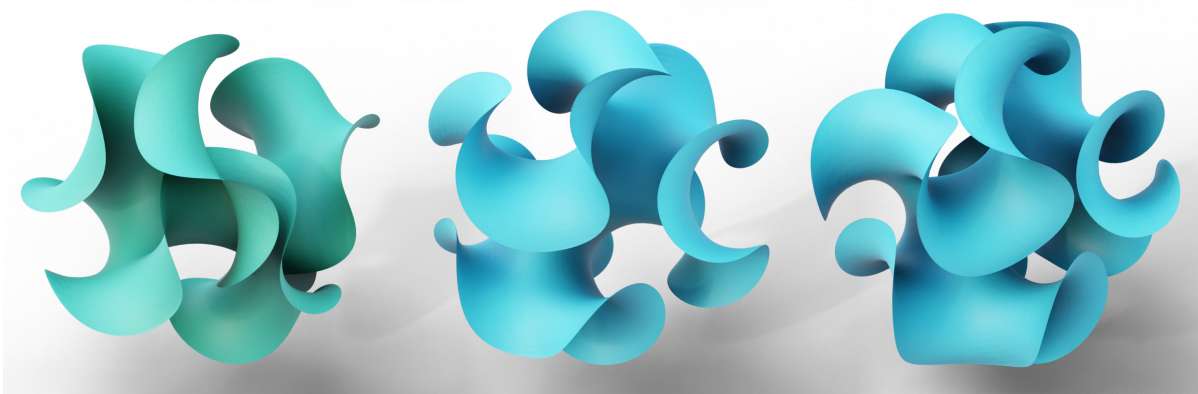


Figure 6.11: *Embedding of hyperbolic surface. Almost isometrically immersing a hyperbolic disk of radius 0.9 using only the elastic energy (green) quickly leads to self-intersections. Adding the tangent-point energy (blue, middle) leads to an embedding, i.e. avoids self-intersections, and allows to also consider larger radii such as 0.93 (blue, right).*



Figure 6.12: *Fixing intersections in a given mesh (gray, intersections in red) by decreasing the sub-repulsive tangent-point energy using a trust-region method (green) leads to undesirable deformations of the shape. In contrast, decreasing the combination elastic and sub-repulsive tangent-point energy (blue) allows preserving the overall shape while removing the self-intersection.*

a rectangular box, where $l \in \mathbb{R}^3$ are the coordinate-wise lower bounds of the box and $u \in \mathbb{R}^3$ the corresponding upper bounds. Then, in Figure 6.13, we reduce the size of the box step-by-step and thus get a tight packing of the shape into successively smaller boxes. Similarly, we consider a barrier term $E_{\text{ball}}[X] := \sum_{v \in \mathcal{V}} (r^2 - \|X(v)\|^2)^{-2}$ for a sphere with given radius $r \in \mathbb{R}_{\geq 0}$ and show a result for reducing this radius in Figure 6.15.

The resulting deformed shapes exhibit a space-filling quality in that they seem to fill the small cube evenly. This again highlights the repulsive nature of the tangent-point energy as it not only prevents self-contact but aims to maintain a distance of the surface to itself which is as large and even as possible. Combined with the other terms this leads to deformations that fill the box and keep as much of the original shape as possible. In contrast, methods that only consider contact to prevent self-intersections, such as [FLJK21], will stop the movement as soon as it reaches the boundary and thus do not have such a space-filling quality. Note that their approach also uses volumetric instead of shell elasticity.

6.4 Conclusion and Outlook

In this chapter, we have developed an approach to modify the metric on the space of discrete shells to push surfaces with self-intersections infinitely far away. This led to the *space of repulsive shells*. The key ingredient for this was the tangent-point energy, a nonlocal repulsive energy originating from the variational investigation of knots. This also entailed that the optimization problems stemming from the variational time-discretization of geodesic calculus on the space of repulsive shells

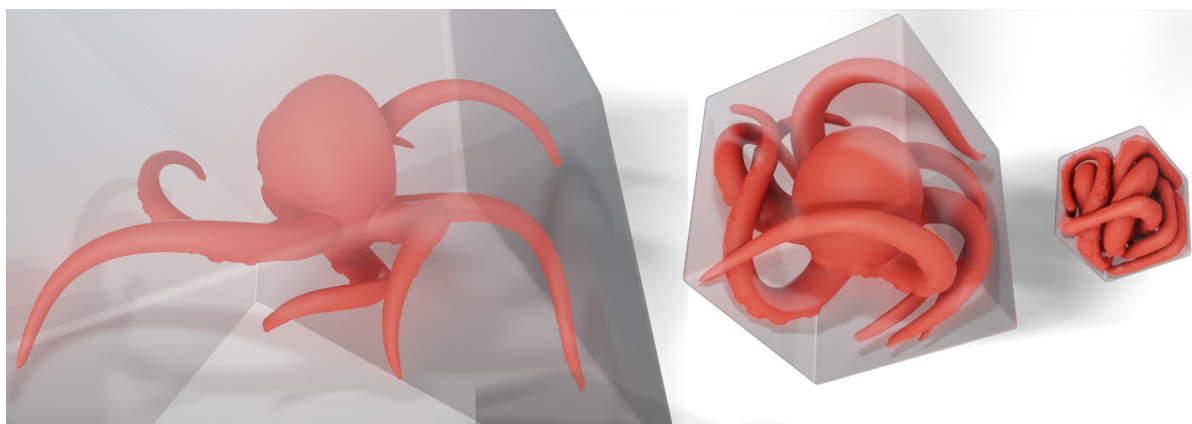


Figure 6.13: *Tightly packing an octopus into a shrinking box. By adding a barrier term to (6.16), we can constrain shapes to smaller and smaller boxes.*

involve nonlocal terms. To this end, we discussed, on the one hand, an efficient and adaptive discretization of the tangent-point energy and, on the other hand, introduced an affective approach to the numerical minimization based on a trust-region method. Through a variety of numerical experiments, we demonstrated that this leads to a useful tool for shape inter- and extrapolation that guarantees avoidance of self-intersections. This is—to the best of our knowledge—the first method for intersection-free shape interpolation that does not require the discretization of ambient space. Furthermore, we also shared a few examples that highlighted the usefulness of the combination of the tangent-point and the elastic energy beyond the shape space framework, which was also considered by Bartels, Meyer, and Palus [BMP22].

While the present discretization of the tangent-point energy is often quite efficient and reliably prevents self-intersections, it can become computationally expensive for surfaces where large parts are close to contact, for example, as in Figure 6.13. To circumvent this, it would be interesting to find a closed formula for the tangent-point energy of two triangles that are not in contact. Furthermore, it might be an option to use the closest points of the triangle pair as quadrature points to obtain a lower bound. From a general point-of-view, it would be great to enable the remeshing of individual steps in the computation of geodesics. Interesting deformations generated by minimizing the tangent-point energy (e.g. in [YBSC21, Figure 14]) incur severe membrane distortions that would lead to degenerate meshes without a proper remeshing approach. However, for the computation of geodesics, we need correspondences between these meshes, which are difficult to obtain. This touches on the general problem of constructing shape spaces that include immersions of different triangle meshes.

We would also like to construct a bending energy based on the tangent-point energy. To understand this, recall that the tangent-point energy is obtained by integrating the inverse tangent-point radii. For curves, the curvature functional is given by integrating the inverse radii of osculating circles and a bending energy is obtained by comparing them for different immersions—loosely speaking. We could proceed similarly with the tangent-point radii to obtain a nonlocal bending energy. However, for surfaces this likely becomes more involved since here second fundamental forms are compared in bending energies. Nevertheless, such a nonlocal bending energy would be interesting as it could further facilitate natural looking interpolation. In our current setup, shapes could move further apart in some points to become closer in others because we only compare the tangent-point energy after integration. A tangent-point bending energy would not allow this.

It would be interesting to combine other elastic models—e.g. of elastic curves—with the tangent-point energy. Finally, another important next step would be to generalize the model reduction approach from the previous chapter to the space of repulsive shells. Especially in the case of composite networks this could allow to efficiently parametrize submanifold of *embeddings* constructed from data.

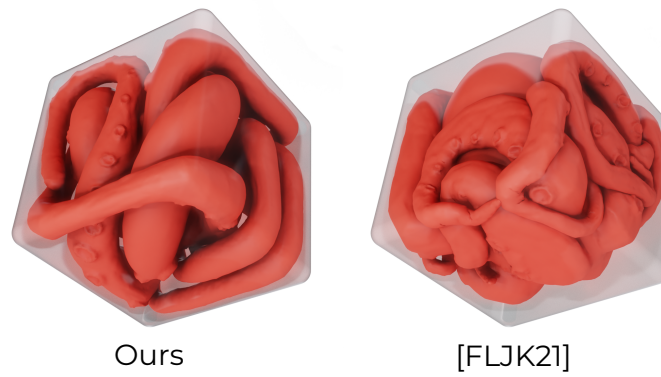


Figure 6.14: Comparison with [FLJK21]. Using a repulsive energy to avoid self-intersections yields a space-filling quality (left) that is not present for methods based on contact mechanics (right).

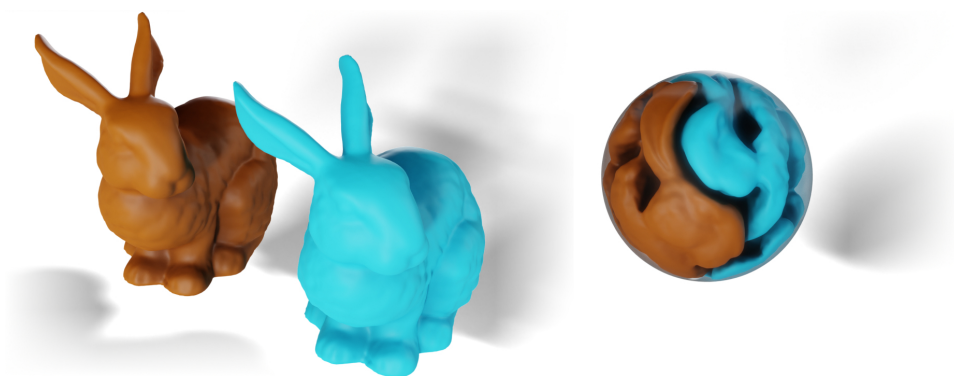


Figure 6.15: Two stanford bunnies packed into a sphere. The barrier approach to packing objects into tight spaces also works for multiple shapes at once and different packing geometries.

Part II

Shape Optimization

Chapter 7

Stochastic Bilevel Shape Optimization

We begin this second part of the thesis with studying a pessimistic stochastic bilevel optimization problem applied in elastic shape optimization. In essence, these are optimization problems of two (multi-dimensional) variables where one variable is constrained to be the minimizer of a second optimization problem parametrized by the other variable. Figuratively speaking, bilevel programs arise from the interplay of two decision makers on different levels of a hierarchy: The leader decides first and passes the upper level decision to the follower. Incorporating the leader's decision as a parameter, the follower then returns an optimal solution of the lower level problem. Bilevel optimization was originally introduced to analyze problems in economy-driven decision making and this remains the primary area of its application. However, in recent years, it has been applied to an increasingly diverse set of problems from other areas.

One such area is elastic shape optimization. So far, applications in this area have considered the physical deformation problem as the follower's problem. Here, we propose a new way to apply bilevel optimization in this setting. Different to previous approaches, we envision the follower taking the role of a test engineer developing load scenarios, which maximize the deformation of an elastic object. Furthermore, the leader takes the role of a design engineer controlling some design variables, such as material distribution, of the elastic object in question, while considering the deformation under the follower's load scenario and other design objectives. The design variables are perturbed stochastically in a pretend manufacturing process between the decisions of the leader and follower. Since the design engineer has to hedge against the worst possible load scenario devised by the test engineer, this leads to a pessimistic stochastic bilevel problem. We believe that this new approach to bilevel shape optimization can enable a new class of worst case optimization in geometric design. To this end, we develop a proof-of-concept application to a discrete shell model for architectural structures.

The rest of this chapter is organized as follows: First, we will begin with a brief overview of bilevel optimization—including its stochastic extension—in Section 7.1, where we will also list approaches related to ours. Then we will formulate the concrete bilevel problem we are interested in and summarize some theoretical results regarding it in Section 7.2. In Section 7.3, we will introduce our proof-of-concept application to discrete shells. Finally, we will discuss results of numerical experiments in Section 7.4 before drawing conclusions in Section 7.5.

Remark. This chapter is the result of joint work with Johanna Burtscheidt, Matthias Claus, Sergio Conti, Martin Rumpf, and Rüdiger Schultz published in [BCC+21].

7.1 A Glimpse of Bilevel Optimization

In this section, we will provide a quick introduction to bilevel optimization and its stochastic extension. The main goal will be to illuminate its central concepts and convey a basic understanding of the challenges that typically arise in the analysis of such problems. This will allow us to discuss the specific problem we are interested in later on. This section will be of course not a comprehensive introduction to the topic of bilevel optimization by any means. For this, we refer interested readers to the books [Dem02] and [DKPK15], which also served as basis for this section. Furthermore, we will also point to some of the central publications pertaining the ideas laid out below.

Bilevel optimization originates from work by Stackelberg [Sta34] on market structures in 1934. There it was formulated as a game with two players (or decision makers) called the leader and the follower. The leader makes their decision by minimizing an objective functional depending on the optimal decision of the follower. This decision of the follower is in turn influenced by the decision of the leader through modification of the follower's objective functional and feasible set. While the leader's problem also contains other conditions and objectives, the decision of the leader typically plays a central role. The optimization problem of the leader is then referred to as the *bilevel optimization problem*. The first mathematical formalization of this idea was developed by Bracken and McGill [BM73] in 1973 and has since been mainly used in economy-driven decision making and, recently, machine learning. For a detailed overview of possible applications, see also [SMD18]. We will introduce this formalization in its modern form.

We begin with the follower's problem, which reads in general as a common parametrized constrained optimization problem

$$\min_{f \in \mathcal{F}} \{j(u, f) \mid g(u, f) \leq 0\}, \quad (7.1)$$

with the follower's objective $j: \mathbb{R}^n \times \mathbb{R}^N \rightarrow \mathbb{R}$, constraints $g: \mathbb{R}^n \times \mathbb{R}^N \rightarrow \mathbb{R}^p$, and a closed set $\mathcal{F} \subset \mathbb{R}^N$ of admissible solutions. It is also called the *lower level problem*. We denote by $\Psi: \mathbb{R}^n \rightrightarrows \mathbb{R}^N$ the corresponding *solution set mapping*, i.e.

$$\Psi(x) := \arg \min_{f \in \mathcal{F}} \{j(u, f) \mid g(u, f) \leq 0\}, \quad (7.2)$$

and by $\psi: \mathbb{R}^n \rightarrow \mathbb{R}$ the *lower level optimal value function*, i.e. $\psi(u) := \min_{f \in \mathcal{F}} \{j(u, f) \mid g(u, f) \leq 0\}$. The notation $g: X \rightrightarrows Y$ is used for a multifunction g that maps the elements of some set X to subsets of some set Y . Hence, f is called the decision of the follower and as the next step we have to formalize how to compute the decision $u \in \mathbb{R}^n$ of the leader. Given the intuitive description from above, one is inclined to write the leader's problem as

$$\begin{aligned} & \underset{u \in \mathcal{U}}{\text{minimize}} && J(u, f) \\ & \text{subject to} && G(u) \leq 0 \text{ and } f \in \Psi(u), \end{aligned} \quad (7.3)$$

with the leader's objective $J: \mathbb{R}^n \times \mathbb{R}^N \rightarrow \mathbb{R}$, constraints $G: \mathbb{R}^n \rightarrow \mathbb{R}^q$, and a closed admissible set $\mathcal{U} \subset \mathbb{R}^n$. This is also called the *upper level problem* and would be the complete *bilevel optimization problem*. However, it is in general not well-defined, because the solution of the follower's problem is potentially non unique and this it is not accounted for in (7.3). To mitigate this issue, there are three commonly-used possibilities, which we will discuss next.

In the first possibility, the leader assumes that the follower is cooperating with them and thus is called optimistic. This means the leader can consider the follower's solution, which also minimizes their own objective. Formally, this yields the new objective $\Phi_o(u) := \min_{f \in \Psi(u)} J(u, f)$ and the *optimistic bilevel optimization problem*

$$\min_{u \in \mathcal{U}} \left\{ \min_{f \in \Psi(u)} J(u, f) \mid G(u) \leq 0 \right\}. \quad (7.4)$$

To show existence of solutions for this problem, a key ingredient is to show the lower semicontinuity of Φ_o . This is in turn implied by the *upper* semicontinuity of the solution set mapping Ψ , which is true for a wide range of problems (cf. [DKPK15, Theorem 3.3]). To derive corresponding optimality conditions, one typically transforms problem (7.4) into a one-level optimization problem. This introduces an extensive theoretical framework for the optimistic formulation.

However, in many applications, the optimistic assumption is not justified, because the leader can neither influence nor anticipate the follower's decision and thus can not rule out malicious actors. Hence, in this second possibility, the leader needs to consider the worst-case scenario, i.e. that the follower chooses the solution maximizing the leader's objective. Formally, this yields the new objective $\Phi_p(u) := \max_{f \in \Psi(u)} J(u, f)$, called *pessimistic solution function*, and the corresponding *pessimistic bilevel optimization problem*

$$\min_{u \in \mathcal{U}} \left\{ \max_{f \in \Psi(u)} J(u, f) \mid G(u) \leq 0 \right\}. \quad (7.5)$$

Our problem will fall into this class. Analysis of this problem is significantly more involved than for its optimistic counterpart. To obtain lower semicontinuity of Φ_p , one needs lower semicontinuity of the solution set mapping Ψ , which is rarely the case. Indeed, existence of solutions can only be assured under restrictive conditions on the follower's problem (cf. [LMP87]). In Section 7.2, we will see that also for our problem Φ_p is instead upper semicontinuous. One approach to guarantee the existence of solutions is to consider a relaxation of the problem, which is analyzed, for example, in the work by Loridan, Lignola, and Morgan [LM96; LM17; LM19]. An appropriate relaxation for our problem will be discussed in Section 7.2.

In the third and final approach, the leader is assumed to have a method to predict the follower's decisions. This means they have a function such that $f(u) \in \Psi(u)$ for all u and if we insert this in to (7.3), we obtain

$$\min_{u \in \mathcal{U}} \{ J(u, f(u)) \mid G(u) \leq 0 \}. \quad (7.6)$$

As $f(u)$ corresponds to the selection of a solution this formulation is called the *selection function approach*. Both, the pessimistic and the optimistic problem can also be regarded as special cases of this approach. However, this formulation is most commonly used when the solution of the follower problem is guaranteed to be unique. In this case, it is closely related to the typical formulation of PDE-constrained optimization problems, for which there exists a broad theoretical framework (cf. [HPUU09]). However, in general bilevel problems, $y(\cdot)$ does not need to be differentiable, and thus tools from nonsmooth analysis are required to study existence of solutions and optimality criteria. Nevertheless, we will adopt and justify this perspective in our numerical approach in Section 7.3 to enable the practical computation of solutions.

Among the many applications of bilevel optimization, there exist some works that study applications in the context of elastic shape optimization. For example, Herskovits et al. [HLDS00] investigate the design of an elastic object, where contact to a rigid obstacle supporting the object is only possible at certain parts of the domain. The corresponding contact problem to compute displacements forms the lower level and finding an optimal geometry of the elastic object under stress constraints forms the upper level. They assume the lower level problem to be convex and thus to have a unique solution, i.e. they consider the third of the approaches listed above. Kočvara and Outrata [KO95; Koč97] also consider a model where the follower problem amounts to computing the displacement under loads for which they assume the existence of a unique solution to devise a numerical approach. In their upper level problem, they optimize the design parameters of a truss structure. In [Zuo15], Zuo investigated an optimistic bilevel problem for car design. On both levels, design parameters are optimized: on the lower level, the mass distribution along the body frame and, on the upper level, the shape of shell segments of the hull. Finally, Sinha et al. [SMD18] recently presented a general overview on bilevel optimization, where especially a wide array of further applications can be found. To the best of our knowledge, the approach introduced in this chapter is the

first investigation of a pessimistic bilevel problem in shape optimization with objective functionals differing from the physical energy of the system on both levels.

Stochastic Extension. Finally, in the last part of this section, let us summarize the key ideas of stochastic bilevel optimization. In the pessimistic problem above, the leader hedged against uncertainty about the decision of the follower by considering a worst-case scenario. Nevertheless, one might be interested in including further uncertainty into the optimization that does not directly stem from the decision of the follower (or leader). When using bilevel optimization for modeling markets, this could, for example, arise from further actors that are not individually modeled in the optimization problems but their decisions are only considered as stochastic effects. Furthermore, when the bilevel problem is used in an engineering context as we will do in the next section, this uncertainty could arise from manufacturing errors. In stochastic bilevel optimization, this additional uncertainty is modeled using a random variable, which enters the upper and lower levels as further parameter. However, only the follower can observe the *realization* of this random variable while deciding. This means the leader has to make their the decision nonanticipatorily, while being aware of the distribution of the randomness, which is independent of the leader's decision.

To become more specific, we consider some probability space $(\Omega, \mathcal{B}, \mathbb{P})$, i.e. a set Ω with σ -algebra \mathcal{B} and probability measure \mathbb{P} , and let $Y: \Omega \rightarrow \mathbb{R}^k$ be a random variable on it, i.e. a \mathcal{B} -Borel measurable function. As explained, the follower observes a realization of this variable, i.e. a $v \in \text{Im } Y \subset \mathbb{R}^k$, when making their decision. This means it enters the follower's problem simply as another parameter, i.e. it still reads as a parametrized constrained optimization problem

$$\min_{f \in \mathcal{F}} \{j(u, f, v) \mid g(u, f, v) \leq 0\}, \quad (7.7)$$

with objective $j: \mathbb{R}^n \times \mathbb{R}^N \times \mathbb{R}^k \rightarrow \mathbb{R}$, constraints $g: \mathbb{R}^n \times \mathbb{R}^N \times \mathbb{R}^k \rightarrow \mathbb{R}^p$, and closed set $\mathcal{Y} \in \mathbb{R}^m$ similar to before. As before, we denote by $\Psi: \mathbb{R}^n \times \mathbb{R}^k \rightrightarrows \mathbb{R}^N$ the corresponding *solution set mapping*, i.e.

$$\Psi(x, v) = \operatorname{argmin}_{f \in \mathcal{F}} \{j(u, f, v) \mid g(x, f, v) \leq 0\}. \quad (7.8)$$

To formulate the leader's problem, we consider the pessimistic scenario and thus use the modified objective $\Phi_p(u, v) := \max_{f \in \Psi(x, v)} J(u, f, v)$, where $J: \mathbb{R}^n \times \mathbb{R}^N \times \mathbb{R}^k \rightarrow \mathbb{R}$ is the objective of the leader including the random parameter. Now, as explained before, the leader is not able to observe $\Phi_p(u, v)$ for individual realizations v and instead has to work with the real-valued random variable $\Phi_p(u, Y): \Omega \rightarrow \mathbb{R}$. One has to show that this random variable is indeed well-defined, which we will do for our problem in Section 7.2.2. To formulate the leader's problem, one has to incorporate this random variable into a new objective for the leader. This means one has to choose a risk measure, i.e. a map taking a random variable to a scalar value. Here, we consider the elementary choice of the expected value defined as $\mathbb{E}[X] := \int_{\Omega} X(\omega) d\mathbb{P}(\omega)$ for some real-valued random variable $X: \Omega \rightarrow \mathbb{R}$. We will also consider a class of possible alternatives, containing for example the expected excess, later on. With all these preliminaries in place, we finally arrive at the *pessimistic stochastic bilevel problem*

$$\min_{u \in \mathcal{U}} \left\{ \mathbb{E} \left[\max_{f \in \Psi(u, Y)} J(u, f, Y) \right] \mid G(u) \leq 0 \right\}. \quad (7.9)$$

In general, one can formulate this problem also in a way such that the constraints on x depend on the random variable Y and are required to hold almost surely. This can be transformed into a problem of the form (7.9) by introducing constraints depending only on u that guarantee the combined constraints hold for almost any realization v of the randomness. We will take this approach later when formulating our stochastic problem and thus consider here constraints depending only on u .

In a similar fashion, one can also introduce an optimistic stochastic bilevel problem. This is the way it was first formulated in [PW99] and, for example, also studied in [BCD20]. They study stochastic linear bilevel problems primarily in the optimistic regime but their results also extend to the pessimistic regime. Further studies of the linear case include [Iva18] and [DIN17]. Stochastic bilevel problems can be used to model complex system in a wide range of applications. This has, for example, been exploited for telecommunication networks [Wer05], in transportation science [Pat08], for option pricing [KP14], and supply chain planning [RSA07]. Closest to our problem are [CPW01] and [MHKP18], which study stochastic shape optimization of elastic structures using a truss resp. a level set model. However, they consider the problem of determining displacements under given forces as the lower level problem and use a linear elastic model, which means that it has a unique solution. In contrast, the follower will play the role of a test engineer in our scenario and optimize the forces applied to the elastic structure. In this case, the problem of determining the displacements could be considered as a third level. Nevertheless, we consider our problem to be a bilevel problem since this third level will have a unique solution given by a linear operator.

7.2 Bilevel Problem Formulation and Analysis

With the necessary preliminaries in place, we are now in the position to formulate our bilevel optimization problems. As already explained, our main motivation is the study of elastic shape optimization problems on discrete shells. However, in this section, we will formulate the bilevel problem in a more general fashion while foreshadowing how this connects to our elastic shape optimization. Furthermore, we will summarize the theoretical results about these general problems from [BCC+21]. We will begin with the deterministic problem and then move on to the stochastic extension.

7.2.1 Deterministic Problem

We will introduce our deterministic bilevel problem starting with some necessary objects, continuing with the lower level optimization problem, and then finally introducing the upper level problem. In this problem, we optimize two variables. On the one hand, we consider the parameters $u \in \mathcal{U} \subseteq (0, \infty)^n$, where \mathcal{U} is the nonempty closed set of admissible parameters. These parameters will describe the material in our application. On the other hand, we consider a variable $f \in \mathbb{R}^N$, which will later represent the forces acting on our elastic object. These forces lie in a nonempty, low-dimensional, convex, and compact set of admissible forces $\mathcal{F} \subset \mathbb{R}^N$.

Then, we consider a function $H: \mathbb{R}^n \rightarrow \mathbb{R}^{N \times N}$ such that the restriction $H|_{\mathcal{U}}$ is continuous and takes values in the cone of symmetric positive definite matrices S_{++}^N . In the elastic model, $H[u]$ will denote the Hessian of a nonlinear elastic energy depending on the material parameters u . Furthermore, we consider a fixed positive-definite matrix $M \in S_{++}^N$, which will be the mass matrix for the discrete reference shell. With this at hand, we can introduce a mapping $y: \mathcal{U} \times \mathbb{R}^N \rightarrow \mathbb{R}^N$, which will, in the application, describe the elastic displacement of the discrete shell with n triangular facets subject to a force distribution f from the set of admissible forces in $\mathcal{F} \subset \mathbb{R}^N$, with N being three times the number of vertices. It is defined by the condition

$$\{y[u, f]\} = \arg \min_{y \in \mathbb{R}^N} \left\{ \frac{1}{2} y^\top H[u] y - y^\top M f \right\}, \quad (7.10)$$

where uniqueness of minimizers follows from the positive-definiteness of $H[u]$. From the application point-of-view, this means $y[u, f]$ is defined as the minimizer of the total free energy of a linearized elasticity model with $H[u]$ denoting the stiffness matrix and M the mass matrix.

Now, we can define the lower level optimal solution set mapping $\Psi: \mathcal{U} \rightrightarrows \mathbb{R}^N$ by

$$\Psi[u] := \operatorname{argmax}_{f \in \mathcal{F}} \{y[u, f]^\top H[u] y[u, f]\}. \quad (7.11)$$

In our application, this set will contain the forces that achieve maximal compliance, i.e. maximal global displacements. Finally, we consider the pessimistic bilevel problem

$$\min_{u \in \mathcal{U}} \left\{ \max_{f \in \Psi[u]} J[u, f] \right\}, \quad (7.12)$$

where $J: \mathcal{U} \times \mathbb{R}^N \rightarrow \mathbb{R}$ denotes the cost functional of the leader, which we assume to be continuous. In our application, this will be a tracking type objective.

The above hierarchical problem (7.10) – (7.12) can also be understood as a three-level program. However, since the stiffness matrix $H[u] \in \mathbb{R}^{N \times N}$ is positive definite for any admissible material parameters $u \in \mathcal{U}$, the third-level problem in (7.10) has a unique solution. For this solution, we obtain the explicit representation

$$y[u, f] = H[u]^{-1} M f \quad (7.13)$$

using first-order optimality conditions. Plugging this solution into the lower level problem yields a bilevel problem. Moreover, (7.13) leads to a simple expression for the lower level optimal value function $\psi: \mathcal{U} \rightarrow \mathbb{R}$,

$$\psi[u] := \max_{f \in \mathcal{F}} \{f^\top M H[u]^{-1} M f\}. \quad (7.14)$$

and to the reformulation of the definition of Ψ in (7.11) as

$$\Psi[u] = \{f \in \mathcal{F} \mid f^\top M H[u]^{-1} M f = \psi[u]\}. \quad (7.15)$$

Now that our deterministic problem setup is complete, we will continue with summarizing the theoretical results from [BCC+21]. The first step is investigating if our problem as formulated above is well-posed and solvable. This leads us to our first

Proposition 7.1. *The mapping $\Phi: \mathcal{U} \rightarrow \mathbb{R}$ defined by*

$$\Phi[u] := \max_{f \in \Psi[u]} J[u, f]$$

is well-defined and upper semicontinuous. Moreover, Φ is continuous at any $u \in \mathcal{U}$ for which $\Psi[u]$ is a singleton.

To show this proposition, one first proves that the lower level optimal value function ψ defined by (7.14) is well-defined and continuous. Furthermore, one proves that the multifunction Ψ is closed. From this, it follows that $\Psi[u]$ is nonempty and compact for every $u \in \mathcal{U}$ and therefore Φ is well-defined by the continuity of J . Then the continuity follows from a straightforward computation.

Furthermore, one can construct examples (cf. [BCC+21] and [Dem02, example on pages 30-31]), where Φ arises as the objective function of a pessimistic bilevel program with a lower level problem that has more than a single optimal solution. In these examples, Φ is not lower semicontinuous and thus can not be expected to be in general. This may prevent the bilevel program (7.12) from having an optimal solution even if \mathcal{U} is compact. So, in general, we can not expect our deterministic problem to be solvable.

To overcome this difficulty, we consider a model where the leader also hedges against η -optimal lower level solutions, as proposed in [LM17]. To this end, we replace Ψ with the mapping $\Psi_\eta: \mathcal{U} \rightrightarrows \mathbb{R}^N$ defined by

$$\Psi_\eta[u] := \{f \in \mathcal{F} \mid \psi[u] - f^\top M H[u]^{-1} M f < \eta\}$$

for some positive constant η , i.e. we include parameters u , whose lower level objective is at most η worse than the optimal value. This results in the relaxed upper level problem

$$\min_{u \in \mathcal{U}} \left\{ \sup_{f \in \Psi_\eta[u]} J[u, f] \right\}. \quad (7.16)$$

As $\Psi[u] \subseteq \Psi_\eta[u]$ holds for any $\eta > 0$ and $u \in \mathcal{U}$, the optimal value in (7.16) yields an upper bound for the optimal value in (7.12). For this relaxed problem, we can prove the following

Proposition 7.2. *The mapping $\Phi_\eta: \mathcal{U} \rightarrow \mathbb{R}$ defined by*

$$\Phi_\eta[u] := \sup_{f \in \Psi_\eta[u]} J[u, f]$$

is well-defined and lower semicontinuous for any $\eta > 0$. In particular, (7.16) is solvable whenever \mathcal{U} is nonempty and compact.

Hence, we have relaxed our deterministic problem to obtain a solvable problem with similar behavior. There have been other similar relaxations of the pessimistic bilevel problem. For example, in [LM96], the alternate model

$$\min_{u \in \mathcal{U}} \left\{ \max_{f \in \bar{\Psi}_\eta[u]} J[u, f] \right\}$$

with

$$\bar{\Psi}_\eta[u] := \{f \in \mathcal{F} \mid \psi[u] - f^\top M H[u]^{-1} M f \leq \eta\}$$

is considered. Under the present assumptions, it can be shown that

$$\liminf_{\eta \searrow 0} \min_{u \in \mathcal{U}} \left\{ \max_{f \in \bar{\Psi}_\eta[u]} J[u, f] \right\} = \min_{u \in \mathcal{U}} \left\{ \max_{f \in \Psi[u]} J[u, f] \right\}.$$

However, the function

$$\bar{\Phi}_\eta[u] := \sup_{f \in \bar{\Psi}_\eta[u]} J[u, f]$$

is not lower semicontinuous in general, which is why we prefer the formulation (7.16).

7.2.2 Stochastic Problem

To model manufacturing inaccuracies, we will extend our deterministic bilevel problem to a stochastic one. Recall, that to obtain a stochastic bilevel program, we add random variables in the upper or lower levels. The realizations of these random variables are only observed by the follower. This means the leader has to make their decision under the constraint that they do not know the realization but only the distribution of the random variables, which is independent of the leader's decision.

In our case, the randomness is supposed to represent manufacturing inaccuracies of the elastic objects and enters the model in the following way. Let $Y: \Omega \rightarrow \mathbb{R}^n$ be a random vector on some probability space $(\Omega, \mathcal{B}, \mathbb{P})$. Then the leader's decision u is replaced by the point-wise perturbation $u \odot v$, where v is a realization of the random vector Y . As before, \odot denotes point-wise multiplication. This means realizations of Y describe, in our application, per-element relative errors of the material parameters occurring during the production.

The leader now seeks to ensure the admissibility of the resulting material parameters $u \odot v$, regardless of the specific realization. This entails that their set of admissible parameters is replaced by

$$\mathcal{U}_Y := \{u \mid u \odot v \in \mathcal{U} \text{ for all } v \in \text{supp } \mu_Y\},$$

where $\mu_Y := \mathbb{P} \circ Y^{-1}$ is the induced Borel probability measure on \mathbb{R}^n . This ensures that perturbed material parameters are almost surely admissible. Note that the set \mathcal{U}_Y is closed as the intersection of closed sets. Typically, we consider situations where $\text{supp } \mu_Y \subseteq [a, b]^n$ holds for some $0 < a < 1 < b$ and both close to one.

From the leader's point of view, the material vector that will be passed down to the lower level after the stochastic perturbation can be understood as a random vector $u \circ Y: \mathcal{U} \circ \Omega \rightarrow \mathbb{R}^n$ which is parameterized by their decision u . Similarly, the upper level outcome is a random variable $\Phi[u \circ Y] \in L^0(\Omega, \mathcal{B}, \mathbb{P})$ for any fixed u by Proposition 7.1. Here, and in the subsequent analysis, we denote the associated classical L^p -spaces with $p \in [1, \infty]$ by $L^p(\Omega, \mathcal{B}, \mathbb{P})$ and use $L^0(\Omega, \mathcal{B}, \mathbb{P})$ for the space of real-valued measurable functions. This allows us to define the random outcome $\mathbb{F}: \mathcal{U}_Y \rightarrow L^\infty(\Omega, \mathcal{B}, \mathbb{P})$ by

$$\mathbb{F}[u] := \Phi[u \circ Y],$$

which is well-defined and continuous as we will see in Theorem 7.4. To formulate the optimization problem for the leader, we thus need to convert it to a scalar assessment. For this, we use a mapping $\mathcal{R}: \mathcal{X} \rightarrow \mathbb{R}$, called risk measure, with

$$L^\infty(\Omega, \mathcal{B}, \mathbb{P}) \subseteq \mathcal{X} \subseteq L^0(\Omega, \mathcal{B}, \mathbb{P})$$

to allow for varying degrees of risk aversion. This leads us to the bilevel program

$$\min_{u \in \mathcal{U}_Y} \{\mathcal{R}[\mathbb{F}[u]]\}, \quad (7.17)$$

whose well-definedness and solvability will be investigated in Theorems 7.4 – 7.6. Typically, \mathcal{R} will be a so-called convex risk measure (cf. [FS11, Definition 4.1]) as introduced in the following

Definition 7.3 (Risk measures). A map $\mathcal{R}: \mathcal{X} \rightarrow \mathbb{R}$ on some subspace \mathcal{X} as before is called a *monetary risk measure* if

1. $\mathcal{R}[Y_1] \leq \mathcal{R}[Y_2]$ for all $Y_1, Y_2 \in \mathcal{X}$ satisfying $Y_1 \leq Y_2$ \mathbb{P} -almost surely, and
2. $\mathcal{R}[Y + m] = \mathcal{R}[Y] + m$ for all $Y \in \mathcal{X}$ and $m \in \mathbb{R}$.

It is called a *convex risk measure* if it also fulfills $\mathcal{R}[\lambda Y_1 + (1 - \lambda) Y_2] \leq \lambda \mathcal{R}[Y_1] + (1 - \lambda) \mathcal{R}[Y_2]$ for all $Y_1, Y_2 \in \mathcal{X}$ and $\lambda \in [0, 1]$.

Furthermore, a risk measure \mathcal{R} is called *law-invariant* if $\mathcal{R}[Y_1] = \mathcal{R}[Y_2]$ for all $Y_1, Y_2 \in \mathcal{X}$ with $\mathbb{P} \circ Y_1^{-1} = \mathbb{P} \circ Y_2^{-1}$.

One particular choice of such a convex risk measure is the expected value, i.e. a risk neutral assessment, which leads to the bilevel program

$$\min_{u \in \mathcal{U}_Y} \{\mathbb{E}[\mathbb{F}[u]]\}, \quad (7.18)$$

which is well-defined as we will already see in Theorem 7.4. This particular choice will also be of primary interest in our application to elastic shape optimization. However, also other popular choices for risk measures, such as the mean-upper semideviation of any order or the so-called Conditional Value-at-Risk fall into this category. In contrast, the expected excess, also a popular risk measure, is not translation invariant. Nevertheless, the results below still apply to it since this property is not necessary in their proofs. This means the following theoretical results apply to a range of widely-used risk measures.

Before we turn to the application, we will first summarize the results from [BCC+21] about the well-posedness and solvability of (7.17) justifying our approach. For this, we will consider the stochastic extension of the classical pessimistic bilevel program (7.10)–(7.12) as well as the relaxed version (7.16). In both situations, we will assume the following

(A1) The support of μ_Y is bounded.

In the classical setting, we will need the following additional assumptions:

(A2) \mathcal{F} is a nonempty, bounded polyhedron, i.e. the convex hull of its nonempty and finite set of extreme points $\mathcal{V} \subseteq \mathcal{F}$.

(A3) μ_Y is absolutely continuous with respect to the Lebesgue measure.

(A4) There exists an open and connected set $\tilde{\mathcal{U}} \subseteq \mathbb{R}^n$, such that $\mathcal{U} \subseteq \tilde{\mathcal{U}}$, $H|_{\tilde{\mathcal{U}}}$ is real analytical, and $H|_{\tilde{\mathcal{U}}}$ takes values in a closed subset of \mathcal{S}_{++}^N .

From these assumptions, one can prove the following central result.

Theorem 7.4. *Assume (A1)-(A4), then the mapping $\mathbb{F}: \mathcal{U}_Y \rightarrow L^\infty(\Omega, \mathcal{B}, \mathbb{P})$ given by*

$$\mathbb{F}[u] := \Phi[u \odot Y]$$

is well-defined and continuous with respect to any L^p -norm with $p \in [1, \infty)$.

To show it, one first proves that the set of discontinuities of Φ is a Lebesgue nullset, which is the main part of the proof. Then one shows point-wise convergence of \mathbb{F} for convergent sequences $\{u_k\}_{k \in \mathbb{N}}$ outside this null set and by dominated convergence this suffices to obtain continuity with respect to any L^p -norm.

This result already proves that (7.18) is well-defined and solvable. Next, we want to extend this to the general case (7.17) with various risk measures. This leads us to

Theorem 7.5. *Assume (A1)-(A4) and let \mathcal{R} be a convex risk measure. Then the function $\mathcal{Q}_{\mathcal{R}}: \mathcal{U}_Y \rightarrow \mathbb{R}$ defined by*

$$\mathcal{Q}_{\mathcal{R}}[u] := \mathcal{R}[\mathbb{F}[u]] = \mathcal{R}[\Phi[u \odot Y]]$$

is continuous. In particular, the bilevel stochastic problem (7.17) has an optimal solution whenever the induced feasible set \mathcal{U}_Y is nonempty and compact.

To show this, one proves that a convex risk measure \mathcal{R} is continuous by exploiting its monotonicity and convexity, see also [CL09, Theorem 4.1]. Then, Theorem 7.5 follows from Theorem 7.4.

Finally, let us consider the stochastic version of the relaxed problem (7.16), where the leader hedges against all η -optimal lower level solutions. For this, we will use the notion of a law-invariant risk measure as introduced in Definition 7.3, i.e. it agrees for all Y_1, Y_2 which induce the same Borel probability measure. The following existence result is obtained for law-invariant, convex risk measures under weaker assumptions, where we no longer restrict the analysis to polyhedral \mathcal{F} and real analytic H due to the hedging.

Theorem 7.6. *Assume (A1) and let \mathcal{R} be a law-invariant convex risk measure. Then the mapping $\mathcal{Q}_{\mathcal{R}, \eta}: \mathcal{U}_Y \rightarrow \mathbb{R}$ given by*

$$\mathcal{Q}_{\mathcal{R}, \eta}[u] := \mathcal{R}[\Phi_\eta[u \odot Y]]$$

is well-defined and lower semicontinuous. In particular, the bilevel stochastic program

$$\min_{u \in \mathcal{U}_Y} \{\mathcal{Q}_{\mathcal{R}, \eta}[u]\}$$

is solvable, whenever \mathcal{U}_Y is nonempty and compact.

This completes our summary of the theoretical results for the stochastic formulations of the stochastic extension of the classical pessimistic bilevel program (7.12) and the modified version (7.16). The results show that these programs are indeed well-defined and solvable and thus we will move on to our application.

7.3 Application to Discrete Shells

In this section, we will consider the application of our bilevel approach to an elastic shape optimization problem. From a high level perspective, we intend to establish the optimal elastic design of a complex roof-type construction. This allows us to develop a pictorial understanding of the roles of the leader and the follower. The leader in this context takes on the tasks of a construction engineer who aims to compute the distribution of material on a given roof geometry that minimizes, under certain constraints, the deviation from the prescribed shape of (parts of) the roof. Here, the deviation is measured via a tracking-type functional, i.e. the change of a subset of the geometry measured in a squared L^2 -norm. The determined material distribution is stochastically perturbed in the actual construction phase due to manufacturing inaccuracies before being handed to the follower. The follower is then considered to be a test engineer. Their task is to perform a worst-case load analysis and determine from a given set of possible forces — for example, load scenarios consisting of wind and gravitational loads — those that maximize the compliance, i.e. how much the object yields under the determined force.

For the geometry of the roof construction, we consider triangular surfaces, where each triangle is considered to be a construction panel which are joined together at common edges. We use triangular surfaces that have been previously studied in the literature on geometric design to find similar self-supporting structures [VHWP12]. Our goal is of course different and we model the mechanical properties of a roof construction using an adaptation of the discrete elastic shell model by Grinspun et al. from [GHDS03], which we have already discussed in Section 2.3. In this context, the membrane distortion describes the deformation of individual panels, while the bending distortion describes the deformation of the joints between neighboring panels. Let us emphasize that the discrete shell approach — although rooted in mathematical elasticity — does not fully and accurately describe the physics of such a construction. Therefore, we use it only as a design tool and not as a computational tool for the full elastostatic modeling, which would be necessary for later planning stages in practical applications. In fact, we consider the discrete shell model mainly as a proof-of-concept for the proposed bilevel optimization approach motivating further research into its application to physically accurate models. We underline this by reporting all physical quantities without units.

Comparing with the notation in the previous section, the design parameter u will represent the thickness of the shell, f the applied forces, and y the resulting displacement of the shell. The minimization in (7.10) then corresponds to the solution of a linear elasticity problem in (7.20), with $H[u]$ representing the stiffness matrix of the elastic energy. The problem in (7.11) corresponds to the follower maximizing the compliance. The leader's cost functional J in (7.12) measures the deviation from the prescribed shape and is defined in (7.22) below.

Discrete Shells. As explained before, we consider the sum of two terms for the elastic stored energy in the modeling of thin shells: the stored energy caused by in-plane membrane distortion and the stored energy reflecting bending distortion [Cia00; Lov92]. The two terms scale linearly and cubically, respectively, in the thickness of the shell. Previously, we have discussed and used a discrete shell model where this thickness was constant on the entire surface. In light of our goal to optimize the material distribution, we will introduce a slight modification allowing for a triangle-wise varying thickness.

For convenience, we quickly reiterate the essential elements of the construction with varying thickness. As before, we denote the mesh of a discrete shell by $\mathcal{S}_h = (\mathbf{V}, \mathbf{E}, \mathbf{T})$ consisting of vertices \mathbf{V} , edges $\mathbf{E} \subset \mathbf{V} \times \mathbf{V}$ and triangles $\mathbf{T} \subset \mathbf{V} \times \mathbf{V} \times \mathbf{V}$. Below, we will again use the notation relating vectors $w \in \mathbb{R}^{k|\mathbf{V}|}$ to maps $w: \mathbf{V} \rightarrow \mathbb{R}^k$ introduced in Section 2.2. Especially, we denote evaluations $w(v)$ of such a map also via indexing to simplify notation, i.e. $w_v := w(v) \in \mathbb{R}^k$. We denote by $\hat{X}: \mathbf{V} \rightarrow \mathbb{R}^3$ the fixed stress-free reference configuration, i.e. immersion, of the discrete shell. Then, we describe the deformed configuration $X = \hat{X} + y$ in terms of the displacement of the vertices $y: \mathbf{V} \rightarrow \mathbb{R}^3$. In

the following, we will denote by l_e the length of an edge $e \in \mathbf{E}$ and by a_τ the area of a face $\tau \in \mathbf{T}$ in the reference configuration and not in the deformed configuration as we did in previous chapters. The same holds for edge- and vertex-associated areas. Finally, the material thickness parameter is assumed to be piece-wise constant on triangles and is denoted by $u: \mathbf{T} \rightarrow (0, \infty)$. We define its evaluation on interior edges as the average of neighboring triangle values, i.e. $u_e := \frac{1}{2}(u_{\tau_l} + u_{\tau_r})$ for $\tau_l, \tau_r \in \mathbf{T}$ being the adjacent triangles of $e \in \mathbf{E}$.

For a displacement y , recall that we denote the Cauchy-Green strain tensor measuring the change of lengths, and consequently area, of a face τ by $\mathcal{G}[y]|_\tau$. Then, the membrane energy depends on this tensor and is defined as

$$\mathcal{W}_{\text{mem}}[u, y] := \sum_{\tau \in \mathbf{T}} a_\tau u_\tau W_{\text{mem}}(\mathcal{G}[y]|_\tau),$$

where we use the neo-Hookean energy density W_{mem} from Section 2.3.1. Recall, that the material parameters are chosen such that the linearization of this energy coincides with the planar, isotropic, linearized elasticity model with Lamé-Navier coefficients μ and λ [Cia90; Lov92]. As before, we use $\mu = \lambda = 1$.

For the bending energy, we again follow [HRS+14] and use an adaptation of the discrete shell bending energy introduced in [GHDS03]. It measures the change of the dihedral angles between neighboring triangles due to the displacement of the vertices. As before, the angle is denoted by $\theta_e(X)$ for an immersion X and the energy takes the form

$$\mathcal{W}_{\text{bend}}[u, y] := \gamma \sum_{e \in \mathbf{E}} u_e^3 \cdot \frac{(\theta_e(\hat{X} + y) - \theta_e(\hat{X}))^2}{a_e} l_e^2 \quad (7.19)$$

for some constant $\gamma > 0$, which in continuum models can be expressed in terms of λ and μ . We use $\gamma = 1$.

The stored elastic energy $\mathcal{W}[u, y]$ is the sum of these two energies,

$$\mathcal{W}[u, y] := \mathcal{W}_{\text{mem}}[u, y] + \mathcal{W}_{\text{bend}}[u, y],$$

where we consider the material parameters u as arguments as we will optimize them later on. The total free energy in the presence of external forces $f: \mathbf{V} \rightarrow \mathbb{R}^3$ reads then as

$$\mathcal{I}[u, f, y] = \mathcal{W}[u, y] - f^\top M y, \quad (7.20)$$

where M is the diagonal mass matrix in $\mathbb{R}^{3|\mathbf{V}| \times 3|\mathbf{V}|}$ with entries a_v at positions (i, i) with $i = 3j - k$ for $j = 1, \dots, |\mathbf{V}|$ and $k = 0, 1, 2$. The elastic displacements resulting from applying the forces to the reference configuration are the minimizers of this energy.

Linearization. In what follows, we will restrict ourselves to the linearization of this model. We denote by $H[u] := \partial_{yy}^2 \mathcal{W}[u, 0]$ the Hessian of the stored elastic energy at zero displacements, and obtain the linearized stored elastic energy

$$\mathcal{W}^{\text{lin}}[u, y] := \frac{1}{2} y^\top H[u] y,$$

as result of a second order Taylor approximation. Furthermore, we obtain the linearized total free energy

$$\mathcal{I}^{\text{lin}}[u, f, y] := \mathcal{W}^{\text{lin}}[u, y] - f^\top M y,$$

whose minimization corresponds to the innermost problem introduced in (7.10). Prescribing suitable boundary data $y_v = 0$ on a set of at least three vertices $v \in \mathbf{V}$, which do not lie on a line, one can deduce (cf. [HRS+14]) that $H[u]$ is a positive-definite matrix. This means, as explained before, that the energy $\mathcal{I}^{\text{lin}}[u, f, \cdot]$ has a unique minimizer for every u and f . It is determined as the unique solution of the associated Euler-Lagrange equation

$$0 = \partial_y \mathcal{I}^{\text{lin}}[u, f, y] = H[u] y - M f.$$

The Optimization Problem. To complete our practical optimization problem, we need to specify the admissible set of material parameters \mathcal{U} , the admissible set of force parameters \mathcal{F} , and the cost functional of the leader J . The objective of the lower level optimal value function ψ is already completely defined in (7.14) and equals the compliance functional evaluated for the displacement $y[u, f]$, i.e. we obtain for the optimal value functional

$$\psi[u] = \max_{f \in \mathcal{F}} \{f^\top M y[u, f]\} = \max_{f \in \mathcal{F}} \{f^\top M H[u]^{-1} M f\}.$$

The admissible set of force parameters \mathcal{F} is assumed to consist of linear combinations of a small number of different load scenarios. Therefore, we consider $B_j \in \mathbb{R}^{3|\mathbf{V}|}$ representing force distributions $B_j: \mathbf{V} \rightarrow \mathbb{R}^3$ on the reference configuration \hat{X} . The components of these vectors could be determined, for example, from the location of the vertex or the inclination of the triangular faces sharing a vertex. The load scenarios B_j are scaled with $F_j \in \mathbb{R}$ and subsequently added together. That means we assume that the forces are of the type $f = BF$, where $F \in \mathbb{R}^d$ for some $d \ll 3|\mathbf{V}|$ are the coefficients, and the columns B_j of the matrix $B \in \mathbb{R}^{3|\mathbf{V}| \times d}$ are the basis of a d -dimensional subspace of forces. Furthermore, we consider different constraints on the values of the scale factors F_j , i.e. we assume that the set \mathcal{F} is given by

$$\mathcal{F} := \left\{ BF \in \mathbb{R}^{3|\mathbf{V}|} \mid F \in \mathbb{R}^d, Q_k^F(F) \geq 0 \text{ for all } k = 1, \dots, K \right\} \quad (7.21)$$

for some smooth functions Q_k^F for $k = 1, \dots, K$. For example, if \mathcal{F} is to consist of the forces which fulfill $|f| \leq \mu$ then one might choose $d = 3|\mathbf{V}|$, $B = \text{Id}$, $K = 1$, and $Q_1^F(F) = \mu^2 - |F|^2$.

In the problem of the leader, we constrain the material thickness parameter u element-wise from below and above by positive constants u^- , and u^+ respectively. This models among other things the capabilities of the manufacturing process. Furthermore, we assume that the total volume of material, determined via the discrete integral of u , is below some fixed positive parameter V^+ , which models cost and weight constraints.

Lastly, the upper level cost functional is considered to be of tracking type and measures the squared discrete L^2 -norm of the displacement on a predefined tracking subset of the whole shell,

$$J[u, f] := y[u, f]^\top (\chi \odot M y[u, f]) = \sum_{v \in \mathbf{V}} \chi_v M_{vv} |y_v[u, f]|^2. \quad (7.22)$$

Here, $\chi: \mathbf{V} \rightarrow \{0, 1\}$ is a discrete characteristic function with value 1 at vertices in the tracking set and 0 elsewhere.

In the stochastic setting, we restrict ourselves to the expected value $\mathbb{E}[F[u]]$ as the risk measure for the optimization (cf. (7.18)). Furthermore, the stochastic perturbation of the distribution of the thickness parameter u is given by i.i.d. truncated normal distributions for each parameter. This means we consider the perturbed material $u \odot Y$ for $Y \sim \mathcal{TN}(1, \sigma^2, v_{\min}, v_{\max})^{|\mathbf{T}|}$, where $\mathcal{TN}(1, \sigma^2, v_{\min}, v_{\max})$ is the truncated normal distribution with average 1 and standard deviation σ , truncated to the interval $[v_{\min}, v_{\max}]$. We further fix constants $0 < u^- < u^+$ and $V^+ > 0$ and define \mathcal{U} implicitly by the condition

$$\mathcal{U}_Y = \left\{ u: \mathbf{T} \rightarrow \mathbb{R} \mid u^- \leq u_\tau \leq u^+ \text{ for all } \tau \in \mathbf{T}, \sum_{\tau \in \mathbf{T}} a_\tau u_\tau \leq V^+ \right\} \subset (0, \infty)^{|\mathbf{T}|}.$$

Barrier Formulation. To numerically solve the resulting bilevel problem (7.12), we replace the restrictions of u and f to admissible sets \mathcal{U}_Y and \mathcal{F} by smooth approximations in order to obtain a differentiable problem. To this end, we use logarithmic barrier functions, as commonly done in interior point methods (see e.g. textbook [NW06]). Hence, with the structural assumptions on the set of admissible forces introduced above, we define the smoothed follower problem by

$$\Psi_\alpha[u] := \arg \max_{F \in \mathbb{R}^d} \left\{ y[u, BF]^\top H[u] y[u, BF] + \alpha^F \sum_{k=1}^K \log(Q_k^F(F)) \right\}, \quad (7.23)$$

Algorithm 2 Stochastic minimization of smoothed bilevel problem**Input:** Initial material parameters u , Regularization parameters α , Maximal and minimal stepsizes t_{\max}, t_{\min} **Output:** Approximate minimizer of smoothed bilevel problem

- 1: Set $u^1 = u, t^0 = t_{\max}$
- 2: **for** $i = 1, \dots, i_{\max}$ **do**
- 3: Draw samples $v^1, \dots, v^K \sim \Upsilon$
- 4: **for** $k = 1, \dots, K$ **do** ▷ Solve follower problems
- 5: Solve (7.23) to compute $\Psi_\alpha[u^i \circ v^k]$
- 6: Compute $\partial_{FF}^2 j(u, \Psi_\alpha[u^i \circ v^k])^{-1}$ and $\partial_{uF}^2 j(u, \Psi_\alpha[u^i \circ v^k])$
- 7: Evaluate $\mathcal{J}[u^i] := \frac{1}{K} \sum_{k=1}^K J[u^i \circ v^k, \Psi_\alpha[u^i \circ v^k]] + \mathcal{R}_\alpha[u^i]$ ▷ Evaluation
- 8: Evaluate $d^i := -D_u \mathcal{J}[u^i]$
- 9: Set $\beta^i = \beta^{i-1}$ ▷ Line search
- 10: **if** $\mathcal{J}(u^i + \beta^i d^i) \leq \mathcal{J}(u^i) + 0.1\beta^i D_u \mathcal{J}(u^i)^T d^i$ **then**
- 11: **repeat**
- 12: $t^i = 2t^i$
- 13: **until** $\mathcal{J}(u^i + \beta^i d^i) > \mathcal{J}(u^i) + 0.1\beta^i D_u \mathcal{J}(u^i)^T d^i$ or $\beta^i \geq \beta_{\max}$
- 14: $\beta^i = \frac{1}{2}\beta^i$
- 15: **else**
- 16: **repeat**
- 17: $\beta^i = \frac{1}{2}\beta^i$
- 18: **until** $\mathcal{J}(u^i + \beta^i d^i) \leq \mathcal{J}(u^i) + 0.1\beta^i D_u \mathcal{J}(u^i)^T d^i$ or $\beta^i < \beta_{\min}$
- 19: Set $u^{i+1} = u^i + \beta^i d^i$

where $\alpha^F > 0$ is an appropriate scaling factor for the barrier terms.

To compute the minimizers in (7.23), we do not aim at identifying global minimizers, which would not be feasible in practical applications unless \mathcal{F} is a convex polyhedron. Instead, we use an ascent method (see below) to compute isolated local minimizers. Thus we assume in the numerical optimization of the leader problem, that the solution of the follower problem is of such type, which allows us to apply conventional nonlinear optimization algorithms. In this framework, we treat the maximizer and the set Ψ_α as interchangeable to simplify notation. In the examples considered below, this assumption is justified by the use of asymmetric triangulations and by the symmetry-breaking random perturbations of the material thickness. Thus, the logarithmic barrier formulation of the expected value optimization problem for the leader is

$$\min_{u \in \mathbb{R}^{\mathbf{T}}} \left\{ \mathbb{E}_\Upsilon [J[u \circ \Upsilon, \Psi_\alpha[u \circ \Upsilon]]] - \alpha^u \sum_{\tau \in \mathbf{T}} a_\tau (\log(u_\tau - u^-) + \log(u^+ - u_\tau)) - \alpha^V \log \left(V^+ - \sum_{\tau \in \mathbf{T}} a_\tau u_\tau \right) \right\}$$

for scaling factors $\alpha^u, \alpha^V > 0$ as before.

Numerical Optimization. Now that we have reformulated the optimization problems, let us discuss how we approach them algorithmically. In our case, the smoothed follower problem (7.23) is a deterministic and smooth optimization problem and computing its first and second derivatives is straightforward. Writing the problem as $\Psi_\alpha[u] = \arg \max_{F \in \mathbb{R}^d} \{j(u, F)\}$ with

$$j(u, F) := F^\top B^\top M H[u]^{-1} M B F + \alpha^F \sum_{k=1}^K \log(Q_k^F(F)), \quad (7.24)$$

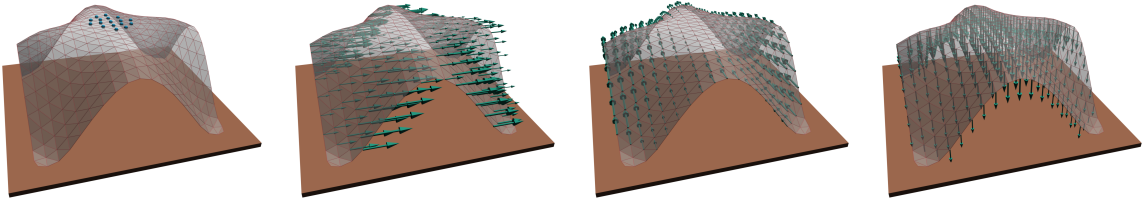


Figure 7.1: The first panel shows the geometry of a simple roof-type geometry, where the tracking set on the roof plateau is marked with blue dots. The Dirichlet vertices are the vertices on the horizontal plane at the corners. The other three panels show the three basis force fields B_1 (horizontal wind in the X direction), B_2 (horizontal wind in the Y direction) and B_3 (vertical gravitational force caused by an overlay on the roof). The scale of the force arrows is arbitrary but constant.

we obtain

$$\begin{aligned} \partial_F j(u, F) &= 2B^\top MH[u]^{-1}MBF + \alpha^F \sum_{k=1}^K \frac{\partial_F Q_k^F(F)}{Q_k^F(F)}, \text{ and} \\ \partial_{FF}^2 j(u, F) &= 2B^\top MH[u]^{-1}MB + \alpha^F \sum_{k=1}^K \frac{Q_k^F(F) \partial_{FF}^2 Q_k^F(F) - (\partial_F Q_k^F(F))(\partial_F Q_k^F(F))^\top}{Q_k^F(F)^2}. \end{aligned} \quad (7.25)$$

With this at hand, we can use a Newton-type method with Armijo backtracking line search (cf. [NW06, Algorithm 3.2]) to compute maximizers of $j(u, \cdot)$.

Next, we want to apply a stochastic gradient method to compute minimizers for the smoothed bilevel problem. To this end, we need to compute the gradient of the objective. For the most part, this is straightforward with the chief difficulty of computing the derivative of the optimizer $\Psi_\alpha[u]$ of the smoothed follower problem w.r.t. the material parameters u . Since we assume that the minimizers of the smoothed follower problem are (locally) unique, we can proceed by the general procedure of shape optimization calculus. This means that we obtain the derivative of $\Psi_\alpha[u]$ by applying the implicit function theorem to the first order optimality conditions $\partial_F j(u, F^*) = 0$ for an maximizer F^* to obtain

$$\partial_u \Psi_\alpha[u] = \partial_{FF}^2 j(u, F^*)^{-1} \partial_{uF}^2 j(u, F^*). \quad (7.26)$$

Therefore, we also compute

$$\partial_{uF}^2 j(u, F) = 2B^\top MH[u]^{-1}(\partial_u H[u])H[u]^{-1}MBF. \quad (7.27)$$

Due to the small dimension d of the space of forces, all these derivatives can be evaluated efficiently.

Finally, we employ stochastic gradient descent [RM51] to compute approximate minimizers. In each iteration of the descent algorithm, we draw finitely many samples v^1, \dots, v^K from the distribution of the material perturbation. Using these samples, we approximate the expected value by the empirical risk $\hat{J}[u] := \frac{1}{K} \sum_{k=1}^K J[u \odot v^k, \Psi_\alpha[u \odot v^k]]$. Then a new iterate is computed by taking a step in the direction of the negative gradient of the combination of the empirical risk and the logarithmic barrier terms. For notational convenience, we collect the barrier terms in a functional \mathcal{R}_α , i.e. $\mathcal{R}_\alpha[u] := -\alpha^u \sum_{\tau \in \mathbb{T}} a_\tau (\log(u_\tau - u^-) + \log(u^+ - u_\tau)) - \alpha^V \log(V^+ - \sum_{\tau \in \mathbb{T}} a_\tau u_\tau)$. To determine the stepsize, we again use an Armijo backtracking line search. Overall this leads to Algorithm 2.

7.4 Experimental Results

With all the ingredients in place, we will investigate our bilevel shape optimization method in a range of numerical experiments. To this end, we will consider discrete shells representing curved roofs and devise corresponding load scenarios. Let us note again, that we only consider this to be a

Example	Dimensions	Forces		Material			Randomness		
		$F_{\max,xy}$	$F_{\max,z}$	u^-	u^+	V^+	σ	v_{\min}	v_{\max}
Figure 7.2	$20 \times 20 \times 10$	0.0015	0.003	0.01	0.2	60	0.1	10^{-2}	2
Figure 7.7 (top)	$20 \times 20 \times 5$	0.005	0.01	0.01	0.2	50	0.05	10^{-2}	2
Figure 7.7 (btm)	$70 \times 70 \times 15$	0.005	0.01	0.01	0.2	330	0.05	10^{-2}	2

Table 7.1: Parameters for examples.

proof-of-concept study and more research needs to be conducted on applications to more realistic models. We will begin with introducing our setup in more detail, then discuss a series of parameter studies on a simple roof-type geometry, and finally show results on more complex examples.

Setup. For each geometry, we fix an orientation so that the negative Z -axis is in the direction of gravity and the supporting ground is in the XY -plane. Then, we select a set of Dirichlet vertices near the ground plane, representing the points on which the structure is supported. Furthermore, we also fix the thickness of the corresponding triangles, removing these variables from the optimization.

The construction is exposed to two type of forces. First, there are forces emulating wind hitting the structure. For a given wind direction and strength, the force on each part of the roof depends on the local orientation. We assume that the magnitude of the force on a vertex is proportional to the absolute value of the scalar product between the vertex normal and the wind direction. For simplicity, we only consider a two-dimensional subset of possible forces, spanned by the basis vectors B_1 and B_2 which represent wind along the positive X - and Y -axis, respectively. In formulas, this reads as $B_{1,v} := |\hat{N}_{v,1}|e_1$ and $B_{2,v} := |\hat{N}_{v,2}|e_2$. The direction and magnitude of the wind is then controlled by the scale factors F_1 and F_2 . We fix a maximal magnitude $F_{\max,xy}$ of the wind-type force and use the constraint function $Q_1^F(F) := F_{\max,xy}^2 - (F_1^2 + F_2^2)$ in (7.23). An example of these two basis vectors demonstrating the dependence on the orientation of the normal is shown in the second and third panel of Figure 7.1. Second, we consider a vertical force, which could emulate the weight of snow or rain overlay on the roof. The magnitude of the corresponding basis vector B_3 on each vertex is the absolute value of the scalar product between the vertex normal and the Z -axis, i.e. $B_{3,v} := -|\hat{N}_{v,3}|e_3$, and is shown in Figure 7.1 on the far right. The magnitude of gravitational load is controlled by the scale factor F_3 . We ensure that it is pointing downward via $Q_2^F(F) := F_3$. Additionally, we limit its magnitude via $Q_3^F(F) := F_{\max,z} - F_3$, where $F_{\max,z}$ is the maximal magnitude of the gravitational force. Therefore the admissible set \mathcal{F} is a cylinder with radius $F_{\max,xy}$ and height $F_{\max,z}$.

We performed most of our investigations on the simple roof-type geometry shown in Figure 7.1. For this problem, the basic parameters, which are used in the examples if not indicated otherwise, are listed in Table 7.1. For the truncation of the normal distribution, we chose v_{\min} and v_{\max} , so that the truncation has little effect and σ is almost identical to the standard deviation of Υ . The weights of the barrier terms were $\alpha^F = 10^{-4}$, $\alpha^u = 1$, and $\alpha^V = 10^{-5}$. For the leader, we consider a tracking set restricted to the central region of the roof plateau as shown in the first panel of Figure 7.1. Finally, we chose $K = 128$ samples for the stochastic gradient descent.

Tracking Set. We begin with comparing the results of the leader minimizing a tracking functional once with global support using $\chi \equiv 1$ and once restricted to the region on the roof plateau. In the first case, the leader and follower essentially consider the same objective up to a weighting by the Hessian of the elastic energy. However, in the second case, their objectives substantially deviate from each other and thus we expect to observe effects of our bilevel approach in this case. To this end, we show the deformed configuration, the optimized distribution of the material thickness, and the magnitude of displacements for both cases in Figure 7.2. As for all examples presented here, in the follower problem, the maximal compliance is attained for a force F representing an extremal point

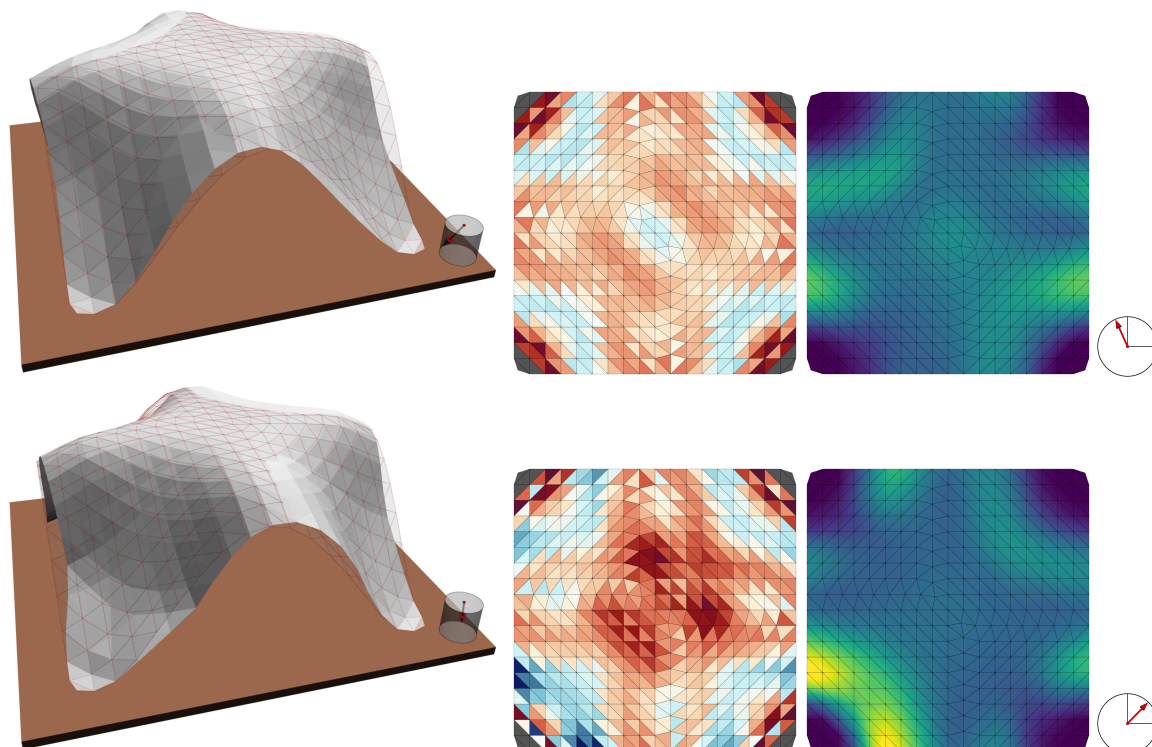


Figure 7.2: Comparison of results for full vertex tracking set (top) and plateau tracking set (bottom) on the simple roof-type geometry already shown in Figure 7.1. On the left, we show the deformed configuration as a gray surface, while the undeformed surfaces is shown as a translucent surface overlaid with red edges. Next to the surfaces, we visualize the direction of the force (F_1, F_2, F_3) chosen by the follower in the cylinder of admissible values. In the middle, we show the resulting material distributions with color map $0 \rightarrow 0.2$, where boundary triangles with all three vertices subject to Dirichlet boundary conditions are shown in gray. On the right, we show the magnitude of the deformation y using the color map $0 \rightarrow \geq 1.5$. Additionally, on the far right, we show the direction of the horizontal forces (F_1, F_2) .

of the cylinder of admissible forces. In the example with localized tracking, and most of the following ones, the element-wise bounds u^+ and u^- are nearly attained for at least some triangles.

In case of the tracking set centered on the roof plateau, one observes a concentration of mass in the center region accompanied by a significant reduction of the thickness close to the four corners where Dirichlet boundary conditions apply. The concentration and corresponding reduction break the symmetry of the configuration w.r.t. the diagonal from the upper left to the lower right. Due to this asymmetric reduction, the follower chooses a force pointing to the upper right and one observes a crease line connecting the two arcs in the front at approximately half of the total height. This is accompanied by large displacements, which are however outside of the tracking region on the plateau and therefore do not affect the leader's objective. In contrast, for the tracking with global support, no such crease with strong displacements occurs. Nevertheless, the deformation exhibits a larger displacement in the central region. Finally, beyond the mass concentration in the middle one also observes the onset of curved beam-like structures connecting the middle region and the four side-arcs of the roof.

Parameter Studies. We now further investigate the influence of the different parameters in our setup. Figure 7.3 shows for the same geometry the impact of the upper bound on the total material volume. As the total permitted mass is increased, the elongated curved “beams” connecting

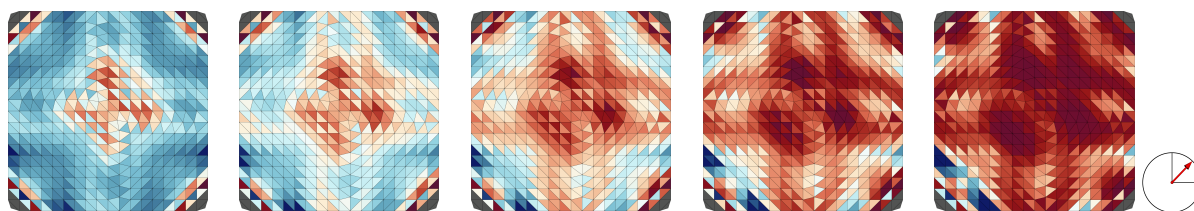



Figure 7.3: A comparison of the material distribution when varying the maximal allowed material volume V^+ while keeping the other parameters fixed. The allowed volume was $V^+ = 40, 50, 60, 70, 80$ from left to right. Material thickness is shown using the color map 0  0.2. On the far right, we show the direction of the horizontal forces, which was the same for all parameters, while the vertical force was always chosen maximally.

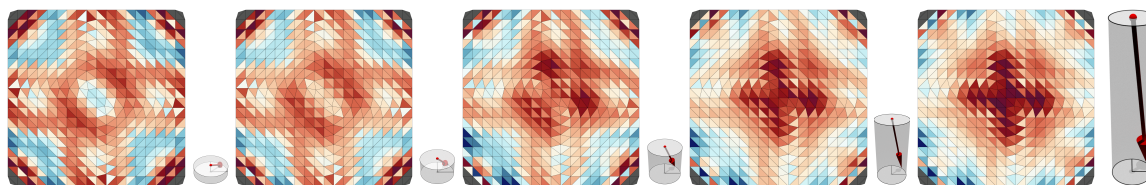



Figure 7.4: A comparison of the material distribution when varying the ratio of vertical to horizontal force $\frac{F_{\max,z}}{F_{\max,x,y}}$, i.e. the shape of the cylinder, while keeping the other parameters, especially the maximal magnitude of horizontal force, fixed. The ratio of vertical to horizontal force was $\frac{F_{\max,z}}{F_{\max,x,y}} = \frac{1}{2}, 1, 2, 4, 8$ from left to right. The material thickness is shown using the color map 0  0.2. On the right of each material distribution, we show the force in the cylinder of admissible values.

the tracking region in the center with the four arcs become thicker. Once the maximal thickness is reached in the center region and along these “beams”, further mass is invested to reinforce the regions close to the Dirichlet boundaries. The curved carrier “beams” and the central region are again designed asymmetrically w.r.t. the diagonal from the upper left to the lower right leading the follower to push towards the upper right.

Next, we investigate the effect of the parameters characterizing the strength of the forces, $F_{\max,z}$ and $F_{\max,x,y}$, while keeping the total amount of material constant. Due to the scaling invariance, we focus on the ratio $\frac{F_{\max,z}}{F_{\max,x,y}}$. In Figure 7.4, we show that with increasing strength of the vertical force, the “beams” become thinner and instead more material is concentrated in the central region. Interestingly, for small values of the ratio between the two forces the material distribution is nearly symmetrical w.r.t. the diagonal from the upper left to the lower right, while it is asymmetric for mid-range ratios and then becomes more symmetric again for large ratios.

Finally, Figure 7.5 shows the impact of the strength of the stochastic perturbation of the material thickness, as measured by the standard deviation, again for the tracking region on the roof plateau. With increasing strength of the stochastic perturbation the optimal structure becomes more diffuse. To understand this effect, let us consider the following illustrating scenario: If the leader were to concentrate material on a single row of triangles, then a significant reduction in thickness for even one of those triangles would render the whole construction ineffective since it disconnects the created beam. If, however, the leader were to spread out the same amount of material on triangles filling a square, then an entire arc of those elements would need to experience a significant reduction in thickness for the overall structure to noticeably lose strength. So, in the second case, more elements with a specific geometric structure would need to have reduced thickness to impact the overall design. Therefore, in the presence of large imprecision in the manufacturing, finely-structured designs would likely be ineffective. This could explain the more diffuse structures for stronger stochastic perturbations.

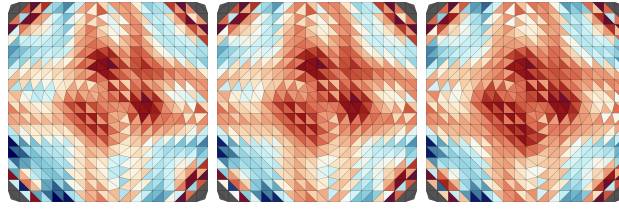


Figure 7.5: Comparison of material distribution when varying the standard deviation σ of the material perturbation while keeping the other parameters fixed. The standard deviation was $\sigma = \frac{5}{100}, \frac{1}{10}, \frac{2}{10}$ from left to right. Material thickness is shown using the color map $0 \rightarrow 0.2$, where fixed boundary triangles are shown in gray.

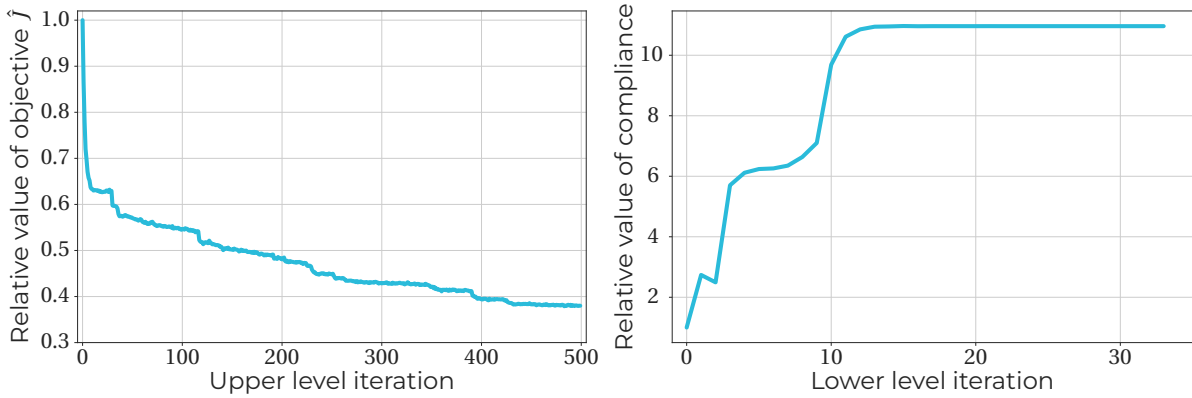


Figure 7.6: Left: upper level relative cost values $\hat{J}[u^i]/\hat{J}[u^0]$ for the iterates of the stochastic gradient descent method (Algorithm 2) in the example shown in the bottom row of Figure 7.2. Right: corresponding lower level compliance cost $\frac{y[u, BF^j]^T H[u] y[u, BF^j]}{y[u, BF^0]^T H[u] y[u, BF^0]}$ for iterates of the Newton-type method for the follower problem in the first upper level descent step and the initial material distribution.

Convergence. We also numerically investigated the convergence behavior of our descent respectively ascent methods. The results for a representative example are shown in Figure 7.6. It depicts the decrease of the upper level cost functional over the iterations of the stochastic descent algorithm and the increase of the lower level compliance cost when solving the follower problem for the initial material distribution. Latter solves of the follower problem typically require 10 to 30 iterations of the Newton-type method per outer iteration. For the former, we can see that it reaches a plateau of stable objective values.

Additional Examples. Lastly, we show two more complex examples of architectural designs of roof structures, inspired by [VHWP12]. These are shown in Figure 7.7. In the top row, we use a closed hall as the reference geometry for our bilevel optimization problem, while, in the bottom row, we use a geometry resembling a double torus cut in half. We again list the basic parameters in Table 7.1. For the example in the top row, the weights of the barrier terms are $\alpha^F = 10^{-4}$, $\alpha^u = 1$, and $\alpha^V = 10^{-3}$. Conversely, for the example in the bottom row, the weights of the barrier terms are $\alpha^F = 10^{-4}$, $\alpha^u = 1$, and $\alpha^V = 10^{-1}$. In both cases, we use the full domain as tracking set.

The main weakness of both structures is the concavity in the central part, which can be easily deformed by the vertical force. Hence, in both optimized solutions, the material is redistributed to prevent this. In the first case, this is done by building a stabilized ledge around the center, while in the second case beam-like structures emerge from the two holes along with another beam from the curve in the front. Furthermore, in the second example, the entrance is also stabilized by adding material at the ends of its arch.

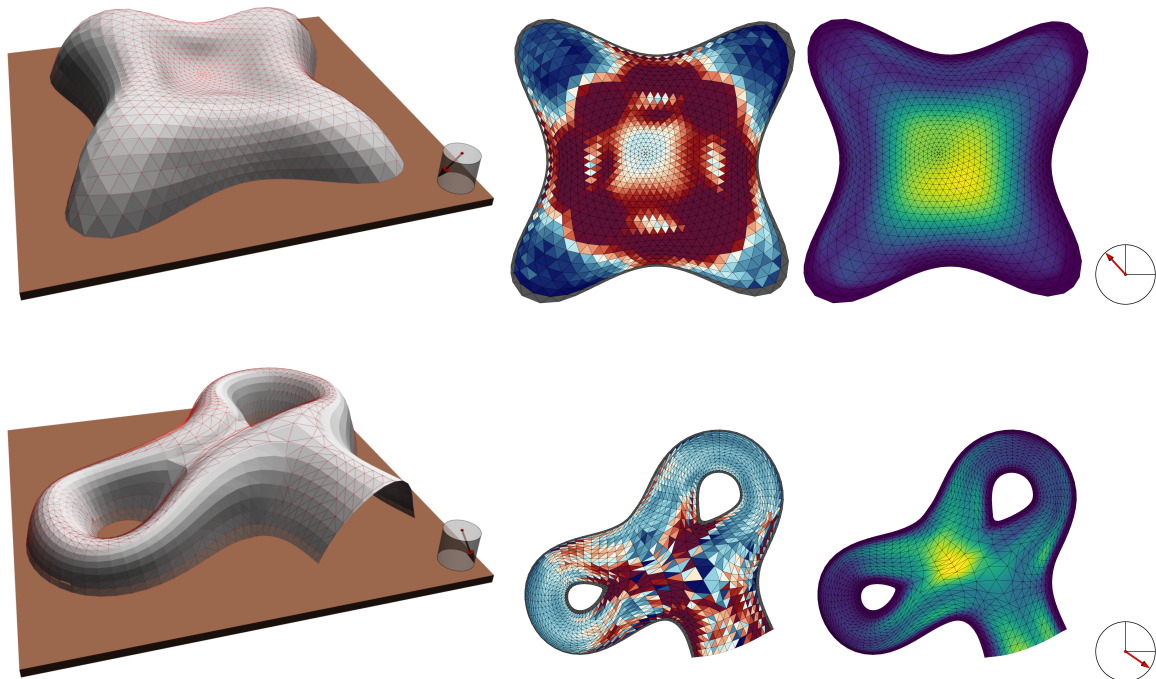




Figure 7.7: Results for two geometrically more complex examples. In both cases, we used a tracking on the entire domain. On the left, we show the deformed configuration as gray surface with the undeformed surfaces as a translucent overlay. Furthermore, we visualize the direction of the force leading to the maximal deformation in the cylinder. In the middle, we see the resulting material distributions using the color map 0  0.2. Boundary triangles for which all vertices are Dirichlet nodes are shown in gray. On the right, the magnitude of the deformation y is displayed using the color map 0  0.7. Additionally, on the far right, we show the two-dimensional direction of the horizontal forces.

7.5 Conclusion and Outlook

In summary, we studied a pessimistic stochastic bilevel problem tailored for elastic shape optimization. Theoretical investigations reveal the continuity of the resulting risk functional under a set of assumptions that can be relaxed when considering regularized models where the leader also hedges against lower level solutions that are close to optimality. As a proof-of-concept, we applied the approach to a material optimization problem on triangular surfaces, where we used a linearization of the discrete shell elasticity model also used in the first part of this thesis. Based on stochastic gradient descent, we developed a numerical approach for this problem and studied its solutions on a range of examples. This revealed interesting, symmetry breaking effects of our bilevel approach, for example, seen in Figure 7.2.

There are only few existing works studying pessimistic stochastic bilevel problems and, to the best of our knowledge, this is the first time pessimistic bilevel optimization has been utilized for a shape optimization problem, where the follower problem does not consist of the physical deformation model. Since it is the first of its kind, we envision that our approach could be very interesting for applications in geometric design. It allows for the optimization of structures under consideration of worst-case load scenarios and stochastic perturbations of design choices, e.g., through manufacturing accuracies. Our current theoretical framework and numerical approach already cover many discretized models based on linear elasticity. Hence, while we are only considering the thickness

of a discrete shell as the material parameter in our current application, in principle, the approach could be applied also to other design variables, for example, phase-field models as used in topology optimization.

Nevertheless, there are multiple interesting directions for extending the framework and numerical setup. For example, it would be interesting to establish our method also for continuous PDE-constrained optimization problems on infinite-dimensional function spaces. Then our current method would represent the spatially discretized case and the goal would be to prove variational convergence results for spatial refinement. This would help to provide the justification of applying our model to physically accurate elasticity models. From the point of view of elasticity, it would be interesting to study the nonlinear elasticity models and investigate the associated nonuniqueness issue in the lowest level problem. This would lead to a proper trilevel problem and bring new challenges for theoretical and numerical investigations. At the same time, it would enable considering models that are valid for stronger deformations.

In the numerical optimization, currently the smoothed problems are considered with fixed regularization parameters α . It would be interesting to extend this to interior point methods — or similar approaches from constrained optimization — and develop a numerical approach to solve the leader's and follower's problem with hard constraints instead of the regularization used here. Furthermore, we currently operate under the assumption that the solution of the lower level problem is unique. This was justified for our examples, but it would still be interesting to investigate numerical methods that allow to handle non-unique solutions of the lower level problem, for example, via methods from global optimization. Overall, this establishes a wide array of potential future work for extending our approach.

Chapter 8

A Phase-field Approach to Surface Segmentation

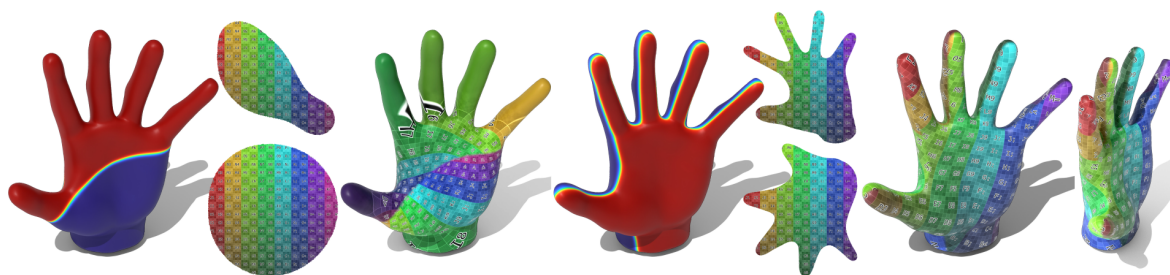


Figure 8.1: *Our approach produces segments with diffuse interfaces, shown here with a red to blue colormap, by solving a variational problem. By virtue of corresponding constraints, we can ensure that these segments are connected and cover the same amount of area. We can produce charts with overlapping support from these diffuse segments, shown are their images in the plane, and use them for mapping textures to the surface. However, when only using the perimeter as objective, as in the example on the left, the charts exhibit high distortion. Thus, we use the Yamabe equation for the logarithmic conformal factor as a PDE constraint and an objective involving this factor, which leads to distortion and perimeter minimizing segments as in the example on the right, where the textured hand is shown from two perspectives.*

One major challenge for the methods we have so far discussed in this thesis are performance considerations. These methods typically involve high-dimensional nonlinear optimization problems that are expensive to solve. Hence, we are interested in methods to reduce the computational complexity of these problems, for example, through reduced order models. In Chapter 5, we have constructed such a reduced approach tailored to the exponential map on the space of discrete shells. However, we are also interested in more general approaches, that, for example, do not require the availability of datasets.

A possible avenue is to consider techniques based on multi-resolution representations. In this final chapter, we start to lay the groundwork for such techniques by developing a novel approach to surface segmentation. That is for a surface \mathcal{S} , we want to construct two surfaces $\mathcal{S}_1, \mathcal{S}_2 \subset \mathcal{S}$ such that combining them gives the whole surface, i.e. $\mathcal{S} = \mathcal{S}_1 \cup \mathcal{S}_2$, and such that their boundary $\gamma := \mathcal{S}_1 \cap \mathcal{S}_2$ is a curve. Our motivation to study such segmentations comes from the goal to construct Tausch-White wavelets [TW03] on complex unstructured triangle meshes. To this end, one needs a so-called cluster tree for the mesh, which corresponds to a hierarchical segmentation. To achieve optimal preconditioning effects via this wavelet method, the segments have to be simply-connected

and should have approximately the same size if they are on the same level (cf. [AHK14]). However, surface segmentations have a variety of further applications in geometry processing. They are a core component in many algorithms that are used, for example, in computational design, collision detection, remeshing, and texture mapping.

In the rest of this chapter, we will consider the computation of atlases for a surface as proof of concept problem for our approach. The probably most famous example of this problem is the construction of an atlas of the earth, where different projections either have to distort the shape or size of landmasses. This example directly shows that there is not a single quantity to determine what constitutes a 'good' atlas. Typically different properties such as preserving angles, low area distortion, number of charts, and interface length need to be traded against each other. We can see an example of the trade-off between distortion and interface length in Figure 8.1. In computer graphics applications, conformal maps, i.e. preserving angles, have emerged as the dominating paradigm for generating charts. This raises the problem of constructing such conformal charts with minimal length and area distortion, which is typically approached by developing algorithms that try to optimally place the segments, and is a very active area of research. There is also a corresponding free boundary problem, where one aims to place cuts on the surface such that it can be flattened with low distortion but will possibly remain in one piece. However, we will focus only on the segmentation problem, i.e. always divide the surface into multiple parts.

There are various ways to mathematically describe the segmentation of a surface. One way is to explicitly parametrize the boundary between the segments as a curve on the surface. This is used in many segmentation techniques for triangle meshes, where typically this curve is restricted to follow edges of the mesh such that one obtains a segmentation into sets of triangles. Therefore, these approaches are typically purely discrete formulations and do not consider an underlying continuous problem. Another approach is to describe the boundary using a level set method, i.e. describe it using the zeroth level set of a function defined on the domain and then vary this function. This is, for example, done in [SC18], which served as major inspiration for our work. In their work, they introduce a continuous PDE-constrained shape optimization of the distortion incurred by conformally mapping the segments to the plane and discretize it via a level set approach. However, the use of level sets requires careful handling of the interface in the optimization and they are typically not well suited to deal with topology changes, e.g. promote them when desirable.

In our approach, we use phase-fields, a model for segments with diffuse interfaces that, in the case of two segments, assign each point on the surface a value between minus one and one. These endpoints figuratively correspond to two pure material phases while the range in between models a diffuse transition. Using phase-fields for computing surface segmentations yields a topologically flexible variational framework. This, however, could entail that our minimizers are disconnected, which is often undesirable, for example, when creating an atlas and also when building a cluster tree. To mitigate this, we employ the connectedness constraint introduced by [DLW17]. Furthermore, we introduce a novel hierarchical approach that multiplicatively combines multiple phase-fields to describe more than two segments. The resulting diffuse hierarchical segmentation can then be used to generate atlases with overlapping charts that are well-suited for applications such as texture mapping. It would also be suitable to generate cluster trees of the surface used in the construction of Tausch-White wavelets. To control the distortion of these atlases, we also consider the Yamabe equation as PDE-constraint to compute the distortion induced by conformally flattening the segments as proposed in [SC18]. We introduce a diffuse version of the Yamabe equation that sidesteps any need to threshold the diffuse representation or explicitly cut the mesh. The resulting end-to-end diffuse formulation of the surface segmentation problem leads to variational problems that are straightforward to discretize with affine finite elements and treat with conventional algorithms for nonlinear optimization.

We will begin the chapter by introducing the Modica–Mortola model for phase-fields on Euclidean domains together with the connectedness penalty in Section 8.1. Then, in Section 8.2, we

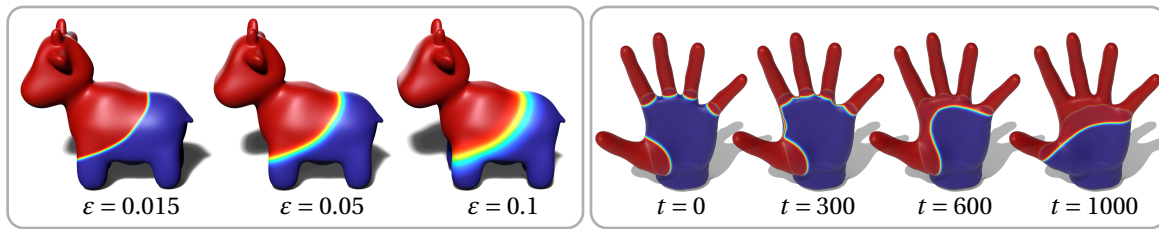


Figure 8.2: Results of the phase-field segmentation (8.22). Left: Interface minimization results for different values of ε yielding different interface widths. Right: Adding the connectedness constraint to the variational problem leads to a gradient descent flow (from which different time steps are shown) that connects phases while minimizing the interface length.

will adapt this phase-field model to surfaces embedded in \mathbb{R}^3 , introduce our segmentation problem in its simplest form, and introduce our approach for hierarchical segmentation using these phase-fields. To control the distortion in this setting, we discuss the Yamabe equation for sharp interfaces, introduce our diffuse version of it, and formulate our PDE-constrained shape optimization problem in Section 8.3. Finally, in Section 8.4, we propose a discretization of this problem and show some results.

Remark. This chapter is the result of joint work with Janos Meny and Martin Rumpf published in [MRS21]. In particular, Janos Meny implemented the method and conducted the numerical experiments presented in this chapter.

8.1 Phase-fields on Euclidean Domains

In this section, we will recall the necessary background on phase-fields on Euclidean domains that we will adapt to surfaces in the following sections and then use to compute segmentations. Concretely, we will focus on phase-fields of *Modica–Mortola*-type. They can be used to describe the segmentation via a smooth function that intuitively models the mixture of two materials. The theoretical result that allows us to make this connection is the Γ -convergence of the Modica–Mortola functional — which variationally describes the separation of these materials — to the perimeter of subdomains. This Modica–Mortola functional is straightforward to discretize with standard finite element methodology, as we will also see later on, and thus lends itself well for the variational formulation of segmentation problems for numerical purposes. Due to the Γ -convergence result, this yields an effective approximation of the perimeter, which has been used extensively for image segmentation [BCM04], approximating motion by mean curvature [DF20], shape optimization [BC03], simulation of material processes [Che03], and many other applications. The diffuse model also has a long history ranging back to the likes of Lord Rayleigh, Gibbs, and Van der Waals in the late 19th century (cf. [Row79]) with important contributions by Cahn, Hilliard, and Allen [CH58; AC72] in the middle of the 20th century, who developed the model in a modern language and thus laid the groundwork for many further developments.

Concretely, we represent the segments using a phase-field function $u \in W^{1,2}(\Omega)$, where $\Omega \subset \mathbb{R}^d$ is an open region. As previously mentioned, one can think of this function describing the mixture of two materials in the domain where -1 respectively 1 indicate the pure material phases. They mix at interfaces with intermediate values between -1 and 1 describing the ratio of this mixture. Then, one postulates that the materials have a desire to be as pure as possible. To describe this variationally, one considers a local chemical potential $\Psi: \mathbb{R} \rightarrow \mathbb{R}$ with two global minima at ± 1 . The resulting total bulk energy of the phase-field is given by $\int_{\Omega} \Psi(u) dx$. Furthermore, one also postulates that if these materials mix they do so smoothly, which is variationally measured by the Dirichlet energy $\int_{\Omega} |\nabla u|^2 dx$ of the phase-field. The weighting of these two complimentary energies controls

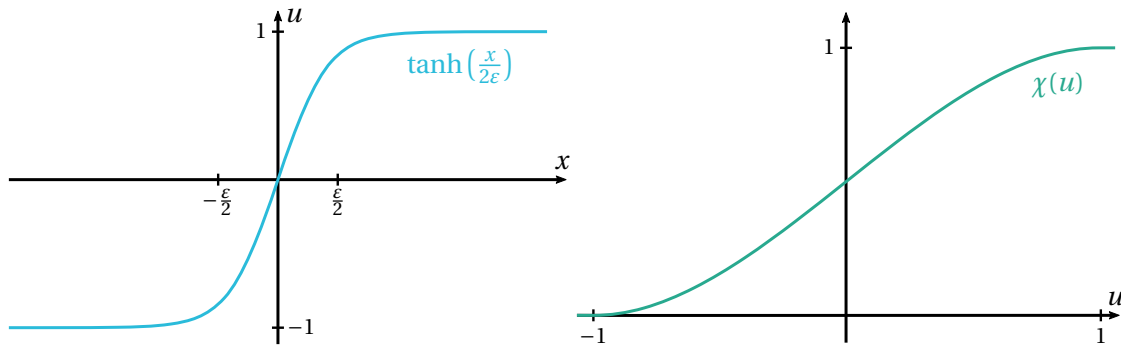


Figure 8.3: Optimal profile and diffuse indicator function. Left: Optimal profile of tanh-type for minimizers of the Modica–Mortola functional. Right: Diffuse indicator function χ depending on phase-field u interpolating cubically between zero and one.

the width of the interface between the two pure phases, which one describes with a parameter $\varepsilon > 0$, as demonstrated in Figure 8.2. With this scaling, we obtain the Modica–Mortola functional

$$\mathcal{P}_\varepsilon[u] := \int_\Omega \frac{\varepsilon}{2} |\nabla u|^2 + \frac{1}{\varepsilon} \Psi(u) dx. \quad (8.1)$$

Depending on the context, this functional is also called the *Ginzburg–Landau* or the (*Van der Waals–Cahn–Hilliard*) functional and its L^2 gradient flow is known as the *Allen–Cahn* equation. There are different possible double-well potentials Ψ and we will use the popular choice $\Psi(u) = \frac{9}{16}(u^2 - 1)^2$.

To compute quantities related to the different phases, it is typically desirable to describe them using approximate indicator functions, i.e. functions that are one inside the phase and zero outside. There are several alternatives for this (e.g. linear and quadratic interpolations) and we choose the piecewise cubic function

$$\chi(u) := \begin{cases} 0 & u \leq -1 \\ \frac{1}{4}(u+1)^2(2-u) & -1 < u < 1 \\ 1 & u \geq 1 \end{cases}$$

as a diffuse indicator for the positive phase $\{u \approx 1\}$, which is shown in Figure 8.3 on the right. χ is C^1 smooth on \mathbb{R} , which is advantageous for the numerical minimization of functionals involving χ . Furthermore, the symmetry $\chi(-u) = 1 - \chi(u)$ ensures that $\chi(-u)$ is a proper approximation of an indicator function of the phase $\{u \approx -1\}$ and allows the definition of a partition of the domain based on χ . With χ at hand, we can for example approximate the area of the phase $\{u \approx 1\}$ via its integration, i.e.

$$\mathcal{A}[u] := \int_\Omega \chi(u) dx, \quad (8.2)$$

and then it follows for the negative phase that $\mathcal{A}[-u] = \int_\Omega dx - \mathcal{A}[u]$.

Sharp Interface Limit. As mentioned before, we want to use this functional as approximation of the perimeter of subsets with sharp interfaces between them. Hence, we can ask ourselves what happens if we let the width of the interface go to zero, i.e. $\varepsilon \rightarrow 0$. This leads to following Γ -convergence result due to Modica and Mortola, which also gives us the convergence of minimizers because of the coerciveness of \mathcal{P}_ε .

Theorem 8.1 (Modica–Mortola [MM77]). *Let $\mathcal{P}_\varepsilon[u] := \infty$ for $u \notin W^{1,2}(\Omega)$, then*

$$\Gamma\text{-}\lim_{\varepsilon \rightarrow 0} \mathcal{P}_\varepsilon = c_\Psi \mathcal{P}, \quad (8.3)$$

with respect to the $L^1(\Omega)$ -topology, where $c_\Psi = \int_{-1}^1 \sqrt{\Psi} du$ and

$$\mathcal{P}[u] := \begin{cases} H^{d-1}(\Omega \cap \partial\{x \in \Omega : u(x) = 1\}) & u(\Omega) \subseteq \{-1, 1\} \\ \infty & \text{else.} \end{cases} \tag{8.4}$$

Here, ∂ denotes the measure-theoretic essential boundary.

Note that we have $c_\Psi = 1$ for our choice of the double-well potential Ψ as above. To prove this theorem, one first considers the one-dimensional case, i.e. $d = 1$, where one finds an optimal profile for a smooth transition between -1 and 1 . This is done by equating the two energy contributions yielding an ordinary differential equation whose solutions under appropriate boundary conditions is then the sought optimal profile. For our concrete choice of Ψ from above, one obtains optimal profiles of tanh-form shown in Figure 8.3 on the left. Then, in the general case, one shows that this optimal profile occurs perpendicular to the interface using the so-called slicing method. The full proof can be found, for example, in [Bra02].

The consequences of this theorem can be understood in two ways. On the one hand, if we consider the Modica–Mortola functional as a physical model for phase transitions the above theorem allows one to model these transition with sharp interfaces. On the other hand, and this is the perspective we take here, we can use the Modica–Mortola functional as smooth approximation of the perimeter which is straightforward to discretize with standard finite element methodology and thus is well suited for numerical approaches.

Connectedness Constraint. Using phase-fields to describe the segmentation allows to easily handle changes of topology during the optimization. However, this also entails that with the functionals introduced so far we now have control over the topology of the resulting segments. As motivated in the beginning, we want these segments to be at least connected in many applications. To this end, we will use a constraint introduced by Dondl, Lemenant, and Wojtowysch [DLW17], which guarantees the path-connectedness of phases, and introduce the necessary background on it here based on [DNWW19].

Recall that an open set $U \subset \mathbb{R}^d$ is connected if and only if it is path-connected, i.e. for every $x, y \in U$ there exists a continuous curve $c: [0, 1] \rightarrow U$ with $c(0) = x$ and $c(1) = y$. Dondl et al. use the length of paths necessary to make an open set path-connected to develop a quantitative notion of connectedness. Let $F \in C(\mathbb{R}^d)$ be a function such that $F \equiv 0$ on \overline{U} and $F > 0$ on $\mathbb{R}^d \setminus \overline{U}$. Then we define the weighted distance

$$d^F(x, y) := \inf \left\{ \int_c F d\mathcal{H}^1 \mid c: [0, 1] \rightarrow \mathbb{R}^d, c(0) = x, c(1) = y \right\}. \tag{8.5}$$

This distance is zero for two points in the same connected component of \overline{U} and positive for ones in different connected components, at least if everything is sufficiently regular. From this follows that a quantitative measure for the path connectedness of U is given by $\int_{U \times U} d^F(x, y) dx dy$.

To use this for phase-fields, we would consider the preimage of an interval $U = u^{-1}([\alpha, \beta])$ and apply the above measure to it. However, this would depend on u in a non-differentiable fashion. This problem can be circumvented by introducing a smooth bump function $H \in C^1(\mathbb{R})$ such that $H > 0$ on (α, β) and $H = 0$ on $\mathbb{R} \setminus (\alpha, \beta)$. Hence, one obtains the functional

$$\mathcal{C}[u] := \int_{\Omega \times \Omega} H(x)H(y)d^F(u)(x, y) dx dy, \tag{8.6}$$

which indeed is differentiable with respect to the phase-field parameter u .

In detail, we follow [DNWW19] and choose the interval $[1 - \sqrt{\varepsilon}, 1]$ for the connectedness of the positive phase with corresponding functions

$$F_\varepsilon(u) := \begin{cases} (u - 1 + \sqrt{\varepsilon})^2 & u \leq 1 - \sqrt{\varepsilon} \\ 0 & u \geq 1 - \sqrt{\varepsilon} \end{cases}, \quad H_\varepsilon(u) := \begin{cases} 0 & u \leq 1 - 2\sqrt{\varepsilon} \\ 1 & u \geq 1 - \sqrt{\varepsilon} \end{cases},$$

where H_ε is smoothly interpolated between the given values using a cubic polynomial, which makes it essentially a restricted version of our diffuse indicator function χ . We denote the connectedness penalty with these specific functions by \mathcal{C}_ε . For the negative phase, we mirror this at zero, i.e. we use $\mathcal{C}_\varepsilon[-u]$.

In Figure 8.2, we see that combining the Modica–Mortola functional (8.1) and the connectedness term (8.6) in a variational problem leads to connected phases minimizing the interface length. Indeed, [DNWW19] prove, in the two-dimensional case, a Γ -convergence result for this combination to the usual perimeter functional \mathcal{P} under a connectedness constraint in the sharp interface limit. To this end, we consider the connected perimeter as discussed in [DMNP22], which is for a measurable set $E \subset \mathbb{R}^2$ defined as

$$\mathcal{P}_C(E) := \left\{ \liminf_{n \rightarrow \infty} \mathcal{P}(E_n) \mid E_n \rightarrow E, E_n \text{ indecomposable} \right\}. \quad (8.7)$$

An open set U_1 is called decomposable if there exist open sets U_1, U_2 such that $U = U_1 \cup U_2$ in the L^1 -sense and $\mathcal{P}(U) = \mathcal{P}(U_1) + \mathcal{P}(U_2)$. It is called indecomposable if it is not decomposable. This is a measure-theoretic generalization of connectedness, and indeed every open connected set with finite perimeter is indecomposable. For smooth sets, we can consider the usual notion of connectedness and define

$$\mathcal{P}_C^r(E) := \left\{ \liminf_{n \rightarrow \infty} \mathcal{P}(E_n) \mid E_n \rightarrow E, E_n \text{ connected and } C^\infty\text{-smooth} \right\}. \quad (8.8)$$

A quick observation for these definitions is that for an indecomposable resp. connected set E we have $\mathcal{P}_C(E) = \mathcal{P}(E)$ resp. $\mathcal{P}_C^r(E) = \mathcal{P}(E)$. Furthermore, Dayrens *et al.* [DMNP22] proved that if $E \subset \mathbb{R}^2$ is an essentially bounded set of finite perimeter such that $\partial E = \partial^* E$ up to sets of \mathcal{H}^1 -measure zero, we have the identities

$$\mathcal{P}_C^r(E) = \mathcal{P}_C(E) = \mathcal{P}(E) + 2\text{St}(E), \quad (8.9)$$

where $\text{St}(E)$ is the length of the Steiner tree of \bar{E} , i.e.

$$\text{St}(E) := \inf \{ \mathcal{H}^1(K) \mid E \cup K \text{ connected} \} \quad (8.10)$$

Note that we consider subsets of a bounded domain, i.e. $E \subset \Omega$, and one would need to adapt the above definitions to it. However, Dondl *et al.* [DNWW19] proved that these adapted functionals agree with the ones given above when the convex hull of E is contained in Ω . This is especially the case for convex domains Ω and to simplify the notation we will assume this is the case for the rest of this section.

With these preliminaries out of the way, we are in the position to formulate the convergence result we are interested in:

Theorem 8.2. *Let $\Omega \subset \mathbb{R}^2$ be open, bounded, and convex. Let \mathcal{C}_ε be the connectedness penalty with the functions F_ε and H_ε as above. Then*

$$\left(\Gamma\text{-}\lim_{\varepsilon \rightarrow 0} \mathcal{P}_\varepsilon + \varepsilon^{-\kappa} \mathcal{C}_\varepsilon \right) [u] = \begin{cases} \mathcal{P}_C^r(\{u = 1\}) & u \in BV(\Omega, \{0, 1\}) \\ \infty & \text{else,} \end{cases} \quad (8.11)$$

with respect to the $L^1(\Omega)$ -topology, where $\kappa > 0$.

The proof for this theorem can be found in [DNWW19]. It shows that indeed the combination of perimeter and connectedness penalty converges to the connected perimeter and thus motivates our usage of it for connected segmentation going forward.

8.2 Phase-fields on Surfaces

So far, we have introduced phase-fields and all derived notions on Euclidean spaces. However, our problem is formulated on curved surfaces of which we do not have a parametrization available. Hence, we will adapt all these notions to surfaces embedded in \mathbb{R}^3 . We will see that this is straightforward to do and gives us a powerful variational framework for segmentation problems on surfaces.

To this end, we consider a Riemannian surface \mathcal{S} embedded in \mathbb{R}^3 . Then, we adapt, for a phase-field function $u \in W^{1,2}(\mathcal{S})$, the Modica–Mortola to read as

$$\mathcal{P}_\varepsilon[u] = \int_{\mathcal{S}} \frac{\varepsilon}{2} |\nabla_{\mathcal{S}} u|^2 + \frac{1}{\varepsilon} \Psi(u) da, \quad (8.12)$$

where the integration is with respect to the Riemannian metric on the surface, and similarly $\nabla_{\mathcal{S}} u$ denotes the weak gradient of u also w.r.t. the metric. We are not aware of a theoretical convergence result for this functional on surfaces, but since all operations in the Γ -convergence proof are local an argument based on charts seems plausible. The diffuse indicator function remains the same as before and thus the area functional becomes $\mathcal{A}[u] = \int_{\mathcal{S}} \chi(u) da$. To adapt the connectedness constraint, we have to adapt the employed distance function to consider curves on the manifold, i.e. $d^F(x, y) = \inf \{ \int_c F d\mathcal{H}^1 \mid c: [0, 1] \rightarrow \mathcal{S}, c(0) = x, c(1) = y \}$. The overall functional is then again adopted by simply replacing the notion of integration, which yields $\mathcal{C}_\varepsilon[u] = \int_{\mathcal{S} \times \mathcal{S}} H_\varepsilon(u(s)) H_\varepsilon(u(t)) d^{F_\varepsilon(u)}(s, t) ds dt$.

Segmentation. Based on the phase-field approach and the corresponding connectedness constraint, we are in the position to formulate our basic segmentation problem. In it, we ask for a segmentation of the surface into two segments, such that each segment is connected, covers half the surface, and the length of the boundary between the segments is minimal. Using the functionals introduced before, the phase-field version of this problem becomes the following variational problem

$$\begin{aligned} & \underset{u \in W^{1,2}(\mathcal{S})}{\text{minimize}} && \mathcal{P}_\varepsilon[u] + \varepsilon^{-\kappa} (\mathcal{C}_\varepsilon[u] + \mathcal{C}_\varepsilon[-u]) \\ & \text{subject to} && \mathcal{A}[u] = \frac{1}{2} \mathcal{H}^2(\mathcal{S}), \end{aligned} \quad (8.13)$$

for some $\kappa > 0$ which is typically chosen to be one in experiments. In Figure 8.2, we show solutions of this problem for two examples.

This basic formulation provides us with a variational framework to formulate further, more involved segmentation problems on surfaces using phase-fields. For example, if we do not require exactly matching areas but rather want to regularize our segmentation, then the area constraint $\mathcal{A}[u] = \frac{1}{2} \mathcal{H}^2(\mathcal{S})$ can also be relaxed using a properly scaled quadratic penalty $(\mathcal{A}[u] - \frac{1}{2} \mathcal{H}^2(\mathcal{S}))^2$. Furthermore, other constraints such as mechanical energies of the phases could be adapted from phase-fields on Euclidean domains or developed based on surface geometry. We will extend this framework with a hierarchical approach to compute more than two segments and with an diffuse adaption of the Yamabe equation to incorporate the distortion incurred by conformally mapping the segments to the plane.

8.2.1 Hierarchical Approach

In this section, we will discuss a hierarchical generalization of the above segmentation approach. In a hierarchical segmentation, previously computed segments are recursively subdivided and this yields a flexible and useful tool. For example, with an increasing number of segments the geometric complexity of each segment decreases, which is advantageous for applications like fabrication. Furthermore, the segmentation hierarchy can be used for multi-resolution feature representations.

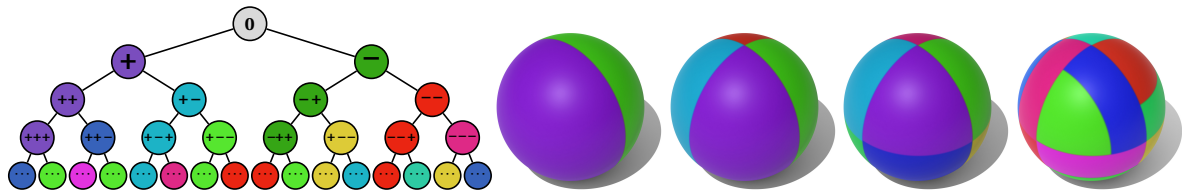


Figure 8.4: Hierarchical segmentation of the sphere: on the left the binary tree with nodes representing the $\alpha \in \{-1, +1\}^K$ with $K = 1, 2, 3, 4$. The nodes are identically color coded as the corresponding segments on the right.

For example, it can be used to generate the cluster tree for Tausch-White wavelets [TW03] on triangle meshes, which require the segments to be connected and approximately equally sized to yield well-conditioned problems.

In our phase-field setup, one could numerically realize this hierarchy by subdividing the used mesh along the zero-level set of the phase-field describing the segment. This is, however, undesirable as it requires cutting the mesh, which can be cumbersome and complicates transferring the resulting segmentation to the original mesh. Furthermore, it obstructs establishing rigorous sharp interface limits of the model. Instead, we construct a fully diffuse hierarchical approach. To this end, we use additional phase-field functions for the subdivision steps and diffuse indicator functions identifying the segment to be further subdivided.

To organize our hierarchical segmentation, we construct a binary tree of segments. We use multi-indices to refer to the different nodes of this tree, i.e. a node on level K of this tree is identified by the label $\alpha \in \{-1, +1\}^K$ describing the subsequent left-right choices in the tree. For the root of the tree, we use the index zero. This indexation is visualized in Figure 8.4. Given a node $\alpha = (\alpha_1, \dots, \alpha_K) \in \{-1, +1\}^K$, we refer to the nodes preceding it in the tree by $\alpha^{k-} := (\alpha_1, \dots, \alpha_{k-1}) \in \{-1, +1\}^{k-1}$ for $k \in \{1, \dots, K\}$, where $\alpha^{1-} := 0$ refers to the root of the tree.

With every node α , we associate a diffuse indicator function χ^α describing the corresponding segment of the surface. Furthermore, with every non-leaf node, we associate a phase-field u^α that subdivides this segment. Then, we can define the χ^α recursively as the product of single level segmentation functions, i.e.

$$\chi^\alpha := \chi(\alpha_K u^{\alpha^{K-}}) \chi^{\alpha^{K-}}$$

where the indicator function of the root is constant one, i.e. $\chi^0 \equiv 1$. From this, we can easily proof the following observation, which guarantees that the χ^α describe a proper decomposition of the surface.

Lemma 8.3. For every K , $\{\chi^\alpha\}_{\alpha \in \{-1, +1\}^K}$ forms a partition of unity on S , i.e.

$$\sum_{\alpha \in \{-1, +1\}^K} \chi^\alpha(x) = 1 \quad (8.14)$$

for all $x \in S$.

Proof. Recall, that by definition of χ , we have $\chi(u) + \chi(-u) \equiv 1$ for all functions u . Then, we see that

$$\begin{aligned} \sum_{\alpha \in \{-1, +1\}^K} \chi^\alpha &= \sum_{\alpha \in \{-1, +1\}^K} \chi(\alpha_K u^{\alpha^{K-}}) \chi^{\alpha^{K-}} \\ &= \sum_{\tilde{\alpha} \in \{-1, +1\}^{K-1}} (\chi(u^{\tilde{\alpha}}) + \chi(-u^{\tilde{\alpha}})) \chi^{\tilde{\alpha}} \\ &= \sum_{\tilde{\alpha} \in \{-1, +1\}^{K-1}} \chi^{\tilde{\alpha}}, \end{aligned}$$

where the first equality is simply by definition and the second by considering the two possible choices for α_K . At this point, we can proceed via induction. \square

To compute the phase-fields u^α , we have to adapt our segmentation problem (8.13) to incorporate the restriction to a diffuse segment given by a χ^α . To this end, we consider a relaxed indicator function $\chi_\eta^\alpha = \chi^\alpha + \eta$ for $0 < \eta \ll 1$, which is supported on the whole surface but is significantly dampened outside the segment described by χ^α . The reason for the dampening will be discussed below. We define a regularized diffuse interface length by

$$\mathcal{P}_{\varepsilon,\eta,\alpha}[u] := \int_S \frac{\varepsilon}{2} \chi_\eta^\alpha |\nabla u|^2 + \frac{1}{\varepsilon} \chi^\alpha \Psi(u) da + \frac{1}{\sigma} \int_S (1 - \chi^\alpha) u^2 da, \quad (8.15)$$

where the first term is a diffuse restriction of the Modica-Mortola functional and the second term, with $0 < \sigma \ll 1$, forces the phase-field to be close to zero outside the considered segment and thus corresponds to zero boundary conditions. Furthermore, we consider a weighted approximation of the surface area functional

$$\mathcal{A}_\alpha[u] := \int_S \chi^\alpha \chi(u) da.$$

Finally, for the connectedness constraint, we simply replace F_ε in the previous formulation by $F_{\varepsilon,\eta,\alpha} := (\chi_\eta^\alpha)^{-1} F_\varepsilon$ and refer to the resulting functional by $\mathcal{C}_{\varepsilon,\eta,\alpha}$. This prevents the paths connecting a disconnected phase taking a shortcut by going outside the segment that is to be subdivided.

With these functionals in place, u^α is given as the solution of

$$\begin{aligned} & \underset{u \in W^{1,2}(S)}{\text{minimize}} && \mathcal{P}_{\varepsilon,\eta,\alpha}[u] + \varepsilon^{-\kappa} (\mathcal{C}_{\varepsilon,\eta,\alpha}[u] + \mathcal{C}_{\varepsilon,\eta,\alpha}[-u]) \\ & \text{subject to} && \mathcal{A}_\alpha[u] = \frac{1}{2} \int_S \chi^\alpha da. \end{aligned} \quad (8.16)$$

As before in the single level segmentation problem, the constraint forces a splitting of the segment with indicator function χ^α into two equally sized segments with indicator functions $\chi^\alpha \chi(u)$ and $\chi^\alpha \chi(-u)$. The relaxation of χ^α in the definition of the diffuse perimeter functional $\mathcal{P}_{\varepsilon,\eta,\alpha}$ is required to ensure well-posedness of the above variational problem. In fact, u is still a function on the whole surface S and the first term of the perimeter functional is strictly coercive but the impact of values outside the current segment to be subdivided is strongly damped by η . As the basic segmentation problem (8.13), this problem could also be extended with other objectives and the penalty could be replaced by a quadratic penalty. The illustrative segmentation seen in Figure 8.4 were computed using this problem. Furthermore, we see in Figure 8.5 that hierarchical segmentation of a torus could quickly lead to disconnected segments, which is prevent by the connectedness constraint.

This top-down approach allows to efficiently compute hierarchical segmentations where the interfaces between segments can flexible relax on each level of the hierarchy. However, this type of approach has the general limitation that the quality of lower level segmentations is limited by the already fixed higher level segmentations. In our experiments, we did not observe this to be a problem. Nevertheless, our phase-field model would also allow to optimize all levels simultaneously by integrating the objectives from (8.16) into a single objective or by using methods from multi-objective optimization.

8.3 A Diffuse Yamabe Equation

The main application of surface segmentations that we consider in this chapter is the generation of atlases. These are frequently used for texture mapping, i.e., transferring textures laid out in the plane to the surface. However, if we use our method as introduced above, these textures will undergo heavy distortions (cf. Figure 8.1 on the left). Furthermore, another potential application is the fabrication of surfaces, where the segments would be produced by bending flat sheets of material, which is also limited by the necessary distortion. Hence, we want to minimize the distortion incurred by conformally mapping segments to the plane that are described via diffuse indicator functions.

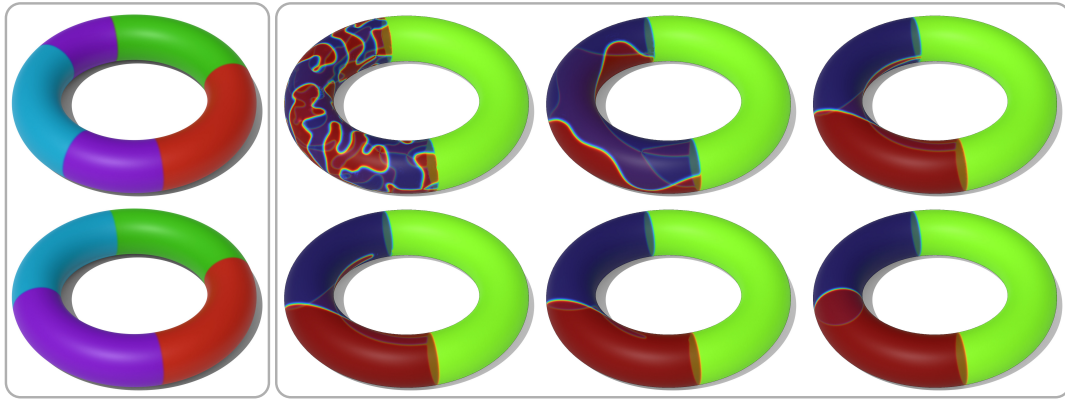


Figure 8.5: Hierarchical segmentation on a torus. Left: the segmentation of a torus might quickly lead to disconnected local minima (top). Adding the connectedness constraint to the objective prevents this (bottom). Right: Different time steps of a gradient descent scheme for the variational problem (8.24) with connectedness constraint on level $K = 2$ and the segmentation of the associated left half of the torus starting from a random initialization.

In this section, we will discuss how to reduce the distortion of these chart mappings via optimization of the underlying surface segmentation. To this end, we will first discuss the necessary basics from differential geometry, and then explain the Yamabe equation for computing the conformal distortion for subsets U of S without having to compute the conformal flattening itself. Finally, we will introduce a diffuse version of it based on our phase-field approach and the corresponding PDE-constrained optimization problem to minimize the distortion.

Conformal Flattening. We will begin with recalling the necessary notions regarding conformal equivalence and maps from differential geometry. For these notions, we follow Kühnel [Küh15].

Definition 8.4 (Conformal equivalence). Let S be a surface. Two metrics g, \tilde{g} on S are conformally equivalent if there exists a smooth function $s: S \rightarrow \mathbb{R}$ such that

$$\tilde{g}_x = e^{2s(x)} g_x$$

for all $x \in S$. The factor e^{2s} is called conformal factor and s is called the logarithmic conformal factor.

The conformal factor always has to be positive, which is the main motivation to write it in the form e^{2s} . With this definition of conformal equivalence at hand, we can extend it to transformations of surfaces.

Definition 8.5 (Conformal map). Let $\phi: S \rightarrow \tilde{S}$ be a diffeomorphism between Riemannian surfaces (S, g) and (\tilde{S}, \tilde{g}) . Then ϕ is called a *conformal map* if the pulled-back metric $\phi^{\#} \tilde{g}$ is conformally equivalent to g .

This means that we can also associate a logarithmic scale factor $s: S \rightarrow \mathbb{R}$ with a conformal map ϕ . Conformal maps preserve the local angles on a surface, i.e. the angles between tangent vectors, but distort lengths and areas. This distortion is quantified by the conformal factor e^{2s} (or equivalently the logarithmic conformal factor s).

Definition 8.6 (Conformal flattening). Let $U \subset S$ be an open subset of a Riemannian surface (S, g) . Then we call a conformal map $\phi: U \rightarrow \mathbb{R}^2$ a *conformal flattening* of U .

By the uniformization theorem (see e.g. [Abi81]), such a conformal map exists for all open subsets $U \subset S$ with disk-like topology.

Yamabe Equation. Our goal is to compute a segmentation of \mathcal{S} that minimizes the distortion incurred by conformally flattening the segments. However, explicitly computing a conformal flattening is a challenging numerical problem in itself and thus we would like to avoid it in our optimization. To this end, we follow the idea of [SC18] to compute s via the Yamabe equation without needing to explicitly compute the flattening. Below, we recall the necessary background on the sharp interface case before we introduce our adaption to diffuse interfaces. We start with the following characterization of how the Gauß curvature differs between conformally equivalent metrics.

Lemma 8.7. *Let (\mathcal{S}, g) be a Riemannian surface. Given a conformally equivalent metric \tilde{g} , the Gauß curvature \tilde{K} induced by it is related to the original Gauß curvature via*

$$e^{2s} \tilde{K} = K + \Delta_{\mathcal{S}} s.$$

A proof for this classical result can, for example, be found in [SY94]. Based on this, Springborn, Schröder, and Pinkall [SSP08] proved that with uniform boundary conditions — corresponding to a uniform scaling of the boundary — this equation actually suffices to compute the best possible scale factor of a flattening in terms of its Dirichlet energy.

Proposition 8.8 ([SSP08]). *Let (\mathcal{S}, g) be a connected Riemannian surface with boundary. Then the logarithmic scale factor s of a conformally equivalent flat metric \tilde{g} minimizes the Dirichlet energy among all such scale factors if it solves the Poisson equation $\Delta_{\mathcal{S}} s = -K$ with boundary conditions $s|_{\partial\mathcal{S}} = \text{const}$.*

The proof can be found in [SSP08] and essentially shows that the variation of the Dirichlet energy w.r.t. harmonic functions is controlled by their boundary values and for s to be a critical point it has to be constant along the boundary.

All of this can also be applied to an open connected subset $U \subset \mathcal{S}$. This leads to the equation

$$\begin{aligned} \Delta_{\mathcal{S}} s &= -K, & U, \\ s &= 0, & \partial U. \end{aligned} \tag{8.17}$$

The choice of zero boundary condition means that the metric does not change on the boundary, i.e. the boundary length is preserved. This PDE is also called the Yamabe equation [Aub98] as it forms the core of the Yamabe problem in higher dimensions. Solving this PDE thus allows us to compute the scale factor with minimal Dirichlet energy when flattening one of our segments without computing the flattening itself. Hence, we can use it as a PDE-constraint in a shape optimization of the segments and use the resulting (logarithmic) scale factor in the objective.

However, the formulation (8.17) requires an explicit subset of the surface, whereas in our phase-field approach these are only described via approximate indicator functions. We will therefore introduce a diffuse version of the Yamabe equation next.

Diffuse Yamabe Equation. To this end, we make use of the variational formulation of the Yamabe equation

$$\min_{s \in W_0^{1,2}(U)} \int_U \frac{1}{2} |\nabla s|^2 - K s \, da. \tag{8.18}$$

For u being a fixed phase-field segmentation of \mathcal{S} as introduced in Section 8.2, we want to solve the Yamabe equation for both diffuse segments identified by $\chi(u)$ and $\chi(-u)$, respectively. Therefore, we replace the restriction of (8.18) to U by the diffuse indicator functions and obtain

Definition 8.9 (Diffuse Yamabe Equation). Given a phase-field $u \in W^{1,2}(\mathcal{S})$, we define the solution s of the *diffuse Yamabe equation* for the positive phase as minimizer of

$$\min_{s \in W^{1,2}(\mathcal{S})} \mathcal{J}_{\varepsilon}[u, s] := \int_{\mathcal{S}} \frac{1}{2} (\chi(u) + \eta) |\nabla s|^2 - \chi(u) s K + \frac{1}{\varepsilon} (1 - \chi(u)) s^2 \, da. \tag{8.19}$$

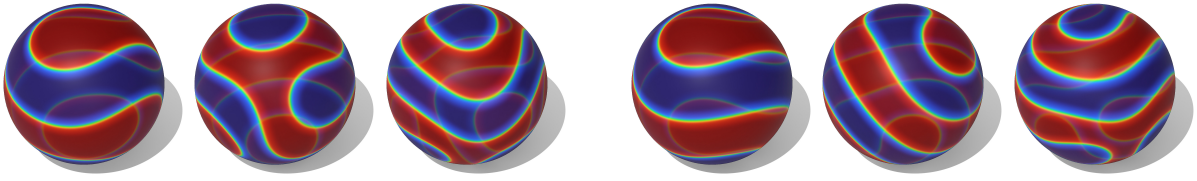


Figure 8.6: Solutions of (8.22) with distortion objective \mathcal{J}^{Yam} on a sphere for different weights of the objective. The three spheres on the left show results without the connectedness constraint leading to disconnected phases with small 'islands', while the three spheres on the right show results using the connectedness constraint. In both cases, the interface parameter was $\varepsilon = 0.023$ and the weights were (from left to right) $\omega = 5, 10, 15$ respectively.

The logarithmic scale factor of the negative phase is then given by solving the diffuse Yamabe equation for $-u$. In the functional \mathcal{Y}_ε , the factor $\chi(\pm u)$ localizes the minimization problem to one of the phases. The Dirichlet boundary conditions are taken care of by a quadratic penalty term on the complementary phase forcing s to zero, which is akin to Nitsche's method for boundary conditions. Furthermore, the η -damping ensures global regularity and coercivity. The ε^{-1} scaling of the penalty in (8.19) ensures an increasing adherence of the boundary conditions for $\varepsilon \rightarrow 0$.

The diffuse Yamabe equation is a quadratic problem in s , which means its solution can be computed by solving a linear problem. Furthermore, due to the uniform coercivity of the weighted Dirichlet energy and the smoothness of all coefficient functions $s[u]$ depends smoothly on u . Computing the Euler-Lagrange equation of (8.19), for fixed u , in direction $\theta \in W^{1,2}(S)$ yields

$$\partial_s \mathcal{Y}_\varepsilon[u, s](\theta) = \int_S (\chi(u) + \eta) \nabla s \cdot \nabla \theta - \chi(u) \theta K + \frac{1}{\varepsilon} (1 - \chi(u)) s \theta \, da.$$

Shape Optimization. Now, we define, as already announced, a new objective functional for our basic segmentation problem (8.13) based on the logarithmic scale factor $\pm s$. Multiple options for this objective have been discussed in [SC18], for example the Dirichlet energy of $\pm s$. However, we will only consider its squared L^2 -norm $\int_S s^2 \, da$, leading to the objective

$$\mathcal{J}^{\text{Yam}}[u] = \int_S s[u]^2 \, da + \int_S s[-u]^2 \, da,$$

where each of $s[\pm u]$ is the unique minimizer of the diffuse Yamabe equation (8.19) for the phase-fields $\pm u$.

This energy has a mechanical interpretation through the *Hencky strain*. If we interpret the conformal flattening of a segment as a deformation, then its Cauchy–Green strain tensor is a diagonal matrix with entries e^{2s} . The *Hencky strain* tensor is defined as the logarithm of the square-root of the Cauchy–Green strain tensor, i.e. $\log \sqrt{D\phi^T D\phi}$, which in our case is a diagonal matrix with entries s . Thus, s describes the Hencky or logarithmic strain of the flattening, which measures the geodesic distance of the deformation gradient to the special orthogonal group in the canonical Riemannian metric on the general linear group, cf. [NEOM13]. Because of this mechanical interpretation, we follow [SC18] and call $\int_S s^2 \, da$ the Hencky objective. This interpretation also motivates its usage in the objective \mathcal{J}^{Yam} , which is then well-suited for applications in design problems, where one is interested in manufacturing the surface by constructing the segments out of flat material (cf. [SC18]).

With this objective in place, we can finally formulate our PDE-constrained optimization problem

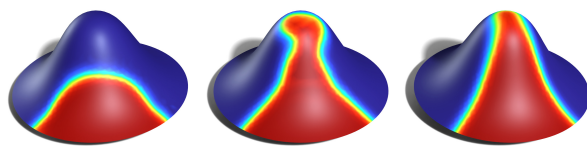


Figure 8.7: Three different time steps of the distortion minimization on a sinusoidal shaped surface. Interface width is given by $\varepsilon = 0.1$ and the distortion weight by $\omega = 10$. The result is similar to an analogous example by [SC18].

as

$$\begin{aligned} & \underset{u \in W^{1,2}(S)}{\text{minimize}} && \mathcal{P}_\varepsilon[u] + \omega \mathcal{J}^{\text{Yam}}[u] + \varepsilon^{-\kappa} (\mathcal{C}_\varepsilon[u] + \mathcal{C}_\varepsilon[-u]) \\ & \text{subject to} && s[\pm u] \text{ minimizing } \mathcal{Y}[\pm u, \cdot], \text{ and} \\ & && \mathcal{A}[u] = \frac{1}{2} \mathcal{H}^2(S). \end{aligned} \tag{8.20}$$

As before, we are looking for connected segments that cover half the surface that minimize a combination of interface length and distortion. The factor ω controls the trade-off between interface length and distortion when flattening. Let us remark that the ε^{-1} scaling of the penalty in (8.19) appears to be the appropriate choice. In our experiments, we observed that, in a descent scheme for (8.20), scaling the penalty with $\varepsilon^{-\beta}$ for $\beta > 1$ the distortion measure is reduced at the expense of a widening of the interface between the segments. Figure 8.6 shows a comparison of distortion optimal segmentation without and with connectedness constraint.

8.4 Discretization, Implementation, and Experiments

With the model in place, we are interested in solving the variational problems on complex surfaces. In this section, we will therefore discretize the different functionals, which will be a straightforward application of finite element methodology for the most part. This will yield an easy to implement method to compute distortion-minimizing hierarchical segmentations of complex triangular surfaces. Afterwards, we will show results of our method on a variety of examples and also demonstrate how it can be used to generate atlases for texture mapping applications.

8.4.1 Discretization

We employ simple affine finite elements to discretize problem (8.13). To this end, we will use a manifold triangle mesh $\mathcal{S}_h = (\mathbf{V}, \mathbf{E}, \mathbf{T})$ with vertices $\mathbf{V} \subset \mathbb{R}^3$, edges $\mathbf{E} \subset \mathbf{V} \times \mathbf{V}$, and faces $\mathbf{T} \subset \mathbf{V} \times \mathbf{V} \times \mathbf{V}$. Here, the subscript h indicates the grid size.

We first briefly recall the definition of the mass and stiffness matrix necessary for the Modica–Mortola functional. Let $\theta_1, \dots, \theta_{|\mathbf{V}|}$ be the nodal hat basis functions, i.e. they are piecewise linear functions on \mathcal{S}_h such that $\theta_i(v_j) = \delta_{ij}$. Then we consider the finite element space spanned by them, i.e. $\mathcal{P}_1(\mathcal{S}_h) := \text{span}\{\theta_1, \dots, \theta_{|\mathbf{V}|}\}$. Furthermore, we assume that the finite element approximation of u is in $\mathcal{P}_1(\mathcal{S}_h)$ with coefficient vector $\mathbf{u} \in \mathbb{R}^{|\mathbf{V}|}$, i.e. $u = \sum_{i=1}^{|\mathbf{V}|} \mathbf{u}_i \theta_i$. The double well potential of the phase-field $\Psi(u)$ is nonlinear and therefore not an element of $\mathcal{P}_1(\mathcal{S}_h)$, even if the phase-field u itself is. Projecting $\Psi(u)$ onto $\mathcal{P}_1(\mathcal{S}_h)$ yields the coefficient vector $\Psi_h(\mathbf{u}) \in \mathbb{R}^{|\mathbf{V}|}$, $\Psi_h(\mathbf{u})_i = \Psi(\mathbf{u}_i)$, i.e. the piecewise linear interpolation of its nodal values. We will use the approximation for this discretization of the bulk potential part of the Modica–Mortola functional instead of a higher-order quadrature scheme.

The stiffness matrix $\mathbf{S} \in \mathbb{R}^{|\mathbf{V}| \times |\mathbf{V}|}$ is the discretization of the symmetric, positive-semidefinite quadratic form $\int_S \nabla f \cdot \nabla g da$ and thus its entries are given by $\mathbf{S}_{ij} := \int_S \nabla \theta_i \cdot \nabla \theta_j da$. This yields the well-known cotan entries [Dzi88] for triangle meshes. Hence, the Dirichlet energy of the phase-field u is

discretized by $\mathbf{u}^T \mathbf{S} \mathbf{u}$. Moreover, the mass matrix $\mathbf{M} \in \mathbb{R}^{|\mathbf{V}| \times |\mathbf{V}|}$ discretizes the quadratic form $\int_S f g da$. We choose a diagonal mass matrix, sometimes also called lumped mass matrix. The lumped masses $\mathbf{m} \in \mathbb{R}^{|\mathbf{V}|}$ are given by averaging the areas of triangles adjacent to a vertex, i.e. $\mathbf{m}_v = \frac{1}{3} \sum_{\tau \in \mathbf{T}: v \in \tau} \mathbf{a}_\tau$, where \mathbf{a}_τ is the area of the triangle τ . The corresponding mass matrix is then the diagonal matrix $\mathbf{M} := \text{diag}(\mathbf{m})$. Using the lumped mass matrix, the integral of $\Psi_h(\mathbf{u})$ over the surface of \mathcal{S}_h can be approximated by a simple dot product $\mathbf{m}^T \Psi_h(\mathbf{u})$, yielding

$$\mathbf{P}_\varepsilon[\mathbf{u}] = \frac{\varepsilon}{2} \mathbf{u}^T \mathbf{S} \mathbf{u} + \frac{1}{\varepsilon} \mathbf{m}^T \Psi_h(\mathbf{u}) \quad (8.21)$$

as the discretization of the Modica–Mortola functional (8.1). Similarly the area $\mathcal{A}(u)$ is approximated by $\mathbf{m}^T \chi_h(\mathbf{u})$ with $\chi_h(\mathbf{u}) \in \mathbb{R}^{|\mathbf{V}|}$ defined via $\chi_h(\mathbf{u})_i = \chi(\mathbf{u}_i)$. With this, we have discretized all basic variational ingredients for phase-fields. Next, we will turn to the more involved elements, i.e. the connectedness constraint, the hierarchical approach, and the Yamabe constraint.

Connectedness Constraint. A convenient discretization of the connectedness constraint was introduced in [DW21], which is based on shortest-path searches in the dual graph of the triangle mesh. We briefly summarize it here for the sake of completeness. First, one computes for every triangle τ the average u_τ of the discrete phase-field u . Then every edge in the dual graph, i.e. every interior edge of the mesh, is equipped with a weight that is given by the product of the average value of $F_\varepsilon(u_\tau)$ of the two adjacent triangles and the average of their diameters. This way distances on the dual graph approximate distances given by $d^{F_\varepsilon}(u)$. Thus, to discretize the double integral $\mathbf{C}_\varepsilon[u]$, components of the dual graph consisting of triangles with zero distance to each other are computed, as well as the distances and shortest paths between these components. This grouping into connected components significantly reduces the necessary runtime, as it prevents computing unnecessary distances between all pairs of triangles. Then the discretization $\mathbf{C}_\varepsilon[\mathbf{u}]$ is given by the sum of pairwise products of integrated values of $H_\varepsilon(u)$ for every pair of distinct components weighted by their distance. From this, also a discrete gradient can be computed, which is supported along the aforementioned shortest paths. This yields an efficient discretization of the connectedness constraint and for further details on this discretization, we refer the reader to [DW21].

Overall, this leads to the finite-dimensional nonlinear optimization problem

$$\begin{aligned} & \underset{\mathbf{u} \in \mathbb{R}^{|\mathbf{V}|}}{\text{minimize}} && \mathbf{P}_\varepsilon[\mathbf{u}] + \varepsilon^{-\kappa} (\mathbf{C}_\varepsilon[\mathbf{u}] + \mathbf{C}_\varepsilon[-\mathbf{u}]) \\ & \text{subject to} && \mathbf{m}^T \chi_h(\mathbf{u}) = \frac{1}{2} a \end{aligned} \quad (8.22)$$

discretizing the basic segmentation problem (8.13) with the total surface area of \mathcal{S}_h given by $a = \text{tr} \mathbf{M}$.

As (8.22) is a nonlinear optimization problem, we will compute local minimizers that depend on the initialization of \mathbf{u} . To this end, we opted for the simple approach of initializing with random values in $[-1, 1]$. However, to reduce the number of necessary iterations, one could also use fast heuristics, such as clustering of normal directions, or input from users to generate the initialization.

Hierarchical Approach. Let $\mathbf{u} \in \mathbb{R}^{|\mathbf{V}|}$ be the FEM basis coefficients of the discrete counterpart of the phase-field u as before. To discretize the functionals of the hierarchical segmentation approach from Section 8.2.1, we first need to discretize the characteristic functions χ^α . Here, we simply approximate χ^α by a piecewise linear interpolation of its nodal values. To this end, we denote the vector of nodal values of χ^α by \mathbf{w}^α , i.e. $\mathbf{w}_j^\alpha = \chi^\alpha(v_j)$ for $v_j \in \mathbf{V}$. Note, that in theory, with each level the χ^α become polynomials of increasingly higher order even for piecewise linear u^α . However, they are (nearly) constant on large parts of the mesh and the transition regions of one indicator function is typically in an area where the indicator function of the parent is constant. Thus, a piecewise linear interpolation and corresponding quadrature should still be sufficiently accurate, which is supported by our experiments.

To discretize the Dirichlet part of the hierarchical Modica–Mortola functional (8.15), we consider the modified stiffness matrix $\mathbf{S}_{ij}^\alpha := \int_S \chi_\eta^\alpha \nabla \theta_i \cdot \nabla \theta_j da$. Then, the Dirichlet part is given by $\mathbf{u}^T \mathbf{S}^\alpha \mathbf{u}$. To compute the entries of \mathbf{S}^α , we use a simple midpoint quadrature, which leads to weighted cotan entries. We use nodal quadrature for the bulk part of the energy, which leads to the discrete approximation $\mathbf{m}^T (\mathbf{w}^\alpha \odot \Psi_h(\mathbf{u}))$. Here, we use \odot to denote the componentwise product of vectors as before. Finally, the penalty of values outside the current segment is also discretized using nodal quadrature yielding the approximation $(\mathbf{1} - \mathbf{w}^\alpha) \odot \mathbf{u} \odot \mathbf{u}$.

Together, the discrete counterpart to the adapted Modica-Mortola functional from (8.15) is given by

$$\mathbf{P}_{\varepsilon, \eta, \alpha}[\mathbf{u}] = \frac{\varepsilon}{2} \mathbf{u}^T \mathbf{S}^\alpha \mathbf{u} + \frac{1}{\varepsilon} \mathbf{m}^T (\mathbf{w}^\alpha \odot \Psi_h(\mathbf{u})) + \frac{1}{\sigma} \mathbf{m}^T ((\mathbf{1} - \mathbf{w}^\alpha) \odot \mathbf{u} \odot \mathbf{u}). \quad (8.23)$$

Furthermore, the discrete version of the hierarchical area functional is simply given by $\mathbf{m}^T (\mathbf{w}^\alpha \odot \chi_h(\mathbf{u}))$. Thus, our discrete optimization problem becomes

$$\begin{aligned} & \underset{\mathbf{u} \in \mathbb{R}^{|\mathcal{V}|}}{\text{minimize}} && \mathbf{P}_{\varepsilon, \eta, \alpha}[\mathbf{u}] \\ & \text{subject to} && \mathbf{m}^T (\mathbf{w}^\alpha \odot \chi_h(\mathbf{u})) = \frac{1}{2} \mathbf{m}^T \mathbf{w}^\alpha. \end{aligned} \quad (8.24)$$

In fact, in the spatially discrete model, the η -regularization of the perimeter functional turned out to be no longer necessary because of the regularizing effect of the finite element discretization.

Yamabe-based Segmentation. We discretize the logarithmic conformal factor via piecewise affine functions as we already did for phase-fields, i.e. we consider $s \in \mathcal{P}_1(\mathcal{S}_h)$ with coefficients $\mathbf{s} \in \mathbb{R}^{|\mathcal{V}|}$. Then, we need a discretization of the Gauß curvature for triangle meshes to discretize the (diffuse) Yamabe equation, for which we use the usual angle-defect discretization (see e.g. [CM03]) which we already introduced in Section 2.2 and briefly recall here. It is given by nodal values $\mathbf{K} \in \mathbb{R}^{|\mathcal{V}|}$, where $\mathbf{K}_i = \frac{1}{\mathbf{m}_i} (2\pi - \sum_{\tau: v_i \in \tau} \gamma_{\tau, v_i})$ for interior vertices v_i and $\mathbf{K}_i = \frac{1}{\mathbf{m}_i} (\pi - \sum_{\tau: v_i \in \tau} \gamma_{\tau, v_i})$ for vertices v_i on the boundary. Recall that $\gamma_{\tau, v}$ is the interior angle in facet τ at the vertex v .

In the discretization of the diffuse Yamabe equation (8.19), we directly consider the linear problem arising from finding solutions of the Euler-Lagrange equation. This will lead to a linear system $\mathbf{A}[\mathbf{u}]\mathbf{s} = \mathbf{b}[\mathbf{u}]$, with $\mathbf{A}[\mathbf{u}] \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{V}|}$ and $\mathbf{b}[\mathbf{u}] \in \mathbb{R}^{|\mathcal{V}|}$. To discretize the right hand side of this equation, i.e. the linear term in (8.19), we again use nodal quadrature, which leads to $\mathbf{b}[\mathbf{u}] := \mathbf{m} \odot \chi_h(\mathbf{u}) \odot K$. For the Dirichlet part of the linear system, we follow the same procedure as for the hierarchical approach and compute the entries of a modified stiffness matrix $\mathbf{S}_{ij}[\mathbf{u}] := \int_S (\chi(u) + \eta) \nabla \theta_i \cdot \nabla \theta_j da$ using midpoint quadrature. As before, this leads to a matrix with weighted cotan entries. Finally, for the quadratic penalty on the negative phase, we use again nodal quadrature leading to a diagonal matrix with entries $(1 - \chi_h(\mathbf{u})) \odot \mathbf{m}$.

Combined, this leads to the discretization of the diffuse Yamabe equation (8.19) by the linear system $\mathbf{A}[\mathbf{u}]\mathbf{s} = \mathbf{b}[\mathbf{u}]$, where

$$\mathbf{A}[\mathbf{u}] := \mathbf{S}[\mathbf{u}] + \frac{1}{\varepsilon} \text{diag}((1 - \chi_h(\mathbf{u})) \odot \mathbf{m}), \quad \mathbf{b}[\mathbf{u}] := \mathbf{m} \odot \chi_h(\mathbf{u}) \odot K.$$

The system matrix $\mathbf{A}[\mathbf{u}]$ is positive-definite due to the global coerciveness of the Dirichlet term and the fact that the diagonal penalty matrix eliminates its kernel. This means that we have a unique solution $\mathbf{s}[\mathbf{u}]$ of the discrete diffuse Yamabe equation for all $\mathbf{u} \in \mathbb{R}^{|\mathcal{V}|}$. Furthermore, the coefficients of the linear system smoothly depend on \mathbf{u} , which means that this solution $\mathbf{s}[\mathbf{u}]$ is also differentiable with respect to \mathbf{u} .

To use this discrete logarithmic scale factor in a shape optimization problem, we discretize \mathbf{J}^{Yam} using nodal quadrature, which yields the discrete objective

$$\mathbf{J}^{\text{Yam}}[\mathbf{u}] = \mathbf{s}[\mathbf{u}]^T \mathbf{M} \mathbf{s}[\mathbf{u}] + \mathbf{s}[-\mathbf{u}]^T \mathbf{M} \mathbf{s}[-\mathbf{u}].$$



Figure 8.8: Distortion minimizing segmentation of the ‘fertility’ model (courtesy of the AIM@SHAPE project). Interface width is given by $\varepsilon = 0.02$ and the distortion weight by $\omega = 0.1$. For this example, minimizing (8.20) yields segment boundaries following paths with negative Gauß curvature.

To minimize this objective $\mathbf{J}^{\text{Yam}}[\mathbf{u}]$, we need to compute in particular its derivatives. Here, we follow the general procedure of shape optimization calculus. Recall, that we have $\mathbf{s}[\mathbf{u}] = \mathbf{A}[\mathbf{u}]^{-1}\mathbf{b}[\mathbf{u}]$, which implies that the first variation is given by

$$\partial_{\mathbf{u}}\mathbf{s}[\mathbf{u}] = \mathbf{A}[\mathbf{u}]^{-1}(\partial_{\mathbf{u}}\mathbf{b}[\mathbf{u}] - \partial_{\mathbf{u}}\mathbf{A}[\mathbf{u}]\mathbf{s}[\mathbf{u}]).$$

By applying the chain rule and using that $\mathbf{A}[\mathbf{u}]$ is symmetric, we obtain

$$\begin{aligned} \partial_{\mathbf{u}}\mathbf{J}^{\text{Yam}}[\mathbf{u}] &= -2(\partial_{\mathbf{u}}\mathbf{b}[\mathbf{u}] - \partial_{\mathbf{u}}\mathbf{A}[\mathbf{u}]\mathbf{s}[\mathbf{u}])^T \mathbf{A}[\mathbf{u}]^{-1}(\mathbf{M}\mathbf{s}[\mathbf{u}]) \\ &\quad - 2(\partial_{\mathbf{u}}\mathbf{b}[-\mathbf{u}] - \partial_{\mathbf{u}}\mathbf{A}[-\mathbf{u}]\mathbf{s}[-\mathbf{u}])^T \mathbf{A}[-\mathbf{u}]^{-1}(\mathbf{M}\mathbf{s}[-\mathbf{u}]). \end{aligned}$$

Hence, computing the derivative mainly amounts to solving two linear systems, where the matrices are the same as for the diffuse scale factors. If we use direct solvers such as a Cholesky decomposition, we can reuse the factorization and only have to compute additional back substitutions. With this, we have introduced discretizations for all elements of our hierarchical segmentation approach with distortion minimization using the Yamabe equation as PDE-constraint.

8.4.2 Results and Applications

The described method was implemented in C++ and tested on variety of examples, which were remeshed isotropically using the method described by [BK04]. The Eigen library [GJ+10] was used for numerical linear algebra and CHOLMOD [CDHR08] and UMFPACK [Dav04] from the SuiteSparse collection were used as direct linear solvers. For unconstrained optimization problems, i.e. when using a quadratic penalty for the area constraint, libLBFGS, a limited memory BFGS implementation based on the work [Noc80], was employed. For constrained optimization problems, the open-source software package Ipopt [WB06] implementing an interior point method was used.

In Figure 8.2 on the left, we see the influence of the interface width on the optimal phase-field in the basic segmentation problem (8.22) using only the area constraint. On the right, we see that this segmentation problem can indeed lead to disconnected phases, which are joined together after adding the connectedness penalty to the objective. In the hierarchical approach, we can see a similar behavior in Figure 8.5.

Adding the (discrete) Yamabe equation as PDE-constraint together with the objective \mathbf{J}^{Yam} allows to also minimize distortion, where the trade-off between interface length and distortion is controlled by the factor ω . In Figure 8.6, the effect of this factor ω is shown when segmenting a sphere. Furthermore, in Figure 8.7, we show steps of a gradient-based optimization of the problem without the area constraint, which leads to results similar to [SC18]. Finally, we see another example of a distortion minimizing segmentation for a more complex surface in Figure 8.8, which yields segment boundaries along paths with negative Gauß curvature in this example.

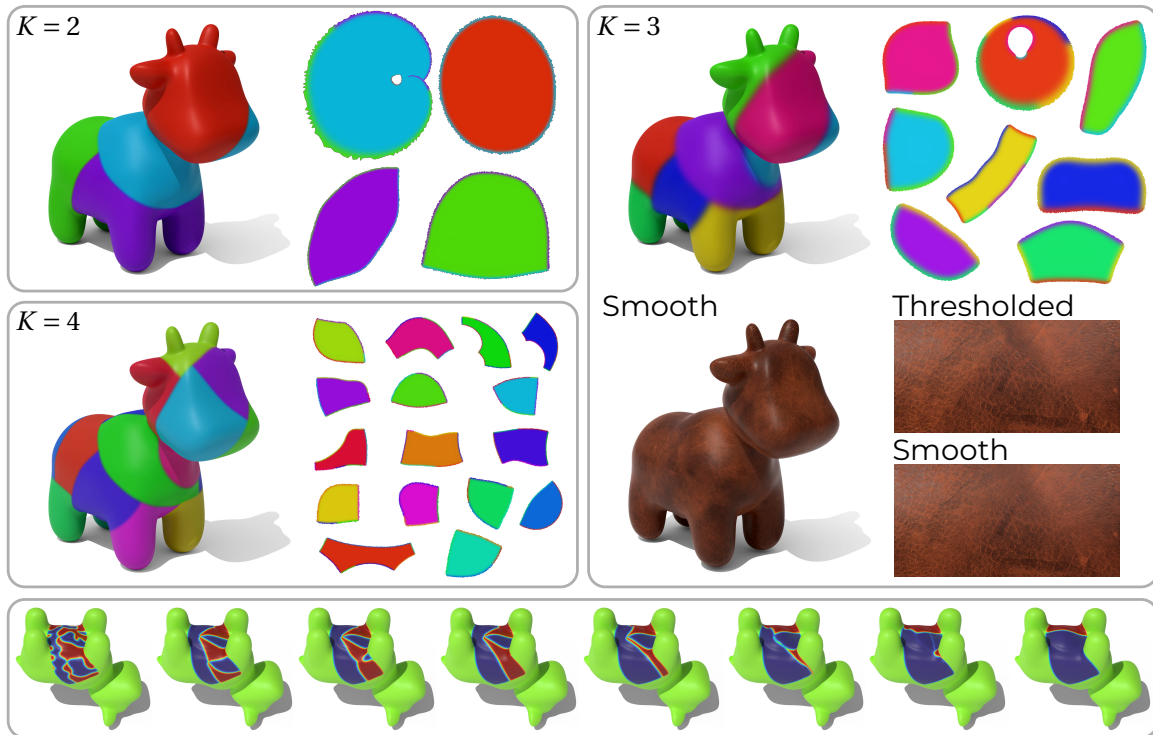


Figure 8.9: Hierarchical segmentation and atlas generation for $K = 2, 3$ and 4 . The scalar functions χ^α are multiplied with RGB color values identifying the different segments. Thereby, the diffuse glueing of the chart domains is visible via the color blending close to the boundary of the chart domains U^α and their flattened counterparts $\varphi^\alpha(U^\alpha)$. On the lower right, texturing using the smooth phase-field based blending is compared with a conventional thresholding approach by zooming in on a textured surface. Note that, in the texturing application, we deliberately used a wider interface width ε to generate smoother transitions between the segments. On the bottom, we show different time steps of the descent scheme for one of the hierarchical segmentation steps on level $K = 4$.

Atlas Generation. To generate atlases from our hierarchical segmentation, we pick up the partition of unity $\{\chi^\alpha\}_{\alpha \in \{-1, +1\}^K}$ on level K of the hierarchy and consider their support as chart domains. However, given that optimal profiles for phase-fields minimizing the Modica–Mortola functional are of tanh-type (see above), the χ^α will get close to zero but never actually become zero for $\varepsilon > 0$. This means, that all χ^α have effectively global support, which would mean we always have to flatten the whole surface, which is of course not our intention. Thus, we introduce a shifted diffuse indicator function given as

$$\chi_\delta(u) := \begin{cases} 0 & u \leq -1 + \delta \\ \frac{1}{4(1-\delta)^3} (u + 1 - \delta)^2 (2(1-\delta) - u) & -1 + \delta < u < 1 - \delta \\ 1 & u \geq 1 - \delta \end{cases}$$

which reaches its global extrema earlier controlled by a small parameter δ with $0 < \delta \ll \varepsilon$. As before, it still holds that $\chi_\delta(u) + \chi_\delta(-u) \equiv 1$ and thus we can construct partitions of unity $\{\chi_\delta^\alpha\}_{\alpha \in \{-1, +1\}^K}$ for all levels K the same way as in Section 8.2.1. Then we define $U^\alpha := \{x \in \mathcal{S} \mid \chi_\delta^\alpha(x) \neq 0\}$ for all levels K as chart domains forming covers of \mathcal{S} . To complete the atlas, we have to compute the actual chart maps $\varphi^\alpha: U^\alpha \rightarrow \mathbb{R}^2$. One can use any state-of-the-art algorithm for conformal flattening for this task. In the implementation, LSCM [LPRM02] was chosen due to its uncomplicated availability, while other, nonlinear algorithms such as CETM [SSP08] would also be possible. In Figure 8.9, we show the resulting atlases for different K .

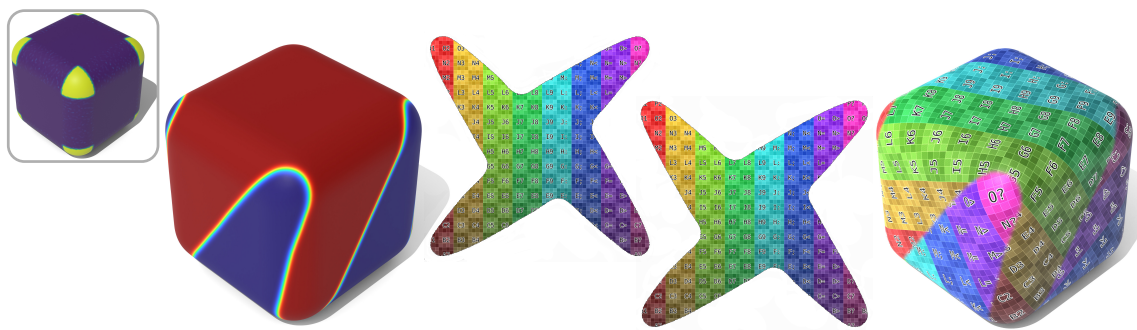


Figure 8.10: *Segmenting and flattening a cube with rounded corners. The cube exhibits isolated regions of Gauß curvature at its rounded corners, which leads to a segmentation with boundaries not following the edges of the cube. Instead, the boundaries are such that their length in the isolated region is maximized while keeping the total length under control. From left to right: The discrete Gauß curvature shown as color map with dark blue indicating approximately zero curvature and yellow approximately 3.12 curvature. Solutions of (8.22) with distortion objective J^{Yam} and distortion weight $\omega = 10$. Conformally flattened segments in the plane with texture applied. Textures mapped to rounded cube, where perpendicular lines show the conformality and equally sized squares show the low distortion of the charts.*

Using the chart maps, any function f defined on \mathbb{R}^2 can be lifted to the surface by the pull-back $f^\alpha := f \circ \varphi^\alpha$. This way we obtain multiple functions defined on the surface, one for each chart. These functions can then be blended via

$$f^S := \sum_{\alpha} \chi_{\delta}^{\alpha} f^{\alpha}$$

using the χ_{δ}^{α} as weighting functions, yielding smooth transitions due to them being a smooth partition of unity.

This blending and the underlying atlas can be useful for different applications requiring a domain decomposition, such as partition of unity methods for finite element simulations and ODE solvers on surfaces. The hierarchical nature of our segmentation could be used to devise multi-resolution approaches for these applications, e.g. in the context of the mentioned Tausch–White wavelets. As a simple proof of concept, here, we only demonstrate its use for an improved texture blending that can outperform conventional thresholding based approaches which segment the mesh into disjoint patches with sharp transitions between them. Concretely, we consider maps $f: \mathbb{R}^2 \rightarrow \mathbb{R}^c$ with \mathbb{R}^c being some color space as demonstrated in Figure 8.9. A similar approach was used in [PB00] for a seamless blending of textures on surfaces using subdivision methodology.

This texture transfer can also be used to visually investigate the scale factor of a conformal flattening by observing the distortion of a texture mapped from the plane. For example, in Figure 8.1, we see on the left that only minimizing the perimeter leads to flattenings incurring severe distortions of mapped textures. On the right, we see that adding the Yamabe constraint and corresponding objective indeed strongly reduces these distortions. Note, that in both cases, the maps to the plane are conformal, which can be seen by the perpendicular intersections of lines in the texture mapped to the surface. Finally, Figure 8.10 shows another example for low distortion segmentations and close to conformal texture transfer from associated chart domains onto the surface, which has highly concentrated Gauß curvature.

8.5 Conclusion and Outlook

In this chapter, we have introduced a phase-field model for the variational segmentation of surfaces. We started with a basic segmentation problem by adapting the Modica–Mortola functional together with a corresponding connectedness constraint from the Euclidean case to curved surfaces. This allowed us to formulate a variational problem for dividing the surface into two connected and equally-sized segments. We extended this approach to divide the surface into a larger number of segments via a hierarchical approach, where the variational problems for further dividing a segment is formulated using the diffuse indicator functions. Finally, we introduced added distortion minimization to the model by using a diffuse version of the Yamabe equation as PDE-constraint and penalizing the computed logarithmic scale factor. For all of this, we introduced a straightforward discretization via affine finite elements that proved to be effective in various experiments.

Our approach has the advantage that it leads to a PDE-constrained optimization problem that is straightforward to discretize and describes the segmentation problem in an end-to-end diffuse way. This means we do not have to worry about explicitly cutting the mesh or how to generate appropriate descent directions as was the case in previous approaches. The downside of this is, that the mesh needs to be of high-resolution for narrow diffuse interfaces and larger weights of the distortion objective. Furthermore, the triangle size has to be mostly isotropic to prevent needing to choose a unnecessarily large ε . This requires access to a process able to generate such high quality meshes, e.g. a subdivision surface model.

This is, however, a limitation that is shared by many PDE-based mesh processing approaches. One possible remedy for this are intrinsic triangulations [BS07], which overlay another triangle mesh over a given one allowing to control different characteristics of these meshes while preserving the extrinsic geometry. Recently, there has been interest in constructing efficient data structures for such triangulations, cf. [SSC19]. Thus they might be an interesting tool to improve the implementation of our method to also work on low-resolution and low-quality meshes.

We envision that our general (hierarchical) phase-field approach might also be interesting for other segmentation problems on surfaces. For example, one could ask for segments with consistent normal directions or convex segments, which is interesting for collision detection applications. In this sense, our current results could be understood as a promising proof-of-concept for phase-field-based approaches in geometry processing.

Regarding our distortion optimization, it would be interesting to extend it to other phase-field models. On the one hand, we could consider vector-valued phase-field that divide the surface into more than one segment at a time. This would, for example, be interesting in fabrication applications, where one wants to divide the surface into many segments than can be flattened without much distortion but is not interested in the hierarchical nature of our approach. On the other hand, it would be interesting to investigate the corresponding free interface problem, i.e. considering actual cuts instead of segments. In terms of phase-fields, this would mean to consider the Ambrosio–Tortorelli functional for the diffuse perimeter and adapt the penalty in the diffuse Yamabe equation.

Finally, our approach is currently also missing a rigorous theoretical investigation. In the diffuse case, it would be worthwhile to rigorously prove the existence and uniqueness of solutions for the diffuse Yamabe equation, and prove the well-posedness of as well as existence of minimizers of the PDE-constrained shape optimization problem. Afterwards, one could investigate convergence to a sharp interface limit again first for the diffuse Yamabe equation and then the shape optimization problem. All this leaves open a number of interesting avenues for potential future work.

Bibliography

- [Abi81] William Abikoff. “The Uniformization Theorem”. In: *The American Mathematical Monthly* 88 (8 1981), p. 574. ISSN: 0002-9890. DOI: 10.2307/2320507.
- [AC72] Samuel M. Allen and John W. Cahn. “Ground state structures in ordered binary alloys with second neighbor interactions”. In: *Acta Metallurgica* 20.3 (1972), pp. 423–433. ISSN: 0001-6160. DOI: 10.1016/0001-6160(72)90037-5.
- [AHK14] Daniel Alm, Helmut Harbrecht, and Ulf Krämer. “The \mathcal{H}^2 -wavelet method”. In: *Journal of Computational and Applied Mathematics* 267 (2014), pp. 131–159. ISSN: 0377-0427. DOI: 10.1016/j.cam.2014.01.030.
- [AZv19a] Felix Ambellan, Stefan Zachow, and Christoph von Tycowicz. “A Surface-Theoretic Approach for Statistical Shape Modeling”. In: *Lecture Notes in Computer Science*. Vol. 11767. 2019, pp. 21–29. ISBN: 978-3-030-32250-2. DOI: 10.1007/978-3-030-32251-9_3.
- [AZv19b] Felix Ambellan, Stefan Zachow, and Christoph von Tycowicz. “An As-Invariant-As-Possible $GL_+(3)$ -Based Statistical Shape Model”. In: *Lecture Notes in Computer Science*. Vol. 11846. 2019, pp. 219–228. ISBN: 978-3-030-33225-9. DOI: 10.1007/978-3-030-33226-6_23.
- [AR18] Nina Amenta and Carlos Rojas. “Dihedral Rigidity and Deformation”. In: *Proceedings Canadian Conference on Computational Geometry*. 2018, pp. 219–226. arXiv: 1810.01546.
- [AKJ08] Steven S. An, Theodore Kim, and Doug L. James. “Optimizing cubature for efficient integration of subspace deformations”. In: *ACM Transactions on Graphics* 27.5 (2008), pp. 1–10. ISSN: 1557-7368. DOI: 10.1145/1409060.1409118.
- [ASK+05] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. “SCAPE: Shape Completion and Animation of People”. In: *ACM Transactions on Graphics* 24.3 (2005), pp. 408–416. ISSN: 0730-0301. DOI: 10.1145/1073204.1073207.
- [AFS06] Marco Attene, Bianca Falcidieno, and Michela Spagnuolo. “Hierarchical mesh segmentation based on fitting primitives”. In: *The Visual Computer* 22.3 (2006), pp. 181–193. ISSN: 0178-2789. DOI: 10.1007/s00371-006-0375-x.
- [Aub98] Thierry Aubin. *Some Nonlinear Problems in Riemannian Geometry*. Springer Monographs in Mathematics. Springer Berlin, Heidelberg, 1998. ISBN: 978-3-662-13006-3. DOI: 10.1007/978-3-662-13006-3.
- [Aze20] Hideyuki Azegami. *Shape Optimization Problems*. Vol. 164. Springer Optimization and Its Applications. Singapore: Springer Singapore, 2020. ISBN: 978-981-15-7618-8. DOI: 10.1007/978-981-15-7618-8.

- [BGMM03] Jayanth R. Banavar, Oscar Gonzalez, John H. Maddocks, and Amos Maritan. “Self-Interactions of Strands and Sheets”. In: *Journal of Statistical Physics* 110 (2003), pp. 35–50. ISSN: 0022-4715. DOI: 10.1023/A:1021010526495.
- [BVGP09] Ilya Baran, Daniel Vlastic, Eitan Grinspun, and Jovan Popovic. “Semantic Deformation Transfer”. In: *ACM Transactions on Graphics* 28.3 (2009), 36:1–36:6. DOI: 10.1145/1531326.1531342.
- [BMP22] Sören Bartels, Frank Meyer, and Christian Palus. “Simulating Self-Avoiding Isometric Plate Bending”. In: *SIAM Journal on Scientific Computing* 44.3 (2022), A1475–A1496. ISSN: 1064-8275. DOI: 10.1137/21M1440001.
- [BHM11] Martin Bauer, Philipp Harms, and Peter W. Michor. “Sobolev Metrics on Shape Space of Surfaces”. In: *Journal of Geometric Mechanics* 3.4 (2011), pp. 389–438. ISSN: 1941-4889. DOI: 10.3934/jgm.2011.3.389.
- [BHM12a] Martin Bauer, Philipp Harms, and Peter W. Michor. “Almost Local Metrics on Shape Space of Hypersurfaces in \mathbb{R}^n -Space”. In: *SIAM Journal on Imaging Sciences* 5 (2012), pp. 244–310. DOI: 10.1137/100807983.
- [BHM12b] Martin Bauer, Philipp Harms, and Peter W. Michor. “Curvature Weighted Metrics on Shape Space of Hypersurfaces in \mathbb{R}^n -Space”. In: *Differential Geometry and its Applications* 30.1 (2012), pp. 33–41. DOI: 10.1016/j.difgeo.2011.10.002.
- [BHM12c] Martin Bauer, Philipp Harms, and Peter W. Michor. “Sobolev Metrics on Shape Space, II: Weighted Sobolev Metrics and Almost Local Metrics”. In: *Journal of Geometric Mechanics* 4.4 (2012), pp. 365–383. DOI: 10.3934/jgm.2012.4.365.
- [BHM20] Martin Bauer, Philipp Harms, and Peter W. Michor. “Fractional Sobolev metrics on spaces of immersions”. In: *Calculus of Variations and Partial Differential Equations* 59.2 (2020), p. 62. ISSN: 1432-0835. DOI: 10.1007/s00526-020-1719-5.
- [Bec07] Amir Beck. “Quadratic Matrix Programming”. In: *SIAM Journal on Optimization* 17.4 (2007), pp. 1224–1238. ISSN: 1052-6234. DOI: 10.1137/05064816X.
- [BMTY05] M. Faisal Beg, Michael I. Miller, Alain Trounev, and Laurent Younes. “Computing Large Deformation Metric Mappings via Geodesic Flows of Diffeomorphisms”. In: *International Journal of Computer Vision* 61.2 (2005), pp. 139–157. ISSN: 1573-1405. DOI: 10.1023/B:VISI.0000043755.93987.aa.
- [BGB08] Mirela Ben-Chen, Craig Gotsman, and Guy Bunin. “Conformal Flattening by Curvature Prescription and Metric Scaling”. In: *Computer Graphics Forum* 27.2 (2008), pp. 449–458. DOI: 10.1111/j.1467-8659.2008.01142.x.
- [BCM04] Michal Beneš, Vladimír Chaloupecký, and Karol Mikula. “Geometrical image segmentation by the Allen–Cahn equation”. In: *Applied Numerical Mathematics* 51 (2-3 2004), pp. 187–205. ISSN: 0168-9274. DOI: 10.1016/J.APNUM.2004.05.001.
- [BSV14] Peter Benner, Ekkehard Sachs, and Stefan Volkwein. “Model Order Reduction for PDE Constrained Optimization”. In: *Trends in PDE Constrained Optimization*. Ed. by Günter Leugering, Peter Benner, Sebastian Engell, Andreas Griewank, Helmut Harbrecht, Michael Hinze, Rolf Rannacher, and Stefan Ulbrich. Vol. 165. Series Title: International Series of Numerical Mathematics. Cham: Springer International Publishing, 2014, pp. 303–326. ISBN: 978-3-319-05083-6. DOI: 10.1007/978-3-319-05083-6_19.
- [BER15] Benjamin Berkels, Alexander Effland, and Martin Rumpf. “Time Discrete Geodesic Paths in the Space of Images”. In: *SIAM Journal on Imaging Sciences* 8.3 (2015), pp. 1457–1488. DOI: 10.1137/140970719.

- [BF12] Andrea L. Bertozzi and Arjuna Flenner. “Diffuse Interface Models on Graphs for Classification of High Dimensional Data”. In: *Multiscale Modeling & Simulation* 10.3 (2012), pp. 1090–1118. ISSN: 1540-3459. DOI: 10.1137/11083109X.
- [Ber99] Dimitri P. Bertsekas. *Nonlinear Programming*. 2nd. Belmont, MA: Athena Scientific, 1999. ISBN: 1-886529-00-0.
- [BKRB18] Karthik Bharath, Sebastian Kurtek, Arvind Rao, and Veerabhadran Baladandayuthapani. “Radiologic Image-Based Statistical Shape Analysis of Brain Tumours”. In: *Journal of the Royal Statistical Society: Series C (Applied Statistics)* 67.5 (2018), pp. 1357–1378. DOI: 10.1111/rssc.12272.
- [Bla13] Simon Blatt. “The energy spaces of the tangent point energies”. In: *Journal of Topology and Analysis* 5.3 (2013), pp. 261–270. ISSN: 1793-5253. DOI: 10.1142/S1793525313500131.
- [BS07] Alexander I. Bobenko and Boris A. Springborn. “A Discrete Laplace–Beltrami Operator for Simplicial Surfaces”. In: *Discrete & Computational Geometry* 38.4 (2007), pp. 740–756. ISSN: 0179-5376. DOI: 10.1007/s00454-007-9006-1.
- [BVGW16] Friedrich Bös, Etienne Vouga, Omer Gottesman, and Max Wardetzky. “On the Incompressibility of Cylindrical Origami Patterns”. In: *Journal of Mechanical Design* 139.2 (2016), pp. 21404–21409. ISSN: 1050-0472. DOI: 10.1115/1.4034970.
- [BK04] Mario Botsch and Leif Kobbelt. “A remeshing approach to multiresolution modeling”. In: *Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry processing*. SGP ’04. New York, New York, USA: Association for Computing Machinery, 2004, p. 185. ISBN: 3905673134. DOI: 10.1145/1057432.1057457.
- [BS08] Mario Botsch and Olga Sorkine. “On Linear Variational Surface Deformation Methods”. In: *IEEE Transactions on Visualization and Computer Graphics* 14.1 (2008), pp. 213–230. ISSN: 1077-2626. DOI: 10.1109/TVCG.2007.1054.
- [BC03] Blaise Bourdin and Antonin Chambolle. “Design-dependent loads in topology optimization”. In: *ESAIM: Control, Optimisation and Calculus of Variations* 9 (2003), pp. 19–48. ISSN: 1292-8119. DOI: 10.1051/COCV:2002070.
- [BPC+11] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. “Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers”. In: *Foundations and Trends in Machine Learning* 3 (2011), pp. 1–122. DOI: 10.1561/22000000016.
- [BM73] Jerome Bracken and James T. McGill. “Mathematical Programs with Optimization Problems in the Constraints”. In: *Operations Research* 21 (1 1973), pp. 37–44. ISSN: 0030-364X. DOI: 10.1287/opre.21.1.37.
- [Bra02] Andrea Braides. *Gamma-Convergence for Beginners*. Oxford Lecture Series in Mathematics and Its Applications. Oxford, New York: Oxford University Press, 2002. ISBN: 978-0-19-850784-0.
- [BH17] Christopher Brandt and Klaus Hildebrandt. “Compressed vibration modes of elastic bodies”. In: *Computer Aided Geometric Design* 52-53 (2017), pp. 297–312. ISSN: 0167-8396. DOI: 10.1016/j.cagd.2017.03.004.
- [BvH16] Christopher Brandt, Christoph von Tycowicz, and Klaus Hildebrandt. “Geometric Flows of Curves in Shape Space for Processing Motion of Deformable Objects”. In: *Computer Graphics Forum* 35.2 (2016), pp. 295–305. DOI: 10.1111/cgf.12832.

- [BSBW14] Alan Brunton, Augusto Salazar, Timo Bolkart, and Stefanie Wuhler. “Review of Statistical Shape Spaces for 3D Data with Comparative Analysis for Human Faces”. In: *Computer Vision and Image Understanding* 128 (2014), pp. 1–17. ISSN: 1077-3142. DOI: 10.1016/j.cviu.2014.05.005.
- [BO95] Gregory Buck and Jeremy Orloff. “A simple energy function for knots”. en. In: *Topology and its Applications* 61.3 (1995), pp. 205–214. ISSN: 0166-8641. DOI: 10.1016/0166-8641(94)00024-W.
- [BCC+21] Johanna Burtscheidt, Matthias Claus, Sergio Conti, Martin Rumpf, Josua Sassen, and Rüdiger Schultz. “A Pessimistic Bilevel Stochastic Problem for Elastic Shape Optimization”. In: *Mathematical Programming* (2021). ISSN: 1436-4646. DOI: 10.1007/s10107-021-01736-w.
- [BCD20] Johanna Burtscheidt, Matthias Claus, and Stephan Dempe. “Risk-Averse Models in Bilevel Stochastic Linear Programming”. In: *SIAM Journal on Optimization* 30.1 (2020), pp. 377–406. ISSN: 1052-6234. DOI: 10.1137/19M1242240.
- [CH58] John W. Cahn and John E. Hilliard. “Free Energy of a Nonuniform System. I. Interfacial Free Energy”. In: *The Journal of Chemical Physics* 28.2 (1958), pp. 258–267. ISSN: 0021-9606. DOI: 10.1063/1.1744102.
- [Car76] Manfredo Perdigão do Carmo. *Differential geometry of curves and surfaces*. Prentice Hall, 1976.
- [Car92] Manfredo Perdigão do Carmo. *Riemannian Geometry*. Birkhäuser, 1992. ISBN: 978-3-7643-3490-1.
- [CR74] Edwin Catmull and Raphael Rom. “A class of local interpolating splines”. In: *Computer Aided Geometric Design*. Academic Press, 1974, pp. 317–326.
- [CDST97] Bernard Chazelle, David P. Dobkin, Nadia Shouraboura, and Ayellet Tal. “Strategies for polyhedral surface decomposition: An experimental study”. In: *Computational Geometry: Theory and Applications* 7.5-6 (1997), pp. 327–342. ISSN: 0925-7721. DOI: 10.1016/S0925-7721(96)00024-7.
- [Che03] Long Qing Chen. “Phase-Field Models for Microstructure Evolution”. In: *Annual Review of Materials Research* 32 (2003), pp. 113–140. ISSN: 0084-6600. DOI: 10.1146/ANNUREV.MATSCI.32.112001.132041.
- [CDHR08] Yanqing Chen, Timothy A. Davis, William W. Hager, and Sivasankaran Rajamanickam. “Algorithm 887: CHOLMOD, Supernodal Sparse Cholesky Factorization and Update/Downdate”. In: *ACM Transactions on Mathematical Software* 35.3 (2008), 22:1–22:14. ISSN: 0098-3500. DOI: 10.1145/1391989.1391995.
- [CL09] Patrick Cheridito and Tianhui Li. “Risk measures on Orlicz hearts”. In: *Mathematical Finance* 19.2 (2009), pp. 189–214. DOI: 10.1111/j.1467-9965.2009.00364.x.
- [CKPS18] Albert Chern, Felix Knöppel, Ulrich Pinkall, and Peter Schröder. “Shape from metric”. In: *ACM Transactions on Graphics* 37.4 (2018), pp. 1–17. ISSN: 0730-0301. DOI: 10.1145/3197517.3201276.
- [CPW01] Snorre Christiansen, Michael Patriksson, and Laura Wynter. “Stochastic bilevel programming in structural optimization”. In: *Structural and Multidisciplinary Optimization* 21.5 (2001), pp. 361–371. ISSN: 1615-147X. DOI: 10.1007/s001580100115.
- [Cia90] Philippe G. Ciarlet. “Mathematical elasticity, volume I: Three-dimensional elasticity”. In: *Acta Applicandae Mathematicae* 18.2 (1990), pp. 190–195. ISSN: 0167-8019. DOI: 10.1007/BF00046568.

- [Cia00] Philippe G. Ciarlet. *Mathematical Elasticity, Volume III: Theory of shells*. North-Holland, 2000.
- [CLR04] Ulrich Clarenz, Nathan Litke, and Martin Rumpf. “Axioms and Variational Problems in Surface Parameterization”. In: *Computer Aided Geometric Design* 21.8 (2004), pp. 727–749. ISSN: 0167-8396. DOI: 10.1016/j.cagd.2004.07.005.
- [CUH16] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. “Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)”. In: *International Conference on Learning Representations*. 2016.
- [CM03] David Cohen-Steiner and Jean-Marie Morvan. “Restricted Delaunay Triangulations and Normal Cycle”. In: *Proceedings of the Nineteenth Conference on Computational Geometry - SCG '03*. New York, New York, USA: Association for Computing Machinery, 2003, p. 312. ISBN: 1-58113-663-3. DOI: 10.1145/777792.777839.
- [CdDS13] Keenan Crane, Fernando de Goes, Mathieu Desbrun, and Peter Schröder. “Digital Geometry Processing with Discrete Exterior Calculus”. In: *ACM SIGGRAPH 2013 Courses*. New York, NY, USA: ACM, 2013. DOI: 10.1145/2504435.2504442.
- [CSSR23] Keenan Crane, Josua Sassen, Henrik Schumacher, and Martin Rumpf. “Repulsive Shells”. in preparation. 2023.
- [Dav04] Timothy A. Davis. “Algorithm 832: UMFPACK V4.3—an Unsymmetric-pattern Multifrontal Method”. In: *ACM Transactions on Mathematical Software* 30.2 (2004), pp. 196–199. ISSN: 0098-3500. DOI: 10.1145/992200.992206.
- [DMNP22] François Dayrens, Simon Masnou, Matteo Novaga, and Marco Pozzetta. “Connected perimeter of planar sets”. In: *Advances in Calculus of Variations* 15.2 (2022), pp. 213–234. ISSN: 1864-8266. DOI: 10.1515/ACV-2019-0050.
- [Dem02] Stephan Dempe. *Foundations of bilevel programming*. Dordrecht [u.a.]: Kluwer Academic Publishers, 2002. ISBN: 978-0-306-48045-4.
- [DIN17] Stephan Dempe, Sergey Ivanov, and Andrey Naumov. “Reduction of the bilevel stochastic optimization problem with quantile objective function to a mixed-integer problem”. In: *Applied Stochastic Models in Business and Industry* 33.5 (2017), pp. 544–554. ISSN: 1524-1904. DOI: 10.1002/asmb.2254.
- [DKPK15] Stephan Dempe, Vyacheslav Kalashnikov, Gerardo A. Pérez-Valdés, and Nataliya Kalashnykova. *Bilevel Programming Problems*. Springer Berlin Heidelberg, 2015. ISBN: 978-3-662-45826-6. DOI: 10.1007/978-3-662-45827-3.
- [DKT08] Mathieu Desbrun, Eva Kanso, and Yiyong Tong. “Discrete differential forms for computational modeling”. In: *Oberwolfach Seminars: Discrete differential geometry*. Vol. 38. Oberwolfach Semin. Birkhäuser, Basel, 2008, pp. 287–324.
- [DLW17] Patrick W. Dondl, Antoine Lemenant, and Stephan Wojtowytsch. “Phase Field Models for Thin Elastic Structures with Topological Constraint”. In: *Archive for Rational Mechanics and Analysis* 223.2 (2017), pp. 693–736. ISSN: 0003-9527. DOI: 10.1007/s00205-016-1043-6.
- [DNWW19] Patrick W. Dondl, Matteo Novaga, Benedikt Wirth, and Stephan Wojtowytsch. “A Phase-field Approximation of the Perimeter under a Connectedness Constraint”. In: *SIAM Journal on Mathematical Analysis* 51.5 (2019), pp. 3902–3920. ISSN: 0036-1410. DOI: 10.1137/18M1225197.
- [DW21] Patrick W. Dondl and Stephan Wojtowytsch. “Keeping it together: A phase-field version of path-connectedness and its implementation”. In: *Journal of Algorithms & Computational Technology* 15 (2021). DOI: 10.1177/17483026211054342.

- [DF20] Qiang Du and Xiaobing Feng. “The phase field method for geometric moving interfaces and their numerical approximations”. In: *Handbook of Numerical Analysis* 21 (2020), pp. 425–508. ISSN: 1570-8659. DOI: 10.1016/BS.HNA.2019.05.001.
- [DJT11] Qiang Du, Lili Ju, and Li Tian. “Finite element approximation of the Cahn–Hilliard equation on surfaces”. In: *Computer Methods in Applied Mechanics and Engineering* 200.29-32 (2011), pp. 2458–2470. ISSN: 0045-7825. DOI: 10.1016/j.cma.2011.04.018.
- [DVTM16] Levi H. Dudte, Etienne Vouga, Tomohiro Tachi, and L. Mahadevan. “Programming Curvature Using Origami Tessellations”. In: *Nature Materials* 15 (2016), pp. 583–588. DOI: 10.1038/nmat4540.
- [DGM98] Paul Dupuis, Ulf Grenander, and Michael I. Miller. “Variational Problems on Flows of Diffeomorphisms for Image Matching”. In: *Quarterly of Applied Mathematics* (1998), pp. 587–600. DOI: 10.1090/qam/1632326.
- [Dzi88] Gerhard Dziuk. “Finite elements for the Beltrami operator on arbitrary surfaces”. In: *Partial Differential Equations and Calculus of Variations*. Ed. by S. Hildebrandt and R. Leis. Vol. 1357. Lecture Notes in Math. Springer, Berlin, 1988, pp. 142–155.
- [DE13] Gerhard Dziuk and Charles M. Elliott. “Finite element methods for surface PDEs”. In: *Acta Numerica* 22 (2013), pp. 289–396. ISSN: 1474-0508. DOI: 10.1017/S096249291300056.
- [EHRW22] Alexander Effland, Behrend Heeren, Martin Rumpf, and Benedikt Wirth. “Consistent curvature approximation on Riemannian shape spaces”. In: *IMA Journal of Numerical Analysis* 42.1 (2022), pp. 78–106. ISSN: 0272-4979, 1464-3642. DOI: 10.1093/imanum/draa092.
- [EC20] Marvin Eisenberger and Daniel Cremers. “Hamiltonian Dynamics for Real-World Shape Interpolation”. In: *European Conference on Computer Vision*. 2020, pp. 179–196. DOI: 10.1007/978-3-030-58548-8_11.
- [ELC19] Marvin Eisenberger, Zorah Löhner, and Daniel Cremers. “Divergence-Free Shape Correspondence by Deformation”. In: *Computer Graphics Forum* 38.5 (2019), pp. 1–12. ISSN: 1467-8659. DOI: 10.1111/cgf.13785.
- [ER15] Charles M. Elliott and Thomas Ranner. “Evolving surface finite element method for the Cahn–Hilliard equation”. In: *Numerische Mathematik* 129.3 (2015), pp. 483–534. ISSN: 0945-3245. DOI: 10.1007/s00211-014-0644-y.
- [EH04] Jeff Erickson and Sarel Har-Peled. “Optimally Cutting a Surface into a Disk”. In: *Discrete and Computational Geometry* 31.1 (2004), pp. 37–59. ISSN: 0179-5376. DOI: 10.1007/s00454-003-2948-z.
- [FLJK21] Yu Fang, Minchen Li, Chenfanfu Jiang, and Danny M. Kaufman. “Guaranteed globally injective 3D deformation processing”. In: *ACM Transactions on Graphics* 40 (4 2021), pp. 1–13. ISSN: 0730-0301. DOI: 10.1145/3450626.3459757.
- [Fed59] Herbert Federer. “Curvature measures”. In: *Transactions of the American Mathematical Society* 93 (3 1959), pp. 418–491. ISSN: 0002-9947. DOI: 10.1090/S0002-9947-1959-0110078-1.
- [FL19] Matthias Fey and Jan E. Lenssen. “Fast Graph Representation Learning with PyTorch Geometric”. In: *ICLR Workshop on Representation Learning on Graphs and Manifolds*. 2019.

- [FLPJ04] P. Thomas Fletcher, Conglin Lu, Stephen M. Pizer, and Sarang Joshi. “Principal Geodesic Analysis for the Study of Nonlinear Statistics of Shape”. In: *IEEE Transactions on Medical Imaging* 23.8 (2004), pp. 995–1005. ISSN: 0278-0062. DOI: 10.1109/TMI.2004.831793.
- [FS11] Hans Föllmer and Alexander Schied. *Stochastic Finance - an introduction in discrete time*. 3. rev. an. Berlin: De Gruyter, 2011. ISBN: 3110218054.
- [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. Springer-Verlag, 2012, pp. 1–14.
- [FJMM03] 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”. In: *Comptes Rendus Mathématique* 336.8 (2003), pp. 697–702. ISSN: 1631073X. DOI: 10.1016/s1631-073x(03)00028-1.
- [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”. In: *Communications on Pure and Applied Mathematics* 55.11 (2002), pp. 1461–1506. DOI: 10.1002/cpa.10048.
- [FB11] Stefan Fröhlich and Mario Botsch. “Example-Driven Deformations Based on Discrete Shells”. In: *Computer Graphics Forum* 30.8 (2011), pp. 2246–2257. DOI: 10.1111/j.1467-8659.2011.01974.x.
- [FJSY09] Matthias Fuchs, Bert Jüttler, Otmar Scherzer, and Huaiping Yang. “Shape Metrics Based on Elastic Deformations”. In: *Journal of Mathematical Imaging and Vision* 35.1 (2009), pp. 86–102. ISSN: 1573-7683. DOI: 10.1007/s10851-009-0156-z.
- [FMD+19] Lawson Fulton, Vismay Modi, David Duvenaud, David I. W. Levin, and Alec Jacobson. “Latent-space Dynamics for Reduced Deformable Simulation”. In: *Computer Graphics Forum* 38.2 (2019), pp. 379–391. ISSN: 1467-8659. DOI: 10.1111/cgf.13645.
- [GLL+16] Lin Gao, Yu-Kun Lai, Dun Liang, Shu-Yu Chen, and Shihong Xia. “Efficient and Flexible Deformation Representation for Data-Driven Surface Modeling”. In: *ACM Transactions on Graphics* 35.5 (2016), 158:1–158:17. ISSN: 1557-7368. DOI: 10.1145/2908736.
- [GLY+21] Lin Gao, Yu-Kun Lai, Jie Yang, Zhang Ling-Xiao, Shihong Xia, Leif Kobbelt, Ling Xiao Zhang, Shihong Xia, and Leif Kobbelt. “Sparse Data Driven Mesh Deformation”. In: *IEEE Transactions on Visualization and Computer Graphics* 27.3 (2021), pp. 2085–2100. ISSN: 1941-0506. DOI: 10.1109/TVCG.2019.2941200.
- [GH97] Michael Garland and Paul S. Heckbert. “Surface simplification using quadric error metrics”. In: *ACM SIGGRAPH 1997 Papers*. SIGGRAPH '97. New York, New York, USA: Association for Computing Machinery, 1997, pp. 209–216. ISBN: 0897918967. DOI: 10.1145/258734.258849.
- [GWH01] Michael Garland, Andrew Willmott, and Paul S. Heckbert. “Hierarchical face clustering on polygonal surfaces”. In: *Proceedings of the 2001 Symposium on Interactive 3D Graphics*. I3D '01. New York, New York, USA: Association for Computing Machinery, 2001, pp. 49–58. ISBN: 1581132921. DOI: 10.1145/364338.364345.
- [GO21] Peter Gladbach and Heiner Olbermann. “Approximation of the Willmore Energy by a Discrete Geometry Model”. In: *Advances in Calculus of Variations* (2021). ISSN: 1864-8266. DOI: 10.1515/acv-2020-0094.

- [Gor73] William B. Gordon. “An analytical criterion for the completeness of Riemannian manifolds”. In: *Proceedings of the American Mathematical Society* 37 (1 1973), pp. 221–225. ISSN: 0002-9939. DOI: 10.1090/S0002-9939-1973-0307112-5.
- [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. Aire-la-Ville, Switzerland, Switzerland: Eurographics Association, 2003, pp. 62–67. ISBN: 1581136595. DOI: 10.5555/846276.846284.
- [GJ+10] Gaël Guennebaud, Benoît Jacob, *et al.* *Eigen v3*. <http://eigen.tuxfamily.org>. 2010.
- [Han07] Andrew J. Hanson. “Visualizing quaternions”. In: *ACM SIGGRAPH 2007 Courses*. SIGGRAPH '07. San Diego, California: Association for Computing Machinery, 2007. ISBN: 9780080474779. DOI: 10.1145/1281500.1281634.
- [HN59] Philip Hartman and Louis Nirenberg. “On Spherical Image Maps Whose Jacobians Do Not Change Sign”. In: *American Journal of Mathematics* 81.4 (1959), p. 901. ISSN: 0002-9327. DOI: 10.2307/2372995.
- [HSA+23] Florine Hartwig, Josua Sassen, Omri Azencot, Martin Rumpf, and Mirela Ben-Chen. “An Elastic Basis for Spectral Shape Correspondence”. under review. 2023.
- [HSS+09] Nils Hasler, Carsten Stoll, Martin Sunkel, Bodo Rosenhahn, and Hans-Peter Seidel. “A Statistical Model of Human Pose and Body Shape”. In: *Computer Graphics Forum* 28.2 (2009), pp. 337–346. ISSN: 0167-7055. DOI: 10.1111/j.1467-8659.2009.01373.x.
- [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*. Bonn, 2017.
- [HRS+14] Behrend Heeren, Martin Rumpf, Peter Schröder, Max Wardetzky, and Benedikt Wirth. “Exploring the Geometry of the Space of Shells”. In: *Computer Graphics Forum* 33.5 (2014), pp. 247–256. DOI: 10.1111/cgf.12450.
- [HRS+16] Behrend Heeren, Martin Rumpf, Peter Schröder, Max Wardetzky, and Benedikt Wirth. “Splines in the Space of Shells”. In: *Computer Graphics Forum* 35.5 (2016), pp. 111–120. DOI: 10.1111/cgf.12968.
- [HRWW12] Behrend Heeren, Martin Rumpf, Max Wardetzky, and Benedikt Wirth. “Time-Discrete Geodesics in the Space of Shells”. In: *Computer Graphics Forum* 31.5 (2012), pp. 1755–1764. DOI: 10.1111/j.1467-8659.2012.03180.x.
- [HS+20] Behrend Heeren, Josua Sassen, *et al.* *The Geometric Optimization And Simulation Toolbox*. 2020. URL: <https://gitlab.com/numod/goast>.
- [HZRS18] Behrend Heeren, Chao Zhang, Martin Rumpf, and William Smith. “Principal Geodesic Analysis in the Space of Discrete Shells”. In: *Computer Graphics Forum* 37.5 (2018), pp. 173–184. DOI: 10.1111/cgf.13500.
- [HLDS00] Jose Herskovits, Anatoli Leontiev, Guanacy Dias, and Gabel Santos. “Contact shape optimization: A bilevel programming approach”. In: *Structural and Multidisciplinary Optimization* 20 (2000), pp. 214–221. DOI: 10.1007/s001580050149.
- [HL22] Roland Herzog and Estefanía Loayza-Romero. “A manifold of planar triangular meshes with complete Riemannian metric”. In: *Mathematics of Computation* 92 (339 2022), pp. 1–50. ISSN: 0025-5718. DOI: 10.1090/mcom/3775.
- [Hes75] Magnus R. Hestenes. *Optimization theory: the finite dimensional case*. Pure and applied mathematics. Wiley, 1975. ISBN: 9780471374718.

- [HPUU09] Michael Hinze, Rene Pinnau, Michael Ulbrich, and Stefan Ulbrich. *Optimization with PDE Constraints*. Vol. 23. Springer Netherlands, 2009. ISBN: 978-1-4020-8838-4. DOI: 10.1007/978-1-4020-8839-1.
- [Hor11] Peter Hornung. “Approximation of Flat $W^{2,2}$ Isometric Immersions by Smooth Ones”. In: *Archive for Rational Mechanics and Analysis* 199.3 (2011), pp. 1015–1067. DOI: 10.1007/s00205-010-0374-y.
- [HRS20] Peter Hornung, Martin Rumpf, and Stefan Simon. “On Material Optimisation for Non-linearly Elastic Plates and Shells”. In: *ESAIM: Control, Optimisation and Calculus of Variations* 26 (2020). DOI: 10.1051/cocv/2020053.
- [HYZ+14] Zhichao Huang, Junfeng Yao, Zichun Zhong, Yang Liu, and Xiaohu Guo. “Sparse Localized Decomposition of Deformation Gradients”. In: *Computer Graphics Forum* 33.7 (2014), pp. 239–248. ISSN: 1467-8659. DOI: 10.1111/cgf.12492.
- [Iva18] Sergey V. Ivanov. “A Bilevel Stochastic Programming Problem with Random Parameters in the Follower’s Objective Function”. In: *Journal of Applied and Industrial Mathematics* 12.4 (2018), pp. 658–667. ISSN: 1990-4789. DOI: 10.1134/S1990478918040063.
- [JKS05] Dan Julius, Vladislav Kraevoy, and Alla Sheffer. “D-Charts: Quasi-Developable Mesh Segmentation”. In: *Computer Graphics Forum*. Vol. 24. 3. 2005, pp. 581–590. DOI: 10.1111/j.1467-8659.2005.00883.x.
- [Kar14] Hermann Karcher. “Riemannian Center of Mass and so called Karcher mean”. In: (2014). DOI: 10.48550/ARXIV.1407.2087. arXiv: 1407.2087.
- [Kar77] Herrmann Karcher. “Riemannian center of mass and mollifier smoothing”. In: *Communications on Pure and Applied Mathematics* 30.5 (1977), pp. 509–541. DOI: 10.1002/cpa.3160300502.
- [KT03] Sagi Katz and Ayellet Tal. “Hierarchical mesh decomposition using fuzzy clustering and cuts”. In: *ACM Transactions on Graphics* 22.3 (2003), pp. 954–961. ISSN: 0730-0301. DOI: 10.1145/882262.882369.
- [Ken84] David G. Kendall. “Shape Manifolds, Procrustean Metrics, and Complex Projective Spaces”. In: *Bulletin of the London Mathematical Society* 16.2 (1984), pp. 81–121. ISSN: 0024-6093. DOI: 10.1112/blms/16.2.81.
- [KMP07] Martin Kilian, Niloy J. Mitra, and Helmut Pottmann. “Geometric modeling in shape space”. In: *ACM Transactions on Graphics* 26.3 (2007), p. 64. ISSN: 0730-0301. DOI: 10.1145/1276377.1276457.
- [KB15] Diederik P. Kingma and Jimmy Ba. “Adam: A Method for Stochastic Optimization”. In: *International Conference on Learning Representations*. 2015.
- [KW17] Thomas N. Kipf and Max Welling. “Semi-Supervised Classification with Graph Convolutional Networks”. In: *International Conference on Learning Representations*. 2017.
- [KG08] Scott Kircher and Michael Garland. “Free-form motion processing”. In: *ACM Transactions on Graphics* 27.2 (2008), pp. 1–13. ISSN: 0730-0301. DOI: 10.1145/1356682.1356685.
- [Koč97] Michal Kočvara. “Topology optimization with displacement constraints: a bilevel programming approach”. In: *Structural Optimization* 14 (4 1997), pp. 256–263. ISSN: 0934-4373. DOI: 10.1007/BF01197948.
- [KO95] Michal Kočvara and Jiří V. Outrata. “On the Solution of Optimum Design Problems with Variational Inequalities”. In: *Recent Advances in Nonsmooth Optimization* (1995), pp. 172–192. DOI: 10.1142/9789812812827_0011.

- [KSM18] Sławomir Kołasiński, Paweł Strzelecki, and Heiko von der Mosel. “Compactness and isotopy finiteness for submanifolds with uniformly bounded geometric curvature energies”. In: *Communications in Analysis and Geometry* 26 (6 2018), pp. 1251–1316. ISSN: 1019-8385. DOI: 10.4310/CAG.2018.v26.n6.a2.
- [KP14] Raimund M. Kovacevic and Georg Ch. Pflug. “Electricity swing option pricing by stochastic bilevel optimization: A survey and new approaches”. In: *European Journal of Operational Research* 237 (2 2014), pp. 389–403. ISSN: 0377-2217. DOI: 10.1016/J.EJOR.2013.12.029.
- [KNP14] Niklas Krauth, Matthias Nieser, and Konrad Polthier. “Differential-based geometry and texture editing with brushes”. In: *Journal of Mathematical Imaging and Vision* 48.2 (2014), pp. 359–368. ISSN: 0924-9907. DOI: 10.1007/s10851-013-0443-6.
- [Küh15] Wolfgang Kühnel. *Differential Geometry: Curves – Surfaces – Manifolds*. American Mathematical Society, 2015. ISBN: 978-1-4704-2675-0.
- [Kui55] Nicolaas H. Kuiper. “On C^1 -isometric imbeddings. I, II”. In: *Nederl. Akad. Wetensch. Proc. Ser. A* **58** = *Indag. Math.* 17 (1955), pp. 545–556, 683–689. DOI: 10.1016/S1385-7258(55)50075-8.
- [KLS+21] Sandrine H. Künzel, Moritz Lindner, Josua Sassen, Philipp T. Möller, Lukas Goerd, Matthias Schmid, Steffen Schmitz-Valckenberg, Frank G. Holz, Monika Fleckenstein, and Maximilian Pfau. “Association of Reading Performance in Geographic Atrophy Secondary to Age-Related Macular Degeneration With Visual Function and Structural Biomarkers”. In: *JAMA Ophthalmology* 139.11 (2021), pp. 1191–1199. ISSN: 2168-6165. DOI: 10.1001/jamaophthamol.2021.3826.
- [KKD+11] Sebastian Kurtek, Eric Klassen, Zhaohua Ding, Sandra Jacobson, Joseph B. Jacobson, Malcolm Avison, and Anuj Srivastava. “Parameterization-Invariant Shape Comparisons of Anatomical Surfaces”. In: *IEEE Transactions on Medical Imaging* 30.3 (2011), pp. 849–858. DOI: 10.1109/TMI.2010.2099130.
- [KKG+11] Sebastian Kurtek, Eric Klassen, John C. Gore, Zhaohua Ding, and Anuj Srivastava. “Classification of Mathematics Deficiency Using Shape and Scale Analysis of 3D Brain Structures”. In: *Medical Imaging 2011: Image Processing*. Ed. by Benoit M. Dawant and David R. Haynor. Vol. 7962. SPIE Proceedings. SPIE, 2011.
- [KXSC16] Sebastian Kurtek, Qian Xie, Chafik Samir, and Michel Canis. “Statistical Model for Simulation of Deformable Elastic Endometrial Tissue Shapes”. In: *Neurocomputing* 173 (2016), pp. 36–41. ISSN: 0925-2312. DOI: 10.1016/j.neucom.2015.03.098.
- [LKSM14] Hamid Laga, Sebastian Kurtek, Anuj Srivastava, and Stanley Miklavcic. “Landmark-Free Statistical Analysis of the Shape of Plant Leaves”. In: *Journal of Theoretical Biology* 363 (2014), pp. 41–52. DOI: 10.1016/j.jtbi.2014.07.036.
- [LZHM06] Yu-Kun Lai, Qian-Yi Zhou, Shi-Min Hu, and Ralph R. Martin. “Feature sensitive mesh segmentation”. In: *Proceedings of the 2006 ACM symposium on Solid and physical modeling - SPM '06*. SPM '06. New York, New York, USA: Association for Computing Machinery, 2006, p. 17. ISBN: 1595933581. DOI: 10.1145/1128888.1128891.
- [Lan95] Serge Lang. *Differential and Riemannian Manifolds*. Vol. 160. Springer-Verlag, New York, 1995.
- [LR95] Hervé Le Dret and Annie Raoult. “The Nonlinear Membrane Model as a Variational Limit of Nonlinear Three-Dimensional Elasticity”. In: *Journal de Mathématiques Pures et Appliquées* 73.6 (1995), pp. 549–578.

- [LR96] Hervé Le Dret and Annie Raoult. “The Membrane Shell Model in Nonlinear Elasticity: A Variational Asymptotic Derivation”. In: *Journal of Nonlinear Science* 6.1 (1996), pp. 59–84. ISSN: 0938-8974. DOI: 10.1007/BF02433810.
- [LS21] Gunther Leobacher and Alexander Steinicke. “Existence, uniqueness and regularity of the projection onto differentiable manifolds”. en. In: *Annals of Global Analysis and Geometry* 60.3 (2021), pp. 559–587. ISSN: 1572-9060. DOI: 10.1007/s10455-021-09788-z.
- [LPRM02] Bruno Lévy, Sylvain Petitjean, Nicolas Ray, and Jérôme Maillot. “Least squares conformal maps for automatic texture atlas generation”. In: *ACM Transactions on Graphics* 21.3 (2002), pp. 362–371. ISSN: 0730-0301. DOI: 10.1145/566654.566590.
- [LKK+19] Minchen Li, Danny M. Kaufman, Vladimir G. Kim, Justin Solomon, and Alla Sheffer. “OptCuts: Joint Optimization of Surface Cuts and Parameterization”. In: *ACM Transactions on Graphics* 37.6 (2019), pp. 1–13. ISSN: 0730-0301. DOI: 10.1145/3272127.3275042.
- [LA07] Jyh-Ming Lien and Nancy M. Amato. “Approximate convex decomposition of polyhedra”. In: *Proceedings of the 2007 ACM symposium on Solid and physical modeling - SPM '07*. SPM '07. New York, New York, USA: Association for Computing Machinery, 2007, p. 121. ISBN: 9781595936660. DOI: 10.1145/1236246.1236265.
- [LM17] M. Beatrice Lignola and Jacqueline Morgan. “Inner Regularizations and Viscosity Solutions for Pessimistic Bilevel Optimization Problems”. In: *Journal of Optimization Theory and Applications* 173.1 (2017), pp. 183–202. DOI: 10.1007/s10957-017-1085-4.
- [LM19] M. Beatrice Lignola and Jacqueline Morgan. “Further on Inner Regularizations in Bilevel Optimization”. In: *Journal of Optimization Theory and Applications* 180.3 (2019), pp. 1087–1097. DOI: 10.1007/s10957-018-1438-7.
- [LSC+04] Yaron Lipman, Olga Sorkine, Daniel Cohen-Or, David Levin, Christian Rössl, and Hans-Peter Seidel. “Differential Coordinates for Interactive Mesh Editing”. In: *Proceedings Shape Modeling Applications, 2004*. Genova, Italy: IEEE, 2004, pp. 181–190. ISBN: 978-0-7695-2075-9. DOI: 10.1109/SMI.2004.1314505.
- [LSLC05] Yaron Lipman, Olga Sorkine, David Levin, and Daniel Cohen-Or. “Linear rotation-invariant coordinates for meshes”. In: *ACM Transactions on Graphics* 24.3 (2005), p. 479. ISSN: 0730-0301. DOI: 10.1145/1073204.1073217.
- [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*. Aire-la-Ville, Switzerland, Switzerland: Eurographics Association, 2005. ISBN: 3-905673-24-X.
- [LSDM10] Xiuwen Liu, Yonggang Shi, Ivo Dinov, and Washington Mio. “A Computational Model of Multidimensional Shape”. In: *International Journal of Computer Vision* 89.1 (2010), pp. 69–83. ISSN: 1573-1405. DOI: 10.1007/s11263-010-0323-0.
- [LLW+19] Yuxin Liu, Guiqing Li, Yupan Wang, Yongwei Nie, and Aihua Mao. “Discrete shell deformation driven by adaptive sparse localized components”. In: *Computers & Graphics* 78 (2019), pp. 76–86. ISSN: 0097-8493. DOI: 10.1016/j.cag.2018.11.005.
- [LM96] Pierre Loridan and Jacqueline Morgan. “Weak via strong Stackelberg problem: New results”. In: *Journal of Global Optimization* 8.3 (1996), pp. 263–287. DOI: 10.1007/BF00121269.
- [Lov92] Augustus Edward Hough Love. *A treatise on the mathematical theory of elasticity*. hal-01307751, 1892.

- [LMP87] Roberto Lucchetti, Fausto Mignanego, and Graziano Pieri. “Existence Theorems of Equilibrium Points in Stackelberg Games with Constraints”. In: *Optimization* 18.6 (1987), pp. 857–866. DOI: 10.1080/02331938708843300.
- [MRSS15] Jan Maas, Martin Rumpf, Carola Schönlieb, and Stefan Simon. “A generalized model for optimal transport of images including dissipation and density modulation”. In: *ESAIM: Mathematical Modelling and Numerical Analysis* 49.6 (2015), pp. 1745–1769. ISSN: 1290-3841. DOI: 10.1051/m2an/2015043.
- [MG09] Khaled Mamou and Faouzi Ghorbel. “A simple and efficient approach for 3D mesh approximate convex decomposition”. In: *2009 16th IEEE International Conference on Image Processing (ICIP)*. IEEE, 2009, pp. 3501–3504. ISBN: 978-1-4244-5653-6. DOI: 10.1109/ICIP.2009.5414068.
- [MHKP18] Jesús Martínez-Frutos, David Herrero-Pérez, Mathieu Kessler, and Francisco Periago. “Risk-averse structural topology optimization under random fields using stochastic expansion methods”. In: *Computer Methods in Applied Mechanics and Engineering* 330 (2018), pp. 180–206. DOI: 10.1016/j.cma.2017.10.026.
- [MTGG11] Sebastian Martin, Bernhard Thomaszewski, Eitan Grinspun, and Markus Gross. “Example-Based Elastic Materials”. In: *ACM Transactions on Graphics* 30.4 (2011), 72:1–72:8. DOI: 10.1145/2010324.1964967.
- [MRS21] Janos Meny, Martin Rumpf, and Josua Sassen. “A Phase-field Approach to Variational Hierarchical Surface Segmentation”. In: *Computer Aided Geometric Design* 89 (2021), p. 102025. ISSN: 0167-8396. DOI: 10.1016/j.cagd.2021.102025.
- [MM07] Peter W. Michor and David Mumford. “An overview of the Riemannian metrics on spaces of curves using the Hamiltonian approach”. In: *Applied and Computational Harmonic Analysis* 23.1 (2007), pp. 74–113. ISSN: 1063-5203. DOI: 10.1016/j.acha.2006.07.004.
- [MTT22] Michael Miller, Daniel Tward, and Alain Trouvé. “Molecular Computational Anatomy: Unifying the Particle to Tissue Continuum via Measure Representations of the Brain”. In: *BME Frontiers* 2022 (2022). ISSN: 2765-8031. DOI: 10.34133/2022/9868673.
- [MY01] Michael Miller and Laurent Younes. “Group Actions, Homeomorphisms, and Matching: A General Framework”. In: *International Journal of Computer Vision* 41.1/2 (2001), pp. 61–84. ISSN: 0920-5691. DOI: 10.1023/A:1011161132514.
- [MTY06] Michael I. Miller, Alain Trouvé, and Laurent Younes. “Geodesic Shooting for Computational Anatomy”. In: *Journal of Mathematical Imaging and Vision* 24.2 (2006), pp. 209–228. DOI: 10.1007/s10851-005-3624-0.
- [MK12] Jan Möbius and Leif Kobbelt. “OpenFlipper: An Open Source Geometry Processing and Rendering Framework”. In: *Curves and Surfaces*. Vol. 6920. Lecture Notes in Computer Science. Springer, 2012, pp. 488–500.
- [MM77] Luciano Modica and Stefano Mortola. “Un esempio di Γ^- -convergenza”. In: *Boll. Un. Mat. Ital. B (5)* 14.1 (1977), pp. 285–299.
- [MG03] Alex Mohr and Michael Gleicher. *Deformation Sensitive Decimation*. Tech. rep. Madison: University of Wisconsin, 2003, pp. 1–5.
- [MOS20] MOSEK ApS. *The MOSEK Optimizer API for C manual. Version 9.2*. 2020. URL: <https://docs.mosek.com/9.2/capi/index.html>.
- [Nas54] John Nash. “ C^1 Isometric Imbeddings”. In: *The Annals of Mathematics* 60.3 (1954), p. 383. DOI: 10.2307/1969840.

- [NEOM13] Patrizio Neff, Bernhard Eidel, Frank Osterbrink, and Robert Martin. “The Hencky Strain Energy $\|\log U\|^2$ Measures the Geodesic Distance of the Deformation Gradient to $SO(n)$ in the Canonical Left-Invariant Riemannian Metric on $GL(n)$ ”. In: *PAMM* 13.1 (2013), pp. 369–370. ISSN: 1617-7061. DOI: 10.1002/pamm.201310180.
- [NVW+13] Thomas Neumann, Kiran Varanasi, Stephan Wenger, Markus Wacker, Marcus Magnor, and Christian Theobalt. “Sparse localized deformation components”. In: *ACM Transactions on Graphics* 32.6 (2013), pp. 1–10. ISSN: 0730-0301. DOI: 10.1145/2508363.2508417.
- [NPV12] Eleonora Di Nezza, Giampiero Palatucci, and Enrico Valdinoci. “Hitchhiker’s guide to the fractional Sobolev spaces”. In: *Bulletin des Sciences Mathématiques* 136 (5 2012), pp. 521–573. ISSN: 0007-4497. DOI: 10.1016/j.bulsci.2011.12.004.
- [Noc80] Jorge Nocedal. “Updating Quasi-Newton Matrices with Limited Storage”. In: *Mathematics of Computation* 35 (1980), p. 773. ISSN: 0025-5718. DOI: 10.2307/2006193.
- [NW06] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. New York, NY, USA: Springer, 2006.
- [Pal03] Richard Palais. *Lecture Notes on the Differential Geometry of Curves and Surfaces*. 2003.
- [PGM+19] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. “PyTorch: An Imperative Style, High-Performance Deep Learning Library”. In: *Advances in Neural Information Processing Systems* 32. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, and R. Garnett. 2019, pp. 8024–8035.
- [Pat08] Michael Patriksson. “On the applicability and solution of bilevel optimization models in transportation science: A study on the existence, stability and computation of optimal solutions to stochastic mathematical programs with equilibrium constraints”. In: *Transportation Research Part B: Methodological* 42 (10 2008), pp. 843–860. ISSN: 0191-2615. DOI: 10.1016/J.TRB.2008.05.001.
- [PW99] Michael Patriksson and Laura Wynter. “Stochastic Mathematical Programs with Equilibrium Constraints”. In: *Operations Research Letters* 25.4 (1999), pp. 159–167. DOI: 10.1016/S0167-6377(99)00052-8.
- [PCG+19] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. “Expressive Body Capture: 3D Hands, Face, and Body from a Single Image”. In: *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 10975–10985. DOI: 10.1109/CVPR.2019.01123.
- [Pen06] Xavier Pennec. “Intrinsic Statistics on Riemannian Manifolds: Basic Tools for Geometric Measurements”. In: *Journal of Mathematical Imaging and Vision* 25.1 (2006), pp. 127–154. ISSN: 1573-7683. DOI: 10.1007/s10851-006-6228-4.
- [PPKC10] Gabriel Peyré, Mickael Péchaud, Renaud Keriven, and Laurent D. Cohen. “Geodesic Methods in Computer Vision and Graphics”. English. In: *Foundations and Trends in Computer Graphics and Vision* 5.3–4 (2010). Publisher: Now Publishers, Inc., pp. 197–397. ISSN: 1572-2759. DOI: 10.1561/06000000029.
- [PHBR20] David Pfau, Irina Higgins, Aleksandar Botev, and Sébastien Racanière. “Disentangling by Subspace Diffusion”. In: *Advances in Neural Information Processing Systems* (2020), pp. 17403–17415.

- [PB00] Dan Piponi and George Borshukov. “Seamless texture mapping of subdivision surfaces by model pelting and texture blending”. In: *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH '00. New York, NY, USA: Association for Computing Machinery, 2000, pp. 471–478. ISBN: 1581132085. DOI: 10.1145/344779.344990.
- [PTH+17] Roi Poranne, Marco Tarini, Sandro Huber, Daniele Panozzo, and Olga Sorkine-Hornung. “Autocuts: simultaneous distortion and cut optimization for UV mapping”. In: *ACM Transactions on Graphics* 36.6 (2017), pp. 1–11. ISSN: 0730-0301. DOI: 10.1145/3130800.3130845.
- [PKC+16] Fabián Prada, Misha Kazhdan, Ming Chuang, Alvaro Collet, and Hugues Hoppe. “Motion Graphs for Unstructured Textured Meshes”. In: *ACM Transactions on Graphics* 35.4 (2016), 108:1–108:14. DOI: 10.1145/2897824.2925967.
- [RESH16] Philipp von Radziewsky, Elmar Eisemann, Hans-Peter Seidel, and Klaus Hildebrandt. “Optimized subspaces for deformation-based modeling and shape interpolation”. In: *Computers & Graphics* 58 (2016), pp. 128–138. ISSN: 0097-8493. DOI: 10.1016/j.cag.2016.05.016.
- [RS21] Philipp Reiter and Henrik Schumacher. “Sobolev Gradients for the Möbius Energy”. In: *Archive for Rational Mechanics and Analysis* 242 (2 2021), pp. 701–746. ISSN: 1432-0673. DOI: 10.1007/S00205-021-01680-1.
- [RE55] Ronald Samuel Rivlin and Jerald Laverne Ericksen. “Stress-Deformation Relations for Isotropic Materials”. In: *Journal of Rational Mechanics and Analysis* 4 (1955), pp. 323–425.
- [RM51] Herbert Robbins and Sutton Monro. “A Stochastic Approximation Method”. In: *The Annals of Mathematical Statistics* 22.3 (1951), pp. 400–407. DOI: 10.1214/aoms/1177729586.
- [RSA07] Emad Roghanian, Seyed Jafar Sadjadi, and Mir Bahador Aryanezhad. “A probabilistic bi-level linear multi-objective programming problem to supply chain planning”. In: *Applied Mathematics and Computation* 188 (1 2007), pp. 786–800. ISSN: 0096-3003. DOI: 10.1016/J.AMC.2006.10.032.
- [Row79] John Shipley Rowlinson. “Translation of J. D. van der Waals’ “The thermodynamik theory of capillarity under the hypothesis of a continuous variation of density””. In: *Journal of Statistical Physics* 1979 20:2 20 (2 1979), pp. 197–200. ISSN: 1572-9613. DOI: 10.1007/BF01011513.
- [RW09] Martin Rumpf and Benedikt Wirth. “A nonlinear elastic shape averaging approach”. In: *SIAM Journal on Imaging Sciences* 2.3 (2009), pp. 800–833. ISSN: 1936-4954. DOI: 10.1137/080738337.
- [RW13] Martin Rumpf and Benedikt Wirth. “Discrete Geodesic Calculus in Shape Space and Applications in the Space of Viscous Fluidic Objects”. In: *SIAM Journal on Imaging Sciences* 6.4 (2013). Publisher: Society for Industrial and Applied Mathematics, pp. 2581–2602. DOI: 10.1137/120870864.
- [RW15] Martin Rumpf and Benedikt Wirth. “Variational time discretization of geodesic calculus”. In: *IMA Journal of Numerical Analysis* 35.3 (2015), pp. 1011–1046. ISSN: 1464-3642. DOI: 10.1093/imanum/dru027.
- [SKSC14] Chafik Samir, Sebastian Kurtek, Anuj Srivastava, and Michel Canis. “Elastic Shape Analysis of Cylindrical Surfaces for 3D/2D Registration in Endometrial Tissue Characterization”. In: *IEEE Transactions on Medical Imaging* 33.5 (2014), pp. 1035–1043. DOI: 10.1109/TMI.2014.2300935.

- [Sas19] Josua Sassen. “Discrete Gauß–Codazzi Equations for Efficient Triangle Mesh Processing”. Master’s Thesis. University of Bonn, 2019.
- [Sas22] Josua Sassen. “Repulsive Shells”. In: *Oberwolfach Reports 2234* (2022), pp. 37–40. DOI: 10.14760/OWR-2022-38.
- [SHHR19] Josua Sassen, Behrend Heeren, Klaus Hildebrandt, and Martin Rumpf. “Solving Variational Problems Using Nonlinear Rotation-invariant Coordinates”. In: *Symposium on Geometry Processing 2019 – Posters*. The Eurographics Association, 2019. DOI: 10.2312/sgp.20191213.
- [SHHR20] Josua Sassen, Behrend Heeren, Klaus Hildebrandt, and Martin Rumpf. “Geometric optimization using nonlinear rotation-invariant coordinates”. In: *Computer Aided Geometric Design 77* (2020), p. 101829. ISSN: 0167-8396. DOI: 10.1016/j.cagd.2020.101829.
- [SHR20] Josua Sassen, Klaus Hildebrandt, and Martin Rumpf. “Nonlinear Deformation Synthesis via Sparse Principal Geodesic Analysis”. In: *Computer Graphics Forum (Proc. SGP) 39.5* (2020), pp. 119–132. DOI: 10.1111/cgf.14073.
- [SHWR23] Josua Sassen, Klaus Hildebrandt, Benedikt Wirth, and Martin Rumpf. “Parametrizing Product Shape Manifolds by Composite Networks”. accepted at International Conference on Learning Representations (ICLR). 2023. URL: https://openreview.net/forum?id=F_EhNDSamN.
- [SY94] Richard M. Schoen and Shing-Tung Yau. *Lectures on differential geometry*. Cambridge, MA : International Press, 1994. ISBN: 978-1-57146-012-7.
- [SC18] Nicholas Sharp and Keenan Crane. “Variational surface cutting”. In: *ACM Transactions on Graphics 37.4* (2018), pp. 1–13. ISSN: 0730-0301. DOI: 10.1145/3197517.3201356.
- [SSC19] Nicholas Sharp, Yousuf Soliman, and Keenan Crane. “Navigating Intrinsic Triangulations”. In: *ACM Transactions on Graphics 38.4* (2019). DOI: 10.1145/3306346.3322979.
- [SM50] Jack Sherman and Winifred J. Morrison. “Adjustment of an Inverse Matrix Corresponding to a Change in One Element of a Given Matrix”. In: *The Annals of Mathematical Statistics 21.1* (1950), pp. 124–127. ISSN: 2168-8990. DOI: 10.1214/aoms/1177729893.
- [SM00] Jianbo Shi and Jitendra Malik. “Normalized Cuts and Image Segmentation”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence 22.8* (2000), pp. 888–905. DOI: 10.1109/34.868688.
- [SMD18] Ankur Sinha, Pekka Malo, and Kalyanmoy Deb. “A Review on Bilevel Optimization: From Classical to Evolutionary Approaches and Applications”. In: *IEEE Transactions on Evolutionary Computation 22.2* (2018), pp. 276–295. DOI: 10.1109/TEVC.2017.2712906.
- [SSC18] Yousuf Soliman, Dejan Slepčev, and Keenan Crane. “Optimal cone singularities for conformal flattening”. In: *ACM Transactions on Graphics 37.4* (2018), pp. 1–17. ISSN: 0730-0301. DOI: 10.1145/3197517.3201367.
- [SCL+04] Olga Sorkine, Daniel Cohen-Or, Yaron Lipman, Marc Alexa, Christian Rössl, and Hans-Peter Seidel. “Laplacian Surface Editing”. In: *Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing*. Nice, France: Association for Computing Machinery, 2004, pp. 175–184. ISBN: 978-3-905673-13-5. DOI: 10.1145/1057432.1057456.

- [SSP08] Boris Springborn, Peter Schröder, and Ulrich Pinkall. “Conformal equivalence of triangle meshes”. In: *ACM Transactions on Graphics* 27.3 (2008), pp. 1–11. ISSN: 0730-0301. DOI: 10.1145/1360612.1360676.
- [Sta34] Heinrich von Stackelberg. *Marktform und Gleichgewicht*. Wien und Berlin: Julius Springer, 1934.
- [SMS+20] Pascal Stadlbauer, Daniel Mlakar, Hans-Peter Seidel, Markus Steinberger, and Rhaleb Zayer. “Interactive Modeling of Cellular Structures on Surfaces with Application to Additive Manufacturing”. In: *Computer Graphics Forum* 39.2 (2020), pp. 277–289. ISSN: 0167-7055. DOI: 10.1111/cgf.13929.
- [Ste83] Trond Steihaug. “The Conjugate Gradient Method and Trust Regions in Large Scale Optimization”. In: *SIAM Journal on Numerical Analysis* 20.3 (1983), pp. 626–637. ISSN: 0036-1429. DOI: 10.1137/0720042.
- [SM13] Paweł Strzelecki and Heiko von der Mosel. “Tangent-Point Repulsive Potentials for a Class of Non-smooth m -dimensional Sets in \mathbb{R}^n ”. Part I: Smoothing and Self-avoidance Effects”. In: *Journal of Geometric Analysis* 23.3 (2013), pp. 1085–1139. ISSN: 1050-6926. DOI: 10.1007/s12220-011-9275-z.
- [SM22] Paweł Strzelecki and Heiko von der Mosel. “Geometric curvature energies: facts, trends, and open problems”. In: *New Directions in Geometric and Applied Knot Theory*. Ed. by Philipp Reiter, Simon Blatt, and Armin Schikorra. Warsaw, Poland: De Gruyter Open Poland, 2022, pp. 8–35. ISBN: 9783110571493. DOI: 10.1515/9783110571493-002.
- [SP04] Robert W. Sumner and Jovan Popović. “Deformation Transfer for Triangle Meshes”. In: *ACM Transaction of Graphics* 23.3 (2004), pp. 399–405. ISSN: 0730-0301. DOI: 10.1145/1015706.1015736.
- [TZY+21] Qingyang Tan, Ling-Xiao Zhang, Jie Yang, Yu-Kun Lai, and Lin Gao. “Mesh-based Variational Autoencoders for Localized Deformation Component Analysis”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2021). ISSN: 1939-3539. DOI: 10.1109/TPAMI.2021.3085887.
- [TW03] Johannes Tausch and Jacob White. “Multiscale bases for the sparse representation of boundary integral operators on complex geometry”. In: *SIAM Journal on Scientific Computing* 24.5 (2003), pp. 1610–1629. ISSN: 1064-8275. DOI: 10.1137/S1064827500369451.
- [Tib96] Robert Tibshirani. “Regression Shrinkage and Selection via the Lasso”. In: *Journal of the Royal Statistical Society* 58.1 (1996), pp. 267–288.
- [TY05a] Alain Trounev and Laurent Younes. “Local Geometry of Deformable Templates”. In: *SIAM Journal on Mathematical Analysis* 37.1 (2005), pp. 17–59. ISSN: 1095-7154. DOI: 10.1137/S0036141002404838.
- [TY05b] Alain Trounev and Laurent Younes. “Metamorphoses Through Lie Group Action”. In: *Foundations of Computational Mathematics* 5.2 (2005), pp. 173–198. ISSN: 1615-3383. DOI: 10.1007/s10208-004-0128-z.
- [TAMZ18] Christoph von Tycowicz, Felix Ambellan, Anirban Mukhopadhyay, and Stefan Zachow. “An Efficient Riemannian Statistical Shape Model Using Differential Coordinates”. In: *Medical Image Analysis* 43 (2018), pp. 1–9. ISSN: 13618423. DOI: 10.1016/j.media.2017.09.004.

- [TSSH13] Christoph von Tycowicz, Christian Schulz, Hans-Peter Seidel, and Klaus Hildebrandt. “An efficient construction of reduced deformable objects”. In: *ACM Transactions on Graphics* 32.6 (2013), 213:1–213:10. ISSN: 0730-0301. DOI: 10.1145/2508363.2508392.
- [TSSH15] Christoph von Tycowicz, Christian Schulz, Hans-Peter Seidel, and Klaus Hildebrandt. “Real-Time Nonlinear Shape Interpolation”. In: *ACM Transactions on Graphics* 34.3 (2015), 34:1–34:10. DOI: 10.1145/2729972.
- [VHWP12] Etienne Vouga, Mathias Höbinger, Johannes Wallner, and Helmut Pottmann. “Design of Self-Supporting Surfaces”. In: *ACM Transactions on Graphics* 31.4 (2012). ISSN: 0730-0301. DOI: 10.1145/2185520.2185583.
- [WB06] Andreas Wächter and Lorenz T. Biegler. “On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming”. In: *Mathematical Programming* 106.1 (2006), pp. 25–57. ISSN: 1436-4646. DOI: 10.1007/s10107-004-0559-y.
- [Wam16] Kevin Wampler. “Fast and Reliable Example-based Mesh IK for Stylized Deformations”. In: *ACM Transactions on Graphics* 35.6 (2016), 235:1–235:12. DOI: 10.1145/2980179.2982433.
- [WLX+18] Guan Wang, Hamid Laga, Ning Xie, Jinyuan Jia, and Hedi Tabia. “The Shape Space of 3D Botanical Tree Models”. In: *ACM Transactions on Graphics* 37.1 (2018), p. 7. DOI: 10.1145/3144456.
- [WLT12] Yuanzhen Wang, Beibei Liu, and Yiying Tong. “Linear surface reconstruction from discrete fundamental forms on triangle meshes”. In: *Computer Graphics Forum* 31.8 (2012), pp. 2277–2287. ISSN: 1467-8659. DOI: 10.1111/j.1467-8659.2012.03153.x.
- [WLZH17] Yupan Wang, Guiqing Li, Zhichao Zeng, and Huayun He. “Articulated-Motion-Aware Sparse Localized Decomposition”. In: *Computer Graphics Forum* 36.8 (2017), pp. 247–259. ISSN: 1467-8659. DOI: 10.1111/cgf.13076.
- [WLZ+21] Yupan Wang, Guiqing Li, Huiqian Zhang, Xinyi Zou, Yuxin Liu, and Yongwei Nie. “PanoMan: Sparse Localized Components-based Model for Full Human Motions”. In: *ACM Transactions on Graphics* 40.2 (2021), pp. 1–17. ISSN: 1557-7368. DOI: 10.1145/3447244.
- [WZ88] Alan Weiser and Sergio E. Zarantonello. “A Note on Piecewise Linear and Multilinear Table Interpolation in Many Dimensions”. In: *Mathematics of Computation* 50.181 (1988), pp. 189–196. DOI: 10.1090/S0025-5718-1988-0917826-0.
- [Wer05] Adrian S. Werner. “Bilevel Stochastic Programming Problems: Analysis and Application to Telecommunications”. PhD thesis. NTNU Fakultet for samfunnsvitenskap og teknologiledelse, 2005. ISBN: 82-471-6975-4.
- [WDAH10] Tim Winkler, Jens Drieseberg, Marc Alexa, and Kai Hormann. “Multi-Scale Geometry Interpolation”. In: *Computer Graphics Forum* 29.2 (2010), pp. 309–318. ISSN: 1467-8659. DOI: 10.1111/j.1467-8659.2009.01600.x.
- [Wir10] Benedikt Wirth. “Variational Methods in Shape Space”. PhD thesis. Rheinische Friedrich-Wilhelms-Universität Bonn, 2010.

- [WBRS09] Benedikt Wirth, Leah Bar, Martin Rumpf, and Guillermo Sapiro. “Geodesics in Shape Space via Variational Time Discretization”. In: *Energy Minimization Methods in Computer Vision and Pattern Recognition*. Ed. by Daniel Cremers, Yuri Boykov, Andrew Blake, and Frank R. Schmidt. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2009, pp. 288–302. ISBN: 978-3-642-03641-5. DOI: 10.1007/978-3-642-03641-5_22.
- [WBS+10] Stefanie Wuhrer, Prosenjit Bose, Chang Shu, Joseph O’Rourke, and Alan Brunton. “Morphing of Triangular Meshes in Shape Space”. In: *International Journal of Shape Modeling* 16.1-2 (2010), pp. 195–212. DOI: 10.1142/S0218654310001341.
- [XJKS14] Qian Xie, Ian H. Jermyn, Sebastian Kurtek, and Anuj Srivastava. “Numerical Inversion of SRNFs for Efficient Elastic Shape Analysis of Star-Shaped Objects”. In: *European Conference on Computer Vision*. Ed. by David J. Fleet, Tomás Pajdla, Bernt Schiele, and Tinne Tuytelaars. Vol. 8693. Lecture Notes in Computer Science. Springer, 2014, pp. 485–499. DOI: 10.1007/978-3-319-10602-1_32.
- [XZWB06] Dong Xu, Hongxin Zhang, Qing Wang, and Hujun Bao. “Poisson Shape Interpolation”. In: *Graphical Models* 68.3 (2006), pp. 268–281. DOI: 10.1016/j.gmod.2006.03.001.
- [YSS+21] Kenneth K. Yamamoto, Toby L. Shearman, Erik J. Struckmeyer, John A. Gemmer, and Shankar C. Venkataramani. “Nature’s forms are frilly, flexible, and functional”. In: *The European Physical Journal E* 44 (7 2021), p. 95. ISSN: 1292-8941. DOI: 10.1140/epje/s10189-021-00099-6.
- [YGZS05] Hitoshi Yamauchi, Stefan Gumhold, Rhaleb Zayer, and Hans-Peter Seidel. “Mesh segmentation driven by Gaussian curvature”. In: *The Visual Computer* 21.8-10 (2005), pp. 659–668. ISSN: 0178-2789. DOI: 10.1007/s00371-005-0319-x.
- [YGL+18] Jie Yang, Lin Gao, Yu-Kun Lai, Paul L. Rosin, and Shihong Xia. “Biharmonic Deformation Transfer with Automatic Key Point Selection”. In: *Graphical Models* 98 (2018), pp. 1–13. DOI: 10.1016/j.gmod.2018.05.003.
- [YLSL11] I-Cheng Yeh, Chao-Hung Lin, Olga Sorkine, and Tong-Yee Lee. “Template-Based 3D Model Fitting Using Dual-Domain Relaxation”. In: *IEEE Transactions on Visualization and Computer Graphics* 17.8 (2011), pp. 1178–1190. DOI: 10.1109/TVCG.2010.124.
- [You10] Laurent Younes. *Shapes and Diffeomorphisms*. Springer, 2010.
- [YBSC21] Chris Yu, Caleb Brakensiek, Henrik Schumacher, and Keenan Crane. “Repulsive Surfaces”. In: *ACM Transactions on Graphics* 40.6 (2021), pp. 1–19. ISSN: 0730-0301. DOI: 10.1145/3478513.3480521.
- [YSC21] Chris Yu, Henrik Schumacher, and Keenan Crane. “Repulsive Curves”. In: *ACM Transactions on Graphics* 40 (2 2021). ISSN: 1557-7368. DOI: 10.1145/3439429.
- [YZX+04] Yizhou Yu, Kun Zhou, Dong Xu, Xiaohan Shi, Hujun Bao, Baining Guo, and Heung-Yeung Shum. “Mesh Editing with Poisson-Based Gradient Field Manipulation”. In: *ACM Transactions on Graphics* 23 (2004), pp. 644–651. DOI: 10.1145/1015706.1015774.
- [ZMSS19] Rhaleb Zayer, Daniel Mlakar, Markus Steinberger, and Hans-Peter Seidel. “Layered fields for natural tessellations on surfaces”. In: *ACM Transactions on Graphics* 37.6 (2019), pp. 1–15. ISSN: 0730-0301. DOI: 10.1145/3272127.3275072.
- [ZSCS04] Li Zhang, Noah Snavely, Brian Curless, and Steven M. Seitz. “Spacetime Faces: High Resolution Capture for Modeling and Animation”. In: vol. 23. 3. New York, NY, USA: Association for Computing Machinery, 2004, pp. 548–558. DOI: 10.1145/1015706.1015759.

- [ZSGS04] Kun Zhou, John Synder, Baining Guo, and Heung-Yeung Shum. “Iso-charts”. In: *Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing*. SGP '04. New York, New York, USA: Association for Computing Machinery, 2004, pp. 45–54. ISBN: 3905673134. DOI: 10.1145/1057432.1057439.
- [ZYCF20] Tianyu Zhu, Chunyang Ye, Shuangming Chai, and Xiao-Ming Fu. “Greedy Cut Construction for Parameterizations”. In: *Computer Graphics Forum* 39.2 (2020), pp. 191–202. ISSN: 0167-7055. DOI: 10.1111/cgf.13923.
- [Zuo15] Wenjie Zuo. “Bi-level optimization for the cross-sectional shape of a thin-walled car body frame with static stiffness and dynamic frequency stiffness constraints”. In: *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* 229.8 (2015), pp. 1046–1059. DOI: 10.1177/0954407014551585.