

Geometric optimization using nonlinear rotation-invariant coordinates

Josua Sassen^{a,*}, Behrend Heeren^a, Klaus Hildebrandt^b, Martin Rumpf^a

^a*Institute for Numerical Simulation, Universität Bonn, Endenicher Allee 60, 53115 Bonn, Germany*

^b*Delft University of Technology, EEMCS – Dept. Intelligent Systems, Van Mourik Broekmanweg 6, 2628 XE Delft, Netherlands*

Abstract

Geometric optimization problems are at the core of many applications in geometry processing. The choice of a representation fitting an optimization problem can considerably simplify solving the problem. We consider the Nonlinear Rotation-Invariant Coordinates (NRIC) that represent the nodal positions of a discrete triangular surface with fixed combinatorics as a vector that stacks all edge lengths and dihedral angles of the mesh. It is known that this representation associates a unique vector to an equivalence class of nodal positions that differ by a rigid body motion. Moreover, integrability conditions that ensure the existence of nodal positions that match a given vector of edge lengths and dihedral angles have been established. The goal of this paper is to develop the machinery needed to use the NRIC for solving geometric optimization problems. First, we use the integrability conditions to derive an implicit description of the space of discrete surfaces as a submanifold of an Euclidean space and a corresponding description of its tangent spaces. Secondly, we reformulate the integrability conditions using quaternions and provide explicit formulas for their first and second derivatives facilitating the use of Hessians in NRIC-based optimization problems. Lastly, we introduce a fast and robust algorithm that reconstructs nodal positions from almost integrable NRIC. We demonstrate the benefits of this approach on a collection of geometric optimization problems. Comparisons to alternative approaches indicate that NRIC-based optimization is particularly effective for problems involving near-isometric deformations.

Keywords: geometric optimization, differential coordinates, surface deformation, isometric deformation

2010 MSC: 65D18, 65K10, 65D17, 74S30

1. Introduction

Geometric optimization problems are central to geometry processing as most methods involve some form of optimization as a step in their pipeline. Shape deformation problems are inherently nonlinear and therefore can be difficult to solve. In particular, problems involving near-isometric deformations are typically ill-conditioned due to the combination of high stretching and low bending resistance. Moreover, physical objectives are often invariant with respect to rigid body motions and the alignment of a mesh in Euclidean space is considered to be a post-processing task. However, this rigid body motion invariance can cause conceptual and numerical issues when working with nodal positions. Thus it is beneficial to find degrees of freedom for mesh description and corresponding deformations which are rigid body motion invariant.

In this work, we study the Nonlinear Rotation-Invariant Coordinates (NRIC) that describe the immersion of a mesh using the edge lengths and dihedral angles of the mesh instead of the nodal positions. 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. For example, when a human character (represented by a triangle mesh) lifts her straight arm, the induced variations in nodal positions comprise the entire arm. However, the same variation encoded in the change of lengths and angles is limited to the shoulder region, *i.e.* the place where the actual physical work is done.

*Corresponding author

Email addresses: josua.sassen@uni-bonn.de (Josua Sassen), heeren@ins.uni-bonn.de (Behrend Heeren), K.A.Hildebrandt@tudelft.nl (Klaus Hildebrandt), martin.rumpf@ins.uni-bonn.de (Martin Rumpf)

Prior work on shape interpolation by Winkler et al. [1] and Fröhlich and Botsch [2] 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.

Contribution. The basis of our approach is the triangle inequalities and the integrability conditions derived by Wang et al. [3]. Our goal is to provide the machinery required to formulate and solve geometric optimization problems entirely in NRIC.

- We reformulate the integrability conditions using quaternions and use this to provide an implicit description of the NRIC manifold along with its tangent spaces.
- We reformulate the nonlinear energy from [4] in NRIC and provide its derivatives to equip the NRIC manifold with a Riemannian metric.
- For solving (constrained) geometric optimization problems in the NRIC manifold, we describe an approach based on the augmented Lagrange method. In this context, we illustrate how to efficiently handle the triangle inequality constraints using the natural barrier term in the nonlinear energy and a modified line search. This also includes explicit formulas for the second derivatives of the integrability conditions, which are needed for evaluating the Hessian of objectives acting on NRIC.
- Finally, we introduce a hybrid algorithm to construct nodal positions of a discrete surface from NRIC which do not necessarily fulfill the integrability conditions. The algorithm uses an adaptive mesh traversal algorithm as initialization to a Gauß–Newton solver. In our experiments, this proves to effectively reduce the number of required Gauß–Newton iterations. Typically, a single iteration is sufficient or even no iteration is needed.

Experiments demonstrate the utility of our framework for various applications such as geodesics in shape space and paper folding. Our approach is particularly well-suited to deal with near isometric deformations of discrete shell surfaces, which is underpinned by a variety of numerical examples.

Organization. The remainder of this paper is organized as follows. After reviewing related work in Section 2, we summarize the necessary background on the established discrete integrability conditions as introduced by Wang et al. [3] in Section 3. We define our NRIC manifold and reformulate the integrability conditions using quaternions in Section 4. In Section 5, we discuss the nonlinear deformation energy. Afterwards, we introduce a corresponding variational calculus in Section 6. The robust reconstruction of nodal positions from lengths and angles is discussed in Section 7. Finally, we show a series of applications in Section 8 and discuss limitations and challenges in Section 9.

2. Related Work

In this section, we discuss relevant work on linear and nonlinear coordinates, rigidity of triangle meshes, shape interpolation, shape spaces and near-isometric deformation.

Linear coordinates. For solving problems in geometry processing, it can be useful to switch from the usual nodal coordinates to a different representation that is adapted to the given task. We distinguish between coordinates that depend linearly and nonlinearly on the nodal coordinates. *Differential coordinates* use discrete differential operators on a triangle mesh to define coordinates. Two examples are *gradient-domain* approaches for meshes [5, 6], which operate on the gradients of functions, and the *Laplace coordinates* [7, 8], which make use of the discrete Laplace–Beltrami operator. Since the differential coordinates depend linearly on the nodal positions, the immersion that best matches given differential coordinates can be found by solving a linear least-squares problem. While linearity of the coordinates facilitates computations, it also fundamentally limits their applicability. For example, shape editing approaches that use linear coordinates often yield unnatural and distorted shapes when larger deformations are involved [9].

Nonlinear coordinates. In classical differential geometry, the fundamental theorem of surfaces [10] states that two immersion of a surface to \mathbb{R}^3 differ by a rigid motion if and only if the first and second fundamental forms agree and provides integrability conditions that guarantee the existence of an immersion for a given first and second fundamental form. This motivates using discrete analogs to the fundamental forms as coordinates for triangle mesh processing. Explicitly, the list of all edges lengths and dihedral angles is used. Analogous to the classical theorem, the nodal positions of two meshes with the same combinatorics agree up to a global rigid motion if and only if all edge lengths and all dihedral angles agree. Integrability conditions that guarantee the existence of nodal positions realizing a given vector of edge lengths and dihedral angles were derived by Wang et al. [3]. The integrability conditions are formulated using moving frames associated with the triangles of the mesh. Already in earlier work, Lipman et al. [11, 12] used moving frames to define coordinates on triangle meshes. Our goal is to extend this line of work by providing the tools and structures needed for solving optimization problems that are formulated in the nonlinear coordinates.

Rigidity. While the existence and uniqueness results for the nonlinear coordinates require both, the edge lengths and the dihedral angles, rigidity results can already be obtained if only edge lengths are considered. For convex polytopes, Cauchy’s and Dehn’s rigidity theorems [13] show rigidity and infinitesimal rigidity and Gluck [14] showed that *almost all* simply-connected polyhedra are rigid. An example of polyhedra that allow for isometric continuous deformations, which are non-rigid, is Cornelly’s sphere [15]. In this paper, we will formulate infinitesimal rigidity in terms of NRIC. In recent work, Amenta and Rojas [16] studied the dihedral rigidity of polyhedra and parametrized triangle meshes via dihedral angles. Related to rigidity is the problem of computing an immersion from prescribed edge lengths. Algorithms for this problem were proposed by Boscaini et al. [17] and Chern et al. [18]. Relaxing the concept of rigidity, conformal geometry identifies metrics that differ only by a conformal factor. Crane et al. [19] study the numerical treatment of the integrability conditions for surfaces in this setting.

Near-isometric deformations. Isometric and near-isometric deformations are important for computational folding of piecewise flat or developable structures [20, 21, 22, 23]. The computation of near-isometric deformation can be done by simulating elastic shells consisting of stiff material with low bending resistance [24, 25, 26]. These materials yield ill-conditioned problems that are difficult to solve numerically. The NRIC perspective improves the numerical accessibility of such problems.

Shape interpolation. Shape interpolation, also called blending or morphing, is an important problem in geometry processing which is used for applications such as deformation transfer [27, 28], motion processing [29], example-based methods for shape editing [2], inverse kinematics [6, 30], and material design [31]. Approaches to shape interpolation based on linear coordinates use non-linear operations for blending the coordinates. For example, the gradient-domain approach of Xu et al. [32] extracts the rotational components from deformation gradients via polar decomposition and applies nonlinear blending operations to these components. While the nonlinear blending helps to compensate for linearization artifacts, it is a difficult task to estimate the local rotations that resolve large deformations. Kircher and Garland [33] and Gao et al. [34] introduce improved nonlinear blending operations for the rotational components. Winkler et al. [1] introduce a scheme for shape interpolation using nonlinear coordinates. Their method linearly blends edge lengths and dihedral angles and uses a multi-scale shape matching algorithm for constructing interpolating shapes. Fröhlich and Botsch [2] model 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. [35] 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. Model reduction approaches that enable real-time shape interpolation have been introduced by von Tycowicz et al. [36] and von Radziewsky et al. [37]. Related to shape interpolation are shape spaces, which are shape manifolds equipped with a Riemannian metric. Shape spaces are used for various applications in computer vision, computational anatomy, and medical imaging. For a general introduction to shape space and their applications, we refer to the textbook of Younes [38]. Kilian et al. [39] introduced a Riemannian metric on spaces of triangle meshes and show that concepts from Riemannian geometry such as the exponential map and parallel transport can be used for geometry processing tasks like deformation transfer, shape interpolation, and extrapolation. Heeren et al. [40, 4] propose an alternative physically-based metric on the shape space of triangle meshes that reflects the viscous dissipation required to physically deform a thin shell. Brandt et al. [41] derive a discrete curve shortening flow

in shape space and use it for processing animations of deformable objects. While the shape spaces study deformations of meshes with fixed connectivity, functional correspondences [42] can be used to blend [43] and analyze [44] pairs of meshes with different connectivity. In recent work, the functional correspondences of intrinsic and extrinsic geometry [45] and deformation fields [46] have been studied.

3. Background

In this section, we briefly review the work by Wang et al. [3] who introduced a discrete version of the fundamental theorem of surfaces. We consider a simplicial surface, which is a simplicial complex $\mathcal{K} = (\mathcal{V}, \mathcal{E}, \mathcal{F})$ consisting of sets of vertices \mathcal{V} , edges $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ and faces $\mathcal{F} \subset \mathcal{V} \times \mathcal{V} \times \mathcal{V}$ such that the topological space $|\mathcal{K}|$ obtained by identifying each face with a standard two-simplex and gluing the faces along the common edges is a two-dimensional manifold. A map $X: \mathcal{V} \rightarrow \mathbb{R}^3$ is called *generic* if for every face, the three vertices are in general position, *i.e.*, there is no straight line in \mathbb{R}^3 containing the three vertices. We define

$$\mathcal{N} := \{X(\mathcal{V}) \mid X: \mathcal{V} \rightarrow \mathbb{R}^3 \text{ generic}\} \subset \mathbb{R}^{3|\mathcal{V}|}, \quad (1)$$

which we denote the space of *discrete surfaces*. For any X , there is a unique map $X^*: |\mathcal{K}| \rightarrow \mathbb{R}^3$ that is continuous, an affine map of each simplex, and interpolates X at the vertices. X^* maps the faces of $|\mathcal{K}|$ to triangles in \mathbb{R}^3 , and, if X^* corresponds to a generic map X , none of the triangles degenerates. We will need this property to ensure that the elastic energies we consider in Section 5 are well-defined.

Since we assume that the underlying simplicial complex remains unchanged, by abuse of notation, we will often refer to the image $X(\mathcal{V})$ of the generic map simply as X . For a discrete surface $X \in \mathcal{N}$, we denote by $l(X) = (l_e(X))_{e \in \mathcal{E}}$ its vector of edge lengths and by $\theta(X) = (\theta_e(X))_{e \in \mathcal{E}}$ its vector of dihedral angles. Wang *et al.* studied necessary and sufficient conditions that an arbitrary tuple $(l, \theta) \in \mathbb{R}^{2|\mathcal{E}|}$ is induced by a discrete surface. The first necessary condition is the triangle inequality, *i.e.*

$$\mathcal{T}_f(l) > 0 \quad \text{for all } f \in \mathcal{F}, \quad (T)$$

where $\mathcal{T}_f(l) = \begin{pmatrix} l_i + l_j - l_k & l_i - l_j + l_k & -l_i + l_j + l_k \end{pmatrix}$ for a face $f \in \mathcal{F}$ with edge lengths l_i, l_j, l_k and the above inequality is to be understood componentwise. Thus, by combining the maps for all faces and extending constantly to dihedral angles we obtain a linear map $\mathcal{T}: \mathbb{R}^{2|\mathcal{E}|} \rightarrow \mathbb{R}^{3|\mathcal{F}|}$ and we see that (T) defines an open convex polytope in $\mathbb{R}^{2|\mathcal{E}|}$. The next set of conditions is referred to as *discrete integrability conditions*. They ensure that we can integrate the local change of geometry induced by the lengths and angles to reconstruct the immersed discrete surface. In other words, the immersion is invariant with respect to the start and order of the reconstruction. For a face $f \in \mathcal{F}$ with (immersed) edges $E_1, E_2, E_3 \in \mathbb{R}^3$, one defines the *standard discrete frame* F_f as the orthogonal matrix with rows $\frac{E_1}{\|E_1\|}$, $\frac{E_1 \times N_f}{\|E_1 \times N_f\|}$, and N_f , where $N_f \in S^2$ is the unit face normal. The normal component requires our discrete surfaces to be globally orientable which we will assume in the following. Then the transition between frames F_j and F_i of adjacent faces $f_i, f_j \in \mathcal{F}$ and a common edge $e \in \mathcal{E}$ can be described by a rotation matrix R_{ij} with $F_j = F_i R_{ij}$. This rotation decomposes into three elementary rotations, *i.e.*

$$R_{ij} = R_2(\gamma_{e,i})R_0(-\theta_e)R_2(\gamma_{j,e}), \quad (2)$$

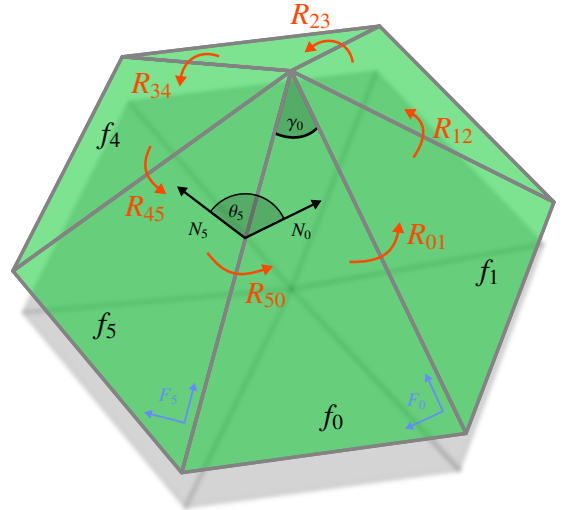


Figure 1: Construction of integrability condition for the 6-loop of faces around a vertex. Specifically, the transition rotation $R_{50} = R_0(\theta_5)R_2(\gamma_0)$ (orange) is constructed from the dihedral angle θ_5 and interior angle γ_0 (both black). It transforms the frame F_5 into frame F_0 (both blue, normal vector not shown), *i.e.* $F_0 = F_5 R_{50}$. The other transition rotations are constructed in the same way and applying them sequentially yields the integrability condition $R_{01}R_{12}R_{23}R_{34}R_{45}R_{50} \stackrel{!}{=} \text{Id}$.

where $R_k(\varphi) \in SO(3)$ denotes a rotation around the k th standard basis vector in \mathbb{R}^3 by $\varphi \in [0, 2\pi]$ and $\gamma_{e,j}$ and $\gamma_{i,e}$ denote the angles between the common edge and the first vector of F_i resp. F_j . In particular, the transition rotations are completely determined by the lengths and angles using the law of cosines. Now let $\mathcal{V}_0 \subset \mathcal{V}$ be the index set of interior vertices. Then for each $v \in \mathcal{V}_0$, which is the center of a n_v -loop of faces f_0, \dots, f_{n_v-1} and edges e_0, \dots, e_{n_v-1} connected to v , we obtain a *closing condition*. To this end, one chooses the frames F_0, \dots, F_{n_v-1} such that e_i always coincides with the first basis vector in F_i . Consequently, the corresponding transition rotations simplify to $R_{ij} = R_0(\theta_i)R_2(\gamma_j)$, where γ_j is the interior angle at v in f_j with $j = i + 1$ modulo n_v and θ_i is the dihedral angle at e_i . Applying the transition property $F_j = F_i R_{ij}$ sequentially along the loop, the identity $F_0 = F_0 \prod_{i=0}^{n_v-1} R_{i,(i+1) \bmod n_v}$ must hold for immersed discrete surfaces. This can be phrased as the integrability condition

$$\mathcal{I}_v(l, \theta) := \prod_{i=0}^{n_v-1} R_{i,(i+1) \bmod n_v} \stackrel{!}{=} \text{Id} \quad (\text{I})$$

for the n_v -loop of faces around all *interior* vertices $v \in \mathcal{V}_0$. Note that the transition rotations in (I) in fact depend on (l, θ) .

Wang et al. [3] proved that the necessary conditions (T) and (I) are indeed sufficient (for simply connected surfaces). In detail, their discrete fundamental theorem of surfaces reads: *If $(l, \theta) \in \mathbb{R}^{2|\mathcal{E}|}$ satisfies (T) and (I), there exists $X \in \mathcal{N}$ (unique up to rigid body motions) such that $l(X) = l$ and $\theta(X) = \theta$.* They also extended this to non-simply connected surfaces but to simplify the exposition, we restrict ourselves to the simply connected case.

4. The NRIC manifold of edge lengths and dihedral angles

In this section, we consider the Nonlinear and Rotation-Invariant Coordinates (NRIC) given as a vector $z = (l_e, \theta_e)_{e \in \mathcal{E}} \in \mathbb{R}^{2|\mathcal{E}|}$ that lists all the edge lengths and dihedral angles of a discrete surface. Using the integrability conditions, we describe the manifold of discrete surfaces as a submanifold of $\mathbb{R}^{2|\mathcal{E}|}$ and derive a scheme for computing its tangent spaces.

NRIC manifold. We consider the map

$$Z: \mathcal{N} \rightarrow \mathbb{R}^{|\mathcal{E}|} \times \mathbb{R}^{|\mathcal{E}|}, \quad X \mapsto (l(X), \theta(X)) \quad (\text{3})$$

that associates to any discrete surface the vector stacking its edge length and dihedral angles. The image of (3) describes the submanifold

$$\mathcal{M} := Z(\mathcal{N}) = \{z \in \mathbb{R}^{2|\mathcal{E}|} \mid \exists X \in \mathcal{N}: Z(X) = z\}. \quad (\text{4})$$

of $\mathbb{R}^{2|\mathcal{E}|}$ that we call the NRIC manifold.

Implicit description. In the following, we will use the conditions (I) and (T) to derive an implicit description of \mathcal{M} . Directly using condition (I) leads to nine scalar constraints per vertex, which is a redundant description since $SO(3)$ is a three-dimensional manifold. Instead, we will introduce a reformulation using unit quaternions as an equivalent representation of spatial rotations.

To this end, let us first briefly recall the necessary basics of quaternions and their relation to spatial rotation such that this section is self-contained, for a detailed treatment we refer to standard textbooks such as [47]. 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 $q_1 q_2$ and hence this correspondence is a homomorphism between $SO(3)$ and the unit quaternions.

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

$$q_0(\varphi) := \cos \frac{\varphi}{2} + \mathbf{i} \sin \frac{\varphi}{2}, \quad q_2(\varphi) := \cos \frac{\varphi}{2} + \mathbf{k} \sin \frac{\varphi}{2} \quad \text{for } \varphi \in [0, 2\pi). \quad (5)$$

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

$$q_{ij} := q_0(\theta_i) q_2(\gamma_j), \quad (6)$$

where again f_0, \dots, f_{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 f_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

$$\mathcal{Q}_v(l, \theta) := \text{vec} \left(\prod_{i=0}^{n_v-1} q_{i, (i+1) \bmod n_v} \right) \stackrel{!}{=} 0 \quad (\text{I}_q) \quad (7)$$

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

Now, we can rewrite the manifold defined in (4) as

$$\mathcal{M} = \{z \in \mathbb{R}^{2|\mathcal{E}|} \mid \mathcal{T}(z) > 0, \mathcal{Q}(z) = 0\}. \quad (7)$$

Here we have collected all constraints in a vector-valued functional $\mathcal{Q}: \mathbb{R}^{2|\mathcal{E}|} \rightarrow \mathbb{R}^{3|\mathcal{V}_0|}$ with $\mathcal{Q} = (\mathcal{Q}_v)_{v \in \mathcal{V}_0}$. Obviously, \mathcal{Q}_v depends solely on the edge lengths of the adjacent faces of v and the dihedral angles at edges centered at v . Given $z \in \mathbb{R}^{2|\mathcal{E}|}$ with $\mathcal{Q}(z) = 0$ one can easily reconstruct vertex coordinates $X \in \mathcal{N}$ with $Z(X) = z$. For a robust and stable reconstruction for $\mathcal{Q}(z) \neq 0$, we refer to Section 7.

Tangent space. The implicit formulation (7) consists of the triangle inequalities defining an open convex polytope and of the nonlinear integrability conditions, which define 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

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

where $D\mathcal{Q}(z)$ is a matrix in $\mathbb{R}^{3|\mathcal{V}_0|, 2|\mathcal{E}|}$. Partial derivatives of \mathcal{Q}_v are given by the chain rule as

$$\partial_{z_k} \mathcal{Q}_v(z) = \text{vec} \left(\sum_{i=0}^{n_v-1} q_{01}(z) \dots \partial_{z_k} q_{i, i+1}(z) \dots q_{n_v-1, 0}(z) \right), \quad (8)$$

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 . We provide details on the gradient computation as well as an implementation in terms of a MATHEMATICA notebook in the supplementary material.

To illustrate the NRIC manifold and its tangent spaces, we will for the remainder of the section discuss an immediate application. With the tangent space at hand, one can verify the *infinitesimal rigidity* of a discrete surface

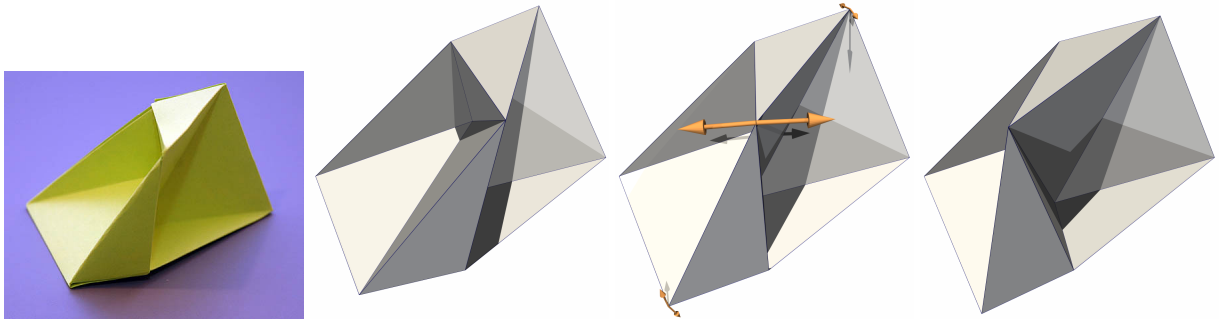


Figure 2: The tangent space reveals an infinitesimal isometric variation at the classical Steffen’s polyhedron (middle). Indeed, extrapolating in this positive (left) resp. negative (right) direction (solely in the θ component) allows for isometric deformations. The extrapolation is implemented via an incremental addition of the infinitesimal isometric variation coupled with a back projection onto \mathcal{M} . See also video in supplementary material for an animation.

with NRIC $z \in \mathcal{M}$. In fact, a necessary condition for the existence of continuous one-parameter families of isometric deformations starting at z is the 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 2. Note, however, that this is surely not a sufficient condition, which we can also observe in Figure 3. Thus, we simply verify 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 $\ker \left(B_{T_z\mathcal{M}} \mid B_\theta \right)$, 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. In Figure 3, we show that the Origami cylinder considered by Bös et al. [21] does not allow for an isometric deformation path which leads to a compression by folding. Indeed, the only nontrivial infinitesimal isometric variation is indicated by arrows (top, right). However, there is no nontrivial family of isometric deformation with this shape as the initial shape. As discussed by Bös et al. [21] the experimental paper deformation (top, left) is not isometric. This is reflected by our criteria for infinitesimal isometric variations when we additionally enforce the dihedral angles on the upper and lower plate to remain constant which leads to $\lambda_0 = 0.015$ clearly indicating the nonexistence of such a variation.

5. Nonlinear energy and geometry of the NRIC manifold

So far, we have introduced a differential structure on \mathcal{M} . Going forward, it will be essential to additionally consider an elastic deformation energy \mathcal{W} between different NRIC as it provides a dissimilarity measure on \mathcal{M} . Although different choices for \mathcal{W} are possible, we will primarily focus on the hyperelastic deformation energy from Heeren et al. [4]. To this end, we reformulate this energy in NRIC to define a physically-motivated Hessian structure on \mathcal{M} . This 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 [2].

Based on models from mathematical physics, the hyperelastic energy used in [4] consists of two separate contributions, *i.e.*

$$\mathcal{W} = \mathcal{W}_{\text{mem}} + \delta^2 \mathcal{W}_{\text{bend}}. \quad (9)$$

From a physical point of view, the first term \mathcal{W}_{mem} will measure the stretching of edges and triangles, *i.e.* local *membrane distortions*. Likewise, the second term $\mathcal{W}_{\text{bend}}$ will measure the difference in bending between triangles, *i.e.* local *bending distortions*. In particular, the global weight δ represents the thickness of a thin elastic material represented by the discrete surface.

In the following, we will investigate separately how the membrane and bending energy introduced in [4] can be reformulated in NRIC. Note that the membrane energy has originally been proposed in [40] whereas the bending

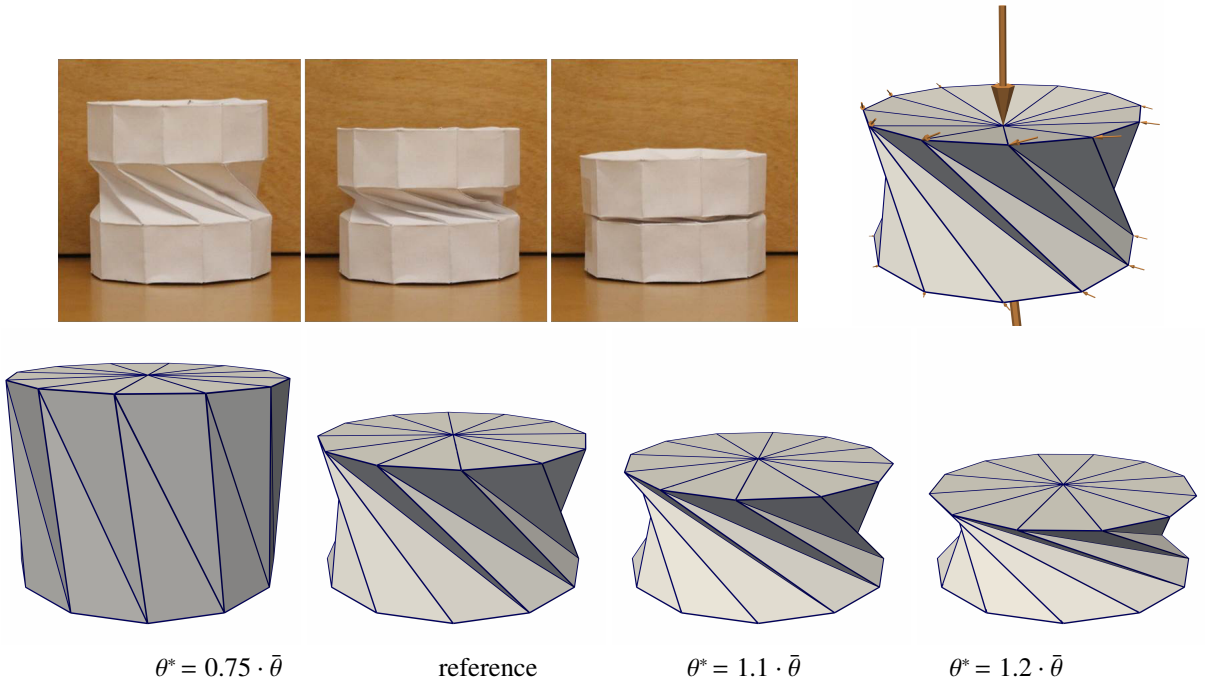


Figure 3: Top: Almost isometric compression of an Origami cylinder as depicted in Bös et al. [21] (left, with permission) along with the only infinitesimal isometric variation (right, indicated by arrows). Bottom: Optimizing (15) on NRIC manifold with $\delta=0$ and hard constraints on target angles 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). See also supplementary video.

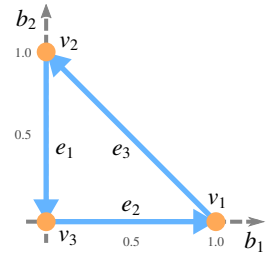
energy has been taken from [48].

Membrane energy. Let $X \in \mathcal{N}$ be a discrete surface and $X^*: |\mathcal{K}| \rightarrow \mathbb{R}^3$ the corresponding continuous, piecewise linear map that interpolates the vertices. The derivative of X^* is constant over each triangle, and, since X is a generic map, the derivative has full rank. This implies that X^* induces a metric G (also called first fundamental form) on $|\mathcal{K}|$. This metric is defined in the interior of the faces and along the edges. It enables measuring the length of arbitrary curves in $|\mathcal{K}|$ and makes $|\mathcal{K}|$ a metric space. Two discrete surfaces, X and \tilde{X} , induce two different metrics, G and \tilde{G} , on $|\mathcal{K}|$. The metric distortion tensor $\mathcal{G}[X, \tilde{X}]$ is defined as the symmetric tensor that at any point in the interior of a triangle satisfies

$$G(\mathcal{G}[X, \tilde{X}]v, w) = \tilde{G}(v, w)$$

for any pair v, w of tangential vectors. The membrane energy evaluates the trace and the determinant of $\mathcal{G}[X, \tilde{X}]$.

For our purpose, it is essential to be able to evaluate the distortion tensor for discrete surfaces given by their NRIC z and \tilde{z} directly without having to reconstruct vertex positions first. In the following, we derive an explicit formula for $\mathcal{G}[z, \tilde{z}]$. For discrete surfaces, the metric and the distortion tensor are constant for every triangle. Consider an arbitrary triangle f in \mathbb{R}^3 . We parametrize f with an affine map $\phi: t \rightarrow f$, where t is the right angled triangle in \mathbb{R}^2 shown in the inset figure. The standard basis b_1, b_2 of \mathbb{R}^2 agrees with second edge and the negative of the first edge of t . Then $d\phi(b_1) = E_2(f)$ and $d\phi(b_2) = -E_1(f)$, where $E_1(f), E_2(f)$ denote the edge vectors of f . Thus, the metric on t induced by ϕ is



$$G|_f = \begin{pmatrix} \|d\phi(b_1)\|^2 & \langle d\phi(b_2), d\phi(b_1) \rangle \\ \langle d\phi(b_2), d\phi(b_1) \rangle & \|d\phi(b_2)\|^2 \end{pmatrix} = \begin{pmatrix} \|E_2(f)\|^2 & -\langle E_1(f), E_2(f) \rangle \\ -\langle E_1(f), E_2(f) \rangle & \|E_1(f)\|^2 \end{pmatrix}, \quad (10)$$

The entries of the metric can be expressed in terms of the length of the edges of f . The diagonal entries are the

squared length of the second and the first edge. The off-diagonal entries are given by 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 . In our case, this is the interior angle of a triangle which can be computed from its edge lengths by the law of cosines. For two NRIC z and \tilde{z} , we can use the formula to compute the metrics $G|_f$ and $\tilde{G}|_f$ for every f . Then, the distortion tensor is given as $\mathcal{G}[X, \tilde{X}]|_f := (G|_f)^{-1} \tilde{G}|_f$. We want to note that the resulting distortion tensor depends on the chosen domain and parametrization. However, we consider isotropic materials for which the membrane energy depends only on the trace and determinant of the distortion tensor. Since the determinant and the trace are invariant under coordinate transformations, we obtain the same results independently of the chosen domain and parametrization. Similarly the roles of the edges could be exchanged, for example, one could consider the second and third edge. This would alter the parametrization and therefore yield a different distortion tensor. Still, the relevant quantities, the determinant and the trace of \mathcal{G} , would be the same.

Having established that the distortion tensor is completely given by the NRIC of discrete surfaces, we can now adapt the membrane energy from [40] applying a nonlinear energy density to it, which has a global minimum at the identity.

Definition 1 (Membrane energy). For a simplicial surface \mathcal{K} , we define the membrane energy on NRIC $z, \tilde{z} \in \mathbb{R}^{2|\mathcal{E}|}$ as

$$\mathcal{W}_{\text{mem}}[z, \tilde{z}] = \sum_{f \in \mathcal{F}} a_f \cdot W_{\text{mem}}(\mathcal{G}[z, \tilde{z}]|_f), \quad (11)$$

where

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

for positive material constants μ and λ and a_f is the area of f computed from edge lengths by Heron's formula.

For more explicit formulas of the energy in terms of edge lengths we refer to the appendix and for the energy's derivatives to the supplementary material.

Bending energy. Next, we adapt the *Discrete Shells* bending energy [48] also used in [4]. 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.

Definition 2 (*Discrete Shells* bending energy). For a simplicial surface \mathcal{K} , we define the *Discrete Shells* bending energy on NRIC $z, \tilde{z} \in \mathbb{R}^{2|\mathcal{E}|}$ as

$$\mathcal{W}_{\text{bend}}[z, \tilde{z}] = \sum_{e \in \mathcal{E}} \frac{(\theta_e - \tilde{\theta}_e)^2}{d_e} l_e^2, \quad (12)$$

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

Finally, we combine the membrane and bending energy in a weighted sum.

Definition 3 (Nonlinear deformation energy). Let \mathcal{K} be a simplicial surface and let $z, \tilde{z} \in \mathbb{R}^{2|\mathcal{E}|}$ be two NRIC. The nonlinear deformation energy is defined by

$$\mathcal{W}_{nl}[z, \tilde{z}] = \mathcal{W}_{\text{mem}}[z, \tilde{z}] + \delta^2 \mathcal{W}_{\text{bend}}[z, \tilde{z}], \quad (13)$$

where \mathcal{W}_{mem} is the membrane energy from Definition 1, $\mathcal{W}_{\text{bend}}$ is the bending energy from Definition 2, and δ represents the thickness of the material.

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 $W_{\text{mem}}(\mathcal{G}[z, \tilde{z}]|_f) \rightarrow \infty$ for $\tilde{a}_f \rightarrow 0$. To see this, we recognize that $\det \mathcal{G}[z, \tilde{z}]|_f = (\det G|_f)^{-1} \det \tilde{G}|_f = a_f^{-2} \tilde{a}_f^2$ and hence $-\log \det \mathcal{G}[z, \tilde{z}]|_f \rightarrow \infty$ when \tilde{a}_f 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.* $W_{\text{mem}}(\mathcal{G}[z, \tilde{z}]|_f) \rightarrow \infty$ if one of the components

of $\mathcal{T}_f(l)$ approaches zero meaning that we get close to violating one of the triangle inequalities. Especially, we set $W_{\text{mem}}(\mathcal{G}[z, \bar{z}])_f = \infty$ if $\mathcal{T}_f(l) > 0$ does not hold. This allows us to characterize the NRIC manifold \mathcal{M} by

$$\mathcal{M} = \{z \in \mathbb{R}^{2|\mathcal{E}|} \mid \mathcal{W}_{nl}[z^*, z] < \infty \text{ for a fixed } z^* \in \mathcal{M}, \mathcal{Q}(z) = 0\}, \quad (14)$$

avoiding the explicit dependence on the triangle inequalities (T) we had before. Note, however, that the integrability conditions (\mathbf{I}_q) are still necessary as finite energy does not guarantee their attainment. The characterization (14) will be helpful later on to devise efficient numerical schemes for solving variational problems on \mathcal{M} .

Quadratic model. Previously, Fröhlich and Botsch [2] used a quadratic deformation model for NRIC, *i.e.* they considered the weighted quadratic energy

$$\mathcal{W}_q[z, z^*] = \sum_{e \in \mathcal{E}} \alpha_e \|l_e - l_e^*\|^2 + \delta^2 \sum_{e \in \mathcal{E}} \beta_e \|\theta_e - \theta_e^*\|^2. \quad (15)$$

In fact, almost the same model has been used in [48] 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 $\bar{z} \in \mathbb{R}^{2|\mathcal{E}|}$. For example, the authors in [48, 2] set in a related context $\alpha_e = l_e^{-2}$ and $\beta_e = l_e^2 d_e^{-1}$, whereas Heeren et al. [49] have chosen $\alpha_e = d_e l_e^{-2}$, for $e \in \mathcal{E}$. Here the (physical) parameter $\delta^2 > 0$ trades the impact on length variations off against angle variations and can be considered as the squared thickness of the material as before. This quadratic energy has no 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 (14). 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 [2], it often leads to natural-looking deformations and we will consider it in some of our examples.

Riemannian metric. For each $z \in \mathcal{M}$, a Riemannian metric g_z is a symmetric, positive definite quadratic form on the tangent space $T_z \mathcal{M}$ measuring the cost of an infinitesimal variation in tangential direction. In our context a tangential vector $w \in T_z \mathcal{M}$ splits into two components $w = (w_l, w_\theta)$, where w_l is the variation of edge lengths and w_θ the variation of dihedral angles. Following Rayleigh's paradigm, $\frac{1}{2}$ times the Hessian of an elastic deformation energy can be considered as a Riemannian metric on the space of discrete surfaces if it is positive definite. Precisely, we obtain the metric for tangent vectors $v, w \in \mathbb{R}^{2|\mathcal{E}|}$ via

$$g_z(v, w) = v^T \left(\frac{1}{2} \text{Hess } \mathcal{W}[z, z] \right) w. \quad (16)$$

As investigated in [4], this is true for the energy defined in (13) with the choice of membrane and bending energies made above. Furthermore, it holds for the quadratic energy (15) if we choose all weights to be positive. With the metric at hand, one can define the Riemannian distance on \mathcal{M} and compute for instance shortest geodesic curves, *cf.* Section 8.

6. Variational problems on the manifold

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 (7), 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 $\mathcal{E}: \mathbb{R}^{2|\mathcal{E}|} \rightarrow \mathbb{R}$ the task is to

$$\begin{aligned} & \underset{z \in \mathbb{R}^{2|\mathcal{E}|}}{\text{minimize}} && \mathcal{E}(z) \\ & \text{subject to} && \mathcal{Q}_v(z) = 0 \text{ for each } v \in \mathcal{V}_0, \\ & && \mathcal{T}_f(z) > 0 \text{ for each } f \in \mathcal{F}. \end{aligned} \quad (\text{OPT})$$

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

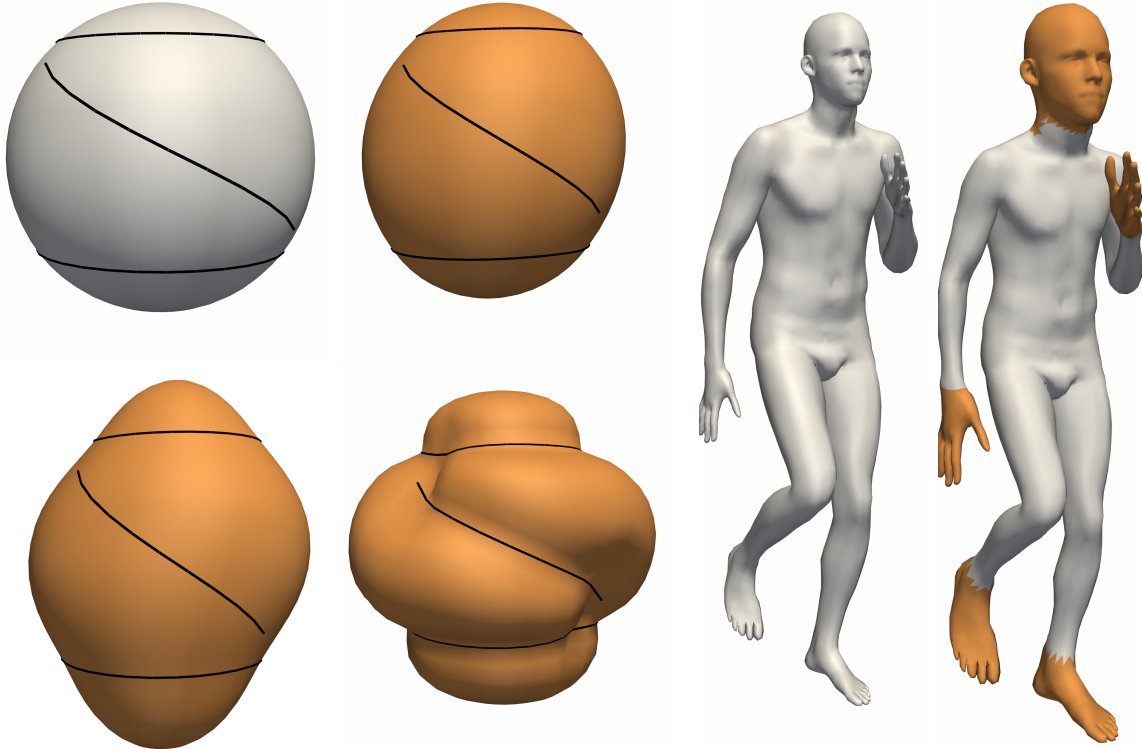


Figure 4: Left: Unit sphere (grey) with black constraint curves to be shortened by means of equality constraints on edge lengths along with different results for varying bending parameter $\delta^2 = 10^{-0.2,4}$ in the nonlinear objective (13). Right: Same experiment but with (orange) constraint areas where the target edge lengths were increased by 30% to simulate a brushing tool.

A simple example of an objective functional \mathcal{E} is given by the dissimilarity to some given z^* on the linear space $\mathbb{R}^{2|\mathcal{E}|}$ 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 we have used the nonlinear energy defined in (13) 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.* [50]).

Ensuring triangle inequalities. One crucial problem we encounter when we try to solve (OPT) are the triangle inequalities which lead to $3|\mathcal{F}|$ 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 from Section 4 and Section 5. We achieve this by a modified line search. First, recall that the set $\mathcal{Z}_{\mathcal{T}} = \{z \in \mathbb{R}^{2|\mathcal{E}|} \mid \mathcal{T}(z) > 0\}$ defines an open connected subset of $\mathbb{R}^{2|\mathcal{E}|}$. 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 $\mathcal{Z}_{\mathcal{T}}$. We accomplish this using backtracking, *i.e.* reducing the stepsize β^k until $z^{k+1} = z^k + \beta^k d^k \in \mathcal{Z}_{\mathcal{T}}$ holds. In implementations, this can easily be achieved by setting $Q_v(z) = \infty$ if $\mathcal{T}_f(z) \not> 0$ for any face f adjacent to $v \in \mathcal{V}_0$.

We can obtain an even more natural approach when we work with the nonlinear membrane energy \mathcal{W}_{mem} . Recall that in Section 5 we introduced the characterization (14) 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 [51, 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 $\mathcal{Z}_{\mathcal{T}}$.

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

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

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

$$DL(z, \lambda) = (D_z L(z, \lambda), D_\lambda L(z, \lambda))^T = (D_z \mathcal{E}(z) - D_z \mathcal{Q}(z) \cdot \lambda, -\mathcal{Q}(z))^T = 0, \quad (17)$$

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 [52, 51]. 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 Lagrangian is defined by

$$L(z, \lambda, \mu) = \mathcal{E}(z) - \mathcal{Q}(z) \cdot \lambda + \frac{\mu}{2} \|\mathcal{Q}(z)\|_2^2, \quad (18)$$

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. On the other hand, λ is updated by an increasingly accurate estimation of the correct multipliers λ^* solving (17). 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 [51]. We want to remark that though our problem (OPT) involves strict inequality constraints, the local convergence theory for the augmented Lagrange method given in [51, Chapter 17] applies to our problem. The triangle inequality constraints define an open set and thus (OPT) can be seen as an equality-constrained problem over an open set. As [51, 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.

An explicit algorithmic description of the method with all involved parameters and derivatives will be provided in the appendix.

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 [51, 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

$$(D_z^2 L(z^k, \lambda^k, \mu^k) + \tau^k \text{Id}) d^k = -D_z L(z^k, \lambda^k, \mu^k). \quad (19)$$

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.* [53, Chapter 1]. In some instances, we could speed-up the minimization by first performing a small number of iterations with a BFGS approximation of the Hessian.

To compute the descent direction as above, we need the gradient and the Hessian of our constraint functionals \mathcal{Q} .

We already evaluated $D_z Q \in \mathbb{R}^{3|\mathcal{V}_0|, 2|\mathcal{E}|}$ in (8) and compute for the Hessian of Q

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

the components as

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

which can also be evaluated with $O(n_v)$ cost. We provide further details on the Hessian computation in the supplementary material.

7. Reconstruction of an immersion

In the preceding sections, we discussed the geometry as well as constrained optimization problems on the NRIC manifold \mathcal{M} , *i.e.* in terms of edge lengths and dihedral angles. The remaining task is to reconstruct for given $z \in \mathcal{M}$ an immersion $X \in \mathcal{N}$ of the simplicial surface in \mathbb{R}^3 with $z = Z(X)$. Beyond the computation of vertex coordinates for $z \in \mathcal{M}$, one frequently asks for an approximate immersion $X \in \mathcal{N}$ for $z \notin \mathcal{M}$ such that $Z(X) \approx z$. Indeed, the computation of just approximate immersions is required in case of

- modeling of deformations energies in terms of dihedral angles and edge lengths, *i.e.* using the linear embedding space $\mathbb{R}^{2|\mathcal{E}|}$ instead of \mathcal{M} ,
- using a high tolerance for the fulfillment of the constraints in the augmented Lagrange or a penalty method,
- coordinates z which are only approximately computed numerically.

Thus, we ask for a reconstruction map $\mathcal{R}: \mathbb{R}^{2|\mathcal{E}|} \rightarrow \mathcal{N}$, such that $\mathcal{R}|_{\mathcal{M}}$ is the right inverse of Z with $Z \circ \mathcal{R} = \text{Id}_{\mathcal{M}}$, where $\text{Id}_{\mathcal{M}}$ is the identity on the NRIC manifold. Let us emphasize that by the rigid body motion invariance of our NRIC approach, we obtain $\mathcal{R} \circ Z(X) = QX$, where $X \in \mathcal{N}$ and $Q \in SE(3)$ is some rigid body motion acting on the immersion.

Variational approach. For some given $z \in \mathbb{R}^{2|\mathcal{E}|}$, where not necessarily $z \in \mathcal{M}$, we are looking for the nodal positions $X \in \mathcal{N}$, such that the resulting $Z(X) \in \mathcal{M}$ is as close as possible to z . Fröhlich and Botsch [2] have used a least squares functional to build a *variational reconstruction*, *i.e.* they compute

$$\arg \min_{X \in \mathbb{R}^{3|\mathcal{V}|}} \mathcal{W}[z, Z(X)] \quad (20)$$

with $\mathcal{W}[z, \tilde{z}]$ describing the proximity of z and \tilde{z} . Note that the solution is only unique up to a rigid deformation. A simple example of a quadratic functional \mathcal{W} is given by (15) as it was used by Fröhlich and Botsch. They proposed a Gauß-Newton method [51, Section 10.3] to solve (20), however, for general $z \in \mathbb{R}^{2|\mathcal{E}|}$, one still has to solve a high-dimensional and nonlinear optimization problem in $\mathbb{R}^{3|\mathcal{V}|}$. If $z \in \mathcal{M}$ and the initialization of the Gauß-Newton method is close to the solution, it usually converges in only a few iterations. However, if z is far away from \mathcal{M} and the initialization is poor, artifacts may occur.

Constructive approach. For $z \in \mathcal{M}$ a *constructive reconstruction* of the immersion $X \in \mathcal{N}$ can be derived by means of frames and transition rotations, as they were used to define the integrability conditions (*cf.* Section 3). This method was introduced by Lipman et al. [11] and further elaborated in [3]. Before we investigate a combination of the

constructive and the variational approach for $z \notin \mathcal{M}$, let us briefly review the constructive reconstruction. Assume we are given an admissible target $z = (l, \theta) \in \mathcal{M}$. Since the reconstruction from lengths and angles is only defined up to rigid body motions, we further assume that we are given the position of one vertex and the orientation of an adjacent triangle f_0 in the form of a frame F_0 . If $f \in \mathcal{F}$ is a neighboring triangle of f_0 , one can infer the induced transition rotations R_{0f} from z and thus determine $F_f = F_0 R_{0f}$. Repeating this iteratively, one can construct frames for all faces. This algorithm is indeed well-defined on simply connected triangulations due to the integrability constraints, *i.e.* if there are two paths connecting a triangle f to f_0 , then the frames constructed along the two paths coincide. Given frames for all faces and hence the orientation of all triangles, one can finally reconstruct the nodal positions.

Adaptive spanning trees. Next, we take into account a violation of the integrability condition (I) for $z \notin \mathcal{M}$ and ask for a reconstruction of an approximate immersion. Let us remark that Wang et al. [3] handle non-admissible targets $z \notin \mathcal{M}$ when modeling surfaces via a modification of l and θ . They study a least-square type functional and relax in a least square sense the identity $F_f = F_0 R_{0f}$ as well as (D.1).

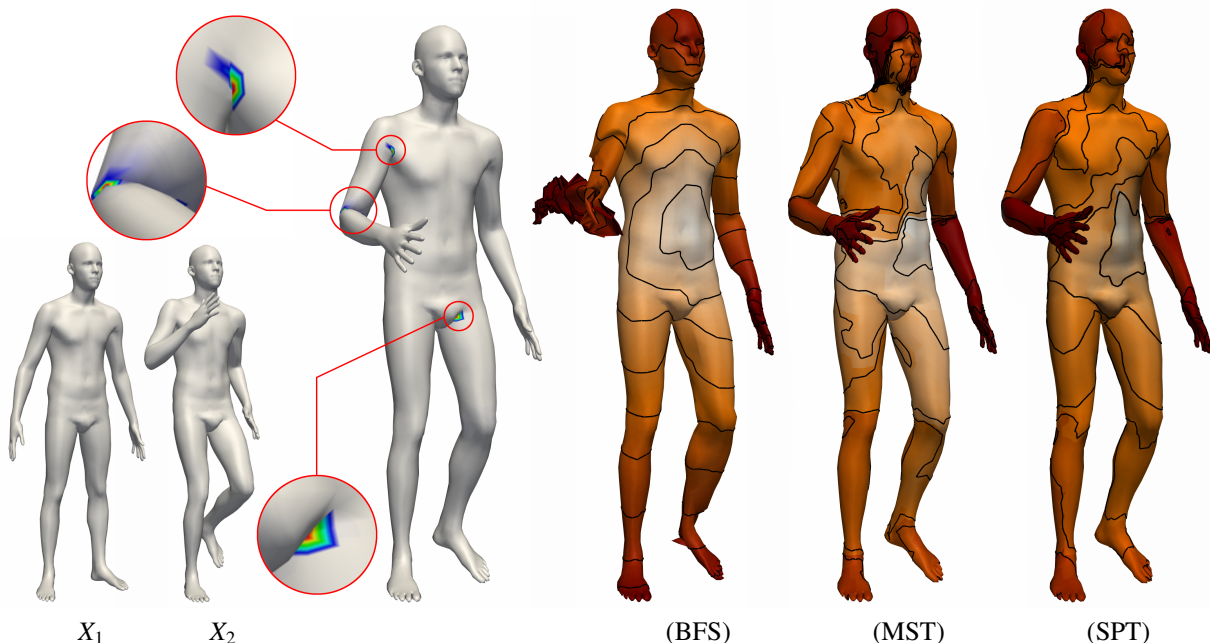


Figure 5: Left: Input shapes X_1 and X_2 (taken from [54]) and reconstruction from linear average $(Z(X_1) + Z(X_2))/2 \notin \mathcal{M}$ with the local violations of the integrability condition as color map. Note that violations are highly concentrated, *e.g.* in the armpits. Rightmost shapes: reconstruction using breadth-first search (BFS), minimal spanning tree (MST) and shortest path tree (SPT). The triangulation is color-coded with respect to the order of traversal. See video for an animation of the reconstruction order.

The direct frame-based reconstruction with a spanning tree of the dual graph built by breadth-first search is very sensitive to violations of the integrability. In fact, the errors occurring when walking over such a violation propagate to all following frames and are even amplified, *cf.* Figure 5. In addition, reconstructing nodal positions of a face along two different paths connecting it to the initial face f_0 , where at least one is passing a zone of violated integrability conditions, leads to substantially different results and thus visual artifacts. However, the regions of violation appear frequently to be highly localized in practice, *cf.* Figure 5. Thus, we build a spanning tree which traverses faces with violation of the integrability condition as late as possible in the mesh traversal for the reconstruction. To this end, we consider the dual graph of \mathcal{K} with weights based on the integrability condition. Each dual edge corresponds to a primal edge $e = (v, v') \in \mathcal{E}$ and we can assign to this dual edge a scalar weight reflecting the lack of integrability w_e by averaging the violation of integrability at the two adjacent vertices v and v' :

$$w_e := \frac{|\text{tr } \mathcal{I}_v(z) - 3| + |\text{tr } \mathcal{I}_{v'}(z) - 3|}{2}, \quad (21)$$

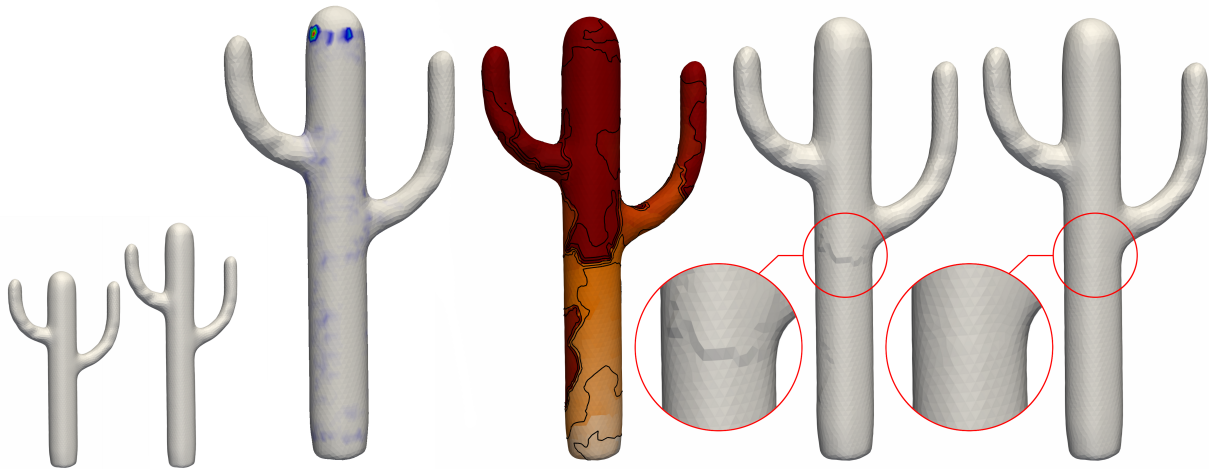


Figure 6: Left: Input shapes X_1 and X_2 (small) and reconstruction from linear average $(Z(X_1) + Z(X_2))/2 \notin \mathcal{M}$ with local violations of the integrability condition as color map. Rightmost shapes: reconstruction order using a minimal spanning tree (order as colormap), visual artefacts along the body and final solution after one step of a Gauß-Newton smoothing.

where $I_v(z)$ is the matrix-valued map defined in (I). Note that $I_v(z) \in SO(3)$ and that $\text{tr } Q = 3 \Leftrightarrow Q = \text{Id}$ for $Q \in SO(3)$. Now, the weights (21) are used to build a spanning tree adapted to the problem. The first variant is to construct a minimal spanning tree (MST) of the dual graph, which is built such that the sum of all edge weights in the tree is minimal. Such a minimal spanning tree can be computed via Prim's algorithm and provides a way to traverse the dual graph while avoiding unnecessarily large violations of the integrability. Another variant is to construct a shortest path tree (SPT), which is built such that the path distance from the root to any other vertex in the tree is the shortest in the whole dual graph. This can be achieved by Dijkstra's algorithm and provides a way to traverse the dual graph such that for each face the sum of integrability violation along the dual path used for its reconstruction is minimal. We compare both novel variants against the original breadth-first search (BFS) in Figure 5. A pseudo code of the entire algorithm is given in the appendix. Formally, the algorithm — using either (MST) or (SPT) — is only defined for $z \in \mathcal{M}$. In particular, the triangle inequality is assumed to be defined. However, the algorithm can easily be generalized for $z \in \mathbb{R}^{2|\mathcal{E}|}$ with $\mathcal{T}_f(z) \leq 0$ for some face f by setting the interior angles of f to zero. By our definition of (I) and edge weights (21) those triangles will be automatically considered as late as possible in the adaptive algorithm.

Preassembled tree. The runtime of the tree-based reconstruction algorithm is dominated by the cost for the construction of the spanning tree. Thus, if one aims at reconstructing numerous immersions of a discrete surface with the same connectivity and a very high resolution (*i.e.* many vertices) it would be desirable to use a preassembled spanning tree. Of course, this preassembled tree has to be reasonable for a large set of lengths and angles. If we are given samples $z_1, \dots, z_n \in \mathcal{M}$ and corresponding edge weights $w^1, \dots, w^n \in \mathbb{R}^{|\mathcal{E}|}$, we simply set $w_e = \max_i w_e^i$ for all $e \in \mathcal{E}$ and construct a spanning tree based on these weights.

Hybrid approach. Just applying our constructive reconstruction algorithm works very well for $z \notin \mathcal{M}$ as long as the violations are localized as in Figure 5. However, we observe imperfect results when the violations are distributed over larger areas, *cf.* Figure 6. In this case, we suggest a hybrid method combining our robust constructive reconstruction and as a post processing the variational reconstruction. In detail, we make use of the (still imperfect) output of our constructive reconstruction to initialize the variational reconstruction as in (20). Typically, a single Gauß-Newton step is sufficient to smooth the result adequately (*cf.* Figure 6).

8. Numerical experiments and comparisons

In this section, we study qualitative and quantitative properties of the NRIC tools and demonstrate that in particular for modeling with near isometric deformations the NRIC manifold outperforms established methods that consider

nodal positions as primal degrees of freedom. To this end, we pick up the generic variational problem (OPT) introduced in Section 6 together with the proposed augmented Lagrange method. In the following, we discuss different objective functionals \mathcal{E} in (18) and depending on the application additional constraints. Note, however, that the *constraint functional* Q in (18), which describes the NRIC manifold implicitly via (7), remains unchanged.

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.* [36]. 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

$$\mathcal{E}(z) = \sum_{i=1}^n \mu_i \mathcal{W}[Z(X_i), z]. \quad (22)$$

In Figure 7, 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 (13) in (22).

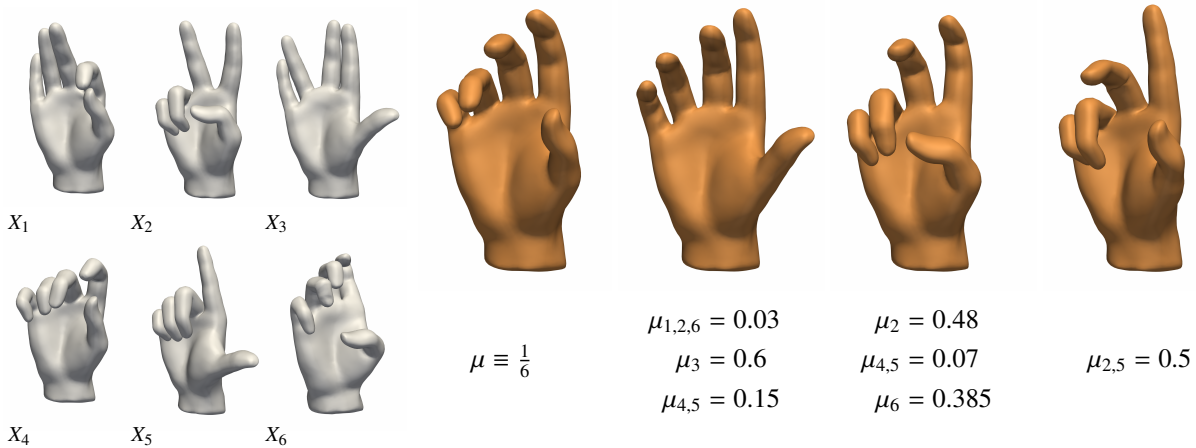


Figure 7: Reconstruction of nodal positions from elastic averages of six hand poses (grey) with different (convex) weights $\mu \in \mathbb{R}^6$ computed as minimizer of (22) on the NRIC manifold.

Isometric deformations via additional constraints. Interesting applications can be described by considering (OPT) along with the simple objective $\mathcal{E}(z) = \mathcal{W}[z^*, z]$ but with additional, simple coordinate constraints. For example, in Figure 4 we have seen experiments where we posed lengths constraints $l_i = l_i^*$ for $i \in I$ on the coordinates $z = (l, \theta)$ for some index set $I \subset \mathcal{E}$ 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 $e \in \mathcal{E}$ along with $\theta_i = \text{const} \neq 0$ if $i \in I$ for some index set $I \subset \mathcal{E}$. Note, that under these length constraints the nonlinear and quadratic energy approach agree if we compute the weights in (15) from the reference z^* . For example, in Figure 8 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 (13) on \mathcal{M} , we obtain a perfect isometric deformation as indicated by the vanishing discrete Gauß curvature (Figure 8, right). In comparison, vertex-based methods as [48] or [40] do not achieve a perfect isometry—even when computed with a very high membrane stiffness (Figure 8, left). For the optimization in nodal positions, we used the energy $X \mapsto \mathcal{W}_{nl}[z^*, Z(X)]$ with a shell thickness parameter $\delta = 10^{-3}$. In fact, further reducing δ one observes numerical instabilities. This is 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.

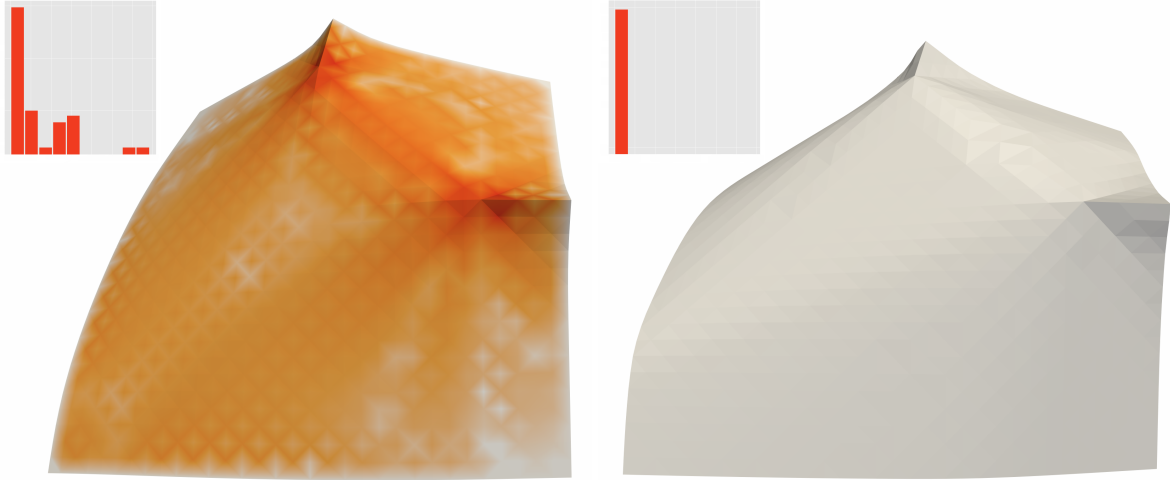



Figure 8: Paper folding with local constraints for dihedral angle: simulation in vertex space (left) leads to infinitesimal isometry violations whereas the result in NRIC is completely isometric (right). The absolute value of discrete Gauss curvature (as angle defect) is shown using the color map 0  0.03, which is zero everywhere on the right. Furthermore, the corresponding histograms are plotted aside the surfaces.

Besides vanishing Gauss curvature, pure isometric deformations exhibit further characteristics, as illustrated in Figure 9. In this example, we have a very similar setup as in Figure 8 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 9 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 [55, 56]. 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 9 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 9, far left).

Time-discrete 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. We focus on the computation of time discretized *geodesics* in the NRIC manifold here. On the manifold \mathcal{M} with metric g defined in (16) a geodesic connecting end points z_A and z_B in \mathcal{M} is the curve $z: [0, 1] \rightarrow \mathcal{M}$ minimizing the *path energy* $\int_0^1 g_{z(t)}(\dot{z}(t), \dot{z}(t)) dt$ subject to $z(0) = z_A$ and $z(1) = z_B$. In particular, the minimizer $(z(t))_{0 \leq t \leq 1}$ obeys the constant speed property $g_z(\dot{z}(t), \dot{z}(t)) = \text{const.}$ Heeren et al. [40] introduced the concept of *time-discrete geodesics* (in a vertex-based approach) as a variational approximation of continuous geodesics. For $K \in \mathbb{N}$, they consider a finite sequence z_0, \dots, z_K in \mathcal{M} with $z_0 = z_A$ and $z_K = z_B$ and define the *time-discrete path energy*

$$E[z_0, \dots, z_K] = K \sum_{k=1}^K \mathcal{W}[z_{k-1}, z_k], \quad (23)$$

where \mathcal{W} is assumed to be a local approximation of the squared Riemannian distance and $z_k \approx z(k/K)$. Minimizers (z_0, \dots, z_K) of (23) for fixed end points z_0 and z_K are said to be time-discrete geodesics. In particular, they obey a discrete constant speed property, *i.e.* there is a uniform energy distribution $\mathcal{W}[z_{k-1}, z_k] \approx \text{const}$ along the curve.

The concept of discrete geodesics directly translates to the NRIC manifold and the path energy in (23) can be considered as an objective functional in (OPT). Note, however, that this increases the number of free variables sub-

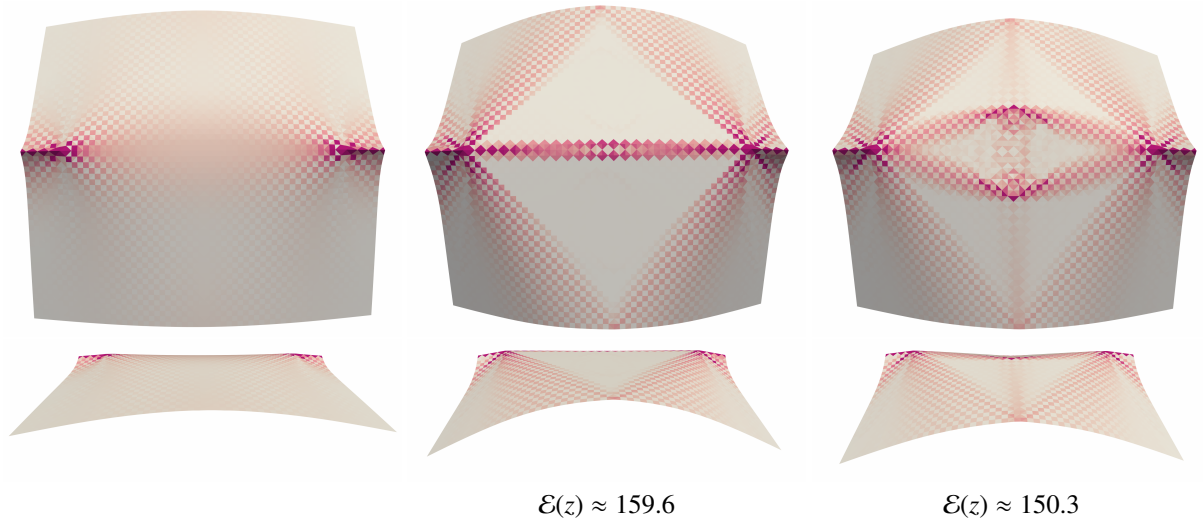


Figure 9: Once more paper folding with local constraints for dihedral angle: simulation in vertex space (left) leads to infinitesimal isometry violations whereas the results in NRIC are completely isometric (middle, right). The result in the middle shows a local minimum obtained by the augmented Lagrange 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 $0 \leq \text{color} \leq \geq 0.005$, flat triangular regions can only be seen in the NRIC simulations.

stantially. In Figure 10, we show different time-discrete geodesics in NRIC where we use the quadratic deformation energy (15) in (23). 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 $\mathbb{R}^{2|\mathcal{E}|}$, which corresponds to a naive transfer of the projection approach by Fröhlich and Botsch [2]. 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 11 we compare the computation of (almost) *isometric* geodesic paths between perfectly isometric end shapes taken from Dudte et al. [57] to vertex-based methods. A similar example is shown in Figure 12, 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 12 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.

Timings. Lastly, let us discuss the runtimes of the proposed method, where all computations were performed using a desktop computer with an Intel(R) Core(TM) i7-4790 CPU and 16 GB RAM. In our framework, we use the Eigen library [58] for linear algebra tasks and CHOLMOD [59] for the Cholesky decomposition. At first, we list timings for the reconstruction performed without parallelization. As a representative example, we report on timings measured on the Dyna dataset [54] (*cf.* Figure 5) where $|\mathcal{V}| \approx 6.9k$. The generation of (MST) or (SPT) takes about 12ms, the generation of a spanning tree via (BFS) takes about 1ms. The traversal of a spanning tree takes about 5ms, and one Gauß-Newton iteration is done in 330ms (with 80% spending in the linear solver). Next, computing the entries of the Hessian of the constraint functional Q requires, again without parallelization, 16ms for the discrete surfaces considered in Figure 8 and 112ms for the Dyna dataset considered in Figure 4. Detailed timings for the optimization described in Section 6 are listed in Table 1. Note, that the evaluation of the augmented Lagrangian L and its derivatives requires substantially more time than computing the entries of $D_z^2 Q$. This originates from computing the square of $D_z Q$ and assembling the Hessian in CSR format because these operations do not benefit from parallelization in our current implementation.

Compared to computations in nodal positions, our method requires more memory due to the increased number of primal degrees of freedom. However, because this number is approximately twice the number of nodal positions the total memory consumption only increases by a constant factor of approximately four.

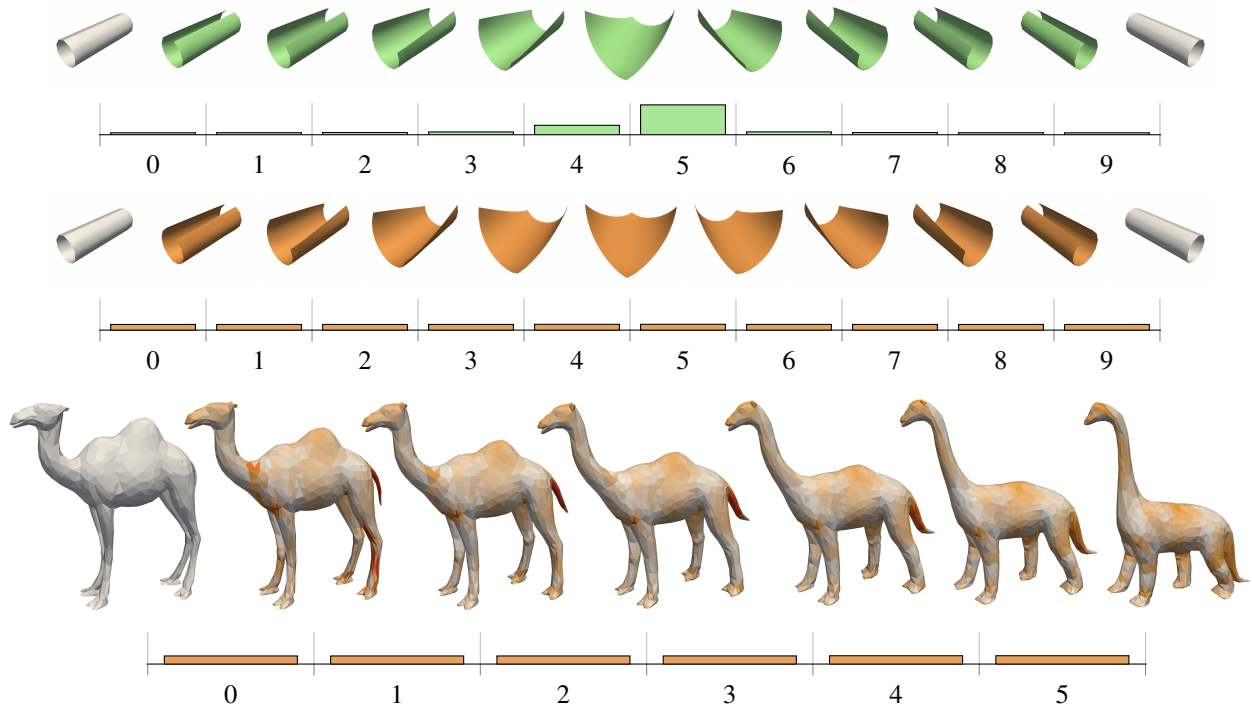


Figure 10: Bottom: discrete geodesic in NRIC with input data from [16] and membrane distortion as colormap (0 \leq \geq 1); above: Linear interpolation in ambient space $\mathbb{R}^{2|E|}$ as in [2] with energy distribution (green) vs. geodesic interpolation on \mathcal{M} with constant energy distribution (orange).

9. Conclusion

We introduce a framework that allows us to pose and solve geometric optimization problems in terms of NRIC. The framework is built on several novel concepts. First, we introduce a Riemannian structure for the NRIC manifold stemming from an implicit description via integrability conditions and a physically-motivated nonlinear elastic energy. In particular, we demonstrate how the notion of a tangent space can be used to identify infinitesimal isometric variations. Second, we present an approach based on the augmented Lagrange method and a modified line search for solving generic optimization problems in NRIC. Third, we develop a hybrid algorithm for the reconstruction of nodal positions from length and angle coordinates that uses a mesh traversal to initialize a Gauß–Newton solver. We tested our framework on different problems including shape interpolation and paper folding. A particular strength is the simulation of true isometric deformations—a task where well-established vertex-based methods often fail.

Limitations and challenges. We see great potential in using NRIC for geometric optimization problems and expect that the techniques we present will be further developed. We plan to formulate an extended geodesic shape space calculus (*cf.* [4]) including geodesic extrapolation and parallel transport in NRIC and expect to profit from the rigid motion invariance of the coordinates and their robustness for near-isometric deformation. In the context of a statistical analysis of shapes, our NRIC formulation enables direct processing of input data without an a priori rigid co-registration. To this end, our NRIC manifold is a natural starting point for the development of a corresponding Riemannian principal component analysis.

Though our experiments demonstrate the benefits of NRIC-based optimization, our current framework has several limitations and poses challenges in making the optimization more efficient. First, the current implementation can 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 modelling have only a small number of

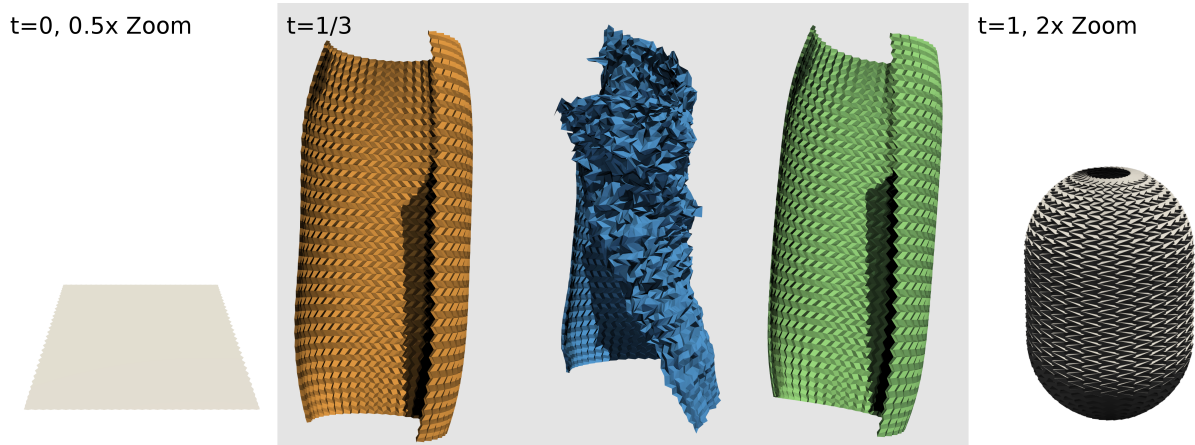


Figure 11: Intermediate shapes at $t = 1/3$ of a discrete geodesic between two perfectly isometric end shapes (grey) taken from Dudte et al. [57] obtained via NRIC optimization (orange) and vertex-based methods as in [40] (blue) resp. [2] (green). Note that we preserve the isometry due to our hard length constraints. In contrast, the vertex-based methods get either stuck in local minima (blue) or reveal artifacts such as unnatural asymmetries (green).

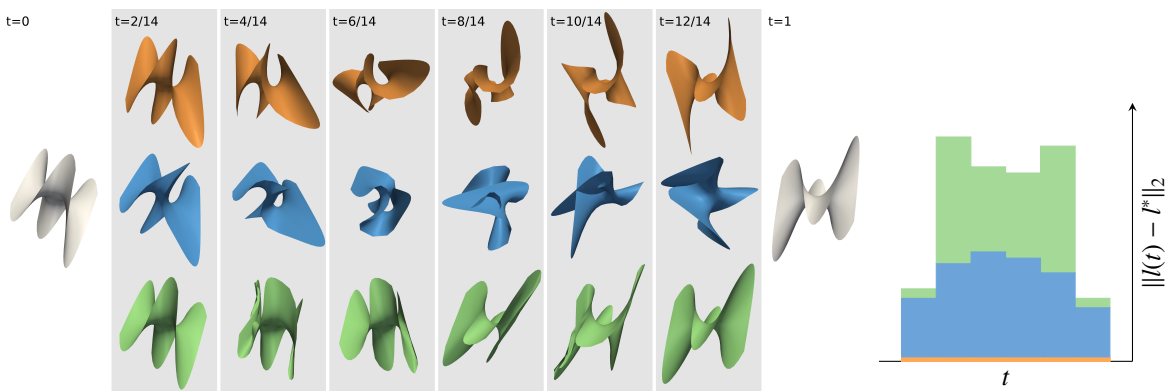


Figure 12: Isometric geodesic paths. Left input shape (I^*, θ^*) as hyperbolic monkey saddle, right input shape is the reflection ($I^*, -\theta^*$). Comparison of discrete geodesics computed in NRIC (orange, perfectly isometric) and by methods based on nodal positions (*i.e.* Heeren et al. [40] (blue) and Fröhlich and Botsch [2]). The latter approaches are not able to resolve pure isometric geodesics as indicated by a histogram of varying lengths on the right. See also supplementary video.

generators of the homology group. However, this necessity of complicated constraints is a general limitation of our method compared to nodal positions.

Second, a fundamental challenge is to reduce the number of degrees of freedom and integrability conditions. Our current framework works with $2|\mathcal{E}|$ variables and $3|\mathcal{V}|$ integrability conditions per shape. This implies a larger number of variables compared to optimization in nodal positions, which in turn means increased memory requirement and more costly iterations. Here, it might be worthwhile to explore model reduction approaches. Furthermore, the triangle inequality constraints are in general challenging to take care of in the implementation. We found in all our experiments that the proposed adapted line search, especially in conjunction with the nonlinear deformation energy, was able to handle them robustly.

Finally, we aim to account for point constraints in our NRIC-based optimization. These type of constraints frequently appear, for instance, in physical simulations as forces or boundary conditions. This could be accomplished by performing a partial reconstruction of the points with attached constraints using an explicit formula that results from tracing the paths in Algorithm 3. Then the derivatives of the explicit formula need to be computed with respect to NRIC to enable their use in optimization problems which might be a feasible task for modern automatic

example		iterations		avg. times per Newton iteration		
Figure	N	aug. Lagrange	Newton	evaluation	solve	line search
4 (left, avg.)	5220	18	83	30 ms	23 ms	3 ms
4 (right)	41328	14	121	281 ms	201 ms	15 ms
7 (avg.)	36552	13	203	263 ms	244 ms	18 ms
8	6272	83	477	34 ms	34 ms	7 ms
9 (right)	24832	46	454	173 ms	84 ms	7 ms
10 (top)	5760	10	15	16 ms	69 ms	2 ms
10 (bottom)	36000	12	65	338 ms	732 ms	10 ms
11	69936	11	44	626 ms	301 ms	22 ms
12	32240	11	173	218 ms	193 ms	5 ms

Table 1: Performance statistics of our approach on the different examples shown before. From left to right: number of degrees of freedom, number of iterations of the augmented Lagrange method, total number of Newton iterations, average time for evaluation of function and derivatives per Newton iteration, average time for computing τ and solving the linear system per Newton iteration, and average time for line search per Newton iteration.

differentiation frameworks. Nonetheless, 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.

Appendix A. Rotations q_{ij}

As noted above, each induced transition rotation depends on one dihedral angle and on the three edge lengths of a triangle. We simplify our notation for the computation of the partial derivatives and define

$$\hat{q}(\theta, a, b, c) := q_0(\theta) q_2 \left(\arccos \left(\frac{a^2 + b^2 - c^2}{2ab} \right) \right), \quad (\text{A.1})$$

where a, b , and c are the edge lengths and θ is the dihedral angle. To simplify the notation even further in the following, we define the rational function $Q(a, b, c) = \frac{a^2 + b^2 - c^2}{2ab}$. The angle of the rotation around the second standard basis vector is given by the law of cosines and using trigonometric formulas we can thus simplify its matrix representation to

$$q_2(\arccos Q) = \sqrt{\frac{1+Q}{2}} + \sqrt{\frac{1-Q}{2}} \mathbf{k}.$$

If we multiply this with the rotation around the zeroth standard basis vector we arrive at

$$\hat{q}(\theta, a, b, c) = \cos \frac{\theta}{2} \sqrt{\frac{1+Q}{2}} + \sin \frac{\theta}{2} \sqrt{\frac{1+Q}{2}} \mathbf{i} - \sin \frac{\theta}{2} \sqrt{\frac{1-Q}{2}} \mathbf{j} + \cos \frac{\theta}{2} \sqrt{\frac{1-Q}{2}} \mathbf{k}.$$

With this representation at hand, it is now possible to compute its first and second derivative. In particular, this is a viable task for a symbolic differentiation tool and we refer to the supplementary material for the results.

Appendix B. Local membrane energy

To understand the nonlinear membrane energy \mathcal{W}_{mem} better and prepare the computation of its derivatives, we study in this section the contribution of a single triangle $f \in \mathcal{F}$. To this end, let a, b, c be the edge lengths of f in the undeformed configuration z and $\tilde{a}, \tilde{b}, \tilde{c}$ the corresponding edge lengths of the deformed \tilde{z} . Our goal now is to express $a_f \cdot W_{\text{mem}}(\mathcal{G}[z, \tilde{z}]_f)$ in terms of these lengths.

We start with the components of G as given in (10). The diagonal entries are of course simply given by b^2 and a^2 . For the off-diagonal entries, recall

$$\langle E_1(f), E_2(f) \rangle = \|E_1(f)\| \|E_2(f)\| \cos(\angle(E_1(f), E_2(f))),$$

where by the law of cosines we have $\cos \angle(E_1(f), E_2(f)) = \frac{a^2+b^2-c^2}{2ab}$. Hence, we get the representation

$$\hat{G}(a, b, c) = \begin{pmatrix} b^2 & -\frac{1}{2}(a^2 + b^2 - c^2) \\ -\frac{1}{2}(a^2 + b^2 - c^2) & a^2 \end{pmatrix}, \quad (\text{B.1})$$

where we use the hat to indicate the local representation of an object. From this, we immediately compute the determinant as

$$\det \hat{G}(a, b, c) = \frac{1}{4}(a+b+c)(-a+b+c)(a-b+c)(a+b-c) = 4a_f(a, b, c)^2, \quad (\text{B.2})$$

with $a_f(a, b, c)$ the triangle area obtained via Heron's formula. The local representation of the discrete distortion tensor is given by

$$\hat{\mathcal{G}}(a, b, c, \tilde{a}, \tilde{b}, \tilde{c}) = \hat{G}(a, b, c)^{-1} \hat{G}(\tilde{a}, \tilde{b}, \tilde{c}). \quad (\text{B.3})$$

By the multiplicativity of the determinant, we obtain

$$\det \hat{\mathcal{G}} = (\det \hat{G}(a, b, c))^{-1} \det \hat{G}(\tilde{a}, \tilde{b}, \tilde{c}) = \frac{a_f(\tilde{a}, \tilde{b}, \tilde{c})^2}{a_f(a, b, c)^2} \quad (\text{B.4})$$

for the determinant of the distortion tensor. Computing the trace requires in contrast an explicit representation of $\hat{\mathcal{G}}$ and we finally obtain

$$\text{tr} \hat{\mathcal{G}} = \frac{1}{8a_f(a, b, c)^2} \left(\tilde{a}^2(-a^2 + b^2 + c^2) + \tilde{b}^2(a^2 - b^2 + c^2) + \tilde{c}^2(a^2 + b^2 - c^2) \right). \quad (\text{B.5})$$

Together, we have assembled all components to write the contribution of f as

$$a_f(a, b, c) \cdot W_{\text{mem}}(\hat{\mathcal{G}}(a, b, c, \tilde{a}, \tilde{b}, \tilde{c})) \quad (\text{B.6})$$

with W_{mem} the energy density from Definition 1. We provide the derivatives of the membrane energy along with the derivatives of the bending energy in the supplementary material.

Appendix C. Optimization algorithms

In this section, we will discuss the algorithmic details of the optimization procedure introduced in Section 6 for our generic problem (OPT). First, we start with detailed description of the augmented Lagrange method adapted from [51] in Algorithm 1, which provides all parameters related to the increase of the penalty parameter and the update of the Lagrange multiplier estimates. In almost all examples, our default parameters $\mu^0 = 10$, $\lambda^0 = 0$, $\mu_+ = 100$, and $\eta_+ = 0.9$ worked reasonably well. Only in the paper folding examples (*cf.* Figure 8 and Figure 9), we decreased μ_+ and η_+ because we noticed this leads to local minima with lower energy values as was discussed in Section 8.

To compute the approximate minimizer, we use the variation of the Newton-type method introduced before which requires the first and second derivatives of the augmented Lagrangian L . The first derivative of L is given by

$$D_z L(z, \lambda, \mu) = D_z \mathcal{E}(z) - D_z \mathcal{Q}(z) \cdot \lambda + \mu D_z \mathcal{Q}(z)^T \mathcal{Q}(z)$$

and hence the second derivative turns out to be

$$D_z^2 L(z, \lambda, \mu) = D_z^2 \mathcal{E}(z) + D_z^2 \mathcal{Q}(z) \cdot (\mu \mathcal{Q}(z) - \lambda) + \mu D_z \mathcal{Q}(z)^T D_z \mathcal{Q}(z),$$

Algorithm 1 Augmented Lagrange method for (OPT), [51, Alg. 17.4]

Input: Initial NRIC z^0 , initial penalty μ^0 , initial multipliers λ^0 , constraint tolerance ε_Q , optimality tolerance ε_L , penalty increase factor μ_+ , tolerance increase exponent η_+

Output: Approximate solution z^k to (OPT)

```

1: for all  $k = 0, 1, 2, \dots, k_{max}$  do
2:   Compute approximate minimizer  $z^{k+1}$  of  $L(\cdot, \lambda^k, \mu^k)$  with  $\|D_z L(z^{k+1}, \lambda^k, \mu^k)\|_2 \leq \omega_k$ 
3:   if  $\|Q(z^{k+1})\|_\infty \leq \varepsilon_Q$  and  $\|D_z L(z^{k+1}, \lambda^k, \mu^k)\|_2 \leq \varepsilon_L$  then                                ▶ Stopping criterion
4:     return  $z^{k+1}$ 
5:   end if
6:   if  $\|Q(z^{k+1})\|_\infty \leq \eta^k$  then                                                                                                    ▶ Recompute multiplier
7:      $\lambda^{k+1} = \lambda^k - \mu^k Q(z^{k+1})$ 
8:      $\mu^{k+1} = \mu^k$ 
9:      $\eta^{k+1} = \max(\eta^k / (\mu^{k+1})^{\eta_+}, \varepsilon_Q)$ 
10:     $\omega^{k+1} = \omega^k / \mu^{k+1}$ 
11:   else                                                                                                                            ▶ Increase penalty parameter
12:      $\lambda^{k+1} = \lambda^k$ 
13:      $\mu^{k+1} = \mu_+ \cdot \mu^k$ 
14:      $\eta^{k+1} = \max(1 / (\mu^{k+1})^{0.1}, \varepsilon_Q)$ 
15:      $\omega^{k+1} = 1 / \mu^{k+1}$ 
16:   end if
17: end for

```

where

$$D_z^2 Q \cdot (\mu Q(z) - \lambda) = \left(\sum_{v \in \mathcal{V}_0} \partial_{z_i} \partial_{z_k} Q_v \cdot (\mu Q_v(z) - \lambda_v) \right)_{l,k=1,\dots,2|\mathcal{E}|}.$$

Again, the detailed derivatives of the integrability constraints and of the energy are provided in the supplementary material. Now, we can provide all the steps of this method in an integrated fashion as Algorithm 2. Again, in almost all examples our default parameters $\tau_+ = 10$ and either $\beta = 10^{-3}$ or $\beta = 10^{-4}$ worked well.

Appendix D. Direct reconstruction algorithm

In Section 7, we have already outlined how the frames can be constructed iteratively using the transition rotations induced by $z \in \mathcal{M}$. To complete the description of the reconstruction algorithm, we also need to detail how to construct the nodal positions. To this end, consider a face with a given discrete frame F and target edge lengths l_0, l_1, l_2 . Then we obtain embedded edge vectors $E_i \in \mathbb{R}^3$ with $\|E_i\| = l_i$ for $i = 0, 1, 2$ by

$$E_1 = l_0 F \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, E_2 = l_1 F \begin{pmatrix} -\cos \gamma_3 \\ \sin \gamma_3 \\ 0 \end{pmatrix}, E_3 = l_2 F \begin{pmatrix} -\cos \gamma_2 \\ -\sin \gamma_2 \\ 0 \end{pmatrix}, \quad (\text{D.1})$$

and finally nodal positions $(X_i)_{i=0,1,2}$ such that $E_i = X_{i-1} - X_{i+1}$. Note that the inner angles γ_2 and γ_3 can be obtained from the edge lengths using the law of cosines. The complete reconstruction algorithm is summarized in Algorithm 3. Note that in practice, we need to construct at most one nodal position per face (except for the first face f_0), and often even none as the positions are already determined.

Acknowledgements

The authors thank Laszlo Bardos from cutoutfoldup.com for the photo of Steffen's polyhedron used in Figure 2, furthermore the authors of [21] and the American Society of Mechanical Engineers for the photo in Figure 3. Moreover, we thank Etienne Vouga and Levi Dudte for the meshes used in Figure 11, Friedrich Bös for information on

Algorithm 2 Line search Newton-type method for L , [51, Alg. 3.1, 3.2, 3.3]

Input: Initial NRIC z^k , gradient tolerance ω^k , initial/minimal shift β , shift increase factor τ_+ **Output:** Approximate minimizer z^{k+1} of $L(\cdot, \lambda^k, \mu^k)$

```
1: Set  $z^{k,0} = z^k$ 
2: for  $j = 0, 1, 2, \dots, j_{max}$  do
3:   Evaluate  $L(z^{k,j}, \lambda^k, \mu^k)$ ,  $D_z L(z^{k,j}, \lambda^k, \mu^k)$ , and  $D_z^2 L(z^{k,j}, \lambda^k, \mu^k)$  ▷ Evaluation
4:   if  $\|D_z L(z^{k,j}, \lambda^k, \mu^k)\| \leq \omega^k$  then ▷ Stopping criterion
5:     return  $z^{k+1} = z^{k,j}$ 
6:   end if
7:   if  $\min_i (D_z^2 L(z^{k,j}, \lambda^k, \mu^k))_{ii} > 0$  then ▷ Determine shift
8:      $\tau_j = 0$ 
9:   else
10:     $\tau_j = -\min_i (D_z^2 L(z^{k,j}, \lambda^k, \mu^k))_{ii} + \beta$ 
11:   end if
12:   loop
13:     Attempt Cholesky decomposition of  $D_z^2 L(z^{k,j}, \lambda^k, \mu^k) + \tau_j \text{Id}$ 
14:     if factorization succeeds then
15:       stop
16:     else
17:        $\tau_j \leftarrow \max(\tau_+ \cdot \tau_j, \beta)$ 
18:     end if
19:   end loop
20:   Solve  $(D_z^2 L(z^{k,j}, \lambda^k, \mu^k) + \tau_j \text{Id}) d_j = -D_z L(z^{k,j}, \lambda^k, \mu^k)$  ▷ Descent direction
21:   repeat ▷ Line search
22:      $\alpha_j = 0.5\alpha_j$ 
23:   until  $L(z^{k,j} + \alpha_j d_j, \lambda^k, \mu^k) \leq L(z^{k,j} + \alpha_j d_j, \lambda^k, \mu^k) + 0.1\alpha_j D_z L(z^{k,j}, \lambda^k, \mu^k)^T d_j$ 
24:   Set  $z^{k,j+1} = z^{k,j} + \alpha_j d_j$ 
25: end for
```

the construction of the Origami cylinder in Figure 3, and Carlos Rojas for the animal meshes used in Figure 10. The authors from Bonn gratefully acknowledge the support provided by the Austrian Science Fund (FWF) through project NFN S117 and by the Deutsche Forschungsgemeinschaft (DFG) through the Hausdorff Center for Mathematics (GZ 204711, Project ID 390685813).

Declarations of interest: none

References

- [1] T. Winkler, J. Drieseberg, M. Alexa, K. Hormann, Multi-scale geometry interpolation, *Comput. Graph. Forum* 29 (2010) 309–318.
- [2] S. Fröhlich, M. Botsch, Example-driven deformations based on discrete shells, *Comput. Graph. Forum* 30 (2011) 2246–2257.
- [3] Y. Wang, B. Liu, Y. Tong, Linear surface reconstruction from discrete fundamental forms on triangle meshes, *Comput. Graph. Forum* 31 (2012) 2277–2287.
- [4] B. Heeren, M. Rumpf, P. Schröder, M. Wardetzky, B. Wirth, Exploring the geometry of the space of shells, *Comput. Graph. Forum* 33 (2014) 247–256.
- [5] Y. Yu, K. Zhou, D. Xu, X. Shi, H. Bao, B. Guo, H.-Y. Shum, Mesh editing with poisson-based gradient field manipulation, *ACM Trans. Graph.* 23 (2004) 644–651.
- [6] R. W. Sumner, J. Popović, Deformation transfer for triangle meshes, *ACM Trans. Graph.* 23 (2004) 399–405.
- [7] O. Sorkine, D. Cohen-Or, Y. Lipman, M. Alexa, C. Rössl, H.-P. Seidel, Laplacian surface editing, in: *Symposium on Geometry Processing*, 2004, pp. 175–184.
- [8] Y. Lipman, O. Sorkine, D. Cohen-Or, D. Levin, C. Roessl, H. P. Seidel, Differential coordinates for interactive mesh editing, in: *Proceedings Shape Modeling International*, 2004, pp. 181–190.
- [9] M. Botsch, O. Sorkine, On linear variational surface deformation methods, *IEEE Transactions on Visualization and Computer Graphics* 14 (2008) 213–230.
- [10] M. P. DoCarmo, *Differential geometry of curves and surfaces*, Prentice Hall (1976).
- [11] Y. Lipman, O. Sorkine, D. Levin, D. Cohen-Or, Linear rotation-invariant coordinates for meshes, *ACM Trans. Graph.* 24 (2005) 479–487.

Algorithm 3 Direct frame-based reconstruction

Input: $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, $z \in \mathbb{R}^{2|\mathcal{E}|}$, $X_0 \in \mathbb{R}^3$ and pair (F_0, f_0)

Output: Nodal positions X with $Z(X) \approx z$

- 1: Evaluate the discrete integrability map $\mathcal{I}_v(z)$ for all $v \in \mathcal{V}$
- 2: Define edge weights (21) for $e = (vv') \in \mathcal{E}$

Alternative MST:

- 3: Construct minimal spanning tree of the dual graph based on w

Alternative SPT:

- 3: Construct shortest path tree of the dual graph based on w
 - 4: Traverse the dual graph of \mathcal{G} following the constructed tree
 - 5: **for all** dual edges (f_i, f_j) **do**
 - 6: Construct induced transition rotation R_{ij}
 - 7: Construct frame $F_j = F_i R_{ij}$
 - 8: Compute interior angles $\gamma_1, \gamma_2, \gamma_3$ of f_j
 - 9: Determine edge vectors E_1, E_2, E_3 according to (D.1)
 - 10: Determine nodal positions for f_j
 - 11: **end for**
-

- [12] Y. Lipman, D. Cohen-Or, R. Gal, D. Levin, Volume and shape preservation via moving frame manipulation, *ACM Trans. Graph.* 26 (2007) 5.
- [13] M. Dehn, über die Starrheit konvexer Polyeder, *Mathematische Annalen* (1916) 466–473.
- [14] H. Gluck, *Geometric Topology*, Springer, 1975.
- [15] R. Connelly, A counterexample to the rigidity conjecture for polyhedra, *Publications Mathématiques de l’IHÉS* 47 (1977) 333–338.
- [16] N. Amenta, C. Rojas, Dihedral rigidity and deformation, in: *Proceedings Canadian Conference on Computational Geometry*, 2018, pp. 219–226.
- [17] D. Boscaini, D. Eynard, D. Kourounis, M. M. Bronstein, Shape-from-operator: Recovering shapes from intrinsic operators, *Comput. Graph. Forum* 34 (2015) 265–274.
- [18] A. Chern, F. Knöppel, U. Pinkall, P. Schröder, Shape from metric, *ACM Trans. Graph.* 37 (2018) 63:1–63:17.
- [19] K. Crane, U. Pinkall, P. Schröder, Spin transformations of discrete surfaces, *ACM Trans. Graph.* 30 (2011).
- [20] M. Kilian, S. Flöry, Z. Chen, N. J. Mitra, A. Sheffer, H. Pottmann, Curved folding, *ACM Trans. Graph.* 27 (2008) 75:1–75:9.
- [21] F. Bös, E. Vouga, O. Gottesman, M. Wardetzky, On the incompressibility of cylindrical origami patterns, *ASME J. Mech. Des.* 139 (2016) 021404–021404–9.
- [22] O. Stein, E. Grinspun, K. Crane, *ACM Trans. Graph.* 37 (2018).
- [23] M. Rabinovich, T. Hoffmann, O. Sorkine-Hornung, Discrete geodesic nets for modeling developable surfaces, *ACM Trans. Graph.* 37 (2018).
- [24] R. Burgoon, Z. J. Wood, E. Grinspun, Discrete shells origami, in: *Proceedings 21st International Conference on Computers and Their Application*, 2006, pp. 180–187.
- [25] J. Solomon, E. Vouga, M. Wardetzky, E. Grinspun, Flexible developable surfaces, *Comput. Graph. Forum* 31 (2012) 1567–1576.
- [26] R. Narain, T. Pfaff, J. F. O’Brien, Folding and crumpling adaptive sheets, *ACM Trans. Graph.* 32 (2013) 51:1–51:8.
- [27] I. Baran, D. Vlastic, E. Grinspun, J. Popovic, Semantic deformation transfer, *ACM Trans. Graph.* 28 (2009) 36:1–36:6.
- [28] J. Yang, L. Gao, Y. Lai, P. L. Rosin, S. Xia, Biharmonic deformation transfer with automatic key point selection, *Graphical Models* 98 (2018) 1–13.
- [29] F. Prada, M. Kazhdan, M. Chuang, A. Collet, H. Hoppe, Motion graphs for unstructured textured meshes, *ACM Trans. Graph.* 35 (2016) 108:1–108:14.
- [30] K. Wampler, Fast and reliable example-based mesh ik for stylized deformations, *ACM Trans. Graph.* 35 (2016) 235:1–235:12.
- [31] S. Martin, B. Thomaszewski, E. Grinspun, M. Gross, Example-based elastic materials, *ACM Trans. Graph.* 30 (2011) 72:1–72:8.
- [32] D. Xu, H. Zhang, Q. Wang, H. Bao, Poisson shape interpolation, in: *Symp. on Solid and Phys. Mod.*, 2005, pp. 267–274.
- [33] S. Kircher, M. Garland, Free-form motion processing, *ACM Trans. Graph.* 27 (2008) 12:1–12:13.
- [34] L. Gao, Y.-K. Lai, D. Liang, S.-Y. Chen, S. Xia, Efficient and flexible deformation representation for data-driven surface modeling, *ACM Trans. Graph.* 35 (2016) 158:1–158:17.
- [35] S. Wuhler, P. Bose, C. Shu, J. O’Rourke, A. Brunton, Morphing of triangular meshes in shape space, *International Journal of Shape Modeling* 16 (2010) 195–212.
- [36] C. von Tycowicz, C. Schulz, H.-P. Seidel, K. Hildebrandt, Real-time nonlinear shape interpolation, *ACM Trans. Graph.* 34 (2015) 34:1–34:10.
- [37] P. von Radziewsky, E. Eisemann, H.-P. Seidel, K. Hildebrandt, Optimized subspaces for deformation-based modeling and shape interpolation, *Computers & Graphics* 58 (2016) 128–138.
- [38] L. Younes, *Shapes and Diffeomorphisms*, Springer, 2010.
- [39] M. Kilian, N. J. Mitra, H. Pottmann, Geometric modeling in shape space, *ACM Trans. Graph.* 26 (2007) 64:1–64:8.
- [40] B. Heeren, M. Rumpf, M. Wardetzky, B. Wirth, Time-discrete geodesics in the space of shells, *Comput. Graph. Forum* 31 (2012) 1755–1764.
- [41] C. Brandt, C. von Tycowicz, K. Hildebrandt, Geometric flows of curves in shape space for processing motion of deformable objects, *Comput. Graph. Forum* 35 (2016) 295–305.

- [42] M. Ovsjanikov, M. Ben-Chen, J. Solomon, A. Butscher, L. J. Guibas, Functional maps: a flexible representation of maps between shapes, *ACM Trans. Graph.* 31 (2012) 30:1–30:11.
- [43] A. Kovnatsky, M. M. Bronstein, A. M. Bronstein, K. Glashoff, R. Kimmel, Coupled quasi-harmonic bases, *Comput. Graph. Forum* 32 (2013) 439–448.
- [44] R. M. Rustamov, M. Ovsjanikov, O. Azencot, M. Ben-Chen, F. Chazal, L. J. Guibas, Map-based exploration of intrinsic shape differences and variability, *ACM Trans. Graph.* 32 (2013) 72:1–72:12.
- [45] E. Corman, J. Solomon, M. Ben-Chen, L. J. Guibas, M. Ovsjanikov, Functional characterization of intrinsic and extrinsic geometry, *ACM Trans. Graph.* 36 (2017) 14:1–14:17.
- [46] E. Corman, M. Ovsjanikov, Functional characterization of deformation fields, *ACM Trans. Graph.* 38 (2019) 8:1–8:19.
- [47] A. J. Hanson, *Visualizing Quaternions*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2006.
- [48] E. Grinspun, A. N. Hirani, M. Desbrun, P. Schröder, Discrete shells, in: *Proc. of ACM SIGGRAPH/Eurographics Symposium on Computer animation*, 2003, pp. 62–67.
- [49] B. Heeren, M. Rumpf, P. Schröder, M. Wardetzky, B. Wirth, Splines in the space of shells, *Comput. Graph. Forum* 35 (2016) 111–120.
- [50] N. Krauth, M. Nieser, K. Polthier, Differential-based geometry and texture editing with brushes, *J. Math. Imaging Vis.* 48 (2014) 359–368.
- [51] J. Nocedal, S. J. Wright, *Numerical Optimization*, 2 ed., Springer, 2006.
- [52] M. R. Hestenes, *Optimization theory: the finite dimensional case*, Pure and applied mathematics, Wiley, 1975.
- [53] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed., Athena Scientific, Belmont, MA, 1999.
- [54] G. Pons-Moll, J. Romero, N. Mahmood, M. J. Black, Dyna: A model of dynamic human shape in motion, *ACM Trans. Graph.* 34 (2015) 120:1–120:14.
- [55] P. Hartman, L. Nirenberg, On spherical image maps whose Jacobians do not change sign, *Amer. J. Math.* 81 (1959) 901–920.
- [56] P. Hornung, Approximation of flat $W^{2,2}$ isometric immersions by smooth ones, *Arch. Ration. Mech. Anal.* 199 (2011) 1015–1067.
- [57] L. H. Dudte, E. Vouga, T. Tachi, L. Mahadevan, Programming curvature using origami tessellations, *Nature Materials* 15 (2016) 583–588.
- [58] G. Guennebaud, B. Jacob, Others, *Eigen v3*, <http://eigen.tuxfamily.org>, 2010.
- [59] Y. Chen, T. A. Davis, W. W. Hager, S. Rajamanickam, Algorithm 887: CHOLMOD, Supernodal Sparse Cholesky Factorization and Update/Downdate, *ACM Transactions on Mathematical Software* 35 (2008) 22:1–22:14.