Shape from Metric

ALBERT CHERN, TU Berlin
FELIX KNÖPPEL, TU Berlin
ULRICH PINKALL, TU Berlin
PETER SCHRÖDER, Caltech

We study the *isometric immersion* problem for orientable surface triangle meshes endowed with only a metric: given the combinatorics of the mesh together with edge lengths, approximate an isometric immersion into $\mathbb{R}^3$. To address this challenge we develop a discrete theory for surface immersions into $\mathbb{R}^3$. It precisely characterizes a discrete immersion, up to subdivision and small perturbations. In particular our discrete theory correctly represents the topology of the space of immersions, *i.e.*, the regular homotopy classes which represent its connected components. Our approach relies on unit quaternions to represent triangle orientations and to encode, in their parallel transport, the topology of the immersion. In unison with this theory we develop a computational apparatus based on a variational principle. Minimizing a non-linear Dirichlet energy optimally finds extrinsic geometry for the given intrinsic geometry and ensures low metric approximation error.

We demonstrate our algorithm with a number of applications from mathematical visualization and art directed isometric shape deformation, which mimics the behavior of thin materials with high membrane stiffness.

CCS Concepts: • **Mathematics of computing** → **Partial differential equations**; *Differential calculus*; *Algebraic topology*; • **Computing methodologies** → **Mesh models**; *Physical simulation*; • **Applied computing** → *Physics*;

Additional Key Words and Phrases: Differential geometry, geometry processing, isometric immersions, spin structures, quaternions.

1 INTRODUCTION

Suppose for some abstract surface we only know its metric. Can you map it into $\mathbb{R}^3$ in such a way that lengths of curves on the surface are preserved and small disks are realized in a 1-1 manner? We wish to solve this *isometric immersion* problem in the discrete setting when the surface is given in terms of a triangle mesh annotated with nothing more than lengths for every edge. A practical algorithm to numerically approximate such isometric immersions could have many applications. In Mathematical Visualization, we may encounter surfaces which are given purely in terms of their

Authors' addresses: Albert Chern, TU Berlin; Felix Knöppel, TU Berlin; Ulrich Pinkall, TU Berlin; Peter Schröder, Caltech.
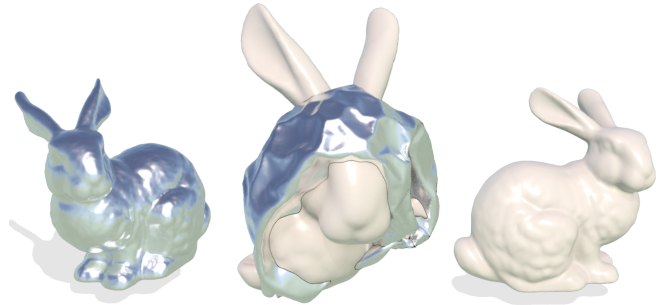
Fig. 1. Turning the bunny inside out while preserving its metric. See the video at 02:32.

metric without any explicit immersion (Fig. 18). In Geometry Processing we may be interested in isometric deformations which stay immersed (Fig. 1). Or, if the metric is manipulated, be it in shape space interpolation, editing, metric flows (Fig. 15), or others, we may need to find an immersion realizing the new metric. And in art directed form finding the characteristic "crumpling" behaviour of close-to-isometric surfaces can be of value (Figs. 12 & 13).

To pursue a program of numerically approximating isometric immersions we of course need to know that they exist in the first place. Fortunately this is the case both in the C^1 (once continuously differentiable) setting [Kuiper 1955; Nash 1954] and in the PL (piecewise linear) setting [Burago and Zalgaller 1960, 1995] (the PL setting also allows for additional piecewise linear subdivisions).

While these theorems ensure that isometric immersions always exist and are even deformable, the respective constructions also imply that there is an isometry arbitrarily near *any* (possibly scaled) immersion. Among this multitude of possibilities (Fig. 2) we want to
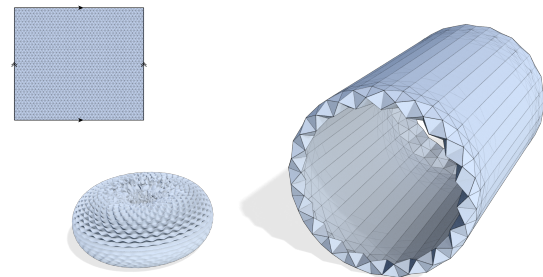

Fig. 2. Two isometric immersions of a flat torus (upper left, not to scale) using extensive "crumpling" (lower left) and nearly none (right). Both immersions of the torus are approximately isometric yet the "crumpled" torus has a very different appearance and apparent size.

choose one which meaningfully captures the *intrinsic* geometry in its *extrinsic* appearance.

Before we can develop an effective algorithm for discrete isometric immersions we must first study the nature of discrete immersions in general. What conditions must be true to ascertain that an assignment of 3-space positions to the vertices of a triangle mesh yields an immersion? What are the obstacles to being immersed and what kind of apparatus is needed to manage these obstructions? More globally, we need to characterize the topology (connected components) of the space of immersions to control which immersions can be deformed into one another while staying immersed throughout the deformation, *i.e.*, belong to the same *regular homotopy class* (Fig. 14).

In this paper we develop a theory for discrete (triangle mesh) immersions which recovers fundamental features of the smooth setting. In particular we can choose and control the regular homotopy class of an immersion we seek, and ensure that any deformation stays in the same class. Due to the finite resolution of a given mesh, this requires the representation of surface detail at finer scales and we develop simple local procedures to detect such detail and account for it appropriately. A critical ingredient in our construction is the representation of triangle orientations with the aid of unit quaternions. The *parallel transport* of these orientations provides the key to encoding the topology of the space of immersions in the discrete setup of a triangle mesh just as it does in the smooth setting.

Based on this theoretical development we design an algorithm for the discrete isometric immersion problem. It takes as input an orientable surface triangle mesh annotated with edge lengths only and outputs vertex positions in $\mathbb{R}^3$. Since the lengths already fix the Euclidean shapes of all triangles, only orientations in $\mathbb{R}^3$ need to be determined for each triangle. Vertex positions then arise as the solution of an ancillary Poisson problem. These orientations represented by unit quaternions are found as minimizers of a (non-linear) Dirichlet energy which encodes *both* the *integrability conditions* for the surface to "stitch together" *and* the topological requirements for being an immersion in a particular regular homotopy class. Triangle meshes can of course only model deformations up to the resolution of the mesh. Since we do not wish to continually adapt the triangulation we resolve properties such as being immersed only up to ϵ-perturbation and additional subdivision. For the same reason isometry is only realized approximately.

To achieve our goal of selecting a meaningful shape from the vast set of possible isometric immersions, an anisotropic norm is used in the Dirichlet energy to control the amount of bending regularization. Driving the associated parameter to zero we "steer towards" geometrically meaningful immersions (with submesh detail represented appropriately) which are close to isometric, *i.e.*, possess only moderate L^2-error.

1.1 Overview

After reviewing related work (Sec. 1.2) and fixing our notation (Sec. 1.3) we develop our tools in the main part of the paper. To characterize immersions (Sec. 2) we study triangle strips and their regular homotopies (Sec. 2.1). Being able to detect their homotopy class, one of only two, becomes the key to recognizing vertices

which are immersed (up to further subdivision and a small perturbation) and those which present essential obstacles to an immersion (Sec. 2.2). To handle regular homotopy information intrinsically, *i.e.*, without any mapping to $\mathbb{R}^3$ assumed, we encode it in the parallel transport of orientations given by unit quaternions (Sec. 3). To ensure that a given generic map $f \colon M \to \mathbb{R}^3$ is in the correct class it must be augmented with detail below the resolution of the mesh (Sec. 4). All considerations up to that point apply to immersions in general. For isometric immersions (Sec. 5) the triangle orientations are subject to integrability conditions (Sec. 5.1), which we use to formulate an energy (Sec. 5.2) to be minimized. We demonstrate the efficacy of our approach (Sec. 6) before concluding (Sec. 7). Complete implementation details can be found in Apps. A and D.

1.2 Related Work

Surface immersions into $\mathbb{R}^3$ were studied for a long time in mathematics before computer graphics researchers and their algorithms could contribute and we begin with a review of the history of immersions before discussing computer graphics algorithms.

1.2.1 History of Immersions. In the 19th century it was known that all topological types of surfaces can be realized in $\mathbb{R}^3$ with self-intersections *if* one allows for certain singularities called *pinch points* (inset). It was Werner Boy, who upended his PhD advisor's (Hilbert) belief, when he showed that

all such surfaces can in fact be immersed [Boy 1903]. Physical models of these are still in many historical collections [Fischer 2017, Ch. 6]. This then begged the question whether there are fundamentally different types of immersions, which cannot be deformed into one another while staying immersed. For the 2-sphere Stephen Smale [1959] showed that there is only one such *regular homotopy class*. Surprisingly, this implied that a sphere can be turned inside out as visualized in the landmark computer graphics film *Turning a Sphere Inside Out* [Max 1976]. Surfaces topologically more complicated than spheres possess multiple regular homotopy classes [Hass and Hughes 1985; Hirsch 1959; Pinkall 1985; Séquin 2011], which our algorithm can distinguish (Fig. 14).
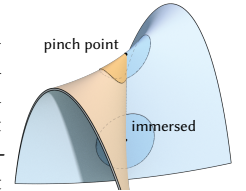
To discuss isometric immersions we must distinguish different smoothness classes to understand what is possible. For C^∞ (infinitely differentiable) isometric immersions very little is known [Berger 2010, Ch. VI.5] other than that no closed surface whose curvature is nowhere positive can be realized in $\mathbb{R}^3$ and the hyperbolic plane admits no isometric immersion. Bounded portions of it though do and our algorithm can find them (Fig. 16). For most applications C^∞ surfaces are too rigid.

What about isometric immersions of simplicial surfaces which are linear on each triangle? In this setting we see surfaces with nowhere positive curvature such as flat tori [Segerman 2016, p. 129] and the surface on the left of Fig. 17 [Barros et al. 2011] which are impossible for C^∞ surfaces. Generically though, they are as rigid as C^∞ surfaces [Bös et al. 2016; Gluck 1974].

The picture changes considerably for C^1 surfaces which possess isometric immersions near any given immersion that does not grow

the original metric [Kuiper 1955; Nash 1954]. Even though the Nash-Kuiper C^1-theory is based on an infinity of finer and finer surface "ripples," it was recently used to create beautiful visualizations of flat tori [Borrelli et al. 2012, 2013].

Isometric C^1 immersions enjoy the same flexibility as general immersions [Gromov 1986]. So in particular it is possible to turn the sphere inside out isometrically. Our algorithm, for the first time, allows us to visualize such isometric regular homotopies (Fig. 1).

Finally we have the class of PL-surfaces, which behave similar to C^1 surfaces. Given a triangle mesh and an immersion that does not grow edge lengths one can construct a nearby isometric immersion after a *finite* (though possibly very large) number of subdivision steps mimicking the Nash-Kuiper "ripples" construction [Burago and Zalgaller 1960, 1995].

1.2.2 Related Algorithms. Methods to realize triangle meshes in $\mathbb{R}^3$ have been considered with varying assumptions on the available data. With only edge lengths one can directly minimize the difference between embedded and target lengths [Boscaini et al. 2015; Isenburg et al. 2001] as a function of unknown vertex positions. Such a *length strain energy* corresponds to the membrane energy of elasticity which measures the difference between the given and desired metric [Chao et al. 2010; Panozzo et al. 2014; Sorkine and Alexa 2007]. A pure membrane energy for surfaces must be regularized, typically with a *bending energy* to avoid overly "crumpled" solutions. Additionally, a good initialization, say with a nearby surface, is needed to find a singularity free realization. Such membrane approaches can also be understood from the point of view of *sparse* multidimensional scaling (MDS). Sparse since only immediate neighbor distances are used [Boscaini et al. 2015]. Note that the more standard MDS approaches aim to find a realization where *extrinsic Euclidean* distances in some low dimensional $\mathbb{R}^n$ closely approximate the given *intrinsic geodesic* distances, a setting quite different from ours.

There is also a class of approaches using rigid triangles and solving for their orientations as functions of length *and* bending data [Baek et al. 2015; Botsch et al. 2006; Bouaziz et al. 2012; Kircher and Garland 2008; Wang et al. 2012; Winkler et al. 2010] (these publications are *representative* of many more in the literature). However, in the absence of bending data (our setting) they do not reliably produce immersions because they use representations insensitive to pinch singularities. Such singularities, where some vertex star cannot be made immersed even by subdivision and perturbation, are of a topological nature and dealing with them requires special care as explained in Sects. 2, 3 and 4.

1.3 Notation

We assume that an abstract Riemannian surface M is given to our algorithm in the form of a triangle mesh, *i.e.*, a simplicial complex $\{V, E, F\}$ consisting of vertices, edges, and facets (triangles). The surface must be oriented, may have a boundary, and can be of arbitrary topology. The main object of our algorithm are the triangles $i \in F$. Among the edges E we distinguish those which are interior as $\mathring{E} \subset E$. An oriented edge $ij \in \mathring{E}$ is named according to its incident triangles,
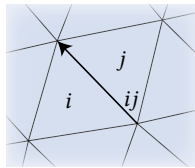
oriented positively with respect to the orientation of $i \in F$ (inset). For vertices V we similarly distinguish interior ones as $\mathring{V} \subset V$. Each triangle $i \in F$ is equipped with a tangent vector space $T_i M$, and each edge $e \in E$ is equipped with an abstract edge vector v_e which can be naturally identified with a vector in $T_i M$ for faces i incident to e.

The piecewise Euclidean metric of M is defined by edge lengths $\ell = (\ell_e > 0)_{e \in E}$ which observe the triangle inequality and uniquely determine a metric for $T_i M$ through $|v_e| = \ell_e$ and, up to rigid motion, a Euclidean triangle in $\mathbb{R}^3$ for each $i \in F$. Our main task is to find an orientation in $\mathbb{R}^3$ for each triangle. We parameterize these orientations with unit quaternions $\lambda_i \in \mathbb{S}^3 \subset \mathbb{H}$. We also call λ_i a *frame* and $\lambda = (\lambda_i)_{i \in F}$ a frame field.

Depending on context we will write quaternions as 4-vectors in terms of the quaternionic units, $q = s\mathbf{1} + u_1\mathbf{i} + u_2\mathbf{j} + u_3\mathbf{k}$ or use the "scalar plus vector" formalism $q = s + \mathbf{u}$ for $\mathbf{u} \in \mathbb{R}^3 \equiv \text{Im}\,\mathbb{H}$. Conjugation then becomes a sign flip on the vector component, $\bar{q} = s - \mathbf{u}$. Vectors embedded in $\mathbb{R}^3$ will be denoted in bold and we do not distinguish between $\mathbb{R}^3$ vectors and imaginary quaternions $\text{Im}\,\mathbb{H}$. The real inner product for quaternions $\langle p, q \rangle = \text{Re}(\bar{p}q)$ corresponds to the Euclidean inner product on $\mathbb{R}^4 \equiv \mathbb{H}$. Rotations of vectors $\mathbf{w} \in \mathbb{R}^3 \equiv \text{Im}\,\mathbb{H}$ are expressed in terms of unit quaternions q as

$$\tilde{\mathbf{w}} = \bar{q}\mathbf{w}q$$

where $q = \cos\left(\frac{\theta}{2}\right) - \sin\left(\frac{\theta}{2}\right)\mathbf{n} = \exp(-\frac{\theta}{2}\mathbf{n})$ describes the rotation by θ around the (unit) axis $\mathbf{n} \in \mathbb{S}^2 \subset \mathbb{R}^3$. Note that q and its antipode $-q$ describe the same rotation $\mathbf{w} \mapsto \tilde{\mathbf{w}}$.

2 DISCRETE IMMERSIONS

As a first step in our study of discrete immersions we must find tools to characterize and classify them.

Immersions of a surface M can be classified up to *regular homotopy*. Two immersions $f, \tilde{f} \colon M \to \mathbb{R}^3$ are regularly homotopic if one can be deformed into the other while remaining immersed throughout the deformation. Recall that an immersion has the local embedding property, *i.e.*, small disks are realized in a 1-1 manner (inset in Sec. 1.2.1). In the PL setting this leads to:

Definition 2.1 (Generic Map & Discrete Immersion). A map $f \colon M \to \mathbb{R}^3$ is *generic* if (1) each triangle in F is mapped linearly onto a non-degenerate triangle in $\mathbb{R}^3$ and (2) no edge in $\mathring{E}$ has f "fold back on itself," *i.e.*, the angle between incident triangle normals is in $(-\pi, \pi)$. (Up to an ϵ-perturbation all maps $f \colon M \to \mathbb{R}^3$ are generic.) The map is a *discrete immersion* if in addition (3) f restricted to the star of any interior vertex is an embedding.

Whether two surfaces are regularly homotopic can be decided by examining regular homotopies of closed strips on the surface. For such strips in $\mathbb{R}^3$ there are only two regular homotopy classes: *figure-8*, which is regularly homotopic to a planar *figure-8*, (Fig. 3, top right), and *figure-0*, which is regularly homotopic to a planar *figure-0* (Fig. 3, bottom right). For triangle meshes we get:

Theorem 2.2 (Regular Homotopy). *(a) Immersions $f, \tilde{f} \colon M \to \mathbb{R}^3$ are regularly homotopic (at least after subdivision) if and only if every closed triangle strip γ in M is figure-8 (resp. figure-0) under both f and $\tilde{f}$. (b) An immersed triangle strip in $\mathbb{R}^3$ is the boundary of an immersed disk if and only if it is figure-0.*
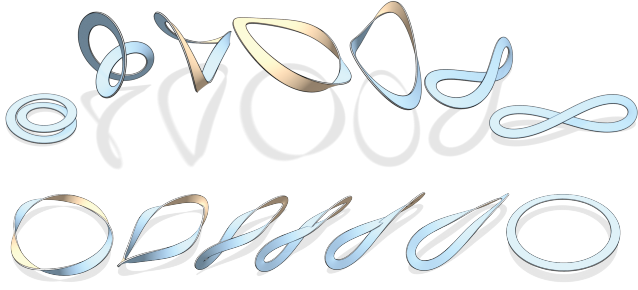
Fig. 3. Regular homotopies of strips in $\mathbb{R}^3$ in each of the two regular homotopy classes: *figure-8* (top), *figure-0* (bottom).

This theorem is well-known for smooth surfaces [Hass and Hughes 1985; Pinkall 1985] and Haefliger and Poénaru showed [1964] that regular homotopy theory for PL-surfaces looks exactly like the smooth theory.

How can we decide in practice whether a given closed triangle strip is *figure-8* or *-0*?

2.1 Closed Triangle Strips

Consider an immersion f of an oriented closed triangle strip γ described by an ordered, cyclic sequence of interior edges $\gamma = (e_0, \ldots, e_{m-1})$ (indexed modulo-m) which are shared by consecutive triangles along the *cycle*. For each such triangle i let $\mathbf{n}_i$ denote the unit normal of its embedding and $\mathbf{t}_i = df(T_i)$ the embedded unit tangent between midpoints of successive edges in γ incident on i (Fig. 4). Then for each $ij = e \in \gamma$ there exists a unique unit quaternion q_{ij} satisfying

$$\bar{q}_{ij}\mathbf{n}_{ij}q_{ij} = \mathbf{n}_j \qquad \bar{q}_{ij}\mathbf{t}_iq_{ij} = \mathbf{t}_j \qquad \mathrm{Re}(q_{ij}) > 0.$$

The first two equations determine q_{ij} up to sign, so we only have to check that the real part of q_{ij} does not vanish. For this note that q_{ij} is the product of two rotations. One around $\mathbf{n}_i$ by the *discrete geodesic curvature* angle κ_{ij}, and the other around the embedded edge $\mathbf{v}_{ij}$ by the *discrete bending* angle α_{ij} between consecutive normals (Fig. 4)

$$q_{ij} = \exp\left(-\frac{\kappa_{ij}}{2}\mathbf{n}_i\right)\exp\left(-\frac{\alpha_{ij}}{2}\frac{\mathbf{v}_{ij}}{|\mathbf{v}_{ij}|}\right). \tag{1}$$

Since $\mathbf{n}_i$ and $\mathbf{v}_{ij}$ are orthogonal we have

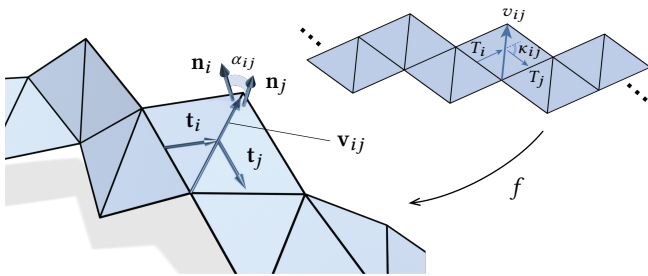$$\mathrm{Re}(q_{ij}) = \cos\left(\frac{\kappa_{ij}}{2}\right)\cos\left(\frac{\alpha_{ij}}{2}\right)$$

Fig. 4. An immersed discrete triangle strip.

which is indeed positive since $\alpha_{ij}, \kappa_{ij} \in (-\pi, \pi)$. Since going around the cycle gets us back to the identity we must have

$$q_{e_0}\cdots q_{e_{m-1}} = \pm 1. \tag{2}$$

Definition 2.3 (Figure 8/0). If Eq. (2) holds with the plus sign we say that the cycle γ is *figure-8*; if it holds with the minus sign it is *figure-0*.

Both cases occur (inset). Consider a cycle immersed in a plane with normal vector $\mathbf{i}$ and vanishing bending angles. In this case the discrete geodesic curvatures reduce to planar exterior angles, and we have

$$q_{e_0}\cdots q_{e_{m-1}} = \exp\left(-\frac{\mathbf{i}}{2}\sum_{ij\in\gamma}\kappa_{ij}\right). \tag{3}$$

Since the sum of the exterior angles of the center curve of the triangle strip gives a 2π multiple of its planar turning number we get $+1$ for even (*figure-8*) and -1 for odd (*figure-0*) turning numbers. Note that closed curves in the plane are classified into regular homotopy classes according to their winding number [Whitney 1937] while strips in $\mathbb{R}^3$ have only two (*figure-8/0*) regular homotopy classes [Kauffman and Banchoff 1977].

Given a generic map $f: M \to \mathbb{R}^3$ and a cycle γ in M, Def. 2.3 applies and we define the *extrinsic* $\mathbb{Z}_2$-valued predicate

$$\mathfrak{q}_f(\gamma) := \begin{cases} 0 & \text{if Eq. (2)} = -1 \text{ (figure-0)} \\ 1 & \text{if Eq. (2)} = +1 \text{ (figure-8)} \end{cases}. \tag{4}$$

2.2 Pinch Points and Almost Immersions

Given an immersion $f: M \to \mathbb{R}^3$ the vertex star of every interior vertex $p \in \mathring{V}$ is embedded. Letting γ_p denote the cycle formed by the triangles incident to p, the local embedding property implies $\mathfrak{q}_f(\gamma_p) = 0$.

Conversely, a generic map f may satisfy $\mathfrak{q}_f(\gamma_p) = 0$ for all $p \in \mathring{V}$ and *not* be an immersion. It can however be made into an immersion after a suitable subdivision and small perturbation. (The need to allow for such local modifications cannot be avoided since no triangle mesh will always have enough resolution for whatever shape it is required to form into.)

Consider a *figure-0* vertex star which is not embedded (Fig. 5, bottom left) and intersect it with a small sphere around the vertex. The intersection is a spherical polygon regularly homotopic to a small circle by assumption. If we perform this regular homotopy on the sphere [Kauffman and Banchoff 1977], while simultaneously shrinking the sphere, and finally cap off the small hole with a disk, the moving curve sweeps out a surface that can replace the non-immersed vertex star by an immersed disk (Fig. 5, bottom middle/right). A suitable modification of the above construction can ensure that the inserted immersed disk is still piecewise linear. This leads us to:

Definition 2.4 (Almost Immersion). We call a generic map $f: M \to \mathbb{R}^3$ an *almost immersion* if every vertex star is *figure-0*.

From a practical viewpoint, almost immersed meshes are very convenient. Testing whether a map f is really an immersion involves

extensive collision detection at each vertex. On the other hand, testing whether f is an almost immersion is straightforward (Eq. (2)). Even better: an almost immersion will stay almost immersed under a deformation so long as no triangle degenerates and all bending angles stay away from π. If we compute such an *almost regular homotopy* it becomes a proxy for a *nearby regular homotopy* living on a finer scale. We just do not bother to increase the resolution by subdivision.
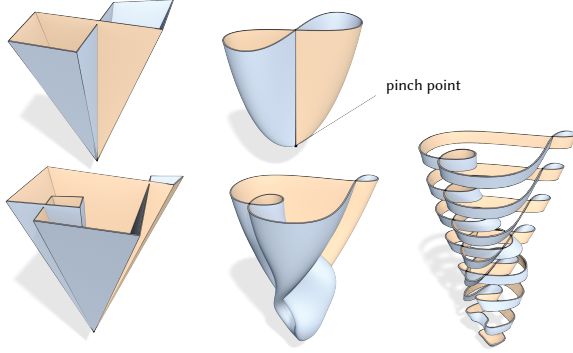

Fig. 5. After subdivision and a small perturbation, a vertex of a generic triangle mesh (left column) either stays as a pinch point (top middle) or can be made immersed (bottom middle; exploded view bottom right).

Conversely, a *figure-8* vertex star is a significant obstacle. We call such a vertex a *pinch point* (Fig. 5, top). By part (b) of Thm. 2.2 it is impossible to resolve such a singularity by a local modification near the vertex. Only a larger scale operation, canceling pinch points in pairs, can make it possible to arrive at an immersion (Sec. 4).

3 DISCRETE SPIN STRUCTURE

In the previous section we relied on a given generic map $f \colon M \to \mathbb{R}^3$ to define the *figure-8/0* property of immersed cycles. When no map is given and an immersion is to be found in the first place we need an apparatus which lets us talk about the *figure-8/0* property intrinsically. Furthermore, to find an immersion in a *prescribed* regular homotopy class we need to be able to specify the *figure-8/0* property of all cycles γ in M (Thm. 2.2). In this section we will show how to encode this information intrinsically.

We begin by mapping the tangent space $T_i M$ of each triangle $i \in F$ to the complex line via linear, orientation preserving isometries

$$\phi_i \colon T_i M \to \mathbb{C}. \tag{5}$$

For two adjacent triangles $i, j \in F$ the *parallel transport* of tangent vectors from $T_i M$ to $T_j M$ now amounts to multiplication with unitary complex numbers r_{ij} defined with respect to the common edge tangent vector

$$r_{ij} \phi_i(v_{ij}) = \phi_j(v_{ij}). \tag{6}$$

The field $r = (r_{ij})_{ij \in \mathring{E}}$ describes the usual *discrete Levi-Civita connection* on triangle meshes expressed with respect to the *basis field* $\phi = (\phi_i)_{i \in F}$ [Crane et al. 2010].

Using the notation of Sec. 2.1 we can also give an intrinsic definition of the discrete geodesic curvature κ_{ij} for the cycle γ

$$r_{ij} \exp(i\kappa_{ij}) \phi_i(T_i) = \phi_j(T_j),$$

with $T_i \in T_i M$ the unit tangent between midpoints of successive edges in γ. Cyclical substitution yields

$$\prod_{ij \in \gamma} r_{ij} = \exp\left(-i \sum_{ij \in \gamma} \kappa_{ij}\right).$$

Returning to the surface M as a whole, and in anticipation of mapping the abstract triangles into $\mathbb{R}^3$, we will now express all complex rotations as quaternionic rotations. Since a quaternionic rotation by a complex number acts on the $(\mathbf{j}, \mathbf{k})$-plane we map ϕ_i to this plane through post-multiplication with $\mathbf{j}$

$$\phi_i \mathbf{j} \colon T_i M \to \mathbb{C}\mathbf{j} = \mathrm{Span}\{\mathbf{j}, \mathbf{k}\} \subset \mathbb{R}^3.$$

In this way the tangent plane of every triangle is identified with the $(\mathbf{j}, \mathbf{k})$-plane in $\mathbb{R}^3$ and Eq. (6) reads as

$$\bar{\tau}_{ji} \phi_i(v_{ij}) \mathbf{j} \tau_{ji} = \phi_j(v_{ij}) \mathbf{j} \tag{7}$$

where $\bar{\tau}_{ji} = \tau_{ij} := \pm\sqrt{r_{ij}}$ is defined only up to sign. In analogy to r, the *intrinsic* field $\tau = (\tau_{ij})_{ij \in \mathring{E}}$ describes the *discrete spin connection* with respect to the basis field ϕ.

We now exploit the $\pm$-sign freedom to encode in the signs of τ the regular homotopy class we wish to prescribe. In analogy to the *extrinsic* Eq. (3) we use the sign of

$$\exp\left(\frac{i}{2} \sum_{ij \in \gamma} \kappa_{ij}\right) \prod_{ij \in \gamma} \tau_{ij} = \pm 1 \tag{8}$$

to encode the prescribed *figure-8/0* property in the *intrinsic* $\mathbb{Z}_2$-valued predicate

$$\mathfrak{q}_\tau(\gamma) := \begin{cases} 0 & \text{if Eq. (8)} = -1 \ (\textit{figure-0}) \\ 1 & \text{if Eq. (8)} = +1 \ (\textit{figure-8}) \end{cases} \tag{9}$$

(*cf.* Eq. (4)).

Given the $2^{|\mathring{E}|}$ possible choices for signing τ, this appears to give rise to a bewildering number of different $\mathfrak{q}_\tau$. However, many of these sign choices define the same $\mathfrak{q}_\tau$. This *gauge* degree of freedom is described by $\mathbb{Z}_2$-valued potentials defined on triangles. Consider $i \in F$ and flip the sign of τ on the interior edges incident to i to define $\tilde{\tau}$ then $\mathfrak{q}_\tau(\gamma) = \mathfrak{q}_{\tilde{\tau}}(\gamma)$ for every cycle γ. Hence if $\tau, \tilde{\tau}$ are equivalent, in the sense that they differ in their signs by an exact (dual) $\mathbb{Z}_2$-valued 1-form, we will have $\mathfrak{q}_\tau = \mathfrak{q}_{\tilde{\tau}}$. Conversely, the difference between τ and $\tilde{\tau}$ with $\mathfrak{q}_\tau = \mathfrak{q}_{\tilde{\tau}}$ will be the discrete derivative of a $\mathbb{Z}_2$-valued function. This is proved in the same way as the familiar fact that a real-valued 1-form, whose integral over all closed curves vanishes, must be exact. Letting $[\tau]$ denote the corresponding equivalence class we define $\mathfrak{q}_{[\tau]} := \mathfrak{q}_\tau$ for any representative $\tau \in [\tau]$.

Together with the desire to produce *immersions* (Thm. 2.2 (b)) this leads to:

Definition 3.1 (Discrete Spin Structure). A *discrete spin structure* $[\tau]$ is an equivalence class of discrete spin connections τ with $\mathfrak{q}_\tau(\gamma_p) = 0$ (*figure-0*) for all $p \in \mathring{V}$.

That the set of discrete spin structures of a given surface is not empty is proven in App. A, where it is shown that there are exactly 2^{β_1} distinct discrete spin structures by constructing them explicitly with the Tree/CoTree spanning tree algorithm [Dłotko 2012; Eppstein 2003; Erickson and Whittlesey 2005]. Here β_1 is the number of generators for the 1$^{\text{st}}$ homology, *i.e.*, $\beta_1 = 2g$ for a genus-g closed, orientable surface, and $\beta_1 = 2g + b - 1$ if there

are b boundary components. The fact that there are 2^{β_1} distinct discrete spin structures matches the smooth setting [Atiyah 1971; Lawson and Michelsohn 1990]. (We note that Def. 3.1 is compatible with [Hoffmann and Ye 2018] where discrete spin structures are used to study discrete Dirac operators.)

4 RIMMED IMMERSIONS

We now face a challenge. Suppose we are given (say, numerically) a generic map $f\colon M \to \mathbb{R}^3$ which may not (yet) be an almost immersion in the prescribed (by $[\tau]$) regular homotopy class. How should it be modified or deformed so as to be of the correct type?

As a first step we treat f as a surface with additional subscale detail consisting of *rims* (inset). A rim is an additional loop at an edge, which switches the *figure-8/0* property of any closed strip crossing it. Our main theorem (Thm. 4.2) then states that such rims can be added so that the *figure-8/0* property of all closed strips is as prescribed by $[\tau]$. In particular, all vertex stars will be *figure-0* and hence almost immersed (Def. 2.4 & Fig. 6). Which edges are rimmed will be encoded relative to τ by the signs of a unit quaternion field λ (Def. 4.1) arising from the polar decomposition of df.

To visualize the almost immersion a rimmed surface represents, we only need to tag the appropriate edges and draw them accordingly. But to fulfill on our goal of showing extrinsic shapes which reflect intrinsic geometry, we want to reduce, hopefully to zero, the number of rims through appropriate large scale deformation of f. This can be accomplished through a variational Ansatz for λ (Sec. 4.2).
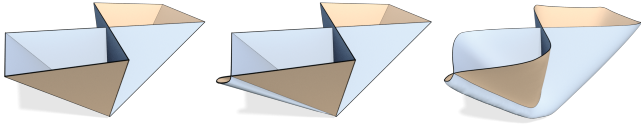

Fig. 6. Adding a rim ending at a *figure-8* vertex (left; middle) resolves the vertex into an almost immersed vertex (right).

4.1 Rim Bits

Given a generic map f we have, for all triangles $i \in F$, a linear map $df_i\colon T_iM \to \mathbb{R}^3$ of rank two which possesses a unique *polar decomposition*

$$df_i(X) = \omega_i(S_iX)$$

where $X \in T_iM$ is a tangent vector, $S_i\colon T_iM \to T_iM$ a positive, self-adjoint linear map and $\omega_i\colon T_iM \to \mathbb{R}^3$ an isometric map

$$\omega_i(X) = \bar{\lambda}_i \phi_i(X)\mathbf{j}\lambda_i \qquad (10)$$

using a suitable unit quaternion λ_i determined only up to sign. With this

$$df_i(X) = \bar{\lambda}_i \phi_i(S_iX)\mathbf{j}\lambda_i. \qquad (11)$$

Given the sign choices of τ, which encode the desired regular homotopy class, we now add rims to edges as described by the following predicate:

Definition 4.1 (Rim Bits). For all interior edges $ij \in \mathring{E}$ we define a $\mathbb{Z}_2$-valued predicate, the *rim bit*

$$\mathfrak{s}_{ij}^{\tau,\lambda} := \begin{cases} 0 & \text{if } \langle \lambda_j, \tau_{ij}\lambda_i \rangle > 0 \\ 1 & \text{if } \langle \lambda_j, \tau_{ij}\lambda_i \rangle < 0 \end{cases}$$

indicating whether $\tau_{ij}\lambda_i$ and λ_j are in the same ($\mathfrak{s}_{ij}^{\tau,\lambda} = 0$) hemisphere of $\mathbb{S}^3$. If not, an additional $360°$ turn, realized by a rim, is present. Note that for generic maps the case $\langle \lambda_j, \tau_{ij}\lambda_i \rangle = 0$ is excluded by Def. 2.1 (2).

The following theorem states that the rim selection according to $\mathfrak{s}^{\tau,\lambda}$ compensates for any discrepancy in *figure-8/0* properties induced by f versus the desired ones given by $\mathfrak{q}_{[\tau]}$:

THEOREM 4.2 (RIM BITS). *Given a spin structure $[\tau]$ on M, a generic map $f\colon M \to \mathbb{R}^3$, and an arbitrary choice of signs for λ in Eq. (11), the* figure-8/0 *property of any cycle $\gamma \in M$ under f and according to the desired spin structure $[\tau]$ are related as*

$$\mathfrak{q}_f(\gamma) = \mathfrak{q}_{[\tau]}(\gamma) + \sum_{ij \in \gamma} \mathfrak{s}_{ij}^{\tau,\lambda} \qquad (\text{mod } 2).$$

(For the proof see App. B.)

Def. 4.1 and Thm. 4.2 tell us that in order to produce a rimmed surface in the desired regular homotopy class, the rimming of edges ($\mathfrak{s}^{\tau,\lambda} = 1$) depends on the signs of both τ and λ. We can exploit this for practical purposes:

4.1.1 Distribution of Rims. Suppose we have made a particular choice of signs for τ to represent a desired regular homotopy class. Then we can use the sign freedom in λ to control the placement of rim edges. For example, we may choose the signs of λ so that $\mathfrak{s}^{\tau,\lambda} = 0$ along a triangle spanning tree. By Thm. 4.2 this eliminates all rims if f already belongs to the prescribed regular homotopy class.

4.1.2 Spin Structures from Immersions. Suppose f is already an immersion, then one can "read off" its spin structure as follows. Perform the polar decomposition of df to get λ with signs chosen arbitrarily. Then for each $ij \in \mathring{E}$ sign τ_{ij} so that $\mathfrak{s}_{ij}^{\tau,\lambda} = 0$. In that case Thm. 4.2 states $\mathfrak{q}_{[\tau]} = \mathfrak{q}_f$ and hence $[\tau]$ is the spin structure corresponding to the regular homotopy class of f.

4.2 Quaternion-Based Approaches

Every generic map f can be seen as an immersion of any regular homotopy class by decorating it with rims. However, those rimmed immersions that can convey an insightful extrinsic shape should only require rims ($\mathfrak{s}^{\tau,\lambda} = 1$) that are few and short (Fig. 7). Fortunately, since the rim information $\mathfrak{s}^{\tau,\lambda}$ is present in the quaternion field λ, we can reduce the total length of rims through a continuous variational problem in λ. To this end consider the *connection Dirichlet energy*

$$\mathcal{E}_1(\lambda) = \tfrac{1}{2} \sum_{ij \in \mathring{E}} w_{ij}|\lambda_j - \tau_{ij}\lambda_i|^2, \qquad (12)$$

for some given edge weights w_{ij}. (Such energies also occur in the context of *angular synchronization* though generally not on a 2-manifold but more general graph structures [Singer 2011; Singer

and Wu 2012].) Observing that

$$|\lambda_j - \tau_{ij}\lambda_i| \begin{cases} < \sqrt{2} & \text{if } s_{ij}^{\tau,\lambda} = 0 \\ > \sqrt{2} & \text{if } s_{ij}^{\tau,\lambda} = 1 \end{cases}$$

using Def. 4.1 and $|\tau_{ij}\lambda_i| = |\lambda_j| = 1$ we see that Eq. (12) is lower bounded by the total length of rimmed edges.
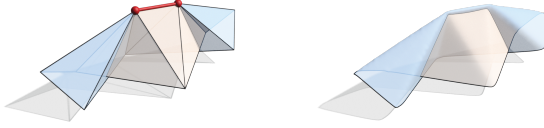

Fig. 7. Unlike scattered isolated pinch points, a pair of pinched vertices connected by a short rim (left) represents an immersion within a localized perturbation and subdivision (right), much like an almost immersed vertex (Fig. 5, bottom).

In Sec. 5 we introduce an energy for the isometric immersion problem which is of the form Eq. (12). Minimizing it will significantly reduce the number of rimmed edges and in practice often remove all of them.

5 APPROACH TO ISOMETRIC IMMERSIONS

We now return to our original goal: finding discrete close-to-isometric almost immersions when no initial map is provided. In this setting the orientations λ are the fundamental variables which, together with the spin structure $[\tau]$, encode the correct immersion. Once the orientations are found we can integrate the surface. For this the orientations must satisfy an *integrability condition* which ensures that neighboring triangles fit together. This integrability condition is encoded in an energy whose minimization yields optimal λ.

5.1 Integrability Conditions

Define a generic isometric map $\omega = (\omega_i)_{i \in F}$ with $\omega_i : T_i M \to \mathbb{R}^3$ given by

$$\omega_i(\cdot) := \bar{\lambda}_i \phi_i(\cdot) \mathbf{j} \lambda_i$$

as a function of the orientation field λ. For ω to realize a surface $f : M \to \mathbb{R}^3$ isometrically we must have $df = \omega$ and ω must satisfy a local integrability condition

$$\omega_i(v_{ij}) = -\omega_j(v_{ji}) \tag{13}$$

at all interior edges $ij \in \mathring{E}$. In that case ω defines an embedded edge vector field as a discrete $\mathbb{R}^3$-valued 1-form $(\mathbf{v}_e)_{e \in E}$ in the sense of *Discrete Exterior Calculus* [Desbrun et al. 2008] with $\mathbf{v}_e := \omega_i(v_e)$ for each edge e incident to face i. If Eq. (13) holds on a simply-connected M, we can construct a unique $f : V \to \mathbb{R}^3$ (up to translation) so that

$$df(v_e) = \mathbf{v}_e$$

If M is not simply-connected, we need $\mathbf{e}$ to be exact in addition to satisfying Eq. (13) (App. D.2).

To turn Eq. (13) into a condition on λ substitute Eqs. (10) and (7) while using the shorthand

$$z_{ij} := \ell_{ij}^{-1} \phi_i(v_{ij}) \tag{14}$$

to find the equivalent statement

$$\overline{(\lambda_j(\overline{\tau_{ij}\lambda_i}))}(z_{ji}\mathbf{j})(\lambda_j(\overline{\tau_{ij}\lambda_i})) = z_{ji}\mathbf{j}.$$

Thus $\lambda_j(\overline{\tau_{ij}\lambda_i})$ must be a rotation around the $z_{ji}\mathbf{j}$ axis. It has an explicit form in

$$\lambda_j(\overline{\tau_{ij}\lambda_i}) = (-1)^{s_{ij}^{\tau,\lambda}} \exp\left(\frac{\alpha_{ij}}{2} z_{ji}\mathbf{j}\right) \tag{15}$$

with $\alpha_{ij} = \alpha_{ji} \in (-\pi, \pi)$ the bending angle between incident triangle normals as before (App. C.1). Since the bending angles are not constrained we express this condition by demanding that

$$\lambda_j(\overline{\tau_{ij}\lambda_i}) \in \text{Span}\{\mathbf{1}, z_{ji}\mathbf{j}\} \tag{16}$$

where quaternions are understood as vectors in $\mathbb{R}^4$. It is not difficult to see (App. C.2) that this condition is equivalent to

$$\lambda_j - \tau_{ij}\lambda_i \in \text{Span}\{z_{ji}\mathbf{j}\mu_{ji}\} \tag{17}$$

with $\mu_{ji} \in \mathbb{S}^3$ the *midframe* between $\tau_{ij}\lambda_i$ and λ_j

$$\mu_{ji} := \frac{\tau_{ij}\lambda_i + \lambda_j}{|\tau_{ij}\lambda_i + \lambda_j|}. \tag{18}$$

Geometrically, μ_{ji} is the average of the frames on i and j represented as a quaternion in the basis of j, a quantity which satisfies $\tau_{ij}\mu_{ij} = \mu_{ji}$ since $\tau_{ji} = \bar{\tau}_{ij}$ and $|\tau_{ij}| = 1$.

5.2 Formulation of an Energy

In this section we formulate an energy based on the integrability of λ, using Eq. (17), which states that the *parallel transported finite difference* $\lambda_j - \tau_{ij}\lambda_i$, as a vector in $\mathbb{H} \equiv \mathbb{R}^4$, should be small in the orthogonal complement of $\text{Span}\{z_{ji}\mathbf{j}\mu_{ji}\}$.

To test this, a natural choice for an orthonormal basis of $\mathbb{H}$ at $ij \in \mathring{E}$ for given λ_i and λ_j, is

$$\{\mu_{ji}, z_{ji}\mathbf{j}\mu_{ji}, \mathbf{i}\mu_{ji}, \mathbf{i}z_{ji}\mathbf{j}\mu_{ji}\}. \tag{19}$$

Using this basis, and four nonnegative dimensionless global parameters $(\epsilon_0, \epsilon_1, \epsilon_2, \epsilon_3) =: \boldsymbol{\epsilon}$, we define an *anisotropic* norm $|\cdot|_{ij,\boldsymbol{\epsilon}}$ for $q \in \mathbb{H}$

$$|q|_{ij,\boldsymbol{\epsilon}}^2 = \epsilon_0 \langle \mu_{ji}, q \rangle^2 + \epsilon_1 \langle z_{ji}\mathbf{j}\mu_{ji}, q \rangle^2 + \epsilon_2 \langle \mathbf{i}\mu_{ji}, q \rangle^2 + \epsilon_3 \langle \mathbf{i}z_{ji}\mathbf{j}\mu_{ji}, q \rangle^2,$$

and use it to define our energy for λ with $|\lambda_i| = 1$

$$\mathcal{E}_{\boldsymbol{\epsilon}}(\lambda) := \frac{1}{2} \sum_{ij \in \mathring{E}} w_{ij} |\lambda_j - \tau_{ij}\lambda_i|_{ij,\boldsymbol{\epsilon}}^2 \tag{20}$$

where w_{ij} are weights for dual edges accounting for the metric. Note that Eq. (20) is identical to Eq. (12) except for weighting the four components of $\lambda_j - \tau_{ij}\lambda_i$ individually. Hence, minimization of Eq. (20) reduces the number of rim edges while enforcing integrability of λ.

5.2.1 Geometric Interpretation. Each component of the norm $|\cdot|_{ij,\boldsymbol{\epsilon}}$ in Eq. (20) measures a geometrically meaningful mode of the local strain arising from the difference of the frames $\lambda_j - \tau_{ij}\lambda_i$. To see this, let $\lambda_j(\overline{\tau_{ij}\lambda_i}) = \exp\left(\frac{\theta}{2}\mathbf{v}\right)$ for a suitably chosen unit rotation axis $\mathbf{v} \in \text{Im }\mathbb{H}$ and rotation angle θ. In these variables an argument similar to the one used in App. C.2 gives $\lambda_j - \tau_{ij}\lambda_i \in \text{Span}\{\mathbf{v}\mu_{ji}\}$. Therefore $|\cdot|_{ij,\boldsymbol{\epsilon}}$ measures the contributions of $\lambda_j(\overline{\tau_{ij}\lambda_i})$ along the respective axes: $z_{ji}\mathbf{j}$ (shared edge vector), $\mathbf{i}$ (normal vector), and $\mathbf{i}z_{ji}\mathbf{j}$ (dual of shared edge vector). Specifically the symmetric

$$\langle z_{ji}\mathbf{j}\mu_{ji}, \lambda_j - \tau_{ij}\lambda_i \rangle =: \zeta_{ij} = \zeta_{ji}.$$

measures bending. Rotation around the normal is measured by the anti-symmetric

$$\langle \mathbf{i}\mu_{ji}, \lambda_j - \tau_{ij}\lambda_i \rangle =: \eta_{ij} = -\eta_{ji},$$

with zero indicating that parallel transport of tangent vectors is correct, while the symmetric

$$\langle \mathbf{i}z_{ji}\mathbf{j}\mu_{ji}, \lambda_j - \tau_{ij}\lambda_i \rangle =: \sigma_{ij} = \sigma_{ji}$$

captures *geodesic torsion*, *i.e.*, rotation around the dual edge.

The first term, associated with ϵ_0, corresponding to the "length axis" $\mathbf{1}$, is missing here since $(\lambda_j - \tau_{ij}\lambda_i)\perp\mu_{ji}$. In our numerical setting though we will include the length axis because we are using a fractional step scheme for the energy minimization.

Defining the L^2 energy of a quantity $\zeta = (\zeta_{ij})_{ij\in\mathring{E}}$ via $\|\zeta\|^2 = \sum_{ij\in\mathring{E}} w_{ij}|\zeta_{ij}|^2$, we can rewrite Eq. (20) as an elastic energy

$$\mathcal{E}_{\epsilon}(\lambda) = \frac{\epsilon_1}{2}\|\zeta\|^2 + \frac{\epsilon_2}{2}\|\eta\|^2 + \frac{\epsilon_3}{2}\|\sigma\|^2.$$

With $\epsilon_{1\ldots 3}$ we can now control specific geometric aspects of the minimization. For example, when in-plane rotation is much more heavily penalized than bending and torsion, $\epsilon_2 \gg \epsilon_3 = \epsilon_1$, a minimizer of $\mathcal{E}_{\epsilon}$ will describe orientations of triangles that tend to induce a parallel transport of tangent vectors corresponding to the Levi-Civita connection. This alone implies that the resulting orientation field is close to locally integrable *up to* some constant rotation about the normals, due to the invariance of $\mathcal{E}_{\epsilon}$ under $\lambda \mapsto \exp(\mathbf{i}c)\lambda$ for c a global constant.

We use a strategy of scheduling the parameter ϵ in stages. Early on we let $\epsilon_2 \gg \epsilon_3 = \epsilon_1$ to fix the normal vectors and relative orientations. Later we fix the remaining global normal rotation by breaking the symmetry $\lambda \mapsto \exp(\mathbf{i}c)\lambda$ via $\epsilon_2 \gg \epsilon_3 \gg \epsilon_1$. Finally we let $\epsilon_1 = 0$ to remove the bending regularization entirely.

5.3 Minimization

To find an optimal λ, we minimize Eq. (20) iteratively with a sequence of stiffness parameters ϵ.

Algorithm 1 Minimization

Input: $\epsilon^{(1)}, \epsilon^{(2)}, \epsilon^{(3)}, \ldots$ with $\epsilon_1^{(n)} \to 0$; $\lambda^{(0)}$;
1: **for** $n = 1, 2, 3, \ldots$ **do**
2: $\quad \lambda^{(n)} \leftarrow \arg\min_{\lambda} \mathcal{E}_{\epsilon^{(n)}}(\lambda)$ with initial guess $\lambda^{(n-1)}$;
3: **end for**

Generally we let $\lambda^{(0)}$ be random and schedule $\epsilon^{(n)}$ in four iterations according to the considerations of Sec. 5.2.1

$$\epsilon^{(1)} = (1, 1, 1, 1), \qquad \epsilon^{(2)} = (1, 0.1, 1, 0.1),$$
$$\epsilon^{(3)} = (1, 0.01, 1, 0.1), \quad \epsilon^{(4)} = (1, 0, 1, 0.1). \tag{21}$$

A detailed description of our numerical implementation of this algorithm can be found in App. D.

5.4 Computing Vertex Positions

Having found a minimizer λ representing ω which is nearly integrable (unless $\mathcal{E}_{\epsilon} = 0$) we construct the averaged embedded edge vector field $(\hat{\mathbf{v}}_e)_{e\in E}$

$$\hat{\mathbf{v}}_{ij} := \tfrac{1}{2}(\omega_i(v_{ij}) - \omega_j(v_{ji}))$$

for $ij \in \mathring{E}$ and $\hat{\mathbf{v}}_e := \omega_i(v_e)$ for each boundary edge e incident to face i. From $\hat{\mathbf{v}}$ we compute the vertex position f by solving $df(v_e) = \hat{\mathbf{v}}_e, e \in E$, in the least-squares sense, which amounts to a Poisson problem (App. D.1).

Before λ is optimal, f is likely not yet an almost immersion and we need to treat f as a rimmed immersion. Following Sec. 4.1, we apply polar decomposition to df to obtain λ^* (Eq. (11)) with signs chosen so that $\langle \lambda_i^*, \lambda_i \rangle > 0$ for each $i \in F$ and visualize the rim bits $\mathfrak{s}^{\tau, \lambda^*}$ of λ^*.

5.4.1 Feedback from Vertex Positions. In the subiterations for minimizing $\mathcal{E}_{\epsilon}$ in Alg. 1 we can optionally feed back λ^* to λ, *i.e.*, replace λ by λ^* or a convex combination of both (App. D.2), to incorporate extrinsic constraints on vertex positions, for example, point constraints. It also improves the optimization of $\mathcal{E}_{\epsilon}$ generally (Fig. 8) and is useful when dealing with non-simply-connected surfaces.

5.5 Summary

Putting all the pieces together the entire algorithm, with all "subroutine" references, is given by:

Algorithm 2 Algorithm overview

Input: Triangle mesh $\{V, E, F\}$; edge lengths $(\ell_e)_{e\in E}$; $\quad\triangleright$ Sec. 1.3
1: Compute intrinsic quantities (App. D), edge weights $(w_{ij})_{ij\in\mathring{E}}$;
2: Build Levi-Civita connection $(\phi_i)_{i\in F}$ and $(r_{ij})_{ij\in\mathring{E}}$; $\triangleright$ Eq. (5, 6)
3: Spin connection $\tau_{ij} \leftarrow \pm\sqrt{r_{ij}}$ with arbitrary sign; $\quad\triangleright$ Eq. (7)
4: Fix the signs of τ so that $\mathfrak{q}_{\tau}$ (Eq. (9)) satisfies Def. 3.1 ;
$\quad\quad\triangleright$ Use Alg. 3, or Sec. 4.1.2 if an initial immersion f is available.
5: Optional: switch regular homotopy class;
$\quad\quad\quad\quad\quad\triangleright$ Flip signs of τ along generators found by Alg. 4.
6: Initialize λ; $\quad\quad\quad\quad\quad\triangleright$ randomly or by Eq. (11) if f is given.
7: Evolve λ by Alg. 1;
$\quad\quad\triangleright$ Use either Alg. 5, Alg. 6, or Alg. 8 which invokes Sec. 5.4.1.
8: $f \leftarrow$ compute vertex positions; $\quad\quad\triangleright$ Sec. 5.4, Alg. 7
9: $\lambda^* \leftarrow$ polar-decompose df (Eq. (11)) with $\langle \lambda_i^*, \lambda_i \rangle > 0$; $\triangleright$ Sec. 5.4
Output: Vertex positions f and rim bits $\mathfrak{s}^{\tau, \lambda^*}$.

---

6 RESULTS AND APPLICATIONS

We have implemented Alg. 2 (Sec. 5.5) in Houdini 16 by SideFX software, using SciPy [Jones et al. 2001] for the numerical solvers. Complete source code is included. Details of the implementation can be found in App. D. Here we report on experiments we performed with our implementation. All surfaces we produce are almost immersions possibly rimmed in the sense of Sec. 4. The energy minimization generally removes all rims, though if some are left they are typically no longer than one or two edges (Fig. 7).

6.0.1 Performance. For the minimization of $\mathcal{E}_{\epsilon}$ in Alg. 2, our experiments used gradient descent (Alg. 5), except for Fig. 16, which used a (quasi-)Newton's method (Alg. 6). (While Newton's method requires fewer iterations, each iteration is substantially more expensive, often making Newton's method slower overall.) Gradient descent typically takes 10–40 subiterations, each being dominated by the solution of a sparse linear system of size $|F|$. We used the

conjugate gradient method without preconditioning, taking about 5 seconds for a $|F| \approx 29$k model on a MacBook Pro with a 2.5 GHz Intel Core i7 Processor. Fig. 8 shows a typical isometry error plot as a function of iteration count. In particular it indicates that the metric error decreases as the minimizer decreases the energy (Sec. 5.2.1).
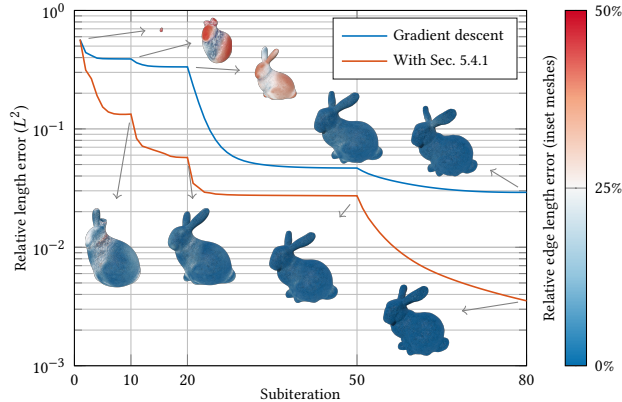

Fig. 8. L^2 relative metric error and its distribution on the surface at each stage of Alg. 2 with and without the additional treatment of Sec. 5.4.1. ϵ is scheduled according to Eq. (21) (four labeled subiteration intervals). Algorithm input are Stanford Bunny edge lengths and a random $\lambda^{(0)}$.

Before continuing with more examples we first compare our method to other possible approaches motivated by the related work in algorithms (Sec. 1.2.2).

6.1 Comparison of Methods

How important is the use of quaternions to avoid pinch singularities in practice? To elucidate this question we consider several alternative approaches.
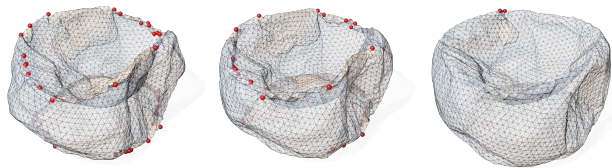

Fig. 9. An arbitrarily deformed sphere (left) and nearby surfaces isometric to the round sphere (middle & right). A method based on minimizing membrane energy (middle) produces acceptable metric fidelity (1% L^2 error), but offers no tendency towards immersions (38 scattered pinch points shown in red). Our method (right) achieves metric fidelity (0.6% L^2 error) and also approaches an immersion (2 pinch points connected by a short rim).

One set of approaches does not use any orientations at all and instead produces close-to-isometric surface realizations by minimizing the metric error, *i.e.*, minimizing an elastic energy, generally regularized with a decreasing bending energy [Boscaini et al. 2015; Isenburg et al. 2001] (and similarly [Bouaziz et al. 2012; Chao et al. 2010; Fröhlich and Botsch 2011; Panozzo et al. 2014]). In our study we initialized the algorithm with random data since we cannot in general rely on some known nearby immersion.

While these methods produce close-to-isometric surface realizations, most of the realizations have pinch points (Fig. 9, middle) even after fine tuning bending regularization (Fig. 10, right). Contrast this with the remaining pinch points from our method, if any, which typically come in pairs connected by short rims (Fig. 9, right) that can be removed easily (Fig. 7).
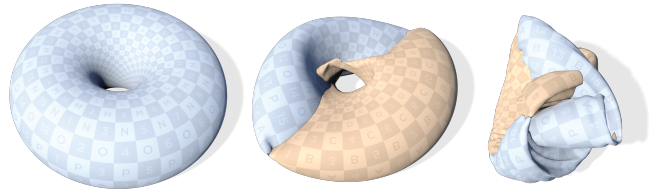

Fig. 10. Other algorithms often get stuck in local minima with pinch points. Using the metric of the torus on the left, the middle shows a surface computed with an SO(3)-based approach inspired by [Wang et al. 2012] (0.9% L^2 error), while the right shows the result of minimizing the length strain energy [Boscaini et al. 2015; Isenburg et al. 2001] (2% L^2 error).

Another class of potentially applicable techniques uses rigid triangles with orientations in the form of 3×3 rotation matrices (SO(3)) [Botsch et al. 2006; Wang et al. 2012]. To understand how much the "blindness" of rotation matrices to pinch singularities matters in practice, we replace the quaternions representing triangle orientations by SO(3) matrices in our implementation. This variant closely resembles [Wang et al. 2012], who report that in the presence of (nearly) compatible bending data (satisfying the discrete Gauss-Codazzi equations [Lipman et al. 2005]) a surface can be found reliably with a single linear solve. Since we assume no bending data, we minimize their non-linear energy with gradient descent. This produces isometric surface realizations with less than 1% L^2 error in the edge lengths. However, control over the regular homotopy class and any energetic incentive towards immersion are lost (Fig. 10, middle).
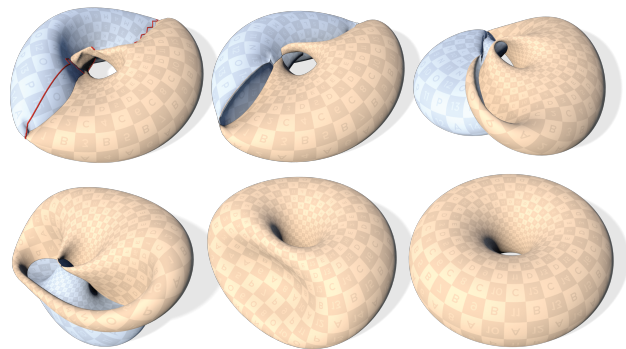

Fig. 11. Our algorithm treats the pinched surface in Fig. 10 (middle) as a rimmed immersion (left), and finds close-to-isometric almost immersion by energetically resolving the rims (left to right).

For both classes of algorithms the results are close to isometric and the orientations almost integrable, while pinch singularities

Table 1. Statistics of relative L^2 length error, mesh size, and number of rims.

	Length error	$(|V|, |E|, |F|)$	# rims
Fig. 12	1.2%	(4002, 12000, 8000)	0
Fig. 13	2.13%	(9120, 27120, 18000)	0
Fig. 14	1.2%, 6.5%, 1.65%, 3.6%	(3000, 9000, 6000)	0, 0, 0, 14
Fig. 15	5.24%	(3087, 9261, 6174)	2
Fig. 16	7%	(10000, 29219, 19220)	0
Fig. 17	1.15%	(4147, 12447, 8298)	1
Fig. 18	0.7%	(5807, 17417, 11608)	8
Fig. 19	2.4%	(5807, 17417, 11608)	4

remain (Fig. 10, middle & right). Our quaternion based algorithm though does not treat these as local minima and instead regards them as rimmed surfaces with high energy (Sec. 4.2). For example, starting with the pinched surfaces in Fig. 10 (middle) our algorithm evolves it into a close-to-isometric almost immersions with a prescribed regular homotopy class (Fig. 11)

6.2 Surfaces with Constant Curvature

We now demonstrate our method by finding close-to-isometric almost immersions for the simplest surfaces, those with constant curvature.

Fig. 12. An isometrically deflated soccer ball following art direction. See the video at 00:08.

6.2.1 Round Sphere.
The only orientable closed surface with constant positive curvature $K = 1$ is the round sphere. If we run our algorithm with initially high but then decreasing regularization, we always find the standard immersion. However, we can also start with a user provided initial vertex positions, run our algorithm with no regularization ($\epsilon_1 = 0$) to arrive at a nearby close-to-isometric almost immersion (Fig. 12). In the same way we can produce art-directed close-to-isometric almost immersions of any surface (Fig. 13).

Fig. 13. A close-to-isometric immersion (right) of a flat cylinder (left) near a user-provided shape (middle). The spin structure was taken from the cylinder on the left. See the video at 02:02.

6.2.2 Flat Cylinders.
Point constraints can be incorporated in our algorithm by including the step described in Sec. 5.4.1 and App. D.2 in the minimization loop, implementing the point constraints in the Poisson problem for computing vertex positions. For example, in Fig. 13 the rigid top and bottom of the soda can provide boundary constraints for the (intrinsically flat) cylindrical piece.

6.2.3 Flat Tori.
A metric with constant curvature on a torus is necessarily flat. For a given flat torus our algorithm can find close-to-isometric almost immersions in each of the four regular homotopy classes (Fig. 14), which we deliberately select with the Tree/Co-Tree algorithm (App. A).
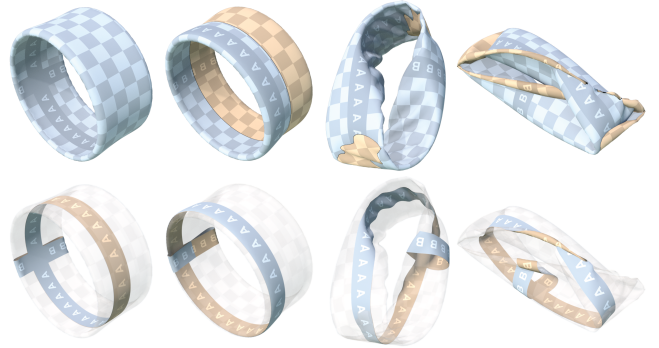

Fig. 14. Four close-to-isometric almost immersions of a flat torus of differing regular homotopy type. These cannot be deformed into one another while staying immersed.

6.2.4 Visualizing Metric Flows.
Given any smooth surface with a metric, it is always possible to scale the metric conformally to arrive at a metric with constant curvature. The so-called *Ricci flow*

$$\dot{g} = -Kg \tag{22}$$

with K denoting the Gaussian curvature, achieves this by flowing the original metric g continuously towards the uniformization metric [Zhang et al. 2015]. Since the metric is manipulated directly there are no natural immersions associated with the flow except in the case of a topological sphere [Crane et al. 2013; Kazhdan et al. 2012]. Our algorithm allows us to visualize the entire flow through the evolution of corresponding immersions with the final surface in Fig. 15 a flat torus.
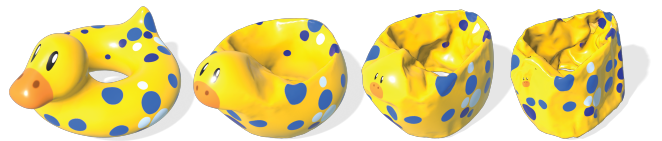

Fig. 15. Ricci flow on a torus visualized via close-to-isometric almost immersions. See the video at 01:01.

6.2.5 Hyperbolic Disk.
We now turn to surfaces with constant negative curvature. Our algorithm can find close-to-isometric almost immersions of large portions of the hyperbolic plane (Fig. 16).
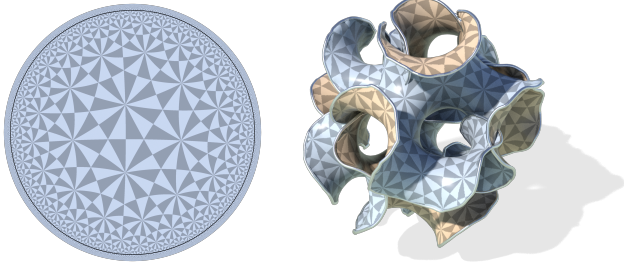
Fig. 16. A close-to-isometric immersion of a large disk (black line) in the hyperbolic plane. See the video at 00:35.

6.2.6 Genus Two Hyperbolic Surfaces. On a compact surface of genus 2, Ricci flow gives us a conformally equivalent metric with curvature $K = -1$. Here a polyhedral surface of genus 2 with all vertices having negative discrete curvature [Barros et al. 2011] (Fig. 17, left) is subdivided piecewise linearly, given a conformally equivalent metric with constant Gaussian curvature $K = -1$, and realized close to isometric in the same regular homotopy class (right).
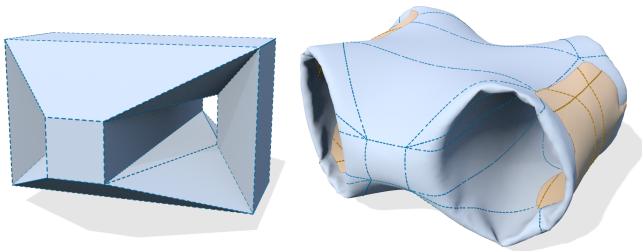

Fig. 17. Left: a polyhedron with negative curvature at all vertices; right: the subdivided polyhedron with constant negative curvature immersed close to isometric.

Quite often the overall geometry of the final uniformized metric is already clearly visible when a small amount of regularization is still present in the energy minimization (Fig. 18).
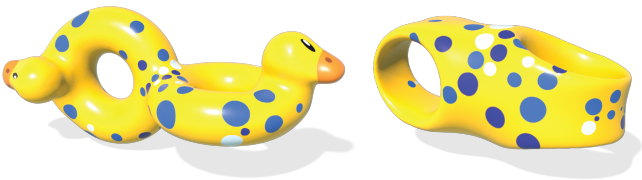

Fig. 18. The conformally equivalent metric with constant negative curvature on the Double Duck, realized with a small amount of bending regularization. The original surface served as initial data. See the video at 01:28.

While Fig. 18 used the original Double Duck as its initial surface, using random initial data results in a different, though still regularly homotopic, close-to-isometric almost immersion (Fig. 19).
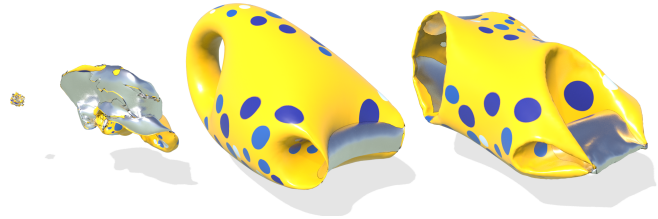

Fig. 19. Progression of our energy minimization with the same hyperbolic metric as in Fig. 18 starting with random initial data (left) and ending with an L^2-error for the metric of 2.13% (right). See the video at 01:44.

6.3 Bunny Eversion

So far the regularization term (ϵ_1) in the energy drives all bending angles zero. If, for some reason, we are given specific target bending angles $\hat{\alpha}_{ij}$, we can incorporate them by modifying the spin connection, replacing τ_{ij} with

$$\tilde{\tau}_{ij} := \exp\left(\tfrac{\hat{\alpha}_{ij}}{2} z_{ji}\mathbf{j}\right)\tau_{ij},$$

which amounts to *re-gauging* the measurement of bending in Eq. (15)

$$\lambda_j(\overline{\tilde{\tau}_{ij}\lambda_i}) = (-1)^{s_{ij}^{\tau,\lambda}} \exp\left(\tfrac{\alpha_{ij}-\hat{\alpha}_{ij}}{2} z_{ji}\mathbf{j}\right).$$

If metric and bending angles are taking from an existing surface, for example a bunny, our algorithm will reliably *reconstruct* this surface. Take now as initial surface for the energy minimization a reflected bunny, *i.e.*, a bunny turned inside out, and put a high penalty on deviation from the original metric. Using a small time step we then witness a *close-to-isometric eversion* of the bunny (Figs. 1 & 20).
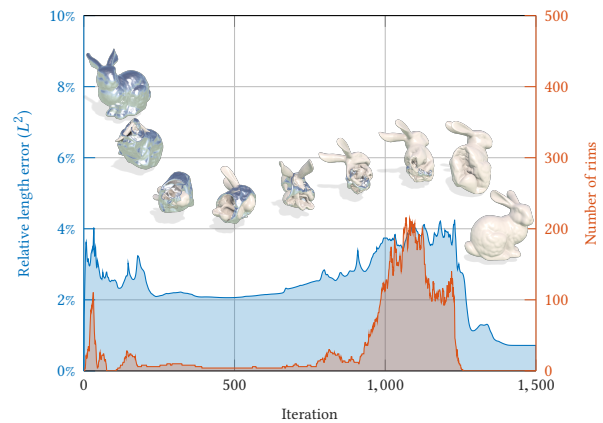
Fig. 20. Statistics of the length error and rim number during the process of turning the Stanford bunny inside out. Model size $(|V|, |E|, |F|) = (14290, 42684, 28576)$.

Note. While we understand how to interpret each rimmed surface from the time discretized sequence as a shorthand for an actual immersion (Sec. 4), it is less clear how they connect in continuous time. In principle there are underlying regular homotopies that interpolate these almost immersions. However, a simple and systematic visualization of these rim interactions, transforming one into

another, between frames as yet needs to be addressed and is left to future work.

While this simple approach to a close-to-isometric immersion works well for the Bunny, we emphasize that it is not a general isometric eversion procedure since we rely on energy descent for a discrete bending energy [Bridson et al. 2003; Grinspun et al. 2003], while keeping the isometry part of the energy small. For a round sphere, for example, it is known though that somewhere along every eversion the Willmore energy is larger than 16π [Li and Yau 1982; Max and Banchoff 1981]. Using this one can show that an eversion cannot be produced by simple bending energy descent.

7 CONCLUSION

In this paper we presented a discrete theory for immersions of triangle meshes into $\mathbb{R}^3$ and applied it to the isometric immersion problem. Quaternions and their transport are the key to encoding the regular homotopy class information and to ensure that deformations stay immersed. We introduce the notion of a rimmed surface which is guaranteed to be in the correct regular homotopy class. Our variational approach based on a connection Dirichlet energy allows us to resolve these rims (unless the mesh is much too coarse) through global deformations while simultaneously finding optimal shapes representing the intrinsic geometry.

We are hopeful that this new theory characterizing immersions in the form of rimmed surfaces will find application in many areas of Geometry Processing. For example, our quaternionic surface descriptions are close relatives of (conformal) spin transformations [Crane et al. 2011], suggesting that our method of close-to-isometric immersions could be extended to conformally equivalent immersions. The physical interpretation of our energy suggests that our approach can form the basis for physical modeling of cloth, thin-shells, and related materials, as well as physical-metaphor-based geometric editing. Our method also has potential in applications that use shape space representations. While compatible metric *and* bending data are required for exact reconstruction of surfaces we find that the metric data alone goes surprisingly far in recovering particular shapes. Additionally our method can, if available, incorporate bending data, which opens up applications that require or manipulate both.

ACKNOWLEDGMENTS

REFERENCES

Michael F. Atiyah. 1971. Riemann Surface and Spin Structures. *Ann. sci. de l'Éco. N. Sup.* 4, 1 (1971), 47–62.

Seung-Yeob Baek, Jeonghun Lim, and Kunwoo Lee. 2015. Isometric Shape Interpolation. *Comp. & Graph.* 46 (2015), 257–263.

Abdenago Barros, Esdras Medeiros, and Romildo Silva. 2011. Two Counterexamples of Global Differential Geometry for Polyhedra. *JP J. Geom. Top.* 11, 1 (2011), 65–76.

Marcel Berger. 2010. *Geometry Revealed.* Springer.

Vincent Borrelli, Saïd Jabrane, Francis Lazarus, and Boris Thibert. 2012. Flat Tori in Three-Dimensional Space and Convex Integration. *Proc. Nat. Acad. Sci.* 109, 19 (2012), 7218–7223. Project page: http://hevea-project.fr/ENIndexHevea.html.

Vincent Borrelli, Saïd Jabrane, Francis Lazarus, and Boris Thibert. 2013. Isometric Embeddings of the Square Flat Torus in Ambient Space. *Ensaios Matemáticos* 24 (2013), 1–91.

Friedrich Bös, Max Wardetzky, Etienne Vouga, and Omer Gottesman. 2016. On the Incompressibility of Cylindrical Origami Patterns. *J. Mech. Des.* 139, 2 (2016), 021404:1–9.

Davide Boscaini, Davide Eynard, Drosos Kourounis, and Michael M. Bronstein. 2015. Shape-from-Operator: Recovering Shapes from Intrinsic Operators. *Comp. Graph. Forum* 34, 2 (2015), 265–274.

Mario Botsch, Mark Pauly, Markus Gross, and Leif Kobbelt. 2006. PriMo: Coupled Prisms for Intuitive Surface Modeling. In *Proc. Symp. Geom. Proc.* 11–20.

Sofien Bouaziz, Mario Deuss, Yuliy Schwartzburg, Thibaut Weise, and Mark Pauly. 2012. Shape-Up: Shaping Discrete Geometry with Projections. *Comp. Graph. Forum* 31, 5 (2012), 1657–1667.

Werner Boy. 1903. Über die Curvatura integra und die Topologie geschlossener Flächen. *Math. Ann.* 57 (1903), 151–184.

Robert Bridson, S. Marino, and Ronald Fedkiw. 2003. Simulation of Clothing with Folds and Wrinkles. In *Proc. Symp. Comp. Anim.* 28–36.

Yu. D. Burago and V. A. Zalgaller. 1960. Polyhedral Embedding of a Flat Metric with Conical Singularities. *Vestnik Leningrad Univ.* 15, 7 (1960), 66–80.

Yu. D. Burago and V. A. Zalgaller. 1995. Isometric Piecewise-Linear Embeddings of Two-Dimensional Manifolds with a Polyhedral Metric into $\mathbb{R}^3$. *Algebra i Analiz* 7, 3 (1995), 76–95. English translation in *St. Petersburg Math. J.* 7, 3 (1996), 369–385.

Isaac Chao, Ulrich Pinkall, Patrick Sanan, and Peter Schröder. 2010. A Simple Geometric Model for Elastic Deformations. *ACM Trans. Graph.* 29, 4 (2010), 38:1–38:6.

Keenan Crane, Mathieu Desbrun, and Peter Schröder. 2010. Trivial Connections on Discrete Surfaces. *Comp. Graph. Forum* 29, 5 (2010), 1525–1533.

Keenan Crane, Ulrich Pinkall, and Peter Schröder. 2011. Spin Transformations of Discrete Surfaces. *ACM Trans. Graph.* 30, 4 (2011), 104:1–104:10.

Keenan Crane, Ulrich Pinkall, and Peter Schröder. 2013. Robust Fairing via Conformal Curvature Flow. *ACM Trans. Graph.* 32, 4 (2013), 61:1–61:10.

Mathieu Desbrun, Eva Kanso, and Yiying Tong. 2008. Discrete Differential Forms for Computational Modeling. In *Discrete Differential Geometry*, Alexander I. Bobenko, Peter Schröder, John M. Sullivan, and Günther M. Ziegler (Eds.). Oberwolfach Seminars, Vol. 38. Birkhäuser Verlag.

Paweł Dłotko. 2012. A Fast Algorithm to Compute Cohomology Group Generators of Orientable 2-Manifolds. *Patt. Recog. Lett.* 33, 11 (2012), 1468–1476.

David Eppstein. 2003. Dynamic Generators of Topologically Embedded Graphs. In *Proc. ACM-SIAM Symp. Disc. Alg.* 599–608.

Jeff Erickson and Kim Whittlesey. 2005. Greedy Optimal Homotopy and Homology Generators. In *Proc. ACM-SIAM Symp. Disc. Alg.* 1038–1046.

Gerd Fischer (Ed.). 2017. *Mathematical Models* (2nd ed.). Springer Spektrum.

Stefan Fröhlich and Mario Botsch. 2011. Example-Driven Deformations Based on Discrete Shells. *Comp. Graph. Forum* 30, 8 (2011), 2246–2257.

Herman Gluck. 1974. Almost all Simply Connected Closed Surfaces are Rigid. In *Geometric Topology (Lecture Notes in Mathematics)*, Vol. 438. Springer, 225–239.

Eitan Grinspun, Anil Hirani, Mathieu Desbrun, and Peter Schröder. 2003. Discrete Shells. In *Proc. Symp. Comp. Anim.* 62–67.

Misha Gromov. 1986. *Partial Differential Relations.* Springer.

André Haefliger and Valentin Poénaru. 1964. La Classification des Immersions Combinatoires. *Pub. Math. l'IHÉS* 23 (1964), 75–91.

Joel Hass and John Hughes. 1985. Immersions of Surfaces in 3-Manifolds. *Topology* 24, 1 (1985), 97–112.

Morris W. Hirsch. 1959. Immersions of Manifolds. *Trans. Amer. Math. Soc.* 93 (1959), 242–276.

Tim Hoffmann and Zi Ye. 2018. A Discrete Extrinsic and Intrinsic Dirac Operator. (2018). arXiv:1802.06278.

Martin Isenburg, Stefan Gumhold, and Craig Gotsman. 2001. Connectivity Shapes. In *Proc. IEEE Vis.* 135–142.

Eric Jones, Travis Oliphant, Pearu Peterson, and others. 2001. SciPy: Open Source Scientific Tools for Python. Online at http://scipy.org/. (2001).

Louis H. Kauffman and Thomas F. Banchoff. 1977. Immersions and mod-2 Quadratic Forms. *Amer. Math. Month.* 84, 3 (1977), 168–185.

Michael Kazhdan, Jake Solomon, and Mirela Ben-Chen. 2012. Can Mean-Curvature Flow Be Made Non-Singular? *Comp. Graph. Forum* 31, 5 (2012), 1745–1754.

Scott Kircher and Michael Garland. 2008. Free-Form Motion Processing. *ACM Trans. Graph.* 27, 2 (2008), 12:1–12:13.

Nicolaas H. Kuiper. 1955. On C^1-isometric Imbeddings I & II. *Indag. Math.* 58 (1955), 545–556, 683–689.

H. Blaine Lawson and Marie-Louise Michelsohn. 1990. *Spin Geometry.* PMS, Vol. 38. Princeton UP.

Peter Li and Shing-Tung Yau. 1982. A New Conformal Invariant and its Applications to the Willmore Conjecture and the first Eigenvalue of Compact Surfaces. *Inv. Math.* 69, 2 (1982), 269–291.

Yaron Lipman, Olga Sorkine, David Levin, and Daniel Cohen-Or. 2005. Linear Rotation-Invariant Coordinates for Meshes. *ACM Trans. Graph.* 24, 3 (2005), 479–487.

Nelson Max. 1976. Turning a Sphere Inside Out. Computer Animation. (1976). Available from CRC Press.

Nelson Max and Tom Banchoff. 1981. *Contributions to Analysis and Geometry.* JH Univ. P., Chapter Every Sphere Eversion has a Quadruple Point, 191–209.

John Nash. 1954. C^1 Isometric Imbeddings. *Math. Ann.* 60, 3 (1954), 383–396.

Deniele Panozzo, Enrico Puppo, Marco Tarini, and Olga Sorkine-Hornung. 2014. Frame Fields: Anisotropic and Non-orthogonal Cross Fields. *ACM Trans. Graph.* 33, 4 (2014), 134:1–134:11.

Ulrich Pinkall. 1985. Regular Homotopy Classes of Immersed Surfaces. *Topology* 24, 4 (1985), 421–434.

Henry Segerman. 2016. *Visualizing Mathematics with 3D Printing.* Johns Hopkins U. P.

Carlo H. Séquin. 2011. *Torus Immersions and Transformations.* Technical Report UCB/EECS-2011-83. UC Berkeley.

Amit Singer. 2011. Angular Synchronization by Eigenvectors and Semidefinite Programming. *App. Comp. Harm. Anal.* 30, 1 (2011), 20–36.

Amit Singer and Hau-Tieng Wu. 2012. Vector Diffusion Maps and the Connection Laplacian. *Comm. Pure Appl. Math.* 65, 8 (2012), 1067–1144.

Stephen Smale. 1959. A Classification of Immersions of the Two-Sphere. *Trans. Amer. Math. Soc.* 90, 2 (1959), 281–290.

Olga Sorkine and Marc Alexa. 2007. As-Rigid-As-Possible Surface Modeling. In *Proc. Symp. Geom. Proc.* 109–116.

Y. Wang, B. Liu, and Y. Tong. 2012. Linear Surface Reconstruction from Discrete Fundamental Forms on Triangle Meshes. *Comp. Graph. Forum* 31, 8 (2012), 2277–2287.

Hassler Whitney. 1937. On Regular Closed Curves in the Plane. *Comp. Math.* 4 (1937), 276–284.

Tim Winkler, Jens Drieseberg, Marc Alexa, and Kai Hormann. 2010. Multi-Scale Geometry Interpolation. *Comp. Graph. Forum* 29, 2 (2010), 309–318.

Min Zhang, Wei Zeng, Ren Guo, Feng Luo, and Xianfeng David Gu. 2015. Survey of Discrete Ricci Flow. *J. Comput. Sci. Technol.* 30, 3 (2015), 598–613.

A SELECTING A SPIN STRUCTURE

A discrete spin structure $[\tau]$ requires $\mathfrak{q}_\tau(\gamma_p) = 0$ for every interior vertex (Def. 3.1). Eq. (8) thus constrains the signs of τ to satisfy

$$\prod_{ij \in \gamma_p} \tau_{ij} = -\exp\left(-\frac{\mathbf{i}}{2} \sum_{ij \in \gamma_p} \kappa_{ij}\right) = \exp\left(\frac{\mathbf{i}}{2}\Omega_p\right) \qquad (23)$$

for all $p \in \mathring{V}$ with Ω_p denoting the usual discrete Gaussian curvature (excess angle) at p. Can this be achieved, and if so, how?

Given an arbitrary choice of sign for τ, Eq. (23) is either satisfied (*figure-0*) or has the wrong sign (*figure-8*), taking the general form

$$\prod_{ij \in \gamma_p} \tau_{ij} = (-1)^{\mathfrak{q}_\tau(\gamma_p)} \exp\left(\frac{\mathbf{i}}{2}\Omega_p\right). \qquad (24)$$

Let $p \in V$ be a *figure-8* vertex. Suppose $\tilde{p}$ is another *figure-8* vertex connected to p by a path Γ of primal edges. Flipping the sign of τ on all edges in Γ *annihilates* both *figure-8* vertices since this introduces an odd number of sign flips in $\prod_{ij \in \gamma_p} \tau_{ij}$ and $\prod_{ij \in \gamma_{\tilde{p}}} \tau_{ij}$ and an even number for all other vertices along the path. If M has a boundary, flipping the sign of τ along a path connecting p to the boundary turns p *figure-0*. Hence all *figure-8* vertices can be removed so long as their number is even for M without boundary. Let M be without boundary. Taking the product of the left-hand side of Eq. (24) over all vertices gives $+1$. Taking the product over all right-hand sides gives $(-1)^{\sum_{p \in V} \mathfrak{q}_\tau(\gamma_p)} \exp(\mathbf{i}\pi\chi)$ for $\chi = 2(1-g)$ the Euler characteristic of M. We conclude that the number of *figure-8* vertices $\sum_{p \in V} \mathfrak{q}_\tau(\gamma_p)$ is even.

Using a vertex spanning tree, rooted at the boundary (if applicable), all *figure-8* vertices can be removed by flipping the sign of τ on every interior edge in the tree which has an odd number of *figure-8* descendents (Alg. 3).

**Algorithm 3** Spin structure

Input: A spin connection τ.
Output: A spin connection τ with $\mathfrak{q}_\tau(\gamma_p) = 0$ for all $p \in \mathring{V}$.
1: Compute $\mathfrak{q}_\tau(\gamma_p)$ for each $p \in \mathring{V}$.
2: Let T be a vertex spanning tree rooted at a boundary (if applicable) vertex.
3: From the leaves to the root, aggregate for each e of T the sum s_e of $\mathfrak{q}_\tau(\gamma_p)$ (mod 2) over the descendents of the edge.
4: For each $e \in T \cap \mathring{E}$, $\tau_e \leftarrow (-1)^{s_e} \tau_e$.

This argument establishes that spin structures, *i.e.*, τ with $\mathfrak{q}_\tau(\gamma_p) = 0$ for all $p \in \mathring{V}$ exist and at least one can be found effectively. Note that flipping the sign of τ along any closed path Γ or any path Γ *closed relative to* ∂M ($\partial \Gamma \subset \partial M$) still yields a spin structure but possibly a different one.

More precisely, for Γ a relatively closed path, flipping the sign of τ along Γ changes the spin structure *if and only if* Γ is a path with a non-trivial 1$^{\text{st}}$ *relative homology* $H_1(M, \partial M)$ (a path that does not enclose a subregion of M by adding only boundary edges to the path). In that case Γ crosses a closed triangle strip γ of a

nontrivial 1$^{\text{st}}$ homology of the dual mesh M^* (inset). Hence switching the sign of τ along Γ controls the value of $\mathfrak{q}_\tau(\gamma)$. Using the Poincaré-Lefschetz duality $H_1(M, \partial M) \cong H_1(M^*)$, switching the sign of τ along each generator $(\Gamma_1, \ldots, \Gamma_{\beta_1})$ for $H_1(M, \partial M)$ effectively enumerates all 2^{β_1} combinations of values of $(\mathfrak{q}_\tau(\gamma_1), \ldots, \mathfrak{q}_\tau(\gamma_{\beta_1})) \in (\mathbb{Z}_2)^{\beta_1}$ where $(\gamma_1, \ldots, \gamma_{\beta_1})$ is a set of generators of $H_1(M^*)$.

To find a set of generators $(\Gamma_1, \ldots, \Gamma_{\beta_1})$ of the relative homology $H_1(M, \partial M)$, we use the Tree/Co-Tree algorithm [Eppstein 2003; Erickson and Whittlesey 2005] in the variant that can deal with boundaries [Dłotko 2012] (Alg. 4).
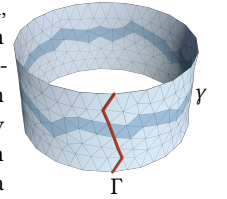
Algorithm 4 Generators for the 1$^{\text{st}}$ relative homology.

1: Let the *Co-Tree* $T^* \subset E^*$ be a triangle (dual) spanning tree.
2: If $\partial M \neq \emptyset$ let $b \in E \setminus \mathring{E}$, and $v_0 \in V \setminus \mathring{V}$ in the same boundary component as b. Else pick any $v_0 \in V$.
3: Let the *Tree* $T \subset (E \setminus b \setminus (T^*)^*)$ be a spanning tree rooted at v_0.
4: Let $G = E \setminus (b \cup T \cup (T^*)^*)$. ▷ $|G| = \beta_1 = 2g + \max\{|\partial M| - 1, 0\}$.
5: $k = 0$; ▷ Counter
6: **for** $e \in G$ **do**
7: $\quad k \leftarrow k + 1$;
8: $\quad$ **if** $e \in \mathring{E}$ **then**
9: $\quad\quad$ Let Γ_k be the unique cycle in $\{e\} \cup T$.
10: $\quad$ **else**
11: $\quad\quad$ Let Γ_k be a path in T from src(e) or dst(e) to v_0.
12: $\quad$ **end if**
13: **end for**
Output: $(\Gamma_1, \ldots, \Gamma_{\beta_1})$ is a set of generators for $H_1(M, \partial M)$.

B PROOF OF THM. 4.2

To prove Thm. 4.2 we proceed in stages. Given an isometric generic map f, App. B.1 establishes that any γ has the correct *figure-8/0* property as long as we account for the rim bits (Def. 4.1) along γ. Next (App. B.2) we show that neither the *figure-8/0* property nor the rim bits depend on the metric. This then allows us to conclude that a not necessarily isometric map is still in the correct regular homotopy class defined with respect to the intended metric (App. B.3).

B.1 Spin Gauss-Bonnet Theorem

We begin with considering a single triangle strip mapped isometrically into $\mathbb{R}^3$ by a generic map f. That is, on each triangle i the polar decomposition Eq. (11) is an isometric map $df_i(X) = \bar{\lambda}_i \phi_i(X) \mathbf{j} \lambda_i$.

LEMMA B.1 (SPIN GAUSS-BONNET THEOREM). *Let M be a closed triangle strip with a metric, and let γ be the periodic ordered sequence of shared edges between triangles. Following Sec. 3 let τ be a discrete spin connection defined over γ. Suppose $f: M \to \mathbb{R}^3$ is an isometric generic map of the strip with*

$$df_i(X) = \bar{\lambda}_i \phi_i(X) \mathbf{j} \lambda_i$$

for some unit quaternions λ_i. Then

$$\mathfrak{q}_f(\gamma) = \mathfrak{q}_\tau(\gamma) + \sum_{e \in \gamma} \mathfrak{s}_e^{\tau, \lambda} \qquad (\text{mod } 2). \qquad (25)$$

PROOF. Without loss of generality we let the face indices along the triangle strip be the ordered sequence $(0, 1, \ldots, m-1)$ hence $\gamma = (e_0, \ldots, e_{m-1}) = (e_{0,1}, \ldots, e_{m-1,0})$ with indices taken modulo-m.

Recall Def. 2.3 and Eq. (2)

$$(-1)^{\mathfrak{q}_f(\gamma)+1} = q_{0,1} \cdots q_{m-1,0} \qquad (26)$$

where each $q_{i,i+1}$ is given in terms of the bending angle and geodesic curvature, $\alpha_{i,i+1}, \kappa_{i,i+1} \in (-\pi, \pi)$ (*cf.* Eq. (1))

$$q_{i,i+1} = \exp\left(-\tfrac{\kappa_{i,i+1}}{2}\mathbf{n}_i\right) \exp\left(-\tfrac{\alpha_{i,i+1}}{2}\tfrac{\mathbf{v}_{i,i+1}}{|\mathbf{v}_{i,i+1}|}\right). \qquad (27)$$

Since $\alpha_{i,i+1}$ is the angle between neighboring triangle normals

$$\mathbf{n}_{i+1} = \exp\left(\tfrac{\alpha_{i,i+1}}{2}\tfrac{\mathbf{v}_{i,i+1}}{|\mathbf{v}_{i,i+1}|}\right)\mathbf{n}_i \exp\left(-\tfrac{\alpha_{i,i+1}}{2}\tfrac{\mathbf{v}_{i,i+1}}{|\mathbf{v}_{i,i+1}|}\right)$$

$$\implies \quad \exp\left(-\tfrac{\alpha_{i,i+1}}{2}\tfrac{\mathbf{v}_{i,i+1}}{|\mathbf{v}_{i,i+1}|}\right)\mathbf{n}_{i+1} = \mathbf{n}_i \exp\left(-\tfrac{\alpha_{i,i+1}}{2}\tfrac{\mathbf{v}_{i,i+1}}{|\mathbf{v}_{i,i+1}|}\right),$$

and therefore

$$\exp\left(-\tfrac{\alpha_{i,i+1}}{2}\tfrac{\mathbf{v}_{i,i+1}}{|\mathbf{v}_{i,i+1}|}\right) \exp\left(-\tfrac{\kappa_{i,i+1}}{2}\mathbf{n}_{i+1}\right)$$
$$= \exp\left(-\tfrac{\kappa_{i,i+1}}{2}\mathbf{n}_i\right) \exp\left(-\tfrac{\alpha_{i,i+1}}{2}\tfrac{\mathbf{v}_{i,i+1}}{|\mathbf{v}_{i,i+1}|}\right). \qquad (28)$$

Returning to the product of Eq. (26), we substitute each $q_{i,i+1}$ from Eq. (27) and repeatedly apply Eq. (28) to "push" the factors involving $\kappa_{i,i+1}$ to the left

$$(-1)^{\mathfrak{q}_f(\gamma)+1} = \left(\prod_{i=0}^{m-1} \exp\left(-\tfrac{\kappa_{i,i+1}}{2}\mathbf{n}_0\right)\right)\left(\prod_{i=0}^{m-1} \exp\left(-\tfrac{\alpha_{i,i+1}}{2}\tfrac{\mathbf{v}_{i,i+1}}{|\mathbf{v}_{i,i+1}|}\right)\right)$$
$$(29)$$

where the symbol $\prod_{i=0}^{m-1} q_i$ represents the *ordered* product $q_0 \cdots q_{m-1}$. Since f is isometric, the frame represented by λ is integrable (Sec. 5.1)

and Eq. (15) applies, from which we find

$$\exp\left(-\tfrac{\alpha_{i,i+1}}{2}\tfrac{\mathbf{v}_{i,i+1}}{|\mathbf{v}_{i,i+1}|}\right) = \bar{\lambda}_{i+1} \exp\left(\tfrac{\alpha_{i,i+1}}{2}z_{i+1,i}\mathbf{j}\right)\lambda_{i+1}$$

$$= (-1)^{\mathfrak{s}_{i,i+1}^{\tau,\lambda}} \bar{\lambda}_{i+1}\lambda_{i+1}(\overline{\tau_{i,i+1}\lambda_i})\lambda_{i+1}$$

$$= (-1)^{\mathfrak{s}_{i,i+1}^{\tau,\lambda}} \bar{\lambda}_i \bar{\tau}_{i,i+1}\lambda_{i+1}$$

allowing us to collapse Eq. (29) into

$$(-1)^{\mathfrak{q}_f(\gamma)+1} = (-1)^{\sum_{e \in \gamma} \mathfrak{s}_e^{\tau,\lambda}}\left(\prod_{i=0}^{m-1} \exp\left(-\tfrac{\kappa_{i,i+1}}{2}\mathbf{n}_0\right)\right)\bar{\lambda}_0\left(\prod_{i=0}^{m-1} \bar{\tau}_{i,i+1}\right)\lambda_0$$

$$= (-1)^{\sum_{e \in \gamma} \mathfrak{s}_e^{\tau,\lambda}} \bar{\lambda}_0 \exp\left(-\tfrac{\mathbf{i}}{2}\sum_{i=0}^{m-1}\kappa_{i,i+1}\right)\left(\prod_{i=0}^{m-1} \bar{\tau}_{i,i+1}\right)\lambda_0$$

using $\prod_{i=0}^{m-1} \exp\left(-\tfrac{\kappa_{i,i+1}}{2}\mathbf{n}_0\right) = \bar{\lambda}_0 \exp\left(-\tfrac{\mathbf{i}}{2}\sum_{i=0}^{m-1}\kappa_{i,i+1}\right)\lambda_0$. Finally, substituting the definition of $\mathfrak{q}_\tau(\gamma)$ (Eq. (8)), we arrive at

$$(-1)^{\mathfrak{q}_f(\gamma)+1} = (-1)^{\sum_{e \in \gamma} \mathfrak{s}_e^{\tau,\lambda}} \bar{\lambda}_0 (-1)^{\mathfrak{q}_\tau(\gamma)+1}\lambda_0$$

$$= (-1)^{\mathfrak{q}_\tau(\gamma)+1+\sum_{e \in \gamma} \mathfrak{s}_e^{\tau,\lambda}}$$

which yields our claim (Eq. (25)). □

B.2 Metric Independence

Though the construction of a spin connection τ in Sec. 3 relies on a metric, the resulting $\mathfrak{q}_\tau$ is a topological quantity *independent* of the metric, as we will show in this section.

Suppose M is endowed with two different discrete metrics ℓ and $\tilde{\ell}$. For ℓ, choose a basis field ϕ (Eq. (5)) and let τ be a prescribed discrete spin connection with respect to ϕ. Now let $\tilde{\phi}$ be a basis field isometric with respect to $\tilde{\ell}$, defined uniquely by requiring that it is related to ϕ through self-adjoint, positive linear maps $S_i: T_i M \to T_i M$ satisfying $\tilde{\phi}_i(X) = \phi_i(S_i X)$ for $X \in T_i M$ and all $i \in F$.

For $ij \in E$ define $\beta_{ij} \in (-\tfrac{\pi}{2}, \tfrac{\pi}{2})$ to be the angle measuring the deviation of $\tilde{\phi}_i(v_{ij})$ from $\phi_i(v_{ij})$

$$\frac{\tilde{\phi}_i(v_{ij})}{|\tilde{\phi}_i(v_{ij})|} = \exp(\mathbf{i}\beta_{ij})\frac{\phi_i(v_{ij})}{|\phi_i(v_{ij})|}. \qquad (30)$$

That β_{ij} can be chosen in $(-\tfrac{\pi}{2}, \tfrac{\pi}{2})$ follows from the isometry of the ϕ_i and the positive definiteness of the S_i using the real inner product on $\mathbb{C} \equiv \mathbb{R}^2$:

$$\langle \phi_i(v_{ij}), \tilde{\phi}_i(v_{ij})\rangle = \langle \phi_i(v_{ij}), \phi_i(S_i v_{ij})\rangle = \langle v_{ij}, S_i v_{ij}\rangle_{T_i M} > 0.$$

Substituting Eq. (6) in Eq. (30) we find that the parallel transports r and $\tilde{r}$ arising from the choice of basis fields ϕ resp. $\tilde{\phi}$ are related as

$$\tilde{r}_{ij} = \exp\left(\mathbf{i}(\beta_{ji} - \beta_{ij})\right)r_{ij}, \qquad (31)$$

which in turn determines $\tilde{\tau}$ up to sign. To choose this sign consistently we define

$$\tilde{\tau}_{ij} := \exp\left(\tfrac{\mathbf{i}}{2}(\beta_{ji} - \beta_{ij})\right)\tau_{ij} \qquad (32)$$

and get:

LEMMA B.2. $\mathfrak{q}_{\tilde{\tau}}(\gamma) = \mathfrak{q}_\tau(\gamma)$ *for every closed triangle strip γ in M.*

PROOF. Let γ be an arbitrary closed triangle strip, and for the sake of the proof label the triangles along the strip by $(0, 1, \ldots, m-1)$.

Using Eq. (8), $\mathfrak{q}_\tau$ resp. $\mathfrak{q}_{\tilde\tau}$ are given by

$$(-1)^{\mathfrak{q}_\tau+1} = \exp\left(\frac{\mathbf{i}}{2}\sum_{i=0}^{m-1}\kappa_{i,i+1}\right)\prod_{i=0}^{m-1}\tau_{i,i+1} \qquad (33a)$$
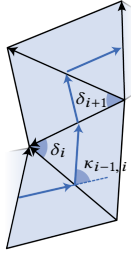
$$(-1)^{\mathfrak{q}_{\tilde\tau}+1} = \exp\left(\frac{\mathbf{i}}{2}\sum_{i=0}^{m-1}\tilde\kappa_{i,i+1}\right)\prod_{i=0}^{m-1}\tilde\tau_{i,i+1} \qquad (33b)$$

involving the total geodesic curvatures $\sum_{i=0}^{m-1}\kappa_{i,i+1}$ resp. $\sum_{i=0}^{m-1}\tilde\kappa_{i,i+1}$ which depend on the metric. To relate the two total geodesic curvatures we use an alternate representation

$$\sum_{i=0}^{m-1}\kappa_{i,i+1} = \sum_{i=0}^{m-1}\delta_i$$

where δ_i, at each triangle i in the strip, is the angle between the edges shared by the previous and the next triangle

$$\delta_i := \arg\left(-\frac{\phi_i(v_{i,i+1})}{\phi_i(v_{i,i-1})}\right) \in (-\pi,\pi).$$

In the $\tilde\ell$ metric, using Eq. (30), these interior angles are given by

$$\tilde\delta_i = \arg\left(-\frac{\tilde\phi_i(v_{i,i+1})}{\tilde\phi_i(v_{i,i-1})}\right) = \arg\left(-\exp\left(\mathbf{i}(\beta_{i,i+1}-\beta_{i,i-1})\right)\frac{\phi_i(v_{i,i+1})}{\phi_i(v_{i,i-1})}\right).$$
$$(34)$$

Since ϕ and $\tilde\phi$ both preserve orientation, δ_i and $\tilde\delta_i$ share the same sign. Together with $\beta_{i,i+1}-\beta_{i,i-1} \in (-\pi,\pi)$ (since $\beta_{i,i+1},\beta_{i,i-1} \in (-\frac{\pi}{2},\frac{\pi}{2})$) this determines the branch for Eq. (34) leading us to conclude

$$\tilde\delta_i = \delta_i + \beta_{i,i+1} - \beta_{i,i-1},$$

and hence

$$\sum_{i=0}^{m-1}\tilde\kappa_{i,i+1} = \sum_{i=0}^{m-1}\tilde\delta_i = \sum_{i=0}^{m-1}(\delta_i + \beta_{i,i+1} - \beta_{i,i-1})$$
$$= \sum_{i=0}^{m-1}\kappa_{i,i+1} + \sum_{i=0}^{m-1}(\beta_{i,i+1} - \beta_{i+1,i}). \qquad (35)$$

The proof is now completed by substituting Eqs. (32) and (35) into Eq. (33b) and noting its equality to Eq. (33a). □

LEMMA B.3. *Given any unit quaternion field λ and Eq. (32)*

$$\mathrm{sgn}\langle\lambda_j,\tilde\tau_{ij}\lambda_i\rangle = \mathrm{sgn}\langle\lambda_j,\tau_{ij}\lambda_i\rangle$$

for all $ij \in \mathring{E}$. Hence, whenever both rim bits are defined (Sec. 4.1), i.e., $\langle\lambda_j,\tau_{ij}\lambda_i\rangle \neq 0$ and $\langle\lambda_j,\tilde\tau_{ij}\lambda_i\rangle \neq 0$

$$\mathsf{s}_{ij}^{\tau,\lambda} = \mathsf{s}_{ij}^{\tilde\tau,\lambda}.$$

PROOF. Eq. (32) gives

$$\langle\lambda_j,\tilde\tau_{ij}\lambda_i\rangle = \cos\left(\frac{\beta_{ji}-\beta_{ij}}{2}\right)\langle\lambda_j,\tau_{ij}\lambda_i\rangle.$$

With $\cos\left(\frac{\beta_{ji}-\beta_{ij}}{2}\right) > 0$ ($\beta_{ij},\beta_{ji} \in (-\frac{\pi}{2},\frac{\pi}{2})$) the claim follows. □

B.3 Proof of Thm. 4.2

PROOF OF THM. 4.2. Let M be the triangle mesh with metric ℓ, ϕ a basis field (Eq. (5)), and τ a given spin connection. A not necessarily isometric generic map $f: M \to \mathbb{R}^3$ then gives rise to polar decompositions (Eq. (11)) for some λ and $S = (S_i)_{i\in F}$.

Let $\tilde\ell$ denote the metric induced by f, i.e., $\tilde\ell_{ij} := |df(v_{ij})| = |\mathbf{v}_{ij}|$ for $ij \in E$, which makes f isometric with respect to $\tilde\ell$. For the basis field $\tilde\phi$ choose $\tilde\phi_i$ which satisfy $\tilde\phi_i(X) := \phi_i(S_iX)$ for $X \in T_iM$ in terms of which Eq. (11) becomes

$$df_i(X) = \bar\lambda_i\tilde\phi_i(X)\mathbf{j}\lambda_i. \qquad (36)$$

This makes App. B.2 applicable to $\tilde\phi$ and $\tilde\ell$. Furthermore, Eq. (36) implies that λ is a unit quaternion field describing the $\mathbb{R}^3$-orientation for the $\tilde\ell$-isometric map f as in Lem. B.1. Following App. B.2 we construct $\tilde\tau$ as in Eq. (32) and apply Lem. B.1 to conclude that along each triangle strip γ in M

$$\mathfrak{q}_f(\gamma) = \mathfrak{q}_{\tilde\tau}(\gamma) + \sum_{e\in\gamma}\mathsf{s}_e^{\tilde\tau,\lambda} \qquad (\mathrm{mod}\ 2).$$

The claim now follows via Lem. B.2 and Lem. B.3. □

C LOCAL INTEGRABILITY DERIVATIONS

C.1 Integrability and Bending Angle

Here we derive Eq. (15), stating that

$$\lambda_j(\overline{\tau_{ij}\lambda_i}) = (-1)^{\mathsf{s}_{ij}^{\tau,\lambda}}\exp\left(\frac{\alpha_{ij}}{2}z_{ji}\mathbf{j}\right).$$

First note that $\mathbf{i}$ is mapped to the triangle normal by λ_i since $\phi_i\mathbf{j}$ maps the tangent space T_iM to the $(\mathbf{j},\mathbf{k})$-plane, $\mathbf{n}_i = \bar\lambda_i\mathbf{i}\lambda_i$. This yields

$$\mathbf{n}_j = \bar\lambda_j\mathbf{i}\lambda_j = \bar\lambda_j(\tau_{ij}\mathbf{i}\bar\tau_{ij})\lambda_j = \bar\lambda_j\tau_{ij}(\lambda_i\mathbf{n}_i\bar\lambda_i)\bar\tau_{ij}\lambda_j$$
$$= \bar\lambda_j\left(\overline{\lambda_j(\overline{\tau_{ij}\lambda_i})}\right)\lambda_j\mathbf{n}_i\bar\lambda_j\left(\lambda_j(\overline{\tau_{ij}\lambda_i})\right)\lambda_j.$$

Comparing with

$$\mathbf{n}_j = \exp\left(\frac{\alpha_{ij}}{2}\frac{\omega_i(v_{ij})}{|\omega_i(v_{ij})|}\right)\mathbf{n}_i\exp\left(-\frac{\alpha_{ij}}{2}\frac{\omega_i(v_{ij})}{|\omega_i(v_{ij})|}\right)$$

we find

$$\bar\lambda_j\left(\lambda_j(\overline{\tau_{ij}\lambda_i})\right)\lambda_j = \exp\left(-\frac{\alpha_{ij}}{2}\frac{\omega_i(v_{ij})}{|\omega_i(v_{ij})|}\right).$$

Subsequently using Eq. (10) and Eq. (14) the assertion, including its sign, follows.

C.2 Local Integrability Condition

Here we derive Eq. (17) from Eq. (16) through a sequence of equivalences between linear spans of vectors in $\mathbb{H} \equiv \mathbb{R}^4$. Firstly we find

$$\text{Eq. (16)} \iff (\tau_{ij}\lambda_i)\bar\lambda_j \in \mathrm{Span}\{1,z_{ji}\mathbf{j}\}$$
$$\iff \tau_{ij}\lambda_i \in \mathrm{Span}\{\lambda_j,z_{ji}\mathbf{j}\lambda_j\}$$
$$\iff \lambda_j - \tau_{ij}\lambda_i \in \mathrm{Span}\{\lambda_j,z_{ji}\mathbf{j}\lambda_j\} \qquad (37)$$

and secondly

$$\text{Eq. (16)} \iff \lambda_j \in \mathrm{Span}\{\tau_{ij}\lambda_i,z_{ji}\mathbf{j}\tau_{ij}\lambda_i\}$$
$$\iff \lambda_j - \tau_{ij}\lambda_i \in \mathrm{Span}\{\tau_{ij}\lambda_i,z_{ji}\mathbf{j}\tau_{ij}\lambda_i\}. \qquad (38)$$
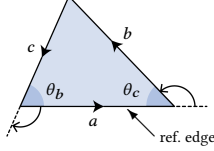
Combining these we get

$$\text{Eq. (37) and Eq. (38)} \iff \lambda_j - \tau_{ij}\lambda_i \in \mathrm{Span}\{\mu_{ji},z_{ji}\mathbf{j}\mu_{ji}\}$$

using the definition of the midframe Eq. (18). Noting that $(\lambda_j - \tau_{ij}\lambda_i)\perp\mu_{ji}$, Eq. (17) follows.

D IMPLEMENTATION DETAILS

In this section we give all needed implementation details to minimize Eq. (20) by gradient descent or Newton's method.

Given (M, ℓ) as input we first determine all Euclidean triangle angles using the half-angle formula

$$\theta_b = 2\tan^{-1}\sqrt{\frac{(\ell_a+\ell_b-\ell_c)(\ell_b+\ell_c-\ell_a)}{(\ell_c+\ell_a-\ell_b)(\ell_a+\ell_b+\ell_c)}}$$

where a, b, c denote the sides of the triangle (inset). These give rise to the usual discrete Gaussian curvatures $\Omega_p = 2\pi - \sum_{i\ni p}\theta_p^i$ where θ_p^i denotes the angle in $i \in F$ incident on $p \in \mathring{V}$. Next we fix ϕ by picking an arbitrary reference edge e in i and setting $\phi_i(e) := \ell_e$. Eq. (14) gives $z_a = 1$, $z_b = \exp(\mathbf{i}(\pi - \theta_c))$, and $z_c = \exp(\mathbf{i}(\theta_b - \pi))$ (inset) for all triangles. With these Eq. (6) yields $r = \left(-\frac{z_{ji}}{z_{ij}}\right)_{ij\in\mathring{E}}$, and, up to sign, $\tau = (\tau_{ij})_{ij\in\mathring{E}}$. The Tree/CoTree algorithm (App. A) is used to choose the signs so that $\mathfrak{q}_\tau(\gamma_p) = 0$ (*figure-0*) for all $p \in \mathring{V}$ and generators of the 1st Homology are selected to be *figure-8* resp. *-0* as desired for the intended regular homotopy class (Def. 3.1). For $ij \in \mathring{E}$ we define the orthonormal quaternions

$$\mathbf{u}_{ji,0} := \mathbf{1}, \quad \mathbf{u}_{ji,1} := z_{ji}\mathbf{j}, \quad \mathbf{u}_{ji,2} := \mathbf{i}, \quad \mathbf{u}_{ji,3} := \mathbf{i}z_{ji}\mathbf{j}$$

rendering the basis of Eq. (19) as $\{\mathbf{u}_{ji,k}\mu_{ji}\}_{k=0\ldots3}$ and Eq. (20) as

$$\mathcal{E}_\epsilon(\lambda) = \tfrac{1}{2}\sum_{ij\in\mathring{E}}\sum_{k=0}^{3} w_{ij}\epsilon_k\langle\mathbf{u}_{ji,k}\mu_{ji}, \lambda_j - \tau_{ij}\lambda_i\rangle^2. \qquad (39)$$

Before proceeding we first express Eq. (39) using only real vectors resp. matrices. For a quaternion $q = a\mathbf{1} + b\mathbf{i} + c\mathbf{j} + d\mathbf{k}$ define $(q)_4$ and $(q)_{4\times4}$ as its real column vector resp. block matrix representation

$$(q)_4 := \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}, \quad (q)_{4\times4} := \begin{bmatrix} a & -b & -c & -d \\ b & a & -d & c \\ c & d & a & -b \\ d & -c & b & a \end{bmatrix}.$$

For two quaternions $p, q \in \mathbb{H}$ we have $(pq)_4 = (p)_{4\times4}(q)_4$, $(\bar{p})_{4\times4} = (p)_{4\times4}^\intercal$ and $\langle p, q\rangle = (p)_4^\intercal(q)_4$.

To write Eq. (39) in real matrix form define the sparse matrix D and the diagonal matrix W consisting of $|\mathring{E}| \times |T|$ resp. $|\mathring{E}| \times |\mathring{E}|$ (4×4) blocks with non-zero entries for each $ij \in \mathring{E}$

$$D_{ij,i} := -(\tau_{ij})_{4\times4}, \quad D_{ij,j} := (1)_{4\times4}, \quad \text{and}$$

$$W_{ij,ij} := w_{ij}\sum_{k=0}^{3}\epsilon_k(\mathbf{u}_{ji,k}\mu_{ji})_4(\mathbf{u}_{ji,k}\mu_{ji})_4^\intercal.$$

Letting now λ denote the $|F| \times 1$ column vector of $\mathbb{R}^4$ blocks $(\lambda_i)_4$, Eq. (39) becomes

$$\mathcal{E}_\epsilon(\lambda) = \tfrac{1}{2}\lambda^\intercal \underbrace{D^\intercal W D}_{=:L} \lambda$$

with its gradient given by an $|F| \times 1$ vector of $\mathbb{R}^4$ blocks

$$\text{grad}\,\mathcal{E}_\epsilon = L\lambda + g.$$

Here g collects the terms due to the λ-dependency of L. Explicitly, for $i \in F$, $(g_i)_4$ depends on the triangles sharing an edge with i

$$(g_i)_4 = \sum_{ij\in\mathring{E}}\left(\frac{\partial\mu_{ji}}{\partial\lambda_i}\right)_{4\times4}\sum_{k=0}^{3}w_{ij}\epsilon_k\langle\mathbf{u}_{ji,k}\mu_{ji}, \lambda_j - \tau_{ij}\lambda_i\rangle\left(\bar{\mathbf{u}}_{ji}(\lambda_j - \tau_{ij}\lambda_i)\right)_4$$

with

$$\left(\frac{\partial\mu_{ji}}{\partial\lambda_i}\right)_{4\times4} = \left((1)_{4\times4} - (\mu_{ji})_4(\mu_{ji})_4^\intercal\right)\frac{(\tau_{ij})_{4\times4}}{|\tau_{ij}\lambda_i+\lambda_j|}$$

using Eq. (18).

The gradient descent method for minimizing $\mathcal{E}_\epsilon$ can now be seen as simulating the time-dependent problem

$$\begin{cases} M\frac{\partial}{\partial t}\lambda = -L\lambda - g + P\lambda \\ |(\lambda_i)_4|^2 = 1, \quad i \in F \end{cases} \qquad (40)$$

with $M = \text{diag}(((A_i)_{4\times4})_{i\in F})$ the block diagonal mass matrix of triangle areas A_i and $P = \text{diag}(((p_i)_{4\times4})_{i\in F})$ the block diagonal matrix of Lagrange multipliers $p\colon F \to \mathbb{R}$ enforcing the constraints $|\lambda_i| = 1$. To simulate Eq. (40) we use a semi-implicit method with projection:

Algorithm 5 Gradient descent for minimizing $\mathcal{E}_\epsilon$

Input: Stiffness ϵ; time step $\Delta t > 0$; initial guess λ.
Output: Approximated minimizer for $\mathcal{E}_\epsilon(\lambda)$.
1: **while** not satisfied **do**
2: $\quad\lambda \leftarrow \textsc{GradientStep}(\lambda, \Delta t)$
3: **end while**
4: **function** $\textsc{GradientStep}(\lambda, \Delta t)$
5: $\quad$ Build g;
6: $\quad\lambda \leftarrow \lambda - \Delta t M^{-1}g$;   ▷ Forward Euler for $M\dot\lambda = -g$.
7: $\quad$ Build L;
8: $\quad\lambda \leftarrow (M + \Delta t L)^{-1}M\lambda$;   ▷ Backward Euler for $M\dot\lambda = -L\lambda$.
9: $\quad(\lambda_i)_4 \leftarrow \frac{(\lambda_i)_4}{|(\lambda_i)_4|}$ for each $i \in F$;   ▷ Constraint projection.
10: $\quad$ **return** λ
11: **end function**

Near a minimum, convergence can be accelerated through the use of Newton's method with $|F|$ equality constraints $|(\lambda_i)_4|^2 - 1 = 0$ for $4|F|$ real variables. The Jacobian matrix for the constraints is an $|F| \times |F|$ diagonal matrix J consisting of (1×4) blocks $J_{i,i} = (\lambda_i)_4^\intercal$.

Newton's method also requires the Hessian matrix of the objective $\mathcal{E}_\epsilon$. We use L to approximate this Hessian matrix to arrive at the following (quasi-)Newton's method.

Algorithm 6 (Quasi-)Newton's method for minimizing $\mathcal{E}_\epsilon$

Input: Stiffness ϵ; initial guess λ.
Output: Approximated minimizer for $\mathcal{E}_\epsilon(\lambda)$.
1: **while** not satisfied **do**
2: $\quad$ Build L, J and g;
3: $\quad\Delta\lambda \leftarrow \text{solve} \begin{bmatrix} L & J^\intercal \\ J & 0 \end{bmatrix}\begin{bmatrix} \Delta\lambda \\ \Delta p \end{bmatrix} = \begin{bmatrix} -L\lambda - g \\ 0 \end{bmatrix}$;
4: $\quad\lambda \leftarrow \lambda + \Delta\lambda$;
5: $\quad(\lambda_i)_4 \leftarrow \frac{(\lambda_i)_4}{|(\lambda_i)_4|}$ for each $i \in F$;   ▷ Constraint projection.
6: **end while**

Once a minimizer has been found we need to compute vertex positions as a final step.

D.1 Computing Vertex Positions

For an optimal λ the local integrability condition Eq. (13) holds approximately, unless the energy of the minimizer vanished. Using Eq. (10) we define a discrete $\mathbb{R}^3$-valued 1-form $\hat{\mathbf{v}} = (\hat{\mathbf{v}}_e)_{e \in E}$ by averaging on interior edges

$$\hat{\mathbf{v}}_{ij} = \frac{\ell_{ij}}{2}\big(\omega_i(v_{ij}) - \omega_j(v_{ji})\big) = \frac{\ell_{ij}}{2}\big(\bar{\lambda}_i z_{ij}\mathbf{j}\lambda_i - \bar{\lambda}_j z_{ji}\mathbf{j}\lambda_j\big), \quad (41)$$

and using direct evaluation with the relevant ω_i for boundary edges. Now $\hat{\mathbf{v}}$ holds all embedded edge vectors which must be integrated into point positions $f : V \to \mathbb{R}^3$. Since this is only possible for exact $\hat{\mathbf{v}}$, we first orthogonally project $\hat{\mathbf{v}}$ onto its exact component. This is expected to introduce only a small error on a simply connected domain since $\hat{\mathbf{v}}$ is *nearly* closed. Subsequently we solve for its "scalar potential" f. All this is accomplished by the Poisson problem

$$\mathsf{d}^\mathsf{T} *_1 \mathsf{d}f = \mathsf{d}^\mathsf{T} *_1 \hat{\mathbf{v}}, \quad (42)$$

where $\hat{\mathbf{v}}$ resp. f are the $|E| \times 3$ resp. $|V| \times 3$ matrices of $\mathbb{R}^3$ values on each edge resp. vertex. The $|E| \times |V|$ sparse matrix d denotes the discrete exterior derivative on 0-forms while the sparse $|E| \times |E|$ diagonal matrix $*_1$ denotes the discrete (diagonal) Hodge star on 1-forms

$$\mathsf{d}_{e,\mathrm{src}(e)} = -1, \quad \mathsf{d}_{e,\mathrm{dst}(e)} = 1, \quad \text{and} \quad (*_1)_{e,e} = \hat{w}_e.$$

where $\mathrm{src}(e)$ and $\mathrm{dst}(e)$ are the source and destination vertices of e, and $\hat{w}_e = w_e^{-1}$ are weights (*e.g.*, cotan) for primal edges.

Algorithm 7 Vertex Computation

Input: λ.
Output: Surface realization $f : V \to \mathbb{R}^3$.
1: $\hat{\mathbf{v}} \leftarrow$ Eq. (41);
2: $f \leftarrow$ solve Eq. (42).

D.2 Penalty Method for Exactness

When the domain is non-simply connected, $\hat{\mathbf{v}}$ may be far from exact even when it is nearly closed ($d\hat{\mathbf{v}} \approx 0$). In that setting we include a penalty term in the energy gradient replacing Eq. (40) by

$$\begin{cases} \mathsf{M}\frac{\partial}{\partial t}\lambda = -\mathsf{L}\lambda - \mathsf{g} - \frac{1}{a}\mathsf{h} + \mathsf{P}\lambda \\ |(\lambda_i)_4|^2 = 1, \quad i \in F \end{cases} \quad (43)$$

with the additional flow direction h designed to bring $\hat{\mathbf{v}}$ to the subspace of exact 1-forms and $a > 0$ serving as penalty parameter.

To construct h we use Eq. (42) to project $\hat{\mathbf{v}}$ to its exact component $df̃$. Unlike the isometric linear map ω_i, the linear map $df̃_i$ is no longer an isometry. Using the polar decomposition (Eq. (11)), let ω_i^* be the closest isometric linear map to $df̃_i$ which determines a quaternionic frame λ_i^* up to sign. Now choose the sign such that $\langle \lambda_i^*, \lambda_i \rangle > 0$. Interpreting $\lambda - \lambda^*$ as the "projection" of $\hat{\mathbf{v}} - df̃$ on the quaternionic frames, we let $\mathsf{h} = \lambda - \lambda^*$ in Eq. (43). Note that for $a \to 0$ the iteration *aside from* GradientStep$(\lambda, \Delta t)$ amounts to alternating projections [Bouaziz et al. 2012]. The gradient step however is essential since it ensures that the spin structure, with

its control over the immersion and homotopy class, enters into the algorithm.

Algorithm 8 Gradient descent with penalty

Input: (Same as Alg. 5), $a \geq 0$;
Output: Approx. minimizer λ for $\mathcal{E}_\epsilon$ yielding exact $\hat{\mathbf{v}}$ for Alg. 7.
1: **while** not satisfied **do**
2:    $\lambda \leftarrow$ GradientStep$(\lambda, \Delta t)$;   ▷ Defined in Alg. 5.
3:    $f̃ \leftarrow$ VertexComputation(λ);   ▷ Alg. 7.
4:    $\omega^* \leftarrow$ RotationPart$(df̃)$;
5:    $\lambda^* \leftarrow$ Quaternion(ω^*);   ▷ Quaternion repres. of rotations.
6:    $\lambda^* \leftarrow$ FixSign(λ, λ^*);   ▷ Ensure $(\lambda_i)_4^\mathsf{T}(\lambda_i^*)_4 > 0$ $i \in F$
7:    $(\lambda_i)_4 \leftarrow \exp(-\frac{\Delta t}{aA_i})(\lambda_i)_4 + \big(1 - \exp(-\frac{\Delta t}{aA_i})\big)(\lambda_i^*)_4$ $i \in F$;
8:    $(\lambda_i)_4 \leftarrow \frac{(\lambda_i)_4}{|(\lambda_i)_4|}$ $i \in F$;   ▷ Constraint projection.
9: **end while**