Curvature-Driven Conformal Deformations

Etienne Corman

Curvature-Driven Conformal Deformations

ETIENNE CORMAN, Université de Lorraine, CNRS, Inria, LORIA, France

In this paper, we introduce a novel approach for computing conformal deformations in $\mathbb{R}^3$ while minimizing curvature-based energies. Curvature-based energies serve as fundamental tools in geometry processing, essential for tasks such as surface fairing, deformation, and approximation using developable or cone metric surfaces. However, accurately computing the geometric embedding, especially for the latter, has been a challenging endeavor. The complexity arises from inherent numerical instabilities in curvature estimation and the intricate nature of differentiating these energies. To address these challenges, we concentrate on conformal deformations, leveraging the curvature tensor as the primary variable in our model. This strategic choice renders curvature-based energies easily applicable, mitigating previous manipulation difficulties. Our key contribution lies in identifying a previously unknown integrability condition that establishes a connection between conformal deformations and changes in curvature. We use this insight to deform surfaces of arbitrary genus, aiming to minimize bending energies or prescribe Gaussian curvature while sticking to positional constraints.

Additional Key Words and Phrases: conformal geometry, Willmore energy, Gaussian curvature

Fig. 1. Minimizing bending energy in Eq. (1) on spot for p equal to 1 (top, Willmore flow), 2 (middle) and 3 (bottom).

1 INTRODUCTION

Curvature flows stand as indispensable tools in computer graphics for surface manipulation. They iteratively deform surfaces by minimizing an energy while complying with given constraints, such as area or volume preservation, positional constraints. They essentially define surfaces as the minimizers of specific energies. Our contribution lies in presenting a unified method for computing *conformal* curvature flows. Our focus revolves around two primary types of curvature-based flows in computer graphics: fairing flows, aimed at surface smoothing, and intrinsic flows, for which computing an embedded representation is often unnecessary.

The first category, fairing flows, progressively molds surfaces to make it as round as possible by minimizing the surface bending. This is quantified by the magnitude of the shape operator S, encapsulating the variations in surface normals across all directions. These techniques have found extensive applications in diverse domains such as biophysics [Canham 1970; Helfrich 1973], computer animation [Bridson et al. 2005; Grinspun et al. 2003; Terzopoulos et al. 1987], inpainting [Bobenko and Schröder 2005; Lévy 2003], and form finding [Joshi and Séquin 2007; Vaxman et al. 2018]. Typically, these bending energies aim to decrease the norm of the shape operator [Hagen and Schulze 1987; Joshi and Séquin 2007; Lott and Pullin 1988; Moreton and Séquin 1992]

$$\mathcal{E}(f) = \frac{1}{2}\int_M \|S\|_F^{2p}\, \mathrm{d}A_f, \quad p \in \mathbb{N}. \tag{1}$$

When $p = 1$ this reduces to the celebrated Willmore energy [Joshi and Séquin 2007; Willmore 1965], however larger values of p create a stronger smoothing effect on regions with higher curvature (see Figure 1). Gruber et al. [2020; 2019] have also explored higher powers of mean curvature, achieving similar deformations. However, their method is confined to this very particular energy and is incompatible with intrinsic flows.

The second category: the intrinsic flows, is commonly used in the computer graphics community to “flatten" the surface not visually but from the intrinsic perspective of an observer “living" on the geometry. The Ricci flow, introduced by Hamilton [1988] and notably utilized in proving the Poincaré conjecture, falls within this category. Combinatorial counterparts of such flows [Chow and Luo 2003; Jin et al. 2008] have found successful applications in conformal mesh parametrization. In computer graphics, these flows naturally arise when computing texture atlases [Sheffer et al. 2007], cone parametrization for quad remeshing [Ben-Chen et al. 2008], or surface approximation using developable patches for manufacturing [Yuan et al. 2023]. Here, the focus is on modifying the *Gaussian curvature*, i.e., the determinant of the shape operator. As this information is independent of the immersion, computations often directly apply to the metric without explicit geometric realization [Springborn et al. 2008] — a focus of interest in this paper and illustrated by Figure 2. While an immersion can be computed from the deformed metric [Chern et al. 2018], this process results in the loss of the relationship between the deformation and the curvature, and thus, cannot be used for fairing flows.

Considering only conformal deformations significantly enhances numerical stability while maintaining the capacity to represent a large family of deformations. Prior research has highlighted the potential for curvature flows to induce extreme surface deformations

Author’s address: Etienne Corman, etienne.corman@cnrs.fr, Université de Lorraine, CNRS, Inria, LORIA, Nancy, 54000, France.

/2024/5-ART

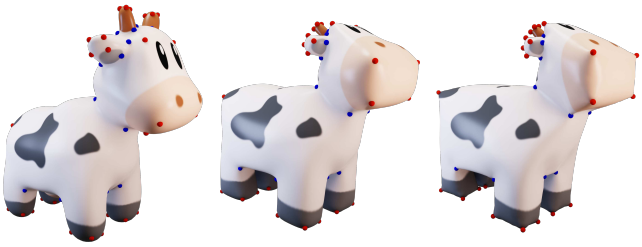


Fig. 2. Immersion of a combinatorial Ricci flow toward a cone metric. Blue spheres highlights $-\pi/2$ cones and red spheres $\pi/2$ cones.

and unwanted metric distortions [Crane et al. 2011, 2013b; Gruber and Aulisa 2020; Soliman et al. 2021], negatively impacting long term numerical stability [Bobenko and Schröder 2005]. Although adaptive remeshing can maintain mesh quality [de Goes et al. 2008], it does not always improve asymptotic stability. Thus, in this paper, we limit ourselves to *conformal* surface deformations, constraining the surface during the flow solely to local scaling and rotation in space.

Numerical stability in computing curvature-based energies poses a notorious challenge. Conventionally, ensuring a geometric realization of the flow involves formulating these energies in terms of vertex displacement or Jacobian matrices [Xu and Zhou 2009]. However, evaluating curvature, a second-order derivative of point positions, on a discrete mesh introduces considerable numerical instability. Further complicating matters, computing the energy gradient, crucial for the flow, necessitates intricate computations [Eigensatz and Pauly 2009; Eigensatz et al. 2008; Gruber and Aulisa 2020; Soliman et al. 2021], sometimes even prior to any discretization.

In contrast, we change variables to directly operate with scale factors and shape operators. These new variables offer two significant advantages: improving long term stability and simplifying manipulations of curvature-based energies. Having access to the complete curvature tensor enables us to handle a wide range of energies, including fairing *and* intrinsic energies. However, a critical question remains: under what conditions can the curvature tensor and the scale factor be integrated to reconstruct the point positions defining a conformal deformation? We answer this question by introducing a previously unknown necessary and sufficient integrability condition for topological spheres. This condition establishes a crucial link between the derivative of local rotations and our variables, i.e., the scale factor and shape operator. We complete the picture with a characterization of flow integrability applicable to surfaces of arbitrary genus and point position constraints.

To concretize these findings, we employ Discrete Exterior Calculus [Hirani 2003] for discretizing these equations while the discrete shape operator is approximated using Regge finite elements [Li 2018; Regge 1961]. Overall, we propose a general algorithm designed for two primary objectives: firstly, the minimization of fairing energies, accommodating high-power norms and vertex position constraints. Secondly, the optimization of a combinatorial Ricci energy [Chow and Luo 2003], achieving a predefined Gaussian curvature density.

Spin transformation. There exist significant connections between our theory and the spin transformations explored by Kamberov et al. [1998] and later applied to geometry processing [Chern et al. 2015; Crane et al. 2011, 2013b]. The integrability condition introduced by Crane et al. [2011] relies solely on the change in mean curvature. In contrast, ours explicitly exposes the change in the *entire* curvature tensor, thereby enabling a broader range of applications. Despite their apparent differences, both of these integrability conditions are equivalent and are sufficient for defining a conformal deformation. Appendix D provides an in-depth discussion of this matter.

2 BACKGROUND ON ROTATION FIELDS

As we will extensively use rotation fields, we provide a brief overview of the definitions and properties crucial for this paper. Throughout we use $\mathrm{SO}(3)$ to denote the set of 3×3 rotation matrices satisfying $R^\top R = RR^\top = I$, $\det(R) > 0$, where I is the identity. We use $\mathfrak{so}(3)$ to denote the set of 3×3 skew-symmetric matrices $\omega^\top = -\omega$, whose non-zero components describe the axis and magnitude of a rotation. The corresponding rotation matrix is obtained through the *exponential map* $\exp : \mathfrak{so}(3) \to \mathrm{SO}(3)$.

For convenience, we introduce the notation $\hat{v} \in \mathfrak{so}(3)$ to designate the skew-symmetric matrix defined by the vector $v = (v_1, v_2, v_3)^\top \in \mathbb{R}^3$ as follows:

$$\hat{v} := \begin{bmatrix} 0 & v_3 & -v_2 \\ -v_3 & 0 & v_1 \\ v_2 & -v_1 & 0 \end{bmatrix}.$$

Conversely, a skew-symmetric matrix $\omega \in \mathfrak{so}(3)$ corresponds to the axis-angle vector of $\vec{\omega} \in \mathbb{R}^3$

$$\vec{\omega} := \begin{bmatrix} \omega_{23} \\ \omega_{31} \\ \omega_{12} \end{bmatrix}.$$

The exponential map can then be evaluated using *Rodrigues' formula*

$$\exp(\hat{v}) = I + \sin(|v|)\frac{\hat{v}}{|v|} + (1-\cos(|v|))\frac{\hat{v}}{|v|}^2,$$

representing the rotation of axis v and angle $|v|$. Notably, the exponential map is not one-to-one: as the angle increases, exp will return to the same rotation many times.

Rotation fields. A rotation field $R : M \subset \mathbb{R}^3 \to \mathrm{SO}(3)$ assigns rotation matrices to each point of the domain M. We define the Darboux derivative of R as the skew-symmetric-valued differential 1-form $\omega : TM \to \mathfrak{so}(3)$ such that

$$\mathrm{d}R = \omega R. \tag{2}$$

The Darboux derivative is closely tied to the exponential map. If ω is constant with respect to evaluation against the tangent to the curve along a path $\gamma : \mathbb{R} \to M$, integrating Eq. (2) along this path yields

$$R_{\gamma(t)} = \exp\left(t\omega_{\gamma(0)}(\gamma'(0))\right) R_{\gamma(0)}.$$

Intuitively, $\omega(X) \in \mathfrak{so}(3)$ represents the axis-angle rotation speed of R in the direction $X \in TM$. Importantly, not every $\mathfrak{so}(3)$-valued 1-form ω is a Darboux derivative of some rotation field. For a more detailed explanation, we refer to Corman and Crane [2019, App. A.2].

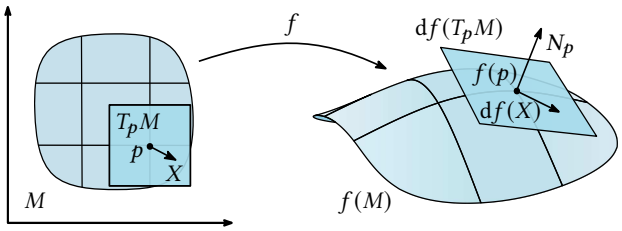


Fig. 3. We represent the geometry of a surface in $\mathbb{R}^3$ using a map f that associates each point on a surface M with its coordinates in Euclidean space.

3 CONFORMAL SURFACE DEFORMATION

3.1 Notations

Let M be a 2-manifold immersed in $\mathbb{R}^3$ through the mapping $f : M \to \mathbb{R}^3$. We denote $N : M \to \mathbb{R}^3$ the unit normal to the surface as in Figure 3. The metric induced by the immersion f is denoted by $\langle ., . \rangle_f$, and $\mathrm{d}A_f$ represents its associated area element. The reference to the immersion is sometimes omitted.

The *shape operator* $S : TM \to TM$ is the linear map defined as the covariant derivative of the normal

$$S_p(X) \coloneqq -\nabla_X N, \quad X \in T_pM.$$

As demonstrated by Do Carmo and Flaherty [1992, p. 128], this linear map is *symmetric* with respect to the metric induced by the immersion, i.e., $\langle S(X), Y \rangle_f = \langle X, S(Y) \rangle_f$ for all $X, Y \in TM$. The Gaussian curvature K_p at point $p \in M$ is the determinant of S_p, and the mean curvature is the half of its trace, $H_p = \mathrm{tr} S_p / 2$.

3.2 Conformal Deformation and Curvature

A deformation of the surface is simply the definition of a new immersion $\tilde{f} : M \to \mathbb{R}^3$. Any *conformal* deformation can be defined by a log-scale factor $u : M \to \mathbb{R}$ and a rotation field $R : M \to \mathrm{SO}(3)$ as follows:

$$\mathrm{d}\tilde{f} = e^u R^\top \mathrm{d}f. \tag{3}$$

As proved in Appendix A, it is necessary that the Darboux derivative ω of the rotation field R satisfies the following integrability condition:

$$\vec{\omega}(X) = - * \mathrm{d}u(X) N - N \times \mathrm{d}f(\tau(X)), \tag{4}$$

where $\tau : TM \to TM$ is a *symmetric* linear map from the tangent bundle to itself.

The quantities u and τ have a direct geometrical meaning. Indeed, the variations of the rotation field are responsible for the change in variations of the surface normals, so the Darboux derivative of the rotation field must be related to a change in curvature.

Theorem 3.1. *If the immersion $\tilde{f}$ is conformal; then, the Darboux derivative ω of the rotation field R is uniquely defined by Eq.* (4) *where u and τ represent, respectively, the change in Gaussian curvature and the change in the shape operator:*

$$\left| \begin{array}{l} \Delta u = K - e^{2u} \tilde{K}, \\ \tau = S - e^u \tilde{S}. \end{array} \right.$$

In Theorem 3.1, the relation between the log-scale factor and the Gaussian curvature is ubiquitous in conformal parametrization [Ben-Chen et al. 2008; Springborn et al. 2008]. The change in shape operator, however, appears to be novel.

Most importantly, the pair u, τ contains enough information to reconstruct the deformation $\tilde{f}$. Indeed, the Darboux derivative ω is recovered from the pair u, τ through Eq. (4), which, in turn, is sufficient to compute the rotation field R as a solution of Eq. (2). Finally, the new immersion is derived from u and R by solving Eq. (3). Therefore, we can (and will) use u and τ as our sole variables when computing conformal curvature flows.

However, not all pairs u, τ define a valid conformal deformation. The integrability condition in Eq. (4), underlying a linear relation between ω and the pair u, τ, relies on the fact that ω is the Darboux derivative of some rotation field which is not guaranteed by the integrability condition.

Theorem 3.2. *If M is simply connected, the vector-valued 1-form $e^u R^\top \mathrm{d}f$ is exact if and only if ω satisfies Eq.* (4) and *Eq.* (2).

As a consequence, u and τ are interdependent in a nonlinear way. We could compute a conformal deformation by directly tackling the nonlinear system of equations of Theorem. 3.2. However, we would be limited to surfaces with trivial topology. Instead, we prefer to follow the path of Crane et al. [2013b], and compute a surface flow that effectively linearizes these equations. This enables us to deal with surfaces of arbitrary genus.

3.3 Surface Flow

Our objective is to compute a conformal deformation minimizing an energy $\mathcal{E}$ that depends solely on u and τ:

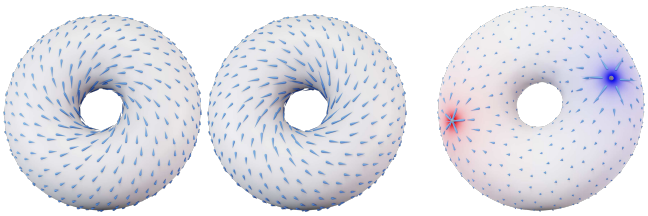
$$\begin{array}{ll} \min & \mathcal{E}(u, \tau) \\ f : M \to \tilde{M} & \text{s.t.} \quad f \text{ conformal} \end{array}$$

For stability and efficiency, we change our variables to the log-scale factor u and the change in curvature τ. Recall that this algorithm is possible because u and τ are sufficient to define a conformal deformation (cf. Sec. 3.2).

Following Crane et al. [2013b], we solve this optimization problem by generating a sequence of immersions $f_t : M \to \mathbb{R}^3$, starting from $f_0 = f$, that converges to the minimum of the optimization problem as t increases. The derivative at time 0 of a quantity u and τ will be noted $\dot{u}$ and $\dot{\tau}$. At time 0, there is no deformation; hence, $u = 0$, $\tau = 0$ and R is uniformly equal to the identity. Similar to the Newton method for optimization, we compute a descent direction $\dot{u}, \dot{\tau}$ by minimizing a quadratic approximation $Q_{\mathcal{E}}(u, \tau)$ of the energy $\mathcal{E}(u, \tau)$ under the flow integrability constraints:

$$\begin{array}{ll} \min & Q_{\mathcal{E}}(\dot{u}, \dot{\tau}) \\ \dot{u} : M \to \mathbb{R} & \text{s.t.} \quad \text{flow constraints (Sec. 3.3).} \\ \dot{\tau} : TM \to TM & \end{array}$$

The flow integrability constraints are derived by taking the time derivative of the integrability conditions outlined in Theorem 3.2. Additionally, for immersions of non-simply connected surfaces and when considering position constraints, specific constraints are necessary. Both cases can be formulated as the preservation of integrals.



(a) (b)

Fig. 4. The flow must preserve the integral over the manifold of (a) harmonic vector fields, and (b) the gradient of g solution of Poisson's equation, so that the relative position of points $p, q \in M$ remains constant.

3.3.1 Darboux Derivative Integrability. From the definition of ω (Eq. (2)) and taking its time derivative at 0:

$$\mathrm{d}\dot{R} = \dot{\omega}R + \omega\dot{R} = \dot{\omega}.$$

Since ω is a linear function of u, τ, its time derivative is $\dot{\omega}(u, \tau) = \omega(\dot{u}, \dot{\tau})$. Thus, $\omega(\dot{u}, \dot{\tau})$ is an exact form, implying that it is both closed

$$\boxed{\mathrm{d}\vec{\omega}(\dot{u}, \dot{\tau}) = 0,} \tag{5}$$

and orthogonal to $\eta_i, i = 1, \ldots, h$ — a basis of harmonic 1-forms

$$\boxed{\int_M \langle \vec{\omega}(\dot{u}, \dot{\tau}), \eta_i \rangle = 0, \quad \forall i \in \{1, \ldots, h\}.} \tag{6}$$

These two conditions ensure that $\omega(\dot{u}, \dot{\tau})$ is still a derivative of a rotation field.

3.3.2 Integral Preservation. As demonstrated in Appendix B, the flow must preserve the integrals over the manifold of all harmonic vector fields (Fig. 4a). Furthermore, it is also possible to constrain the relative positions of points $p, q \in M$ by ensuring that the flow to preserve the integral of gradient of the function g (Fig. 4b), where g is the solution of the equation $\Delta g = \delta_p - \delta_q$.

Let $v : M \to TM$ be the tangent vector field representing either a harmonic vector field or the gradient of the function g. Our aim is to ensure the preservation of the integral $\int_M \mathrm{d}f_t(v)\, \mathrm{d}A_{f_0}$ throughout the flow. Consider a $\mathbb{R}^3$-valued 1-form β that solves

$$\Delta_1 \beta = *\mathrm{d}\left(\mathrm{d}f(v)\right).$$

where $\Delta_1 := \mathrm{d}\delta + \delta\mathrm{d}$ is the Hodge Laplacian of 1-form, and $\delta := *\mathrm{d}*$ the co-differential [Morita 2001, p. 155]. As demonstrated in Appendix B, $\dot{u}$ and $\dot{\tau}$ must satisfy:

$$\boxed{\int_M \dot{u}\mathrm{d}f(v)\, \mathrm{d}A_f + \int_M \omega(\dot{u}, \dot{\tau}) \wedge \beta = 0.} \tag{7}$$

4 DISCRETIZATION

In practice, our geometrical flow is performed on a manifold oriented triangle mesh $\mathcal{M} = (V, E, F)$ *without* boundaries. Here, $i \in V$ represents vertices, $ij \in E$ denotes edges, and $ijk \in F$ signifies triangles. The immersion is represented by a vector $f_i \in \mathbb{R}^3$ per vertex $i \in V$, and edge vectors are succinctly noted as $e_{ij} = f_i - f_j$. We assume that the triangle areas, given by $A_{ijk} := e_{ij} \times e_{ik}$, are strictly positive. Consequently, any inner angle α_i^{jk} at a vertex $i \in V$ within the triangle $ijk \in F$ is strictly bounded between 0 and π, and each triangle is equipped with an outward unit normal $N_{ijk} \in \mathbb{R}^3$.

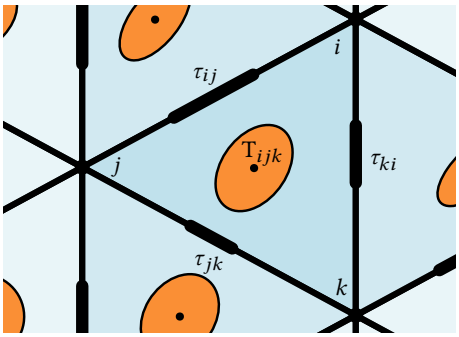
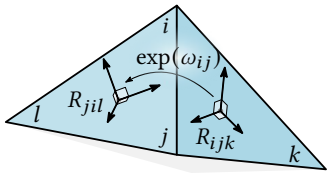


Fig. 5. The symmetric matrix T_{ijk} (depicted with an orange ellipse) is interpolated from the values $\tau \in \mathbb{R}^{|E|}$ assigned to edges (represented by the bold line lengths). The value τ_{ij} corresponds to the tangent-tangent component of the matrix $\langle t_{ij}, T_{ijk} t_{ij} \rangle$.

Our discretization relies on using Regge finite elements [Li 2018; Regge 1961] to approximate the symmetric tensor τ. In their lowest order, Regge elements assign a value $\tau \in \mathbb{R}^{|E|}$ per edge, which is interpolated into a constant symmetric matrix $\mathrm{T}_{ijk} \in \mathrm{Sym}_2(\mathbb{R})$ (see Fig. 5). Here, $\mathrm{Sym}_2(\mathbb{R})$ represents the set of symmetric matrices expressed in local triangle coordinates. Further details on Regge finite elements can be found in Section 4.1.

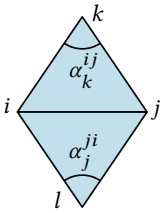
We define the log-scale factor $u \in \mathbb{R}^{|V|}$ as a value per vertex. The rotation field R_{ijk} is an assignment of rotation matrix per triangle $ijk \in F$, and its Darboux derivative $\omega_{ij} \in \mathfrak{so}(3)$ is defined per *oriented dual edge* ij, as in [Corman and Crane 2019]. Thus, $\vec{\omega}_{ij}$ is the axis-angle representation of the rotation between R_{ijk} and R_{jil}, namely:



$$R_{jil} = \exp\left(\omega_{ij}\right) R_{ijk}. \tag{8}$$

We also define the discrete Laplacian Δ as the weighted graph Laplacian $W \in \mathbb{R}^{|V| \times |V|}$, where

$$(Wf)_i = \sum_{ij \in E} \underbrace{\tfrac{1}{2}\left(\cot \alpha_k^{ij} + \cot \alpha_l^{ji}\right)}_{:= w_{ij}} (f_j - f_i)$$



for each vertex $i \in V$, where k, l denote the vertices opposite to edge ij. The *cotan-weights* w_{ij} simply account for the shape of the triangles (see Crane et al. [2013a, Chapter 6]).

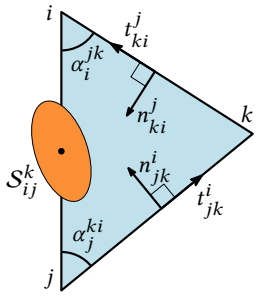
The discretization of the main integrability condition (Eq. (4)) is discussed in Section 4.2, and the discrete flow integrability conditions are detailed in Section 4.3.

4.1 Regge Finite Element

In this section, we present a succinct overview of low-order Regge finite elements [Li 2018], and their application to approximate the shape operator on a triangle mesh.

We denote $t_{ij}^k \in \mathbb{R}^2$ as the unit edge vector expressed in an arbitrary local basis of triangle ijk and $n_{ij}^k \coloneqq Jt_{ij}^k \in \mathbb{R}^2$ as the unit edge normal obtained by a 90° counter-clockwise rotation of the edge vector around the triangle normal.

Regge elements offer a low-order approximation of symmetric matrix fields on triangle meshes. This field assigns a symmetric matrix $\mathrm{T}_{ijk} \in \mathrm{Sym}_2(\mathbb{R})$ to each triangle ijk, which is constant on that triangle. While the field is globally discontinuous, its tangent-tangent component (i.e., $\langle t_{ij}^k, \mathrm{T}_{ijk} t_{ij}^k \rangle$) remains continuous across the mesh. Regge elements are uniquely described by a single degree of freedom (dof) τ_{ij} per edge, representing this continuous component.



The Regge basis function $\mathcal{S}_{ij}^k$ is nonzero only within the triangle ijk and interpolates the dof on edge ij within ijk. As such, it is the unique symmetric matrix on triangle $ijk \in F$ such that its tangent-tangent components are equal to one on edge ij and to zero on edges jk and ki:

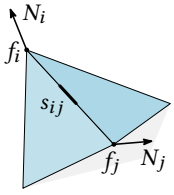
$$\mathcal{S}_{ij}^k \coloneqq -\frac{(n_{jk}^i)(n_{ki}^j)^\top + (n_{ki}^j)(n_{jk}^i)^\top}{2 \sin \alpha_i^{jk} \sin \alpha_j^{ki}}.$$

With these bases, the dofs $\tau \in \mathbb{R}^{|E|}$ are interpolated to a constant symmetric matrix $\mathrm{T}_{ijk} \in \mathrm{Sym}_2(\mathbb{R})$ inside triangle $ijk \in F$ by the linear combination:

$$\mathrm{T}_{ijk} = \tau_{ij}\mathcal{S}_{ij}^k + \tau_{jk}\mathcal{S}_{jk}^i + \tau_{ki}\mathcal{S}_{ki}^j.$$

The Regge finite elements basis is entirely intrinsic to the surface, hence, many equations can be expressed as trigonometric functions of the triangle inner angles α.

Curvature tensor estimation. To approximate the shape operator $S(t) \coloneqq -\nabla_t N$, we must evaluate its tangent-tangent component at each edge $ij \in E$. We define a vertex normal $N_i \in \mathbb{R}^3$ at vertex $i \in V$ as the average of adjacent normals weighted by triangle area. The dof s_{ij} is then computed by approximating the covariant derivative by finite differences:



$$s_{ij} = \frac{1}{|f_i - f_j|^2} \left\langle N_i - N_j, f_i - f_j \right\rangle.$$

The symmetric curvature tensor, denoted as S_{ijk}, is interpolated across any triangle $ijk \in F$ by a linear combination of the Regge bases. Figure 6 illustrates the approximation capabilities of Regge

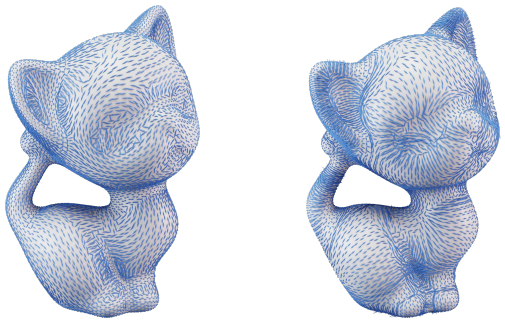
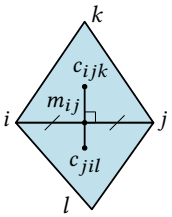


Fig. 6. Smallest (left) and largest (right) principal curvature directions obtained by eigen-decomposition of the discrete shape operator approximated by Regge finite elements.

basis by computing the principal curvature directions at each face of a triangle mesh.

4.2 Discrete Infinitesimal Rotation

We have now all the tools to discretize the conformal Darboux derivative presented in Eq. (4). Drawing inspiration from the Discrete Exterior Calculus (DEC) framework [Hirani 2003], we represent the discrete dual 1-form ω_{ij} as the integral of the continuous 1-form along the dual path connecting the circumcenters c_{ijk}, c_{jil} of two neighboring triangles $ijk, jil \in F$. Given that the gradient of the log-scale factor ∇u and the symmetric tensor T remain constant per triangle, we can decompose this integral into two parts: the first extending from the circumcenter c_{ijk} to the edge mid point m_{ij} in the direction n_{ij}^k and the second from m_{ij} to c_{jil} along $-n_{ji}^l$. This decomposition yields:



$$\vec{\omega}_{ij} = \vec{\omega}_{ij}^k - \vec{\omega}_{ji}^l, \tag{9}$$

where $\vec{\omega}_{ij}^k = \int_{c_{ijk}}^{m_{ij}} \vec{\omega}(n_{ij}^k)$. The distance from c_{ijk} to m_{ij} is $|e_{ij}| \cot \alpha_k^{ij}/2$, resulting in the expression for the half-rotation:

$$\vec{\omega}_{ij}^k = \frac{1}{2}|e_{ij}| \cot \alpha_k^{ij} \vec{\omega}(n_{ij}^k). \tag{10}$$

The first part of Eq. (4) relies solely on the log-scale factor differential $\mathrm{d}u$, and is evaluated in the direction of the edge normal n_{ij}^k using the property of the Hodge star on the 1-form (i.e., $*\mathrm{d}u(n) = \mathrm{d}u(Jn)$):

$$* \,\mathrm{d}u(n_{ij}^k) N_{ijk} = -\mathrm{d}u(t_{ij}^k) N_{ijk} = \frac{u_i - u_j}{|f_i - f_j|} N_{ijk}. \tag{11}$$

The second part of Eq. (4) (i.e., $n \times \mathrm{d}f(\tau)$) depends on the symmetric tensor τ and requires a specific treatment. The matrix T_{ijk} applied to the direction n_{ij}^k can be decomposed in the basis (t_{ij}^k, n_{ij}^k) in triangle ijk. This decomposition reads as $\mathrm{T}_{ijk} n_{ij}^k = t_{ij}^k \langle t_{ij}^k, \mathrm{T}_{ijk} n_{ij}^k \rangle + n_{ij}^k \langle n_{ij}^k, \mathrm{T}_{ijk} n_{ij}^k \rangle$. The differential of the immersion $\mathrm{d}f$ transforms tangent vectors into vectors immersed in $\mathbb{R}^3$, so $\mathrm{d}f(t_{ij}^k) = e_{ij}/|e_{ij}|$

and $\mathrm{d}f(n_{ij}^k) = N_{ijk} \times e_{ij}/|e_{ij}|$. This leads to the discrete formulation:

$$\begin{aligned} N_{ijk} \times \mathrm{d}f(\tau(n_{ij}^k)) &= N_{ijk} \times \tfrac{e_{ij}}{|e_{ij}|} \langle t_{ij}^k, \mathrm{T}_{ijk} n_{ij}^k \rangle \\ &\quad - \tfrac{e_{ij}}{|e_{ij}|} \langle n_{ij}^k, \mathrm{T}_{ijk} n_{ij}^k \rangle. \end{aligned} \tag{12}$$

Putting together Eq. (10), Eq. (11), Eq. (12) and the decomposition of T_{ijk} into the the Regge basis, we obtain the formula for the discrete half rotation:

$$\begin{aligned} \vec{\omega}_{ij}^k &= -\tfrac{1}{2} \cot \alpha_k^{ij} N_{ijk} \left(u_i - u_j\right) \\ &\quad + \tfrac{1}{2} \cot \alpha_k^{ij} \textstyle\sum_{abc \in \Pi_{ijk}} \tau_{ab} e_{ij} \times N_{ijk} \langle t_{ij}^k, \mathcal{S}_{ab}^c n_{ij}^k \rangle \\ &\quad + \tfrac{1}{2} \cot \alpha_k^{ij} \textstyle\sum_{abc \in \Pi_{ijk}} \tau_{ab} e_{ij} \langle n_{ij}^k, \mathcal{S}_{ab}^c n_{ij}^k \rangle \end{aligned}$$

Here, $\Pi_{ijk} = \{ijk, jki, kij\}$ is the set of circular permutations of indices ijk, and the terms $\langle t_{ij}^k, \mathcal{S}_{ab}^c n_{ij}^k \rangle$ and $\langle n_{ij}^k, \mathcal{S}_{ab}^c n_{ij}^k \rangle$ are functions of the inner angles α:

$$\begin{aligned} \langle t_{ij}^k, \mathcal{S}_{ab}^c n_{ij}^k \rangle &= (2\delta_{ak} - 1) \tfrac{1}{2} \cot \alpha_a^{bc} - (2\delta_{aj} - 1) \tfrac{1}{2} \cot \alpha_b^{ca}, \\ \langle n_{ij}^k, \mathcal{S}_{ab}^c n_{ij}^k \rangle &= 1 - \delta_{ai} - \cot \alpha_a^{bc} \cot \alpha_b^{ca}. \end{aligned}$$

where δ_{ab} is the Kronecker delta, equal to 1 if $a = b$ and 0 otherwise.

4.3 Discrete Integrability Conditions

Using our discrete Darboux derivative, we are ready to derive the discretization of the integrability conditions for the surface flow exposed in Section 3.3. There are two constraints to consider: i) the integrability of $\dot{\omega}$ into a rotation field (Eqs. (5) and (6)) and ii) the non-simply connected constraints and position constraints both covered by Eq. (7).

4.3.1 Exactness. The discrete Darboux derivative ω is a dual 1-form in the sense of DEC [Hirani 2003]. As explained in Section 3.3, it is imperative to ensure that it is exact, or equivalently closed and orthogonal to all harmonic 1-forms.

First, the discrete exterior derivative of ω must vanish (Eq. (5)), meaning that the sum of ω_{ij} on any dual cycles centered on vertex $i \in V$ must be zero:

$$(\mathrm{d}\omega)_i = \sum_{ij \in E} \omega_{ij} = 0. \tag{13}$$

Second, ω must be orthogonal to all discrete harmonic 1-forms. In DEC a harmonic *primal* 1-form $\eta : E \to \mathbb{R}$ is a value per edge whose differential and co-differential are zero [Desbrun et al. 2006]. There are exactly $2g := 2 - |V| + |E| - |F|$ independent harmonic 1-forms on a mesh without boundaries, and a basis $\{\eta^i\}, i = 1, \dots 2g$ can be computed using the Gram-Schmidt process proposed by Crane et al. [2013b, Appendix C.1]. Using the primal-dual scalar product defined by Desbrun et al. [2005], we obtain the discrete integrability condition:

$$\sum_{ij \in E} \eta_{ij}^a \omega_{ij} = 0, \quad \forall a \in \{1, \dots, 2g\}. \tag{14}$$

4.3.2 Integral Preservation. We ensure the preservation of the integral of a face-based vector field $v : F \to \mathbb{R}^3$ by discretizing Eq. (7). We define the differential form $\beta_{ij} \in \mathbb{R}^3$ to be a vector-valued *primal* 1-form, assigning a vector at each oriented edge $ij \in E$. By definition, this form is a solution of three linear systems (one per coordinate)

$$\Delta_1 \beta = \mathrm{d}_1^\top v,$$

where the Hodge Laplacian $\Delta_1 \in \mathbb{R}^{|E| \times |E|}$ is defined by Fisher et al. [2007, p. 3], and $\mathrm{d}_1 \in \mathbb{R}^{|F| \times |E|}$ is the exterior derivative of a dual 1-form which sums the edge values adjacent to a triangle $(\mathrm{d}_1 \eta)_{ijk} = \eta_{ij} + \eta_{jk} + \eta_{ki}$.

The first term $\int_M \dot{u} \mathrm{d}f(v) \, \mathrm{d}A_f$ in Eq. (7) is discretized by integrating the piece-wise constant vector, multiplied by the piecewise linear log-scale factor $u : V \to \mathbb{R}$. We immediately obtain the weighted sum: $\frac{1}{3} \sum_{ijk \in F} A_{ijk} v_{ijk} (u_i + u_j + u_k)$.

The discretization of the second term, i.e., $\omega \wedge \beta$, is obtained using the discrete wedge product between a dual and a primal 1-form defined by Hirani et al. [2003] as the scalar product of the two vectors.

Assembling these two pieces together, the discretization of Eq. (7) reads:

$$\frac{1}{3} \sum_{ijk \in F} A_{ijk} v_{ijk} (u_i + u_j + u_k) + \sum_{ij \in E} \vec{\omega}_{ij} \times \beta_{ij} = 0. \tag{15}$$

Non-simply connected surfaces. To preserve non-simply connected surfaces during the flow, it is necessary to compute a basis for harmonic vector fields $v : F \to \mathbb{R}^3$ whose integrals over the mesh remain constant, as detailed in Appendix B.1. This involves "converting" the harmonic 1-form basis $\{\eta^i\}, i = 1, \dots, 2g$, defined in Section 4.3.1, into vector fields using the algorithm proposed by Crane et al. [2013b, App. C.1].

Position constraints. Suppose we wish to constrain the flow to preserve the position of the vertices $i_a \in V$ for $a = 1, \dots, n$. Let δ_i^a be the discrete Dirac function, which is equal to one at vertex $i = i_a$ and zero otherwise. We consider all the $n(n-1)/2$ pairs of distinct constrained vertices to build a the family of functions $\{g^{ab}\}, a, b \in \{1, \dots, n\}$, solutions of Poisson's equation

$$W g^{ab} = \delta^a - \delta^b.$$

As explained in Appendix B.2, the vector fields $v^{ab} = \nabla g^{ab}$, where ∇ is the piecewise linear gradient as defined in [Botsch et al. 2010, p. 44], should satisfy Eq. (15).

5 ALGORITHM

Now that we have discretized our integrability conditions, we are prepared to elaborate on our discrete algorithm for conformal flow, as outlined in Algorithm 1. Each iteration involves three steps: i) an optimization step determines a descent direction by solving a quadratic programming problem (Sec. 5.1), ii) the rotation field R is computed from its Darboux derivative ω by solving a linear least-squares problem (Sec. 5.2) and iii) an as-conformal-as possible deformation is obtained by solving Eq. (3) in a least-squares sense (Sec. 5.3).

Algorithm 1 SurfaceFlow$(M, f, \xi_{\max})$

Input: A manifold, orientable triangle mesh $M = (V, E, F)$ with vertex coordinates $f : V \to \mathbb{R}^3$, a step size limitation angle $\xi_{\max}$.

Output: Deformed vertex positions $\tilde{f}$.

1: $\rho \leftarrow 10^{-4}$
2: **for** $i = 1 \dots$ itmax **do** ▷ *Optimization loop*

Evaluate energy $\mathcal{E}$ for vertex coordinates f ▷Sec. 6
Compute gradient $G_{\mathcal{E}}$ and Hessian $H_{\mathcal{E}}$ wrt u, τ ▷Sec. 6
Augment the Hessian: $H \leftarrow H_{\mathcal{E}} + \rho|G_{\mathcal{E}}|I$ ▷Sec. 5.1
Build integrability constraint matrix C ▷Eqs. (13), (14), (15)
$u, \tau \leftarrow \arg\min_{C\left[\begin{smallmatrix}u\\ \tau\end{smallmatrix}\right]=0} \frac{1}{2}\left[\begin{smallmatrix}u\\ \tau\end{smallmatrix}\right]^\top H\left[\begin{smallmatrix}u\\ \tau\end{smallmatrix}\right] + G_{\mathcal{E}}^\top\left[\begin{smallmatrix}u\\ \tau\end{smallmatrix}\right]$ ▷Sec. 5.1
Compute ω from u, τ ▷Eq. (9)
Rescale ω so that $|\omega_{ij}| \leq \xi_{\max}$ for all $ij \in E$ ▷Sec. 5.1
Compute rotation field R from ω ▷Sec. 5.2
Find new vertex position $\tilde{f}$ from R and u ▷Sec. 5.3
Optional: Update ρ with backtracking ▷Alg. 2
return $\tilde{f}$

5.1 Optimization

Each iteration begins by determining a suitable descent direction that satisfies the integrability constraints exposed in Section 4.3 while reducing the objective function $\mathcal{E} : (u, \tau) \to \mathbb{R}$. To achieve this, the energy is approximated by a quadratic function based on its second-order Taylor expansion at the point $u = 0_{\mathbb{R}^{|V|}}, \tau = 0_{\mathbb{R}^{|E|}}$, expressed as:

$$\mathcal{E}(u, \tau) \approx \mathcal{E}(0, 0) + G_{\mathcal{E}}^\top\left[\begin{smallmatrix}u\\ \tau\end{smallmatrix}\right] + \frac{1}{2}\left[\begin{smallmatrix}u\\ \tau\end{smallmatrix}\right]^\top H_{\mathcal{E}}\left[\begin{smallmatrix}u\\ \tau\end{smallmatrix}\right].$$

Here, $G_{\mathcal{E}} \in \mathbb{R}^{|V|+|E|}$ represents the gradient and $H_{\mathcal{E}} \in \mathbb{R}^{|V|+|E|\times|V|+|E|}$ denotes the Hessian matrix of $\mathcal{E}$ with respect to u and τ. Examples of such objectives and approximations are provided in Section 6.

To improve stability, the Hessian is augmented by the diagonal mass matrix $I = \begin{bmatrix} M & 0_{\mathbb{R}^{|V|\times|E|}} \\ 0_{\mathbb{R}^{|E|\times|V|}} & Q \end{bmatrix} \in \mathbb{R}^{|V|+|E|\times|V|+|E|}$, where $M \in \mathbb{R}^{|V|\times|V|}$ and $Q \in \mathbb{R}^{|E|\times|E|}$ are the respective lumped mass matrices of linear and Regge finite elements:

$$M_{ii} = \frac{1}{3}\sum_{ijk\in F} A_{ijk}, \quad Q_{ij,ij} = \frac{1}{3}\left(A_{ijk} + A_{jil}\right).$$

Overall, at each iteration, we solve the quadratic programming problem:

$$\begin{aligned} \min_{\substack{u:V\to\mathbb{R}\\ \tau:E\to\mathbb{R}}} \quad & \mathcal{E}(0,0) + G_{\mathcal{E}}^\top\left[\begin{smallmatrix}u\\ \tau\end{smallmatrix}\right] + \tfrac{1}{2}\left[\begin{smallmatrix}u\\ \tau\end{smallmatrix}\right]^\top [H_{\mathcal{E}} + \alpha I]\left[\begin{smallmatrix}u\\ \tau\end{smallmatrix}\right] \\ \text{subect to:} & \\ & \textstyle\sum_{ij\in E}\omega_{ij}(u,\tau) = 0, && \forall i \in V \\ & \textstyle\sum_{ij\in E}\eta_{ij}^a\omega_{ij}(u,\tau) = 0, && \forall a = 1,\dots,2g \\ & \textstyle\sum_{i\in V} u_i \sum_{ijk\in F}\frac{A_{ijk}}{3}v_{ijk}^b + \sum_{ij\in E}\omega_{ij}(u,\tau)\beta_{ij}^b = 0, && \forall b = 1,\dots,n \end{aligned}$$

The parameter $\alpha \in \mathbb{R}$ is selected to be proportional to the norm of the gradient, ensuring adaptability at each step:

$$\alpha = \rho\left(G_{\mathcal{E}}^\top I^{-1} G_{\mathcal{E}}\right)^{\frac{1}{2}},$$

with $\rho = 10^{-4}$.

Backtracking. Optionally, the value of ρ can be updated through the backtracking procedure in Alg. 2. This adaptation is useful for the strongly non-convex energies outlined in Sections 6.1 and 7.4. If the objective $\tilde{\mathcal{E}}$ computed on the deformed mesh $\tilde{f}$ is greater than the previous state $\mathcal{E}$, it indicates that an oversized step has failed to decrease the energy. Hence, we discard $\tilde{f}$ and compute a new descent direction, which is closer to a gradient descent, by increasing ρ by of factor 2. This procedure is repeated until an immersion $\tilde{f}$ is found that decreases the objective. In such a case, the deformation replaces the initial mesh (i.e., $f \leftarrow \tilde{f}$) and ρ is decreased by of factor 0.9 to encourage large steps. Algorithm 2 provides a pseudo-code of this algorithm.

Algorithm 2 BACKTRACKING$(M, f, \tilde{f}, \rho)$

Input: A manifold, orientable triangle mesh $M = (V, E, F)$ with vertex coordinates $f : V \to \mathbb{R}^3$, and deformed vertex positions $\tilde{f} : V \to \mathbb{R}^3$. The current penalty $\rho \in \mathbb{R}$.
Output: New vertex positions $\bar{f} : V \to \mathbb{R}^3$ and updated penalty $\rho \in \mathbb{R}$.

Evaluate old objective value $\mathcal{E}$ from f
Evaluate new objective value $\tilde{\mathcal{E}}$ from $\tilde{f}$
if $\tilde{\mathcal{E}} < \mathcal{E}$ **then** ▷If objective decreases
  $\bar{f} \leftarrow \tilde{f}$ ▷Keep new positions
  $\rho \leftarrow \text{MAX}(0.9\rho, 10^{-4})$ ▷Increase step size
else ▷If objective increases
  $\bar{f} \leftarrow f$ ▷Keep old positions
  $\rho \leftarrow 2\rho$ ▷Decrease step size
return $\bar{f}, \rho$

Step size limitation. To prevent numerical instability, we enforce an upper bound on the rotation speed of the field R. Recall that for an edge $ij \in E$, $|\vec{\omega}_{ij}|$ represents the angle of rotation between R_{ijk} and R_{jil}. To mitigate the risk of oversized steps, we rescale u and τ so that the maximum rotation angle over all edges, i.e., $\xi = \max_{ij\in E}|\vec{\omega}_{ij}|$, is smaller than a specified threshold $\xi_{\max}$. More precisely, we compute $t = \min\left(\frac{\xi_{\max}}{\xi}, 1\right)$ and update the variables $u \leftarrow tu$ and $\tau \leftarrow t\tau$. In practice, we set $\xi_{\max}$ to 25°, unless otherwise stated. While other strategies are discussed in Section 7.1, this straightforward step size limitation is often sufficient to ensure convergence under mesh refinement (see Fig. 12).

Convergence to a global minimum. The flow integrability constraints are a linearization of the *nonlinear* integrability conditions outlined in Theorem 3.1. Hence, the optimization problem, and the algorithm may converge to a local minimum.

5.2 Rotation Field

Given a pair u, τ defining a conformal deformation and a descent direction for the objective $\mathcal{E}$, the Darboux derivative ω_{ij} is computed at each edge ij using Eq. (9). These derivatives define the rotation field $R : F \to \text{SO}(3)$ solution of Eq. (8) for all edge $ij \in E$:

$$R_{jil} = \exp\left(\omega_{ij}\right) R_{ijk}.$$

Due to discretization errors, this system does not have a solution. Hence, we compute R in the least-squares sense, seeking the field that best accommodates the system for all edges. For efficiency, the rotation matrix R_{ijk} is represented by a unit quaternion $\lambda_{ijk} \in \mathbb{H}$ per triangle $ijk \in F$. The rotation matrix $\exp\left(\omega_{ij}\right)$ is converted to a unit quaternion using the quaternion exponential map $\exp_{\mathbb{H}}(v) := \cos|v| + \frac{v}{|v|}\sin|v| \in \mathbb{H}$ applied to half the vector $\vec{\omega}_{ij} \in \mathbb{R}^3 \equiv \text{Im}\mathbb{H}$. Thus, the quaternion field $\lambda : F \to \mathbb{H}$ representing $R : F \to \text{SO}(3)$

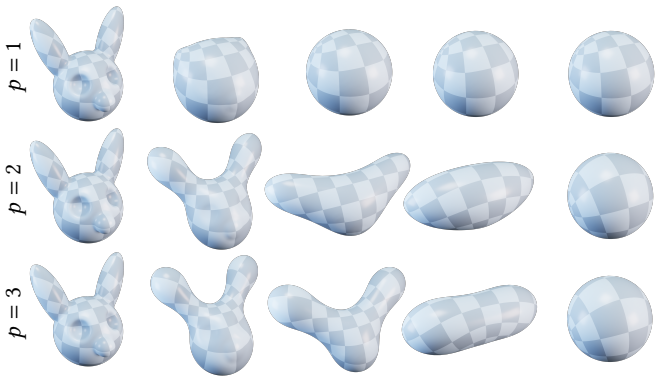


Fig. 7. Bending energy optimization for $p = 1$ (Willmore flow, top), 2 (middle) and 3 (bottom). It produces a stronger smoothing effect on high curvature regions for higher values of p.

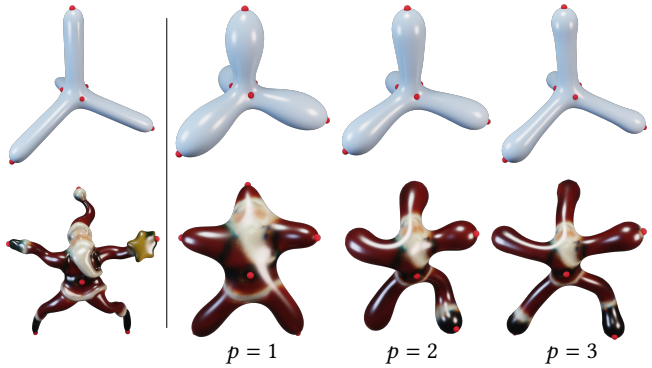


Fig. 8. Bending energy minima (Sec. 6.1) for p equal to 1 (aka Willmore flow), 2 and 3 with positional constraints highlighted in red. The santa is computed with $\xi_{\max} = 5°$.

is a solution of the optimization problem:

$$\min_{\lambda:F\to\mathbb{H}} \quad \sum_{ij\in E} \left|\lambda_{jil} - \exp_{\mathbb{H}}\left(\frac{\vec{\omega}_{ij}}{2}\right)\lambda_{ijk}\right|^2 \qquad (16)$$
$$\text{s.t.} \quad |\lambda_{ijk}| = 1, \quad \forall ijk \in F.$$

As it is common in frame field design [Knöppel et al. 2013], the unit constraint per triangle is replaced by a more manageable norm over all triangles $\sum_{ijk\in F} |\lambda_{ijk}|^2 = 1$. With this modification, λ becomes the eigenvector associated to the smallest eigenvalue of the symmetric quadratic form defined by the objective in Eq. (16). Once the optimal λ is determined, it is projected to a unit quaternion $\lambda_{ijk} \leftarrow \lambda_{ijk}/|\lambda_{ijk}|$ on each triangle ijk before being converted to a rotation matrix $R_{ijk} \in \mathrm{SO}(3)$.

5.3 Immersion

The new vertex positions $\tilde{f} : V \to \mathbb{R}^3$ are computed from the log-scale factor $u : V \to \mathbb{R}$ and a rotation field $R : F \to \mathrm{SO}(3)$. This process involves discretizing and solving Eq. (3) in a least-squares sense.

Following the approach of Springborn et al. [2008], the log-scale factors u_i and u_j are averaged at each edge ij to obtain a total scale factor $e^{(u_i+u_j)/2}$. Similarly, a rotation $R_{ij} \coloneqq \frac{1}{2}(R_{ijk} + R_{jil})$ is computed for each edge $ij \in E$ by averaging the contributions of the rotation field on the two adjacent triangles $ijk, jil \in F$. We aim to solve the discretized Eq. (3), represented by the linear system of equations

$$\tilde{f}_i - \tilde{f}_j = \underbrace{e^{\frac{1}{2}(u_i+u_j)} R_{ij}^{\top}(f_i - f_j)}_{=:\mu_{ij}}, \quad \forall ij \in E,$$

for unknown values $\tilde{f} : V \to \mathbb{R}^3$.

Similarly to Crane et al. [2011; 2013b], we solve a linear least squares problem weighted by the cotan-weights. The resulting system of linear equations takes the form of a Poisson problem for each coordinate $a = 1, 2, 3$:

$$W\tilde{f}^a = \mathrm{div}\mu^a.$$

Here, W is the discrete Laplacian defined in Sec. 4 and $\mathrm{div}\mu$ denotes the discrete divergence given by $(\mathrm{div}\mu)_i \coloneqq \sum_{ij\in E} w_{ij}\mu_{ij}$ for each vertex $i \in V$. The solution of this system is not, in general, a discrete conformal deformation in the sense of Springborn et al. [2008]. However, conformality is reached under mesh refinement (see Sec. 7.1).

To prevent infinite growth induced by certain objective functions (cf. Sec. 6.1), the new vertex positions are rescaled, ensuring that the total area remains constant.

6 OBJECTIVE FUNCTIONS

Multiple objective functions can be used in our algorithm. We present here two concrete applications: a discretization of the bending energy introduced by Eq. (1) and a discrete Ricci flow.

6.1 Bending Energy

The bending energy, as defined in Eq. (1), can be minimized within our theoretical framework by reformulating it using our variables u and τ. Using the conformal scaling of the volume form (i.e., $\mathrm{d}A_{\tilde{f}} = e^{2u}\,\mathrm{d}A_f$), Eq. (1) is transformed to suit our optimization algorithm:

$$\tfrac{1}{2}\int_M \|\tilde{S}\|_F^{2p}\,\mathrm{d}A_{\tilde{f}} = \tfrac{1}{2}\int_M \|S - \tau\|_F^{2p} e^{2(1-p)u}\,\mathrm{d}A_f.$$

It is worth noting that this energy reduces to the Willmore energy when $p = 1$.

This function is discretized by integrating the log-scale factor u over each triangle:

$$\mathcal{E}(u,\tau) = \tfrac{1}{2}\sum_{ijk\in F} \|S_{ijk} - \mathrm{T}_{ijk}\|_F^{2p} e^{\frac{2}{3}(1-p)(u_i+u_j+u_k)} A_{ijk}. \qquad (17)$$

The quadratic approximation of Eq. (17), formed by the gradient $G_{\mathcal{E}}$ and the Hessian matrix $H_{\mathcal{E}}$, can be computed solely in term of cotangents of triangle inner angles. Given the non-convex nature of this energy, we omit the cross partial derivative terms of the Hessian matrix, as explained in Appendix C.1. Additionally, for $p > 1$, the energy consistently decreases when the log-scale factor u increases. To prevent infinite area growth, the mesh is rescaled to unit area at each iteration.

Results of this flow are illustrated in Figures 1, 7 and 8. As anticipated, higher power results in stronger smoothing effects, with high-curvature regions quickly becoming spherical. This behavior

contrasts with the Willmore flow ($p = 1$) where high curvature areas are addressed by shrinking.

6.2 Combinatorial Ricci Flow

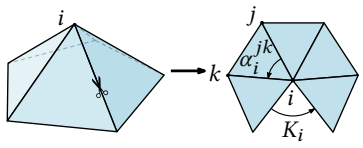
Our combinatorial Ricci flow is designed to achieve prescribed *angle defects* at each vertex. Angle defects serve as discrete approximations of the Gaussian curvature integrated within the neighborhood of a vertex [Botsch et al. 2010, p. 47]. Hence, a sphere-like triangulation with constant angle defect is not confined to a geometric sphere, as is the case with constant Gaussian curvature. Nonetheless, developable triangle meshes are still characterized by their vanishing angle defects. This characteristic makes angle defects a prominent quantity in geometry processing and is utilized in the computation of various objects, including cone parametrizations, polycubes, and approximations of developable surfaces.

Gaussian curvature prescription is also possible with our formulation. However, it requires the minimization of a non-convex energy compensating for the change in area, as demonstrated by Theorem 3.1.

Angle defect. On a triangle mesh, the integrated Gaussian curvature is approximated by the angle defect $K : V \to \mathbb{R}$ at each vertex. At vertex i, K_i is defined as the difference between 2π and the sum of all corner angles incident to i:

$$K_i = 2\pi - \sum_{ijk \in F} \alpha_k^{ij}.$$

Following [Ben-Chen et al. 2008; Springborn et al. 2008], the change in integrated Gaussian curvature in Theorem 3.1 is discretized by the equation



$$Wu = K - \tilde{K}, \tag{18}$$

where $W \in \mathbb{R}^{|V|\times|V|}$ is the cotan-Laplacian and $K, \tilde{K} \in \mathbb{R}^{|V|}$ are respectively the angle defect of the initial and deformed mesh.

Combinatorial Ricci energy. To prescribe a given angle defect $\bar{K} : V \to \mathbb{R}$, our goal is to minimize the gap between the target defect $\bar{K}$ and the defect obtained after conformal deformation $\tilde{K}$ in the least-squares sense. To achieve this, we minimize the quadratic energy

$$\mathcal{E}(u) = \tfrac{1}{2}(\tilde{K} - \bar{K})^\top W^{-1}(\tilde{K} - \bar{K}),$$

where the inverse Laplacian lowers the derivative order with respect to the log-scale factor. Incidentally, the gradient of this energy leads to the discrete Ricci flow of Jin et al. [2008].

Using Eq. (18) and ignoring the constant term, the objective function boils down to:

$$\mathcal{E}(u) = \tfrac{1}{2}u^\top W u + (K - \bar{K})^\top u.$$

Angle defect prescription. Prescribing angle defects through an immersed combinatorial Ricci flow allows us to obtain a geometric realization of cone metrics, which are ubiquitous in mesh parametrization. As shown in Figure 2 and the top row of Figure 9, we are able to compute embeddings of cone metrics extracted from a polycube [Tarini et al. 2004] whose immersion exists and can be realized in $\mathbb{R}^3$. However, our computations do *not* attempt to align normals with Cartesian axes. Figure 11 and the bottom row of Figure 9 illustrate immersions of cone metrics obtained using a curvature concentration method [Ben-Chen et al. 2008] and the singularities of a smooth frame field [Ray et al. 2008]. In this case, the existence of an immersion is not guaranteed.

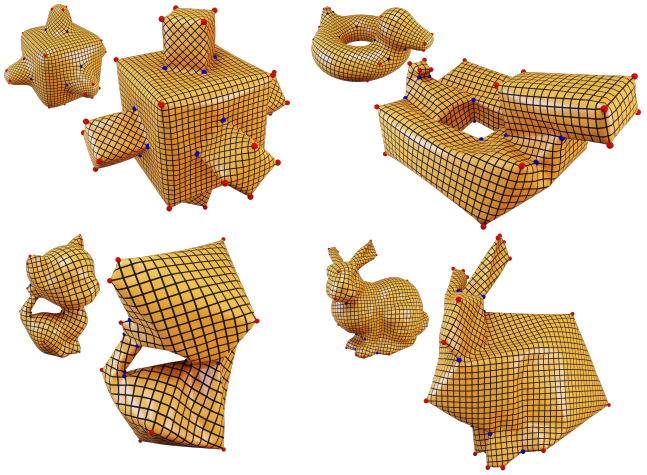


Fig. 9. Four cone metrics ($-\pi/2$ cones in blue, $\pi/2$ cones in red) are immersed in $\mathbb{R}^3$ with our discrete Ricci flow. Top row: the cones are extracted from a polycube. Bottom row: the cones are extracted from a frame field. Even though an immersion does not always exist in the latter case, the final iteration deviates from the target angle defect by no more than 2°.

Fig. 10. Conformal deformation to a mesh with zero angle defect (and Gaussian curvature) at each vertex, visualized here at iterations 1, 15 and 150. Right: the final iteration maximum absolute angle defect is 1.3°.

The angle defect prescription also allows us to visualize a flow toward a flat torus (Fig. 10) or a space of everywhere negative Gaussian curvature, illustrated in Figure 17. Across all the ricci flow examples, the angle defect deviates by no more than 4° from the target. This error is highest near ridges and cone points (Fig. 11 and Fig. 10), and so is the conformal distortion (Fig. 11, right).

7 EVALUATION AND COMPARISON

Our algorithm is implemented in MATLAB and tested on a 3.6 GHz Intel Core i9 machine with 8 cores. All the key components of the methods (finding a descent direction by solving the quadratic programming problem, computing the rotation field R through eigendecomposition, and reconstructing the deformation with a linear solve), are computed using the built-in functions QUADPROG, EIGS,

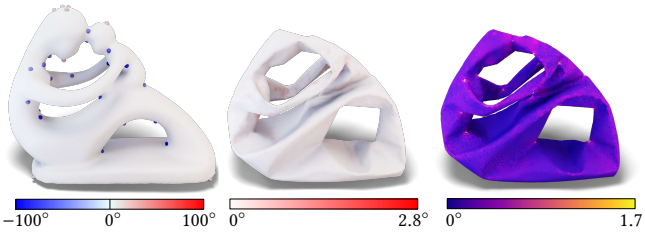


Fig. 11. Angle defect distribution obtained with [Ben-Chen et al. 2008] (left) whose cones are highlighted with colored spheres. Its immersion deviates from the target angle defect by at most 2.8° (middle) while staying close to a conformal deformation (right).

and MLDIVIDE. One iteration takes approximately 15 seconds for a 12k vertices mesh with 47k degrees of freedom. The primary computational cost is solving the quadratic programming problem. The execution time and number of iterations relative to the mesh sizes for each figure are reported in Appendix E. *The code and deformation videos are also included in the supplementary material.*

We assess the quality of a map $\varphi : f(M) \to \tilde{f}(M)$ through the quasi-conformal error C. This error is defined as the ratio of largest to smallest singular value of the differential $\mathrm{d}\varphi$ [Sander et al. 2001]. An exactly conformal deformation is achieved when $C \equiv 1$.

7.1 Convergence Under Refinement

Due to spatial and temporal discretization errors, integrability is never exactly satisfied. However, we can check that our finite element simulation converges to a conformal deformation under surface refinement.

In this experiment, we apply our discrete Willmore flow to a collection of meshes converging to a smooth surface. The flow is stopped when the objective function shows insignificant progress and meets the criterion: $|\tilde{\mathcal{E}} - \mathcal{E}| \leq 10^{-4}\mathcal{E}$ as shown in Figure 12, top. Given that this flow is known to converge to a sphere, we measure, for each final mesh, the distance to a sphere by evaluating the maximum gap between the distance of each vertex to the center of mass and the target radius. The left plot in Figure 12 illustrates the quadratic convergence of our algorithm with respect to the mean edge length. Importantly, this convergence rate remains unaffected by the choice the step size limitation strategy.

Nevertheless, the convergence to a *conformal* deformation depends on the choice of maximal rotation angle $\xi_{\max}$ allowed between two adjacent faces (Fig. 12, right). This is explained by the fact that this angle is independent of the mesh resolution. Consequently, on fine meshes the field R can rotate at a higher speed than on coarse meshes with larger edge lengths. Thus, for a fixed $\xi_{\max}$, the flow will allow larger deformations on a fine mesh than on a coarse approximation. In this case, convergence to a conformal deformation is not observed under mesh refinement, and the conformal distortion stabilizes at a small value. Empirically, we find that linear convergence is obtained when $\xi_{\max}$ is chosen proportionally to the squared mean edge length. However, it requires more iterations for larger meshes to reach convergence. For simplicity, we choose to keep $\xi_{\max}$ has an adjustable parameter.

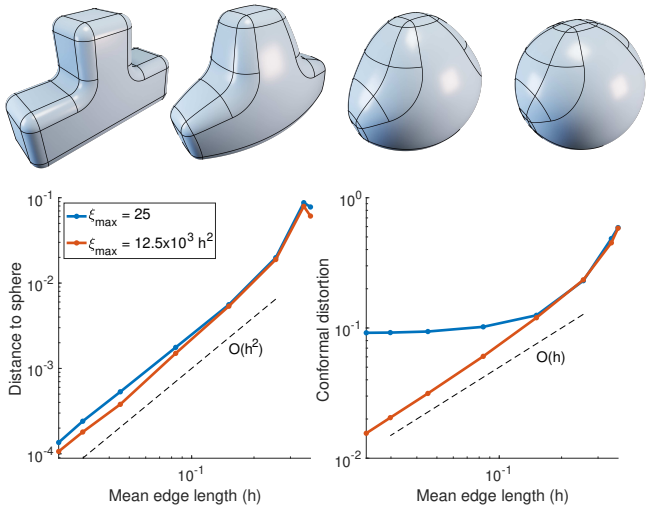
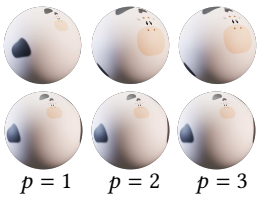


Fig. 12. Convergence under mesh refinement of our Willmore flow applied to a T-shape (top far left). The log-log plot of the distance to a sphere of the converged surface versus the mean edge length h (bottom left) shows that our algorithm converges to a sphere independently of the choice of step size limitation $\xi_{\max}$. However, to ensure convergence to a *conformal* deformation, $\xi_{\max}$ should chosen be proportional to h^2.

7.2 Conformal Equivalence of Spherical Parametrizations

In Figure 1, our bending flow exhibits convergence toward a sphere for three different values of p (inset top row). Given that all conformal maps from the sphere to itself are known to be Möbius transformations, we propose to verify whether these deformations are conformally equivalent to each other. To achieve this, we employ the Möbius registration algorithm proposed by Baden et al. [2018] to compute the optimal conformal alignment of the three spherical parametrizations (inset bottom row). Our findings indicate that the angular distance between any pair of corresponding vertices is at most 0.63 degrees, thus confirming conformal equivalence within acceptable numerical margins.



For the dog model in Figure 7, the maximum error is 2.88 degrees. This discrepancy primarily stems from the significant conformal distortion observed when $p = 1$. By reducing the step size $\xi_{\max}$ to 5° instead of 25°, the maximum error decreases to 0.83 degrees. This illustrates the trade-off between speed and conformal distortion discussed in Section 7.1.

7.3 Willmore Flow Comparison

Several algorithms are available for computing a flow from a surface to a sphere. We compare our method with three of them in terms of numerical stability and conformal distortion. In order to assess their numerical stability, we use the largest possible step size for

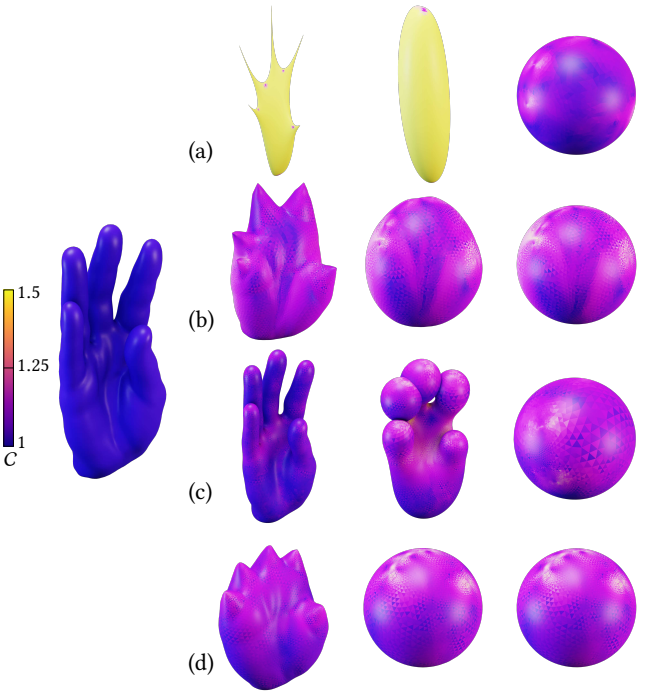


Fig. 13. Conformal distortion comparison of four geometric flows converging to a sphere: (a) [Kazhdan et al. 2012], (b) [Crane et al. 2013b], (c) [Gruber and Aulisa 2020] and (d) our method, applied to a hand model after 1, 4 and 25 iterations. The algorithm of Gruber et Aulisa [2020] is visualized at iterations 20, 50 and 90. We set parameters to achieve maximal convergence speed.

each method. As explained in Section 7.1, there is a trade-off between the number of iterations and the conformal distortion. As we favor speed, all the deformations suffer from high, but comparable, distortion.

All the algorithms are implemented in MATLAB, except for [Gruber and Aulisa 2020], for which a C++ code is provided by the authors. The four algorithms undergo testing on the same machine using a 10k vertices mesh. Figure 13 illustrates the evolution of the quasi-conformal distortion during the flow, while Figure 14 records the evolution of the Willmore energy versus the number of iterations and the execution time.

The modified mean curvature flow [Kazhdan et al. 2012] can take exceptionally large steps toward the sphere, but its intermediate meshes deviate significantly from being conformal and exhibit high distortion. In contrast to the other three algorithms, its primary objective is not to minimize the Willmore energy.

Our method is comparable to Crane et al. [2013b] in terms of intermediate surfaces and conformal distortion. Although our execution time per iteration is higher (11.5s in average) than this method (5.9s in average), we can take larger steps, especially near convergence, resulting in comparable total running times for both methods.

The Willmore flow of Gruber and Aulisa [2020] requires very small time steps in the early stages of the optimization to prevent blow up. As described in the paper, careful step size scheduling

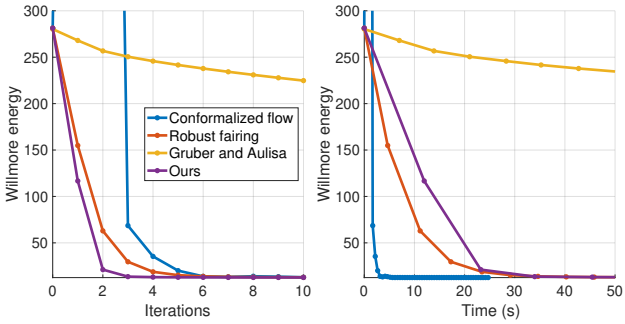


Fig. 14. Willmore energy evolution of four geometric flows: conformalized flow [Kazhdan et al. 2012], robust fairing [Crane et al. 2013b], Gruber and Aulisa [2020] and ours. All methods eventually reach the energy minimum and are set up to achieve maximum convergence speed, except ours whose step size limitation ξ_{max} is set to 25°. Our method is the fastest in number of iterations but is comparable to [Crane et al. 2013b] in term of execution time. The surface evolution for each method is illustrated in Figure 13.

following a geometric progression is necessary. Even though the cost of one iteration (8.3s) is lower than our method, it takes longer to reach convergence. The deformation stays close to conformal during the simulation, even though it is not its primary objective.

7.4 p-Willmore Flow

Our method is versatile and accommodates various objective functions. For instance, Gruber et al. [2020; 2019] introduced the p-Willmore energy:

$$\frac{1}{2}\int_M \operatorname{tr}(\tilde{S})^{2p}\,\mathrm{d}A_{\tilde{f}},$$

as an efficient flow for smoothing surfaces. Its numerical computation involves an ad hoc weak formulation of the curvature, with an additional conformal penalty to preserve the triangle shapes during the flow.

Within our framework, we can optimize this energy by reformulating it using our variables u and τ:

$$\frac{1}{2}\int_M \operatorname{tr}(\tilde{S})^{2p}\,\mathrm{d}A_{\tilde{f}} = \frac{1}{2}\int_M \operatorname{tr}\,(S-\tau)^{2p}\,e^{2(1-p)u}\,\mathrm{d}A_f.$$

This function is discretized by integrating the log-scale factor u over each triangle:

$$\mathcal{E}(u,\tau) = \frac{1}{2}\sum_{ijk\in F} \operatorname{tr}\left(S_{ijk} - \mathrm{T}_{ijk}\right)^{2p} e^{\frac{2}{3}(1-p)(u_i+u_j+u_k)} A_{ijk}. \quad (19)$$

Similar to the bending energy in Section 6.1, minimizing this functional would lead to infinite surface growth, so we rescale the mesh at each iteration. The quadratic approximation is computed in Appendix C.2.

The top rows of Figure 16 illustrate this flow for $p = 2$ and $p = 3$. For this mesh, the p-Willmore flow appears to be comparable to the bending flow in Figure 7. However, we observe that this energy can reach undesirable minima, as saddle points ($H = 0$) tend to be weakly penalized. For example, when imposing positional constraints, instead of yielding a smooth interpolation, this energy

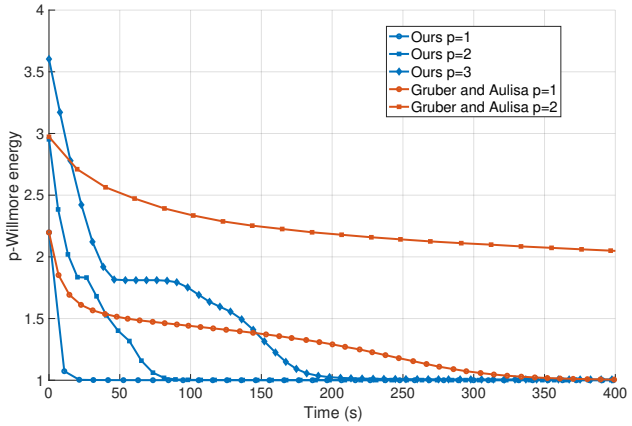


Fig. 15. The p-Willmore energy minimization of Gruber and Aulisa [2020] compared to our method. The energy is scaled so that the minimum value is 1 for any p. Our method is always faster, requires less iterations and its cost per iteration is independent of the value p. The surface evolution of each method is illustrated in Fig. 7 for $p = 1$ and Fig. 16 otherwise.

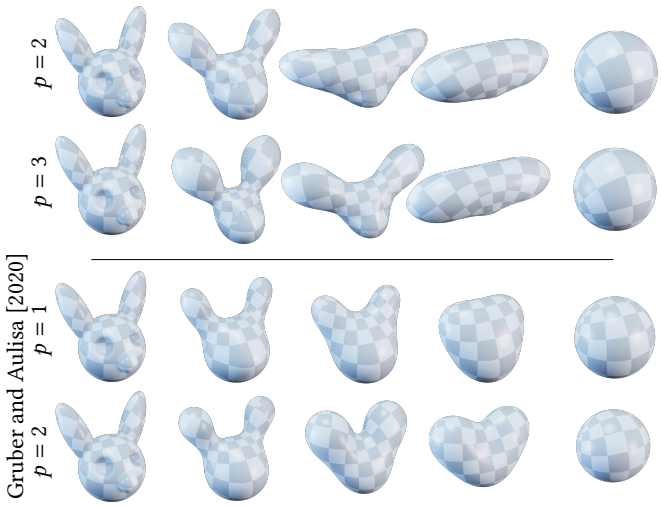


Fig. 16. The p-Willmore flow computed with our method (top rows) compared to Gruber and Aulisa [2020] (bottom rows).

can lead to surfaces reminiscent of a constant-mean-curvature, as seen in Figure 19.

In comparison to Gruber and Aulisa [2020], our algorithm exhibits greater stability, allowing larger steps and faster convergence to the sphere both in terms of number of iterations and execution time (Fig. 15). Additionally, our average execution time per iteration remains lower than 10 seconds, independently of the parameter p. In contrast, the p-Willmore flow experiences a range from 9s/it for a standard Willmore flow to 23s/it for $p = 2$. The latter takes 200 iterations and 76 minutes to reach convergence.

As illustrated in Figure 16, our method requires higher values of p to achieve a flow comparable to Gruber and Aulisa [2020]. The high smoothness of their shapes appears to be a side effect of their conformal penalty, which relies on the normal field, creating a stiffer deformation than that imposed by the energy. Nonetheless, their average conformal distortion remains comparable to ours, with their worst score being 1.15 for $p = 2$ against 1.09 for our Willmore flow.

7.5 Shape From Metric

Chern et al. [2018] proposed an algorithm for realizing geometric immersions of a given metric. This method, when combined with a discrete Ricci flow [Zhang et al. 2015], can also compute conformal deformations to achieve prescribed angle defects.

In this experiment, target angle defect remains constant across all vertices and, in accordance with the Gauss-Bonnet theorem, must be equal to $-4\pi/|V|$. Since the integrated Gaussian curvature is prescribed instead of the Gaussian curvature itself, the resulting surface is *not* an immersion of a hyperbolic space but rather an immersion of a space with everywhere negative Gaussian curvature.

As depicted in Figure 17, their results exhibit a comparable maximum angle defect error, albeit some error spikes. Their conformal error is also higher with a maximum of 1.6 against 1.3 with our method. The surfaces are different as an immersion may not exist and several approximations are possible.

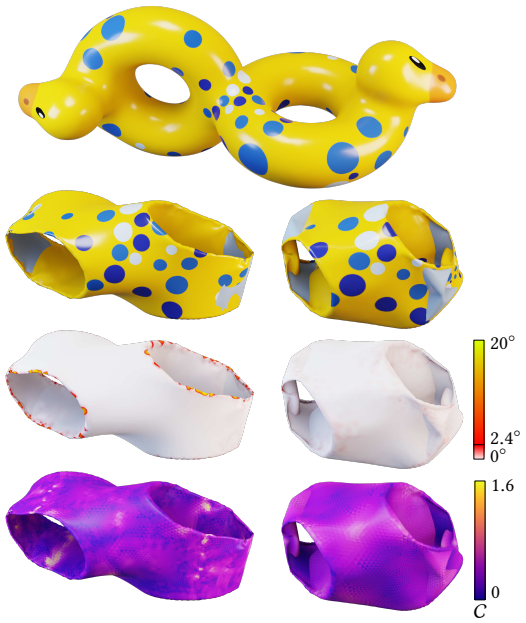


Fig. 17. A conformal immersion of a double torus with everywhere negative Gaussian curvature is computed with our combinatorial Ricci flow (right) compared to the shape from metric algorithm [Chern et al. 2018] (left). Shape from metric generates spikes in deviation from the target angle defect (middle row) and our immersion is less conformally distorted (bottom row).

8 LIMITATIONS AND FUTURE WORK

Failure cases. The main limitation of our method, inherent to any low-order finite element simulation, is its sensitivity to triangulation quality and how closely the triangle mesh approximates an underlying smooth surface. In Figure 18 (left), the inadequate curvature

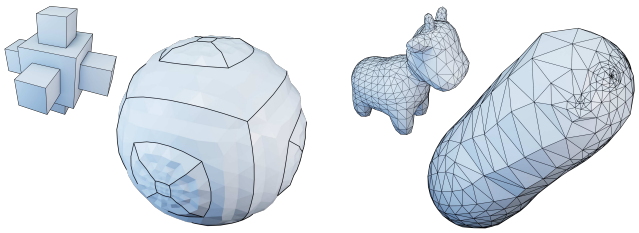


Fig. 18. Due to poor curvature approximation, our bending flow may fail in presence of sharp edges (left, $p = 1$) or on coarse meshes (right, $p = 2$).

approximation of sharp features results in pronounced discretization artifacts upon convergence, even when starting from a high density Delaunay triangulation. Coarse meshes exacerbate this issue while suffering with a significant triangle distortion during deformation. As shown in Figure 18 (right), this distortion can occasionally prevent the convergence of the our bending flow, regardless of the chosen step size.

Boundary conditions. Our focus was on surfaces *without* boundaries. However, boundary conditions can be studied both in the theoretical and practical aspects. Essentially, it involves modifying the integrability conditions to account for boundary edges forming loops around boundary vertices.

Future work. We are hopeful that this paradigm will find applications in various areas of geometry processing and that extending the analysis beyond conformality will enable numerically robust and precise control of surface curvature.

REFERENCES

Alex Baden, Keenan Crane, and Misha Kazhdan. 2018. Möbius registration. In *Computer Graphics Forum*, Vol. 37. Wiley Online Library, 211–220.

Mirela Ben-Chen, Craig Gotsman, and Guy Bunin. 2008. Conformal flattening by curvature prescription and metric scaling. In *Computer Graphics Forum*, Vol. 27. Wiley Online Library, 449–458.

Alexander I Bobenko and Peter Schröder. 2005. Discrete willmore flow. In *ACM SIGGRAPH 2005 Courses*. 5–es.

Mario Botsch, Leif Kobbelt, Mark Pauly, Pierre Alliez, and Bruno Lévy. 2010. *Polygon mesh processing*. CRC press.

Robert Bridson, Sebastian Marino, and Ronald Fedkiw. 2005. Simulation of clothing with folds and wrinkles. In *ACM SIGGRAPH 2005 Courses*. 3–es.

Peter B Canham. 1970. The minimum energy of bending as a possible explanation of the biconcave shape of the human red blood cell. *Journal of theoretical biology* 26, 1 (1970), 61–81.

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

Albert Chern, Ulrich Pinkall, and Peter Schröder. 2015. Close-to-conformal deformations of volumes. *ACM Transactions on Graphics (TOG)* 34, 4 (2015), 1–13.

Pascal Cherrier. 1984. Problemes de Neumann non linéaires sur les variétés riemanniennes. *Journal of Functional Analysis* 57, 2 (1984), 154–206.

Bennett Chow and Feng Luo. 2003. Combinatorial Ricci flows on surfaces. *Journal of Differential Geometry* 63, 1 (2003), 97–129.

Etienne Corman and Keenan Crane. 2019. Symmetric Moving Frames. *ACM Trans. Graph.* 38, 4 (2019).

Keenan Crane. 2013. *Conformal Geometry Processing*. Ph. D. Dissertation. Caltech.

Keenan Crane, Fernando De Goes, Mathieu Desbrun, and Peter Schröder. 2013a. Digital geometry processing with discrete exterior calculus. In *ACM SIGGRAPH 2013 Courses*. 1–126.

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

Keenan Crane, Ulrich Pinkall, and Peter Schröder. 2013b. Robust fairing via conformal curvature flow. *ACM Transactions on Graphics (TOG)* 32, 4 (2013), 1–10.

Fernando de Goes, Siome Goldenstein, and Luiz Velho. 2008. A simple and flexible framework to adapt dynamic meshes. *Computers & Graphics* 32, 2 (2008), 141–148.

Mathieu Desbrun, Eva Kanso, and Yiying Tong. 2006. Discrete differential forms for computational modeling. In *ACM SIGGRAPH 2006 Courses*. 39–54.

Mathieu Desbrun, Melvin Leok, and Jerrold E Marsden. 2005. Discrete Poincaré lemma. *Applied Numerical Mathematics* 53, 2-4 (2005), 231–248.

Manfredo Perdigao Do Carmo and J Flaherty Francis. 1992. *Riemannian geometry*. Vol. 6. Springer.

Michael Eigensatz and Mark Pauly. 2009. Positional, metric, and curvature control for constraint-based surface deformation. In *Computer Graphics Forum*, Vol. 28. Wiley Online Library, 551–558.

Michael Eigensatz, Robert W Sumner, and Mark Pauly. 2008. Curvature-domain shape processing. In *Computer Graphics Forum*, Vol. 27. Wiley Online Library, 241–250.

Matthew Fisher, Peter Schröder, Mathieu Desbrun, and Hugues Hoppe. 2007. Design of tangent vector fields. *ACM transactions on graphics (TOG)* 26, 3 (2007), 56–es.

Eitan Grinspun, Anil N Hirani, Mathieu Desbrun, and Peter Schröder. 2003. Discrete shells. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*. Citeseer, 62–67.

Anthony Gruber and Eugenio Aulisa. 2020. Computational p-Willmore flow with conformal penalty. *ACM Transactions on Graphics (TOG)* 39, 5 (2020), 1–16.

Anthony Gruber, Magdalena Toda, and Hung Tran. 2019. On the variation of curvature functionals in a space form with application to a generalized Willmore energy. *Annals of Global Analysis and Geometry* 56 (2019), 147–165.

Hans Hagen and Guido Schulze. 1987. Automatic smoothing with geometric surface patches. *Computer Aided Geometric Design* 4, 3 (1987), 231–235.

Richard S Hamilton. 1988. The Ricci flow on surfaces, Mathematics and general relativity. *Contemp. Math.* 71 (1988), 237–261.

Wolfgang Helfrich. 1973. Elastic properties of lipid bilayers: theory and possible experiments. *Zeitschrift für Naturforschung c* 28, 11-12 (1973), 693–703.

Anil Nirmal Hirani. 2003. *Discrete exterior calculus*. Ph. D. Dissertation. California Institute of Technology.

Miao Jin, Junho Kim, Feng Luo, and Xianfeng Gu. 2008. Discrete surface Ricci flow. *IEEE Transactions on Visualization and Computer Graphics* 14, 5 (2008), 1030–1043.

Pushkar Joshi and Carlo Séquin. 2007. Energy minimizers for curvature-based surface functionals. *Computer-Aided Design and Applications* 4, 5 (2007), 607–617.

G Kamberov, F Pedit, and U Pinkall. 1998. Bonnet pairs and isothermic surfaces. *DUKE MATHEMATICAL JOURNAL* 92, 3 (1998).

Michael Kazhdan, Jake Solomon, and Mirela Ben-Chen. 2012. Can mean-curvature flow be modified to be non-singular?. In *Computer Graphics Forum*, Vol. 31. Wiley Online Library, 1745–1754.

Felix Knöppel, Keenan Crane, Ulrich Pinkall, and Peter Schröder. 2013. Globally optimal direction fields. *ACM Trans. Graph.* 32, 4 (2013).

Bruno Lévy. 2003. Dual domain extrapolation. *ACM Transactions on Graphics (TOG)* 22, 3 (2003), 364–369.

Lizao Li. 2018. *Regge finite elements with applications in solid mechanics and relativity*. Ph. D. Dissertation. University of Minnesota.

Nigel James Lott and DI Pullin. 1988. Method for fairing b-spline surfaces. *Computer-Aided Design* 20, 10 (1988), 597–600.

Henry P Moreton and Carlo H Séquin. 1992. Functional optimization for fair surface design. *ACM SIGGRAPH Computer Graphics* 26, 2 (1992), 167–176.

Shigeyuki Morita. 2001. *Geometry of differential forms*. Number 201. American Mathematical Soc.

Barrett O'Neill. 2006. *Elementary differential geometry*. Academic press.

Nicolas Ray, Bruno Vallet, Wan Chiu Li, and Bruno Lévy. 2008. N-symmetry direction field design. *ACM Transactions on Graphics (TOG)* 27, 2 (2008), 1–13.

Tullio Regge. 1961. General relativity without coordinates. *Il Nuovo Cimento (1955-1965)* 19 (1961), 558–571.

Pedro V Sander, John Snyder, Steven J Gortler, and Hugues Hoppe. 2001. Texture mapping progressive meshes. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. 409–416.

Alla Sheffer, Emil Praun, Kenneth Rose, et al. 2007. Mesh parameterization methods and their applications. *Foundations and Trends® in Computer Graphics and Vision* 2, 2 (2007), 105–171.

Yousuf Soliman, Albert Chern, Olga Diamanti, Felix Knöppel, Ulrich Pinkall, and Peter Schröder. 2021. Constrained willmore surfaces. *ACM Transactions on Graphics (TOG)* 40, 4 (2021), 1–17.

Boris Springborn, Peter Schröder, and Ulrich Pinkall. 2008. Conformal equivalence of triangle meshes. In *ACM SIGGRAPH 2008 papers*. 1–11.

Marco Tarini, Kai Hormann, Paolo Cignoni, and Claudio Montani. 2004. Polycube-maps. *ACM transactions on graphics (TOG)* 23, 3 (2004), 853–860.

Demetri Terzopoulos, John Platt, Alan Barr, and Kurt Fleischer. 1987. Elastically deformable models. In *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*. 205–214.

Amir Vaxman, Christian Müller, and Ofir Weber. 2018. Canonical möbius subdivision. *ACM Transactions on Graphics (TOG)* 37, 6 (2018), 1–15.

Thomas J Willmore. 1965. Note on embedded surfaces. *An. Sti. Univ."Al. I. Cuza" Iasi Sect. I a Mat.(NS) B* 11, 493-496 (1965), 20.

Wei-Wei Xu and Kun Zhou. 2009. Gradient domain mesh deformation—A survey. *Journal of computer science and technology* 24 (2009), 6–18.
Chao Yuan, Nan Cao, and Yang Shi. 2023. A Survey of Developable Surfaces: From Shape Modeling to Manufacturing. *arXiv preprint arXiv:2304.09587* (2023).
Min Zhang, Wei Zeng, Ren Guo, Feng Luo, and Xianfeng David Gu. 2015. Survey on discrete surface Ricci flow. *Journal of Computer Science and Technology* 30 (2015), 598–613.

A PROOF: CONFORMAL DEFORMATIONS AND DARBOUX DERIVATIVE

In this section, we provide a proof for Theorems 3.1 and 3.2. The proof is organized into three parts. First, we compute the integrability condition on the Darboux derivative and deduce Theorem 3.2. Second, we demonstrate that the conformal deformation described in Eq. (3) defines an adapted frame on the deformed surface. Third, leveraging the results of Cartan's method of moving frames, we establish the relationship between the log-scale factor u and the symmetric linear map τ to changes in curvature.

For a detailed introduction to Cartan's method, we refer the interested reader to O'Neil [2006].

Adapted frames. Throughout the proof, we will need three local coordinate systems: an extrinsic basis adapted to the surface immersion $(\bar{e}_1, \bar{e}_2, \bar{e}_3)$, a tangent vector basis $\bar{X}_1, \bar{X}_2 \in TM$ and a 1-form basis $\bar{\sigma}_1, \bar{\sigma}_2 : TM \to \mathbb{R}$.

To achieve this, we assign at each point of a small domain $\Omega \subset M$, an adapted orthonormal frame $\bar{E} = \begin{bmatrix} \bar{e}_1 & \bar{e}_2 & \bar{e}_3 \end{bmatrix} \in \mathrm{SO}(3)$, forming a smooth rotation field. These frames are chosen so that $\bar{e}_3$ aligns with the surface normal N. Since $\bar{e}_1, \bar{e}_2 \in \mathbb{R}^3$ are tangent to the surface, they are associated to tangent orthogonal vectors $\bar{X}_1, \bar{X}_2 \in TM$ satisfying $\mathrm{d}f(\bar{X}_1) = \bar{e}_1, \mathrm{d}f(\bar{X}_2) = \bar{e}_2$.

The coframes $\bar{\sigma}_i : TM \to \mathbb{R}$ are defined as the dual of these tangent vectors $\bar{\sigma}_i(\bar{X}_j) = \delta_{ij}$ (or $\bar{\sigma}_i(X) = \langle \bar{X}_i, X \rangle$) and form a basis of 1-forms. Hence, any 1-form α can be decomposed as:

$$\alpha = \alpha(\bar{X}_1)\bar{\sigma}_1 + \alpha(\bar{X}_2)\bar{\sigma}_2. \tag{20}$$

Even though the frames and coframes are defined locally, all our results are independent of these local coordinate systems.

A.1 Integrability Condition

Our objective is to compute ω, the Darboux derivative of the rotation field R (see Sec. 2), by taking the exterior derivative of Eq. (3).

Following the definition of conformal deformations in Eq. (3), the vector-valued 1-form $e^u R^\top \mathrm{d}f$ must be closed:

$$0 = \mathrm{d}(\mathrm{d}f) = \mathrm{d}(e^u R^\top \mathrm{d}f) = e^u R^\top (\mathrm{d}u \wedge \mathrm{d}f - \omega \wedge \mathrm{d}f). \tag{21}$$

The term $e^u R^\top$ in Eq. (21) is invertible and can be simplified:

$$\omega \wedge \mathrm{d}f = \mathrm{d}u \wedge \mathrm{d}f. \tag{22}$$

Evaluating the wedge products in the local basis $(\bar{X}_1, \bar{X}_2)$, yields:

$$\omega(\bar{X}_1)\bar{e}_2 - \omega(\bar{X}_2)\bar{e}_1 = \mathrm{d}u(\bar{X}_1)\bar{e}_2 - \mathrm{d}u(\bar{X}_2)\bar{e}_1.$$

Taking the scalar product with the adapted frames $(\bar{e}_1, \bar{e}_2, \bar{e}_3)$, leads to three equations characterizing ω:

$$\left| \begin{array}{l} \bar{e}_1^\top \omega(\bar{X}_1)\bar{e}_2 = -\mathrm{d}u(\bar{X}_2), \\ \bar{e}_2^\top \omega(\bar{X}_2)\bar{e}_1 = -\mathrm{d}u(\bar{X}_1), \\ \bar{e}_3^\top \omega(\bar{X}_1)\bar{e}_2 = \bar{e}_3^\top \omega(\bar{X}_2)\bar{e}_1. \end{array} \right.$$

Furthermore, $\bar{e}_2^\top \omega \bar{e}_3$ and $\bar{e}_3^\top \omega \bar{e}_1$ can be decomposed in the 1-form basis of coframes $(\bar{\sigma}_1, \bar{\sigma}_2)$ (see Eq. (20)), using three real-valued functions $a, b, c : M \to \mathbb{R}$:

$$\left| \begin{array}{l} \bar{e}_1^\top \omega \bar{e}_2 = *\mathrm{d}u, \\ \bar{e}_2^\top \omega \bar{e}_3 = -b\bar{\sigma}_1 - c\bar{\sigma}_2, \\ \bar{e}_3^\top \omega \bar{e}_1 = a\bar{\sigma}_1 + b\bar{\sigma}_2. \end{array} \right. \tag{23}$$

Noting A the $\mathfrak{so}(3)$-valued 1-form $A_{ij} := \bar{e}_i^\top \omega \bar{e}_j$, we have $\omega = \bar{E}A\bar{E}^\top$. Breaking down this matrix multiplication, we obtain:

$$\omega = A_{12}(\bar{e}_1\bar{e}_2^\top - \bar{e}_2\bar{e}_1^\top) + A_{23}(\bar{e}_2\bar{e}_3^\top - \bar{e}_3\bar{e}_2^\top) + A_{31}(\bar{e}_3\bar{e}_1^\top - \bar{e}_1\bar{e}_3^\top).$$

Here, the matrix $\bar{e}_i\bar{e}_j^\top - \bar{e}_j\bar{e}_i^\top$ is simply the cross product matrix associated to the vector $\bar{e}_j \times \bar{e}_i = -\bar{e}_k$. Hence, the $\mathfrak{so}(3)$-valued 1-form ω reduces to:

$$\vec{\omega} = -A_{12}\bar{e}_3 - A_{23}\bar{e}_1 - A_{31}\bar{e}_2.$$

Replacing the coefficients of A with Eq. (23), yields:

$$\vec{\omega} = - * \mathrm{d}uN + (b\bar{\sigma}_1 + c\bar{\sigma}_2)\bar{e}_1 - (a\bar{\sigma}_1 + b\bar{\sigma}_2)\bar{e}_2.$$

To obtain the integrability condition in Eq. (4), we introduce the symmetric linear map $\tau : TM \to TM$, whose decomposition in the local basis $(\bar{X}_1, \bar{X}_2)$ reads:

$$\tau(X) = \begin{bmatrix} \bar{X}_1 & \bar{X}_2 \end{bmatrix} \begin{bmatrix} a & b \\ b & c \end{bmatrix} \begin{bmatrix} \bar{\sigma}_1(X) \\ \bar{\sigma}_2(X) \end{bmatrix}.$$

Necessary and sufficient condition. In summary, Eq. (4) is a reformulation of Eq. (22) which is equivalent to Eq. (21), expressing the closedness of $e^u R^\top \mathrm{d}f$. Therefore, if M is simply connected and assuming that ω is the Darboux derivative of some rotation field R, the form $e^u R^\top \mathrm{d}f$ is closed, hence exact. This concludes the proof of Theorem 3.2.

A.2 Adapted Frames on the Deformed Surface

Darboux derivatives. The variation of the frame field $\bar{E}$ can be described by the left Darboux derivative $\bar{\omega}^L : M \to \mathfrak{so}(3)$, as presented in Section 2:

$$\mathrm{d}\bar{E} = \bar{\omega}^L \bar{E}.$$

The derivative $\bar{\omega}^L$ represents the speed of rotation in the canonical basis of $\mathbb{R}^3$. Instead, the method of the moving frames utilizes the *right* Darboux derivative $\bar{\omega}^R : TM \to \mathfrak{so}(3)$, defined as:

$$\mathrm{d}\bar{E} = \bar{E}\bar{\omega}^R.$$

In this case, the speed rotation is expressed in the basis of the frame itself. For instance, the coefficient $\bar{\omega}^R_{12}$ denotes the speed of rotation around the axis $\bar{e}_3$, while $\bar{\omega}^L_{12}$ represents the speed of rotation around the third axis of the ambient space.

Conformal change of frames. Following Eq. (3), the immersion $\tilde{f}$ maps $\bar{e}_i$ to $e^u R^\top \bar{e}_i$. Thus, the orthonormal frame $\tilde{E} := R^\top \bar{E}$ is adapted to the deformed surface, meaning that the vectors $\tilde{e}_1, \tilde{e}_2 \in \mathbb{R}^3$ are tangent to the deformed immersed surface and $\tilde{e}_3 \in \mathbb{R}^3$ is its normal. Hence, there exist tangent vectors $\tilde{X}_1, \tilde{X}_2 \in TM$ inverse image of $\tilde{e}_1, \tilde{e}_2$ by the immersion, i.e., $\mathrm{d}\tilde{f}(\tilde{X}_i) := \tilde{e}_i$. The coframes dual to $\tilde{X}_1, \tilde{X}_2$ are denoted by $\tilde{\sigma}_1, \tilde{\sigma}_2 : TM \to \mathbb{R}$.

Due to conformality, $\bar{X}$ and $\tilde{X}$ are equal up to scale:

$$\mathrm{d}f(\bar{X}_1) = \bar{e}_1 = R\tilde{e}_1 = R\mathrm{d}\tilde{f}(\tilde{X}_1) = e^u \mathrm{d}f(\tilde{X}_1) = \mathrm{d}f\left(e^u \tilde{X}_1\right).$$

This leads to a conformal scaling between the frames (resp. coframes) defined by f and the deformed surface defined by $\tilde{f}$:

$$\bar{X}_i = e^u \tilde{X}_i, \quad \bar{\sigma}_i = e^{-u}\tilde{\sigma}_i. \tag{24}$$

The right Darboux derivative $\tilde{\omega}^R$ of the frame $\tilde{E}$ is given by:

$$\mathrm{d}\tilde{E} = \mathrm{d}(R^\top \bar{E}) = -R^\top \omega \bar{E} + R^\top \bar{E}\bar{\omega}^R = \tilde{E}(-\bar{E}^\top \omega \bar{E} + \bar{\omega}^R),$$

thus $\tilde{\omega}^R = \bar{\omega}^R - \bar{E}^\top \omega \bar{E}$. Detailing each component of $\tilde{E}$ and injecting Eq. (23), we conclude that the Darboux derivative is altered by the conformal deformation in the following way:

$$\left|\begin{array}{l} \tilde{\omega}^R_{12} = \bar{\omega}^R_{12} - *\mathrm{d}u, \\ \tilde{\omega}^R_{23} = \bar{\omega}^R_{23} + b\bar{\sigma}_1 + c\bar{\sigma}_2, \\ \tilde{\omega}^R_{31} = \bar{\omega}^R_{31} - a\bar{\sigma}_1 - b\bar{\sigma}_2. \end{array}\right. \tag{25}$$

A.3 Relation to Curvature

We are ready to evaluate the impact of the deformation on the curvature as the right Darboux derivative and the coframes uniquely characterize the surface curvature [O'Neill 2006, p. 270]:

$$\mathrm{d}\bar{\omega}^R_{12} = K\bar{\sigma}_1 \wedge \bar{\sigma}_2, \tag{26}$$

$$S = \begin{bmatrix} \bar{\omega}^R_{31}(\bar{X}_1) & \bar{\omega}^R_{31}(\bar{X}_2) \\ -\bar{\omega}^R_{23}(\bar{X}_1) & -\bar{\omega}^R_{23}(\bar{X}_2) \end{bmatrix}. \tag{27}$$

Here, the shape operator $S : TM \to TM$ is the opposite of the normal covariant derivative $S(X) := -\nabla_X N$. (Note that [O'Neill 2006] uses a row-based indexation of matrices, so that their ω is opposite to ours.)

Gaussian curvature. By injecting Eqs. (25) and (24) into Eq. (26), and after simplification, we recover the infamous Cherrier formula [Ben-Chen et al. 2008; Cherrier 1984]:

$$\Delta u = K - e^{2u}\tilde{K}.$$

Curvature tensor. By injecting Eqs. (25) and (24) into Eq. (27), the symmetric linear map τ turns out to be the change in shape operator:

$$\begin{aligned} \tilde{S} &= \begin{bmatrix} \tilde{\omega}^R_{31}(\tilde{X}_1) & \tilde{\omega}^R_{31}(\tilde{X}_2) \\ -\tilde{\omega}^R_{23}(\tilde{X}_1) & -\tilde{\omega}^R_{23}(\tilde{X}_2) \end{bmatrix} \\ &= \begin{bmatrix} \bar{\omega}^R_{31}(\tilde{X}_1) & \bar{\omega}^R_{31}(\tilde{X}_2) \\ -\bar{\omega}^R_{23}(\tilde{X}_1) & -\bar{\omega}^R_{23}(\tilde{X}_2) \end{bmatrix} - e^{-u}\begin{bmatrix} a & b \\ b & c \end{bmatrix} \\ &= e^{-u}S - e^{-u}\tau. \end{aligned}$$

B PROOF: VECTOR FIELD INTEGRAL PRESERVATION

Let $v : M \to TM$ be a tangent vector field. We aim to have the flow preserve the integral $\int_M \mathrm{d}\tilde{f}(v)\,\mathrm{d}A_f$ over time. Taking the time derivative at time 0 yields

$$\int_M \dot{u}\mathrm{d}f(v)\,\mathrm{d}A_f = -\int_M \dot{R}^\top \mathrm{d}f(v)\,\mathrm{d}A_f. \tag{28}$$

The Hodge decomposition [Morita 2001, p. 160] of the 2-forms $\mathrm{d}f(v)\,\mathrm{d}A_f$ yields $\mathrm{d}f(v)\,\mathrm{d}A_f = \mathrm{d}\beta + c\,\mathrm{d}A_f$, where $c \in \mathbb{R}^3$ is a constant and $\beta : TM \to \mathbb{R}^3$ is a vector-valued 1-form. In practice, β is computed by solving the equation $\Delta_1\beta = *\mathrm{d}(\mathrm{d}f(v))$ where $\Delta_1 = \delta\mathrm{d} + \mathrm{d}\delta$ is the 1-form Hodge Laplacian [Morita 2001, p. 155].

Moreover, by taking the exterior and time derivative at time zero of Eq. (2), we find that $\mathrm{d}\dot{R} = \dot{\omega}$. Since $\dot{R}$ is only known through $\dot{\omega}$, it can be chosen up to an additive constant. Without loss of generality, we assume that $\dot{R}$ sums to zero, i.e., $\int_M \dot{R}\,\mathrm{d}A_f = 0$.

Therefore, Eq. (28) is further transformed as:

$$\int_M \dot{u}\mathrm{d}f(v)\,\mathrm{d}A_f = -\int_M \dot{R}^\top \mathrm{d}\beta - \cancel{\int_M \dot{R}^\top \mathrm{d}A_f\, c}.$$

Using Stokes' theorem on a manifold without boundary, we obtain:

$$\int_M \dot{u}\mathrm{d}f(v)\,\mathrm{d}A_f = -\cancel{\int_M \mathrm{d}(\dot{R}^\top\beta)} + \int_M \mathrm{d}\dot{R}^\top \wedge \beta = -\int_M \dot{\omega}\wedge\beta.$$

B.1 Non-simply Connected Surfaces

To define a deformation, the 1-form $e^u R^\top \mathrm{d}f$ must be exact, implying orthogonality to all harmonic 1-forms η:

$$\int_M e^u R^\top \mathrm{d}f \wedge *\eta = 0.$$

Evaluating the 2-form $\mathrm{d}f \wedge *\eta$ on the basis $X_1, X_2 \in TM$ yields:

$$\begin{aligned} \mathrm{d}f \wedge *\eta(X_1, X_2) &= \mathrm{d}f(X_1)\eta(X_1) + \mathrm{d}f(X_2)\eta(X_2) \\ &= \mathrm{d}f\left(\eta(X_1)X_1 + \eta(X_2)X_2\right). \end{aligned}$$

Let $v = \eta(X_1)X_1 + \eta(X_2)X_2$ be the tangent vector field dual to the form η, we have the integrability condition:

$$\int_M e^u R^\top \mathrm{d}f(v)\,\mathrm{d}A_f = 0.$$

The flow must preserve the integral $\mathrm{d}\tilde{f}(v)$ over time.

B.2 Relative Point Position

Given two points $p, q \in M$, we wish to preserve their relative position, i.e., the vector $\tilde{f}(p) - \tilde{f}(q) \in R^3$, during the deformation.

Let $g : M \to \mathbb{R}$ the solution of Poisson's equation $\Delta g = \delta_p - \delta_q$, where δ_p is the Dirac delta at point $p \in M$. Then, the integral of the ith component of the vector $\mathrm{d}\tilde{f}(\nabla g)$ represents the difference in position between points p and q:

$$\begin{aligned} \int_M \mathrm{d}\tilde{f}(\nabla g)_i\,\mathrm{d}A_f = \int_M \langle \nabla g, \nabla \tilde{f}_i\rangle\,\mathrm{d}A_f &= -\int_M \Delta g \tilde{f}_i\,\mathrm{d}A_f \\ &= -\tilde{f}_i(p) + \tilde{f}_i(q). \end{aligned}$$

Thus, the integral of $\mathrm{d}\tilde{f}(\nabla g)$ over the surface should remain constant during the flow.

C DISCRETE ENERGIES

This section is dedicated to computing a quadratic approximation of the discrete energy function $\mathcal{E} : (u, \tau) \to \mathbb{R}$ at the point $u = 0_{\mathbb{R}^{|V|}}, \tau = 0_{\mathbb{R}^{|E|}}$.

C.1 Bending Energy

We first consider the bending energy $\mathcal{E}$ as defined in Eq. (17).

Gradient. The gradient G^u_i with respect to the variable u_i is obtained by summing over all triangles $ijk \in F$ incident to vertex $i \in V$:

$$G^u_i = \sum_{ijk\in F} \tfrac{1}{3}(1-p)\|S_{ijk}\|_F^{2p} A_{ijk}.$$

Similarly, the derivative G^τ_{ij} with respect to the variable τ_{ij} is obtained by a summing over all triangles $ijk \in F$ adjacent to the edge $ij \in E$:

$$G^\tau_{ij} = -\sum_{ijk \in F} p\|S_{ijk}\|_F^{2(p-1)} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}^\top Q_{ijk} \begin{bmatrix} s_{ij} \\ s_{jk} \\ s_{ki} \end{bmatrix} A_{ijk}.$$

Here, $Q_{ijk} \in \mathbb{R}^{3\times 3}$ is the matrix storing all inner products between the three Regge basis functions defined on triangle ijk:

$$Q_{ijk} = \begin{bmatrix} \langle \mathcal{S}^k_{ij}, \mathcal{S}^k_{ij} \rangle_F & \langle \mathcal{S}^k_{ij}, \mathcal{S}^i_{jk} \rangle_F & \langle \mathcal{S}^k_{ij}, \mathcal{S}^j_{ki} \rangle_F \\ \langle \mathcal{S}^i_{jk}, \mathcal{S}^k_{ij} \rangle_F & \langle \mathcal{S}^i_{jk}, \mathcal{S}^i_{jk} \rangle_F & \langle \mathcal{S}^i_{jk}, \mathcal{S}^j_{ki} \rangle_F \\ \langle \mathcal{S}^j_{ki}, \mathcal{S}^k_{ij} \rangle_F & \langle \mathcal{S}^j_{ki}, \mathcal{S}^i_{jk} \rangle_F & \langle \mathcal{S}^j_{ki}, \mathcal{S}^j_{ki} \rangle_F \end{bmatrix}.$$

The coefficients of Q_{ijk} are trigonometric functions of the corner angles α:

$$\begin{aligned} \|\mathcal{S}^k_{ij}\|_F^2 &= 1 + \cot^2 \alpha_i^{jk} \cot^2 \alpha_j^{ki} + \tfrac{1}{2}\left(\cot \alpha_i^{jk} - \cot \alpha_j^{ki}\right)^2, \\ \langle \mathcal{S}^k_{ij}, \mathcal{S}^i_{jk} \rangle_F &= \left(\cot \alpha_k^{ij} \cot \alpha_i^{jk} - \tfrac{1}{2}\right)\left(\cot^2 \alpha_j^{ki} + 1\right). \end{aligned}$$

Hessian matrix. Given that the bending energy defined in Eq. (17) becomes non-convex when $p > 1$, we convexify its quadratic approximation by omitting the cross partial derivatives with respect to u and τ. The second derivatives with respect to u and τ results in a block-diagonal approximation of the Hessian matrix $H_{\mathcal{E}}$. The hessian matrix is assembled by summing the contributions $H_{ijk} \in \mathbb{R}^{3\times 3}$ from each triangle $ijk \in F$ to the corresponding coefficients of $H_{\mathcal{E}}$. This approach is a standard practice for finite element matrices.

The second derivatives with respect to u_i, u_j, u_k are the same value for each coefficient:

$$H^u_{ijk} = \tfrac{2}{9}(1-p)^2 \|S_{ijk}\|_F^{2p} A_{ijk} 1_{\mathbb{R}^{3\times 3}}.$$

The Hessian matrix $H_{ijk} \in \mathbb{R}^{3\times 3}$ with respect to $\tau_{ij}, \tau_{jk}, \tau_{ki}$ in triangle ijk is given by:

$$\begin{aligned} H^\tau_{ijk} &= p\|S_{ijk}\|_F^{2(p-1)} Q_{ijk} A_{ijk} \\ &\quad + 2p(p-1)\|S_{ijk}\|_F^{2(p-2)} Q_{ijk} \begin{bmatrix} s_{ij} \\ s_{jk} \\ s_{ki} \end{bmatrix} \begin{bmatrix} s_{ij} \\ s_{jk} \\ s_{ki} \end{bmatrix}^\top Q_{ijk} A_{ijk}. \end{aligned}$$

C.2 p-Willmore Energy

Let us examine the discrete p-Willmore energy $\mathcal{E} : (u, \tau) \to \mathbb{R}$ defined in Eq. (19). To evaluate this energy, it is crucial to compute the trace of the symmetric matrix T_{ijk} for each triangle ijk. As detailed in Sec. 4.1, this matrix is expressed as a linear combination of Regge basis functions whose traces are determined by the formula:

$$\mathrm{tr}\mathcal{S}^i_{jk} = 1 - \cot \alpha_j^{ki} \cot \alpha_k^{ij}.$$

Gradient. The gradient G^u_i with respect to u_i is obtained by summing over all triangles $ijk \in F$ incident to vertex $i \in V$:

$$G^u_i = \sum_{ijk \in F} \tfrac{1}{3}(1-p)\mathrm{tr}\left(S_{ijk}\right)^{2p} A_{ijk}.$$

The derivative G^τ_{ij} with respect to the variable τ_{ij} is obtained by a summing over all triangles $ijk \in F$ adjacent to the edge $ij \in E$:

$$G^\tau_{ij} = -\sum_{ijk \in F} p\mathrm{tr}\left(S_{ijk}\right)^{2p-1} \mathrm{d}_{ijk} A_{ijk},$$

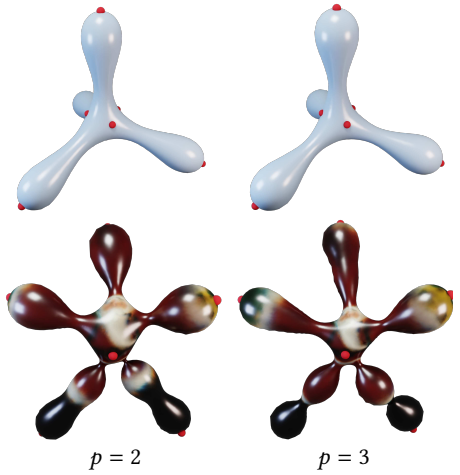


Fig. 19. Minima of the p-Willmore energy (Sec. 7.4) for p equal to 2 and 3 with positional constraints highlighted in red. In contrast to the bending energy (Fig. 8), the p-Willmore flow can converge to minima reminiscent of a constant-mean-curvature surface (bottom). We use $\xi_{\max} = 5°$ for the santa.

where $\mathrm{d}_{ijk} \in \mathbb{R}^3$ is the gradient of $\mathrm{tr}\left(\mathrm{T}_{ijk}\right)$ with respect to $\tau_{ij}, \tau_{jk}, \tau_{ki}$:

$$\mathrm{d}_{ijk} = \begin{bmatrix} 1 - \cot \alpha_i^{jk} \cot \alpha_j^{ki} \\ 1 - \cot \alpha_j^{ki} \cot \alpha_k^{ij} \\ 1 - \cot \alpha_k^{ij} \cot \alpha_i^{jk} \end{bmatrix}.$$

Hessian matrix. Similar to Section C.1, we omit the cross derivatives and assemble the Hessian matrix by aggregating derivatives from individual triangles.

The second derivatives with respect to u_i, u_j, u_k are the same value at each coefficient:

$$H^u_{ijk} = \tfrac{2}{9}(1-p)^2 \mathrm{tr}\left(S_{ijk}\right)^{2p} A_{ijk} 1_{\mathbb{R}^{3\times 3}}.$$

The Hessian matrix with respect to $\tau_{ij}, \tau_{jk}, \tau_{ki}$ in triangle ijk is the symmetric matrix:

$$H^\tau_{ijk} = p(2p-1)\mathrm{tr}\left(S_{ijk}\right)^{2p-2} \mathrm{d}_{ijk} \mathrm{d}_{ijk}^\top A_{ijk}.$$

D SPIN TRANSFORMATION

In this section, we study the theoretical connections between our method and spin transformations [Crane et al. 2011; Kamberov et al. 1998]. Our theory is rewritten with the quaternion formalism used in these articles. This naturally leads to showing that the integrability conditions of Crane et al. [2011] is implied by ours. We then conclude that both integrability conditions are equivalent in defining conformal deformations in the case of simply connected surfaces. Nevertheless, the approach introduced in this paper offers a finer control over the curvature tensor.

D.1 Conformal Deformation

In the main text, we define a deformation of the surface is simply the definition of a new immersion $\tilde{f} : M \to \mathbb{R}^3$. Any *conformal*

deformation can be defined by a log-scale factor $u : M \to \mathbb{R}$ and a rotation field $R : M \to \mathrm{SO}(3)$ as follows:

$$\mathrm{d}\tilde{f} = e^u R^\top \mathrm{d}f.$$

We proved that the Darboux derivative ω of the rotation field R satisfies the following integrability condition:

$$\vec{\omega}(X) = - * \,\mathrm{d}u(X)N - N \times \mathrm{d}f(\tau(X)),$$

where $\tau : TM \to TM$ is a *symmetric* linear map from the tangent bundle to itself.

D.2 Quaternions and conformal deformations

The key idea behind spin transformations is to represent the surface immersion $f : M \to \mathrm{Im}\mathbb{H}$ as a map from a manifold M to the space of imaginary quaternions $\mathrm{Im}\mathbb{H}$, identified with the Euclidean space $\mathbb{R}^3$. A conformal deformation of the surface is then represented by the immersion $\tilde{f} : M \to \mathrm{Im}$ characterized by:

$$\mathrm{d}\tilde{f} = \bar{\lambda}\mathrm{d}f\lambda, \tag{29}$$

where $\lambda : M \to \mathbb{H}$ is a quaternion field.

This formulation is closely related to the one presented in this paper. In particular, the Darboux derivative ω of our rotation field $R : M \to \mathrm{SO}(3)$ also appears in the exterior derivative of λ.

Lemma D.1. *If the quaternion field $\lambda : M \to \mathbb{H}$ satisfies Eq.* (29)*, then*

$$\mathrm{d}\lambda = \tfrac{1}{2}(\mathrm{d}u + \vec{\omega})\lambda, \tag{30}$$

where $u : M \to \mathrm{Re}\mathbb{H}$ is the log-scale factor and $\omega : TM \to \mathfrak{so}(3)$ is a Darboux derivative and satisfies Eq. (4)*. Here, $\vec{\omega} : TM \to \mathrm{Im}\mathbb{H}$ is a $\mathrm{Im}\mathbb{H}$-valued 1-form.*

Proof. To prove Eq. (30), we compute the Darboux derivatives of the two linear maps defining conformality: $Q_{\mathbb{R}^3}(v) \coloneqq e^u R v$ for any $v : M \to \mathbb{R}^3$ and $Q_{\mathbb{H}}(q) \coloneqq \lambda q \bar{\lambda}$ for any field of imaginary quaternions $q : M \to \mathrm{Im}\mathbb{H}$. They respectively represent the conformal deformation described in Eq. (3) and its representation in the space of quaternions depicted in Eq. (29). For any vector field v identified as a field of imaginary quaternions q, these two maps are equivalent if their Darboux derivative $\mathrm{d}Q \circ Q^{-1}$ are equal.

The Darboux derivative of $Q_{\mathbb{R}^3}$ is given by

$$\mathrm{d}Q_{\mathbb{R}^3} Q_{\mathbb{R}^3}^{-1} = (\mathrm{d}u e^u R + e^u \omega R) e^{-u} R^\top = \mathrm{d}u I + \omega. \tag{31}$$

To compute the Darboux derivative of $Q_{\mathbb{H}}$ in the space of quaternion, we need the inverse of $Q_{\mathbb{H}}$ (i.e., $Q_{\mathbb{H}}^{-1}(q) = \lambda^{-1} q \bar{\lambda}^{-1}$) and its differential (i.e., $\mathrm{d}Q_{\mathbb{H}}(q) = \mathrm{d}\lambda q \bar{\lambda} + \lambda q \mathrm{d}\bar{\lambda}$). Then, the Darboux derivative applied to $q : M \to \mathrm{Im}\mathbb{H}$ reads

$$\begin{aligned}\left(\mathrm{d}Q_{\mathbb{H}} \circ Q_{\mathbb{H}}^{-1}\right)(q) &= \mathrm{d}\lambda Q_{\mathbb{H}}^{-1}(q)\bar{\lambda} + \lambda Q_{\mathbb{H}}^{-1}(q)\mathrm{d}\bar{\lambda} \\ &= \mathrm{d}\lambda\lambda^{-1} q + q\overline{\mathrm{d}\lambda\lambda^{-1}} \\ &= 2\mathrm{Re}(\mathrm{d}\lambda\lambda^{-1})q + 2\mathrm{Im}(\mathrm{d}\lambda\lambda^{-1}) \times q.\end{aligned}$$

By comparing with Eq. (31), we observe that the real part of $2\mathrm{d}\lambda\lambda^{-1}$ corresponds to $\mathrm{d}u$, while the imaginary part corresponds to the $\mathfrak{so}(3)$-valued 1-form ω. Hence, using the vector notation $\vec{\omega} \in \mathrm{Im}\mathbb{H}$, we get: $2\mathrm{d}\lambda\lambda^{-1} = \mathrm{d}u + \vec{\omega}$.

Since Eq. (29) defines a conformal deformation, the Darboux derivative ω satisfies Eq. (4). □

D.3 Mean curvature change

We are now ready to prove that Lemma 2.3 of Kamberov et al. [1998], and later used as integrability condition by Crane et al. [2011; 2013b], is a direct consequence of Theorem 3.1.

Lemma D.2. *If the quaternion field $\lambda : M \to \mathbb{H}$ satisfies Eq.* (29)*, then Lemma 2.3 of Kamberov et al. [1998]:*

$$\mathrm{d}f \wedge \mathrm{d}\lambda = -(H - e^u \tilde{H})\lambda \,\mathrm{d}A_f, \tag{32}$$

holds true.

Proof. Using Lemma D.1 and replacing ω by Eq. (4), we obtain a characterization of λ with respect to u and τ:

$$2\mathrm{d}\lambda = (\mathrm{d}u - *\mathrm{d}uN - N\mathrm{d}f(\tau))\,\lambda, \tag{33}$$

where $N : M \to \mathrm{Im}\mathbb{H}$ is the normal field represented as an imaginary quaternion field.

To prove Eq. (32), we take the wedge product between $\mathrm{d}f$ and Eq. (33). Assuming that the immersion $f : M \to \mathrm{Im}\mathbb{H}$ is conformal, we have that $N\mathrm{d}f(X) = \mathrm{d}f(JX)$ for all tangent vectors $X \in TM$, where J is the 90° counterclockwise rotation around the normal (see [Crane 2013, p. 15]). Hence, the wedge product between $\mathrm{d}f$ and the terms depending on the scale factor u cancels out:

$$\mathrm{d}f \wedge (\mathrm{d}u - *\mathrm{d}uN) = -\mathrm{d}f(J\nabla u)\,\mathrm{d}A_f + N\mathrm{d}f(\nabla u)\,\mathrm{d}A_f = 0.$$

Using the definition of the quaternion multiplication, the last term, depending on the curvature change τ, becomes the trace of the symmetric tensor:

$$\begin{aligned}(\mathrm{d}f \wedge N\mathrm{d}f(\tau))(X,Y) &= \mathrm{d}f(X)N\mathrm{d}f(\tau(Y)) - \mathrm{d}f(Y)N\mathrm{d}f(\tau(X)) \\ &= \langle N, \mathrm{d}f(X) \times \mathrm{d}f(\tau(Y)) - \mathrm{d}f(Y) \times \mathrm{d}f(\tau(X))\rangle \\ &= \mathrm{tr}\tau\,\langle N, \mathrm{d}f(X) \times \mathrm{d}f(Y)\rangle.\end{aligned}$$

Finally, using Theorem 3.1, we obtain

$$\mathrm{d}f \wedge \mathrm{d}\lambda = -(H - e^u\tilde{H})\lambda\,\mathrm{d}A_f,$$

where $\mathrm{d}A_f(X,Y) \coloneqq \langle N, \mathrm{d}f(X) \times \mathrm{d}f(Y)\rangle$ is the volume form [Crane 2013, p. 21]. □

D.4 Equivalence

So far, we have demonstrated that our integrability condition in Eqs. (4) and (30) implies that of Crane et al. [2011] in Eq. (32). On simply connected surfaces, both conditions are equivalent.

Theorem D.3. *For simply connected surfaces, there exist $\lambda, u, \tilde{H}$ satisfying Eq.* (32) *if and only if there exist λ, u, τ satisfying Eq.* (4) *and Eq.* (30)*.*

Proof. *Necessary condition.* Suppose that Eq. (4) and Eq. (30) are satisfied by λ, u, τ and M is simply connected, then we can apply Theorem 3.2 to obtain a map $\tilde{f} : M \to \mathbb{R}^3 \equiv \mathrm{Im}\mathbb{H}$ satisfying Eq. (29). Hence, Lemma D.2 applies and Eq. (32) is satisfied.

Sufficient condition. Suppose that Lemma D.2 is satisfied by λ and u. Following Crane et al. [2011, Appendix B], Eq. (32) is equivalent to the fact that the 1-form $\bar{\lambda}\mathrm{d}f\lambda$ is closed. Therefore, if M is simply connected, the form $\bar{\lambda}\mathrm{d}f\lambda$ is also exact and there exists a map $\tilde{f} : M \to \mathrm{Im}\mathbb{H}$ satisfying Eq. (29). Since the deformation is conformal, we can apply Theorem 3.1 and there exist λ, u, τ satisfying Eq. (4) and Eq. (30). □

Table 1. Statistics for each figure: number of degrees of freedom in the finite element approximation (i.e., $|V| + |E|$), number of iterations, number of steps requiring backtracking and execution time per iterations in seconds.

		# dofs	# it.	# oversteps	time per it.
Fig. 1	$p = 1$	46850	33	0	12.9s
Fig. 1	$p = 2$	46850	50	0	12.2s
Fig. 1	$p = 3$	46850	82	0	14.6s
Fig. 2		46850	46	19	15.2s
Fig. 7	$p = 1$	34482	23	0	7.0s
Fig. 7	$p = 2$	34482	25	0	7.2s
Fig. 7	$p = 3$	34482	47	0	8.9s
Fig. 8	$p = 1$, top	15994	25	0	10.4s
Fig. 8	$p = 2$, top	15994	60	0	9.1s
Fig. 8	$p = 3$, top	15994	60	0	11.3s
Fig. 8	$p = 1$, bot.	39994	60	0	22.1s
Fig. 8	$p = 2$, bot.	39994	66	7	25.2s
Fig. 8	$p = 3$, bot.	39994	93	6	25.4s
Fig. 9	cube	79874	106	0	35.2s
Fig. 9	bob	49392	81	0	26.9s
Fig. 9	kitten	20000	65	0	7.2s
Fig. 9	bunny	57154	86	0	21.4s
Fig. 10		49392	96	0	26.3s
Fig. 11		55902	151	0	44.2s
Fig. 12		81550	11	0	34.0s
Fig. 13		39982	25	0	11.5s
Fig. 16	$p = 2$	34482	21	0	10.0s
Fig. 16	$p = 3$	34482	44	3	8.1s
Fig. 17		92942	100	0	65.7s
Fig. 19	$p = 2$, top	15994	57	0	7.4s
Fig. 19	$p = 3$, top	15994	51	1	9.9s
Fig. 19	$p = 2$, bot.	39994	58	2	18.1s
Fig. 19	$p = 3$, bot.	39994	76	1	24.2s

E STATISTICS

Statistics of execution time for each figure are reported in Table 1.