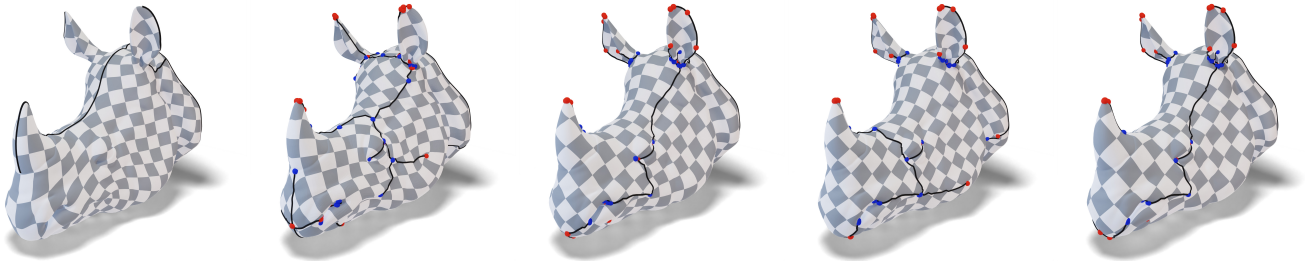
The Method of Moving Frames for Surface Global Parametrization

GUILLAUME COIFFIER and ETIENNE CORMAN, Université de Lorraine, CNRS, Inria, LORIA, France



No cones	146 cones	80 cones	86 cones	78 cones
Scale = 1.041	Scale = 1.039	Scale = 1.015	Scale = **1.000**	Scale = 1.083
Stretch = 1.413	Stretch = 1.420	Stretch = 1.188	Stretch = 1.246	Stretch = **1.079**
(a) Free Boundary	(b) Fixed singularities	(c) min ARAP	(d) min AREA	(e) min LSCM

Fig. 1. Our discretization of the moving frames method allows us to *simultaneously* place singularity cones and compute a global parametrization in a single optimization problem. This figure presents our parametrizations of the *Rhino head* model computed for a variety of constraints: (a) by preventing cones from appearing, thus leading to a free boundary parametrization with user-defined cuts, (b) by following the topology of the principal directions of curvature cross field, (c) by penalizing non-isometric deformations, (d) by minimizing area distortion, (e) by penalizing non-conformal deformations, effectively reducing stretch distortion. We represent positive $\pi/2$ cones in red and negative $-\pi/2$ cones in blue. Distortion measurements are computed as a mean over all triangles weighted by area.

This article introduces a new representation of surface global parametrization based on *Cartan's method of moving frames*. We show that a system of *structure equations*, characterizing the local coordinates changes with respect to a local frame system, completely characterizes the set of possible cone parametrizations. The discretization of this system provably provides necessary and sufficient conditions for the existence of a valid mapping. We are able to derive a versatile algorithm for surface parametrization, allowing feature constraints and singularities. As the first structure equation is independent of the global coordinate system, we do not require prior knowledge of cuts or cone positions. So, a single non-linear least-square problem is enough to place quantized cones while minimizing a given distortion energy. We are therefore able to take full advantage of the link between the parametrization geometry and the topology of its cone metric to solve challenging constrained parametrization problems.

CCS Concepts: • **Computing methodologies** → **Mesh geometry models**; **Mesh models**; Texturing.

Additional Key Words and Phrases: Global parametrization, Seamless parametrization, Singularity placement, Distortion minimization, Quadmeshing

Authors' address: Guillaume Coiffier, guillaume.coiffier@inria.fr; Etienne Corman, Université de Lorraine, CNRS, Inria, LORIA, Nancy, 54000, France, etienne.corman@cnrs.fr.

0730-0301/2023/1-ART1 $15.00
https://doi.org/10.1145/3604282

ACM Reference Format:
Guillaume Coiffier and Etienne Corman. 2023. The Method of Moving Frames for Surface Global Parametrization. *ACM Trans. Graph.* 1, 1, Article 1 (January 2023), 19 pages. https://doi.org/10.1145/3604282

1 INTRODUCTION

Computing global surface parametrizations is a central problem of geometry processing due to its numerous applications in mesh quadrangulation, texture mapping with invisible seams or solving differential equations. Depending on the problem to solve, specific constraints are often required, such as alignment with boundary, feature edges or creases, as-well-as controlled distortion.

A global parametrization can be defined as assigning a flat metric everywhere except at a few local exceptions called singularities or cones. These extraordinary points are bound to appear to compensate the topology of non-torus-like surfaces and also have a great impact on the parametrization distortion: depending on the surface curvature, the number of cones and their positions can drastically change the mesh quality. In particular, when singularity indices are integer multiple of 1/4, finding optimal positions minimizing the overall distortion is challenging and often leads to sophisticated integer programming problems.

To overcome this apparent complexity, there exists two broad classes of algorithms. The first relies on two-steps: first choosing the singularity locations using a cross field or other proxies and then compute a *global rotationally seamless parametrization*. This procedure, however, loosens the link between cone placement and the distortion of the final result. As a consequence, a quad-mesh extracted from this parametrization can present quads that stray far away from squares because its connectivity was optimized almost independently of its geometry. The second type of methods

restricts the parametrization to simpler deformations (typically conformal) allowing for simple distortion estimation but ends up using either greedy algorithms or highly specialized integer optimization algorithm for cone placement.

This is where differential geometry tools come to the rescue. *Cartan's method of moving frames* provides a rich theory to design and describe local deformations. This framework uses local frames as references, making local coordinates translation and rotation invariant [Lipman et al. 2007]. Furthermore, the *first structure equation* provides a necessary and sufficient condition to the existence of an embedded surface. This equation describes how differential coordinates should change relative to the frame's motion, effectively removing all influence of the ambient coordinate systems on the deformation.

In this paper, we extend Cartan's method to *singular* frame fields and we prove that any solution of the derived structure equations is a valid cone parametrization. Most importantly, we provide a vertex-based discretization of the smooth theory which provably preserves all its properties. The absence of a global coordinate system allows us to compute parametrizations without prior knowledge of the cut positions and to automatically place quantized cones optimizing for a given distortion energy. Moreover, we study useful user prescribed constraints, such as feature or boundary alignment and cone locations as shown in Fig. 1.

Our technical contributions include:

(1) A formalization of surface parametrization using Cartan's method of moving frames;
(2) An extension of Cartan's method to cone parametrization;
(3) A versatile tool allowing all types of constraints as-well-as simple free-boundary parametrization;
(4) An algorithm for simultaneously computing a parametrization and quantized cone positions without integer variables.

1.1 Related Work

Cartan and Geometry Processing. In geometry processing, Cartan's method of moving frames has drawn a lot of attention for its rotation and translation invariant representation of metric and curvature [Lipman et al. 2005]. In a follow-up work, Lipman et al. [2007] use this representation to design volume preserving and as-isometric-as possible deformations. However, these works do not consider the structure equations. Wang et al. [2012] also design general deformations and provide discrete conditions, akin to the structure equations, for the existence of a deformation. However, in practice, frames are computed independently of the deformation and structure equations are never used. As a consequence, these methods are limited to suboptimal solutions and if they consider parametrization as an application, cones are out of scope.

Previous work using moving frame theory mostly focuses on deformations of volumes or surfaces with normals. In this context, the second structure equation is non-linear and describing a discrete equivalent of it is in itself a challenge [Corman and Crane 2019]. In contrast, we only study intrinsic properties of surfaces where the second structure equation is a simple linear equation [Crane et al. 2010]. The additional information provided by the *first* structure equation, which relates the local frames to the local deformation, allows us to compute maps from a surface to the plane.

In this work, the discretization of the structure equations is achieved by decomposing the mesh into disjoint vertex-based charts. We show that a valid parametrization can be constructed from any solutions of these discrete equations. No such guaranty can be given when using the vertex-based discretization of differential operators proposed by [Liu et al. 2016].

Distorsion Minimizing Injective Maps. As our algorithm is a parametrization algorithm, it is interesting to consider our work as a specific instance of the broader problem of computing distortion minimizing injective maps [Hormann et al. 2008]. Given a target topology (disk [Lévy et al. 2002; Tutte 1963], orbifold [Aigerman and Lipman 2015], sphere [Kazhdan et al. 2012] or cone metric [Sawhney and Crane 2018; Springborn et al. 2008]), standard distortion minimization finds an optimal geometry in parameter space. In our setting, we are able to both find the geometry *and* act on the topology to minimize our distortion criterion. Since this problem is particularly challenging, the most common approach is to first solve for the topology independently of the geometry using a practical proxy such as cross fields.

Cross Field Based Parametrizations. A cross field is an extremely useful tool for seamless parametrization since it has the same symmetry as a square and thus fixes the element orientations. Direction fields generation is generally well understood for surface meshes. A complete review of the literature is out-of-scope of this paper and we refer to the relevant surveys [de Goes et al. 2016; Vaxman et al. 2016]. When needed, we will represent symmetric frames by the n^{th} root of a complex number [Palacios and Zhang 2007; Ray et al. 2008]. Most recent results guarantee that the smoothest cross field can be computed using a diffusion/renormalization scheme [Viertel and Osting 2019]. However, as shown by our experiments, the smoothest cross field topology is not necessarily the optimal choice to minimize the parametrization distortion.

Parametrizations based on cross fields are computed in two steps. First, the smoothing of a cross field outputs crosses over each element. This local basis can be interpreted as the rotation part of the polar decomposition of the parametrization's Jacobian matrix [Panozzo et al. 2014]. Second, the extraction of the parametrization boils down to the recovery of the missing part of the Jacobian. The simplest method is to find the functions whose gradients are closest to the field directions [Kälberer et al. 2007]. This can also be done using global periodic functions [Fang et al. 2018; Ray et al. 2006] with the advantage of constraining singularities to integer coordinates. A more stable approach traces the motorcycle graph of the field and then solves for the graph edge lengths [Myles et al. 2014]. Local injectivity can be ensured very efficiently using a generalization of Tutte's embedding [Bright et al. 2017; Campen et al. 2019].

Fu and Liu [2016] take an ingenious approach: they divide the input mesh into a set of independent triangles and constrain adjacent simplices to be equal up to the cross field rotation. This simple constraint allows them to compute seamless global parametrizations for triangle-based cross fields *without* prior knowledge of the

parametrization cuts. In this paper, we use a similar formulation but adapted to cross field defined at vertices so that singularities are not bound to appear at vertices.

While convenient, cross field based approaches have one major drawback: not all cross fields can generate a valid parametrization. In particular, some can exhibit limit cycles creating degenerated triangles in parameter space. Many solutions have been proposed to circumvent this issue. Limit cycles can be detected and fixed locally by adding a singularity dipole [Myles et al. 2014]. However, this solution stays local and does not take into account the global problem of minimizing the distortion.

Diamanti et al. [2015] provide a system of necessary and sufficient equations for the existence of a seamless cone parametrization. The Jacobian of the deformation is expressed in a symmetry invariant way using the PolyVector representation. Like ours, this representation allow them to compute a cone parametrization freely placing 1/4-index singularity minimizing a distortion criterion. However, in practice, the algorithm is prone to add multiple singularities, even when starting from a cross field with integrable singularities.

Works on Chebyshev Nets have also introduced algorithms able to place singularities at the same time as computing geometry for the simple reason that these nets do not always exist on general surfaces and often require specific cone placement. Sageman-Furnas et al. [2019] use a PolyVector representation but its integrability condition depends on the frame field matching, thus requires an integer optimization. Moreover, a post-processing step is needed to compute the parametrization as the construction is not exact. Later works, such as [Liu et al. 2020], alternate between computing a parametrization and a frame field are able to slowly move the singular points.

Maybe closer too our representation, Pluta et al. [2021] introduce the so-called "coordinate power fields". This representation simply relies on the definition of seamless parametrization: edges of adjacent triangles must be equal up to $\pi/2$ rotations. To avoid integer variables they raise edge vectors in the complex plane to the fourth power. In this setup, singularities are unlikely to appear as it would require creating a new cut along which all the variables must change. In our setting, singularities can be created easily by adjusting the frame field local rotations.

Cone Metric Deformation. Overall, a cross field is a convenient way of placing *quantized* singularities necessary for quad-meshing, however, it is not the only way to solve this problem. Most alternatives rely on conformal deformations as they offer a simple relationship between area distortion and cone positions. Greedy methods place singularities in regions of high distortion or curvature [Ben-Chen et al. 2008], sometimes using a diffusion process [Vintescu et al. 2017]. Myles and Zorin [2012] compute seamless parametrizations by incrementally flattening regions of smallest Gaussian curvature. In its follow-up [Myles and Zorin 2013], they enable feature alignment using a greedy mixed-integer algorithm operating on loops angle defect. However, these methods only use an approximation of the distortion during optimization and the actual parametrization is only computed in a post-processing step. Again, the link between distortion and cone placement is weakened. Only Springborn et al. [2008] actually use the real mapping distortion but $\pm 1/4$-index singularities are again obtained by a rounding procedure *after* finding their positions.

Soliman et al. [2018] and Fang et al. [2021] are some of the only non-greedy approaches available. They also rely on conformal deformation and are very efficient for minimizing area distortion. However, *quantized* cone points are not considered. In a follow-up [Li et al. 2022], the authors consider quarter-index cone using an ad hoc integer optimization scheme but are still limited to conformal parametrizations.

Levi [2021] proposes to compute distortion minimizing seamless maps by directly using its definition as a constraint in an integer programming solver. Unlike [Fang et al. 2021; Myles and Zorin 2013], this approach is greedy and never backtracks on previously made decisions. We instead propose a set of constraints that can be solved by any smooth optimization solver with no integer variables involved, for surfaces with and without boundaries, resulting in reduced stretch distortion and well distributed singularities.

All these methods suffer from the same problem as cross fields: not all sets of cones admit a valid parametrization. For surfaces without boundary, prescribed cones satisfying the Gauss-Bonnet theorem are always feasible [Campen et al. 2019; Levi 2022] and a few obstructions exist for controlling non-contractible cycles [Shen et al. 2022]. However, the existence of boundaries or feature curves can lead to situations where quantized cones and Gauss-Bonnet theorem are no longer sufficient conditions. It appears to us that the only way to be certain that a set of singularities is consistent with a seamless mapping is to *compute* the underlying parametrization.

2 SMOOTH FORMULATION

Let us start by describing the theoretical background used to build our global parametrization technique, which is, at its core, a discretization of Cartan's method of moving frames. This simple and intuitive framework only uses two ingredients: local frames and Jacobian matrices expressed in local coordinates. The theory provides two *structure equations* precisely describing how these two quantities must relate in order to create a valid parametrization of the surface.

In all the paper, Euclidean planes, either as parametric spaces or tangents to a surface, will be identified to the space of complex numbers $\mathbb{C}$. Orthogonal frames will be represented as elements of the space $\mathbb{U}$ of *unit* complex number.

2.1 Cartan's Method of Moving Frames

Although Cartan's theory is a very broad topic [Sharpe 1997], we will limit ourselves to the study of a mapping $f : M \to \mathbb{C}$ sending a compact smooth orientable surface M to the complex plane. As a first step, we will further ask that M has disk topology. The results of this section can be found in the translation of Cartan's lecture on Riemannian geometry [Cartan et al. 2001].

Cartan's key idea is to assign, at each point of a surface, an orthogonal frame represented by a unit complex number $z : M \to \mathbb{U}$ that will be used as a local reference system. The differential map $\mathrm{d}f : TM \to T\mathbb{C} \equiv \mathbb{C}$, mapping tangent vectors of M to vectors in parameter space, is itself projected into the local frame leading to

the complex-valued 1-form $\sigma : TM \to \mathbb{C}$ as illustrated in Fig. 2. Cartan decomposes the differential map as a product of frame and local deformation:

$$(\mathrm{d}f)_p = z_p \sigma_p. \tag{1}$$

The complex-valued 1-form σ_p is nothing more than a representation of the Jacobian of the deformation at point $p \in M$ in the coordinate system z_p.

Obviously, the frame field z and the local deformation tensor σ cannot be chosen arbitrarily and must depend on the frame's rotation. The main results of Cartan [Cartan et al. 2001, Thm. 1. p. 38] is that there exists a mapping f and a frame field z solution of the system Eqs. (1) if and only if a system of two differential equations is satisfied.

Field of local references. The only requirement on the reference field z is that it should be smooth and unit norm. Thus, locally there exists a smooth angle function θ such that $z = \exp(\imath\theta)$. The frame's speed of rotation in direction X is given by differential form $\omega(X) \coloneqq \mathrm{d}\theta(X)$. Therefore, the frame evolution is entirely captured by the differential equation:

$$\mathrm{d}z = \imath\omega z. \tag{2}$$

Eq. (2) is well-known in geometry processing for computing cross fields [Crane et al. 2010; Ray et al. 2008] or even quad or stripe patterns [Knöppel et al. 2015; Ray et al. 2006]. As proved in [Cartan et al. 2001; Sharpe 1997], the frame field is uniquely determined by its rotation speed ω.

Local deformations. The evolution of deformation form σ is by definition deeply related to the evolution of the frame on the surface as it must compensate for the frame's rotation. The *first structure equation* describes exactly this relationship and, as proved in Thm. 2.1, it is also a necessary an sufficient condition for the existence of the mapping f defined in Eq. (1).

Theorem 2.1. *Given a frame field whose rotation speed is ω, there exists a unique solution of the system (1) up to translation/rotation if and only if σ is solution of Cartan's* first structure equation*:*

$$\mathrm{d}\sigma + \imath\omega \wedge \sigma = 0. \tag{3}$$

Proof. The structure equation is equivalent to showing that $z\sigma$ is a closed-form:

$$\mathrm{d}(z\sigma) = \mathrm{d}z \wedge \sigma + z\mathrm{d}\sigma = z(\imath\omega \wedge \sigma + \mathrm{d}\sigma).$$

Since M is assumed to be disk-like, $z\sigma$ is exact if and only if it is closed.

A proof in a more general context can be found in [Cartan et al. 2001, Thm. 1, p. 38]. □

Local injectivity. If the differential map and the deformation tensor σ differ by a rotation, they still share many properties. For instance, the local change of area is invariant by rotation and can be computed directly from σ. For parametrizations, we would like to obtain locally injective maps, or stated differently that the local areas stay positive.

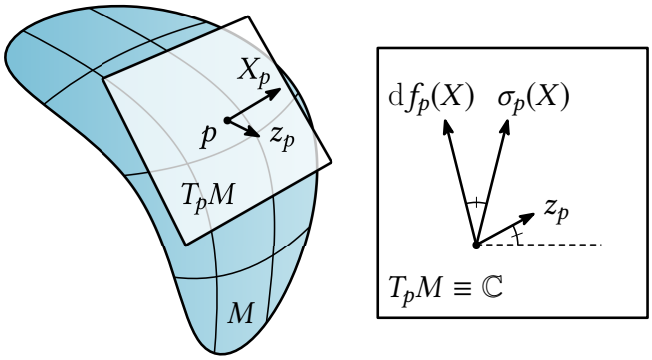


Fig. 2. Cartan's method of moving frames projects the differential map $\mathrm{d}f$ at a point $p \in M$ in a local basis $(z_p, \imath z_p)$ to obtain a deformation σ_p in the local referential.

One way to ensure this is by checking that σ does not change the sign of the cross product between tangent vectors:

$$\star\mathrm{Im}(\bar{\sigma} \wedge \sigma) > 0. \tag{4}$$

2.2 Cartan's Method of Symmetric Moving Frames

So far, Cartan's method provided us with two integrability conditions for disk-like domains. In this section, we will extend this method to *global* parametrizations by allowing cone metrics — metrics with zero Gaussian curvature everywhere except at a few singular points. We will show that simply allowing symmetric frame fields is enough to guarantee parametrization with cone angles matching the singularity indices.

Field of local references. One obstruction to extending the theory to general topology is that, in general, we cannot define a regular reference field everywhere. Thus, we will consider frame with rotational symmetry [Palacios and Zhang 2007]. Two frames are deemed equivalent if they are equal up to a rotation in the symmetry group Γ. As shown in [Crane et al. 2010], these symmetric fields still define a rotation speed ω and Eq. (1) still admits a solution but this time within the quotient space $\mathbb{U}/\Gamma$.

Local deformations. Interestingly, the first structure equation Eq. (3), obtained by differentiating Eq. (1), does not change as it only depends of the frame's rotation speed.

Now we can define a global parametrization of a "cut" manifold M^c. The surface M^c is defined by puncturing M at singular points and "cutting" the domain in order to recover a disk topology. We are able to show, in Thm. 2.2, that any solution of the first structure equation is a global parametrization of M^c.

Theorem 2.2. *Given a 1-from σ satisfying Eq. (3) and a symmetric frame field solution of Eq. (2) in the quotient space $\mathbb{U}/\Gamma$, there exists a mapping $f : M^c \to \mathbb{C}$ such that:*

$$\mathrm{d}f_p = z_p \sigma_p, \quad \forall p \in M^c.$$

Moreover, the image of a pair of vectors tangent to the cuts are equal up to a rotation of the symmetry group Γ in parameter space.

We prove Theorem 2.2 in Appendix A. The key idea is that the frame field z is continuous in the quotient space $\mathbb{U}/\Gamma$ but discontinuous over the space of rotation $\mathbb{U}$ whereas the local deformation σ is continuous everywhere away from singularities. As a consequence, the mapping $f : M^c \to \mathbb{C}$ exhibits cuts where the frame field is discontinuous. Moreover, to go from one side of the cut to the other, we need to apply a rotation equal to the frame field jump.

2.3 Parametrization Design

Following Cartan's method, computing a parametrization is equivalent to finding local frames z with rotation speed ω and a 1-form σ satisfying the structure equations. Thus, we can setup an optimization problem aiming at finding z, ω, σ minimizing a distortion energy $\mathcal{D}$ and satisfying Eqs. (2), (3) and (4):

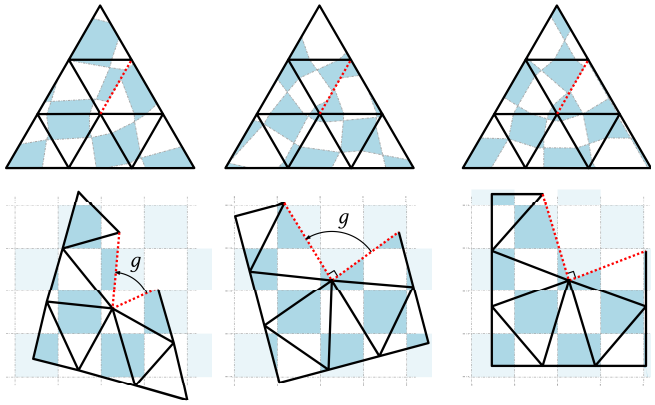
$$\begin{array}{lll} \min & \mathcal{D}(\sigma) & \\ z : M \to \mathbb{U}/\Gamma & & \mathrm{d}z - \imath\omega z = 0 \\ \omega : M \to \mathbb{R} & \text{s.t.} & \mathrm{d}\sigma + \imath\omega \wedge \sigma = 0 \\ \sigma : TM \to \mathbb{C} & & \star\mathrm{Im}(\bar{\sigma} \wedge \sigma) > 0 \end{array} \tag{5}$$

The optimization constraints ensure the existence of a global parametrization of the cut surface M^c but never use the position of the cuts or the singular points. In fact, cones are determined during optimization and cuts are discovered during the construction of the parametrization when solving Eq. (1). This formulation allows for a great range of possible constraints: boundary constraints, feature constraints, distortion measure, singularity positions as-well-as a lot of freedom to let the cone be placed according to the distortion minimization.

Feasible region. In theory, for a mapping $f : M^c \to \mathbb{C}$, there are an infinite number of pairs z, σ satisfying Thm. 2.2. In particular, one can rotate the frame and the deformation tensor at each point by a rotation $r_p \in \mathbb{U}$ such that $\bar{z}_p := r_p z_p, \bar{\sigma}_p := r_p^{-1}\sigma_p$ does not alter the deformation — i.e. $\mathrm{d}f_p = \bar{z}_p\bar{\sigma}_p$. In order to satisfy Eq. (2) and Eq. (3), the rotation speed $\bar{\omega}$ of $\bar{z}$ must change according to the formula $\bar{\omega} = \omega - \imath r^{-1}\mathrm{d}r$. Thus, for any rotation field, if z, ω, σ are feasible then $\bar{z}, \bar{\omega}, \bar{\sigma}$ are also feasible with possibly the same distortion energy. Therefore, the feasible region in Eq. (5) is extremely large and the optimization problem is highly non-convex. In practice, the size of the feasible set allows us to easily converge to a feasible solution (expects for some specific configurations discussed in Sec. 9). However, we cannot guarantee that we will find a global minimum of the distortion energy.

2.4 Overview

Our goal is to discretize the optimization problem in Eq. (5). We will first recall definitions necessary for discrete parametrization in Sec. 3. The constraints on frame rotations will be the topic of Sec. 4 and the first structure equation will be studied in Sec. 5. We will show in Sec. 6 that the discrete constraints are equivalent to the existence of a discrete parametrization. Sec. 7 discusses boundary conditions and feature alignment. Sec. 8 and Sec. 9 describe our numerical optimization scheme as-well-as our results and applications.



(a) Cone Param. (b) Seamless (c) Adapted + Seamless

Fig. 3. Three parametrization properties used in this paper illustrated by texture mapping (top) and uv-coordinates (bottom).

3 DISCRETE SETTING

In this section, we give a formal definition to three key properties of parametrization mappings used throughout the paper: cone parametrization, rotationally seamless parametrization and feature adaptation. These properties are illustrated in Fig. 3.

3.1 Discrete Parametrization

A *piecewise linear parametrization* is a map $f : \mathcal{M} \subset \mathbb{R}^3 \to \mathbb{C}$ assigning, to every triangle corner $c \in \mathbb{R}^3$ a coordinate in the (complex) plane $f_t(c) = (u, v)$. Coordinates for points inside a triangle are then linearly interpolated from its corners. Such a parametrization can exhibit *cuts* along some edges, namely adjacent triangles in the initial mesh are no longer adjacent in the parametrization. The linear functions g relating an edge on one side of the cut to its duplicated version on the other side of the cut are called *transition functions* (see Fig. 3a). Given a closed loop of triangles $(t_0, t_1, \ldots, t_p, t_0)$, all incident to the same vertex $i \in V$, the vertex is said *singular* if the accumulated transition function $g_i = g_{t_0t_p} \circ \ldots \circ g_{t_2t_1} \circ g_{t_1t_0}$ is not identity.

We say that a parametrization is *rotationally seamless* if duplicated edge vectors are equal up to a $k\pi/2$ rotation for any two adjacent triangles (Fig. 3b). More generally, a parametrization has quantized cones if all transition functions are rotations of angle $2k\pi/n$ for $n \in \mathbb{N}^+$.

Definition 3.1. A parametrization has *quantized* cones if for two adjacent triangles $t_1, t_2 \in T$ sharing an edge $ij \in E$, their edge vectors are related by the transition function:

$$f_{t_1}(p_i) - f_{t_1}(p_j) = \exp(\imath 2k\pi/n)\left(f_{t_2}(p_i) - f_{t_2}(p_j)\right), \quad k \in \{0, \ldots, n-1\}.$$

In particular, a parametrization is *rotationally seamless* if $n = 4$.

Another important property is that the surface boundaries and feature edges are isolines of the parametrization (Fig. 3c).

Definition 3.2. A parametrization is *adapted* to its boundary (or feature lines) if all boundary edges $ij \in E$ of a boundary triangle

$t \in T$ have one zero coordinate in parameter space:

$$\exists k \in \{1,2\}, \quad \langle f_t(p_i) - f_t(p_j), e_k \rangle = 0$$

where $e_1 = 1 + \imath 0, e_2 = 0 + \imath$ are the plane axes.

3.2 The Cayley Map

During the process of discretizing the integrability equations, we will see that rotations are represented using the *Cayley map* [Kobilarov et al. 2009]. By definition this map, noted $cay : \mathbb{R} \to \mathbb{U}$, associates to a real value $\alpha \in \mathbb{R}$ a complex fraction equal to the rotation of angle $2 \arctan(\alpha/2) \in (-\pi, \pi)$:

$$cay(\alpha) := \frac{1 - \imath\alpha/2}{1 + \imath\alpha/2} = \exp\left(2\imath \arctan \frac{\alpha}{2}\right). \tag{6}$$

The Cayley map is often used as an efficient way of parametrizing the space of rotations by a polynomial function [Zhang et al. 2021]. Moreover, it defines a single covering of the space of rotations which is a desirable property for numerical optimization.

4 LOCAL FRAMES

In this section we propose a discretization of our first integrability equation Eq. (2). To do so, we need to define our local frames and how to compare them on a curved surface using a parallel transport. As we consider tangent vectors defined at vertices, the material exposed in Sec. 4.2 is inspired by [Knöppel et al. 2013].

4.1 Vertex-Based Frame Fields and Singular Triangles

The choice of vertex-based frame fields can come as unusual in the context of cone parametrization. Indeed, the more common choice of frames on faces defines singularities at vertices and cuts along edges which is, in practice, very convenient. However, frames on vertices give more room for singularity placement as a cone may appear anywhere inside a triangle. Therefore, our algorithm succeeds in parametrizing poorly sampled surfaces by adding a new vertex inside each singular triangle as demonstrated in Fig. 4. Such parametrizations are inaccessible to face-based algorithms.

Frames on faces are also easy to constrain at boundaries and at feature curves. Again, these constraints are also easily accessible for vertex-based field by adapting the flattening of the local vertex chart, as we will see in Sec. 7.

Most importantly, we can prove that our framework has to create cones of same index as the frame field. This is not guaranteed in the face-based paradigm due to possible creation of unwanted "double covering" [Garanzha et al. 2022]. In any case, a face-based discretization of our integrability equations is also possible.

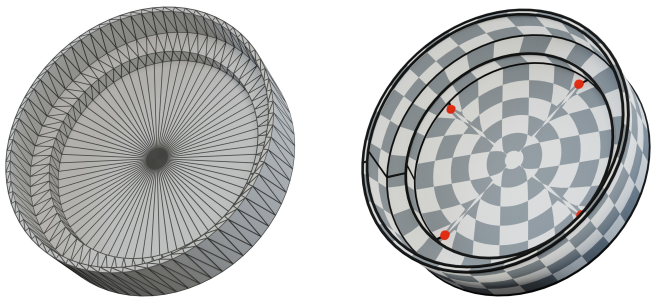


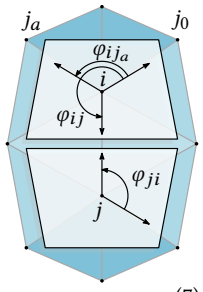
Fig. 4. Sampling local frames at vertices creates cones inside triangles. Our algorithm adds extra vertices inside singular triangles allowing us to parametrize extremely coarse surfaces. In comparison, a face-based method is unable to create a valid parametrization due to the lack of degrees of freedom.

4.2 Parallel Transport From Levi-Civita connection

Our construction of a vertex-based frame field follows the one by Knöppel et al. [2013]. A tangent vector $v_i \in \mathbb{C}$ is expressed in the local basis of the vertex i. This local basis defines a *tangent space* at this vertex. Vertices on a triangulated surface are generally not flat as the inner angles θ_i^{jk} of triangles incident to i do not sum to 2π. A local coordinate system is constructed by intrinsically "flattening" each vertex, namely inner angles are normalized:

$$\tilde{\theta}_i^{jk} := 2\pi\theta_i^{jk}/\Theta_i$$

where $\Theta_i = \sum_{ijk} \theta_i^{jk}$ is the total inner angle at vertex i. At each vertex, we assign a reference edge ij_0 whose angle coordinate φ_{ij_0} is by definition zero. The angles of other ordered edges $ij_0, \ldots, ij_n$ are obtained by accumulating modified inner angles:



$$\varphi_{ij_a} := \sum_{p=0}^{a-1} \tilde{\theta}_i^{j_p j_{p+1}}. \tag{7}$$

To compare adjacent vectors, we define the parallel transport $\rho : E \to \mathbb{R}$ as the rotation angle aligning the basis at j to the one at i:

$$\rho_{ij} := \varphi_{ji} - \varphi_{ij} + \pi,$$

by comparing the angles of the shared edge ij. The change of basis is then $r_{ij} = \exp(\imath\rho_{ij})$. By construction, this parallel transport is associated to the Levi-Civita connection. As shown by Knöppel et al. [2013], its integrated Gaussian curvature $K \in (-\pi, \pi]$ defined as

$$\exp(\imath K_{ijk}) = r_{ki} r_{jk} r_{ij}$$

satisfies the Gauss-Bonnet theorem (for meshes without boundaries).

4.3 Discrete Frame Field Condition

We are now ready to discretize the frame equation Eq. (2) using Discrete Exterior Calculus [Hirani 2003].

A frame field is an assignment of unit complex number z per vertex. So, the exterior derivative of the frame is simply the frames difference corrected by the parallel transport:

$$(\mathrm{d}z)_{ij} = z_j - r_{ij} z_i.$$

For the second part of the equation, we integrate a wedge product between a 1-form and a 0-form along a primal edge following [Hirani 2003; Ptáčková and Velho 2021]:

$$(\omega z)_{ij} = \imath\omega_{ij}(z_j + r_{ij} z_i)/2.$$

Combining these two equations, the discrete frame constraint Eq. (2) reads:

$$(1 + \imath\omega_{ij}/2) r_{ij} z_i = (1 - \imath\omega_{ij}/2) z_j. \tag{8}$$

Eq. (8) implies that the frame at vertex i is equal to the frame at vertex j up to a multiplication by $cay(\omega_{ij})$. So adjacent frames are equal up to a rotation of angle $2\arctan(\omega_{ij}/2)$. This is precisely the intuition behind the continuous equation.

4.4 Quantized Singularity Cones Fields

According to Thm. 2.2, we can define a cone parametrization if the local bases admit quantized cone singularities. This property is satisfied exactly by discrete symmetric frame fields [Ray et al. 2008].

A rotationally symmetric n-direction field (n-RoSy field [Vaxman et al. 2016]) assigns at each vertex a set of n directions:

$$\{\exp(\imath 2k\pi/n)\, z_i, \quad k = 0, \dots, n\}.$$

A key insight is that, when raised to the n^{th} power, this set reduces to a single complex value:

$$v_i := z_i^n.$$

Thus, it can be uniquely represented by a unit complex number whose n^{th} roots gives the n directions [Palacios and Zhang 2007; Ray et al. 2008]. For v to represent a n-RoSy direction field, we simply raise Eq. (8) to the power n:

$$\left| \quad (1 + \imath\omega_{ij}/2)^n r_{ij}^n v_j \quad = (1 - \imath\omega_{ij}/2)^n v_i. \right. \tag{$\mathcal{F}_n$}$$

The symmetric field may be smooth over the surface, it can still represent a discontinuous frame field. Taking the n^{th} root of Eq. ($\mathcal{F}_n$), we can quantify the rotation jumps in the frames by an integer $k_{ij} \in \mathbb{Z}$ per edge:

$$z_j = \exp\left(2\imath\pi k_{ij}/n\right) cay(\omega_{ij})^{-1} r_{ij} z_i.$$

We will see with Theorem 6.1 that a rotation jump by an angle $2\pi k_{ij}/n$ in the frame field will create a cut with the same transition functions in the parametrization.

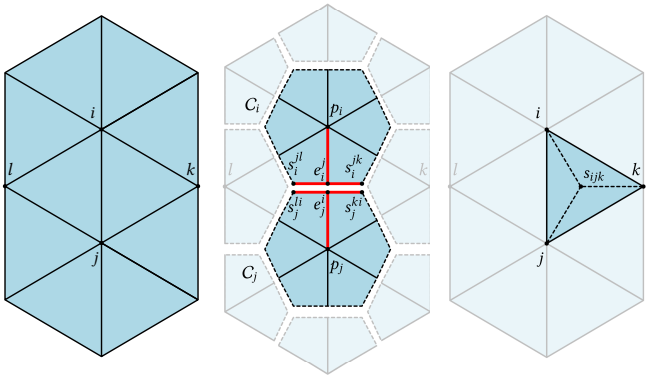
5 FIRST STRUCTURE EQUATION

We are now ready to discretize the other equation needed for integrability a.k.a. the first structure equation Eq. (3).

5.1 Vertex Charts

The 1-form σ expresses a deformation of the mesh in the coordinates of a local frame. Thus, we need to define a neighborhood or *chart* affected by this local basis.

As singularities may appear inside triangles, the charts must include a point inside each triangle physically representing the (potential) cone. For this reason, we build our charts as the intersection of mesh $\mathcal{M}$ with its dual. Namely, each triangle is split into three quads, as in Fig. 5b, by adding three edges linking the triangle center to edge midpoints. We explicitly build the chart C_i at vertex i as the union of quads containing i. In this construction, adjacent charts are disjoint but share two types of duplicated edges: primal edges from the input mesh and dual edges linking edge centers to triangle centers. Both types of edges are highlighted in red in Fig. 5b. We define the chart collection $\mathcal{M}^c = \cup_{i \in V} C_i$ as the set of all charts



(a) $\mathcal{M}$ (b) $\mathcal{M}^c$ (c) $\mathcal{M}_p$

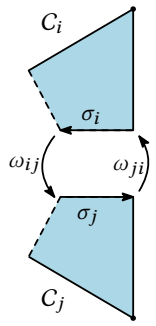
Fig. 5. The initial mesh (a) is subdivided in a chart collection (b) used for imposing integrability constraints. Edge midpoints are denoted with letter e and triangle center point with the letter s. We do not parametrize the initial mesh but a subdivision of it where singularities s are inserted inside singular triangles (c).

emanating from the subdivision of the input mesh. Effectively, we will parametrize the mesh $\mathcal{M}_p$ exactly equal to the initial mesh except at singular triangles which will be split into three to insert singular points (Fig. 5c).

In the DEC formalism, 1-forms are discretized using their integrated values along edges. By definition, the deformation of an edge vector u of $\mathcal{M}$ is given by $\sigma(u) \in \mathbb{C}$. To ease the notations, we directly use the deformed edges as variables by using specific letters for each point type of the deformed mesh. The letter p denotes the vertex coordinates (with a complex number) of the input mesh in parameter space, the letter e is the midpoint of an edge and s the center of the triangle and thus a potential singularity. Moreover, under-scripts are reserved to make explicit the membership of a vector to a chart. For example, $s_i^{jk} - e_i^j$ is a deformed dual edge in the chart C_i whereas $s_j^{ki} - e_j^i$ is its copy in the chart C_j.

5.2 Discrete Structure Equation

In order to discretize Eq. (3), we use Discrete Exterior Calculus [Hirani 2003] on "lens" complexes as introduced in [Soliman et al. 2021]. A lens complex is obtained by gluing opposite halfedges of adjacent charts. Note that σ_i is expressed in the basis z_i, thus, in order to compare two 1-forms living in adjacent charts, we must compensate for the basis rotation by multiplying by the *inverse* parallel transport. So, along a dual edge, the integral exterior derivative reads:



$$(\mathrm{d}\sigma)_{ij} := r_{ji}(s_i^{jk} - e_i^j) - (s_j^{ki} - e_j^i).$$

The wedge product between a primal 1-form ω and a lens complex can be thought of as a wedge product on a virtual quadrangle made of primal and dual edges where the frame rotations ω on dual edges are zero. Using the wedge product on a

quadrilateral element defined in [Krantz and Parks 2008; Ptáčková and Velho 2021], we obtain:

$$(\omega \wedge \sigma)_{ij} = \imath\omega_{ij}(-r_{ji}(s_i^{jk} - e_i^j) - (s_j^{ki} - e_j^i))/2.$$

When applied to both the primal and dual edges of a chart, we obtain the system of equations:

$$\left| \begin{array}{lcll} \left(1 + \imath\omega_{ij}/2\right)\left(e_j^i - p_j\right) & = - & \left(1 - \imath\omega_{ij}/2\right) r_{ji}\left(e_i^j - p_i\right), & \forall ij \in E \\ \left(1 + \imath\omega_{ij}/2\right)\left(s_j^{ki} - e_j^i\right) & = & \left(1 - \imath\omega_{ij}/2\right) r_{ji}\left(s_i^{jk} - e_i^j\right). & \forall ijk \in F \end{array} \right. \tag{$\mathcal{E}$}$$

Again, we notice that any edges shared by two adjacent charts are equal up to a rotation but this time it involves the inverse Cayley transform $cay(\omega_{ij})^{-1}$ and the inverse parallel transport.

Note that Eqs. ($\mathcal{E}$) are written only in term of the chart edges. So, without loss of generality, we can reduce the number of variables by fixing the charts translations and setting $p_i = 0$, $\forall i \in V$.

5.3 Local Injectivity

To obtain a valid parametrization, we also need to ensure local injectivity. In other words, triangle areas must remain positive in parameter space. This boils down to two different constraints. First, the orientation of primal edges in vertex charts should be preserved:

$$\left| \det(p_i - e_i^j, p_i - e_i^k) \quad > 0. \right. \tag{$\mathcal{I}_1$}$$

Second, the singularity point s_{ijk} should stay inside triangle ijk:

$$\left| \begin{array}{ll} \det(p_i - e_i^j, p_i - s_i^{jk}) & > 0, \\ \det(p_i - s_i^{jk}, p_i - e_i^k) & > 0. \end{array} \right. \tag{$\mathcal{I}_2$}$$

6 QUANTIZED CONE PARAMETRIZATION

Now that we have all integrability conditions in hand, we are ready to reconstruct the parametrization from local bases and chart coordinates. Furthermore, assuming that the charts satisfy the system of equations in Eqs. ($\mathcal{E}$) and Eq. ($\mathcal{F}_n$), we can show that it defines a valid quantized cone parametrization.

6.1 Parametrization Reconstruction

Recovering a parametrization from the chart collection $\mathcal{M}^c$, is very simple as we just need to integrate Eq. (1). This equation states that parametrized edges are simply the chart edges but rotated by the basis z:

$$(\mathrm{d}f)_i = z_i\sigma_i.$$

Since edges of two adjacent charts C_i, C_j and two adjacent frames z_i, z_j are both equal up to a rotation of $cay(\omega_{ij})r_{ij}$, the rotated charts z_iC_i, z_jC_j can be stitched together by finding a translation. However, when using a reconstruction based on vertex charts, singular triangles will be split into three quads and parametrization cuts will follow dual edges. Thus, in order to minimize connectivity changes, it is better to first reconstruct triangles from quads and then build the parametrization from triangles.

To do so, we need to recover the local bases z from their power representation v. As shown by Thm. 2.2, the discontinuities of the frame field determine the position of the parametrization cuts. Since there are n vectors satisfying $v = z^n$, the choice of the root of unity should not be random. Thus, we compute the bases along a spanning tree on quads prioritizing coherent bases on triangles so that cuts will appear on primal edges and changes in mesh topology will be minimal.

Then, we rotate the quads according to their bases z and we reconstruct all triangles independently by applying the appropriate translation to each quad. Singular triangles are triangulated by adding edges linking the singular point s and triangle vertices. However, if the singularity position is known in parameter space, its position on the surface mesh is not yet determined. To overcome this issue, we place the point s_{ijk} on the surface using the barycentric coordinates in parameter space.

The parametrization is then reconstructed by recursively stitching triangles along the spanning tree. This guarantees that cuts only appear at primal edges except for singular triangles who are split into three quads.

In summary, the parametrization is computed by following the steps:

(1) Recover a basis z per quads from the power vector v by finding the smallest symmetry rotation along a tree;
(2) Rotate each quad with their respective basis;
(3) Reconstruct triangles and remesh singular ones;
(4) Recursively stitch adjacent triangles along the tree.

This reconstruction process is summarized in Fig. 6.

6.2 Theoretical Guarantees

Crucially, we can prove that if Eqs. ($\mathcal{E}$), ($\mathcal{I}_1$), ($\mathcal{F}_n$) and ($\mathcal{I}_2$) are satisfied, the reconstruction leads to a valid quantized parametrization whose transition functions are equal to the rotational discontinuity of the frame field z. Fig. 1 shows an example of parametrization obtained by solving Cartan's structure equation.

Theorem 6.1. *If a set of charts, a frame field and its rotations satisfy the system of Eqs. ($\mathcal{E}$), ($\mathcal{F}_n$), ($\mathcal{I}_1$) and ($\mathcal{I}_2$), then we can recover a locally injective parametrization with quantized cones whose transition functions are rotation equal to the frame field discontinuities.*

The proof can be found in App. B.

7 BOUNDARY AND FEATURE EDGE CONSTRAINTS

Another requirement we discussed in Sec. 3 is that the parametrization should be *adapted* to the mesh boundary and features edges (Fig 3c). Additional constraints are therefore required on the cross field and on vertex charts. We process feature edges extracted from a boundary or user specified in the same way. We assume that the user provides us, at each vertex of a feature curve, with the target angle $\Omega_i = 2\pi k_i/n$ in parameter space.

Single vertex corner. We define a vertex corner as a set of adjacent triangles incident to the same vertex i and delimited by two feature edges ij_0 and ij_a. In this paragraph, we consider a single vertex corner. We assume that target corner angles $\Omega_i^{j_0j_a}$ integer multiple of $2\pi/n$ are provided as input. At a corner ij_0j_a, inner angles are normalized to match the prescribed angle:

$$\tilde{\theta}_i^{j_pj_{p+1}} := \left(\Omega_i^{j_0j_a}/\Theta_i^{j_0j_a}\right)\theta_i^{j_pj_{p+1}}$$

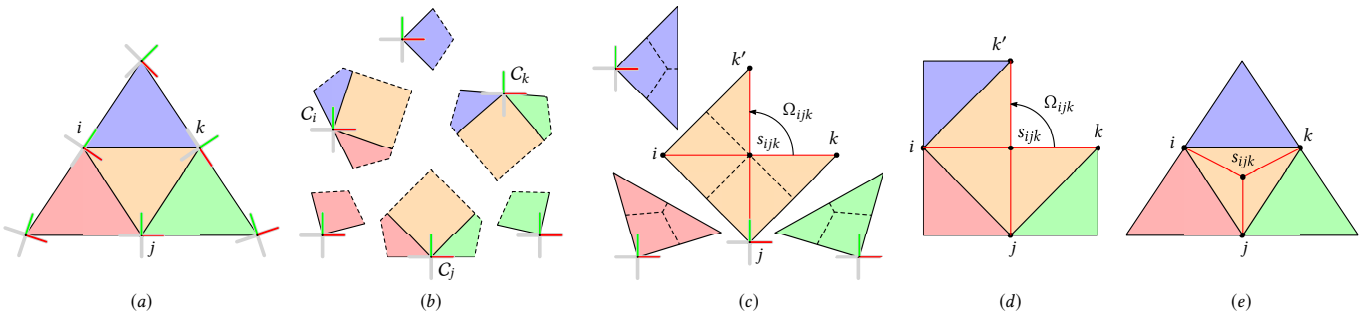


Fig. 6. Reconstruction of a parametrization from frames and charts. A triangle mesh (a) is divided into a set of vertex charts (b) satisfying the integrability conditions of Eqs. ($\mathcal{E}$), namely duplicated edges are equal up to the cross field rotation $cay(\omega)^{-1}$. The triangles are assembled independently (c). The triangle ijk is singular so the vertex s_{ijk} is added along with new edges in red. The angle Ω_{ijk} at the cut matches the discontinuity of the frame field. The parametrization (d), obtained by gluing adjacent triangles, corresponds to the mesh (e) where the singular point is inserted in triangle ijk using its barycentric coordinates in parameter space.

where $\Theta_i^{j_0 j_a}$ is the total inner angle between edges ij_0 and ij_a. The edge angles in the tangent plane are obtained by accumulating the modified inner angles as in Eq. (7). The representation vector v_i is constrained to be equal to $e^{n\imath\varphi_{ij_0}}$ so that one of its root agrees by construction with the feature edge direction. An illustration of this procedure is given in Fig. 7.

The corresponding vertex corner in chart C_i must also be constrained to the corner angle. To do so, the primal edge vector corresponding to a feature edge must remain orthogonal to the feature edge normal. Therefore, at a vertex corner, we enforce three constraints linear with respect to the chart coordinates and the cross field:

$$\left| \begin{array}{l} v_i = \exp(n\imath\varphi_{ij_0}), \\ \langle p_i - e_i^{j_0}, \imath \exp(\imath\varphi_{ij_0})\rangle = 0, \\ \langle p_i - e_i^{j_a}, \imath \exp(\imath\varphi_{ij_a})\rangle = 0. \end{array} \right. \quad (\mathcal{B})$$

Note that since the frame field is constrained on feature vertices, the rotation ω is also known along feature edges.

Singular vertices. Let us consider the case where an *interior* vertex i is incident to multiple feature edges $(ij_{a_0}, ij_{a_1}, \ldots, ij_{a_n})$ forming $n+1$ corners. If the sum of all corner angles $\Omega_i = \sum_{p=0}^{n-1} \Omega_i^{j_{a_p} j_{a_{p+1}}} + \Omega_i^{j_{a_n} j_{a_0}}$ is an integer multiple of 2π, then the flattening of each corner creates a valid vertex chart. However, when $\Omega_i \neq 2k\pi$, vertex i is *singular* and an additional seam is necessary to lay out edges in the tangent plane. Thus, we cut open the vertex neighborhood at edge ij_{a_0} introducing a duplicated edge ij'_{a_0}. For the parametrization of chart C_i to remain valid, edges on each side of the cut must be equal in length, yielding an additional linear constraint:

$$\left| \; \langle p_i - e_i^{j_{a_0}}, \exp(\imath\varphi_{ij_{a_0}})\rangle = \langle p_i - e_i^{j'_{a_0}}, \exp(\imath\varphi_{ij'_{a_0}})\rangle. \quad (\mathcal{B})$$

Fig. 8 illustrates the computation of the tangent plane at a cube corner where $\Omega_i = 3\pi/2$.

8 NUMERICAL OPTIMIZATION

So far, we have defined a system of necessary equations for computing valid global parametrizations. In this section, we describe our optimization process in order to compute the vertex charts, the frame field rotations ω and local bases z (represented, when necessary, by its power vector v) which are solutions of this system.

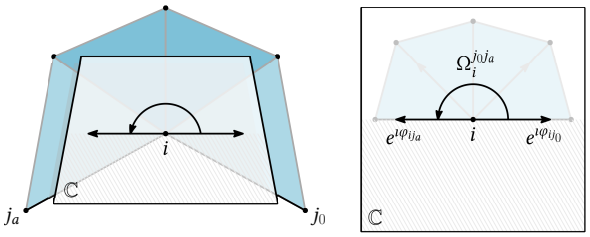


Fig. 7. Boundary edges, depicted in black, are laid out in the tangent complex plane so that they match the provided corner angle $\Omega_i^{j_0 j_a}$.

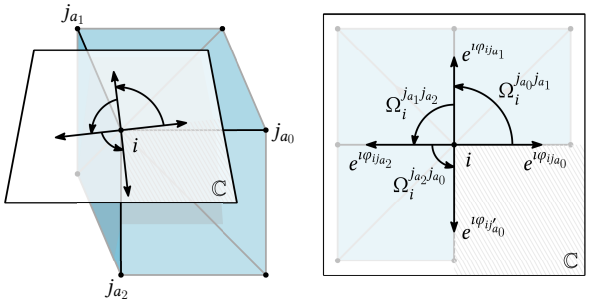


Fig. 8. Three feature edges, depicted in black, are laid out in the tangent complex plane. A seam is introduced at edge ij_{a_0} because the sum of the corner angles Ω is not an integer multiple of 2π.

8.1 Non-Linear Least-Squares

Like the smooth optimization problem of Eq. (5), we would like to find the chart vertices (x, y), the frame rotation ω and the field power vector v solution of the constrained optimization problem:

$$\begin{array}{ll} \min\limits_{\omega,v,x,y} & \mathcal{D}(x,y) \\ \text{s.t.} & (\mathcal{E}), (\mathcal{F}_n), (\mathcal{B}), (\mathcal{I}_1), (\mathcal{I}_2), \end{array} \quad (9)$$

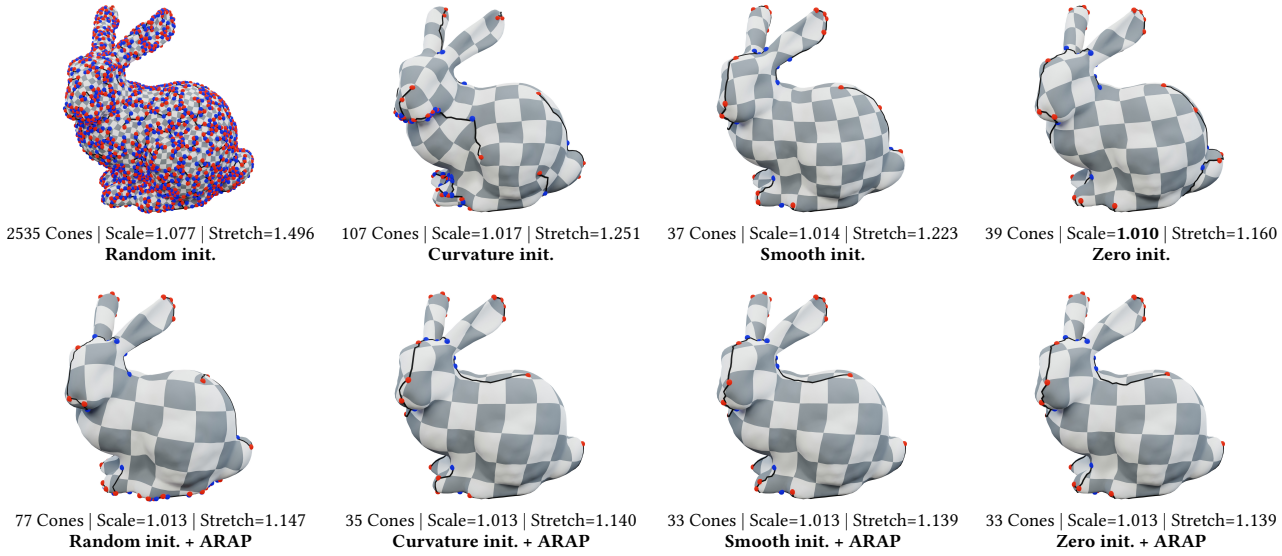


Fig. 9. Parametrization of the Stanford Bunny with different frame field initializations. From left to right: random frame per vertex, principal curvature direction, smooth frame field [Viertel and Osting 2019] and zero except on boundary vertices. The top row is optimized without any distortion energy and converges to the same singularities than the frame field. The second row minimizes the ARAP distortion (defined in Sec.8.2). For reasonable initializations, results of these optimizations share very similar cone distribution and distortion metrics. For the random initialization, undesirable singularities still cluster around the boundary with distortion minimization.

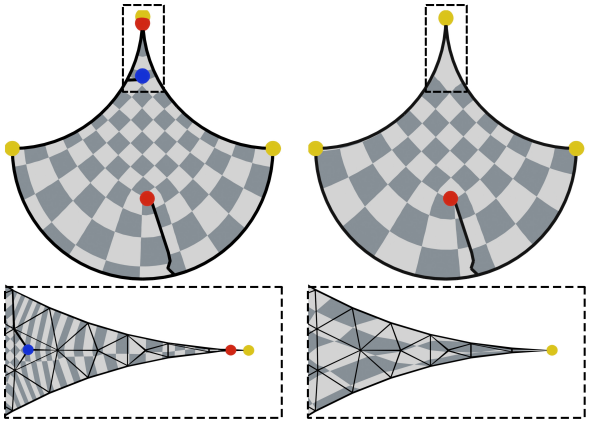


Fig. 10. Influence of the initialization of the cross field rotations on a sharp corner parametrization. Left: Initialization with zeros rotation. Right: Initialization with modified parallel transport prevents a dipole of singularities to appear and reduces overall distortion. Singularities $\frac{\pi}{2}$ are depicted in red, $-\frac{\pi}{2}$ in blue and boundary corners in yellow.

where $\mathcal{D} : \mathcal{M}^c \to \mathbb{R}$ is any given distortion energy. Note that to solve the optimization problem in Eq. (9), we do not need to know in advance the position of the cones or the cuts; everything will be decided during optimization.

To simplify the problem, the inequality constraints of Eqs. ($\mathcal{I}_1$) and ($\mathcal{I}_2$) are enforced using the continuously differentiable barrier function $B_\eta : \mathbb{R}_{>0} \to \mathbb{R}$:

$$B_\eta(x) = \begin{cases} +\infty & x \leq 0, \\ \log(\frac{x}{\eta}) & 0 < x \leq \eta, \\ 0 & \eta \leq x. \end{cases}$$

Given a determinant d whose initial value is d_0, we set the threshold to $\eta = d_0/2$.

Furthermore, we enforce non-linear constraints as a sum of squares energy. Abusing the notations, let $\mathcal{E}$ and $\mathcal{F}_n$ denote the vectors formed by the differences of left and right hand sides in equations ($\mathcal{E}$) and ($\mathcal{F}_n$). Let $\mathcal{I}_1$ and $\mathcal{I}_2$ be the vectors of determinants appearing in equations ($\mathcal{I}_1$) and ($\mathcal{I}_2$). We reformulate the constraints as the energy:

$$R(\omega, v, x, y) = \|\mathcal{E}\|_2^2 + \lambda_F \|\mathcal{F}_n\|_2^2 + \lambda_{\text{det}} \|B_\eta(\mathcal{I}_1)\|_2^2 + \lambda_{\text{det}} \|B_\eta(\mathcal{I}_2)\|_2^2,$$

where the function B_η is applied componentwise. This leads to a non-linear least-squares optimization with linear constraints:

$$\min_{\omega,v,x,y} \varepsilon \mathcal{D}(x, y) + R(\omega, v, x, y) \qquad \text{s.t. } (\mathcal{B}). \tag{10}$$

We use the Levenberg-Marquardt algorithm, an optimizer tailored for solving non-linear least squares with zero as global minimum, to minimize this objective function. Each optimization step consists in solving a quadratic optimization problem with linear constraints involving the function's Jacobian matrices. We follow the implementation of [Marumo et al. 2020] whose evolution of the damping parameter ensures global convergence toward a local minimum and a second order rate of convergence near a zero of the objective function.

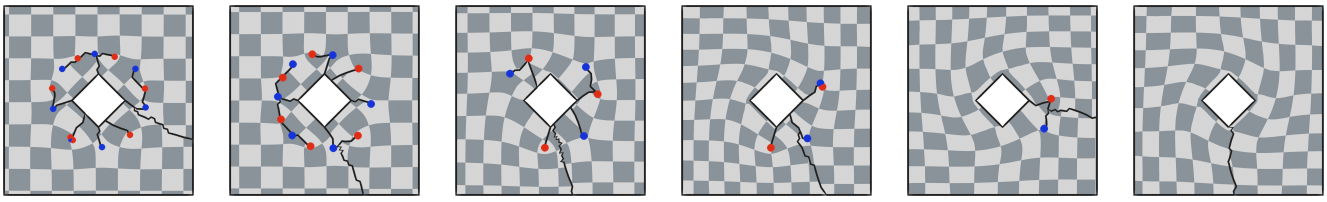


Fig. 11. Influence of the parameter λ_F on the singularity configurations when we optimize for the constraints without distortion. Small values of λ_F produce more singularity cones and minimum stretch. In contrast, large values create smoother frame fields and higher stretch.

The problem in Eq. (10) is solved several times for decreasing values of ε. For our distortion minimizing experiments, we start at $\varepsilon = 10^2$ and divide by a factor of 10 until $\varepsilon = 10^{-4}$, before solving one last time with $\varepsilon = 0$. This penalty scheme does not provide strong guarantee of convergence toward a global minimum of the distortion. However, in practice the choice of distortion energy greatly impacts singularity positions (see for example Figs. 1 and 9) and distortion measure is indeed lower and competitive with other methods (Figs. 20 or 12).

Note that for surfaces with boundary or feature edges, we do not need to force v to be unit norm: it suffices that the frame field is constrained somewhere so that whenever Eqs. (8) are satisfied, the field has unit norm everywhere. For surfaces without features, we constrain v at a random vertex to avoid the trivial zero solution.

8.2 Distortion Energy

Many differentiable distortion metrics can be optimized to obtain interesting parametrizations. In our experiments, we consider three of these metrics.

As-Rigid-As-Possible distortion. The first way to prevent charts distortion is to impose an isometric map between initial and final charts. This boils down to minimizing the so-called As-Rigid-As-Possible energy [Sorkine and Alexa 2007] which forces the map Jacobians to be orthogonal matrices:

$$\mathcal{D}_{\text{ARAP}}(x,y) = \sum_{i\in V}\sum_{t\in T_{C_i}} \|J_t^T J_t - I_2\|^2$$

We refer to this energy as the *ARAP* or *isometric* energy.

Conformal distortion. One important application of seamless parametrizations is quad remeshing. In this context, it is often desirable to avoid shearing *i.e.* quads with non-orthogonal edges as much as possible . The simplest way to promote these square-like quads is to penalize parametrizations whose parameter gradients are non-orthogonal. Thus, as a distortion energy, we use the *LSCM* [Lévy et al. 2002] energy summed over all charts:

$$\mathcal{D}_{\text{LSCM}}(x,y) = \sum_{i\in V}\sum_{t\in T_{C_i}} \|\nabla y_t - \mathcal{R}\nabla x_t\|^2,$$

where (x, y) are the chart vertex coordinates and $\mathcal{R}$ is the 90° counter-clockwise rotation around the normal. The gradient operator is computed with respect to the initial isometric chart parametrization described in Sec. 8.3.

Area distortion. Finally, one can require that the quad elements present the same overall size everywhere, which can be achieved by penalizing the change in chart area. This boils down to a constraint on the determinants of the Jacobian matrices, which should be as close as possible to 1. However, this class of area preserving deformations is not sufficient on its own as it also contains solutions with dramatic shearing. We therefore balance our area distortion term with the previously defined $\mathcal{D}_{\text{LSCM}}$:

$$\mathcal{D}_{\text{AREA}}(x,y) = \sum_{i\in V}\sum_{t\in T_{C_i}} \|\det(J_t) - 1\|^2 + \mu\mathcal{D}_{\text{LSCM}}(x,y)$$

We set μ to 0.1 in our experiments.

Metrics comparison. Fig. 1 illustrates the influence of $\mathcal{D}_{\text{ARAP}}$, $\mathcal{D}_{\text{LSCM}}$ and $\mathcal{D}_{\text{AREA}}$ on cone positions and overall parameterization geometry. We quantitatively compare the resulting parametrizations using three criteria: the number of cones, the *scale* and the *stretch* distortion.

The *scale* distortion, defined over each triangle as $(\det J + \det J^{-1})/2$, quantifies how much triangles' area changes between initial geometry and parameter space. The *stretch* distortion, defined as the ratio σ_2/σ_1 of the sorted singular values ($\sigma_2 > \sigma_1$) of the Jacobian matrices, quantifies the deformation of angles. Both metrics have an optimal value of 1. We usually present them as a integrated quantity over the entire model.

In Fig 12, we run our algorithm with distortion minimization on a large dataset of models and plot the resulting *scale* distortion against *stretch* distortion. As expected, $\mathcal{D}_{\text{LSCM}}$ reduces the stretch distortion, at a cost of a greater variation on the scaling of triangles thus producing parametrizations that are more conformal. On the other hand, minimizing $\mathcal{D}_{\text{AREA}}$ leads to parametrizations with little to no scale distortion. $\mathcal{D}_{\text{ARAP}}$ yields a good compromise between the previous two.

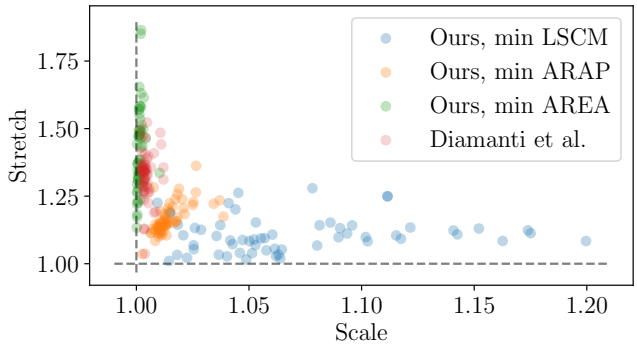


Fig. 12. Scatter plot of total stretch vs total scale observed on each models provided by [Diamanti et al. 2015]. We consider three optimization scenarios: minimization of $\mathcal{D}_{\text{ARAP}}$, $\mathcal{D}_{\text{LSCM}}$ and $\mathcal{D}_{\text{LSCM}}$.

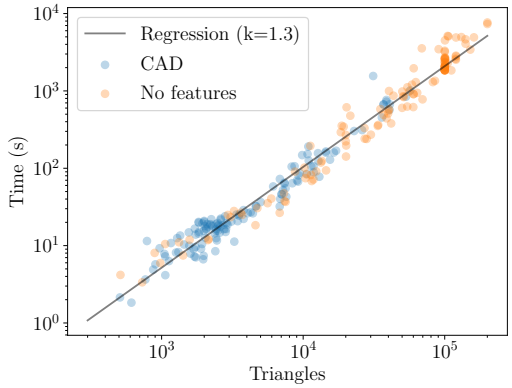


Fig. 13. Running time (in seconds) with respect to triangle count for surfaces with (142) and without (132) features.

8.3 Initialization

As for any non-convex optimization, the question of a good initialization arises and initial values for our variables x, y, ω and v should be chosen carefully.

Charts. The charts coordinates are initialized using the tangent plane flattening introduced in Secs. 4 and 7. Given the normalized triangle inner angles, the chart primal edges are laid out in the plane with their initial edge lengths. The chart is completed by initializing points s at the triangle barycenters. These initial charts keep the triangle edge lengths unchanged so they can be seen as an *isometric* parametrization of the input triangle mesh. Since the system is invariant by chart translation, the central vertex is set to zero and all coordinates in a chart are given in relation to the origin.

Rotations. For surfaces without boundaries or feature edges, the frame rotations ω are simply initialized to zero. In some cases, this initialization can lead to highly distorted solutions. Typically, for CAD models where feature edges meet at very acute angles, a singularity appears directly inside the sharp triangle. Inspired by Desobry et al. [2021], we compute an initial frame rotation ω by forcing the cross field to be regular on all triangles $T(p)$ belonging to 3-ring of a sharp corner p. In practice, we solve the quadratic problem:

$$\begin{aligned} \min_{\omega \in \mathbb{R}^{|E|}} \quad & \textstyle\sum_{ij \in E} \|\omega_{ij}\|_2^2 \\ \text{s.t.} \quad & \omega_{ij} + \omega_{jk} + \omega_{ki} = -K_{ijk}, \qquad \forall ijk \in T(p) \\ & \omega_{ij} \text{ fixed on feature edges} \end{aligned}$$

Fig. 10 illustrates how this initialization drastically changes the parametrization and reduces the distortion.

Frame field. Many different frame field initializations are possible as long as they respect boundary and feature edges constraints (see Sec. 7). In Fig. 9 (and Fig. 1 of the supplemental), we compare four initializations: a random cross field, the principal curvature directions obtained with [Cohen-Steiner and Morvan 2003], the smoothest cross field computed *via* [Viertel and Osting 2019] or zero everywhere except at boundary and feature vertices. When no distortion is minimized, the algorithm converges to the singularity configuration specified by the cross field which is the closest valid parametrization. On the other hand, any reasonable initialization will allow our algorithm to converge to a very similar cone distribution as soon as a distortion energy is minimized. This is due to the fact that the distortion energy dominates the optimization in early iterations and guides the charts towards finding the position minimizing it. The cross field and its rotations will conform with the charts when the distortion weight decreases.

8.4 Parameter Choice

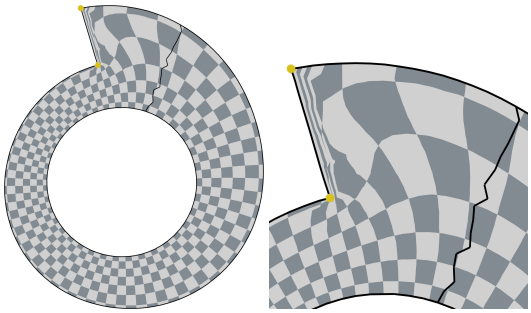
Since our objective function of Eq. (9) is a sum of several terms, their balance influences the final solution. While every global parametrization is a zero of the energy $\|\mathcal{E}\|_2^2 + \lambda_F \|\mathcal{F}_n\|_2^2$, the cone distribution depends on the value of λ_F. For small λ_F, the chart term $\|\mathcal{E}\|_2^2$ converges first. The optimizer prioritizes the validity of the parametrization over the quantization of singularities to multiple of $2\pi/n$. This results in parametrizations with more cones, and often smaller distortion. On the other hand, for large λ_F, the optimization prioritizes the convergence of the frame field variables and the quantization of rotations. The final parametrization exhibits fewer singularity cones and a smoother frame field. This phenomenon is illustrated in Fig. 11, where the same mesh is parametrized for six values of λ_F. As Myles and Zorin [2012], we observe that adding well-placed singularities tend to decrease the distortion. In our experiments in Sec. 9, our default value is $\lambda_F = 10$ representing an interesting trade-off. The other weighting term λ_{det} is set to 1.

9 APPLICATIONS AND EVALUATION

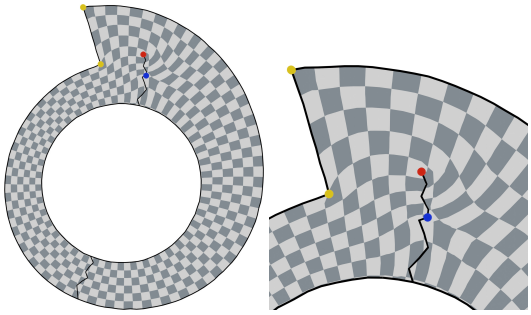
In this section, we present results obtained with our method on a large selection of models and comparison with previous works.

Database and implementation. Our results are obtained on a database of *274* triangular meshes. This includes *131 models without feature edges* gathered from [Myles et al. 2014] and [Levi 2021] or created manually for specific test cases (like the model of Figure 14). Additionally, we collected *143 CAD models* with correct feature edge flagging from the Mambo CAD dataset[1] as well as models with

[1] https://gitlab.com/franck.ledoux/mambo

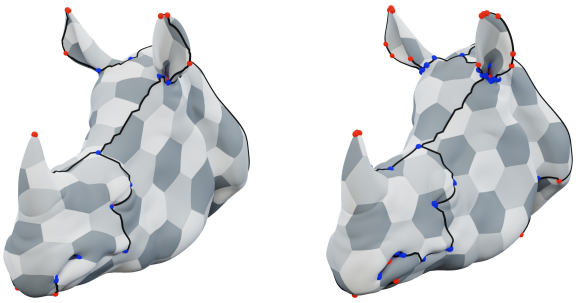


(a) Typical failure case of our method.



(b) Changing the energy weights allows our method to place a dipole of singularities and retrieve a correct parametrization.

Fig. 14. On this model our method fails to find a valid parametrization ($\lambda_F = 10$) (a). By setting $\lambda_F = 0.01$, we are able to find a valid solution (b).

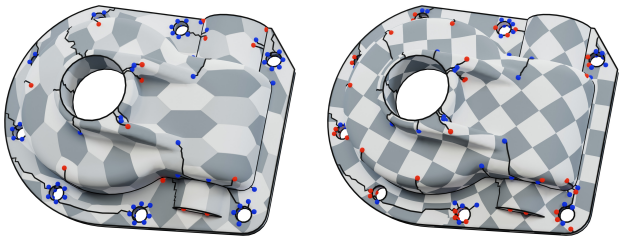


55 Cones
Scale=1.019
Stretch=1.206

(a) *Rhino*, $n = 3$

138 Cones
Scale=1.004
Stretch=1.087

(b) *Rhino*, $n = 6$



160 Cones
Scale=1.019
Stretch=1.295

(c) *Casting*, $n = 6$

166 Cones
Scale=1.006
Stretch=1.143

(d) *Casting*, $n = 8$

Fig. 15. Results of our algorithm on two models for different singularity cones quantized to $\pm 2\pi/n$ for $n = 3, 6$ and 8. On the bottom row, the parametrization of the *Casting* model is aligned with its feature edges, on which odd values of n are incompatible due to π not being a valid corner angle. When n is multiple of 3, we represent the parametrizations using a hexagonal pattern, instead of the classical checker pattern.

sharp edges from [Myles et al. 2014]. A complete gallery of our results is available in the supplemental material.

All our experiments were conducted on a Ubuntu 18.04 workstation with a eight-core, 2.6-GHz Intel Core i5. Our implementation was done in python and accelerated with Numba [Lam et al. 2015]. We solve the quadratic programming problems inside the Levenberg-Marquardt algorithm using the open source solver OSQP [Stellato et al. 2020] linked to the Pardiso linear solver [Alappat et al. 2020; Bollhöfer et al. 2019, 2020]. Computation times for the optimization without distortion are displayed in Fig. 13.

9.1 Seamless Parametrization

We first setup our algorithm to compute a rotationally seamless parametrization for all of the 274 models. Comprehensive results are available in supplemental materials including 3D models with uv-coordinates as well as renders and distortion evaluation. For the 131 models with no features, we output a valid parametrization for 123 of them. Since we need to avoid the trivial zero solution, we initialize our optimization with a smooth frame field (as described in Section 8.3). For the 143 models with feature edges, we obtain 132 valid results. We initialize those models with fixed frames on feature edges and zero elsewhere.

The 19 failure cases observed in the whole database are of two types.

1. Low quality triangles. 17 of the 19 failed models present flat or nearly-flat triangles, with angles as low as 1e-4 radians. This creates numerical instabilities or infinite values in the log-barrier B_η (Eq. (8.1)). A simple local remeshing to get rid of bad quality elements was sufficient to ensure convergence.

2. Limit Cycles. The two remaining cases are due to topological failure preventing the convergence of the optimization scheme. This is typically created by a frame field with a *limit cycle* whose parametrization is degenerated. Fig. 14a shows an example of such a configuration. In this case, the algorithm is not able to converge towards a zero of the objective function but tries to reach a global minimum at infinity. Standard parameters of our optimization are not able to escape this situation. However, we can lower the weight on the frame field constraint ($\mathcal{F}_n$) to $\lambda_F = 0.01$, so that the edge constraint ($\mathcal{E}$) dominates the energy (see Section 8.4). This way our algorithm converges to a global minimum and introduces new singularities as in Fig. 14b.

9.2 Other Applications

Besides seamless parametrization, we now demonstrate the versatility of our method through a series of other applications.

Cone placement of arbitrary indices. Using Equation ($\mathcal{F}_n$), we can constrain cone singularities to any user-chosen value of $\pm 2\pi/n$ by

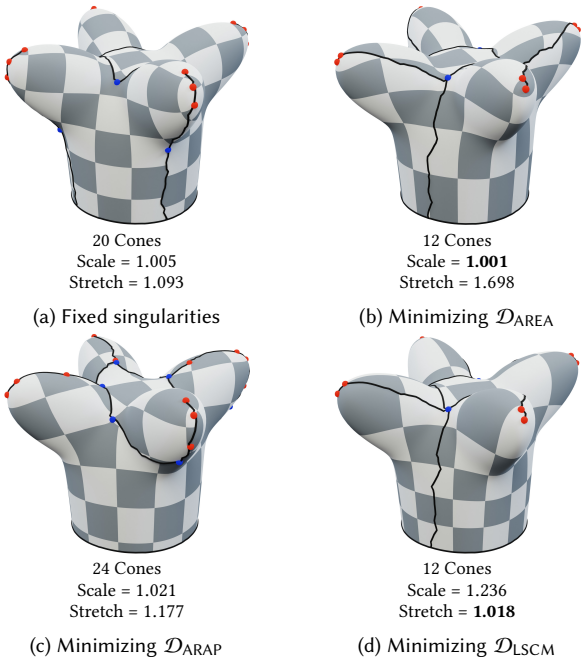


Fig. 16. Comparison between a parametrization with fixed cone singularities from a smooth frame field [Viertel and Osting 2019] and the one obtained by optimizing cones placement. Minimizing distortion leads to different cone placements. (b) Minimizing $\mathcal{D}_{\text{AREA}}$ effectively reduces the scale distortion (c) Minimizing the ARAP energy results in a tradeoff with a different topology. (d) Minimizing the LSCM energy outputs a parametrization that is almost conformal.

changing the parameter n. While choosing $n = 4$ leads to rotationally seamless maps used for quad remeshing, higher order can be easily computed by our method with minimal change. On Figure 15, we present two models, one with feature and one without, that are parametrized using $n = 3$,6 and 8. Higher values of n naturally result in a greater number of singularity cones, as well as less distorted parametrizations at the expense of more cones and seams.

Fixed topology parametrization. Finally, it is possible to lock the frame field and rotations variables v and ω during optimization in order to compute parametrization with a user prescribed cone distribution and alignment constraints. Fig. 1b shows a parametrization of Rhino's head based on the principal curvature directions.

This allows us to investigate the benefits of simultaneously optimizing for the cone distribution and the parametrization geometry. In Figure 16, we run our algorithm with the *fixed* frame field generated by Viertel and Osting [2019]. We compare the resulting parametrization to our full algorithm using the three distortions of Section 8.2. We observe that, by minimizing LSCM, our method outputs a parametrization with less cones that better minimizes the *stretch* distortion at the price of more *scale* distortion. The opposite holds when minimizing the area energy. Again, the ARAP energy yields the most isometric deformation and thus balances both distortions. Interestingly, a smooth frame field is also an excellent proxy to obtain area-preserving maps.

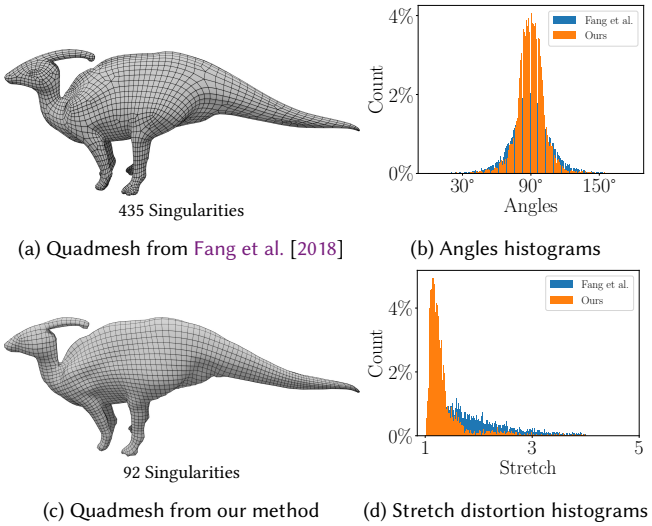
Fig. 17. Free boundary parametrization using our algorithm on disk-shaped models (top), or models with user-prescribed seams (bottom).

Free boundary parametrization. So far, we focused on computing a parametrization with a cone metric where cones are constrained to have $2\pi/n$ curvature, with $n > 1$. By setting $n = 1$ in Eq. ($\mathcal{F}_n$), cones are therefore forced to take values of either 2π, which implies that at least one chart will be flattened or flipped, thus contradicting the injectivity condition of Eq. ($\mathcal{I}_1$), or -2π, which implies a "double covering" which is impossible in our setting (see App. B). Therefore, no singularity can appear in a parameterization that satisfies Eq. ($\mathcal{F}_n$) for $n = 1$. Besides, by omitting boundary constraints of Eq. ($\mathcal{B}$) in the optimization, our algorithm is able to produce free boundary parametrization with minimal distortion for disk-shaped models. Fig. 17 presents four of such parametrizations.

9.3 Comparison With Other Methods

We now provide comparisons with previous works computing rotationally seamless maps for $n = 4$ by considering our two distortion metrics *scale* and *stretch*. For a more precise analysis than just an integrated value, we plot distortion histograms in log scale of the triangle count. When quad meshes are provided, they are obtained by computing an integer grid map using [Bommes et al. 2013] and by then extracting quads with [Ebke et al. 2013]. Most importantly, they have the same singularities (positions and indices) as the input parametrizations. More results and comparisons can be found in the supplemental material.

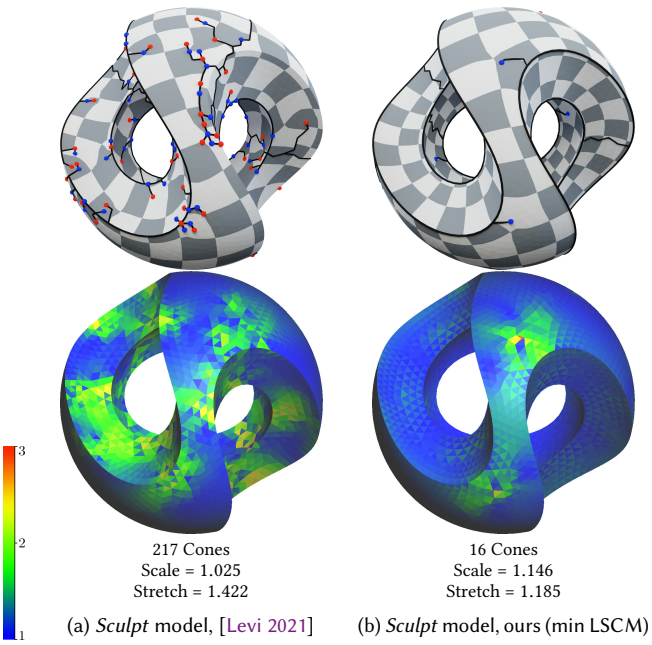
Comparison with Levi [2021]. Levi directly computes a seamless parametrization from a triangle mesh using a greedy algorithm. However, this method seems very dependent on its initialization: for the *Dilo* model in Fig. 20, the singularities appear on a single side of the mesh and seem unrelated to the geometry. The parametrizations provided by the author minimize the maximum stretch over all triangles. This can be observed in Fig. 19 where the stretch is bounded. In comparison, our method yields a higher maximal value



(a) Quadmesh from Fang et al. [2018] (b) Angles histograms

(c) Quadmesh from our method (d) Stretch distortion histograms

Fig. 18. Quad remeshing comparison with Fang et al. [2018] on the *Dilo* model. Our parametrization was initialized using principal curvature direction and optimized without distortion. Our quads have less stretching and have angles closer to 90°.



(a) *Sculpt* model, [Levi 2021] (b) *Sculpt* model, ours (min LSCM)

Fig. 19. Comparison with Levi [2021] on the sculpt model. Both methods minimize the stretch distortion. As shown by the heat map and the histogram, our method achieves comparable triangle stretch with ten times less cone singularities.

but a smaller stretch for a majority of triangles (see histograms), as well as a reduced number of cone singularities – 16 against 217. Our optimization does not attempt to reduce triangle scaling in this case, thus our scale distortion is not as competitive.

More in depth comparisons are available in supplemental material.

Comparison with Diamanti et al. [2015]. The integrable Polyvector field method takes an initial frame field and optimizes it to guarantee its integrability. However, this process is prone to introduce new singularities. In Fig. 12, we compare our method on the 62 meshes provided by the authors. The integrable Polyvector field method performs very well for scale minimization. However, we are able to outperform them while placing 15% less cones on average.

Comparison with Myles and Zorin [2012]. Myles and Zorin propose a greedy algorithm for cone placement along with a post-processing procedure to obtain $n\pi/2$ singularity cones. Cones tend to be placed in high curvature regions which can be suboptimal. In Figs. 20, our method is shown to match their scale distortion and outperform their stretch distortion on the *Dilo* model.

Comparison with Fang et al. [2018]. Finally, we compare our results to a method producing quad-meshes without the quantization step. Fang et al. [2018] use periodic functions to integrate a frame field directly into an integer seamless parametrization. This type of methods generally output a high quality mesh on most of the input surface. However, some localized regions need to be repaired which often introduce new singularities.

Fig. 18 shows the repartition of quad angles and stretch for our respective quad meshes of the *Dilo* model. Our quads are more orthogonal and suffer less from stretching, while our meshes present fewer singular vertices. An additional comparison of quad generation is provided in Sec. 2 of the supplemental material.

10 CONCLUSION AND FUTURE WORKS

By discretizing Cartan's structure equations, we were able to construct a general algorithm for global parametrization where cones and seams do not need to be known in advance, but can instead be computed along with the geometry. This all-in-one optimization of the topology and the geometry enables us to find cone points whose positions minimize any given distortion energy. Unlike most previous methods our solver is not greedy and does not rely on integer variables.

In the perspective of quad meshing, the main limitation of our work is that it does not handle integer coordinates of cone points. We still rely on a post-processing quantization step which impacts the overall distortion distribution. While the cone placement is optimal for the seamless parametrization, there is no guarantee of optimality for the final quadmesh.

A second limitation is that our current implementation is slower than previous methods because of the a higher number of variables – $2|F|+4|E|$ for vertex charts coordinates, $|E|$ for the parallel transport and $2|V|$ for the cross field.

Finally, since Cartan's method of moving frame is not limited to two-dimensional geometry, the structure equations can be written for higher dimension meshes. This opens the door to future works for computing valid seamless parametrization for tetrahedral meshes in the context of hexahedral remeshing.

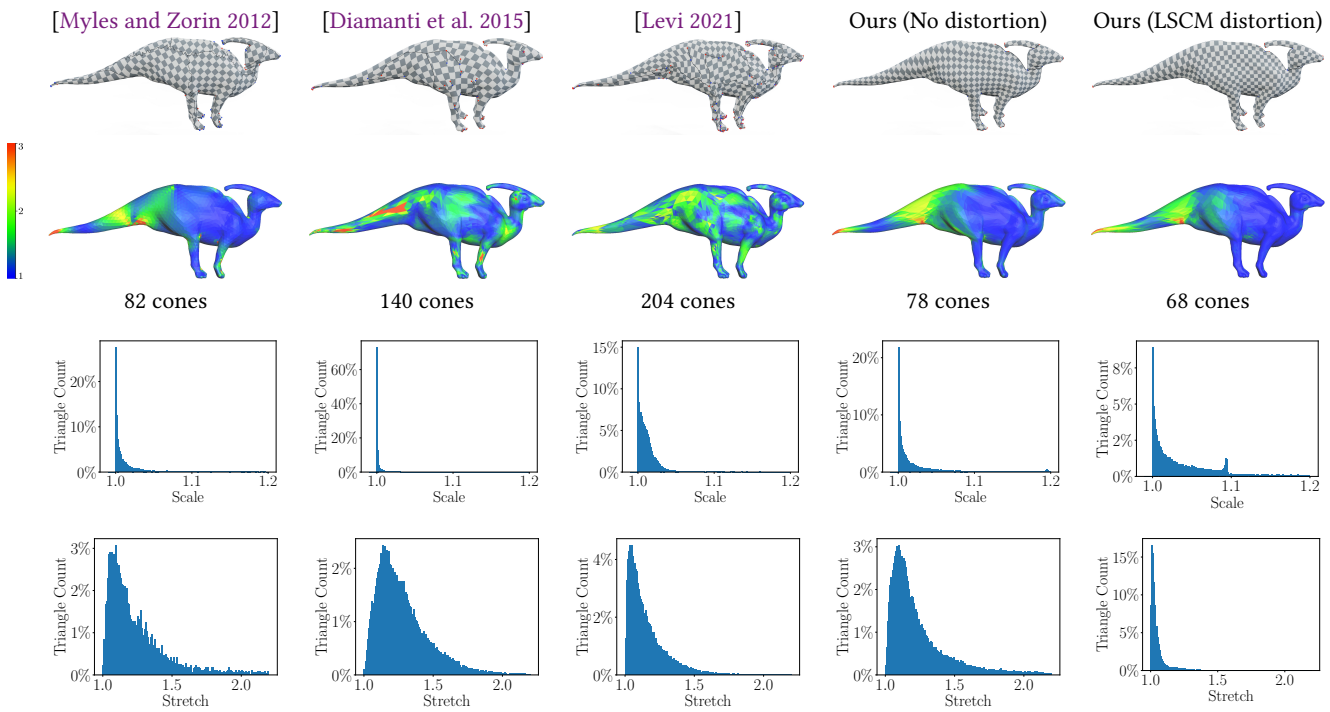


Fig. 20. Comparison of three baseline methods with our results on the dilo model. Top row: texture mapping with highlighted seams and singularities . Second row: stretch distortion distribution over the model from 1 (blue) to 3 or higher (red). Third row: histogram of stretch distortion on triangles. Our method is best at minimizing the stretch distortion even compared to the conformal algorithm [Myles and Zorin 2012].

REFERENCES

Noam Aigerman and Yaron Lipman. 2015. Orbifold tutte embeddings. *ACM Trans. Graph.* 34, 6 (2015), 190–1.

Christie Alappat, Achim Basermann, Alan R. Bishop, Holger Fehske, Georg Hager, Olaf Schenk, Jonas Thies, and Gerhard Wellein. 2020. A Recursive Algebraic Coloring Technique for Hardware-Efficient Symmetric Sparse Matrix-Vector Multiplication. *ACM Trans. Parallel Comput.* 7, 3, Article 19 (June 2020), 37 pages. https://doi.org/10.1145/3399732

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.

Matthias Bollhöfer, Aryan Eftekhari, Simon Scheidegger, and Olaf Schenk. 2019. Large-scale Sparse Inverse Covariance Matrix Estimation. *SIAM Journal on Scientific Computing* 41, 1 (2019), A380–A401. https://doi.org/10.1137/17M1147615 arXiv:https://doi.org/10.1137/17M1147615

Matthias Bollhöfer, Olaf Schenk, Radim Janalik, Steve Hamm, and Kiran Gullapalli. 2020. State-of-the-Art Sparse Direct Solvers. In *Parallel Algorithms in Computational Science and Engineering*, Ananth Grama and Ahmed H. Sameh (Eds.). Springer International Publishing, Cham, 3–33. https://doi.org/10.1007/978-3-030-43736-7_1

David Bommes, Marcel Campen, Hans-Christian Ebke, Pierre Alliez, and Leif Kobbelt. 2013. Integer-grid maps for reliable quad meshing. *ACM Transactions on Graphics (TOG)* 32, 4 (2013), 1–12.

Alon Bright, Edward Chien, and Ofir Weber. 2017. Harmonic global parametrization with rational holonomy. *ACM Transactions on Graphics (TOG)* 36, 4 (2017), 1–15.

Marcel Campen, Hanxiao Shen, Jiaran Zhou, and Denis Zorin. 2019. Seamless Parametrization with Arbitrary Cones for Arbitrary Genus. *ACM Transactions on Graphics* 39, 1 (Dec. 2019), 2:1–2:19. https://doi.org/10.1145/3360511

Élie Cartan et al. 2001. *Riemannian Geometry in an Orthogonal Frame: From Lectures Delivered by Élie Cartan at the Sorbonne in 1926-1927.* World Scientific.

David Cohen-Steiner and Jean-Marie Morvan. 2003. Restricted Delaunay Triangulations and Normal Cycle. In *Proceedings of the Nineteenth Annual Symposium on Computational Geometry (SCG '03)*. Association for Computing Machinery, New York, NY, USA, 312–321. https://doi.org/10.1145/777792.777839

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

Keenan Crane, Mathieu Desbrun, and Peter Schröder. 2010. Trivial connections on discrete surfaces. In *Computer Graphics Forum*, Vol. 29. Wiley Online Library, 1525–1533.

Fernando de Goes, Mathieu Desbrun, and Yiying Tong. 2016. Vector field processing on triangle meshes. In *ACM SIGGRAPH 2016 Courses*. 1–49.

David Desobry, François Protais, Nicolas Ray, Etienne Corman, and Dmitry Sokolov. 2021. Frame Fields for CAD Models. In *International Symposium on Visual Computing*. Springer, 421–434.

Olga Diamanti, Amir Vaxman, Daniele Panozzo, and Olga Sorkine-Hornung. 2015. Integrable polyvector fields. *ACM Transactions on Graphics (TOG)* 34, 4 (2015), 1–12.

Hans-Christian Ebke, David Bommes, Marcel Campen, and Leif Kobbelt. 2013. QEx: Robust quad mesh extraction. *ACM Transactions on Graphics (TOG)* 32, 6 (2013), 1–10.

Qing Fang, Wenqing Ouyang, Mo Li, Ligang Liu, and Xiao-Ming Fu. 2021. Computing sparse cones with bounded distortion for conformal parameterizations. *ACM Transactions on Graphics (TOG)* 40, 6 (2021), 1–9.

Xianzhong Fang, Hujun Bao, Yiying Tong, Mathieu Desbrun, and Jin Huang. 2018. Quadrangulation through morse-parameterization hybridization. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–15.

Xiao-Ming Fu and Yang Liu. 2016. Computing inversion-free mappings by simplex assembly. *ACM Transactions on Graphics (TOG)* 35, 6 (2016), 1–12.

Vladimir Garanzha, Igor Kaporin, Liudmila Kudryavtseva, François Protais, David Desobry, and Dmitry Sokolov. 2022. Practical lowest distortion mapping. *arXiv preprint arXiv:2201.12112* (2022).

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

Kai Hormann, Konrad Polthier, and Alia Sheffer. 2008. Mesh Parameterization: Theory and Practice. In *ACM SIGGRAPH ASIA 2008 Courses* (Singapore) *(SIGGRAPH Asia '08)*. Association for Computing Machinery, New York, NY, USA, Article 12, 87 pages. https://doi.org/10.1145/1508044.1508091

Felix Kälberer, Matthias Nieser, and Konrad Polthier. 2007. Quadcover-surface parameterization using branched coverings. In *Computer graphics forum*, Vol. 26. Wiley Online Library, 375–384.

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).
Felix Knöppel, Keenan Crane, Ulrich Pinkall, and Peter Schröder. 2015. Stripe Patterns on Surfaces. *ACM Trans. Graph.* 34, 4 (2015).
Marin Kobilarov, Keenan Crane, and Mathieu Desbrun. 2009. Lie group integrators for animation and control of vehicles. *ACM transactions on Graphics (TOG)* 28, 2 (2009), 1–14.
Steven G Krantz and Harold R Parks. 2008. *Geometric integration theory*. Springer Science & Business Media.
Siu Kwan Lam, Antoine Pitrou, and Stanley Seibert. 2015. Numba: A llvm-based python jit compiler. In *Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC*. 1–6.
Zohar Levi. 2021. Direct Seamless Parametrization. *ACM Transactions on Graphics (TOG)* 40, 1 (2021), 1–14.
Zohar Levi. 2022. Seamless Parametrization of Spheres with Controlled Singularities. In *Computer Graphics Forum*, Vol. 41. Wiley Online Library, 57–68.
Bruno Lévy, Sylvain Petitjean, Nicolas Ray, and Jérome Maillot. 2002. Least squares conformal maps for automatic texture atlas generation. *ACM transactions on graphics (TOG)* 21, 3 (2002), 362–371.
Mo Li, Qing Fang, Wenqing Ouyang, Ligang Liu, and Xiao-Ming Fu. 2022. Computing sparse integer-constrained cones for conformal parameterizations. *ACM Transactions on Graphics (TOG)* 41, 4 (2022), 1–13.
Yaron Lipman, Daniel Cohen-Or, Ran Gal, and David Levin. 2007. Volume and shape preservation via moving frame manipulation. *ACM Transactions on Graphics (TOG)* 26, 1 (2007), 5–es.
Yaron Lipman, Olga Sorkine, David Levin, and Daniel Cohen-Or. 2005. Linear rotation-invariant coordinates for meshes. *ACM Transactions on Graphics (ToG)* 24, 3 (2005), 479–487.
Beibei Liu, Yiying Tong, Fernando De Goes, and Mathieu Desbrun. 2016. Discrete connection and covariant derivative for vector field analysis and design. *ACM Transactions on Graphics (TOG)* 35, 3 (2016), 1–17.
Hao-Yu Liu, Zhong-Yuan Liu, Zheng-Yu Zhao, Ligang Liu, and Xiao-Ming Fu. 2020. Practical fabrication of discrete chebyshev nets. In *Computer Graphics Forum*, Vol. 39. Wiley Online Library, 13–26.
Naoki Marumo, Takayuki Okuno, and Akiko Takeda. 2020. Constrained Levenberg-Marquardt method with global complexity bound. *arXiv preprint arXiv:2004.08259* (2020).
Ashish Myles, Nico Pietroni, and Denis Zorin. 2014. Robust field-aligned global parametrization. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 1–14.
Ashish Myles and Denis Zorin. 2012. Global parametrization by incremental flattening. *ACM Transactions on Graphics (TOG)* 31, 4 (2012), 1–11.
Ashish Myles and Denis Zorin. 2013. Controlled-distortion constrained global parametrization. *ACM Transactions on Graphics (TOG)* 32, 4 (2013), 1–14.
Jonathan Palacios and Eugene Zhang. 2007. Rotational symmetry field design on surfaces. *ACM Transactions on Graphics (TOG)* 26, 3 (2007), 55–es.
Daniele Panozzo, Enrico Puppo, Marco Tarini, and Olga Sorkine-Hornung. 2014. Frame fields: Anisotropic and non-orthogonal cross fields. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 1–11.
Kacper Pluta, Michal Edelstein, Amir Vaxman, and Mirela Ben-Chen. 2021. PH-CPF: planar hexagonal meshing using coordinate power fields. *ACM Transactions on Graphics (TOG)* 40, 4 (2021), 1–19.
Lenka Ptáčková and Luiz Velho. 2021. A simple and complete discrete exterior calculus on general polygonal meshes. *Computer Aided Geometric Design* 88 (2021), 102002.
Nicolas Ray, Wan Chiu Li, Bruno Lévy, Alla Sheffer, and Pierre Alliez. 2006. Periodic global parameterization. *ACM Transactions on Graphics (TOG)* 25, 4 (2006), 1460–1485.
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.
Andrew O Sageman-Furnas, Albert Chern, Mirela Ben-Chen, and Amir Vaxman. 2019. Chebyshev nets from commuting polyvector fields. *ACM Transactions on Graphics (TOG)* 38, 6 (2019), 1–16.
Rohan Sawhney and Keenan Crane. 2018. Boundary first flattening. *ACM Transactions on Graphics (ToG)* 37, 1 (2018), 5.
RW Sharpe. 1997. Cartans Generalization of Kleins Erlangen Program. *Differential geometry, Graduate Texts in Mathematics* 166 (1997).
Hanxiao Shen, Leyi Zhu, Ryan Capouellez, Daniele Panozzo, Marcel Campen, and Denis Zorin. 2022. Which cross fields can be quadrangulated? global parameterization from prescribed holonomy signatures. *ACM Transactions on Graphics (TOG)* 41, 4 (2022), 1–12.
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.
Yousuf Soliman, Dejan Slepčev, and Keenan Crane. 2018. Optimal cone singularities for conformal flattening. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–17.
Olga Sorkine and Marc Alexa. 2007. As-rigid-as-possible surface modeling. In *Symposium on Geometry processing*, Vol. 4. 109–116.
Boris Springborn, Peter Schröder, and Ulrich Pinkall. 2008. Conformal equivalence of triangle meshes. In *ACM SIGGRAPH 2008 papers*. 1–11.
B. Stellato, G. Banjac, P. Goulart, A. Bemporad, and S. Boyd. 2020. OSQP: an operator splitting solver for quadratic programs. *Mathematical Programming Computation* 12, 4 (2020), 637–672. https://doi.org/10.1007/s12532-020-00179-2
William Thomas Tutte. 1963. How to draw a graph. *Proceedings of the London Mathematical Society* 3, 1 (1963), 743–767.
Amir Vaxman, Marcel Campen, Olga Diamanti, Daniele Panozzo, David Bommes, Klaus Hildebrandt, and Mirela Ben-Chen. 2016. Directional field synthesis, design, and processing. In *Computer Graphics Forum*, Vol. 35. Wiley Online Library, 545–572.
Ryan Viertel and Braxton Osting. 2019. An Approach to Quad Meshing Based on Harmonic Cross-Valued Maps and the Ginzburg–Landau Theory. *SIAM Journal on Scientific Computing* 41, 1 (2019), A452–A479.
Ana Maria Vintescu, Florent Dupont, and Guillaume Lavoué. 2017. Least squares affine transitions for global parameterization. (2017).
Yuanzhen Wang, Beibei Liu, and Yiying Tong. 2012. Linear surface reconstruction from discrete fundamental forms on triangle meshes. In *Computer Graphics Forum*, Vol. 31. Wiley Online Library, 2277–2287.
Jiayi Eris Zhang, Alec Jacobson, and Marc Alexa. 2021. Fast Updates for Least-Squares Rotational Alignment. In *Computer Graphics Forum*, Vol. 40. Wiley Online Library, 13–22.

A PROOF OF THEOREM 2.2

The proof will follow three steps: 1) we demonstrate that there is a mapping f satisfying Eq. (1) in M^c, 2) we show that σ is continuous away from singular points even though the frame field is discontinuous in $\mathbb{U}$ and 3) the vector tangent to the cuts are equal up to a rotation due to the discontinuity in the frame z.

First, we show the existence of a mapping f solution of Eq. (1). By construction, there exists a *smooth* frame field $z : M^c \to \mathbb{U}$ whose Darboux derivative is ω. Since Eq. (3) holds true, the form $z\sigma$ is closed. Using the Poincaré lemma, we conclude that $z\sigma$ is exact on M^c as it is simply connected. Therefore, there exists a map $f : M^c \to \mathbb{C}$ solution of:

$$\mathrm{d}f_p = z_p \sigma_p, \quad \forall p \in M^c.$$

Secondly, we are going to prove that, even though z is *discontinuous* in $\mathbb{U}$, there always exists a form σ satisfying the first structure equation Eq. (3) which is *smooth* away from singularities and independent of z and ω. To do so, we restrict ourselves to conformal deformations by choosing $\sigma = e^u \mathrm{d}x$ where $\mathrm{d}x = \mathrm{d}x_1 + \imath \mathrm{d}x_2$ is a local 1-form orthonormal basis. Injecting this into Eq. (3), we show that the log-scale factor u is solution of the system of equations:

$$\mathrm{d}u \wedge \mathrm{d}x_1 + \omega \wedge \mathrm{d}x_2 = 0,$$
$$\mathrm{d}u \wedge \mathrm{d}x_2 - \omega \wedge \mathrm{d}x_1 = 0.$$

The only solution of this system is: $\omega = - \star \mathrm{d}u$. The frame field rotation ω uniquely defines the frames up to global rotation [Sharpe 1997] and can be related to the Gaussian curvature K and the frame cones points Ω by the equation [Corman and Crane 2019; Crane et al. 2010]:

$$\mathrm{d}\omega = -K + \Omega.$$

Therefore, u is solution of the Poisson's equation:

$$\Delta u = K - \Omega. \tag{11}$$

Eq. (11) always has a smooth solution away from singular points. Thus, σ is locally smooth away from singularities and is independent of the frame field discontinuities.

Third, we are going to study the properties of the parametrization cuts. Let $p \in M$ be a point at a cut and p_+, p_- are the corresponding points on each side of the cut in M^c. By construction, there exists a

frame field $z : M^c \to \mathbb{U}$ whose rotation speed ω is smooth on M^c but discontinuous on M. At a discontinuity, there exists a transition rotation $g_p \in \Gamma$, such that: $z_{p_-} = g_p z_{p_+}$. Since σ is continuous on M^c, we have: $\sigma_{p_-} = \sigma_{p_+}$. Therefore, a tangent vector $X \in T_pM$ has two limit images by the differential map: $\mathrm{d}f_{p_+}(X_p) = z_{p_+}\sigma_p(X_p)$ and $\mathrm{d}f_{p_-}(X_p) = z_{p_-}\sigma_p(X_p)$. Using the continuity of σ, we have:

$$\mathrm{d}f_{p_+}(X_p) = g_p \mathrm{d}f_{p_-}(X_p), \quad g_p \in \Gamma.$$

Therefore, vectors tangent to the cut are mapped to vector equal up to a rotation defined by the frame field discontinuity.

B PROOF OF THEOREM 6.1

In order to show that our constraints define a valid parametrization with quantized cone points in the sense of Def. 3.1. We will follow four steps. 1) We will show the validity of the triangle reconstruction (Sec. 6) and that transition functions are rotations of angle $2k\pi/n$. 2) The parametrization is locally injective. 3) The parametrization's angle defects are equal to the singular indices of the frame field up to a *subtraction* of an integer multiple of 2π. 4) Due to the Gauss-Bonnet theorem, parasite singularities cannot appear.

Triangle reconstruction. First, we will show that the parametrization cuts appear at the frame field discontinuities and that the rotation jumps match those of the frames. Since the power field satisfies Eq. ($\mathcal{F}_n$), the bases z rotations are given by the formula:

$$z_j = \exp\left(2\imath\pi k_{ij}/n\right) cay(\omega_{ij})^{-1} r_{ij} z_i,$$

where k_{ij} is the jump index at edge ij.

Let $\sigma_i \in \mathbb{C}, \sigma_j \in \mathbb{C}$ be duplicated *primal* edge vectors of adjacent charts C_i, C_j and $a = z_i\sigma_i, b = z_j\sigma_j$ their coordinates in parameter space. Using Eqs. ($\mathcal{E}$) and Eq. (8), we see that they are opposite in parameter space:

$$\begin{aligned} b &= z_j\sigma_j, \\ &= \left(\exp\left(2\imath\pi k_{ij}/n\right) cay(\omega_{ij})^{-1} r_{ij} z_i\right)\left(-cay(\omega_{ij}) r_{ij}^{-1}\sigma_i\right), \\ &= -\exp\left(2\imath\pi k_{ij}/n\right) z_i\sigma_i, \\ &= -\exp\left(2\imath\pi k_{ij}/n\right) a. \end{aligned}$$

The same reasoning can be be applied to *dual* edges. Thus, primal and dual edges are equal up to the frame rotational discontinuity $d_{ij} := \exp\left(2\imath\pi k_{ij}/n\right)$. Therefore, our parametrization transition functions are rotation quantized by the frame field. In particular, our parametrization is seamless whenever $n = 4$.

To achieve our triangle reconstruction of the parametrization, three cases can occur. 1) A triangle is not traversed by a cut and it will be reconstructed as a triangle in the parametrization. 2) A *non-singular* triangle is traversed by a cut, thus the cut can be displaced on primal edges to recover a non-singular triangle. 3) A *singular* triangle is reconstructed with a cut appearing in one of the dual edge linking the point s. By displacing the cut, we recover a triangulation of the singular triangle with the cut linking s to one of the vertex as in Fig. 6.

Injectivity and feature constraints. Eqs. ($\mathcal{I}_1$) ensure local injectivity as all triangles (p_i, p_j, s_{ijk}) composing the parametrization must have positive area.

Parametrization angle defects. We are now going to precisely relate the parametrization angle defects with the frame singular indices.

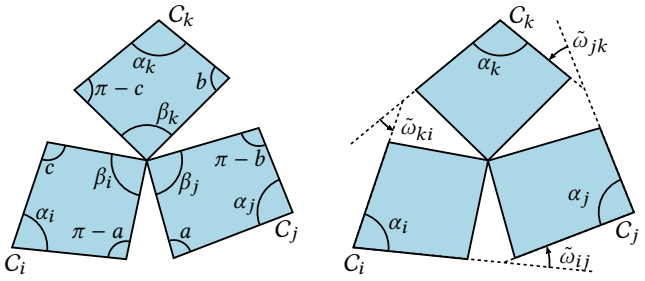


Fig. 21. The triangle ijk is decomposed into the three charts C_i, C_j and C_k satisfying Eqs. ($\mathcal{E}$). Left: the angles α denote the triangle angle and β the angle at the singularity point. Right: primal edges of the charts are equal up to the rotation $cay(\omega)$ whose angle is $\tilde{\omega} = 2\arctan(\omega/2)$ (see Eq. (6)).

We showed that the parametrization transition functions match the frame field jumps. Therefore, the parametrization angle defect at a singular point is equal to the total rotation angle of the frame up to an integer multiple of 2π. We will show that this additional parasite angle corresponding to integer singularities can only be negative.

Let us consider a triangle ijk and its chart decomposition as illustrated by Fig. 21. Let Ω_{ijk} be the frame field total rotation around the triangle, that is to say the singularity index prescribed by the frame field:

$$\Omega_{ijk} = \tilde{\omega}_{ij} + \tilde{\omega}_{jk} + \tilde{\omega}_{ki},$$

where $\tilde{\omega} = 2\arctan(\omega/2)$ are the angle of the rotation between two frames, parametrized by the Cayley transform (Eq. (6)).

Let Θ_{ijk} be the angle defect of triangle ijk in parameter space, by definition:

$$\Theta_{ijk} = 2\pi - (\beta_i + \beta_j + \beta_k)$$

Because of constraints ($\mathcal{I}_1$) and ($\mathcal{I}_2$), the three charts C_i, C_j and C_k, decomposing triangle ijk, are non self-intersecting quadrilaterals. Thus, their four inner angles, defined in Fig. 21 (left), must sum to 2π:

$$\begin{aligned} \alpha_i + \beta_i + c + \pi - a &= 2\pi, \\ \alpha_j + \beta_j + a + \pi - b &= 2\pi, \\ \alpha_k + \beta_k + b + \pi - c &= 2\pi. \end{aligned}$$

Moreover, by summing these three equations, we obtain a link between the angle defect Θ and the triangle angles α:

$$\Theta_{ijk} = \alpha_i + \alpha_j + \alpha_k - \pi.$$

We are now ready to relate Θ with the frame field singularity angle Ω. Let us consider the hexagon formed by extending primal edges of charts until intersection, as in Fig. 21 (right). This hexagon can be self-intersecting but remains star-shaped around its singularity point s. We can express the sum of its inner angles in term of its turning number $k > 0$:

$$\alpha_i + \alpha_j + \alpha_k + 3\pi - \tilde{\omega}_{ij} - \tilde{\omega}_{jk} - \tilde{\omega}_{ki} = \pi(6 - 2k).$$

Rearranging the terms, the parametrization angle defect Θ is equal to the frame field total rotation Ω up to a negative integer

multiple of 2π:

$$\Theta_{ijk} = \Omega_{ijk} - 2\pi k, \quad k \in \mathbb{N}. \tag{12}$$

Double coverings. We are now going to prove that the parametrization singularities are exactly those prescribed by the frames. Eq. (12) shows that inside triangles the parametrization angle defect can differ from the frame field singularities by a negative integer multiple of 2π. Vertices of the input mesh also have prescribed singularity indices: 0 for inner vertices and user prescribed on feature edges. At these vertices, unwanted *negative* integer index singularities could also appear even if all triangles have positive area (see Fig. 22).

However, due to the Gauss-Bonnet theorem, an additional *negative* singularity can appear only if it is compensated by an additional *positive* singularity. This is impossible because only *negative* parasite singularities can appear at both triangles and vertices.

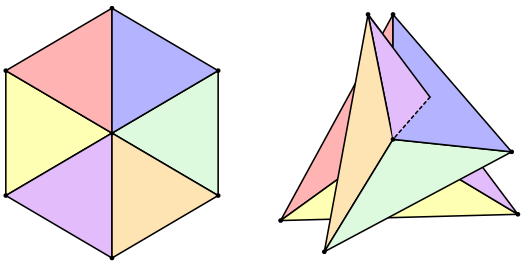


Fig. 22. Two versions of the same vertex rings: one with an angle defect of 0 and the other with an angle defect of 2π, both have only positive area triangles.