Computing Smooth and Integrable Cross Fields via Iterative Singularity Adjustment

Long Ma
SDU, China

Ying He
NTU, Singapore

Jianmin Zheng
NTU, Singapore

Yuanfeng Zhou
SDU, China

Shiqing Xin
SDU, China

Caiming Zhang
SDU, China

Wenping Wang
TAMU, USA

Abstract

We propose a new method for computing smooth and integrable cross fields on 2D and 3D surfaces. We first compute smooth cross fields by minimizing the Dirichlet energy. Unlike the existing optimization based approaches, our method determines the singularity configuration, i.e., the number of singularities, their locations and indices, via iteratively adjusting singularities. The singularities can move, merge and split, as like charges repel and unlike charges attract. Once all singularities stop moving, we obtain a cross field with (locally) lowest Dirichlet energy. In simply connected domains, such a cross field is guaranteed to be integrable. However, this property does not hold in multiply connected domains. To make a smooth cross field integrable, we construct a vector field c, which characterizes how far the cross field is away from a curl-free field. Then we optimize the locations of singularities by moving them along the field lines of c. Our method is fundamentally different from the existing integer programming-based approaches, since it does not require any special numerical solver. It is fully automatic and also has a parameter to control the number of singularities. Our method is well suited for smooth models in which exact boundary alignment and sparse hard directional constraints are desired, and can guide seamless conformal parameterization and T-junction-free quadrangulation. We will make the source code publicly available.

***Keywords:** Cross field, integrability, singularity placement, iterative singularity adjustment*

1. Introduction

A cross field, also known as 4-RoSy field [22, 25], is a directional field that exhibits rotational symmetry and is invariant under rotations of $\frac{\pi}{2}$ radians. Originally introduced by Hertzmann and Zorin as a computational tool for nonphotorealistic cross-hatching rendering [14], cross fields have gained significant prominence in the construction of high-quality quadrilateral meshes [3, 15]. These meshes are highly desired in various engineering fields, including simulation [19] and finite element analysis [33].

Constrained by the topology of the surface on which it is defined, a cross field cannot possess an arbitrary arrangement of singularities. If the singularities of the cross field are predetermined, including their number, placement, and indices, smooth cross fields can be generated directly by solving a sparse linear system [9]. The computation of smooth cross fields with unknown singularities is commonly formulated as an optimization problem, and numerous methods exist for solving it [3, 17, 2, 18, 12].

Cross fields are commonly utilized as a guidance in the computation of global conformal parameterization, from which quadrilateral meshes can be extracted. A cross field is integrable if it aligns well with the gradient of some scalar field on the surface. When the computed cross fields are integrable, the parameterization can be obtained directly by tracing the integral curves of the cross fields. However, for non-integrable cross fields, it is necessary to compute a pair of parameters (u, v) by minimizing the difference between the gradients of the parameters and the directions of the cross fields. In cases where the integrability of the cross fields is poor, the resulting parameterization may be misaligned with the desired cross field and can even exhibit inversions [11].

It is well known that in simply connected domains, a smooth cross field, which is the one the minimal Dirichlet energy, is guaranteed to be integrable [6]. Unfortunately, this desirable property does not extend to multiply connected domains, where the non-trivial topology imposes significant constrains on the solution space for achieving both smoothness and integrability in cross fields.

Existing research efforts in the field have primarily concentrated on computing smooth cross fields with optimal singularities, with relatively less attention given to the is-

sue of integrability. Only a few attempts have been made to explicitly address the integrability problem by relaxing either the orthogonality [11, 26] or the equal length constraints [24] of the cross fields, thereby expanding the solution space. To the best of our knowledge, there is currently no research that addresses both the smoothness and integrability of cross fields on surfaces with arbitrary topology.

This paper aims at investigating the relationship between the smoothness and integrability of cross fields defined on smooth surfaces of arbitrary topology. Let Ω be a smooth surface and θ a cross field with the lowest Dirichlet energy defined on Ω. By utilizing the singularities of θ and the boundary (if present) of Ω as boundary conditions, we compute a conformal factor $\varphi : \Omega \to \mathbb{R}$ by solving a Poisson's equation. Subsequently, we define a vector field $\mathbf{c}$, referred to as the discrepancy field, for all regular points, which quantifies the difference between the gradient of the conformal factor $\nabla\varphi$ and the rotation vector of cross field θ.

In simply connected domains where θ is free of directional constraints, the discrepancy is always a zero vector. In multiply connected domains, $\mathbf{c}$ is globally divergence free and locally curl free. We then prove that a smooth cross field θ is integrable if and only if the discrepancy field vanishes everywhere except at the singularities.

The established necessary and sufficient condition in the continuous setting serves a foundation for a novel algorithm for computing cross fields on triangle meshes with nontrivial topology. Our method is fully automatic and supports boundary alignment and sparse hard directional constraints. It also allows the user to balance the number of singularities and the Dirichlet energy by tuning the disk size r_0, which in turns controls the area distortion of the induced global conformal parameterization. Through extensive evaluations, we show that the algorithm enables the generation of high-quality global conformal parameterizations that are well aligned with the cross fields. See Figure 1 for an example of our method.

2. Related Works

For the sake of brevity, we restrict our survey to the most relevant works on cross fields, non-orthogonal fields and cone optimization in surface parameterization. We refer the readers to [31] for a comprehensive survey of orthogonal and non-orthogonal directional fields.

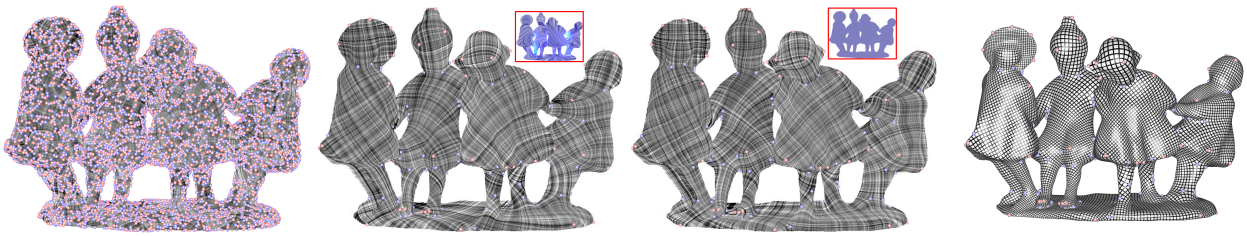
Cross fields were first introduced by Hertzmann and Zorin as a computational tool for nonphotorealistic crosshatching rendering [14]. They were then generalized to N-rotational symmetric directional field (N-RoSy) in [22][25]. After that, smooth cross and N-RoSy fields have been extensively studied [31]. If singularities are given (i.e., their number, placement and indices are fixed), smooth cross fields can be created efficiently by solving a sparse linear system [9]. The design of cross fields with unknown singularities (number, placement and indices) are often formulated as a non-linear optimization problem with or without integer variables [3, 2, 18, 12, 4, 34].

Cross fields are inherently isometric, meaning that they do not encode scale information. Frame fields, representing a more generalized form of cross fields, are characterized by their non-orthogonality and non-unit-lengths. This unique aspect of frame fields make them suitable for anisotropic quadrangulation with varying element size [23]. Viewing frame fields as cross fields in a specific Riemannian metric, Jiang et al. [16] first computed a discrete metric on the input surface that is compatible with input constraints and then optimized the cross field in this customized metric to obtain final frame field. Furthermore, 2D singularity-free frame fields have found applications in line drawing vectorization [13].

PolyVectors are sets of vectors that are unordered and represented as the roots of a complex polynomial. A smooth PolyVector field can be computed by solving a sparse linear system without integer variables [10]. By eliminating discrete curls, one can obtain integrable PolyVector fields. These integrable fields are particularly useful for computing global parameterization that adheres to specific alignment constraints [11]. Sageman-Furnas et al. [27] formulated the global parameterization problem in terms of commuting PolyVector fields, and designed an efficient optimization method to solve it. Based on the modification of holonomic signature loops, Shen et al. [28] developed a method aimed at generating parameterizations that are locally injective and globally holonomic.

Cone singularities play a critical role in controlling the area distortion of surface parameterization. Various heuristics and optimization frameworks are proposed to automatically determine the singularity configuration[1, 30, 20, 21, 5, 8, 29]. Campen et al. [7] showed that for arbitrary given sets of topologically admissible parametric cones with prescribed curvature, a global seamless parametrization always exists.

Our paper has close contact to the seminal paper of Bunin [6], who developed an elegant algorithm for computing conformal parameterization of simply connected domains. His method computes conformal factor φ by solving Poisson's equation with user-provided singularities. The gradient of φ induces an integrable cross field, which yields a conformal parameterization. Manual singularity selection heavily relies on human geometric intuition and trial and error. We extend Bunin's original idea to surfaces of arbitrary topology by tackling the challenge of integrability. Our method can automatically determine the singularity configuration and also allows the user to control the number of singularities by tuning a parameter with intuitive meaning.



(a) Initial random cross field θ_0 (b) Smooth cross field θ_1 (c) Smooth & integrable field θ_2 (d) Induced parameterization (u, v)

Figure 1. Based on the theoretical results, we develop an automatic method for computing smooth and integrable cross fields on 3D surfaces of complex topology. The algorithm takes as input an random cross field with 4,341 positive (red) and 4,397 negative (blue) singularities (a). It first smoothens the cross field by iteratively adjusting singularities. When two singularities of opposite indices meet, they annihilate, resulting in a drop of the Dirichlet energy. When all singularities stop moving, the Dirichlet energy reaches a local minimal. The corresponding cross field is smooth and has only 85 positive and 141 negative singularities (b). However, this cross field is not integrable since the surface is multiply connected. We visualize the vector field $\mathbf{c}$ (see the insets) and show that moving singularities along the field direction of $\mathbf{c}$ reduces $\mathbf{c}$, hereby improving integrability. When $\mathbf{c} \equiv \mathbf{0}$, the cross field is both smooth and integrable (c), which naturally induces a global conformal parameterization (d) whose parameter lines are well aligned with the field. Each positive (resp. negative) singularity has a cone angle $\frac{\pi}{4}$ (resp. $\frac{-\pi}{4}$). The area and angle distortions are $E_{AD} = 1.015$ and $E_{SD} = 1.076$, respectively. The closer the distortion metrics to 1, the lower the distortions. We show the discrepancy field $\mathbf{c}$ in the small insets in (b) and (c).

3. Smooth Cross Fields

A cross field on a surface is defined as a pair of perpendicular tangent directions. The degree of freedom for a cross field is 1. Consider a cross field θ defined on a smooth surface Ω. The cross field undergoes rotation as it moves along a path on the surface. To quantify this rotation, we associate a unique vector field ω with θ, where $\omega \cdot \mathrm{d}\mathbf{l}$ represents the rotation angle along the line element $\mathrm{d}\mathbf{l}$, and $\cdot$ denotes the vector dot product. The magnitude of ω represents the rotation speed. In this paper, we refer to ω as the *rotation vector field* of the cross field θ. It is important to note that the rotation vector ω is undefined at singularities due to its magnitude becoming infinity.

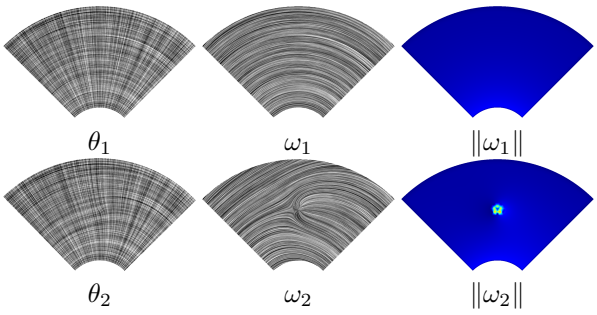


Figure 2. Cross field and its associated rotation vector field. Top: A smooth cross field θ_1 is generated using boundary constraints only. Its induced rotation field ω_1 has a vanishing divergence everywhere. Bottom: In the case of the cross field θ_2 where a directional constraint is added at an interior point, it induces a rotation vector field ω_2 in which that point becomes a source of the rotation field. We visualize the magnitude $\|\omega\|$ using the heat color map, where cold colors indicate small magnitudes and warm colors large magnitudes.

The rotation vector field ω can be seen as analogous to the gradient field of a scalar function, as the cross experiences the highest rotation speed while parallelly transported along ω. However, we cannot express ω as $\nabla\theta$, since θ is not a function, lacking a reference direction unless Ω has zero Gauss curvature everywhere. In Figure 2, we provide two examples of cross fields and their associated rotation vector fields.

In cross fields, singularities are isolated points where the direction field shows discontinuity, and they are inevitable in all but the simplest shapes. Each singularity, denoted as s_i, is associated with a non-zero index $I(s_i)$. This index, which is a multiple of $\frac{1}{4}$, characterizes the behavior of the cross field in the vicinity of s_i. Figure 3 illustrates the typical types of singularities, with their indices ranging from $-1/2$ to $1/2$. According to the Poincaré-Hopf Theorem, for a closed, orientable surface of genus g, the sum of all the indices of singularities in a cross field equals the Euler characteristic $2 - 2g$.

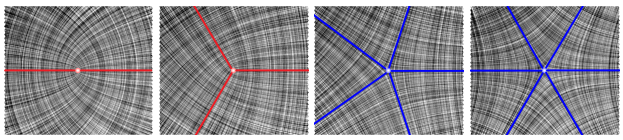


Figure 3. Illustration of singularities in cross fields. The red and blue curves are the integral curves intersecting the singularity.

By applying the Gauss-Bonnet Theorem, it is easy to see that for an arbitrary simply connected subset $D \subset \Omega$, the rotation vector field ω satisfies [6]

$$\oint_{\partial D} \omega \cdot \mathrm{d}\mathbf{l} = -\iint_D K \mathrm{d}\sigma + 2\pi \sum_{s_i \in D} I(s_i), \qquad (1)$$

where K is the Gaussian curvature, and $\mathrm{d}\mathbf{l}$ and $\mathrm{d}\sigma$ are the

line and area elements, respectively.

Let $\mathcal{S} = \{s_i | s_i \in \Omega\}$ denote the set of singularities of the cross field θ. Equation (1), which represents a constraint on ω in integral form, can also be expressed in a differential form as follows:

$$(\nabla \times \omega)(x) \cdot \mathbf{n} = -K(x) + 2\pi \sum_{s_i \in \mathcal{S}} \delta(x - s_i), \quad (2)$$

where x is an interior point of Ω, $\mathbf{n}$ is the surface normal and $\delta(x - s_i)$ denotes the Dirac delta function.

Since the rotation vector ω describes the rotational behavior of the cross field θ between points, excluding singularities, it is natural to use $\|\omega\|^2$ as a measure of the smoothness of the cross field θ. The smoothness energy of cross field θ is then defined as:

$$E_{\text{smooth}} = \iint_{\Omega} \|\omega\|^2 \, \mathrm{d}\sigma. \quad (3)$$

By applying the variational method to (3), it becomes apparent that if the cross field θ has a locally minimal smoothness energy, the corresponding rotation vector field ω must be divergence-free, satisfying the condition

$$\nabla \cdot \omega = 0. \quad (4)$$

We define the smoothness of a cross field θ by the condition that its associated rotation vector field ω satisfies (4) . All subsequent discussions regarding cross field in this paper assume their smoothness, as this is a prerequisite for the application of Bunin's theory.

4. Simply-Connected Domains

As our work builds upon the theoretical framework established by Bunin, we provide a brief overview of his key findings in this section. Throughout this section, we refer to Ω as a simply connected domain.

4.1. Boundary-aligned Conformal Parameterization

Let $\mathbf{r}(u, v) : \mathbb{R}^2 \to \Omega$ be a conformal parametrization of Ω, whose parameter lines are aligned with the domain boundary. The tangent vectors $\mathbf{r}_u$ and $\mathbf{r}_v$ are perpendicular to each other

$$\mathbf{r}_u \perp \mathbf{r}_v$$

and have equal length

$$\|\mathbf{r}_u\| = \|\mathbf{r}_v\|.$$

Thus, the normalized tangent vectors $(\mathbf{r}_u/\|\mathbf{r}_u\|, \mathbf{r}_v/\|\mathbf{r}_v\|)$ form a cross field[1] on Ω.

[1]For notation purpose, we use a 2-tuple of orthogonal vectors to denote a cross field, which allows us to match them to the gradients of the u and v parameters.

Since the parameterization $\mathbf{r}$ is aligned with the surface boundary $\partial\Omega$, the cross field should rotate by an angle equal to the geodesic curvature of the boundary when traversing along $\partial\Omega$. Consequently, the associated rotation vector ω satisfies the equation

$$\omega \cdot \mathbf{t} = \kappa_g$$

for all boundary points, where $\mathbf{t}$ represents the unit tangent vector along $\partial\Omega$ and κ_g denotes the geodesic curvature.

Conformal factor $e^{-\varphi}$, which is defined as

$$e^{-\varphi} = \|\mathbf{r}_u\|,$$

characterizes the local cell size. Intuitively speaking, the gradient $\nabla\varphi$ (shown in orange in Figure 4) represents the direction in which the change rate of cell size is highest.

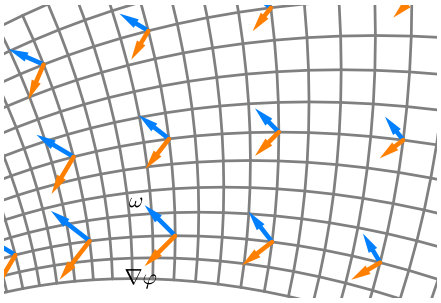


Figure 4. Gradients of a boundary-aligned conformal parameterization.

Bunin proved that for simply connected domains, the gradient of conformal factor φ at interior points is perpendicular to the rotation vector, i.e.,

$$\nabla\varphi = \mathbf{n} \times \omega, \quad (5)$$

where $\mathbf{n}$ the outward normal of Ω. For any boundary point, the following condition

$$\nabla\varphi \cdot \mathbf{n}_b = \kappa_g$$

holds, where $\mathbf{n}_b$ is the normal of the boundary $\partial\Omega$.

Substituting (5) into (1) yields

$$\oint_{\partial\Omega} \nabla\varphi \cdot \mathbf{n}_b \mathrm{d}l = -\iint_{\Omega} K \mathrm{d}\sigma + 2\pi \sum_{s_i \in \Omega} I(s_i), \quad (6)$$

Since Ω is simply connected, applying the Gauss-Bonnet Theorem yields

$$\sum_{s_i \in \Omega} I(s_i) = \frac{1}{2\pi} \left(\iint_{\Omega} K \mathrm{d}\sigma + \oint_{\partial\Omega} \kappa_g \mathrm{d}l \right) = 1. \quad (7)$$

Equation (7) is a necessary condition for the singularities of a conformal parameterization on a simply connected domain. Bunin showed that this condition is also sufficient [6].

Therefore, given an arbitrary set of singularities that satisfy (7), there exists a unique conformal parametrization with parameter lines aligned with the domain boundary.

Since $\|\nabla\varphi\| = \|\omega\|$, the smoothness energy of the cross field θ is equal to the Dirichlet energy of the conformal factor

$$E_{\text{smooth}}(\omega) = E_{\text{Dirichlet}}(\varphi),$$

revealing the deep relationship between smooth cross fields and boundary-aligned conformal parameterization.

Note that Equation (5) immediately implies

$$\begin{aligned} \nabla \cdot \omega &= \nabla \cdot (\nabla\varphi \times \mathbf{n}) \\ &= (\nabla \times \nabla\varphi) \cdot \mathbf{n} - \nabla\varphi \cdot (\nabla \times \mathbf{n}) \\ &= 0, \end{aligned}$$

which confirms the rotation vector field ω is divergence free as in Equation (4). Bunin also proved that for simply connected domains, the condition $\nabla \cdot \omega = 0$ is sufficient for θ being smooth. Therefore, computing a boundary-aligned conformal parameterization in simply connected domains is equivalent to constructing a smooth cross field. In other words, a smooth cross field in simply connected domains is guaranteed to be integrable.

4.2. Bunin's Algorithm

Given a set of singularities satisfying Equation (7) in a simply-connected domain, Bunin's algorithm first solves the differential form of (6), which is Poisson's equation

$$\nabla \cdot \nabla\varphi(x) = 2\pi \sum_{s_i \in \mathcal{S}} \delta(x - s_i) - K(x), \tag{8}$$

for interior point $x \notin \partial\Omega$, with the Neumann condition to boundary points $x \in \partial\Omega$

$$\nabla\varphi(x) \cdot \mathbf{n}_b = \kappa_g(x). \tag{9}$$

After obtaining the conformal factor $e^{-\varphi}$, Bunin constructed a cross field θ using the gradient $\nabla\varphi$. Since the cross field θ is integrable, the conformal parameterization can be obtained by integrating the cross field θ.

5. Multiply-Connected Domains

In simply-connected domains, Bunin showed that φ satisfying (8) and (9) can guarantee the existence of cross field with rotation vector $\omega \perp \nabla\varphi$. Unfortunately, such a property does not hold in multiply connected domains. The key difference between simply- and multiply-connected domain is that not every simple loop in a multiply-connected domain borders a region.

Let Ω be a multiply connected domain with or without boundary. Let $\varphi : \Omega \to \mathbb{R}$ be a scalar function defined on Ω satisfying (8). Let θ be an integrable cross field sharing the same singularities of φ. Consider a simple loop $C \subset \Omega$ that does not pass through any singularity. Parallel transport a cross along C. When returning to the starting point, the cross overlaps with its initial position. This means that the angle change in the parallel transport of the cross along C is a multiple of $\frac{\pi}{2}$ radians,

$$\oint_C (\omega \cdot \mathbf{t} - \kappa_g)\, \mathrm{d}l = \frac{k\pi}{2},$$

for some integer $k \in \mathbb{Z}$, where $\mathbf{t}$ is the unit tangent vector and κ_g is the geodesic curvature of C. Assume the rotation vector ω of cross field θ is perpendicular to the gradient $\nabla\varphi$. Then φ satisfies

$$\oint_C (-\nabla\varphi \cdot \mathbf{n}_b - \kappa_g)\, \mathrm{d}l = \frac{k\pi}{2}, \quad k \in \mathbb{Z} \tag{10}$$

where $\mathbf{n}_b$ is the normal vector of C. Equation (10) is the *additional* condition that φ must obey to ensure the existence of integrable cross field θ with $\omega \perp \nabla\varphi$.

Applying the Green's theorem and the Gauss-Bonnet theorem, it is easy to show that Equation (10) holds automatically if C is the boundary of a simply connected domain and φ satisfies (8). Thus, this condition is redundant for simply connected domains.

However, Equation (10) does not hold if C does not border a region. Figure 5 shows an example that φ, defined on a two-holed annulus, satisfies (8) but its gradient $\nabla\varphi$ does not satisfy (10). Therefore, Bunin's algorithm works only for simply connected domains.

Let S be a smooth surface. To obtain a conformal parameterization well aligned with a given cross field $V = (\alpha, \beta)$, V must be integrable. Specifically, a cross field on a smooth surface S is integrable if there exists scalar functions $u : S \to \mathbb{R}$, $v : S \to \mathbb{R}$, $\lambda : S \to \mathbb{R}_+$ and $\mu : S \to \mathbb{R}_+$ such that $\nabla u = \lambda\alpha$ and $\nabla v = \mu\beta$, where ∇ is the gradient operator. Since the curl of a gradient is zero, the integrable condition is equivalent to

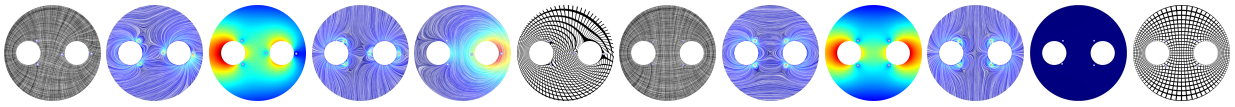
$$\nabla \times \alpha = \nabla \times \beta = \mathbf{0}.$$

In other words, a cross field is integrable it satisfies the curl-free condition.

If the cross field V has singularities and/or the surface has a non-disk topology, one needs to cut S open into a single, disk-topology patch with all singularities on its boundaries. Although the function values u (resp. v) are discontinuous on the two sides of the seam, as long as their gradients ∇u (resp. ∇v) match, the parameterization is seamless.

We define the *discrepancy* field to measure the difference between the gradient of conformal factor and the cross field's rotation vector,

$$\mathbf{c} \triangleq \nabla\varphi - \mathbf{n} \times \omega. \tag{11}$$



(a) θ_1 (b) ω_1 (c) φ_1 (d) $\nabla\varphi_1$ (e) $\mathbf{c}_1$ (f) (u_1, v_1) (g) θ_2 (h) ω_2 (i) φ_2 (j) $\nabla\varphi_2$ (k) $\mathbf{c}_2$ (l) (u_2, v_2)

Figure 5. Smoothness does not automatically imply integrability in multiply connected domains. (a) Consider a cross field $\theta_1 = (\alpha, \beta)$ with locally minimal smoothness energy defined on a two-holed disc. There are four singularities of indices $\frac{-1}{4}$. (b) The associated rotation vector field ω_1 is divergence-free. (c) Using the singularities of θ_1, we compute the conformal factor $e^{-\varphi_1}$ by solving Poisson's equation (8). (d) The gradient $\nabla\varphi_1$ is not perpendicular to the rotation vector ω_1. (e) Consequently, the discrepancy field $\mathbf{c}_1 = \nabla\varphi_1 - \mathbf{n} \times \omega_1$ does not vanish, indicating that the cross field θ_1 is not integrable. (f) As a result, the induced parameterization (u_1, v_1), which is the minimizer of $\|\nabla u_1 - \alpha\|^2 + \|\nabla v_1 - \beta\|^2$, is neither conformal nor aligned with θ_1. (g-k) Optimizing the locations of singularities can significantly reduce the discrepancy field, leading to an integrable cross field θ_2. (l) shows the induced conformal parameterization (u_2, v_2) that is well aligned with the cross field θ_2.

We prove a smooth cross field is integrable if and only if the discrepancy vanishes everywhere except at the singularities.

Theorem 1 *Let S be a smooth surface of arbitrary topology and $\theta = (\alpha, \beta)$ a smooth and boundary-aligned cross field defined on S. The cross field θ is integrable if and only if its discrepancy vector $\mathbf{c} = \mathbf{0}$ vanishes for all regular points.*

6. Continuous Setting

We present a new approach for automatically computing a smooth and integrable cross field that 1) aligns with the surface's boundaries and the user-specified directional constraints (if any); 2) has an optimal set of singularities so that the induced conformal parameterization has low area distortion; and 3) works for surfaces of arbitrary topology.

We first construct a smooth cross field θ by minimizing the Dirichlet energy (3). Though θ has locally minimal energy, it is not integrable in multiply connected domains. To eliminate curls, we fix the singularities' indices and number, and then optimize their locations.

In computing smooth cross field and eliminating curls, we treat singularities as moving geodesic disks and determine their number, indices and locations by iteratively adjusting singularities. This feature distinguishes our method from the existing optimization based approaches.

6.1. Computing Smooth Cross Fields

6.1.1 Singularity Movement

Starting from a randomly initialized cross field $\theta^{(0)}$, we first compute conformal factor φ by solving Poisson's equation (8) with the singularities of θ. The gradient $\nabla\varphi$ and its $i\frac{\pi}{2}$ rotations, $i = 1, 2, 3$, form a new cross field $\theta^{(1)}$, which is obviously smoother than $\theta^{(0)}$.

Let $\omega^{(1)}$ be the rotation vector field of $\theta^{(1)}$. Since $\|\omega^{(1)}\| = \|\nabla\varphi\|$, the smoothness of cross field $\theta^{(1)}$ is also the Dirichlet energy of φ. As pointed out by Knöppel et al. [18], $\|\nabla\varphi\| \to \infty$ is infinite at singularities. So we must remove singularities to obtain a finite energy. We model each singularity s_i as a geodesic disk $D_{r_0}(s_i)$ and define the smoothness energy for $\theta^{(1)}$ as follows

$$E_{\text{smooth}}(\theta^{(1)}, r_0) = \iint_{\Omega \setminus \bigcup_{s_i \in \mathcal{S}} D_{r_0}(s_i)} \|\nabla\varphi\|^2 \mathrm{d}A. \quad (12)$$

The disk radius r_0, which is a user-specified parameter, plays an important role in controlling the number of singularities and the area distortion of conformal parameterization.

Notice that $\nabla\varphi$ is implicitly controlled by 1) the Gaussian curvature K of the domain Ω and 2) the singularities. Since we are not allowed to change K, a possible way for reducing the Dirichlet energy of $\theta^{(1)}$ is to modify the singularities.

Consider an arbitrary interior, non-singular point $x \in \Omega \setminus \partial\Omega$ with $\nabla \cdot \nabla\varphi(x) = -K(x) \neq 0$. Note that the magnitude $\|\nabla\varphi(x)\|$ increases in the direction $\nabla\varphi$ and decreases in the opposite direction $-\nabla\varphi$. Moving the geodesic disk $D_{r_0}(s_i)$ singularity along $(\nabla \cdot \nabla\varphi)\nabla\varphi$ reduces the area integral $\iint_{\Omega \setminus D_{r_0}(s_i)} \|\nabla\varphi\|^2$, hereby decreases the Dirichlet energy. This intuition motivates us to move each singularity along the field direction of the gradient field $\nabla\varphi$.

If singularities were isolated points, moving a singularity would be a piece of cake since the moving direction is readily available. However, as mentioned above, we model singularities as geodesic disks to obtain finite Dirichlet energies. Therefore, we need to consider the individual "force" for every point inside the geodesic disk and then compute a combined "force" to move the disk. Specifically, we need to compute the area integral

$$\mathbf{F}(s_i) = \iint_{D_{r_0}(s_i)} (\nabla \cdot \nabla\varphi)\nabla\varphi \mathrm{d}A.$$

In Appendix A, we show that the above area integral is equivalent to a loop integral on the disk boundary

$$\mathbf{F}(s_i) = \oint_{\partial D_{r_0}(s_i)} \mathbf{T} \cdot \mathbf{n} \mathrm{d}l, \quad (13)$$

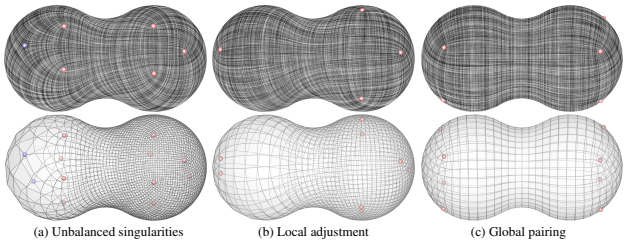
where $\mathbf{T}$ is Maxwell tensor

$$\mathbf{T} = \nabla\varphi \otimes \nabla\varphi - \frac{1}{2}\|\nabla\varphi\|^2\mathbf{I}. \tag{14}$$

We can move all singularities with the forces $\mathbf{F}(s_i)$, $i = 1, 2, \cdots, |\mathcal{S}|$. When two singularities of opposite indices meet, they cancel out each other, resulting a sharp decrease of the smoothness energy. Furthermore, a singularity with index $\pm\frac{1}{2}$ can also split into two singularities with index $\pm\frac{1}{4}$. When all singularities stop moving, the smoothness energy reaches a local minimum and the induced cross field θ is smooth.

6.1.2 Global Singularity Pairing

We observe that for surfaces with simple geometry (i.e., the Gaussian curvatures are of the same sign), electric forces can drive singularities to globally optimal locations, as shown in Figure 11. However, for general surfaces with mixed positive and negative Gaussian curvatures, singularities often get stuck at locally optimal positions, since electrostatic forces only drive them locally and cannot merge two singularities that are far away to each other (see Figure 6).



(a) Unbalanced singularities (b) Local adjustment (c) Global pairing

Figure 6. Local adjustment vs global pairing. (a) Given a symmetric peanut-shaped surface, we design a smooth cross field with unbalanced singularities: 4 positive and 2 negative singularities on the left side, and 6 positive singularities on the right. (b) Using only local adjustment, two negative singularities are merged with two nearby positive ones, resulting in 4 singularities on the left. However, since the extra singularities on the right cannot go to the left side, the resulting cross field still has unbalanced singularities. (c) Using the global pairing strategy, we can drive singularities in a much large scale, producing a balanced result with 4 singularities on each side.

To overcome this difficulty, we propose a simple yet effective heuristic that merges singularities at a *large* scale. Specifically, we identify a pair of vertices, named $v_{\max}$ and $v_{\min}$, which have the largest and smallest conformal factors, respectively. We call them the *peak* and the *valley*. Then we compute a shortest path $\gamma(v_{\max}, v_{\min})$ between them. Next we neutralize the value difference by moving $v_{\max}$ and $v_{\min}$ towards each other along γ.

Since the peak $v_{\max}$ and the valley $v_{\min}$ are not necessarily singularities, we consider four cases:

- Case 1: both $v_{\max}$ and $v_{\min}$ are regular vertices. A pair of singularities with opposite signs are created, i.e., a negative singularity at $v_{\max}$ and a positive singularity at $v_{\min}$. Therefore, the total number of singularities is increased by 2.
- Case 2: $v_{\max}$ is a positive singularity and $v_{\min}$ is regular. The singularity is moved from $v_{\max}$ to $v_{\min}$. After that, $v_{\max}$ becomes regular and $v_{\min}$ is a positive singularity.
- Case 3: $v_{\max}$ is regular and $v_{\min}$ is a negative singularity. The singularity is moved from $v_{\min}$ to $v_{\max}$.
- Case 4: both $v_{\max}$ and $v_{\min}$ are singular. The two singularities annihilate, reducing the number of singularities by 2.

The radius of geodesic disk r_0 has a significant impact on singularities. Recall that given two disks centered at the same point but with different radii, the small disk has a higher Dirichlet energy, since it has larger integration domain. The smaller the radius r_0, the higher the Dirichlet energy, the steeper the peak and the valley of the conformal factor φ, therefore, the higher the chances of both $v_{\max}$ and $v_{\min}$ being singular (case 4), and the fewer the number of singularities. Conversely, the larger the radius r_0, the lower the Dirichlet energy, the shallower the peaks and the valleys of φ, therefore, the lower the chances of $v_{\max}$ and $v_{\min}$ being singular (case 1) and the more the number of singularities. In our algorithm, the user can tune r_0 to control the number of singularities.

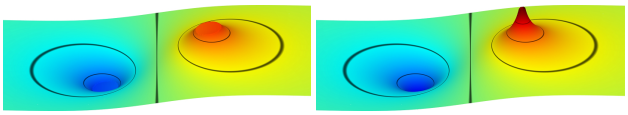
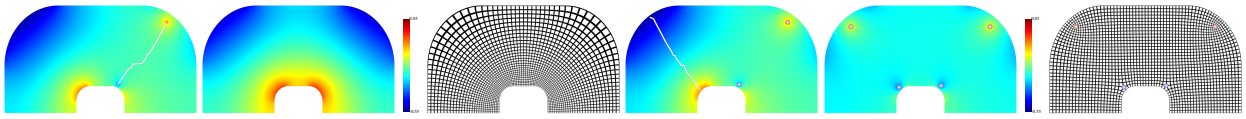


Figure 7. Consider two sets of singularities with the same indices but different radii. The singularities on the left have larger radii than the ones on the right. The small singularity (right) produces a stronger vector field $\nabla\varphi$ with steeper peaks and valleys of the value φ. We visualize φ as a color-coded height function.

Remark 1. Our global pairing strategy shares some spirit with Ben-Chen et al.'s method [1], which iteratively adds singularities to the locations with the largest conformal factor difference. The algorithm terminates when the conformal factor difference is below a user-specified threshold. Our global pairing strategy also deals with the vertex pair with the largest conformal factor difference. However, we allow them to move towards each other along a simple path. After they meet, the conformal factor field will change significantly. We keep the new field if it has lower energy than the previous one. Then we perform the local singularity adjustment to further reduce the Dirichlet energy. Our method terminates when global pairing cannot yield a cross field with lower energy.



(a) (b) (c) (d) (e) (f)

Figure 8. The disk radius r_0 plays a critical role in computing smooth cross fields. This toy model has 1 positive singularity and 1 negative singularity, drawn as orange and blue spheres respectively. Given the same conformal factor φ, we set a small radius in (a) and a large radius in (d). Let A be the surface area. (a) With a small radius $r_0 = \sqrt{A}10^{-3.5}$, the smoothness energy is $E^{(a)} = 3.816$. Since both $v_{\max}$ and $v_{\min}$ are singular, applying the global pairing strategy annihilates the singularities (case 4) and produces a singularity-free cross field (b) with energy $E^{(b)} = 3.009 < E^{(a)}$. The resulting parameterization is shown in (c). (d) With a large radius $r_0 = \sqrt{A}10^{-2.0}$, the smoothness energy is $E^{(d)} = 2.460$. Since both $v_{\max}$ and $v_{\min}$ are regular, pairing them produces a pair of singularities with opposite sign (case 1). Now the cross field has 4 singularities (e) and its Dirichlet energy is $E^{(e)} = 2.019 < E^{(b)}$. Therefore, the induced parameterization (f) has lower area distortion than (c).

6.2. Eliminating Curls

Minimizing the Dirichlet energy produces a smooth cross field, which in general contains curls if the domain is multiply connected. As discussed above, the key challenge is that in a multiply connected domain, the rotation vector ω of a smooth cross field θ may not be perpendicular to the gradient of φ that satisfies (8) and shares the same set of singularities as θ (see Figure 5(a-d)). To characterize the difference between ω and $\nabla\varphi$, we define a vector field $\mathbf{c}$

$$\mathbf{c} = \nabla\varphi - \mathbf{n} \times \omega, \tag{15}$$

where $\mathbf{n}$ is the unit outward normal of Ω. Clearly, if $\mathbf{c} \equiv 0$, the cross field θ is integrable. In the following, we first examine the properties of $\mathbf{c}$, and then present our method to eliminate curls in θ.

6.2.1 Properties

Since ω and φ share the same set of singularities, subtracting (2) from (8) yields $(\nabla \times \omega) \cdot \mathbf{n} = \nabla \cdot \nabla\varphi$. Computing the divergence of $\mathbf{c}$, we obtain

$$\begin{aligned} \nabla \cdot \mathbf{c} &= \nabla \cdot (\nabla\varphi - \mathbf{n} \times \omega) \\ &= \nabla \cdot \nabla\varphi - (\nabla \times \mathbf{n}) \cdot \omega - \mathbf{n} \cdot (\nabla \times \omega) \\ &= 0, \end{aligned} \tag{16}$$

which means $\mathbf{c}$ is divergence free.

Also note that θ is a local minimizer of E_{smooth}. Thus, we have $\nabla \cdot \omega = 0$. Computing curl of $\mathbf{c}$ yields

$$\begin{aligned} \nabla \times \mathbf{c} &= \nabla \times (\nabla\varphi - \mathbf{n} \times \omega), \\ &= \nabla \times \nabla\varphi - \nabla \times (\mathbf{n} \times \omega) \\ &= \mathbf{0}. \end{aligned} \tag{17}$$

Therefore, vector field $\mathbf{c}$ is also *locally* curl-free. Since there are no sources and sinks in $\mathbf{c}$, each integral curve of $\mathbf{c}$ is a cycle.

We show that moving the singularities along the field lines decreases the energy $\iint \|\mathbf{c}\|^2$. Therefore, a naïve approach could proceed as follows: compute the forces for all singularities and move them along the field line with a small time step. Repeat the procedure until all singularities stop moving. This approach is conceptually simple, however it is time consuming, since the forces are local and can only update the singularities locally. To improve the performance, we develop a two-step approach: in the first step, we rotate the cross field θ while fixing the locations of singularities, resulting in a sharp change of $\mathbf{c}$; in the second step, we move the singularities along the field lines, changing $\|\mathbf{c}\|$ in a continuous manner.

6.2.2 Discrete Adjustment

Let C a non-contractible loop that passes no singularity and has consistent sign of $\int \mathbf{c} \cdot \mathrm{d}\mathbf{l}$. Denote by C^+ the side satisfying $\mathbf{c} \cdot \mathrm{d}\mathbf{l} > 0$ and C^- the opposite side. We solve Laplace's equation

$$\triangle\psi = 0 \tag{18}$$

with Dirichlet boundary condition

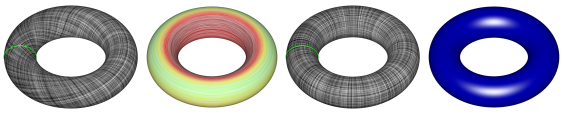
$$\psi|_{C^+} - \psi|_{C^-} = \frac{\pi}{2}. \tag{19}$$

Then we rotate θ by an angle ψ. The rotated cross field is still smooth since ψ is the solution of Laplace's equation. It is also continuous across C due to the boundary condition (19) and rotational symmetry of the cross field.

To see how the rotation changes $\mathbf{c}$, let us compute the loop integral J that measures the strength of $\mathbf{c}$

$$J = \oint_{C^+} \mathbf{c} \cdot \mathrm{d}\mathbf{l} = \oint_{C^+} \nabla\varphi \cdot \mathrm{d}\mathbf{l} - \oint_{C^+} \omega \cdot \mathbf{n}_b \mathrm{d}l = -\oint_{C^+} \omega \cdot \mathbf{n}_b \mathrm{d}l,$$

where $\mathbf{n}_b$ is the outward normal of C. The last equality comes from the fact that φ is a single-valued function and its gradient is curl free. J is positive due to the chosen orientation of C^+. Therefore, J can be viewed as the inward flux of vector field ω. Rotating θ by an angle ψ means ω is also rotated by the same angle. As a result, J changes sharply due to the rotation.



θ_1 $\mathbf{c}_1$ θ_2 $\mathbf{c}_2$

Figure 9. Discrete adjustment. The cross field θ_0 is smooth but non-integrable due to $\mathbf{c}_1 \neq \mathbf{0}$. We compute a rotation angle ψ by solving Laplace's equation (18) with Dirichlet boundary condition (19) on C (green curve). The rotated cross field θ_2 is still smooth and continuous along C, but its vector field $\mathbf{c}_2$ vanishes, hereby θ_2 is integrable.

6.2.3 Continuous Adjustment

In continuous adjustment, we move singularities along field lines. To see why this kind of singularity movement reduces $\mathbf{c}$, let us consider a non-contractible loop L passing through no singularity and compute the flux

$$I = \oint_L \mathbf{c} \cdot \mathbf{n}_b \mathrm{d}l = \oint_L \nabla\varphi \cdot \mathbf{n}_b \mathrm{d}l - \oint_L \omega \cdot \mathbf{t} \mathrm{d}l,$$

where $\mathbf{n}_b$ is the normal of L and $\mathbf{t}$ is the tangent of L. The flux measures the net number of field lines through L, indicating the strength of the field $\mathbf{c}$. Note that $\oint_L \omega \cdot \mathbf{t} \mathrm{d}l$ is a multiple of $\frac{\pi}{2}$ radians, which is a constant. Therefore, flux I totally depends on the first loop integral.

Now we examine the movement of a singularity s near L. Denote by L_{in} the side of L that $\mathbf{c}$ enters, and L_{out} the side that $\mathbf{c}$ leaves. There are 4 cases to consider:

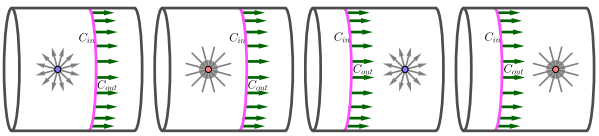
- $I(s) < 0$ and s lies on the same side of L_{in}. Moving s along $-\mathbf{c}$ makes it away from L. As a result, $\nabla\varphi$'s contribution to the integral becomes smaller, so I is reduced. See Figure 10(a).
- $I(s) > 0$ and s lies on the same side of L_{in}. Moving s along $\mathbf{c}$ makes it closer to L. For points on L, $\nabla\varphi$ and $\mathbf{c}$ are of opposite direction, so the contribution of $\nabla\varphi$ to the integral decreases, so does I. See Figure 10(b).
- $I(s) < 0$ and s lies on the same side of L_{out}. Moving s along $-\mathbf{c}$ makes it closer to L. Since $\nabla\varphi$ and c are of opposite directions for points on L, I decreases. See Figure 10(c).
- $I(s) > 0$ and s lies on the same side of L_{out}. Moving s along $\mathbf{c}$ makes it away from L, so the contribution of $\nabla\varphi$ to I decreases. See Figure 10(d).

Putting it all together, we can reduce the flux I by moving a positive singularity in the direction $\mathbf{c}$ and a negative singularity in the direction $-\mathbf{c}$.

The divergence-free feature of $\mathbf{c}$ implies that two cycles of the same homology class have equal flux. Thus, if we reduce the flux for a cycle L, then the fluxes for all cycles that are homologous to L are also reduced.

Algorithm 1: Computing Smooth Cross Fields

Data: A triangular mesh $M = (V, E, F)$, the disk radius r_0, and (optional) sparse hard directional constraints defined on triangular faces
Result: A smooth cross field θ
/* Initialization */
for each face $f \in F$ **do**
  if f is on the boundary **then**
    generate a fixed cross using the boundary direction on f;
  else if f has a directional constraint **then**
    generate a fixed cross using the constraint on f;
  else
    generate a random cross on f;
end
Factor the Laplacian matrix using Cholesky decomposition;
/* Smoothing θ via iterative singularity adjustment */
$k = 0$; $E_{\min} = \infty$; $\theta^* = \theta$;
while $k < k_{\max}$ **do**
  Compute conformal factor φ using the singularities of θ;
  Find $v_{\max}$ and $v_{\min}$ with the largest and smallest φ;
  Find a shortest path γ from $v_{\max}$ to $v_{\min}$;
  for every edge $e_{ij} \in \gamma$ **do**
    /* Assume v_i (resp. v_j) is close to $v_{\max}$ (resp. $v_{\min}$) */
    Move singularity from v_i to v_j;
  end
  Re-compute φ using the singularities of θ;
  for each edge $e_{ij} \in E$ **do**
    /* F_{ij} records the maximal force on edge e_{ij} so far */ $F_{ij} = 0$;
  end
  do
    done = true
    for each singularity s_i **do**
      Compute Maxwell tensor $\mathbf{T} = \nabla\varphi \otimes \nabla\varphi - \frac{1}{2}\|\nabla\varphi\|^2\mathbf{I}$;
      Compute force $\mathbf{F}(s_i) = \oint_{\partial D_{r_0}(s_i)} \mathbf{T}\mathbf{n}\mathrm{d}l$;
      Find edge $e_{ij} = (s_i, v_j)$ with the largest $|\mathbf{F}(s_i) \cdot \mathbf{e}_{ij}|$;
      if $|\mathbf{F}(s_i)| > F_{ij}$ **then**
        Move singularity from s_i to v_j;
        $F_{ij} = |\mathbf{F}(s_i)|$
      end
      Compute φ using the singularities of θ;
  while !done;
  Compute $E_{\mathrm{smooth}}(\theta)$;
  if $E_{\mathrm{smooth}(\theta)} < E_{\min}$ **then**
    $k = 0$; $E_{\min} = E_{\mathrm{smooth}}(\theta)$; $\theta^* = \theta$;
  else
    $k++$;
end
Output θ^*;



(a) (b) (c) (d)

Figure 10. Continuous adjustment. Moving singularity along the field lines reduces $\mathbf{c}$. A positive (orange) singularity corresponds to locally maximal φ, i.e., $\nabla \cdot \nabla\varphi < 0$, so that the gradients $\nabla\varphi$ point towards the singularity. Conversely, a negative (blue) singularity corresponds to locally minimal φ, $\nabla \cdot \nabla\varphi > 0$, so the gradients $\nabla\varphi$ emanate from the point.

7. Discrete Algorithm

Our algorithm is a fairly straightforward implementation of the physical model presented in Sections 6.1 and 6.2. We show the pseudo code in Algorithm 1 and the implementation details in Appendix.

Step 1: Initializing θ. For each boundary face and face with directional constraint, we create a fixed cross using the given direction. Then we generate a random cross for all other faces.

Step 2: Computing Smooth Cross Field. Taking the singularities of the initial random cross field as input, we solve Poisson's equation (8) to compute the initial conformal factor φ. Find the vertices $v_{\max}$ and $v_{\min}$ with the highest and lowest φ. Merge them via a simple path and update the field $\nabla\varphi$. Then move the singularities along the field lines. When the singularities stop moving, we obtain an φ with a locally minimal Dirichlet energy. If the current energy is lower than the previous energy, we call the merging $v_{\max}$ and $v_{\min}$ successful. The algorithm then merges another pair of vertices with the largest value difference and repeats the above procedure. Step 2 stops if there are $k_{\max}$ failed merges in a row. We set $k_{\max} = 3$ in our implementation.

Step 3: Eliminating Curls. The electrostatic field computed in Step 2 induces a smooth cross field θ, which may not be integrable if the input surface is multiply connected. The vector field $\mathbf{c} = \nabla\varphi - \mathbf{n} \times \omega$ measures the curls in cross field θ. In Step 3, we aim at making $\mathbf{c} \equiv \mathbf{0}$ (if possible) by adjusting θ and the locations of singularities separately.

We perform the discrete adjustment (Step 3.1) in a trial-and-error manner. We find a point p with the highest $\|\mathbf{c}\|$ and a non-contractible loop C that passes through p and has consistent sign of $\int \mathbf{c} \cdot \mathrm{d}\mathbf{l}$. Then we solve Laplace's equation $\triangle\psi = 0$ (18) using C as the boundary condition. We rotate θ and ω by angle ψ. The rotated cross field θ yields a new $\mathbf{c}$. We compute the energy

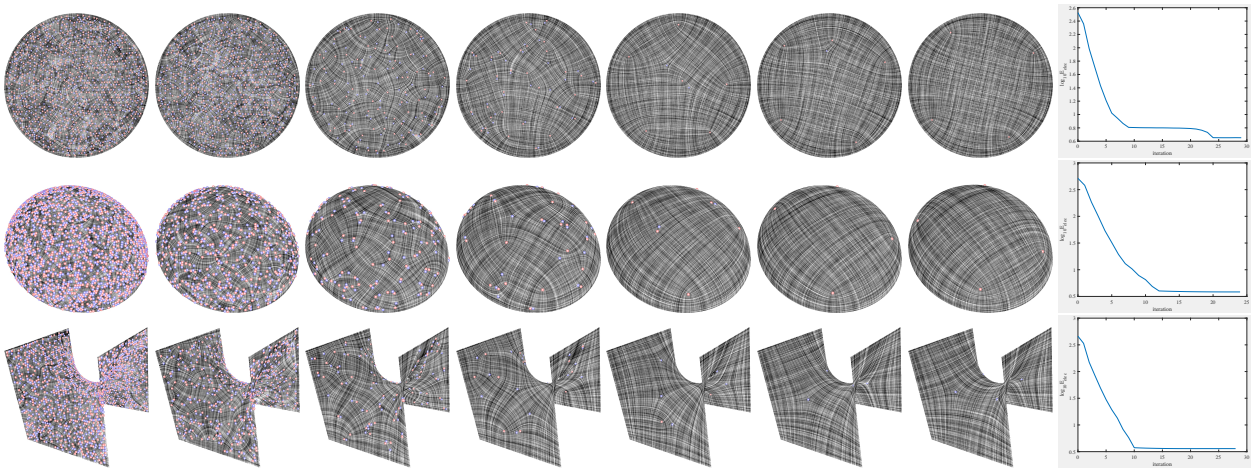
$$E_{\mathbf{c}} = \iint_M \|\mathbf{c}\|^2 \mathrm{d}\sigma,$$

which is the potential energy of $\mathbf{c}$. If the current energy is less than the previous one, the adjustment is successful. We keep doing the coarse adjustment until the energy cannot be decreased any longer.

The continuous adjustment (Step 3.2) optimizes the singularities' locations to further reduce $\|\mathbf{c}\|$ and is also carried out in an iterative manner. We define the electrostatic

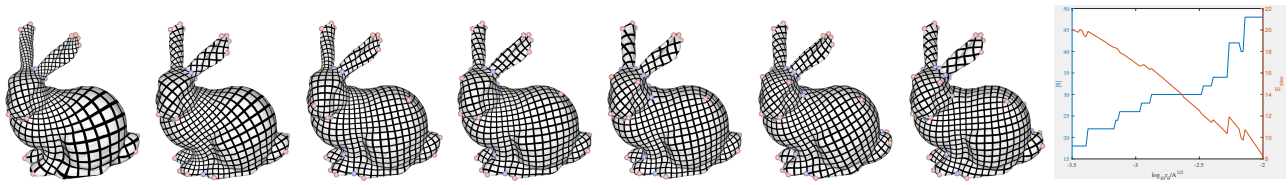
Algorithm 2: Curl Elimination

Data: A triangular mesh $M = (V, E, F)$ and a smooth cross field θ defined on triangular faces
Result: A smooth and integrable cross field θ
Computer the rotation vector ω of θ;
Compute $\mathbf{c} = \nabla\varphi - \mathbf{n} \times \omega$;
/* Reducing $\mathbf{c}$ via rotating θ */
$E_{\min} = \iint_M \|\mathbf{c}\|^2 \mathrm{d}\sigma$;
while true **do**
  Find the point p with the highest $\|\mathbf{c}\|$;
  Find a loop C containing p and having consistent sign $\mathbf{c} \cdot \mathrm{d}\mathbf{l}$;
  Solve Laplace's equation (18) to compute ψ;
  Rotate θ and ω by angle ψ;
  Update $\mathbf{c}$ with the rotated ω;
  Compute $E_{\mathbf{c}} = \iint_M \|\mathbf{c}^2\| \mathrm{d}\sigma$
  if $E_{\mathbf{c}} < E_{\min}$ **then**
    Save the current cross field θ; $E_{\min} = E_{\mathbf{c}}$;
  else
    Restore the previously saved θ; break;
end
/* Reducing $\mathbf{c}$ via moving singularities */
Compute φ using the singularities of θ;
Set $f_{\max}|_{e_{ij}} = 0$ for each edge $e_{ij} \in E$;
done = false;
while !done **do**
  done = true;
  for each singularity s_i **do**
    Set $\mathbf{f}_c(s_i) = 2\pi I(s_i)\mathbf{c}(s_i)$ for each singularity s_i;
    Find the singularity v_i with the largest force $\|\mathbf{f}_c\|$;
    Find the incident edge e_{ij} with the largest $|\mathbf{f}_{\mathbf{c}} \cdot \mathbf{f}_{\mathbf{c}}|$;
    Compute $f_{ij} = \frac{\mathbf{f}_{\mathbf{c}}(v_i) \cdot \overrightarrow{v_i v_j}}{\|\overrightarrow{v_i v_j}\|}$;
    if $|f_{ij}| > f_{\max}|_{e_{ij}}$ **then**
      $f_{max}|_{e_{ij}} = f_{ij}$;
      Move singularity from s_i to v_j;
      Update θ and ω;
      Compute φ with the same set of singularities of θ;
      Update $\mathbf{c} = \nabla\varphi - \mathbf{n} \times \omega$;
      done = false;
  end
end



Initialization $i = 1$ $i = 3$ $i = 5$ $i = 10$ $i = 20$ $i = 30$ Convergence plot

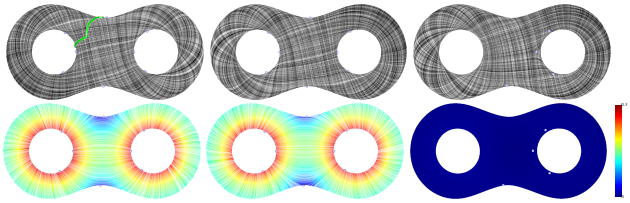
Figure 11. Electrostatic forces can drive singularities to globally optimal positions on surfaces with Gaussian curvatures of the same sign. Random cross fields are generated on a 2D disk ($K = 0$), a hemiellipsoid ($K > 0$) and a monkey saddle surface ($K < 0$). After 30 iterations, . See also the accompanying video.



$(3.4, 24, 1.527)$ $(3.0, 28, 1.120)$ $(2.7, 32, 1.053)$ $(2.5, 34, 1.035)$ $(2.3, 38, 1.089)$ $(2.1, 40, 1.051)$ $(1.7, 42, 1.050)$

Figure 12. Controlling area distortions by the disk radius r_0. The larger the radius, the more singularities generated, which usually leads to parameterization with less area distortion. The 3-tuple below each figure is $-\log_{10} r_0$, the number of singularities $|S|$ and the area distortion metric E_{AD}. The right-most figure shows the r_0-E_{AD}-$|S|$ plots.

force for singularity s as $2\pi I(s)\mathbf{c}(s)$. In each iteration, we find a singularity s_i with the largest force. Then we move it along the field line. To avoid getting stuck in an endless loop, we track the force for each mesh edge. We only continue the movement if the current force is greater than the previously stored force. The algorithm terminates if no singularity can move. If the final energy $E_{\mathbf{c}}$ is small, the cross field is smooth and integrable; otherwise, the directional constraints are incompatible, and the user needs to remove and/or modify some constraints.



(a) $E_{\mathbf{c}} = 0.266$ (b) $E_{\mathbf{c}} = 0.257$ (c) $E_{\mathbf{c}} = 1.182 \times 10^{-4}$

Figure 13. Discrete and continuous adjustments for curl elimination. (a) A smooth cross field with curls. (b) After discrete adjustment of θ, the curl is reduced, but not vanished yet. (c) Fine tuning the locations of singularities makes $\mathbf{c}$ sufficiently small, and the resulting θ becomes integrable.

Our algorithm has only one parameter, the disk radius r_0 which controls the number of singularities. It is fully automatic and does not require any user interaction.

Remark 2. It is also worth noting that since $\|\mathbf{c}\|$ is finite at singularities, we do not need to remove singularities when computing $E_{\mathbf{c}}$. Also, $E_{\mathbf{c}}$ is unitless since the unit of $\|\mathbf{c}\|$ is the reciprocal of edge length.

Remark 3. Though the discrete setting (Section 5) shares most of the spirit of the continuous setting (Section 4), there are two major differences that require us to pay more attention in our algorithm design. First, in the continuous setting, we treat singularities as moving geodesic disks with radius r_0. Since computing geodesic distances is expensive, to simplify our implementation, we model singularities as vertices together with their 1-ring neighboring triangles. To compute the smoothness energy E_{smooth}, we first compute the area integral without the 1-ring triangles incident to singularities and then add a compensation term for each singularity (see Section A.4 in Supplementary Material). Such a term reduces the bias caused by the different triangle sizes. Similarly, we have a compensation term in computing conformal factor φ at singularities (see Section A.9 in Supplementary Material). Second, in the continuous

setting, we simply move each singularity s_i following the direction $\mathbf{F}(s_i)$. In the discrete setting, singularities are always centered at vertices, so we can only move them from vertices to vertices. In other words, we move singularities along mesh edges. To do this, we find the edge e_{ij} that is incident to s_i and has the least angle with $\mathbf{F}(s_i)$, and then move singularity from s_i to v_j.

8. Experimental Results

We implemented our algorithm in C++ and tested it on a PC with Inter(R) Xeon(R) E5-1650 CPU and 16 GB memory. Our algorithm requires only a standard sparse linear solver for computing conformal factor φ. In our implementation, we used Eigen[2] to pre-factor the Laplacian matrix. After that, solving Poisson's equation (8) takes only linear time. Since our method iteratively moves singularities on triangle meshes in both energy minimization and curl-elimination, we assume the input meshes are reasonably dense so that they can provide sufficient degrees of freedom. For low resolution inputs, we apply subdivision to increase the resolution. The model sizes in our paper range from 5K to 50K triangles. Our method is fairly efficient to compute the smooth and integrable cross fields, taking less than 20 seconds for all test models.

Global conformal parameterization. To demonstrate the efficacy of our method, we apply it to compute global conformal parameterization. We first cut the input mesh into a topological disk so that the singularities are on the boundary. Then we adopt the classic MIQ solver [3] to compute an integer-valued parameterization. Thanks to the integrability of our cross fields, the resulting conformal parameterizations match our cross fields closely and are seamless. We adopt two commonly used metrics, the shape distortion E_{SD} and the area distortion E_{AD}, to measure the quality of conformal parameterization. They are derived by analyzing the first fundamental form of a parametrized surface. Specifically, E_{SD} is calculated based on the aspect ratio of an ellipse that is generated by the metric matrix, while E_{AD} is determined from the scalar stretch of the local area. The equality $E_{SD} = 1$ holds when the parameterization is exactly conformal. Since both metrics are no less than one, the closer the value to 1, the better the quality.

Relation between disk radius r_0 and singularity configuration. The disk radius r_0 has an important impact on the number of singularities and the smoothness energy of the cross field. Note that we exclude the geodesic disks when computing the area integral for the smoothness energy E_{smooth}. When the radius increases, the size of the integral domain decreases, resulting in a smaller integral value. This implies that we can add more singularities, each of which has small index, without increasing the total smoothness energy. To ensure that the parameter is independent of the model size, we define the radius $r_0 = 10^{-s}\sqrt{A}$, where A is the surface area. We recommend the exponent $s \in [1, 4]$. Figure 12 shows the parameterization results on Stanford Bunny with varying disk radii. We observe that the number of singularities is roughly positively correlated with r_0. The small fluctuation in Figure 12 (right) is due to the trial-and-error strategy in global pairing (Section 6.1.2).

Local injectivity is a desired feature of parameterization methods. Although our method has no theoretical guarantee of local injectivity, evaluation on the 8i Voxelized Full Bodies (8iVFB) database[3] shows that our method does not violate the local injectivity property. Figure 14 (rightmost) shows a typical example of parameterized model in the database.

Boundary alignment. Our method supports boundary alignment, since it takes boundary curves as the Dirichlet boundary condition in computing cross fields (3). As a result, the filed lines are well aligned with the boundaries. See Figure 15. This boundary alignment feature is particularly desired in free-form architecture design. See Figure 19 for an example.

Directional constraints are highly desired in cross field design and conformal parameterization. In general, adding constraints produces curls in the cross field, making it non-integrable. Fortunately, curls can be eliminated if there are a sufficient number of singularities (see Figure 17). Our method allows the user to specify *sparse hard* directional constraints in a trial-and-error manner (see Figure 24). We observed that sparse constraints are fairly effective to produce geometry-aware cross fields for simple shapes. For example, one constraint on the base of the Greek Sculpture model (Figure 14, rightmost) yields a perfect result, i.e., there are 8 singularities on the corners of the base. Placing 2 constraints on each finger of the hand model, we also obtained cross fields that are well aligned with the cylindrical geometry.

Sharp features. Our method partially supports sharp features via specifying sparse constraints on sharp edges. Figure 25 shows the Fandisk model with 102 constraints, and the resulting cross fields and conformal parameterization are well aligned with most of the sharp features.

Extension. Our electrostatic model can be extended for computing N-RoSy fields for arbitrary $N \in \mathbb{Z}_+$, where the field index $I(s_i)$ is a multiple of $\frac{1}{N}$. Equations (8)(2) and (6) for computing φ and θ, and the algorithms for minimizing the potential energy and eliminating curls remain unchanged, since they do not rely on a particular value of N. Figure 18 shows the result of 6-Rosy field and the induced isotropic triangle mesh.

Comparison with Ben-Chen et al. [1]'s method. Both our method and Ben-Chen et al.'s method use heuristics by con-

[2] http://eigen.tuxfamily.org/

[3] http://plenodb.jpeg.org/pc/8ilabs

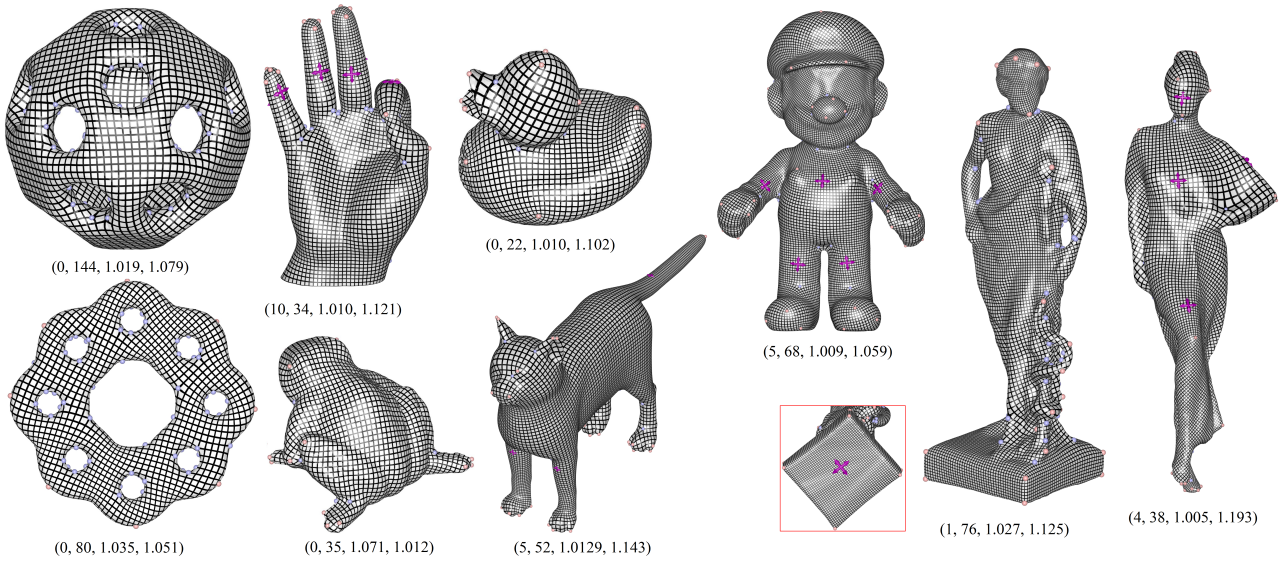


Figure 14. Seamless conformal parameterization induced from our cross fields. Our method works well for smooth surfaces with or without boundaries. The 4-tuple below each figure is the number of constraints $|C|$, the number of singularities N_s, the angle distortion E_{SD} and the area distortion E_{AD}. Image is rendered in high-resolution, allowing zoom-in examination.

Method	Domain	Orthogonal	Efficiency	Boundary alignment	Directional constraints	Dirichlet energy	Integrability	Singularity number	Sharp features
MIQ [3]	3D	Yes	Fair	Yes	Dense, hard/soft	Low	No	Fair	Yes
GODF [18]	3D	Yes	Fast	No	Dense, hard/soft	Low	No	Fair	Yes
FF [23]	3D	No	Fair	Yes	Dense, hard	Low	No	Fair	Yes
IPF [11]	3D	No	Fair	Yes	Dense, hard/soft	Fair	Yes	Fair	Yes
IOQ [12]	3D	Yes	Fair	No	No	Low	No	Small	No
m-harmonic [4]	3D	Yes	Fast	Yes	Dense, hard/soft	Low	No	Fair	No
GL-functional [32]	2D	Yes	Fast	Yes	No	Low	No	Fair	No
OF [34]	3D	Yes	Fair	Yes	Dense, soft	Fair	No	Fair	Yes
Chebshev Net [27]	3D	No	Fair	Yes	No	Low	Yes	Small	Yes
Metric Customization [16]	3D	No	Fair	Yes	Dense hard/soft	Fair	No	Fair	Yes
Ours	3D	Yes	Fair	Yes	Sparse, hard	Low	Yes	Fair	Fair

Table 1. Qualitative comparison of representative cross and directional field methods.

sidering the pair of points which have the largest conformal factor difference. However, the differences between the two methods are fundamental. First, the strategies are totally different. Starting from a singularity-free configuration, their method iteratively adds new singularities at the pair of points with the largest conformal factor difference. Their method stops when the conformal factor difference is below a threshold. Our method starts from a random cross field (which usually contains a large number of singularities) and then adjusts the singularities by moving them following the gradient of conformal factor. To avoid getting stuck at local optimal, we force the two points with the largest conformal difference to move towards each other, hereby reducing the number of singularities. Second, their strategy is based on the intuition that the point with the maximal/minimal conformal factor is a good singularity candidate. Our strategy, in contrast, is to annihilate the pair of points with the largest potential difference, which reduces the gap between the peak and the valley of φ, hereby is likely to reduce the Dirichlet energy of the field and make the field smoother (see Figures 8 and 12). Third, our method allows the user to tune the number of singularities via tuning the disk radius (see Figure 12), while Ben-Chen's method controls the number of singularities by specifying the threshold of conformal factor difference. Fourth, our method aims at computing integrable cross fields, and their method uses conformal factor to scale the metric and then computes a 2D embedding. Our method yields seamless parameterizations and their parameterizations have seams. Fifth, our method supports sparse hard directional constraints and boundary alignment, while Ben-Chen et al.'s method cannot.

Comparison with the GL-functional [32]. Ginzburg-

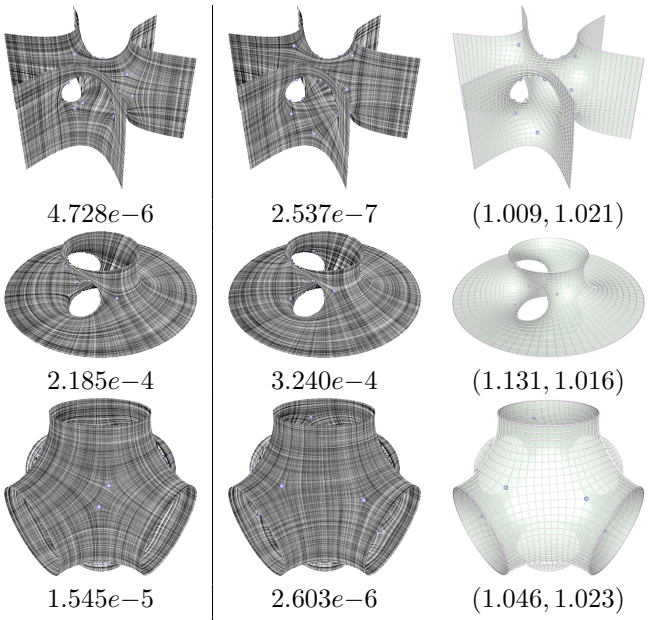


Figure 15. Models with boundaries. The parameter lines are well aligned with the boundaries and the locations of the singularities also roughly capture the global/local symmetry of the input surfaces. No directional constraints are added. The value below each cross field is the smoothness energy and the 2-tuple for the quad meshes indicate the angle and area distortions.

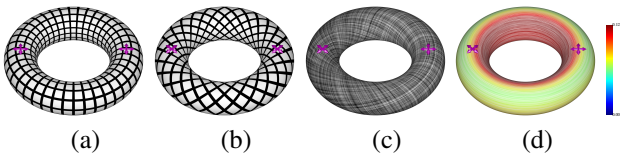


Figure 16. Compatible directional constraints. (a-b) show two examples of conformal parameterization with compatible constraints. (c) shows a pair of incompatible constraints, and the resulting cross field is not curl-free. (d) shows the vector field $\mathbf{c} \neq \mathbf{0}$.

Landau theory is sound and elegant, and can compute boundary-aligned cross fields. However, it does not support directional constraints and cannot guarantee integrability. Also, Viertel and Osting [32] presented only planar cases in their paper. To our knowledge, there is no implementation of GL theory on curved surfaces in the literature. Our method supports sparse directional constraints, allows the user to control the number of singularities, and guarantees integrability. Moreover, our method works for both planar domains and curved surfaces of arbitrary topology, since the singularity operations (e.g., movement, merging and split) are local and our method does not require a global coordinate system. Moreover, in [32], the GL-functional is computed over the entire domain including singularities. In contrast, our method removes singularities and their vicinities to obtain tessellation independent energy. As a by-product, our method allows the user to tune the radius of the geodesic

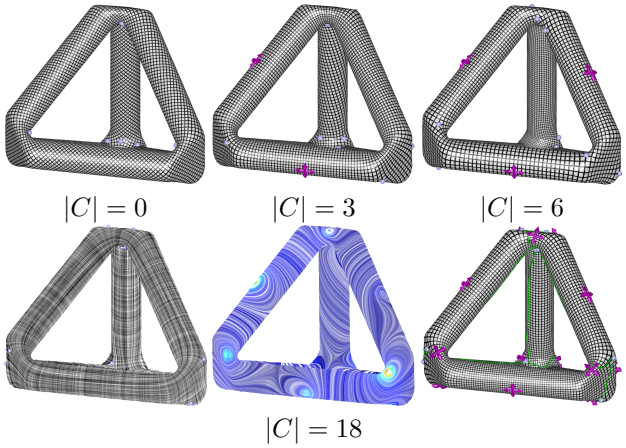


Figure 17. Our method allows sparse hard directional constraints. The first row shows the toy model with 0, 3 and 6 constraints, respectively. We observe that 6 constraints are sufficient to yield high-quality geometry-aware parameterization. The second row shows a cross field with 18 constraints, which is still smooth but not integrable any longer. The non-integrability can be visualized by the vector field $\mathbf{c}$, which does not vanish. As a result, we must cut the model into a topological disk and then compute a conformal parameterization with seam (green curves).

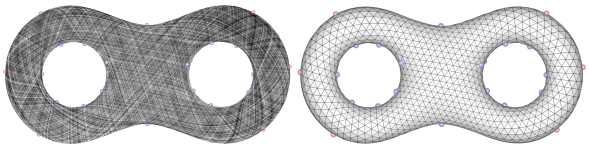


Figure 18. Our method can be extended to compute smooth N-RoSy fields. For the 6-RoSy field on the two-holed torus, there are 16 valence-7 and 6 valence-5 vertices, and the other vertices are regular, i.e., with valence 6.

disk to balance the smoothness energy and the number of singularities, whereas their method cannot control the number of singularities.

Comparison with IOQ [12]. First, IOQ is theoretically sound and has an elegant formulation based on the resistance distance matrix. Our method is loosely based on a physics model that drives singularities to optimal locations. Second, their method can be easily parallelized on GPUs, but is still computationally expensive. For example, their approximate algorithm takes around 20 seconds for a mesh with 50K faces on an Intel Core i7 CPU and an NVIDIA GTX 1080 Ti GPU, while our method takes no more than 3 seconds on the CPU only. Third, IOQ does not guarantee integrability and cannot support directional constraints or boundary alignment, while our method can. Fourth, IOQ is numerically stable and can tolerate geometric noise of certain degree, while our method requires the input meshes are smooth and free of noise. Fifth, for most of the test models, IOQ produces fewer singularities than ours. However, we observed that on high-genus models, their results often have

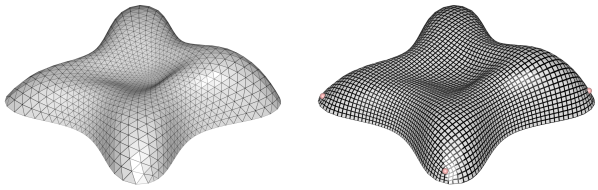


Figure 19. Applying our method to architectural geometry can yield high-quality quadrilateral meshes, which provide more aesthetic value than triangle meshes. This example has 4 singularities and low angle and area distortions $E_{SD} = 1.009$ and $E_{AD} = 1.033$. Thanks to our boundary alignment feature, the parameter lines are well aligned with the surface boundary.

larger area and angle distortions than ours (see Figure 21).

Comparison with other field methods. Table 1 list the major directional field methods with features. We quantitatively compare our method with 5 representative directional field methods: mixed integer quadrangulation (MIQ) [3], globally optimal direction field (GODF) [18], integer-only cross field (IOQ) [12], frame field (FF) [23], and integrable polyvector field (IPF) [11]. Note that MIQ, GODF and IOQ are cross fields, and FF and IPF are non-orthogonal fields.

A major application of cross field is to guide a global parameterization. Our results with and without curl elimination share the same number of singularities and indices. Optimizing their locations can eliminate curls and make the cross field integrable. Thanks to integrability, our parameterizations are well aligned with the cross fields for all test models. To quantitatively measure the alignment, we define the following metric

$$E_{\text{align}} = \iint (h\nabla u - \mathbf{e}_u)^2 + (h\nabla v - \mathbf{e}_v)^2 \mathrm{d}\sigma,$$

where $(\mathbf{e}_u, \mathbf{e}_v)$ are the cross directions and h is the global scale factor [3]. In Figure 21, we compare our method with the other cross field methods in terms of integrability. We observe that the alignment measures of our results are consistently better than the other cross field methods.

IPF can also compute integrable fields, but their fields are non-orthogonal. As a result, their induced parameterizations have larger angle distortions than ours. Since PolyVector fields have larger solution space than cross fields, IPF allows the user to specify more soft constraints in an intuitive manner (e.g., by simple sketches on the model). Our method, which is restricted to cross fields, can only take sparse hard constraints in a trial-and-error manner. GODF supports soft directional constraints, and MIQ supports both hard and soft directional constraints. IOQ does not support directional constraints nor boundary alignment.

We observe that IOQ produces the fewest singularities for most of the testing models. For the simple toy model (Fig. 24 Row 1), our method generates more singularities than the other approaches. These singularities are well placed that faithfully captures the global and local symmetry. For models with non-trivial topology, our method yields comparable number of singularities as with the other methods (see Fig. 24 rows 2 and 3, and Fig. 21).

We also compare our method with a recent approach [29] that computes conformal parameterization with optimal cone singularity and minimal area distortion (MAD). As a convex optimization, MAD is numerically robust and efficient, and yields results with a small number of singularities and a small total cone angles. However, MAD does not support directional constraints and boundary alignment and its produced parameterizations have seams. See Figure 22.

By specifying sparse hard constraints, our method can partially support sharp features. The metric customization method [16] is flexible to support both sparse and dense constraints, and also works well for sharp features. However, the cost is lack of orthogonality. The octahedral frame method [34] minimizes energies with *soft* normal alignment constraints, therefore, it is able to fully support sharp features and compute feature-aligned cross fields. The Chebyshev Net method [27] addresses integrability by adding a soft constraint in the GL-functional, and it also allows boundary alignment constraints and supports sharp features. However, it produces polyvector fields, which in general are not orthogonal.

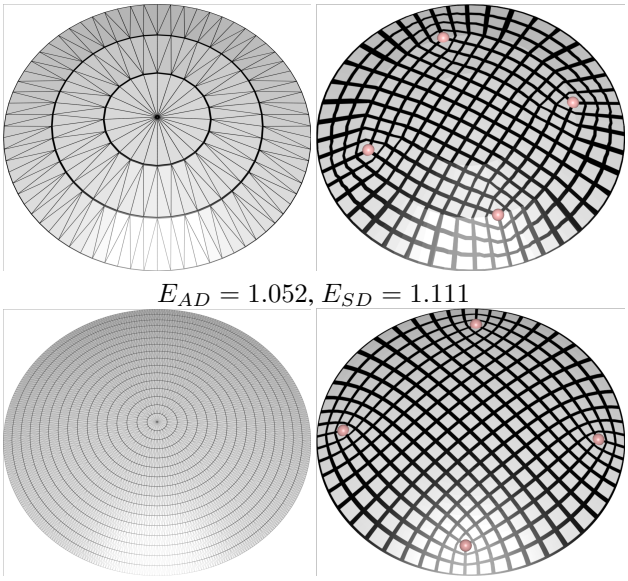


Figure 20. Our method necessitates a triangle mesh of relatively high resolution to adequately provide the necessary degrees of freedom for singularity placement. Row 1: Given a triangle mesh with a high degree of anisotropy and low resolution, the singularities are constrained to the mesh vertices, leading to large angle and area distortions in the resulting parameterization. Row 2: Increasing the mesh resolution can reduce these distortions.

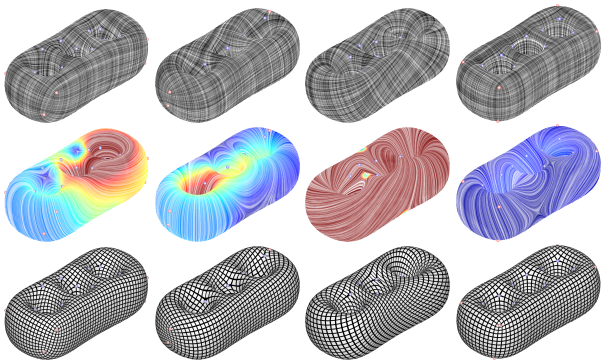


Figure 21. Thanks to integrability, our parameterization is well aligned with the cross field. We observe misalignment in the results of MIQ, GODF and IOQ, which are not integrable. E_{align} is the alignment measure. The 4-tuple is E_{AD}, E_{SD}, the number of position and negative singularities, respectively.

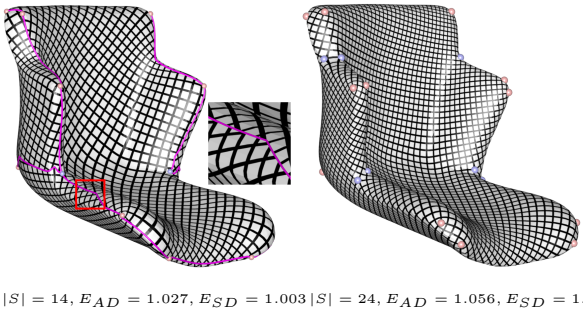


Figure 22. The optimal cone singularity method [29] can produce very few singularities and low distortions. However, their parameterization is not seamless.

9. Conclusions

In the paper, we investigate the integrability problem of cross fields on surfaces of arbitrary topology. A cross field is smooth if it has a locally minimal Dirichlet energy. However, unless the surface is simply connected, a smooth cross field is not necessarily integrable. We prove that a smooth cross field is integrable if $\mathbf{c} \equiv \mathbf{0}$ holds everywhere. Based on the theoretical results, we develop a novel method for computing smooth and integrable cross fields on 3D surfaces of complex topology. Our method is fully automatic and also has a parameter that can balance the number of singularities and the smoothness energy. The proposed method is well suited for smooth models in which exact boundary alignment and sparse hard directional constraints are desired, and can guide seamless conformal parameterization and T-junction-free quadrangulation.

References

[1] M. Ben-Chen, C. Gotsman, and G. Bunin. Conformal flattening by curvature prescription and metric scaling. *Comput. Graph. Forum*, 27(2):449–458, 2008. 2, 7, 12

[2] D. Bommes, M. Campen, H. Ebke, P. Alliez, and L. Kobbelt. Integer-grid maps for reliable quad meshing. *ACM Trans. Graph.*, 32(4):98:1–98:12, 2013. 1, 2

[3] D. Bommes, H. Zimmer, and L. Kobbelt. Mixed-integer quadrangulation. *ACM Trans. Graph.*, 28(3), 2009. 1, 2, 12, 13, 15, 17

[4] C. Brandt, L. Scandolo, E. Eisemann, and K. Hildebrandt. Modeling n-symmetry vector fields using higher-order energies. *ACM Trans. Graph.*, 37(2):18:1–18:18, 2018. 2, 13

[5] A. Bright, E. Chien, and O. Weber. Harmonic global parametrization with rational holonomy. *ACM Transactions on Graphics (TOG)*, 36(4):89, 2017. 2

[6] G. Bunin. A continuum theory for unstructured mesh generation in two dimensions. *Computer Aided Geometric Design*, 25(1):14 – 40, 2008. 1, 2, 3, 4

[7] M. Campen, H. Shen, J. Zhou, and D. Zorin. Seamless parametrization with arbitrary cones for arbitrary genus. *ACM Trans. Graph.*, 39(1):2:1–2:19, 2020. 2

[8] M. Campen and D. Zorin. Similarity maps and field-guided t-splines: a perfect couple. *ACM Trans. Graph.*, 36(4):91:1–91:16, 2017. 2

[9] K. Crane, M. Desbrun, and P. Schröder. Trivial connections on discrete surfaces. *Comput. Graph. Forum*, 29(5):1525–1533, 2010. 1, 2

[10] O. Diamanti, A. Vaxman, D. Panozzo, and O. Sorkine-Hornung. Designing n-PolyVector fields with complex polynomials. *Computer Graphics Forum*, 33(5):1–11, 2014. 2

[11] O. Diamanti, A. Vaxman, D. Panozzo, and O. Sorkine-Hornung. Integrable polyvector fields. *ACM Trans. Graph.*, 34(4):38, 2015. 1, 2, 13, 15

[12] N. Farchi and M. Ben-Chen. Integer-only cross field computation. *ACM Transactions on Graphics (TOG)*, 37(4), 2018. 1, 2, 13, 14, 15, 17

[13] O. Gutan, S. Hegde, E. Berumen, M. Bessmeltsev, and E. Chien. Singularity-free frame fields for line drawing vectorization. *Computer Graphics Forum*, 42, 08 2023. 2

[14] A. Hertzmann and D. Zorin. Illustrating smooth surfaces. In *Proceedings of SIGGRAPH '00*, pages 517–526, 2000. 1, 2

[15] W. Jakob, M. Tarini, D. Panozzo, and O. Sorkine-Hornung. Instant field-aligned meshes. *ACM Trans. Graph.*, 34(6):189–1, 2015. 1

[16] T. Jiang, X. Fang, J. Huang, H. Bao, Y. Tong, and M. Desbrun. Frame field generation through metric customization. *ACM Trans. Graph.*, 34(4):40:1–40:11, July 2015. 2, 13, 15

[17] F. Kälberer, M. Nieser, and K. Polthier. Quadcover - surface parameterization using branched coverings. *Comput. Graph. Forum*, 26(3):375–384, 2007. 1

[18] F. Knöppel, K. Crane, U. Pinkall, and P. Schröder. Globally optimal direction fields. *ACM Trans. Graph.*, 32(4), 2013. 1, 2, 6, 13, 15, 17

[19] X. Liang, M. S. Ebeida, and Y. Zhang. Guaranteed-quality all-quadrilateral mesh generation with feature preservation.

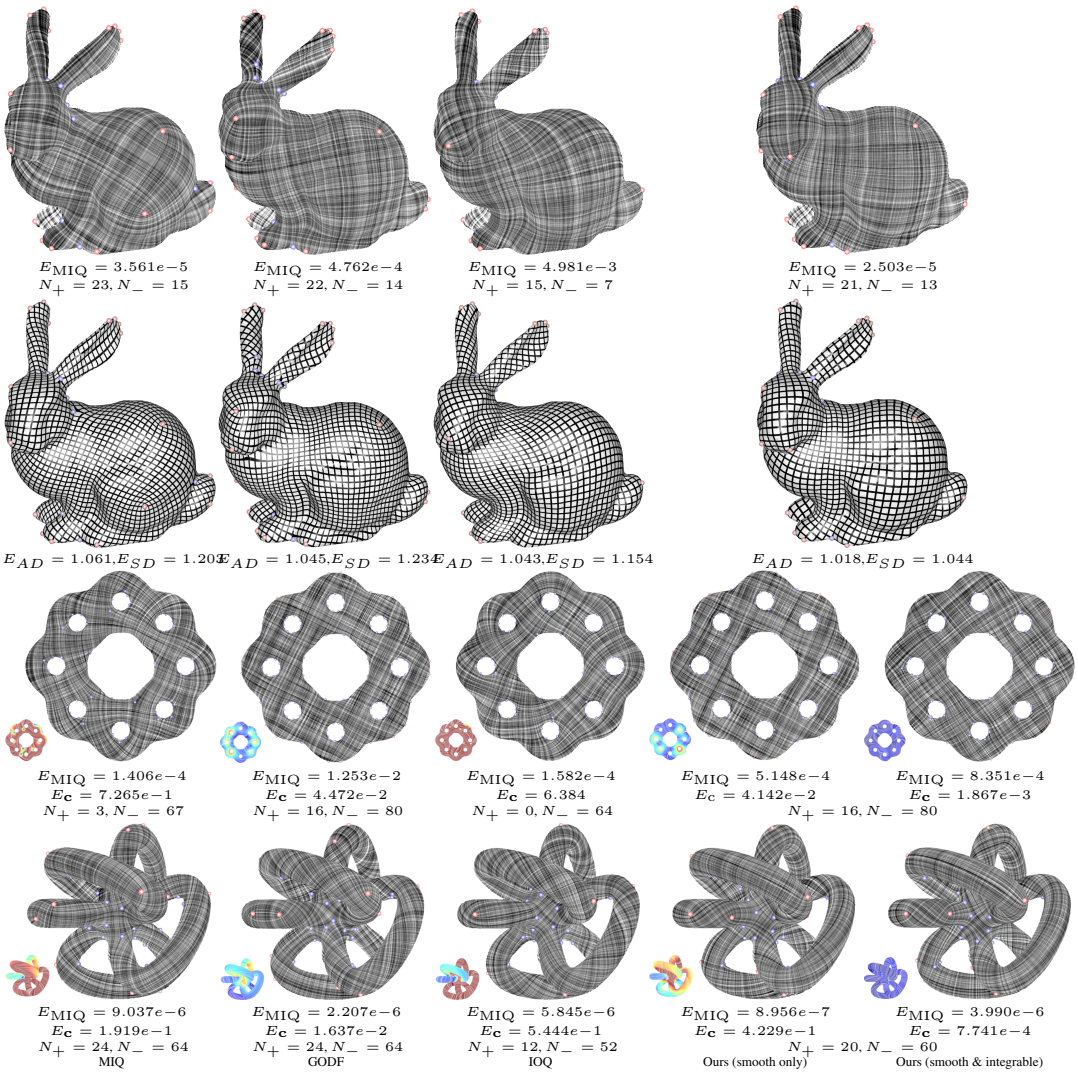


Figure 23. Comparing to other cross field methods, including MIQ [3], GODF [18] and IOQ [12]. For simply connected domains, smooth cross fields are also integrable. But for multiply connected domains, only our method can produce integrable cross fields. The small insets show the vector field $\mathbf{c}$. E_{MIQ} is the MIQ energy [3] for measuring the smoothness of cross fields (without removing the singularities). $E_{\mathbf{c}}$ measures how far the cross field is from an integrable field. N_+ and N_- are the number of singularities with indices $\frac{1}{4}$ and $\frac{-1}{4}$, respectively.

Computer Methods in Applied Mechanics and Engineering, 199(29):2072–2083, 2010. 1

[20] A. Myles and D. Zorin. Global parametrization by incremental flattening. *ACM Trans. Graph.*, 31(4), July 2012. 2

[21] A. Myles and D. Zorin. Controlled-distortion constrained global parametrization. *ACM Trans. Graph.*, 32(4), July 2013. 2

[22] J. Palacios and E. Zhang. Rotational symmetry field design on surfaces. *ACM Trans. Graph.*, 26(3):55, 2007. 1, 2

[23] D. Panozzo, E. Puppo, M. Tarini, and O. Sorkine-Hornung. Frame fields: anisotropic and non-orthogonal cross fields. *ACM Trans. Graph.*, 33(4):134:1–134:11, 2014. 2, 13, 15

[24] N. Ray, W. Li, B. Lévy, A. Sheffer, and P. Alliez. Periodic global parameterization. *ACM Trans. Graph.*, 25(4):1460–1485, 2006. 2

[25] N. Ray, B. Vallet, W. Li, and B. Lévy. N-symmetry direction field design. *ACM Trans. Graph.*, 27(2):10:1–10:13, 2008. 1, 2

[26] A. O. Sageman-Furnas, A. Chern, M. Ben-Chen, and A. Vaxman. Chebyshev nets from commuting polyvector fields. *ACM Trans. Graph.*, 38(6), Nov. 2019. 2

[27] A. O. Sageman-Furnas, A. Chern, M. Ben-Chen, and A. Vaxman. Chebyshev nets from commuting polyvector fields. *ACM Trans. Graph.*, 38(6), Nov. 2019. 2, 13, 15

[28] H. Shen, L. Zhu, R. Capouellez, D. Panozzo, M. Campen, and D. Zorin. Which cross fields can be quadrangulated?: global parameterization from prescribed holonomy signatures. *ACM Transactions on Graphics*, 41:1–12, 07 2022. 2

[29] Y. Soliman, D. Slepčev, and K. Crane. Optimal cone singu-

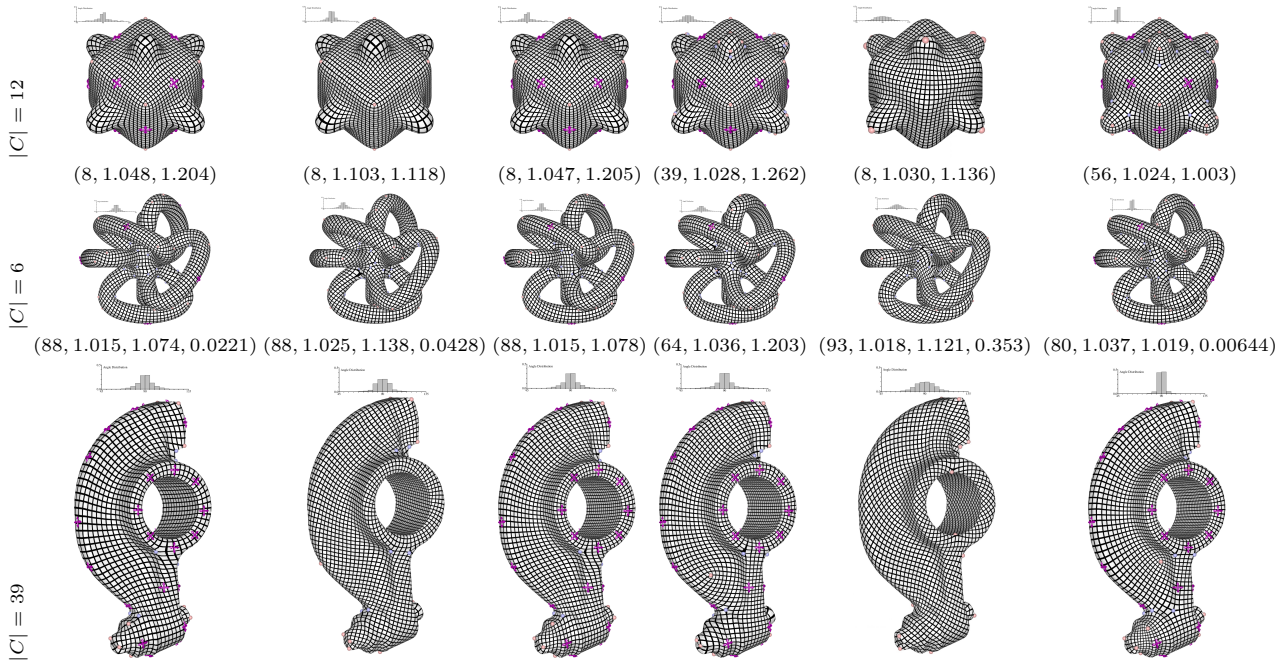


Figure 24. Comparison. The positive and negative singularities are colored in pink and cyan, respectively. The pink crosses are the **hard** directional constraints. $|C|$ is the number of constraints. The numbers below each figure are the number of singularities $|S|$, area distortion E_{AD}, and angle distortion E_{SD}. For cross field methods (i.e., MIQ, GODF, IOQ and ours), the last number is the alignment measure E_{align}. The histograms show the angle distribution. The images are rendered in high resolution, allowing zoom-in examination. Note that GODF supports soft constraints and IOQ does not support directional constraints.

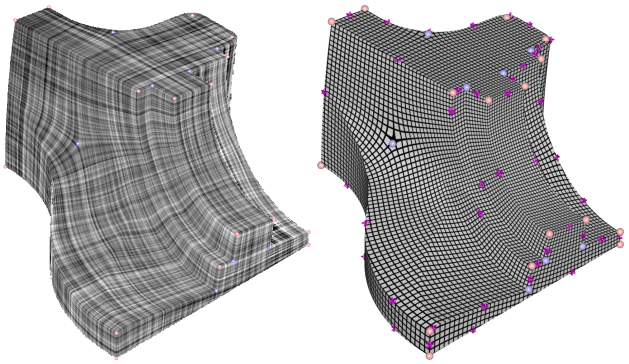


Figure 25. Our method can partially support sharp features by specifying sparse hard directional constraints. $|C| = 102$, $N_+ = 20$, $N_- = 12$, $E_{AD} = 1.01521$, $E_{SD} = 1.06955$.

larities for conformal flattening. *ACM Trans. Graph.*, 37(4), 2018. 2, 15, 16

[30] B. Springborn, P. Schröder, and U. Pinkall. Conformal equivalence of triangle meshes. *ACM Trans. Graph.*, 27(3):77, 2008. 2

[31] A. Vaxman, M. Campen, O. Diamanti, D. Panozzo, D. Bommes, K. Hildebrandt, and M. Ben-Chen. Directional field synthesis, design, and processing. In *Computer Graphics Forum*, volume 35, pages 545–572. Wiley Online Library, 2016. 2

[32] R. Viertel and B. Osting. An approach to quad meshing based on harmonic cross-valued maps and the ginzburg-landau theory. *SIAM Journal on Scientific Computing*, 41, 08 2017. 13, 14

[33] G. Xu, M. Li, B. Mourrain, T. Rabczuk, J. Xu, and S. P. Bordas. Constructing iga-suitable planar parameterization from complex cad boundary by domain partition and global/local optimization. *Computer Methods in Applied Mechanics and Engineering*, 328:175–200, 2018. 1

[34] P. Zhang, J. Vekhter, E. Chien, D. Bommes, E. Vouga, and J. Solomon. Octahedral frames for feature-aligned cross fields. *ACM Trans. Graph.*, 39(3):25:1–25:13, 2020. 2, 13, 15