

Bijjective Density-Equalizing Quasiconformal Map for Multiply Connected Open Surfaces*

Zhiyuan Lyu[†], Gary P. T. Choi[†], and Lok Ming Lui[†]

Abstract. This paper proposes a novel method for computing bijective density-equalizing quasiconformal flattening maps for multiply connected open surfaces. In conventional density-equalizing maps, shape deformations are solely driven by prescribed constraints on the density distribution, defined as the population per unit area, while the bijectivity and local geometric distortions of the mappings are uncontrolled. Also, prior methods have primarily focused on simply connected open surfaces but not surfaces with more complicated topologies. Our proposed method overcomes these issues by formulating the density diffusion process as a quasiconformal flow, which allows us to effectively control the local geometric distortion and guarantee the bijectivity of the mapping by solving an energy minimization problem involving the Beltrami coefficient of the mapping. To achieve an optimal parameterization of multiply connected surfaces, we develop an iterative scheme that optimizes both the shape of the target planar circular domain and the density-equalizing quasiconformal map onto it. In addition, landmark constraints can be incorporated into our proposed method for consistent feature alignment. The method can also be naturally applied to simply connected open surfaces. By changing the prescribed population, a large variety of surface flattening maps with different desired properties can be achieved. The method is tested on both synthetic and real examples, demonstrating its efficacy in various applications in computer graphics and medical imaging.

Key words. density-equalizing map, quasiconformal map, Beltrami coefficient, area-preserving parameterization, diffusion, multiply connected surface

MSC codes. 65D18, 68U05, 30C70, 76R50

DOI. 10.1137/23M1594376

1. Introduction. In recent years, three-dimensional (3D) data have become increasingly ubiquitous in various fields, including computer graphics, computer vision, and medical imaging. In computer graphics, 3D models are used to create realistic animations, games, and virtual environments. In computer vision, 3D data are used for 3D scene understanding, object recognition, and tracking. In medical imaging, 3D data are used to create detailed models of organs and tissues, aiding in diagnosis and treatment planning. However, processing and manipulating 3D data can be challenging due to its complex and irregular nature. One important technique for addressing this challenge is surface parameterization, which involves mapping a surface in \mathbb{R}^3 onto a simpler parameter domain while preserving certain desired

*Received by the editors August 15, 2023; accepted for publication (in revised form) January 2, 2024; published electronically March 28, 2024.

<https://doi.org/10.1137/23M1594376>

Funding: This work was supported in part by the CUHK Vice-Chancellor Early Career Professorship Research Startup Fund (Project Code 4933624) (to the second author) and the Croucher Foundation Start-up Allowance (to the second author). This work was also supported by HKRGC GRF (14306721) (to the third author).

[†]Department of Mathematics, The Chinese University of Hong Kong, Hong Kong (zylyu@math.cuhk.edu.hk, ptchoi@math.cuhk.edu.hk, lmlui@math.cuhk.edu.hk).

properties. Surface parameterizations have a wide range of applications in processing 3D data. For example, remeshing involves generating a new mesh with desired properties from an existing mesh, which can be achieved more easily on a parameterized surface. Visualization of 3D models is also facilitated by parameterization, as it allows for efficient rendering and texture mapping. One can also solve partial differential equations (PDEs) on a surface by parameterizing it onto a simpler two-dimensional (2D) domain where the PDEs can be solved more easily. Once the solution is obtained on the 2D domain, it can be mapped back to the original surface using the inverse of the parameterization map.

Different surface parameterization methods have been proposed depending on the properties of the surface parameterizations required for a specific application. Conformal parameterizations, for example, preserve angles and shapes locally and are commonly used in texture mapping and geometric processing. By contrast, area-preserving parameterizations preserve areas and are useful in physical simulations where mass is conserved. Another type of surface parameterizations is the distance-preserving parameterization, which preserves distances between points on the surface.

Density-equalizing cartograms have recently gained attention in the field of surface parameterizations as a type of map projection technique. In [20], Gastner and Newman proposed a novel algorithm for generating area cartograms based on the principle of density diffusion. The method deforms a given map such that the density of a prescribed quantity, defined by the population per unit area, becomes constant throughout the deformed map. The technique has found widespread use in data visualization, particularly in the visualization of sociological data such as global population, income, and age-of-death in different regions [18]. Recently, Choi and Rycroft [11] proposed a finite-element approach for computing density-equalizing flattening maps of simply connected open surfaces in \mathbb{R}^3 onto a parameter domain in \mathbb{R}^2 , which has later been extended and applied to medical imaging [5] and 3D microstructure morphology [47, 48]. However, there remains a gap in the literature with respect to the computation of density-equalizing flattening maps for multiply connected open surfaces. Additionally, current approaches for computing density-equalizing parameterizations are limited in their ability to control local geometric distortion and bijectivity. These challenges pose significant obstacles to the applicability of these approaches in practical applications.

Motivated by this, we propose in this paper a novel method for computing bijective density-equalizing quasiconformal (DEQ) flattening maps for multiply connected open surfaces, with controlled local geometric distortions and prescribed constraints on the density distribution (see Figure 1). Our proposed method utilizes quasiconformal flows to control the local geometric distortion during the diffusion process of the density distribution. Specifically, by formulating the diffusion process as a flow of the Beltrami coefficient, which measures the quasiconformality of the map, we can effectively control the local geometric distortion of the map by minimizing an energy functional involving the Beltrami coefficient terms. The bijectivity of the map is guaranteed by a projection operator that truncates the magnitude of the Beltrami coefficient. Our algorithm is capable of generating a large variety of flattening maps with different desired properties for various real-world applications by changing the initial population distribution. Moreover, the proposed method can flatten a multiply connected open surface to a 2D circular domain, i.e., a unit disk with inner circular holes, where the size and location of the inner holes are further optimized to achieve quasiconformality and

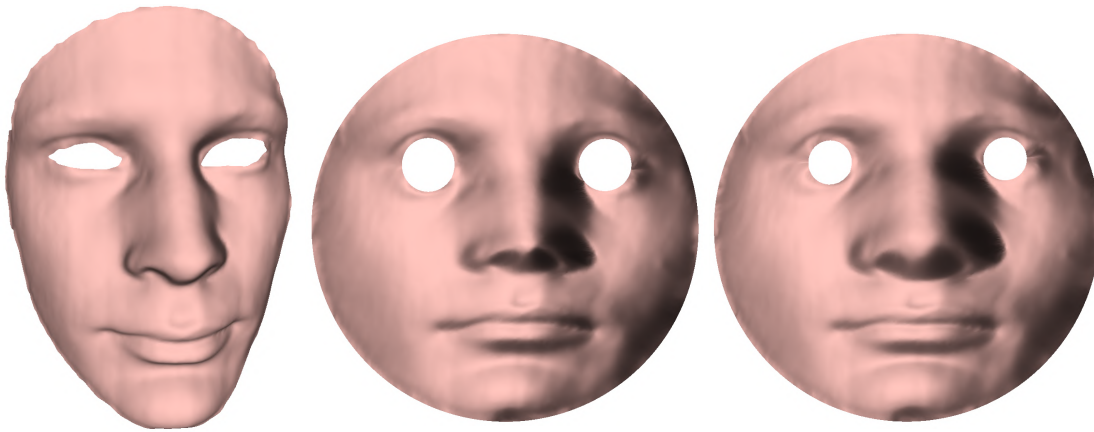


Figure 1. *Bijjective DEQ maps obtained by our proposed algorithm. Given a multiply connected open surface and a prescribed density distribution, our method optimizes both the shape of the target 2D circular domain and the flattening map of the surface onto the domain to achieve density equalization, with the bijectivity of the mapping guaranteed by quasiconformal theory. By changing the prescribed density distribution, different effects can be achieved. For instance, we can parameterize a human face (left) onto the plane with the nose shrunk (middle) or enlarged (right) by changing the prescribed density distribution at the nose. Note that the size and position of the inner circular holes are also optimized and hence different in the two parameterization results.*

density equalization across the entire target domain. Additionally, our algorithm can accommodate interior landmark constraints to ensure consistent feature alignment. The method is also naturally applicable to simply connected open surfaces. The effectiveness of our proposed algorithm has been verified through testing on both synthetic and real-world examples, showcasing its potential in numerous applications within the fields of computer graphics and medical imaging.

The contributions of this paper can be summarized as follows:

1. We present an algorithm for computing density-equalizing surface parameterizations that overcomes the limitations of the conventional approaches in terms of bijectivity and quasiconformality.
2. To the best of our knowledge, this is the first work addressing the density-equalizing mapping problem as a quasiconformal flow problem, which allows us to incorporate quasiconformal techniques to minimize local geometric distortion in the density-equalizing parameterization.
3. Our algorithm is capable of computing a bijective DEQ parameterization for multiply connected open surfaces, which cannot be handled by conventional approaches. It can also be naturally applied to simply connected open surfaces.
4. Our algorithm optimizes the geometry of the target 2D circular domain, i.e., the centers and radii of the inner circles in the circular domain, to obtain a parameterization with reduced geometric distortion for multiply connected surfaces.
5. Our algorithm is the first density-equalizing mapping approach that incorporates feature landmark alignments, resulting in a bijective landmark-matching DEQ map.
6. Our proposed algorithm can be applied to various practical applications including texture mapping, surface remeshing, and medical visualization, demonstrating the efficacy and practicality of our method.

The organization of the paper is as follows. In section 2, we review the previous works on surface parameterization and density-equalizing maps. In section 3, we introduce the mathematical concepts related to our work. In section 4, we present our proposed mathematical models for computing bijective DEQ maps. In section 5, we describe the details of our proposed algorithms. Experimental results are presented in section 6 for demonstrating the effectiveness of our proposed method. In section 7, we present applications of our method in different fields. We conclude our work and discuss future directions in section 8.

2. Related work. Over the past few decades, surface parameterization has been widely studied in the area of geometry processing, graphics, and computer vision (see [19, 27, 49] for surveys). In particular, since isometric (both angle-preserving and area-preserving) parameterization is generally impossible to achieve for nondevelopable surfaces, a large variety of surface parameterization algorithms have been proposed for minimizing either the angle distortion (conformal parameterization) or the area distortion (area-preserving parameterization).

Conformal parameterization [22] preserves the local geometry of surfaces but leads to area distortion. Existing conformal parameterization methods for simply connected open surfaces include least-squares conformal map [33], holomorphic 1-form [21], Yamabe flow [40, 53], spectral conformal mapping [42], fast disk conformal map [14], conformal energy minimization [56], linear disk conformal map [8], and parallelizable global conformal parameterization [6, 7]. Compared with simply connected open surfaces, the conformal mapping of multiply connected surfaces is less studied. Existing conformal parameterization methods for multiply connected surfaces include the Koebe's iteration method [29], generalized Koebe's method [59], Laurent series [30], discrete conformal equivalence [2], and polyannulus conformal map [4].

Quasiconformal maps are a generalization of conformal maps. In recent years, there has been an increasing interest in quasiconformal surface parameterization methods [9]. For instance, Zeng et al. [58] developed the auxiliary metric method for computing surface quasiconformal mappings. Lui et al. [39] developed a method for optimizing surface registrations using Beltrami holomorphic flow. In [1, 35], Lipman et al. proposed methods for computing bounded distortion mappings. Weber, Myles, and Zoring [54] developed a method for computing extremal quasiconformal maps minimizing the maximal conformal distortion. In [3], Chien, Levi, and Weber developed a framework for bounded distortion parameterization in the space of metrics. In [36], Lui et al. developed the linear Beltrami solver (LBS) method for efficiently computing quasiconformal maps. Other recent works on quasiconformal surface parameterization include the quasiconformal energy minimization [25], fast spherical quasiconformal parameterization [10], least-squares quasiconformal map [45], parallelizable global quasiconformal map [62], and disk Beltrami solver [23].

Area-preserving parameterization can preserve the area measure of surfaces while inducing angle distortion. Compared with conformal and quasiconformal parameterizations, there are only a few works focusing on area-preserving mappings. Existing area-preserving parameterization methods include the locally authalic map [17], Lie advection [63], optimal mass transport [16, 61], density-equalizing map (DEM) [5, 11] and stretch energy minimization [55, 57].

It is worth noting that there are also various parameterization approaches aiming to balance the trade-offs between conformal and area distortions, such as the most isometric

parameterizations [26], isometric metric [51], optimized conformal parameterization with controllable area distortions [32], isometric distortion energy minimization [46], and combined quasiconformal and optimal mass transportation spherical parameterization [41]. Besides, there are mapping methods aiming to achieve low geometric distortion while satisfying prescribed landmark constraints, such as the landmark-constrained optimized conformal parameterization [13, 37, 38, 52, 53] and quasiconformal landmark registration [31]. Other related parameterization methods include the optimal conformal parameterization for surfaces with arbitrary topology [34], composite majorization [50], and the simplicial complex augmentation framework [28].

3. Mathematical background. This work aims to propose a method for computing bijective DEQ maps for connected open surfaces. The method relies on the physical principle of density diffusion and quasiconformal theory. In this section, we review some basic concepts of quasiconformal theory and density-equalizing maps.

3.1. Quasiconformal theory. A surface \mathcal{S} with a conformal structure is called a *Riemann surface*. Given two Riemann surfaces \mathcal{M} and \mathcal{N} , a map $f : \mathcal{M} \rightarrow \mathcal{N}$ is *conformal* if it preserves the surface metric up to the *conformal factor*. Quasiconformal mapping is an extension of conformal mapping in that it allows bounded conformal distortions. Intuitively, conformal mappings map infinitesimal circles to infinitesimal circles, while quasiconformal mappings map infinitesimal circles to infinitesimal ellipses with bounded eccentricity. Mathematically, $f : \mathbb{C} \rightarrow \mathbb{C}$ is *quasiconformal* if and only if it satisfies the Beltrami equation

$$(3.1) \quad \frac{\partial f}{\partial \bar{z}} = \mu(z) \frac{\partial f}{\partial z}$$

for some complex-valued function μ satisfying $\|\mu\|_\infty < 1$. μ is called the *Beltrami coefficient*, which is a measure of quasiconformality, i.e., how far the map at each point deviates from a conformal map. Infinitesimally, around a point p , with respect to its local parameter, f can be expressed as follows:

$$(3.2) \quad f(z) = f(p) + f_z(p)z + f_{\bar{z}}(p)\bar{z} = f(p) + f_z(p)(z + \mu(p)\bar{z}).$$

It is easy to see that f is conformal around p if and only if $\mu(p) = 0$. Inside the local parameter domain, f may be considered as a map composed of a translation to $f(p)$ together with a stretch map $S(z) = z + \mu(p)\bar{z}$, which is postcomposed by a multiplication of $f_z(p)$, which is conformal. All the quasiconformal distortion of $S(z)$ is caused by $\mu(p)$. Specifically, $S(z)$ maps a small circle to a small ellipse (see Figure 2 for an illustration). Meanwhile, we can determine the angles of the directions of maximal magnification and shrinkage and their magnitude by $\mu(p)$. The maximal dilation of f is given by

$$(3.3) \quad K(f) = \frac{1 + \|\mu\|_\infty}{1 - \|\mu\|_\infty}.$$

Suppose $f : \Omega_1 \rightarrow \Omega_2$ and $g : \Omega_2 \rightarrow \Omega_3$ are two quasiconformal maps with the Beltrami coefficients μ_f and μ_g respectively. Then, the Beltrami coefficient of the composition map $g \circ f : \Omega_1 \rightarrow \Omega_3$ is given by

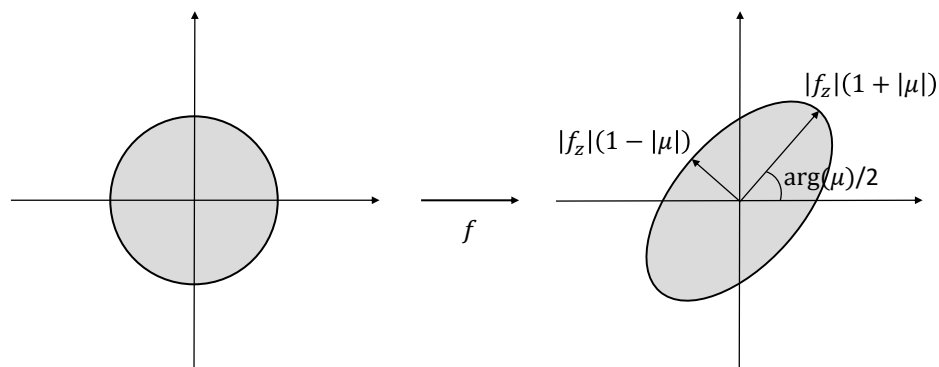


Figure 2. An illustration of how the Beltrami coefficient μ determines the quasiconformal distortion of a map f . Specifically, the maximal magnification factor is given by $|f_z|(1 + |\mu|)$, the maximal shrinkage factor is given by $|f_z|(1 - |\mu|)$, and the orientation change is given by $\arg(\mu)/2$.

$$(3.4) \quad \mu_{g \circ f} = \frac{\mu_f + (\bar{f}_z/f_z)(\mu_g \circ f)}{1 + (\bar{f}_z/f_z)\bar{\mu}_f(\mu_g \circ f)}.$$

Angle distortions of a map can be eliminated using a suitable composition map. Specifically, for any given map f , by finding another map g with $\mu_g = \mu_{f^{-1}}$, we can ensure that $\mu_{g \circ f} = 0$. This implies that the composition map $g \circ f$ is conformal.

For quasiconformal maps of Riemann surfaces, one can generalize the concept of Beltrami coefficients to Beltrami differentials via the local charts of the surfaces. A *Beltrami differential* $\mu(z) \frac{d\bar{z}}{dz}$ on a Riemann surface \mathcal{S} is an assignment to each chart (U_α, ϕ_α) of an L_∞ complex-valued function μ_α , defined on local parameter z_α such that

$$(3.5) \quad \mu(z_\alpha) \frac{d\bar{z}_\alpha}{dz_\alpha} = \mu(z_\beta) \frac{d\bar{z}_\beta}{dz_\beta}$$

on the domain which is also covered by another chart (U_β, ϕ_β) , where $\frac{d\bar{z}_\beta}{dz_\beta} = \frac{d}{dz_\alpha} \phi_{\alpha\beta}$ and $\phi_{\alpha\beta} = \phi_\beta \circ \phi_\alpha^{-1}$. Note that the Beltrami coefficient can be used to represent the Beltrami differential of the surface if the surface can be covered by a simple chart. As our work focuses on connected open surfaces, the Beltrami coefficient is used.

3.2. Linear Beltrami solver. Here we briefly describe the *LBS* method [36] for reconstructing a quasiconformal map $f = u + iv$ from its Beltrami coefficient $\mu = \rho + i\tau$. Consider the real and imaginary parts of the Beltrami equation separately:

$$(3.6) \quad \mu(x, y) = \frac{(u_x - v_y) + i(v_x + u_y)}{(u_x + v_y) + i(v_x - u_y)}.$$

From this formula, v_x and v_y can be expressed as a combination of u_x and u_y ,

$$(3.7) \quad \begin{pmatrix} v_y \\ -v_x \end{pmatrix} = A \begin{pmatrix} u_x \\ u_y \end{pmatrix},$$

where

$$(3.8) \quad A = \begin{pmatrix} \alpha_1 & \alpha_2 \\ \alpha_2 & \alpha_3 \end{pmatrix}, \quad \alpha_1 = \frac{(\rho-1)^2 + \tau^2}{1 - \rho^2 - \tau^2}, \quad \alpha_2 = -\frac{2\tau}{1 - \rho^2 - \tau^2}, \quad \alpha_3 = \frac{1 + 2\rho + \rho^2 + \tau^2}{1 - \rho^2 - \tau^2}.$$

Similarly, u_x and u_y can be expressed as

$$(3.9) \quad \begin{pmatrix} -u_y \\ v_x \end{pmatrix} = A \begin{pmatrix} v_x \\ v_y \end{pmatrix}.$$

Since

$$(3.10) \quad \nabla \cdot \begin{pmatrix} v_y \\ -v_x \end{pmatrix} = v_{xy} - v_{xy} = 0 \quad \text{and} \quad \nabla \cdot \begin{pmatrix} -u_y \\ u_x \end{pmatrix} = u_{xy} - u_{xy} = 0,$$

from the above equations, we have

$$(3.11) \quad \nabla \cdot \left(A \begin{pmatrix} u_x \\ u_y \end{pmatrix} \right) = 0 \quad \text{and} \quad \nabla \cdot \left(A \begin{pmatrix} v_x \\ v_y \end{pmatrix} \right) = 0.$$

In the discrete case, the above equations are two sparse symmetric positive definite linear systems. We can obtain u_x , u_y , v_x , and v_y for any given Beltrami coefficient μ . In this paper, we denote the above method by $f = \mathbf{LBS}(\mu)$.

3.3. Density-equalizing maps. Gastner and Newman [20] proposed a method for producing area cartograms based on density diffusion. Given a planar domain, a positive quantity called *population* is defined at every point on it. A density field ρ is then defined by the population per unit area. For instance, one can consider a geographical map of a country and define the quantity as the population of different regions. The density ρ is then the population density of different regions. Density-equalizing maps aim to deform the domain by enlarging the regions with higher ρ and shrinking the regions with lower ρ (see Figure 3 for an illustration). This can be achieved by equalizing ρ through the advection equation

$$(3.12) \quad \frac{\partial \rho}{\partial t} + \nabla \cdot \mathbf{j} = 0,$$

where \mathbf{j} is the density flux. By Fick's first law, the flux is given by

$$(3.13) \quad \mathbf{j} = -\nabla \rho.$$

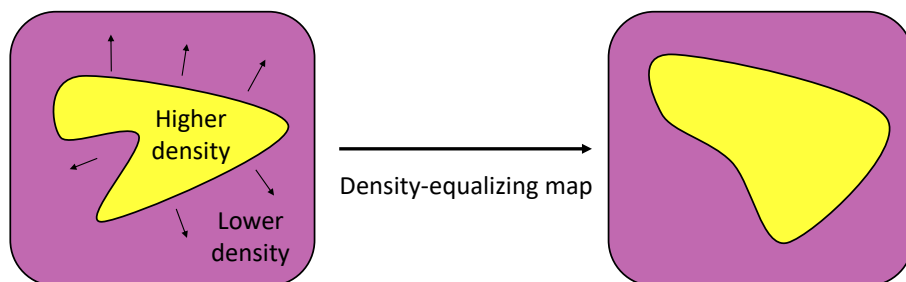


Figure 3. An illustration of the density-equalizing maps. The density diffusion creates a velocity field that enlarges regions with higher density and shrinks regions with lower density until the density is equalized.

This yields the diffusion equation

$$(3.14) \quad \frac{\partial \rho}{\partial t} - \Delta \rho = 0.$$

By the definition of flux, we have $\mathbf{j} = \mathbf{v}(\mathbf{r}, t)\rho$, where $\mathbf{v}(\mathbf{r}, t)$ is the velocity of tracers carried by the flux. Therefore, we have

$$(3.15) \quad \mathbf{v}(\mathbf{r}, t) = \frac{\mathbf{j}}{\rho} = -\frac{\nabla \rho}{\rho}.$$

One can see that $\frac{\nabla \rho}{\rho}$ is independent of the absolute scale of ρ .

As we solve the diffusion equation to reach a steady state, the domain will be deformed according to the velocity field. Finally, the density of the domain will be equal everywhere. To describe the deformation of the domain, the displacement of the tracers is given by

$$(3.16) \quad \mathbf{r}(t) = \mathbf{r}(0) + \int_0^t \mathbf{v}(\mathbf{r}, \tau) d\tau,$$

where $\mathbf{r}(0)$ is the initial state of the domain. When $t \rightarrow \infty$, the tracers $\mathbf{r}(t)$ produce the final 2D domain, which is density equalizing. However, note that the domain will expand infinitely if there is no constraint on its boundary shape. To avoid infinite expansion, a large auxiliary region (also known as a *sea*) surrounding the domain of interest is used. More specifically, by defining the density in the sea as the average density of the domain of interest and including the sea in the computation of the diffusion cartogram, the domain of interest will not expand infinitely. In [11], Choi and Rycroft extended the idea of diffusion cartogram for surface mappings and proposed a finite-element approach for computing density-equalizing flattening maps for simply connected open surfaces.

While the above approaches have been successfully applied to a wide range of problems, they are limited to 2D planar domains and simply connected open surfaces but not general multiply connected surfaces. Also, there is no theoretical guarantee for the bijectivity of the resulting density-equalizing maps in the computation. More specifically, there may be mesh foldovers in the mapping results. Moreover, as the deformation by density-equalizing maps is merely based on the prescribed density, these methods do not preserve other local geometric structures of the surfaces. Our work aims to address these issues and achieve bijective DEQ maps.

4. Proposed mathematical models. Let \mathcal{S} be a multiply connected open surface in \mathbb{R}^3 , with $\partial \mathcal{S} = \gamma_0 - \gamma_1 - \dots - \gamma_k$ representing its boundaries, where $k \geq 1$, γ_0 is the outer boundary, and $\gamma_1, \dots, \gamma_k$ are the inner boundaries. Suppose a population is prescribed everywhere on \mathcal{S} . Our goal is to obtain an optimal quasiconformal map $f: \mathcal{S} \rightarrow \mathcal{D} \subset \mathbb{R}^2$ such that the final density (population per unit area) after applying f becomes a constant. Here, \mathcal{D} is a planar circular domain with unit outer radius, i.e., a unit disk with k inner circular holes,

$$(4.1) \quad \mathcal{D} = \mathbb{D} \setminus \bigcup_{i=1}^k B_{r_i}(c_i),$$

where \mathbb{D} denotes the unit disk, $r_i \in \mathbb{R}$, $c_i \in \mathbb{C}$, $B_{r_i}(c_i) = \{z \in \mathbb{D} : |z - c_i| < r_i\}$ for $i = 1, \dots, k$, $\bigcup_{i=1}^k B_{r_i}(c_i) \subset \mathbb{D}$, and $B_{r_i}(c_i) \cap B_{r_j}(c_j) = \emptyset$ for any $1 \leq i, j \leq k$ with $i \neq j$.

4.1. Bijective DEQ map. For ease of computation, we divide the desired optimal quasiconformal map $f : \mathcal{S} \rightarrow \mathcal{D}$ into two parts, $f = \tilde{f} \circ g_0$, where $g_0 : \mathcal{S} \rightarrow \mathcal{D}^0$ is a conformal flattening map of \mathcal{S} onto a planar circular domain \mathcal{D}^0 , and $\tilde{f} : \mathcal{D}^0 \rightarrow \mathcal{D}$ is a planar quasiconformal map from \mathcal{D}^0 to an optimal planar circular domain \mathcal{D} . Then, the problem of finding f can be simplified as the problem of finding \tilde{f} on the plane. Here, we propose a variational model for finding the optimal \tilde{f} that minimizes an energy functional E_{DEQ} consisting of a density term and two Beltrami coefficient terms,

$$(4.2) \quad \tilde{f} = \arg \min_{g: \mathcal{D}^0 \rightarrow \mathcal{D}} E_{\text{DEQ}}(g) := \arg \min_{g: \mathcal{D}^0 \rightarrow \mathcal{D}} \left\{ \int_{\mathcal{D}^0} |\nabla \rho(g)|^2 + \alpha \int_{\mathcal{D}^0} |\mu(g)|^2 + \beta \int_{\mathcal{D}^0} |\nabla \mu(g)|^2 \right\}$$

subject to

$$(4.3) \quad \|\mu(\tilde{f})\|_{\infty} < 1,$$

$$(4.4) \quad (\tilde{f} \circ g_0)(\gamma_i) = \partial B_{r_i}(c_i) \quad \text{for } 0 \leq i \leq k.$$

In this context, $\nabla \rho$ denotes the gradient of the density function ρ , $\mu(g)$ and $\mu(\tilde{f})$ represent the Beltrami coefficients of g and \tilde{f} , respectively, and $\partial B_{r_i}(c_i)$ denotes the outer boundary. Here, α and β are two user-defined nonnegative weighting parameters for controlling the quasiconformality. Specifically, the first term $\int_{\mathcal{D}^0} |\nabla \rho(g)|^2$ ensures that the spatial variation of the density function ρ decreases. The second term $\int_{\mathcal{D}^0} |\mu(g)|^2$ helps reduce the quasiconformal distortion and ensure the bijectivity of the mapping. Meanwhile, the third term $\int_{\mathcal{D}^0} |\nabla \mu(g)|^2$ helps enhance the smoothness of the mapping. As our focus is on the density-equalizing effect of the mapping, both α and β should be small. We call the resulting composition map $f = \tilde{f} \circ g_0$ the *bijective DEQ map* for \mathcal{S} .

The Jacobian $J_{\tilde{f}}$ of \tilde{f} can be represented by

$$(4.5) \quad J_{\tilde{f}} = \left| \frac{\partial \tilde{f}}{\partial z} \right|^2 (1 - |\mu(\tilde{f})|^2).$$

One can see that the constraint on the norm of μ implies $J_{\tilde{f}} > 0$ everywhere. By the inverse function theorem, the mapping \tilde{f} is thus locally diffeomorphic everywhere. Meanwhile, $|J_{\tilde{f}}| > 0$ also implies that \tilde{f} induces a Riemannian metric on \mathcal{S} . More specifically, an auxiliary metric $|dz + \mu \bar{d}z|^2$ can be induced by the Beltrami coefficient μ on \mathcal{D}^0 . Note that \tilde{f} is a conformal map under the auxiliary metric. By the Riemann mapping theorem for multiply connected domains [24], every multiply connected Riemann surface with a valid Riemannian metric can be mapped by a conformal and one-to-one transformation onto a planar circular domain. Therefore, \tilde{f} is a global bijection on \mathcal{D}^0 . It is easy to see that if \tilde{f} is locally diffeomorphic everywhere and globally bijective, \tilde{f} is a diffeomorphism. Consequently, the composition map $f = \tilde{f} \circ g_0$ is also a diffeomorphism.

Moreover, a minimizer of the above optimization problem (4.2) is guaranteed to exist by the following proposition.

Proposition 4.1 (existence of the minimizer of E_{DEQ}). *Let*

$$(4.6) \quad \mathcal{A} = \{\nu \in C^1(\mathcal{D}^0) : \|D\nu\|_\infty \leq C_1; \|\nu\|_\infty \leq 1 - \epsilon; \nu \text{ is the Beltrami coefficient of a quasiconformal map}\}$$

for some $C_1 > 0$ and small $\epsilon > 0$, where \mathcal{D}^0 is a circular domain. Then E_{DEQ} has a minimizer in \mathcal{A} . In fact, \mathcal{A} is compact.

Proof. Here we follow the approach in [31] to prove that \mathcal{A} is complete and totally bounded. First, note that the existence of conformal maps (with Beltrami coefficient $\nu = 0$) implies that \mathcal{A} is nonempty. Now, we prove that \mathcal{A} is complete. Define $\|\mu\|_s = \|\mu\|_\infty + \|D\mu\|_\infty$. Let $\{\nu_n\}_{n=1}^\infty$ be a Cauchy sequence in \mathcal{A} under the norm $\|\cdot\|_s$. It is worth noting that $\{\nu_n\}_{n=1}^\infty$ and $\{D\nu_n\}_{n=1}^\infty$ are two Cauchy sequences with respect to the $\|\cdot\|_\infty$ norm. Since $\mathcal{A} \subset W^{1,\infty}(\mathcal{D}^0)$ and $W^{1,\infty}$ is complete, $D\nu_n$ converges uniformly to some $h \in W^{1,\infty}(\mathcal{D}^0)$. Moreover, h is continuous since every $D\nu_n$ is continuous. Besides, ν_n is uniformly convergent in $W^{1,\infty}$ as well. Based on the above, $\nu_n \rightarrow \nu$ and $D\nu_n \rightarrow D\nu$ uniformly for some $\nu \in C^1(\mathcal{D}^0)$. Additionally, $\|\nu_n\|_\infty \leq 1 - \epsilon$ and $\|D\nu_n\|_\infty \leq C_1$ for every n imply that $\|\nu\|_\infty \leq 1 - \epsilon$ and $\|D\nu\|_\infty \leq C_1$, respectively. Therefore, $\nu \in \mathcal{A}$. Consequently, \mathcal{A} is complete.

Next, we prove that \mathcal{A} is totally bounded. To achieve this, we will show that for any given $\eta > 0$, there exists a finite collection of open balls with radius η . These balls will have their centers located within \mathcal{A} , and their union will cover the entire \mathcal{A} . First, we construct a net for \mathcal{A} . Note that \mathcal{D}^0 represents a planar circular domain with circular holes. Let $\bar{\Omega}$ be the unit rectangle and $\mathcal{D}^0 \subset \bar{\Omega}$. Now, we construct a regular grid on $\bar{\Omega}$ with edge length $\frac{1}{m}$. After reindexing, we denote the grid points as $\{p_i\}_{i \in I}$. Next, we define a smooth tent function on each grid point as follows: $\mathcal{T}_{m,n,L} = \{\frac{k}{n} \tilde{\delta}_{p_i}^L|_{\mathcal{D}^0}\}_{i \in I, 0 \leq k \leq n}$. In this context, $\tilde{\delta}_{p_i}^L$ represents the smooth approximation of the tent function δ_{p_i} subject to the conditions $\|D\tilde{\delta}_{p_i}^L - m\mathbf{Id}\|_\infty + \|\tilde{\delta}_{p_i}^L - \delta_{p_i}\|_\infty \leq \frac{1}{L}$ and $\|\tilde{\delta}_{p_i}^L\|_\infty \leq C_1$. Now we consider: $\mathcal{B}_{m,n,L} = \{\tilde{\nu} \in \mathcal{A} : \tilde{\nu} = \sum_i T_i, T_i \in \mathcal{T}_{m,n,L}\}$. It is important to note that due to the finite number of grid points, the functions in $\mathcal{B}_{m,n,L}$ are also finite in number. Finally, we construct a finite union cover of \mathcal{A} . For any $\eta > 0$ and any $\nu \in \mathcal{A}$, we can choose some sufficiently large numbers m, n, L such that there exists a sum of tent functions $\bar{\nu} = \sum_i \frac{k}{n} \delta_{p_i}|_{\mathcal{D}^0}$ and $\tilde{\nu} \in \mathcal{B}_{m,n,L}$ satisfying $\|\nu - \bar{\nu}\|_\infty < \frac{\eta}{4}$, $\|\bar{\nu} - \tilde{\nu}\|_\infty < \frac{\eta}{4}$, $\|D\nu - D\bar{\nu}\|_\infty < \frac{\eta}{4}$, and $\|D\bar{\nu} - D\tilde{\nu}\|_\infty < \frac{\eta}{4}$. Then we have $\|\nu - \tilde{\nu}\|_s \leq \|\nu - \bar{\nu}\|_s + \|\bar{\nu} - \tilde{\nu}\|_s < \eta$. Hence, $\mathcal{A} \subset \cup_{\tilde{\nu} \in \mathcal{B}_{m,n,L}} B_\eta(\tilde{\nu})$. Here, $B_\eta(\tilde{\nu})$ is the open neighborhood ball of $\tilde{\nu}$ with radius η . Consequently, \mathcal{A} is a totally bounded set.

Since \mathcal{A} is complete and totally bounded, \mathcal{A} is compact. Also, note that $\nabla\rho$ is an L^2 function and E_{DEQ} is continuous in \mathcal{A} . Hence, E_{DEQ} has a minimizer in \mathcal{A} . ■

With the theoretical existence of the minimizer of (4.2) proved, in section 5.1 we develop an iterative algorithm to solve for the optimal planar circular domain \mathcal{D} and the optimal DEQ map $f : \mathcal{S} \rightarrow \mathcal{D}$. We remark that while the existence of the global minimizer is proved theoretically, our iterative algorithm is based on gradient descent and may only converge to local minima.

4.2. Bijective landmark-matching DEQ (LDEQ) map. In many practical situations, it is desirable to compute flattening maps with certain feature points on a surface mapped to

some prescribed positions. We can achieve this by incorporating interior landmark-matching constraints into the bijective density-equalizing quasiconformal model.

Let $\{p_i\}_{i=1}^m$ be a set of feature landmarks defined at the interior of \mathcal{S} and $\{q_i\}_{i=1}^m$ be the corresponding target positions defined on the plane. Our objective is to find an optimal planar domain \mathcal{D} and a *bijective LDEQ* map $f: \mathcal{S} \rightarrow \mathcal{D}$ satisfying $f(p_i) = q_i$ for all $i = 1, \dots, m$.

To achieve this, we again consider $f = \tilde{f} \circ g_0$, where $g_0: \mathcal{S} \rightarrow \mathcal{D}^0$ is a conformal flattening map of \mathcal{S} onto a planar circular domain \mathcal{D}^0 and $\tilde{f}: \mathcal{D}^0 \rightarrow \mathcal{D}$ is a planar quasiconformal map. We propose a variational approach to obtain \tilde{f} by minimizing the following energy functional E_{LDEQ} :

$$(4.7) \quad \tilde{f} = \arg \min_{g: \mathcal{D}^0 \rightarrow \mathcal{D}} E_{\text{LDEQ}}(g) := \arg \min_{g: \mathcal{D}^0 \rightarrow \mathcal{D}} \left\{ \int_{\mathcal{D}^0} |\nabla \rho(g)|^2 + \alpha \int_{\mathcal{D}^0} |\mu(g)|^2 + \beta \int_{\mathcal{D}^0} |\nabla \mu(g)|^2 \right\}$$

subject to

$$(4.8) \quad (\tilde{f} \circ g_0)(p_i) = q_i \quad \text{for } 1 \leq i \leq m,$$

$$(4.9) \quad (\tilde{f} \circ g_0)(\gamma_i) = \partial B_{r_i}(c_i) \quad \text{for } 0 \leq i \leq k,$$

$$(4.10) \quad \|\mu(\tilde{f})\|_\infty < 1.$$

Again, here $\mu(\tilde{f})$ and $\mu(g)$ denote the Beltrami coefficient of \tilde{f} and g , respectively, $\partial B_{r_0}(c_0)$ denotes the outer boundary, and α and β are two user-defined small nonnegative weighting parameters for controlling the quasiconformality.

Analogously to the previous model, the existence of the solution to the optimization problem (4.7) can be proved as follows.

Proposition 4.2 (existence of the minimizer of E_{LDEQ}). *Let*

$$(4.11) \quad \mathcal{A} = \{ \nu \in C^1(\mathcal{D}^0) : \|D\nu\|_\infty \leq C_1; \|\nu\|_\infty \leq 1 - \epsilon; \nu \text{ is the Beltrami coefficient of a quasiconformal map } g^\nu \text{ with } g^\nu(z_i) = w_i \text{ for all } 1 \leq i \leq m \}$$

for some $C_1 > 0$ and small $\epsilon > 0$, where \mathcal{D}^0 is a circular domain, $\{z_i\}_{i=1}^m$ is a set of points on \mathcal{D}^0 , and $\{w_i\}_{i=1}^m$ is a set of points on \mathcal{D} . Then E_{LDEQ} has a minimizer in \mathcal{A} . In fact, \mathcal{A} is compact.

Proof. Note that $\mathcal{A} \neq \emptyset$. Now, we prove that \mathcal{A} is complete. Define $\|\mu\|_s = \|\mu\|_\infty + \|D\mu\|_\infty$. Let $\{\nu_n\}_{n=1}^\infty$ be a Cauchy sequence in \mathcal{A} under the norm $\|\cdot\|_s$. Then, $\{D\nu_n\}_{n=1}^\infty$ is also a Cauchy sequence with respect to the $\|\cdot\|_\infty$ norm. Since $\mathcal{A} \subset W^{1,\infty}(\mathcal{D}^0)$ and $W^{1,\infty}$ is complete, $D\nu_n \rightarrow h$ uniformly for some $h \in W^{1,\infty}(\mathcal{D}^0)$. Since $D\nu_n$ is continuous, h is also continuous. Moreover, ν_n is convergent in $W^{1,\infty}(\mathcal{D}^0)$ as well. Hence, $\nu_n \rightarrow \nu$ and $D\nu_n \rightarrow D\nu$ uniformly for some $\nu \in C^1(\mathcal{D}^0)$. In addition, $\|\nu_n\|_\infty \leq 1 - \epsilon$ for all n implies that $\|\nu\|_\infty \leq 1 - \epsilon$. Also, $\|D\nu_n\|_\infty \leq C_1$ implies that $\|D\nu\|_\infty \leq C_1$. Since $\nu_n \rightarrow \nu$ uniformly, $g^{\nu_n} \rightarrow g^\nu$ locally uniformly. More specifically, we have $g^\nu(z_i) = w_i$ for $i = 1, 2, \dots, m$. Therefore, $\nu \in \mathcal{A}$. Consequently, \mathcal{A} is Cauchy complete.

Next, we prove that \mathcal{A} is totally bounded. It is easy to see that \mathcal{A} is a subset of $\tilde{\mathcal{A}} = \{\nu \in C^1(\mathcal{D}^0) : \|D\nu\|_\infty \leq C_1; \|\nu\|_\infty \leq 1 - \epsilon; \nu \text{ is the Beltrami coefficient of a quasiconformal map}\}$, and we have already proved that $\tilde{\mathcal{A}}$ is totally bounded in the proof of Proposition 4.1. Since any subset of a totally bounded set is totally bounded, \mathcal{A} is totally bounded.

As a result, \mathcal{A} is compact. Therefore, E_{LDEQ} has a minimizer in \mathcal{A} . ■

In section 5.2, we develop an iterative algorithm for finding the optimal domain \mathcal{D} and the LDEQ map $f : \mathcal{S} \rightarrow \mathcal{D}$. Analogously to the previous case, here we remark that our gradient-based algorithm may only converge to local minima.

5. Proposed algorithms. Consider a given multiply connected open surface \mathcal{S} in \mathbb{R}^3 with $\partial\mathcal{S} = \gamma_0 - \gamma_1 - \dots - \gamma_k$ discretized as a triangle mesh $(\mathcal{V}, \mathcal{E}, \mathcal{F})$, where \mathcal{V} is the set of vertices, \mathcal{E} is the set of edges, and \mathcal{F} is the set of triangular faces. Suppose a population is assigned to each triangle face of the mesh. We first apply an initial conformal flattening map to map \mathcal{S} onto a planar circular domain. The face density can then be defined as $\rho^{\mathcal{F}} = \frac{\text{population}}{\text{area of the triangle}}$ on the circular domain. One can also define the point density $\rho^{\mathcal{V}}$ by considering the 1-ring face neighborhood of each vertex (see Appendix A for the detailed formulation).

Below, we develop two algorithms based on the models (4.2) and (4.7) for computing bijective DEQ maps and bijective LDEQ maps.

5.1. Bijective DEQ map. Recall that our objective is to find a circular domain $\mathcal{D} \in \mathbb{R}^2$ and a diffeomorphism $f : \mathcal{S} \rightarrow \mathcal{D}$ that satisfies two criteria: first, f should be a bijective density-equalizing map between \mathcal{S} and \mathcal{D} and, second, it should minimize the local geometric distortion. Equivalently, we want to find an optimal Beltrami coefficient such that the diffeomorphism associated with it satisfies the two criteria above. Due to the presence of higher-order derivative terms, it is challenging to directly solve the optimization problem (4.2) to get the mapping. Instead, we consider using the Beltrami coefficient as the variable of the optimization problem. The variational problem (4.2) for \tilde{f} can be transformed into the problem of computing the optimal Beltrami coefficient ν^* that minimizes the following energy functional E_{DEQ} ,

$$(5.1) \quad E_{\text{DEQ}}(\nu) := \int_{\mathcal{D}^0} |\nabla \rho(g)|^2 + \alpha \int_{\mathcal{D}^0} |\nu|^2 + \beta \int_{\mathcal{D}^0} |\nabla \nu|^2$$

subject to

$$(5.2) \quad \|\nu\|_\infty < 1,$$

$$(5.3) \quad \nu = \mu(g),$$

where \mathcal{D}^0 is the initial flattened circular domain, g is a quasiconformal map, $\mu(g)$ is the Beltrami coefficient of g , and ρ is the density function given by the population defined on \mathcal{S} .

Below, we develop an algorithm for finding the optimal map based on the above formulation. Our proposed algorithm involves three major components:

- *Initial conformal flattening map:* We first conformally flatten \mathcal{S} onto a planar circular domain \mathcal{D}^0 with k inner circular holes (section 5.1.1).

- *Geometry modification*: We compute a modification mapping and obtain a new circular domain with the size and position of the inner boundaries adjusted (section 5.1.2).
- *Beltrami density-equalizing descent*: We search for a density-equalizing descent direction for the Beltrami coefficient on the new circular domain and reconstruct an updated mapping using it (section 5.1.3).

5.1.1. Initial conformal flattening map. To begin, we compute an initial flattening map $g_0 : \mathcal{S} \rightarrow \mathcal{D}^0$, where $\mathcal{D}^0 = \mathbb{D} \setminus \cup_{i=1}^k B_{r_i}(c_i)$ is a planar circular domain with k inner circular holes as described in (4.1), where r_i and c_i are the radius and center of the i th inner circular hole of \mathcal{D}^0 , and $g_0(\gamma_i) = \partial B_{r_i}(c_i)$ for all $i = 1, \dots, k$.

In particular, here we look for a conformal flattening map as the initialization. Recall that our goal is to achieve a density-equalizing map with reduced quasiconformal distortion. By starting with a conformal parameterization, we can ensure that the initial quasiconformal distortion is low and the triangulations are regular. This facilitates the subsequent computation of the DEQ map.

To achieve this, various existing parameterization methods for multiply connected open surfaces can be applied. For instance, Koebe's iteration method [29] can be used.

5.1.2. Geometry modification of the circular domain. After getting the initial flattening map, we can then look for a density-equalizing map on the plane. However, it is worth noting that while all inner holes of the flattened domain \mathcal{D}^0 are perfectly circular, their size and position may not be optimal for the density diffusion process. To improve the accuracy and efficiency of the density-equalizing map, we consider modifying the geometry of \mathcal{D}^0 by changing the centers and radii of its inner circular holes. More specifically, our geometry modification approach involves two processes: *domain alteration* and *domain reconstruction* (see Figure 4). After the modification, we get a new circular domain \mathcal{D}^1 with the position and size of the circular holes updated.

We first consider the domain alteration, which is performed on the boundaries of \mathcal{D}^0 . In this step, our goal is to modify the boundaries $\partial\mathcal{D}^0$ and get a new set of circular holes $\partial\mathcal{D}^1$ such that they are more suitable for the subsequent density-equalizing process. Therefore,

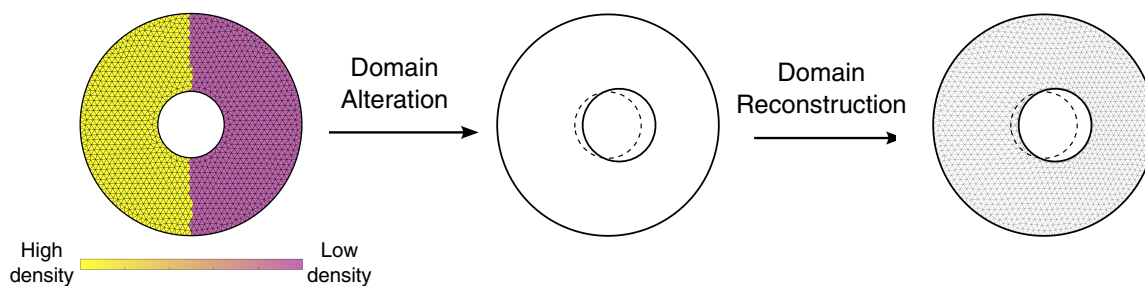


Figure 4. An illustration of the geometry modification (GM) algorithm. Given a planar circular domain, we first change the size and position of the inner circular holes based on the density gradient. Then, we reconstruct a smooth quasiconformal map based on the updated boundaries. Here, the dashed curve represents the previous inner boundary, and the solid curve represents its updated position.

the holes $B_{r_i}(c_i)$ ($i = 1, 2, \dots, k$) should deform naturally based on the density gradient in the domain while remaining to be circular. In other words, the deformation of each hole must be a combination of translation, scaling, and rotation, where the translation of the hole can be considered as a translation of the hole center, the scaling can be considered as a movement of all points along the radial direction, and the rotation can be considered as a movement of all points along the tangential direction. This motivates us to consider minimizing the following domain modification functional:

$$(5.4) \quad E_{\text{bdy}}(\nu, \mathbf{c}, \mathbf{r}, \mathbf{T}) := \int_{\partial\mathcal{D}^0} |\nabla\rho(\tilde{g}, \mathbf{c}, \mathbf{r}, \mathbf{T})|^2 + \alpha \int_{\partial\mathcal{D}^0} |\nu(\mathbf{c}, \mathbf{r}, \mathbf{T})|^2 + \beta \int_{\partial\mathcal{D}^0} |\nabla\nu(\mathbf{c}, \mathbf{r}, \mathbf{T})|^2$$

subject to $\|\nu\|_\infty < 1$ and $\nu = \mu(\tilde{g})$. Here, $\tilde{g}: \mathcal{D}^0 \rightarrow \mathcal{D}^1$ is the desired modification map, \mathbf{c} is the set of centers of all inner boundaries of \mathcal{D}^0 , \mathbf{r} is the set of radii of all inner boundaries, and \mathbf{T} is the set of tangent vectors on all inner boundaries. More specifically, $\mu(\tilde{g})$ is the Beltrami coefficient of \tilde{g} and $\rho(\tilde{g}, \mathbf{c}, \mathbf{r}, \mathbf{T})$ is the density function. For simplicity, we denote

$$(5.5) \quad E_{\text{bdy}}^1 = \int_{\partial\mathcal{D}^0} |\nabla\rho(\tilde{g}, \mathbf{c}, \mathbf{r}, \mathbf{T})|^2,$$

$$(5.6) \quad E_{\text{bdy}}^2 = \int_{\partial\mathcal{D}^0} |\nu(\mathbf{c}, \mathbf{r}, \mathbf{T})|^2,$$

$$(5.7) \quad E_{\text{bdy}}^3 = \int_{\partial\mathcal{D}^0} |\nabla\nu(\mathbf{c}, \mathbf{r}, \mathbf{T})|^2.$$

Now, we first compute the alteration of E_{bdy}^1 . By the diffusion equation and Fick's law, the diffusion-based velocity is given by:

$$(5.8) \quad \mathbf{v} = -\frac{\nabla\rho^\nu}{\rho^\nu},$$

where ρ^ν is the vertex density (see Appendix A for the discretization and computation).

Instead of using the gradient, we utilize the velocity \mathbf{v} , which points towards the global minimum, as the direction of motion. Therefore, when deforming the inner boundaries, we aim to find suitable directions for the centers and boundary points that are in proximity to the velocity vector \mathbf{v} . This ensures that the points move towards the global minimum. The minimization of the energy E_{bdy}^1 can be interpreted as the minimization of the following energy:

$$(5.9) \quad \tilde{E}_{\text{bdy}}^1(\mathbf{c}, \mathbf{r}, \mathbf{T}) = \sum_{i=1}^k \int_{\partial B_{r_i}(c_i)} |v_i - \mathbf{v}|^2 = \sum_{i=1}^k \sum_{j=1}^{n_i} |v_{i,j} - \mathbf{v}_{i,j}|^2$$

subject to the constraint that all boundaries $\{\partial B_{r_i}(c_i)\}_{i=1}^k$ remain circular after the deformation. Here, n_i represents the number of vertices on the boundary $\partial B_{r_i}(c_i)$, and $v_{i,j}$ and $\mathbf{v}_{i,j}$ denote the velocity v and \mathbf{v} at the j th vertex on the i th circular hole, respectively.

Now, for any point on an inner circular hole, it can be expressed as $c + re^{i\theta}$ for some c, r, θ . Similarly, its updated position under a deformation can be expressed as $(c + \Delta c) +$

$(r + \Delta r)e^{i(\theta + \Delta\theta)}$ for some $\Delta c, \Delta r, \Delta\theta$. Therefore, the velocity v_i at any boundary point on the i th inner circular hole can be decomposed into three distinct components: a translation of the hole center, a scaling of the hole along the radial direction of the hole, and a rotation along the tangential direction of the hole. In other words, we have

$$(5.10) \quad v_i = v_i^c + v_i^r + v_i^T,$$

where v_i^c is the translational velocity associated with the center, v_i^r is the radial velocity, and v_i^T is the tangential velocity. We can then rewrite (5.9) as follows,

$$(5.11) \quad \begin{aligned} \tilde{E}_{\text{bdy}}^1(\mathbf{c}, \mathbf{r}, \mathbf{T}) &= \sum_{i=1}^k \left(\int_{\partial B_{r_i}(c_i)} |v_i^c - \mathbf{v}_i^c|^2 + \int_{\partial B_{r_i}(c_i)} |v_i^r - \mathbf{v}_i^r|^2 + \int_{\partial B_{r_i}(c_i)} |v_i^T - \mathbf{v}_i^T|^2 \right) \\ &= \sum_{i=1}^k \sum_{j=1}^{n_i} (|v_{i,j}^c - \mathbf{v}_{i,j}^c|^2 + |v_{i,j}^r - \mathbf{v}_{i,j}^r|^2 + |v_{i,j}^T - \mathbf{v}_{i,j}^T|^2), \end{aligned}$$

where $v_{i,j}^c, v_{i,j}^r, v_{i,j}^T$ and $\mathbf{v}_{i,j}^c, \mathbf{v}_{i,j}^r, \mathbf{v}_{i,j}^T$ denote the translational, radial, and tangential velocity components of v and \mathbf{v} at the j th vertex on the i th inner circular hole, respectively.

To ensure that the inner boundary $\partial B_{r_i}(c_i)$ is still a circular hole after translation, the translation velocity $v_{i,j}^c$ should be the same for all boundary points and, hence, we can simply denote it as v_i^c without ambiguity. The radial velocity $v_{i,j}^r$ must be equal to $p_i^r \vec{r}_{i,j}$ for all boundary points, where p_i^r is a real number and $\vec{r}_{i,j}$ is the outer unit normal vector at each point. Similarly, the tangent velocity $v_{i,j}^T$ can be denoted as $p_i^T \vec{T}_{i,j}$, where p_i^T is a real number and $\vec{T}_{i,j}$ is the unit tangent vector at each point. Hence, the optimization problem (5.11) can be rewritten as:

$$(5.12) \quad \tilde{E}_{\text{bdy}}^1(\mathbf{c}, \mathbf{p}^r, \mathbf{p}^T) = \sum_{i=1}^k \sum_{j=1}^{n_i} \left(|v_i^c - \mathbf{v}_{i,j}^c|^2 + |p_i^r \vec{r}_{i,j} - \mathbf{v}_{i,j}^r|^2 + |p_i^T \vec{T}_{i,j} - \mathbf{v}_{i,j}^T|^2 \right),$$

where $\mathbf{p}^r = (p_1^r, \dots, p_k^r)$ and $\mathbf{p}^T = (p_1^T, \dots, p_k^T)$. Now, for the i th inner circular hole, we denote

$$(5.13) \quad \tilde{E}_{\text{bdy}_i}^1(c_i, p_i^r, p_i^T) = \sum_{j=1}^{n_i} |v_i^c - \mathbf{v}_{i,j}^c|^2 + \sum_{j=1}^{n_i} |p_i^r \vec{r}_{i,j} - \mathbf{v}_{i,j}^r|^2 + \sum_{j=1}^{n_i} |p_i^T \vec{T}_{i,j} - \mathbf{v}_{i,j}^T|^2.$$

Using the principle of minimum energy and (5.8), the minimum values of the three parts in $\tilde{E}_{\text{bdy}_i}^1(c_i, p_i^r, p_i^T)$ can be obtained by

$$(5.14) \quad \tilde{v}_i^c = \frac{1}{n_i} \sum_{j=1}^{n_i} (\mathbf{v}_{i,j}^c) = \frac{1}{n_i} \sum_{j=1}^{n_i} \left(-\frac{\nabla \rho^{\mathcal{V}}(w_{i,j})}{\rho^{\mathcal{V}}(w_{i,j})} \right)_{i,j}^c,$$

$$(5.15) \quad \tilde{p}_i^r = \frac{1}{n_i} \sum_{j=1}^{n_i} (\mathbf{v}_{i,j}^r) \cdot \vec{r}_{i,j} = \frac{1}{n_i} \sum_{j=1}^{n_i} \left(-\frac{\nabla \rho^{\mathcal{V}}(w_{i,j})}{\rho^{\mathcal{V}}(w_{i,j})} \right)_{i,j}^r \cdot \vec{r}_{i,j},$$

$$(5.16) \quad \tilde{p}_i^T = \frac{1}{n_i} \sum_{j=1}^{n_i} (\mathbf{v}_{i,j}^T) \cdot \vec{T}_{i,j} = \frac{1}{n_i} \sum_{j=1}^{n_i} \left(-\frac{\nabla \rho^{\mathcal{V}}(w_{i,j})}{\rho^{\mathcal{V}}(w_{i,j})} \right)_{i,j}^T \cdot \vec{T}_{i,j},$$

where $\rho^{\mathcal{V}}(w_{i,j})$ is the vertex density at the boundary point $w_{i,j}$. The descent direction associated with the energy E_{bdy}^1 for all inner boundaries is given by

$$(5.17) \quad dw^1 = \sum_{i=1}^k \sum_{j=1}^{n_i} (\tilde{v}_{c_i} + \tilde{p}_i^T \vec{r}_{i,j} + \tilde{p}_i^T \vec{T}_{i,j}).$$

Until now, we have only dealt with the optimization problem (5.9). However, as the value of dw^1 is solely linked to the diffusion process, the bijectivity and geometry structure are not preserved in general. Therefore, our next step is to consider the other two energy terms E_{bdy}^2 and E_{bdy}^3 on the inner boundaries.

First, we express the energy E_{bdy}^2 as follows,

$$(5.18) \quad E_{\text{bdy}}^2(\mathbf{c}, \mathbf{r}, \mathbf{T}) = \sum_{i=1}^k \sum_{\mathcal{T}_l \in \mathcal{N}^{\mathcal{F}}(\gamma_i)} \text{Area}(\mathcal{T}_l) |\nu_l|^2,$$

where $\mathcal{N}^{\mathcal{F}}(\gamma_i)$ is the 1-ring face neighborhood of the i th inner boundary, and ν_l is the boundary Beltrami coefficient associated with \mathcal{T}_l . Specifically, ν_l can be expressed in terms of the center c_i , radius vector $r_{i,j}$, and tangent vector $T_{i,j}$ of some vertex on the boundary. The detailed computation of $\nu_l = \nu_l(c_i, r_{i,j}, T_{i,j})$ is provided in Appendix B.

The descent directions associated with E_{bdy}^2 for the i th inner hole can be obtained by

$$(5.19) \quad dw_{i,1}^2 = \frac{1}{n_i} \sum_{j=1}^{n_i} \sum_{\mathcal{T}_l \in \mathcal{N}_{i,j}^{\mathcal{F}}} 2 \text{Area}(\mathcal{T}_l) \left(\nu_l(c_i, r_{i,j}, T_{i,j}) \frac{\partial \nu_l}{\partial r_{i,j}} \right),$$

$$(5.20) \quad dw_{i,2}^2 = \frac{1}{n_i} \sum_{j=1}^{n_i} \sum_{\mathcal{T}_l \in \mathcal{N}_{i,j}^{\mathcal{F}}} 2 \text{Area}(\mathcal{T}_l) \left(\nu_l(c_i, r_{i,j}, T_{i,j}) \frac{\partial \nu_l}{\partial T_{i,j}} \right),$$

$$(5.21) \quad dw_{i,3}^2 = \frac{1}{n_i} \sum_{j=1}^{n_i} \sum_{\mathcal{T}_l \in \mathcal{N}_{i,j}^{\mathcal{F}}} 2 \text{Area}(\mathcal{T}_l) \left(\nu_l(c_i, r_{i,j}, T_{i,j}) \frac{\partial \nu_l}{\partial c_i} \right).$$

Here, $\mathcal{N}_{i,j}^{\mathcal{F}}$ is the set of 1-ring neighborhood triangle elements that contain the j th vertex of the i th inner hole, the partial derivatives $\frac{\partial \nu_l}{\partial r_{i,j}}$ and $\frac{\partial \nu_l}{\partial T_{i,j}}$ correspond to the descent direction along the radial and tangential directions, respectively, and $\frac{\partial \nu_l}{\partial c_i}$ represents the descent direction associated with the center of the circular hole.

Combining the above formulas for $dw_{i,1}^2, dw_{i,2}^2, dw_{i,3}^2$ for $i = 1, \dots, k$, the final descent direction associated with the energy E_{bdy}^2 for all inner boundaries is given by

$$(5.22) \quad dw^2 = \sum_{i=1}^k (dw_{i,1}^2 + dw_{i,2}^2 + dw_{i,3}^2).$$

Then, we express the smoothness term E_{bdy}^3 as follows,

$$(5.23) \quad E_{\text{bdy}}^3(\mathbf{c}, \mathbf{r}, \mathbf{T}) = \sum_{i=1}^k \sum_{\mathcal{T}_i \in \mathcal{N}^{\mathcal{F}}(\gamma_i)} \left(\frac{\cot \theta_l^0}{2} (\nu(v_l^1) - \nu(v_l^2))^2 + \frac{\cot \theta_l^1}{2} (\nu(v_l^2) - \nu(v_l^0))^2 + \frac{\cot \theta_l^2}{2} (\nu(v_l^0) - \nu(v_l^1))^2 \right),$$

where $\mathcal{T}_l = [v_l^0, v_l^1, v_l^2]$ is a triangle element on the i th inner boundary, $\theta_l^0, \theta_l^1, \theta_l^2$ are the angles in \mathcal{T}_l , and $\nu(v_l^0), \nu(v_l^1), \nu(v_l^2)$ are the vertex Beltrami coefficients at v_l^0, v_l^1, v_l^2 , which can be obtained by taking the average of the Beltrami coefficients at their 1-ring face neighborhood. Below, we denote $\nu(v_l^0), \nu(v_l^1), \nu(v_l^2)$ as $\nu_l^0, \nu_l^1, \nu_l^2$ for simplicity.

The descent direction associated with E_{bdy}^3 for the i th inner hole can be given by the three following terms:

$$(5.24) \quad dw_{i,1}^3 = \frac{1}{n_i} \sum_{j=1}^{n_i} \sum_{\mathcal{T}_i \in \mathcal{N}_{i,j}^{\mathcal{F}}} \left(\cot \theta_l^0 (\nu_l^1 - \nu_l^2) \left(\frac{\partial \nu_l^1}{\partial r_{i,j}} - \frac{\partial \nu_l^2}{\partial r_{i,j}} \right) + \cot \theta_l^1 (\nu_l^2 - \nu_l^0) \left(\frac{\partial \nu_l^2}{\partial r_{i,j}} - \frac{\partial \nu_l^0}{\partial r_{i,j}} \right) + \cot \theta_l^2 (\nu_l^0 - \nu_l^1) \left(\frac{\partial \nu_l^0}{\partial r_{i,j}} - \frac{\partial \nu_l^1}{\partial r_{i,j}} \right) \right),$$

$$(5.25) \quad dw_{i,2}^3 = \frac{1}{n_i} \sum_{j=1}^{n_i} \sum_{\mathcal{T}_i \in \mathcal{N}_{i,j}^{\mathcal{F}}} \left(\cot \theta_l^0 (\nu_l^1 - \nu_l^2) \left(\frac{\partial \nu_l^1}{\partial T_{i,j}} - \frac{\partial \nu_l^2}{\partial T_{i,j}} \right) + \cot \theta_l^1 (\nu_l^2 - \nu_l^0) \left(\frac{\partial \nu_l^2}{\partial T_{i,j}} - \frac{\partial \nu_l^0}{\partial T_{i,j}} \right) + \cot \theta_l^2 (\nu_l^0 - \nu_l^1) \left(\frac{\partial \nu_l^0}{\partial T_{i,j}} - \frac{\partial \nu_l^1}{\partial T_{i,j}} \right) \right),$$

$$(5.26) \quad dw_{i,3}^3 = \frac{1}{n_i} \sum_{j=1}^{n_i} \sum_{\mathcal{T}_i \in \mathcal{N}_{i,j}^{\mathcal{F}}} \left(\cot \theta_l^0 (\nu_l^1 - \nu_l^2) \left(\frac{\partial \nu_l^1}{\partial c_i} - \frac{\partial \nu_l^2}{\partial c_i} \right) + \cot \theta_l^1 (\nu_l^2 - \nu_l^0) \left(\frac{\partial \nu_l^2}{\partial c_i} - \frac{\partial \nu_l^0}{\partial c_i} \right) + \cot \theta_l^2 (\nu_l^0 - \nu_l^1) \left(\frac{\partial \nu_l^0}{\partial c_i} - \frac{\partial \nu_l^1}{\partial c_i} \right) \right).$$

Combining the above formulas for $dw_{i,1}^3, dw_{i,2}^3, dw_{i,3}^3$ for $i = 1, \dots, k$, the descent direction associated with the energy E_{bdy}^3 for all inner boundaries is given by

$$(5.27) \quad dw^3 = \sum_{i=1}^k (dw_{i,1}^3 + dw_{i,2}^3 + dw_{i,3}^3).$$

Finally, combining (5.17), (5.22), and (5.27), the overall descent direction for the domain alteration is given by

$$(5.28) \quad dw = dw^1 + \alpha dw^2 + \beta dw^3.$$

We can then obtain the updated boundaries $\partial\mathcal{D}^1$ by

$$(5.29) \quad \partial\mathcal{D}^1 = \partial\mathcal{D}^0 + \delta t dw,$$

where δt is a prescribed time step size. We will later introduce an iterative scheme involving this updating step so that the boundaries become optimal over time.

After obtaining $\partial\mathcal{D}^1$, we move on to the domain reconstruction step. Here, our goal is to obtain a quasiconformal map \tilde{g} such that $\mu(\tilde{g})$ is close to ν and $\tilde{g}(\partial\mathcal{D}^0) = \partial\mathcal{D}^1$. Also, it is desirable that the quasiconformal distortion of \tilde{g} is low so that the resulting mesh is sufficiently regular to serve as a good starting point for the subsequent density-equalization step. To achieve this, we compute the descent direction of the following energy involving the Beltrami coefficient:

$$(5.30) \quad E_R(\nu) = \alpha \int_{\mathcal{D}^0} |\nu|^2 + \beta \int_{\mathcal{D}^0} |\nabla\nu|^2.$$

Here, the first term $\alpha \int_{\mathcal{D}^0} |\nu|^2$ is utilized to ensure that the quasiconformal distortion of the resulting map is small, while the second term $\beta \int_{\mathcal{D}^0} |\nabla\nu|^2$ is used to enhance the smoothness of the resulting map. Below, we denote the two integrals as $E_R^1(\nu) = \int_{\mathcal{D}^0} |\nu|^2$ and $E_R^2(\nu) = \int_{\mathcal{D}^0} |\nabla\nu|^2$ for simplicity.

For $E_R^1(\nu) = \sum_{i=1}^{|\mathcal{F}|} \text{Area}(\mathcal{T}_i) |\nu(\mathcal{T}_i)|^2$, where \mathcal{T}_i is the i th triangle element, the descent direction $d\nu^1$ can be computed by

$$(5.31) \quad \begin{aligned} \frac{d}{dt} \Big|_{t=0} E_R^1(\nu + \delta t \omega) &= \sum_{i=1}^{|\mathcal{F}|} \text{Area}(\mathcal{T}_i) \frac{d}{dt} \Big|_{t=0} |\nu(\mathcal{T}_i) + \delta t \omega_i|^2 \\ &= 2 \sum_{i=1}^{|\mathcal{F}|} \text{Area}(\mathcal{T}_i) \nu(\mathcal{T}_i) \omega_i. \end{aligned}$$

The derivative is negative when $\omega_i = -\nu(\mathcal{T}_i)$ for $i = 1, 2, \dots, |\mathcal{F}|$. Hence, the descent direction of the Beltrami coefficient is $d\nu^1 = -\nu$.

Using the same method, E_R^2 can be discretized on the triangle mesh

$$(5.32) \quad \begin{aligned} E_R^2(\nu) &= \sum_{i=1}^{|\mathcal{F}|} \int_{\mathcal{T}_i} |\nabla\nu|^2 \\ &= \sum_{i=1}^{|\mathcal{F}|} \left(\frac{\cot \theta_{i,0}}{2} (\nu(v_{i,1}) - \nu(v_{i,2}))^2 + \frac{\cot \theta_{i,1}}{2} (\nu(v_{i,2}) - \nu(v_{i,0}))^2 \right. \\ &\quad \left. + \frac{\cot \theta_{i,2}}{2} (\nu(v_{i,0}) - \nu(v_{i,1}))^2 \right), \end{aligned}$$

Algorithm 5.1. GM.

Input: A circular domain \mathcal{D}^0 with density ρ and initial Beltrami coefficient ν .

Output: A modification map $\tilde{g}: \mathcal{D}^0 \rightarrow \mathcal{D}^1$, where \mathcal{D}^1 is a new circular domain.

- 1: Compute the velocity $\mathbf{v} = -\frac{\nabla \rho^\nu}{\rho^\nu}$;
- 2: Obtain the new boundaries $\partial \mathcal{D}^1$ based on \mathbf{v} using (5.17)–(5.29);
- 3: Compute $\tilde{\nu}$ using (5.31)–(5.34);
- 4: Use the LBS method to obtain the modification map \tilde{g} from $\tilde{\nu}$;
- 5: Obtain the new circular domain $\mathcal{D}^1 = \tilde{g}(\mathcal{D}^0)$.

where $\mathcal{T}_i = [v_{i,0}, v_{i,1}, v_{i,2}]$ is the i th triangle element and $\theta_{i,0}, \theta_{i,1}, \theta_{i,2}$ are the three angles in it. Then we can obtain the increment $d\nu^2 = \Delta\nu$ as well.

Therefore, the descent direction for the problem (5.30) is

$$(5.33) \quad d\nu = \alpha d\nu^1 + \beta d\nu^2.$$

We can then obtain the updated Beltrami coefficient by:

$$(5.34) \quad \tilde{\nu} = \nu + \delta t d\nu,$$

where δt is the time step size. Using the LBS algorithm [36], we can compute the modification map \tilde{g} , and the new circular domain \mathcal{D}^1 is given by $\mathcal{D}^1 = \tilde{g}(\mathcal{D}^0)$. The proposed GM method is summarized in Algorithm 5.1.

5.1.3. Beltrami density-equalizing descent. From the above GM step, we obtain a modified circular domain. However, the density may not be well-equalized in the modified domain. Therefore, our next step is to use the modified domain as the initial map and search for a suitable descent direction for density equalization by solving the minimization problem (5.1).

We first discuss the energy $E_{\text{DEQ}}^1 = \int |\nabla \rho|^\nu$ in (5.1), where $\rho = \rho_0$ is the density on $\tilde{g}_1 \circ g_0(\mathcal{S})$. Here, we aim to express the descent direction for this energy as a descent direction of the Beltrami coefficient. By diffusion theory and Fick's law, the velocity induced by the density gradient is

$$(5.35) \quad dg = -\frac{\nabla \rho^\nu}{\rho^\nu},$$

where the computation of $\nabla \rho^\nu$ is described in Appendix A. To preserve the circular boundary shape obtained by Algorithm 5.1, we enforce the Neumann boundary condition $\nabla \rho \cdot \mathbf{n} = 0$ as in [11] here, where \mathbf{n} is the unit normal vector at each boundary. It is noteworthy that while the shape of each boundary is fixed, the boundary vertices are still allowed to slide along the boundary based on the density gradient to achieve a better density-equalizing effect.

It is easy to see that as \tilde{g}_1 is perturbed, its associated Beltrami coefficient ν_0 will be adjusted by $d\nu^1$. We can compute the adjustment via the Beltrami equation as follows,

$$(5.36) \quad \frac{\partial(\tilde{g}_1 + dg)}{\partial \bar{z}} = (\nu_0 + d\nu^1) \frac{\partial(\tilde{g}_1 + dg)}{\partial z},$$

which implies

$$(5.37) \quad \frac{\partial \tilde{g}_1}{\partial \bar{z}} + \frac{\partial dg}{\partial \bar{z}} = \nu_0 \frac{\partial \tilde{g}_1}{\partial z} + \nu_0 \frac{\partial dg}{\partial z} + d\nu^1 \frac{\partial \tilde{g}_1}{\partial z} + d\nu^1 \frac{\partial dg}{\partial z}.$$

Note that $\frac{\partial \tilde{g}_1}{\partial \bar{z}} = \nu_0 \frac{\partial \tilde{g}_1}{\partial z}$. Hence, $d\nu^1$ can be obtained by

$$(5.38) \quad d\nu^1 = \left(\frac{\partial dg}{\partial \bar{z}} - \nu_0 \frac{\partial dg}{\partial z} \right) / \frac{\partial(\tilde{g}_1 + dg)}{\partial z}.$$

Next, for the second term $E_{\text{DEQ}}^2 = \int |\nu_0|^2 = \sum_{i=1}^{|\mathcal{F}|} \text{Area}(\mathcal{T}_i) |\nu_0(\mathcal{T}_i)|^2$, we compute the descent direction $d\nu^2$ by minimizing it iteratively. Analogously to (5.31), the Euler–Lagrange equation is given by

$$(5.39) \quad \frac{d}{dt} \Big|_{t=0} E_{\text{DEQ}}^2(\nu_0 + \delta t \omega) = 2 \sum_{i=1}^{|\mathcal{F}|} \text{Area}(\mathcal{T}_i) \nu_0(\mathcal{T}_i) \omega_i.$$

The derivative is negative when $\omega_i = -\nu_0(\mathcal{T}_i)$ for $i = 1, 2, \dots, |\mathcal{F}|$. Hence, we obtain the descent direction of the Beltrami coefficient: $d\nu^2 = -\nu_0$.

For the third part $E_{\text{DEQ}}^3 = \int |\nabla \nu_0|^2 = \sum_{i=1}^{|\mathcal{F}|} \int_{\mathcal{T}_i} |\nabla \nu_0|^2$, we first discretize it on the triangle mesh as in (5.32),

$$(5.40) \quad E_{\text{DEQ}}^3(\nu_0) = \sum_{i=1}^{|\mathcal{F}|} \left(\frac{\cot \theta_{i,0}}{2} (\nu_0(v_{i,1}) - \nu_0(v_{i,2}))^2 + \frac{\cot \theta_{i,1}}{2} (\nu_0(v_{i,2}) - \nu_0(v_{i,0}))^2 + \frac{\cot \theta_{i,2}}{2} (\nu_0(v_{i,0}) - \nu_0(v_{i,1}))^2 \right),$$

where $\mathcal{T}_i = [v_{i,0}, v_{i,1}, v_{i,2}]$ is the i th triangle element and $\theta_{i,0}, \theta_{i,1}, \theta_{i,2}$ are the three angles in it. We can then obtain the increment $d\nu^3 = \Delta \nu_0$.

Therefore, the descent direction for solving the optimization problem (5.1) is

$$(5.41) \quad d\nu = d\nu^1 + \alpha d\nu^2 + \beta d\nu^3.$$

Using the above descent direction formula, the updated Beltrami coefficient can be expressed as

$$(5.42) \quad \tilde{\nu}_1 = \nu_0 + \delta t d\nu,$$

where δt is the time step size.

After obtaining $\tilde{\nu}_1$, we apply the following cutoff function \mathbb{I} to it,

$$(5.43) \quad \mathbb{I}(\tilde{\nu}_1(z)) = \begin{cases} \tilde{\nu}_1(z) & \text{if } |\tilde{\nu}_1(z)| < 1, \\ (1 - \delta)\tilde{\nu}_1(z) & \text{if } |\tilde{\nu}_1(z)| \geq 1, \end{cases}$$

where δ is a prescribed cutoff parameter. In practice, we set $\delta = 0.1$. We denote $\nu_1 = \mathbb{I}(\tilde{\nu}_1)$. If $\|\nu_1\|_\infty$ is still larger than 1, we apply the cutoff function \mathbb{I} repeatedly until $\|\nu_1\|_\infty < 1$, which

Algorithm 5.2. BDED.

Input: A connected open triangulated surface \mathcal{S} , a population on each triangle of \mathcal{S} , a conformal flattening map $g_0 : \mathcal{S} \rightarrow \mathcal{D}^0$, a modification map $\tilde{g}_1 : \mathcal{D}^0 \rightarrow \mathcal{D}^1$.

Output: A map $g_1 : \mathcal{D}^0 \rightarrow \mathcal{D}^1$.

- 1: Compute the initial density function ρ_0 on $\tilde{g}_1 \circ g_0(\mathcal{S})$;
- 2: Compute the velocity $\mathbf{v}_0 = -\frac{\nabla \rho_0}{\rho_0}$;
- 3: Initialize $\nu_0 = \mu(\tilde{g}_1)$;
- 4: Fixing the boundary shape, obtain $\tilde{\nu}_1$ based on \mathbf{v}_0 and ν_0 using (5.38)–(5.42);
- 5: Obtain $\nu_1 = \mathbb{I}(\tilde{\nu}_1)$;
- 6: Use the LBS method to reconstruct g_1 from ν_1 ;

guarantees that the quasiconformal map associated with ν_1 is bijective. Finally, we obtain a bijective deforming map $g_1 : \mathcal{D}^0 \rightarrow \mathcal{D}^1$ by applying the LBS algorithm:

$$(5.44) \quad g_1 = \mathbf{LBS}(\nu_1).$$

We call this procedure *Beltrami density-equalizing descent (BDED)* and summarize it in Algorithm 5.2.

5.1.4. DEQ algorithm. In this section, we propose a method for computing the bijective DEQ maps.

For any given connected open triangle mesh \mathcal{S} , our goal is to find a quasiconformal mapping $f : \mathcal{S} \rightarrow \mathbb{R}^2$ such that $f(\mathcal{S})$ is a circular domain in \mathbb{R}^2 and the final density per unit area in $f(\mathcal{S})$ becomes a constant. Based on the GM and BDED steps, an iterative scheme is proposed to solve the optimization problem (5.1) (see Figure 5). More specifically, each iteration consists of two parts. First, a modification function is computed for obtaining a new planar circular domain using the GM step. Second, a diffusion-based updated map is calculated for the new domain using the BDED step. We then repeat the GM and BDED steps to further improve both the domain geometry and the mapping until convergence.

It is natural to ask why alternating between GM and BDED is needed to achieve the optimal final mapping result. To explain this, note that in our approach we regard the diffusion process as a quasiconformal flow. In the GM step, the domain alteration is induced by the energy (5.4) in the neighborhood of the inner boundaries. Then in the BDED step, we search for a diffusion-based descent direction with the new domain geometry prescribed. As the density at the regions away from the boundaries is not fully utilized in the GM step, the domain reconstructed and the descent direction result may not be optimal. By alternating between the GM and BDED steps, we can allow the density-equalization process to take place in a dynamically changing domain, thereby yielding an optimal circular domain and an optimal DEQ map.

The detailed procedure of the iterative scheme is as follows. Suppose g_n is obtained at the n th iteration, and \mathcal{D}^n is the corresponding circular domain. The density function ρ_n^y can be calculated using the same method as above. Then, we compute the modification map $\tilde{g}_{n+1} : \mathcal{D}^0 \rightarrow \mathcal{D}^{n+1}$ by Algorithm 5.1, where \mathcal{D}^{n+1} is the updated circular domain. Fixing the

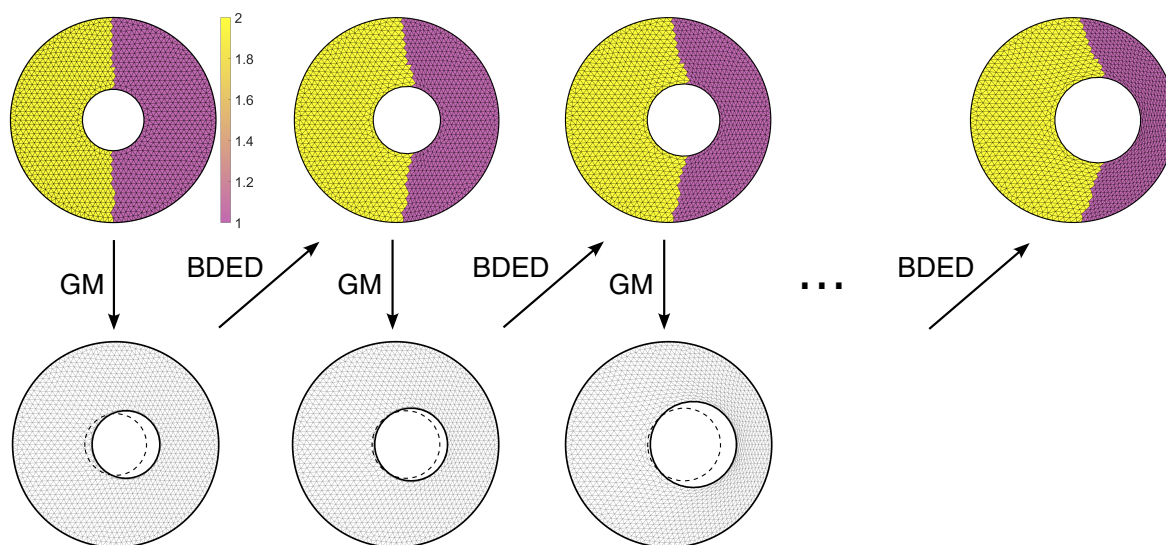


Figure 5. The proposed iterative scheme for computing DEQ maps. The GM method and the BDED method are applied iteratively to improve both the geometry of the target circular domain and the mapping result. Here, each triangle element is color-coded with the input population. The dashed curve in each GM step represents the previous inner boundary, and the solid curve represents its updated position.

shape of \mathcal{D}^{n+1} , we then compute the updated map $g_{n+1} : \mathcal{D}^0 \rightarrow \mathcal{D}^{n+1}$ using the BDED step and obtain the updated Beltrami coefficient $\nu_{n+1} = \mu(g_{n+1})$. We repeat the iteration and stop when $|E_{\text{DEQ}}^n - E_{\text{DEQ}}^{n-1}| < \epsilon$ for some stopping parameter ϵ , where E_{DEQ}^n denotes the energy at the n th iteration. The final DEQ map is then given by

$$(5.45) \quad f = g_N \circ g_0,$$

where N is the total number of iterations.

The proposed bijective DEQ mapping algorithm is summarized in Algorithm 5.3.

We remark that by skipping the GM step in the DEQ iterative scheme, a shape-preserving DEQ map can be obtained, where the centers and radii of the circular holes will remain unchanged throughout the iterations.

5.2. Bijective LDEQ map. After establishing the DEQ method, we extend it to incorporate landmark-matching constraints based on the model (4.7). Let $\{p_i\}_{i=1}^m$ and $\{q_i\}_{i=1}^m$ denote the landmark vertices at the interior of \mathcal{D}^0 and their corresponding target positions, respectively. Analogously to the DEQ method, here we find an optimal planar circular domain \mathcal{D} and an optimal Beltrami coefficient $\nu^* : \mathcal{D}^0 \rightarrow \mathbb{C}$, minimizing the following variational model,

$$(5.46) \quad E_{\text{LDEQ}}(g, \nu) = \int_{\mathcal{D}^0} |\nabla \rho(g)|^2 + \alpha \int_{\mathcal{D}^0} |\nu|^2 + \beta \int_{\mathcal{D}^0} |\nabla \nu|^2$$

subject to

$$(5.47) \quad \|\nu\|_\infty < 1,$$

Algorithm 5.3. Bijective DEQ map.

Input: A connected open triangulated surface \mathcal{S} , a population on each triangle, and a stopping parameter ϵ .

Output: A DEQ map $f : \mathcal{S} \rightarrow \mathcal{D}$, where \mathcal{D} is an optimal circular domain.

- 1: Compute an initial conformal flattening map $g_0 : \mathcal{S} \rightarrow \mathcal{D}^0$, where \mathcal{D}^0 is a circular domain.
 - 2: Compute the initial density function ρ_0 ;
 - 3: Set $n = 0$;
 - 4: **repeat**
 - 5: Apply the GM method (Algorithm 5.1) to get a modification map $\tilde{g}_{n+1} : \mathcal{D}^0 \rightarrow \mathcal{D}^{n+1}$, where \mathcal{D}^{n+1} is a new circular domain;
 - 6: Apply the BDED method (Algorithm 5.2) to obtain a map $g_{n+1} : \mathcal{D}^0 \rightarrow \mathcal{D}^{n+1}$;
 - 7: Update ρ_{n+1} based on g_{n+1} ;
 - 8: Update $n = n + 1$;
 - 9: **until** $|E_{\text{DEQ}}^n - E_{\text{DEQ}}^{n-1}| < \epsilon$;
 - 10: **return** $f = g_N \circ g_0$, where N is the total number of iterations;
-

$$(5.48) \quad \nu = \mu(g),$$

$$(5.49) \quad g(p_i) = q_i, \quad i = 1, \dots, m,$$

where \mathcal{D}^0 is the initial flattened circular domain, g is a quasiconformal map, $\mu(g)$ is the Beltrami coefficient of g , and ρ is the density function given by the population defined on \mathcal{S} . To achieve this, we develop an iterative scheme analogous to the DEQ algorithm, with some components of the BDED method modified to incorporate the landmark constraints.

More specifically, here we again start with an initial flattening map $g_0 : \mathcal{S} \rightarrow \mathcal{D}^0$ and apply the GM method to obtain a new circular domain \mathcal{D}^1 and a modification map $\tilde{g}_1 : \mathcal{D}^0 \rightarrow \mathcal{D}^1$. Then, we move on to the BDED step with the following changes. Instead of directly using the LBS method to solve the constraint optimization problem (5.46), we follow the approach in [31] and solve the problem using the penalty splitting method. We consider the following energy functional,

$$(5.50) \quad E_{\text{LDEQ}}^{\text{split}}(\nu, \mu) = \int_{\mathcal{D}^0} |\nabla \rho(g)|^2 + \alpha \int_{\mathcal{D}^0} |\nu|^2 + \beta \int_{\mathcal{D}^0} |\nabla \nu|^2 + \eta \int_{\mathcal{D}^0} |\nu - \mu(g)|^2$$

subject to the constraints that $\|\nu\|_\infty < 1$ and $g(p_i) = q_i$ for $i = 1, \dots, m$. Here, note that we have included an extra term $\eta \int_{\mathcal{D}^0} |\nu - \mu(g)|^2$. Recall that the LBS method reconstructs a mapping g based on a given Beltrami coefficient ν . However, if landmark constraints are enforced, the resulting Beltrami coefficient $\mu(g)$ may not be close to the input ν . In other words, the resulting mapping may not achieve the desired density-equalizing effect encoded by ν . To resolve this issue, here we include the term $\eta \int_{\mathcal{D}^0} |\nu - \mu(g)|^2$ in the energy functional, where η is a sufficiently large parameter. This term enforces that the Beltrami coefficient $\mu(g)$ is close to the solution ν of the model (5.46) even with the extra landmark constraints. The

resulting mapping is then a bijective landmark-matching density-equalizing map. We then repeat the GM step and the modified BDED step until convergence.

The detailed numerical procedure of the new approach is as follows. After getting the updated circular domain \mathcal{D}^{n+1} and the modification map $\tilde{g}_{n+1} : \mathcal{D}^0 \rightarrow \mathcal{D}^{n+1}$ from the GM step, we compute the bijective landmark-matching density-equalizing map $g_{n+1} : \mathcal{D}^0 \rightarrow \mathcal{D}^{n+1}$. Assume ν_n and μ_n are obtained at the n th iteration. Fixing of ν_n, μ_{n+1} can be obtained by minimizing

$$(5.51) \quad E_1^{\text{split}}(\nu_n, \mu) = \sum_{i=1}^{|\mathcal{F}|} \left(\int_{\mathcal{T}_i} |\nabla \rho_n^{\mathcal{F}}|^2 + \eta \int_{\mathcal{T}_i} |\nu_n - \mu|^2 \right).$$

By using the velocity $\mathbf{v}_n = -\frac{\nabla \rho_n^{\mathcal{V}}}{\rho_n^{\mathcal{V}}}$ induced by the density function $\rho_n^{\mathcal{V}}$, we can get the adjustment to the Beltrami coefficient $d\mu^1$ of the first term. For the second term, the descent direction is given by $d\mu^2 = -2(\mu - \nu_n)$. Then, the final adjustment is

$$(5.52) \quad d\mu = d\mu^1 + \eta d\mu^2.$$

By the formula for the descent direction above, we can obtain the updated Beltrami coefficient

$$(5.53) \quad \mu_{n+1} = \mu_n + \delta t d\mu,$$

where δt is the time step size.

Then, we follow the procedure in [31] and use LBS to compute $g_{n+1}^* : \mathcal{D}^0 \rightarrow \mathcal{D}^{n+1}$ such that $\mu(g_{n+1}^*) = \mu_{n+1}$ and g_{n+1}^* matches all the landmarks constraints, and then update μ_{n+1} by $\mu_{n+1} = \mu(g_{n+1}^*)$. Once μ_{n+1} is obtained, we fix it and minimize the following energy

$$(5.54) \quad E_2^{\text{split}}(\nu, \mu_{n+1}) = \sum_{i=1}^{|\mathcal{F}|} \left(\alpha \int_{\mathcal{T}_i} |\nu|^2 + \beta \int_{\mathcal{T}_i} |\nabla \nu|^2 + \eta \int_{\mathcal{T}_i} |\nu - \mu_{n+1}|^2 \right)$$

to get the updated Beltrami coefficient ν_{n+1} . By considering the Euler–Lagrange equation, minimizing the above energy is equivalent to solving

$$(5.55) \quad (\alpha I - \beta \Delta + \eta I) \nu_{n+1} = \eta \mu_{n+1},$$

where $\Delta = A^{-1}L$ (see Appendix A for the discretization details).

However, ν_{n+1} calculated above may not be associated with a landmark-matching density-equalizing diffeomorphism. Therefore, we use LBS with ν_{n+1} as the input together with the landmark constraints to obtain a landmark-matching quasiconformal map $g_{n+1} : \mathcal{D}^0 \rightarrow \mathcal{D}^{n+1}$, whose Beltrami coefficient $\mu(g_{n+1})$ closely resembles ν_{n+1} . Then, using the direction $d = \mu(g_{n+1}) - \nu_{n+1}$ as the descent direction, we update ν_{n+1} by $\nu_{n+1} \leftarrow \nu_{n+1} + \delta t d$, where δt is the time step size. Analogously to the DEQ algorithm, we repeat the above GM and density-equalizing mapping processes until $|E_{\text{LDEQ}}^n - E_{\text{LDEQ}}^{n-1}| < \epsilon$ for some stopping parameter ϵ , where E_{LDEQ}^n denotes the energy at the n th iteration. The proposed bijective LDEQ mapping algorithm is summarized in Algorithm 5.4.

Algorithm 5.4. Bijective LDEQ map.

Input: A connected open triangulated surface \mathcal{S} , prescribed interior landmark correspondences $\{p_i\}_{i=1}^m \leftrightarrow \{q_i\}_{i=1}^m$, a population on each triangle, and a stopping parameter ϵ .

Output: A bijective landmark-matching DEQ map $f : \mathcal{S} \rightarrow \mathcal{D}$, where \mathcal{D} is an optimal circular domain.

- 1: Compute an initial conformal flattening map $g_0 : \mathcal{S} \rightarrow \mathcal{D}^0$, where \mathcal{D}^0 is a circular domain.
 - 2: Compute the initial density function ρ_0 ;
 - 3: Set $n = 0$;
 - 4: Initialize $\mu_0 = \nu_0 = 0$;
 - 5: **repeat**
 - 6: Apply the GM method (Algorithm 5.1) to get a modification map $\tilde{g}_{n+1} : \mathcal{D}^0 \rightarrow \mathcal{D}^{n+1}$, where \mathcal{D}^{n+1} is a new circular domain;
 - 7: Fixing the boundary shape, obtain μ_{n+1} using μ_n and ν_n by (5.51)–(5.53);
 - 8: Use LBS to reconstruct $g_{n+1}^* : \mathcal{D}^0 \rightarrow \mathcal{D}^{n+1}$ based on μ_{n+1} together with the landmark constraints;
 - 9: $\mu_{n+1} \leftarrow \mu(g_{n+1}^*)$;
 - 10: Obtain ν_{n+1} using μ_{n+1} by (5.55);
 - 11: Use LBS to reconstruct $g_{n+1} : \mathcal{D}^0 \rightarrow \mathcal{D}^{n+1}$ based on ν_{n+1} together with the landmark constraints;
 - 12: $\nu_{n+1} \leftarrow \nu_{n+1} + \delta t d$;
 - 13: Update ρ_{n+1} based on g_{n+1} ;
 - 14: Update $n = n + 1$;
 - 15: **until** $|E_{\text{LDEQ}}^n - E_{\text{LDEQ}}^{n-1}| < \epsilon$;
 - 16: **return** $f = g_N \circ g_0$, where N is the total number of iterations;
-

6. Experimental results. In this section, we present experimental results to demonstrate the effectiveness of our proposed DEQ and LDEQ algorithms. The algorithms are implemented using MATLAB R2021a on the Windows platform. All experiments are conducted on a computer with an Intel(R) Core(TM) i9-12900 2.40 GHz processor and 32 GB memory. All surfaces are discretized in the form of triangular meshes. In the following experiments, the stopping parameter is set to be $\epsilon = 10^{-2}$, and the time step size is set to be $\delta t = 0.1$.

6.1. Bijective DEQ map. We first test our proposed DEQ algorithm on synthetic and real data. In all experiments, we set $\alpha = 0.1$ and $\beta = 0.05$. Koebe’s iteration method [29] is used for the initial flattening map.

We begin by presenting a synthetic example of mapping an annulus with four different density regions (Figure 6(a)). The mapping result obtained by our proposed DEQ method is shown in Figure 6(b), from which we can see that the position and size of the inner circular hole are optimized, yielding a smooth and bijective mapping result. As shown in the density histograms in Figures 6(c)–(d), the DEQ method effectively equalizes the density. From the

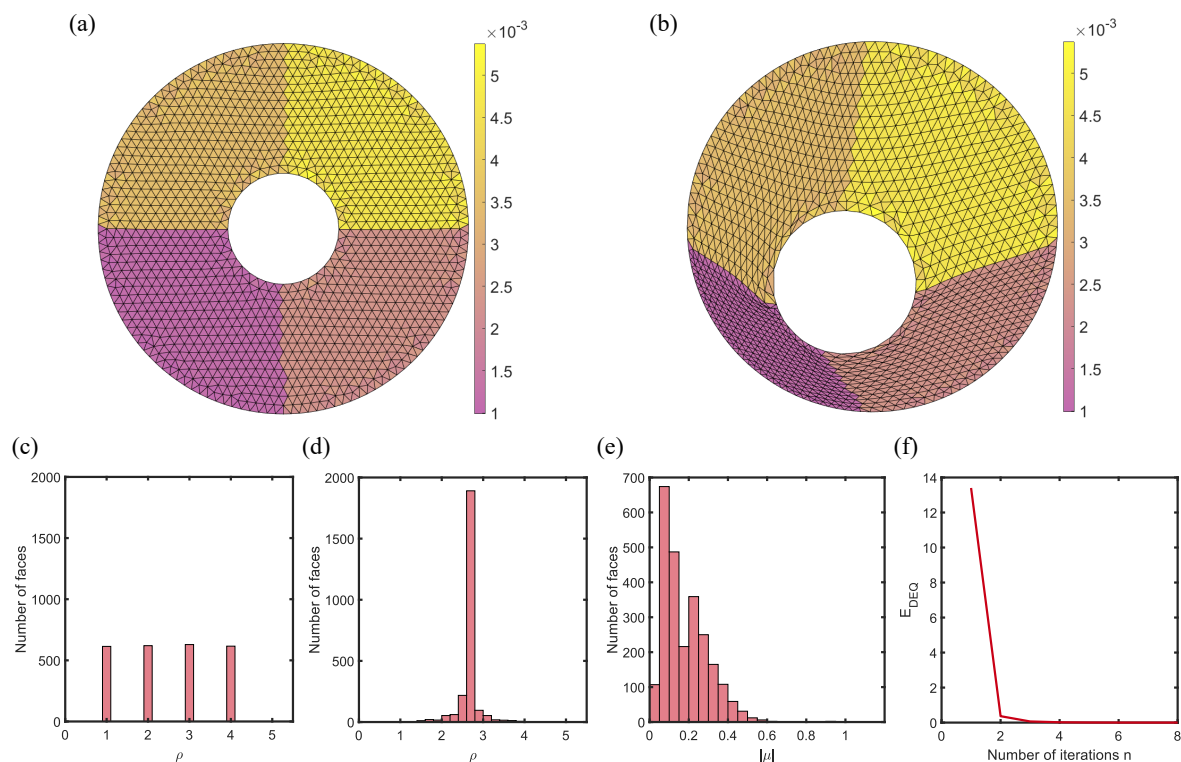


Figure 6. DEQ map of an annulus with four different density regions. (a) The initial shape color-coded with the prescribed population. (b) The final DEQ map. (c) The histogram of the initial density on every triangle element. (d) The histogram of the final density. (e) The histogram of the norm of the Beltrami coefficient $|\mu|$. (f) The energy E_{DEQ} throughout the iterations.

histogram of the norm of the Beltrami coefficient in Figure 6(e), we can also see that the quasiconformal distortion is low. Moreover, from the energy plot in Figure 6(f), we can see that the method converges quickly.

It is natural to ask whether the GM of the circular domain is important for achieving a good DEQ map. In Figure 7(a), we consider a multiply connected domain with two holes and set the population to be two times larger at the middle part of the domain. Note that the prescribed population will naturally cause an expansion at the center of the domain, pushing the vertices to the two sides. Now, if we skip the GM step and directly compute a DEQ map with the domain shape fixed (i.e., a shape-preserving DEQ map), it can be observed in Figure 7(b) that the mapping result has large distortion near the two holes. In particular, as the density gradient near the holes induces a velocity field that attempts to translate and enlarge the holes while they are enforced to be fixed, the triangle elements surrounding the two holes are largely distorted. By contrast, if we include the GM step, the DEQ mapping result becomes much more natural (Figure 7(c)), with the two holes moved to the two sides and enlarged. As shown in the density histograms in Figures 7(d)–(e), including the GM step gives a better density-equalization effect. We can also compare the norm of the Beltrami coefficients of the two mapping results as shown in Figures 7(f)–(g), from which we can see

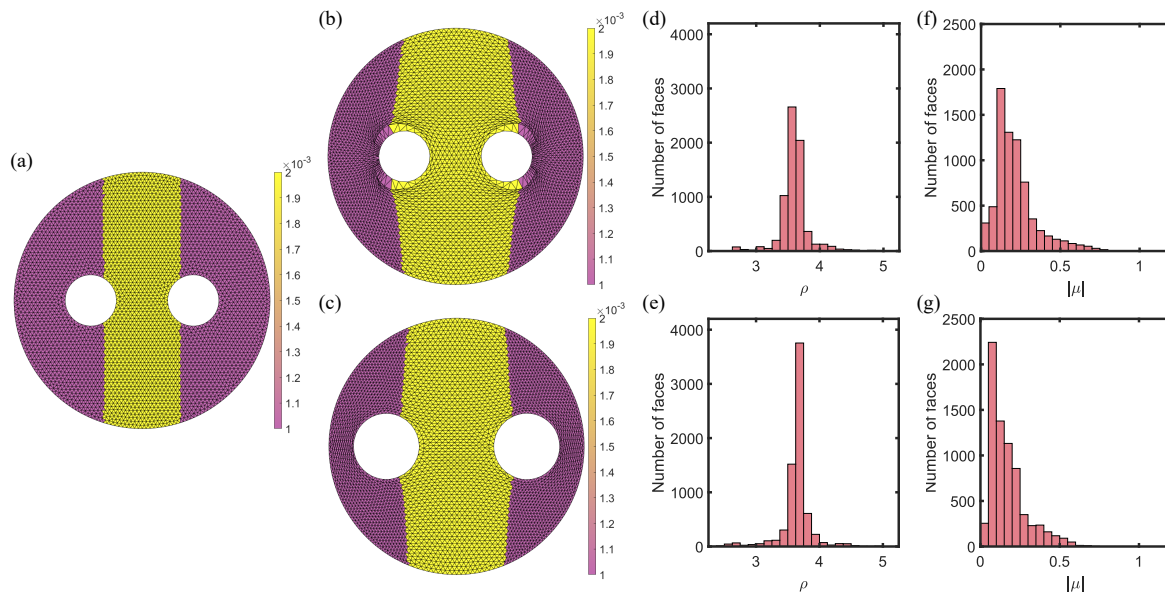


Figure 7. The importance of the GM step in the computation of DEQ maps. (a) A multiply connected domain with 2 holes, color-coded with the population prescribed on each triangle. (b) The DEQ mapping result without the GM step. (c) The DEQ mapping result with the GM step. (d) The final density histogram of DEQ without GM. (e) The final density histogram of DEQ. (f) The histogram of the norm of the Beltrami coefficient $|\mu|$ of DEQ without GM. (g) The histogram of $|\mu|$ of DEQ.

that the mapping result with GM yields a lower quasiconformal distortion. This demonstrates the significance of the GM step.

After testing our proposed algorithm with the above synthetic examples, we consider computing a bijective density-equalizing flattening map of a human face model with 1 hole using our algorithm (Figure 8(a)). Here, we set the population as the area of each triangle element on the original surface and apply the DEQ method in order to achieve an area-preserving parameterization. Figures 8(b) and (c) show the initial flattening map and the final mapping result obtained by the DEQ algorithm, from which we can see that the mouth, eyes, and eyebrows are shrunk in the initial map and restored in the final mapping result. The histogram of the initial density in Figure 8(d) shows that the area distortion in the initial flattening map is large. By contrast, the final density highly concentrates at 1 as shown in Figure 8(e), which indicates that the final bijective DEQ map effectively preserves the area ratio. We also consider the histogram of the norm of the Beltrami coefficient $|\mu|$ in Figure 8(f), which shows that the quasiconformal distortion of the mapping result is also small. Figure 8(g) shows the change of the energy E_{DEQ} throughout the iterations, from which we can see that our method converges rapidly.

Figure 9(a) shows another human face with two holes located at the eyes. It can be observed that the nose is shrunk in the initial flattening map (Figure 9(b)). By using the DEQ method with the triangle area as the prescribed population, we obtain an area-preserving parameterization (Figure 9(c)) with the nose enlarged. Again, we can see that the density is effectively equalized (Figures 9(d)–(e)) and the quasiconformal distortion is low (Figure 9(f)). Also, the energy E_{DEQ} is significantly reduced throughout the iterations (Figure 9(g)).

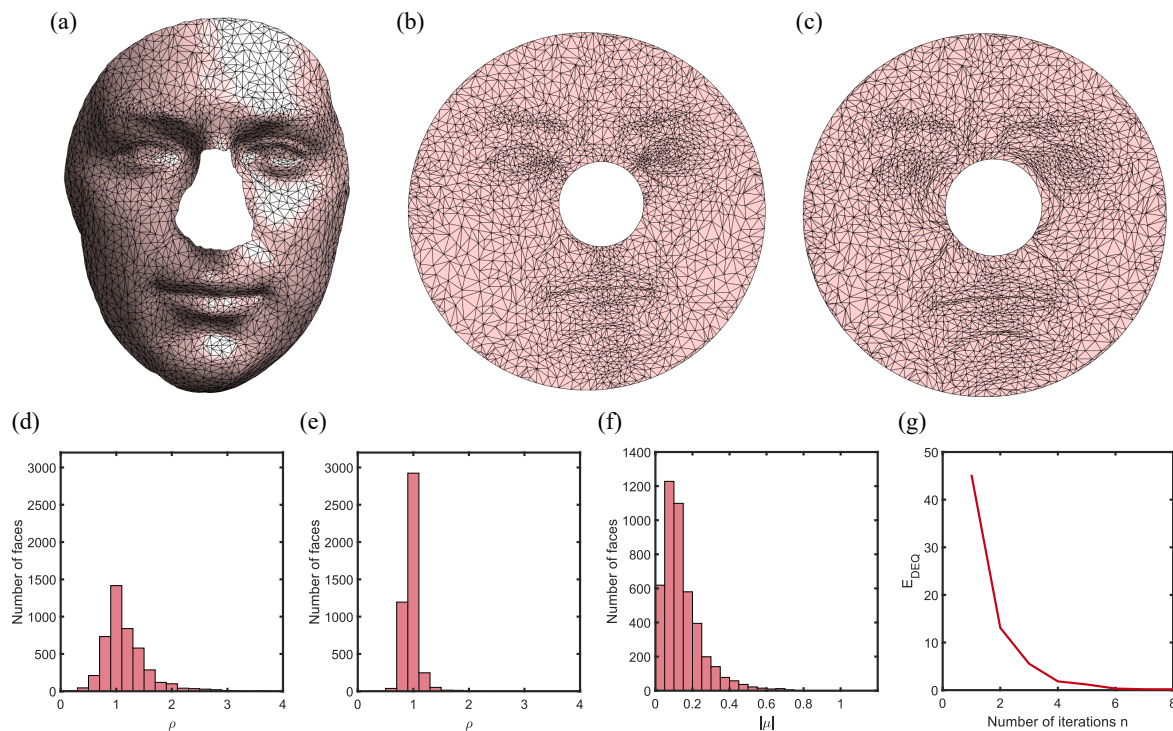


Figure 8. Bijective area-preserving parameterization of a human face with 1 hole. (a) The input surface. (b) The initial flattening map. (c) The final bijective DEQ map. (d) The histogram of the initial density on every triangle element. (e) The histogram of the final density. (f) The histogram of the norm of the Beltrami coefficient $|\mu|$. (g) The energy E_{DEQ} throughout the iterations.

It is noteworthy that our proposed algorithm can also be applied to simply connected open surfaces. Specifically, we can skip the GM step and directly apply the BDED method iteratively (i.e., using the shape-preserving DEQ approach) to obtain a bijective density-equalizing disk parameterization. We can then compare the existing DEM method [11] and our DEQ method for mapping simply connected open surfaces. For a fair comparison, the same stopping criterion is used in the two methods. Here, we consider flattening the human face in Figure 10(a) using the two methods, with the population set to be the area of each triangle element on the mesh except the mouth, and the population at the mouth set to be twice the area of the triangles there. As shown in the DEM result in Figure 10(b), the DEM method may lead to mesh foldovers. By contrast, the proposed DEQ method preserves the bijectivity of the mapping and so there is no mesh overlap as shown in Figure 10(c). We can further compare the final density histograms (Figure 10(d)–(e)) and the histograms of the Beltrami coefficient $|\mu|$ (Figure 10(f)–(g)), from which we can see that the DEQ method achieves a comparable density-equalizing effect while significantly reducing the quasiconformal distortion.

Figure 11 shows several other examples of simply connected and multiply connected open surfaces and the DEQ mapping results, from which we can again see that the mappings are highly density-equalizing and bijective.

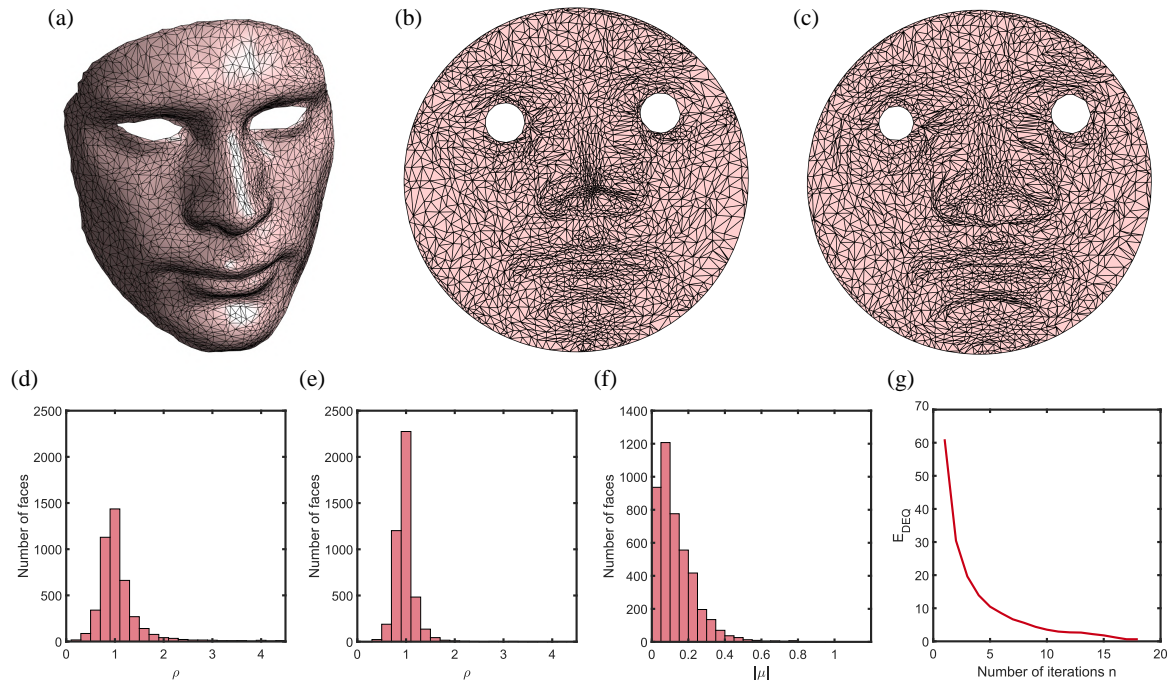


Figure 9. Bijective area-preserving parameterization of a human face with 2 holes. (a) The input surface. (b) The initial flattening map. (c) The final bijective DEQ map. (d) The histogram of the initial density on every triangle element. (e) The histogram of the final density. (f) The histogram of the norm of the Beltrami coefficient $|\mu|$. (g) The energy E_{DEQ} throughout the iterations.

To provide a more quantitative analysis of our DEQ algorithm, we present detailed statistics in Table 1, from which we can see that our method is highly efficient and accurate. Specifically, the algorithm requires less than 30 iterations in going back and forth between the GM and BDED steps for most examples and takes only a few seconds in total. By considering the variance of the normalized final density, we can see that the mapping results are highly density equalizing. Also, the mean value of the norm of the Beltrami coefficient shows that our method yields a low quasiconformal distortion. Moreover, as the bijectivity is guaranteed by our method, there is no mesh overlap in the mapping results.

Table 2 presents a quantitative comparison between DEQ and DEQ without the GM step. As shown in the table, incorporating the GM step results in a better density-equalization effect and lower quasiconformal distortion. Moreover, it is noteworthy that including the GM step in the DEQ method significantly accelerates the computation. This can be explained by the fact that the GM step effectively simplifies the optimization problem, thereby reducing the number of iterations needed. Recall that our goal is to find a bijective density-equalizing diffeomorphism by solving (5.1). As the initial circular domain may not be suitable for the deformation, more iterations may be necessary to achieve the desired result. By contrast, with the GM step included, we can modify the domain based on the diffusion process and quasiconformal energy, thereby getting a more suitable domain for the deformation. Consequently, the iterative process is accelerated significantly with the use of GM.

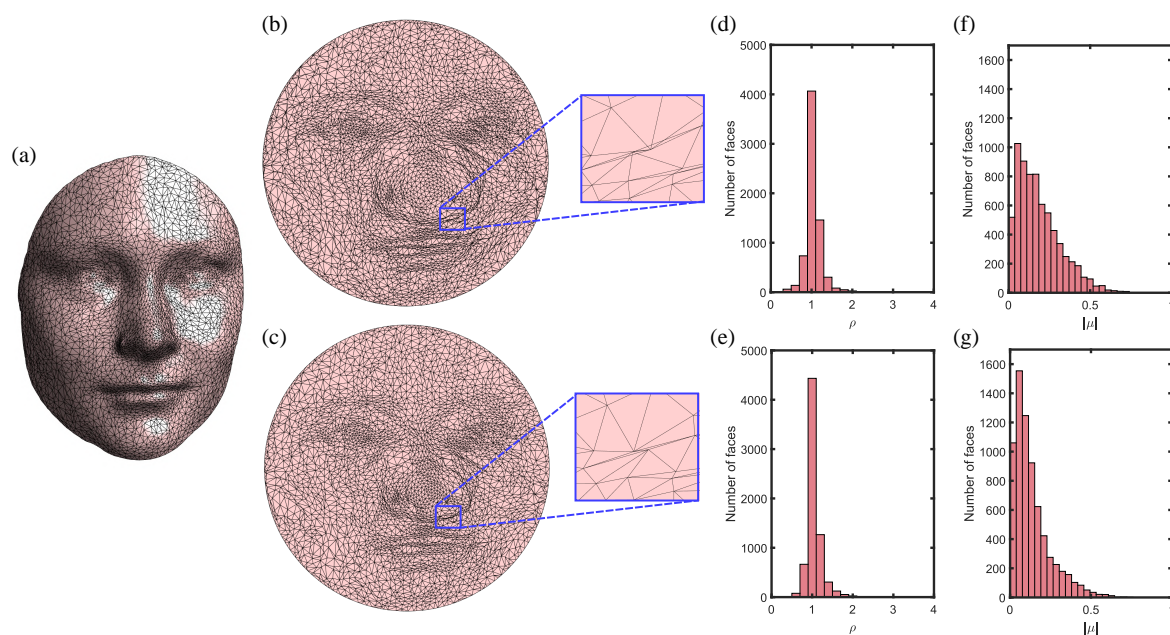


Figure 10. Comparison between DEM [11] and the proposed DEQ method for simply connected open surfaces. (a) A simply connected open face. (b) The DEM result with a zoom-in showing the presence of mesh overlaps. (c) The DEQ result with a zoom-in of the same region. (d) The final density histogram of DEM. (e) The final density histogram of DEQ. (f) The histogram of $|\mu|$ of DEM. (g) The histogram of $|\mu|$ of DEQ.

We further compare the performance of our DEQ method and the existing DEM method [11] for mapping simply connected open surfaces, with the same stopping criterion used in the two methods. As shown in Table 3, DEQ effectively reduces the quasiconformal distortion while achieving a comparable density-equalizing effect. It also outperforms DEM in terms of the bijectivity of the mappings. This demonstrates the effectiveness of the proposed DEQ method for not only multiply connected open surfaces but also simply connected open surfaces.

6.2. Bijective LDEQ maps. Next, we test our proposed LDEQ algorithm for computing landmark-matching density-equalizing maps. In all experiments, we set $\alpha = 0.1$, $\beta = 0.05$, and $\eta = 10$. We first consider a synthetic example of an annulus in \mathbb{R}^2 with the input population and prescribed constraints shown in Figure 12(a). Note that the input population is identical to the one used in the example in Figure 6. It is easy to see that the LDEQ mapping result is bijective and all the corresponding landmarks are matched (Figure 12(b)). Also, as shown in the histograms of the initial density, final density, and norm of the Beltrami coefficient (Figures 12(c)–(e)), the density is well-equalized and the quasiconformal distortion is low. We can further compare the LDEQ mapping result with the DEQ mapping result in Figure 6, from which we can clearly see the effect of the landmark constraints on the mapping result.

We then consider computing the LDEQ mappings for multiply connected open surfaces in \mathbb{R}^3 . The first two rows in Figure 13 show two examples of multiply connected human face surfaces and the LDEQ mapping results. Here, the input population is the area of every triangle element, and landmark constraints are enforced to control the position of prominent features such as the mouth or the nose of the human faces. From the LDEQ mapping results,

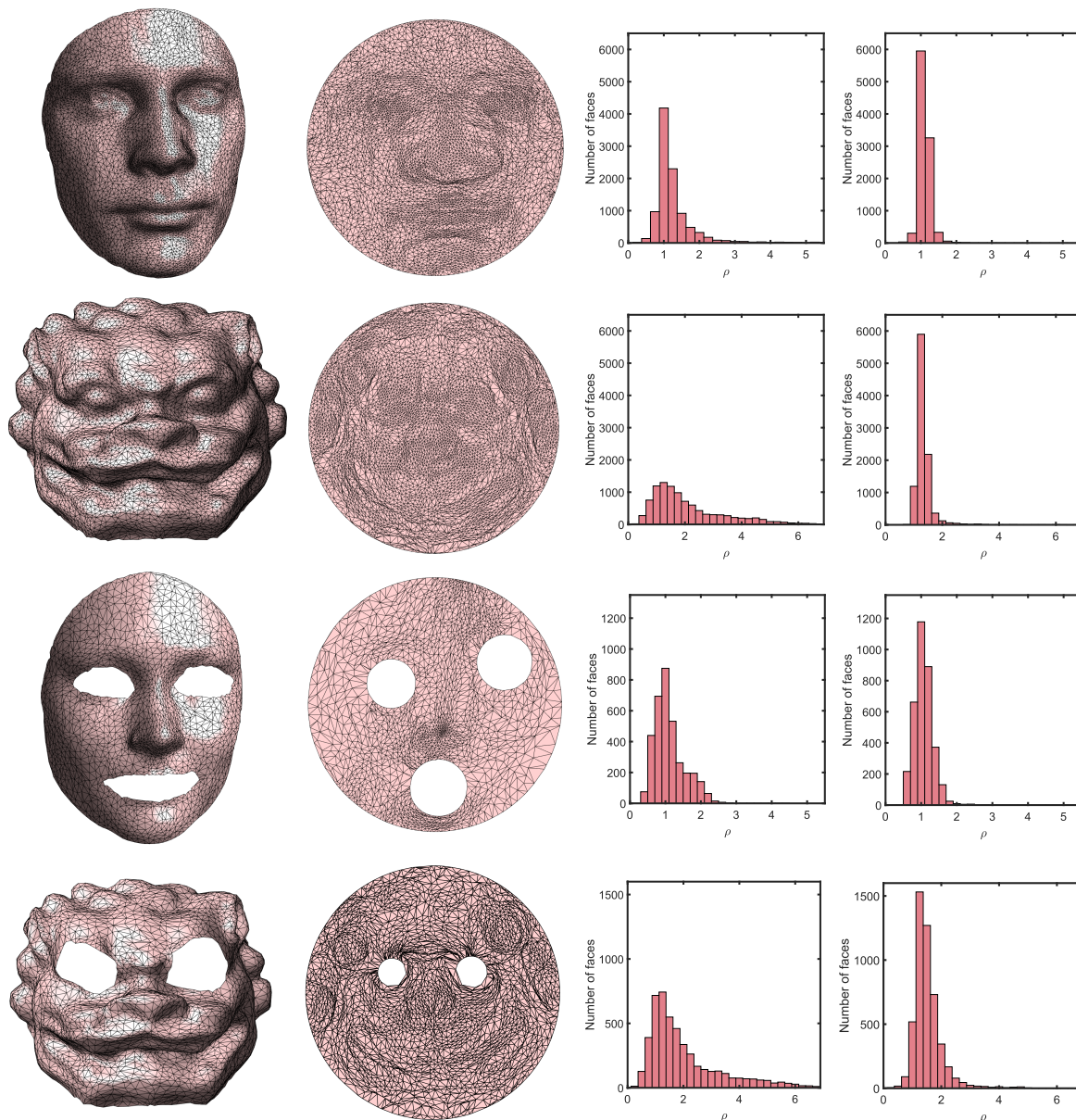


Figure 11. More examples of simply connected and multiply connected open surfaces and the DEQ mapping results. Each row shows one example. Left to right: The input surfaces, the DEQ mapping results, the histogram of the initial density, and the histogram of the final density.

it can be observed that the landmark constraints are satisfied and the mappings are bijective. Comparing the initial and final density histograms, we can also see that the density is well-equalized under the LDEQ method.

Analogously to the DEQ method, the LDEQ method can be applied to not only multiply connected open surfaces but also simply connected open surfaces. The last two rows in

Table 1

The performance of our DEQ algorithm. For each surface, we record the number of triangle elements, the time taken for the entire algorithm, the number of iterations, the variance of the normalized final density $\tilde{\rho} = \frac{\rho}{\text{Mean}(\rho)}$, where $\rho = \frac{\text{given population}}{\text{final area of triangle elements}}$, the mean value of the norm of the Beltrami coefficient μ , and the number of overlaps.

Surface	# faces	Time(s)	# iterations	Var($\tilde{\rho}$)	Mean($ \mu $)	# Overlaps
Lion (0 holes)	9999	2.42	32	0.07	0.18	0
Alex (0 holes)	10000	0.38	5	0.05	0.14	0
Square (0 holes)	10368	0.65	12	0.02	0.24	0
Annulus (1 hole)	2479	0.25	8	0.10	0.17	0
Rectangle (1 hole)	3000	0.28	11	0.10	0.19	0
Amoeba (1 hole)	14255	2.12	23	0.06	0.27	0
Alex (1 hole)	37999	1.23	13	0.03	0.11	0
Lion (2 holes)	4900	0.43	18	0.12	0.25	0
David (2 holes)	48853	4.86	28	0.03	0.15	0
Sophie (3 holes)	4000	0.33	14	0.05	0.16	0
David (3 holes)	47550	5.95	36	0.08	0.22	0

Table 2

The performance of the proposed DEQ algorithm compared with DEQ without the GM step.

Surface	DEQ			DEQ without GM		
	Time(s)	Var($\tilde{\rho}$)	Mean($ \mu $)	Time(s)	Var($\tilde{\rho}$)	Mean($ \mu $)
Annulus (1 hole)	0.25	0.10	0.17	0.28	0.13	0.19
Rectangle (1 hole)	0.28	0.10	0.19	0.42	0.13	0.26
Amoeba (1 hole)	2.12	0.06	0.27	3.32	0.10	0.29
Alex (1 hole)	1.23	0.03	0.11	2.12	0.04	0.17
David (2 holes)	4.86	0.03	0.15	7.54	0.08	0.20
David (3 holes)	5.95	0.08	0.22	8.25	0.10	0.28

Table 3

Comparison between the proposed DEQ method and the DEM method [11] for simply connected open surfaces.

Surface	# faces	DEQ				DEM			
		Time(s)	Var($\tilde{\rho}$)	Mean($ \mu $)	# Overlaps	Time(s)	Var($\tilde{\rho}$)	Mean($ \mu $)	# Overlaps
Peaks	4108	0.20	0.02	0.44	0	0.09	0.01	0.47	45
face	5000	1.35	0.07	0.13	0	1.01	0.05	0.21	128
Lion	9999	2.16	0.08	0.17	0	1.20	0.05	0.25	81
Alex	10000	0.27	0.06	0.15	0	0.15	0.04	0.28	63
Square	10368	0.74	0.02	0.23	0	0.52	0.01	0.27	22

Figure 13 show two examples of simply connected open surfaces and the LDEQ mapping results. Again, it can be observed that the mapping results are landmark-matching and bijective, and the density is well-equalized.

For a more quantitative analysis, Table 4 shows the performance of our LDEQ algorithm in terms of the computational time, the number of iterations, the variance of the normalized final density, the mean of the norm of the Beltrami coefficient, and the number of overlaps. It is noteworthy that because of the landmark constraints, LDEQ results in larger $\text{sd}(\rho)$

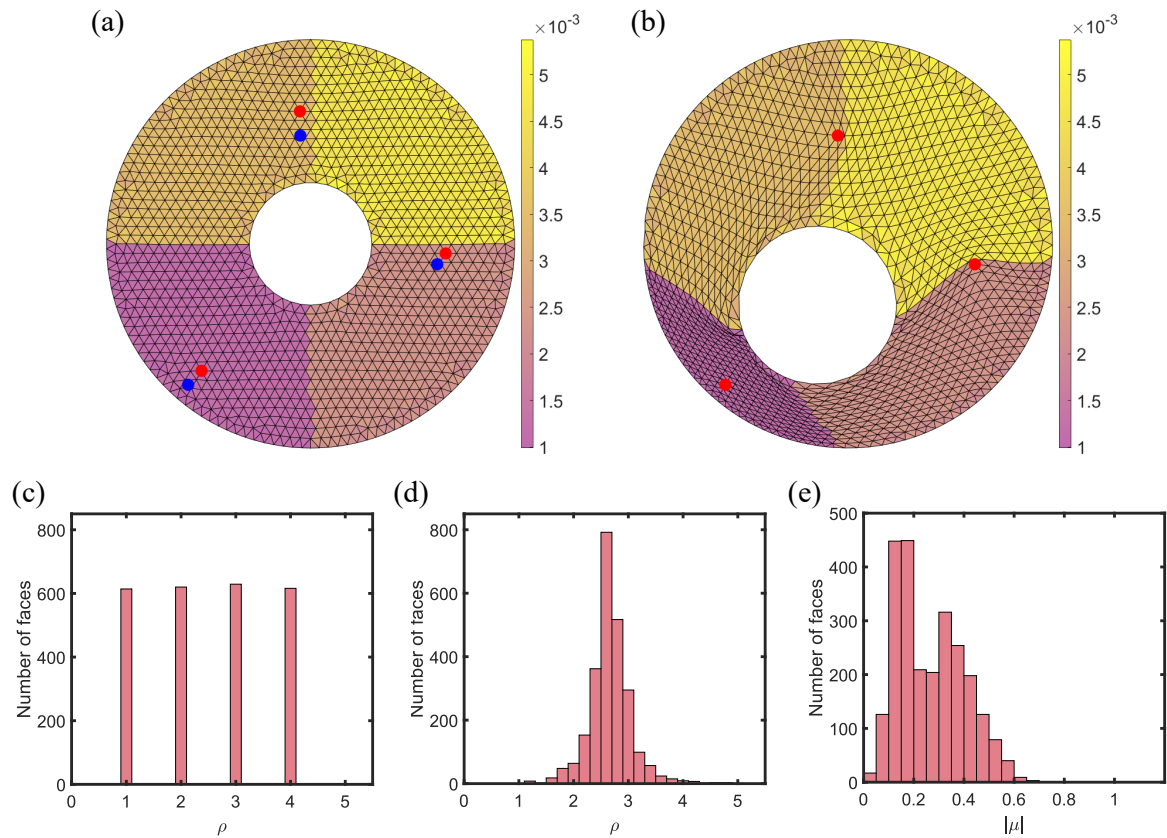


Figure 12. Bijective LDEQ map of an annulus with 4 different density regions. (a) The initial shape color-coded with the prescribed population. The source landmarks and target positions are highlighted in red and blue, respectively. (b) The LDEQ map with the red landmark vertices mapped to the target positions. (c) The histogram of the initial density. (d) The histogram of the final density. (e) The histogram of the norm of the Beltrami coefficient $|\mu|$.

and $\text{mean}(|\mu|)$ in general when compared to DEQ as in Table 1. Also, the computation of LDEQ is slower than that of DEQ. Nevertheless, LDEQ can effectively incorporate landmark constraints and guarantee the bijectivity of the mapping results.

6.3. Extreme cases. After testing the DEQ and LDEQ algorithms using the above experiments, here we consider some more extreme examples to further analyze the performance of our methods.

First, note that in the DEQ and LDEQ algorithms, the boundary vertices are allowed to move throughout the iterative process so as to achieve density equalization. It is natural to ask how the mapping results will be affected if the boundary vertices are required to be fixed. In Figures 14 and 15, we show two sets of examples of mapping simply connected and multiply connected domains with or without constraining the boundary vertices. It can be observed that fixing the boundary vertices will lead to a large geometric distortion, and the final mapping result is far from a density-equalizing map. This can be explained by the fact that the algorithm needs to preserve the bijectivity of the mapping throughout the process,

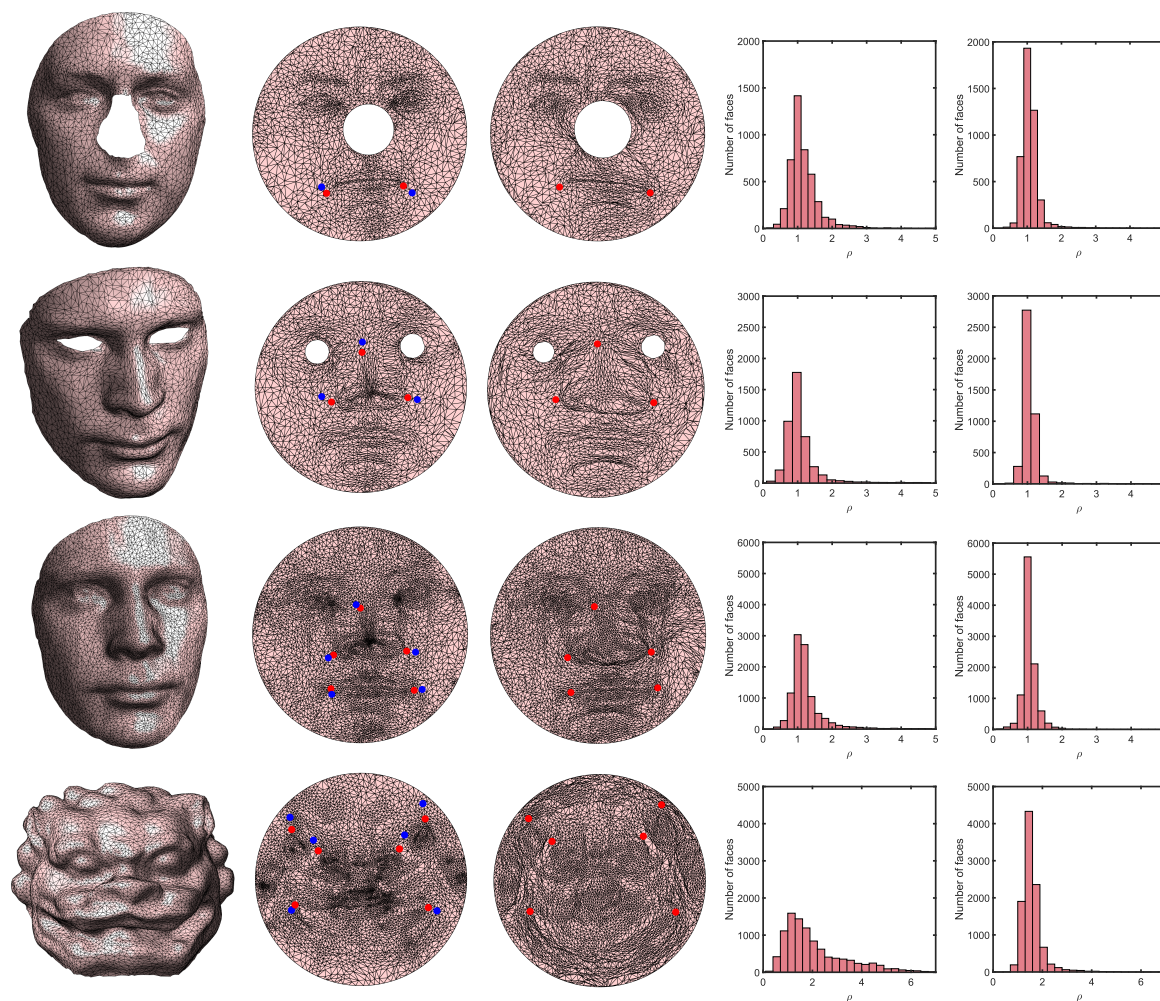


Figure 13. Examples of multiply connected and simply connected open surfaces and the LDEQ mapping results. Each row shows one example. Left to right: The input surface, the initial flattening map with the landmark constraints (red dots: source; blue dots: target), the LDEQ mapping result with the red dots mapped to the target position, the histogram of the initial density, and the histogram of the final density.

and hence even the nonboundary vertices cannot move freely based on the density flow if the boundary vertices are fixed. By contrast, by allowing the boundary vertices to move, the domain can be deformed much more naturally, and a better density-equalizing effect can be achieved.

One may also ask if there is any limitation on the landmark constraints that the LDEQ algorithm can handle. Note that in general, there is no mapping that can satisfy both the density-equalizing condition and the landmark-matching condition simultaneously. Moreover, if the desired landmark displacements are significantly against the density diffusion direction, the final mapping may have large geometric distortions. Figure 16(a) shows an example of mapping an annulus with two diametrically opposite landmarks desired to swap positions. It

Table 4
The performance of our LDEQ algorithm.

Surface	# faces	Time(s)	# iterations	Var($\bar{\rho}$)	Mean($ \mu $)	# overlaps
Alex (0 holes)	10000	0.65	8	0.07	0.17	0
Annulus (1 hole)	2480	0.42	12	0.11	0.26	0
Sophie (1 hole)	4499	0.35	15	0.06	0.16	0
David (2 holes)	4400	0.75	26	0.06	0.18	0
Lion (2 holes)	4900	0.63	22	0.12	0.25	0

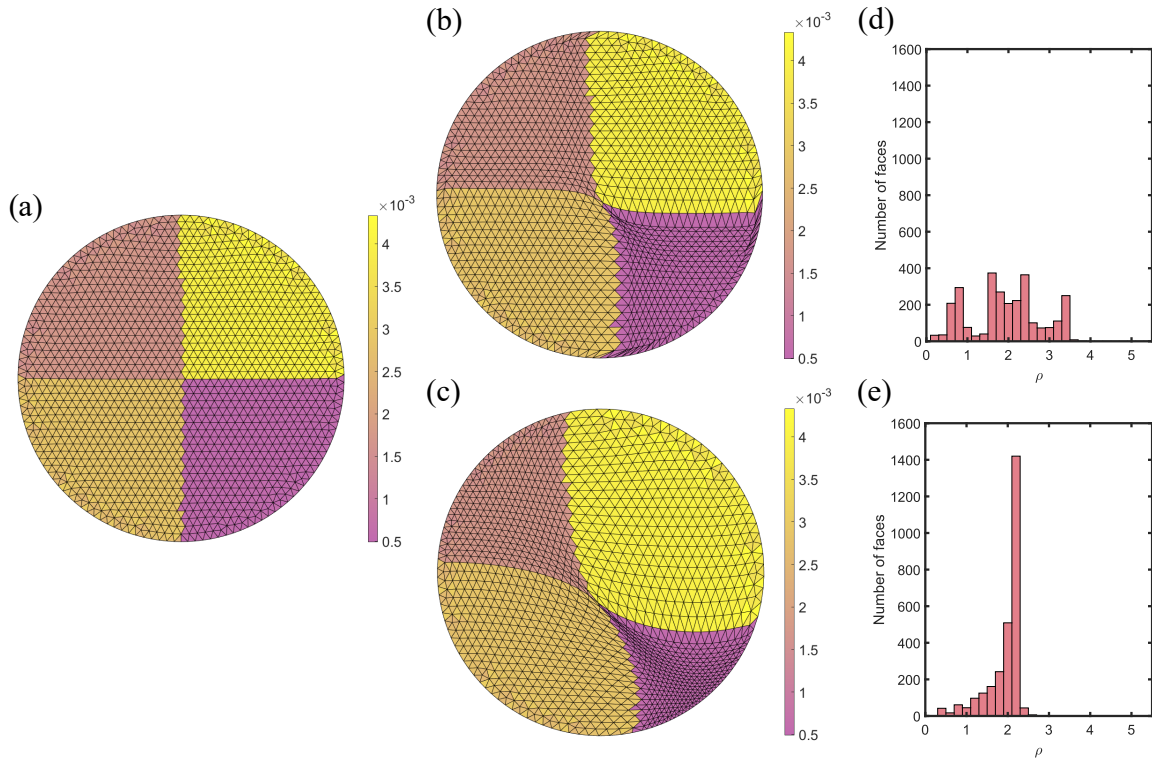


Figure 14. The effect of fixing the boundary vertices in the proposed DEQ method for simply connected open surfaces. (a) A simply connected domain color-coded with the population prescribed on each triangle. (b) The DEQ mapping result with all boundary vertices fixed. (c) The DEQ mapping result without constraining the boundary vertices. (d) The final density histogram of the result in (b). (e) The final density histogram of the result in (c).

can be observed in the LDEQ result (Figure 16(b)) that the final mapping is bijective, but many triangle elements become highly squeezed under the large deformation. Moreover, while the landmarks are very close to their corresponding target positions, the landmark mismatch is not perfectly zero. To explain this, note that the LDEQ algorithm enforces the bijectivity of the mapping at every iteration. As shown in the histogram of the norm of the Beltrami coefficient in Figure 16(e), there are some triangular faces with $|\mu|$ very close to 1 in the mapping result, which indicates that they are highly squeezed and close to degenerate. Then, any further tiny movement of the landmark vertices toward the target positions will lead

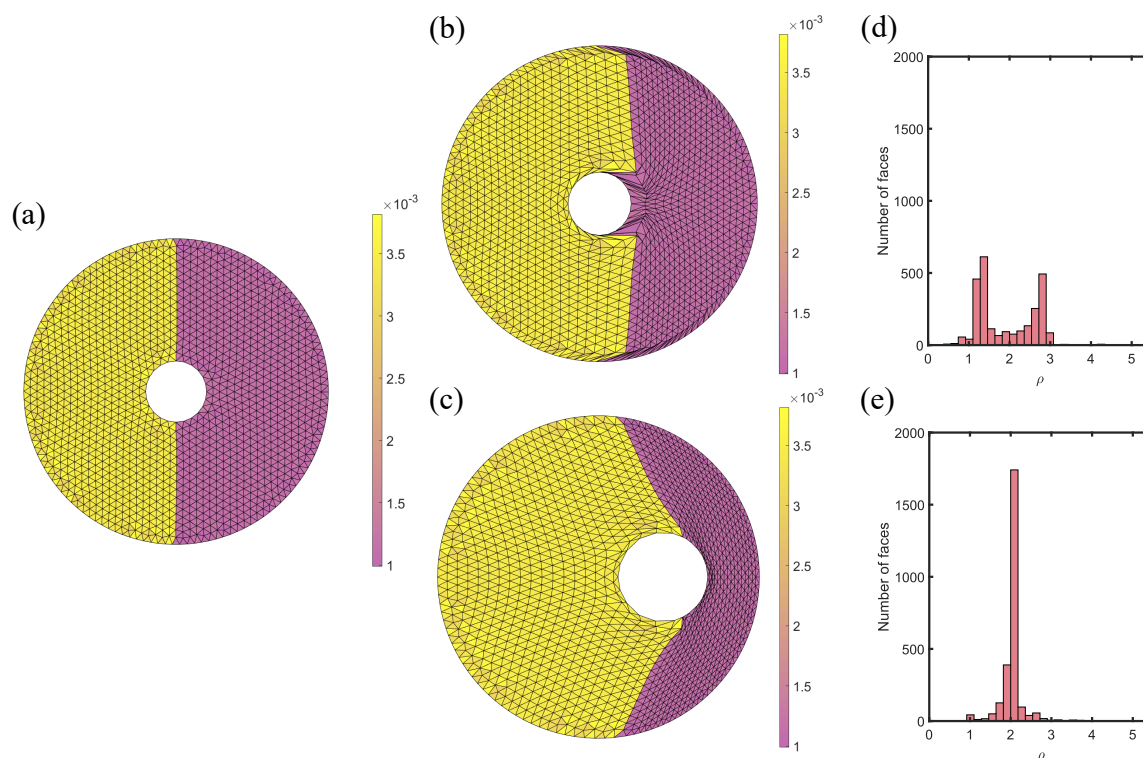


Figure 15. The effect of fixing the boundary vertices in the proposed DEQ method for multiply connected open surfaces. (a) A multiply connected domain color-coded with the population prescribed on each triangle. (b) The DEQ mapping result with all boundary vertices fixed. (c) The DEQ mapping result without constraining the boundary vertices. (d) The final density histogram of the result in (b). (e) The final density histogram of the result in (c).

to mesh overlaps (with $|\mu| > 1$), which are subsequently restored under the cutoff function \mathbb{I} and so the velocity field becomes 0. Therefore, the deformation will reach a limit after the landmarks are displaced to a certain extent. This example shows a limitation of our LDEQ method in handling certain extremely large landmark displacements. Nevertheless, from the histograms of the initial and final densities in Figures 16(c)–(d), we can see that the density-equalization effect in the final mapping result is still satisfactory even under such extreme landmark constraints. In Figure 17, we further consider another example with the same annulus domain and landmark constraints as in Figure 16 but a different prescribed population. Note that because of the difference in the population, the density gradient in the mapping problem here will be different from the last example. Consequently, the final LDEQ mapping result will also be different. The two examples together suggest that there can be multiple possible mapping results even with the same domain shape and landmark constraints.

7. Applications. In this section, we present several applications of our proposed bijective DEQ mapping method.

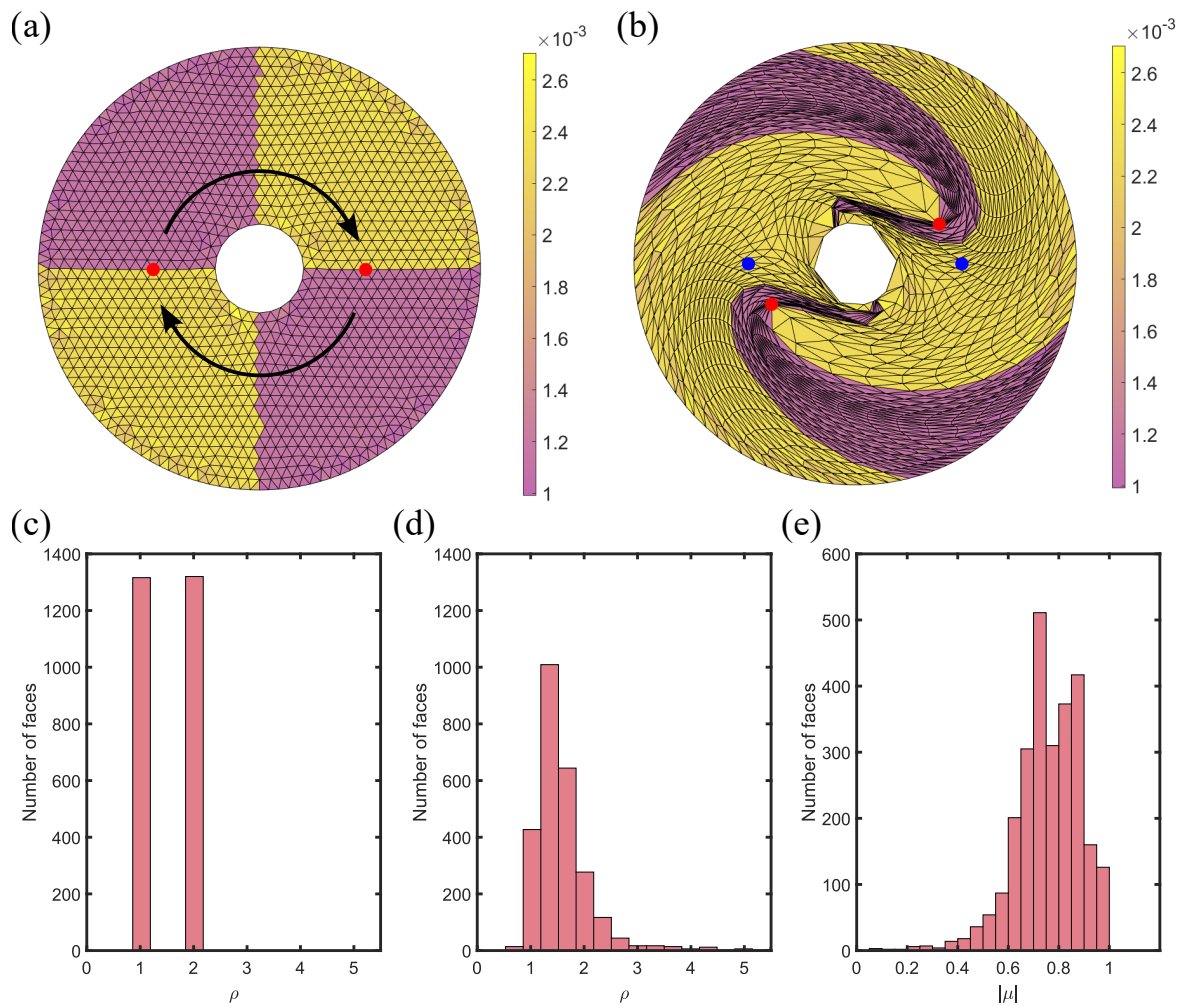


Figure 16. An example with extremely large landmark displacements. (a) An annulus color-coded with the prescribed population. Two diametrically opposite landmarks (highlighted in red) are desired to swap positions as indicated by the two arrows. (b) The LDEQ mapping result, with the final position of the landmark vertices highlighted in red and the desired target position highlighted in blue. It is noteworthy that the landmarks are not completely aligned due to the constraint in the bijectivity of the mapping. (c) The histogram of the initial density. (d) The histogram of the final density. (e) The histogram of the norm of the Beltrami coefficient $|\mu|$.

7.1. Surface remeshing. Our proposed method can be applied to surface remeshing. Given an open surface \mathcal{S} in \mathbb{R}^3 , the goal is to construct a new mesh structure on the surface with better quality. To achieve this, we can first prescribe a population on the surface and apply our proposed algorithm to compute a bijective DEQ map $f : \mathcal{S} \rightarrow \mathcal{D}$ onto a planar circular domain. Then, we can construct a regular mesh structure \mathcal{M} on \mathcal{D} . Finally, using inverse mapping f^{-1} , the regular mesh structure \mathcal{M} can be mapped back onto the surface \mathcal{S} and we obtain the remeshed surface $f^{-1}(\mathcal{M})$.

It is noteworthy that the effect of the DEQ map is closely related to the initial population distribution. Assigning larger populations to specific regions can increase the level of detail in

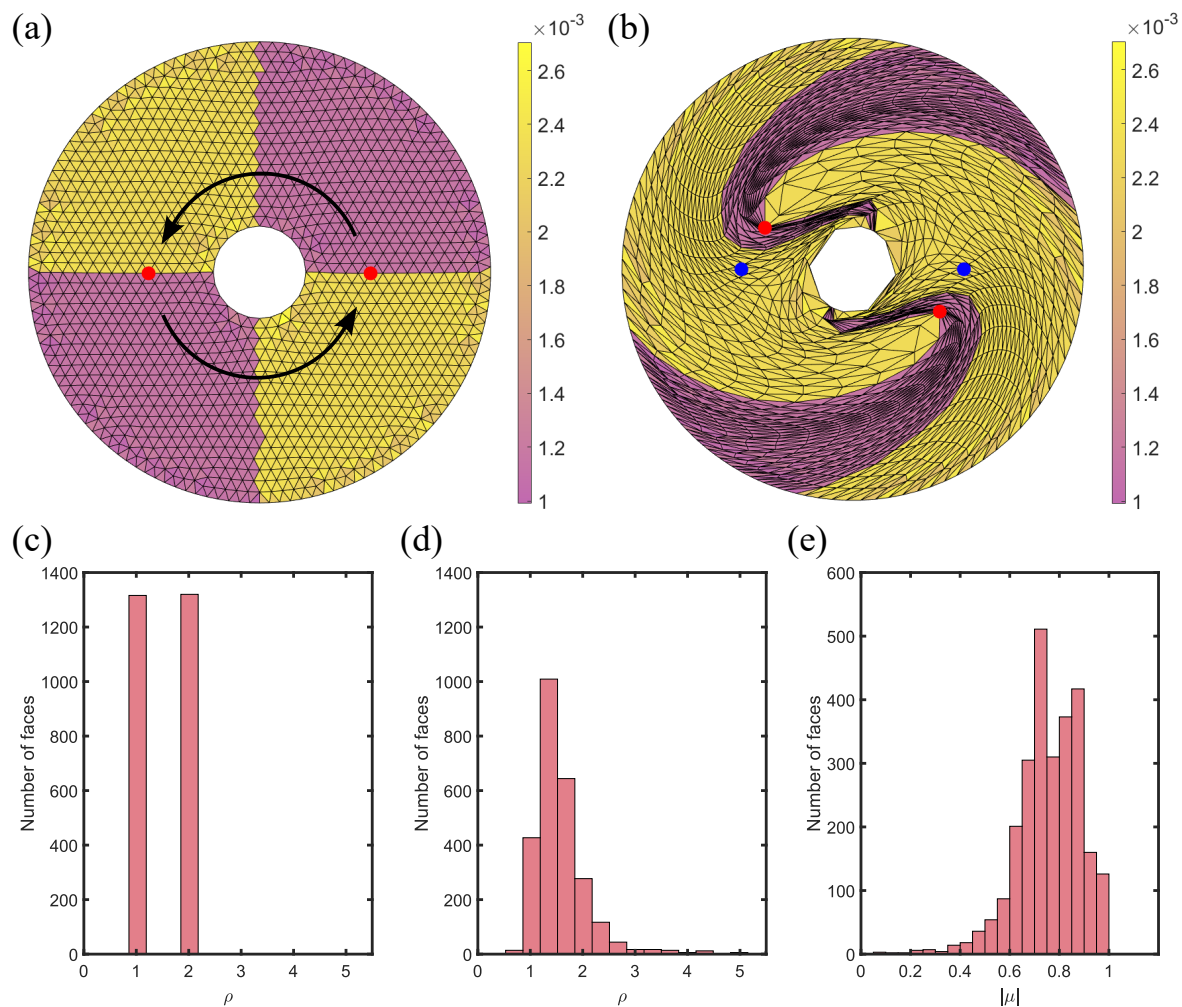


Figure 17. Another example with the same landmark displacements as in Figure 16 but a different prescribed population. (a) An annulus color-coded with the prescribed population. Two diametrically opposite landmarks (highlighted in red) are desired to swap positions. (b) The LDEQ mapping result, with the final position of the landmark vertices highlighted in red and the desired target position highlighted in blue. (c) The histogram of the initial density. (d) The histogram of the final density. (e) The histogram of the norm of the Beltrami coefficient $|\mu|$.

those areas, as they are enlarged in the DEQ mapping result and more points of \mathcal{M} are mapped onto those parts. Consequently, f^{-1} maps more points back onto the corresponding regions of \mathcal{S} , resulting in the display of more details in those domains. Also, we can further preserve certain desired features of \mathcal{S} during the remeshing process by using the proposed LDEQ method. Specifically, we can first place landmarks around the features and then compute the LDEQ mapping $g : \mathcal{S} \rightarrow \mathcal{D}$ with the landmarks mapped to their corresponding target positions. A regular mesh \mathcal{M} is then constructed on \mathcal{D} using DistMesh [44], and finally \mathcal{M} is interpolated onto \mathcal{S} using the inverse mapping g^{-1} . The resulting mesh $g^{-1}(\mathcal{M})$ is then a feature-preserving remeshed representation of \mathcal{S} .

Figure 18 shows two examples of remeshing a human face model using the DEQ method. We first consider a multiply connected human face with 2 holes (Figure 18(a)), in which we can see that the initial triangulation is highly irregular. Figure 18(b) shows the remeshing results obtained via conformal parameterization. It can be observed that the triangulations are highly nonuniform at the nose in the conformal remeshing results, which can be explained by the fact that the conformal method does not control the area distortion of the parameterization. By contrast, using our proposed DEQ method with the population set to be proportional to the triangle area of the original mesh, we can achieve a much better remeshing result with a more uniform distribution (Figure 18(c)). As discussed previously, our method can also be applied to simply connected open surfaces. In the second example, we consider remeshing a simply connected human face (Figure 18(d)) using quad meshes. Again, we can see in the quad-

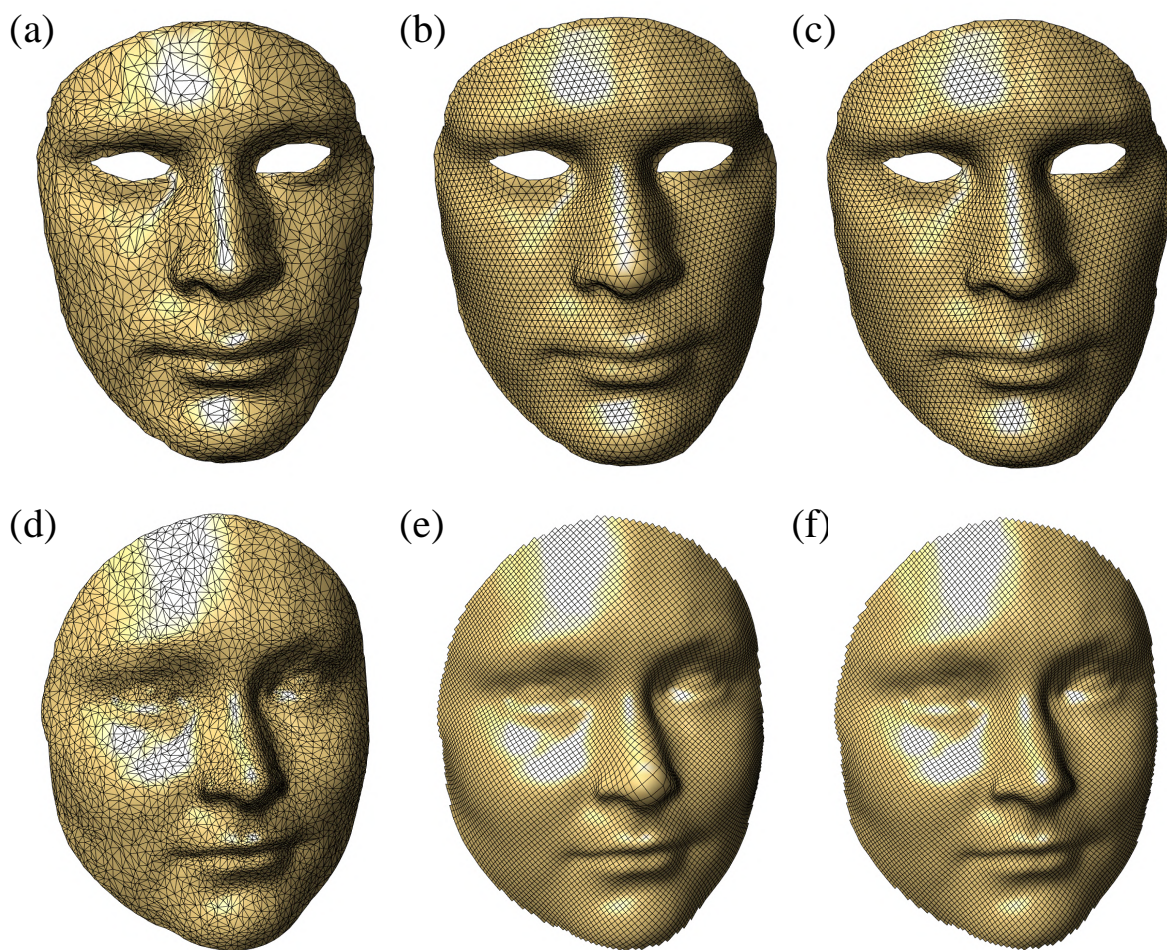


Figure 18. Remeshing human face models. (a) A multiply connected human face mesh with 2 holes. (b) The remeshing result via conformal parameterization. (c) The remeshing result via the proposed method. (d) A simply connected human face mesh. (e) The quad-remeshing result via conformal parameterization. (f) The quad-remeshing result via the proposed method.

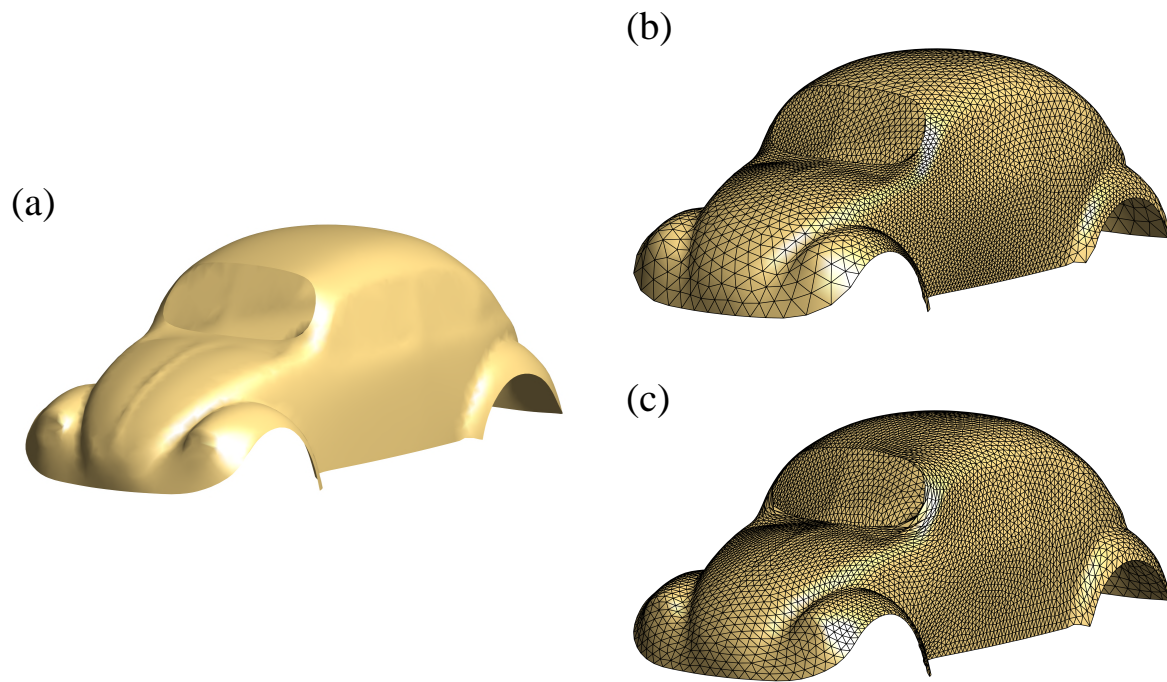


Figure 19. Remeshing a car model. (a) The original car surface. (b) The remeshing result via conformal parameterization. (c) The remeshing result via the proposed method.

remeshing result in Figure 18(e) that the conformal method leads to a nonuniform distribution, while the result obtained via our DEQ method is much more uniform (Figure 18(f)).

We then consider another experiment of remeshing a multiply connected car model (Figure 19(a)) using the LDEQ method. It can be observed that the remeshing result via conformal parameterization (Figure 19(b)) loses the prominent features of the car hood and the headlamps. Also, the triangle density of the remeshed surface is highly uneven. By contrast, using our proposed algorithm, we can achieve a much more balanced distribution of points. Here, landmarks are placed at the car hood and the headlamps in the mapping computation to ensure the preservation of the prominent features, yielding the remeshing result in Figure 19(c).

The above experiments show that our method is more advantageous than the conformal method for surface meshing.

7.2. Texture mapping. Our algorithm can be used for texture mapping on connected open surfaces. Specifically, after flattening a given 3D surface onto a planar circular domain, we can design a texture on the circular domain freely. We then use the inverse mapping to map the texture back onto the original surface.

Note that conformal parameterizations are widely used for texture mapping because they can preserve the local geometry. However, the area distortion of the designed texture is large. When compared with the conformal method, our DEQ algorithm can preserve the area if we set the initial population as the area of the original surface. Moreover, since our method is

obtained by solving (4.2), the angle distortion is also small while the density of the unit area becomes constant.

Figure 20 shows two examples of our method applied to different surfaces. A simply connected human face and a multiply connected human face are shown in Figures 20(a) and (d), respectively. We first design different textures and map them back onto the surfaces using conformal parameterization (Figures 20(b) and (e)). It can be observed that the right angles in the checkerboard pattern are well-preserved on the human face. However, some area distortions are visible around the nose and chin, indicating that the conformal method does not work well for those regions. By contrast, by applying the DEQ method, we can obtain texture mapping results with the checkerboard patterns uniformly mapped onto the faces (Figures 20(c) and (f)), which indicates the area is not distorted during the process. Moreover, it can be observed that orthogonality is preserved in most regions of the human

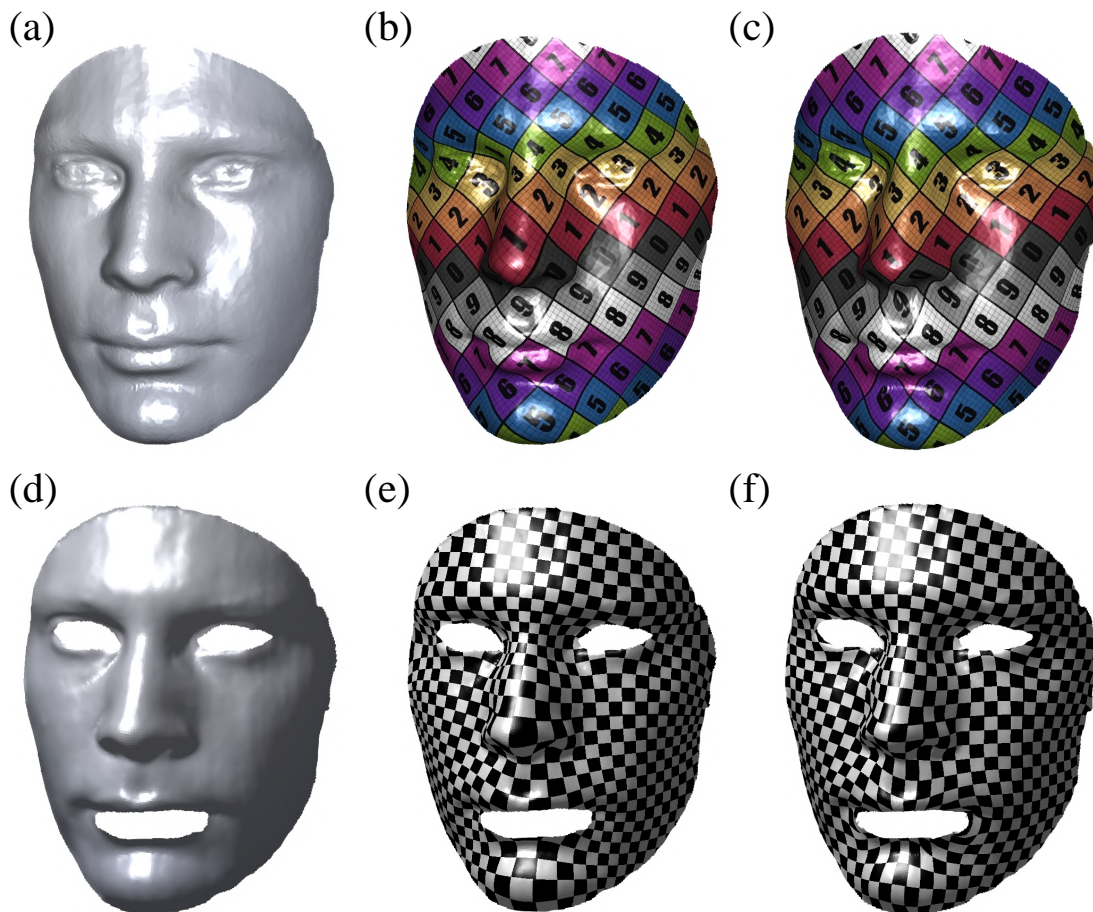


Figure 20. Texture mapping using the conformal and DEQ parameterizations. (a) A simply connected human face. (b) Texture mapping using conformal parameterization. (c) Texture mapping using the DEQ method. (d) A multiply connected human face. (e) Texture mapping using conformal parameterization. (f) Texture mapping using the DEQ method.

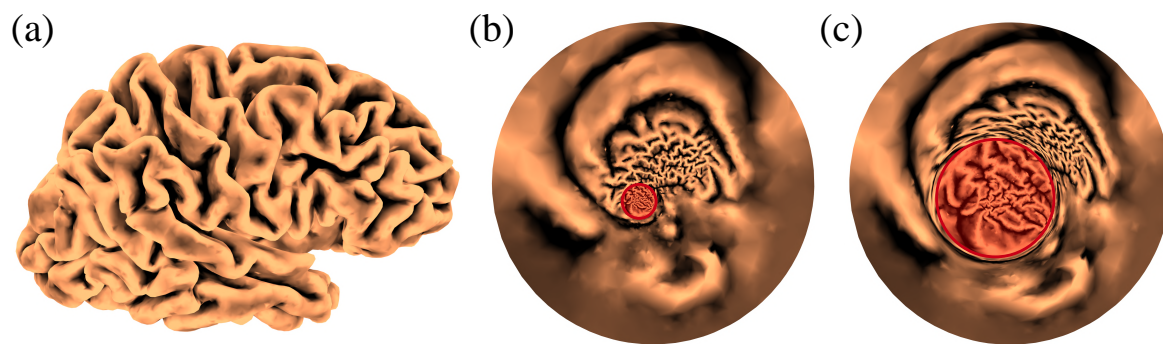


Figure 21. Application of our proposed method to medical visualization. (a) A highly convoluted human cortical surface. (b) Conformal parameterization of the surface onto the unit disk, where gyri and sulci are highly squeezed in a small region (highlighted in red). (c) The highlighted region is effectively enlarged using the DEQ method.

face, which is attributed to the inclusion of the quasiconformality terms in the DEQ energy minimization model. The examples show that our DEQ method can achieve density-equalizing texture mapping with minimal angle distortion.

7.3. Medical visualization. Our method can also be applied to medical visualization as an interactive shape magnifier. Given a complicated anatomical structure such as a human cortical surface (Figure 21(a)), one common approach for achieving a better visualization of it is to flatten it onto a simpler domain. However, conformal parameterization of such surfaces will easily produce a flattening result with certain parts highly squeezed in a small region (such as the sulci and gyri in Figure 21(b)), as the area distortion of conformal mappings is uncontrolled. This significantly hinders the visualization and shape analysis of such medical structures.

To overcome this issue, we can apply our DEQ method with a suitable population prescribed. More specifically, the user can first prescribe a larger population as a region of interest (ROI). Then, by computing a DEQ map, we can effectively enlarge the ROI while preserving the bijectivity of the parameterization result (Figure 21(c)). Moreover, as the quasiconformal distortion is also taken into consideration in the DEQ model, the local geometry of the ROI is well-preserved under the mapping.

8. Discussion. In this paper, we proposed a novel method for computing bijective DEQ mappings for both simply connected and multiply connected open surfaces. The main strategy is to minimize an energy functional that comprises a density term and two quasiconformality terms, where the density term measures the density distribution and the quasiconformality terms control the quasiconformal distortion and smoothness of the mapping. By minimizing the energy functional, our method can obtain a density-equalizing parameterization onto a 2D circular domain for any given connected open surface, with the bijectivity preserved and the quasiconformal distortion controlled. Landmark constraints can also be included in the mapping computation. Besides the applications to surface remeshing, texture mapping, and medical visualization demonstrated in our paper, the proposed approach can be potentially combined with other mapping methods for multiply connected domains [15, 43] for a broader range of shape analysis and geometry processing tasks.

While we have demonstrated the effectiveness of our method for handling different surfaces, it is noteworthy that the density diffusion formulation assumes a smooth density. Therefore, a highly discontinuous and extreme density function induced by the input population may affect the accuracy of the final mapping result. Also, as discussed previously, our gradient-based algorithms may only converge to local minima. Besides, the current approach involves solving for an optimal Beltrami coefficient and reconstructing the entire circular domain at each iteration, which may be time consuming for large datasets. With the recent development of parallelizable methods for conformal [6] and quasiconformal [62] mapping, a natural next step is to explore the parallel computation of density-equalizing maps; besides, our current work only focuses on connected open surfaces in \mathbb{R}^3 . In the future, we plan to extend the method for closed surfaces and volumetric domains [12, 60].

Appendix A. Discretization and computation of density diffusion. Below, we describe the discretization and computation of density diffusion. Readers are referred to [11] for more details.

A.1. Discretization of density. Note that for every vertex $v \in \mathcal{V}$, the point density $\rho^{\mathcal{V}}(v)$ depends on the face density $\rho^{\mathcal{F}}$ of its 1-ring face neighborhood. Hence, the vertex density is given by

$$(A.1) \quad \rho^{\mathcal{V}}(v) = \frac{\sum_{\mathcal{T} \in \mathcal{N}^{\mathcal{F}}(v)} \rho^{\mathcal{F}}(\mathcal{T}) \text{Area}(\mathcal{T})}{\sum_{\mathcal{T} \in \mathcal{N}^{\mathcal{F}}(v)} \text{Area}(\mathcal{T})},$$

where $\text{Area}(\mathcal{T})$ is the area of the face \mathcal{T} and $\mathcal{N}^{\mathcal{F}}(v)$ is the set of faces in the 1-ring neighborhood of the vertex v . If we regard $\rho^{\mathcal{V}}$ as a $|\mathcal{V}| \times 1$ matrix and $\rho^{\mathcal{F}}$ as an $|\mathcal{F}| \times 1$ matrix, the vertex density on the whole surface can be given by

$$(A.2) \quad \rho^{\mathcal{V}} = W^{\mathcal{F}\mathcal{V}} \rho^{\mathcal{F}},$$

where $W^{\mathcal{F}\mathcal{V}}$ is an $|\mathcal{F}| \times |\mathcal{V}|$ sparse matrix,

$$(A.3) \quad W^{\mathcal{F}\mathcal{V}} = \begin{pmatrix} W_{1,:} / \|W_{1,:}\|_1 \\ W_{2,:} / \|W_{2,:}\|_1 \\ \vdots \\ W_{|\mathcal{V}|,:} / \|W_{|\mathcal{V}|,:}\|_1 \end{pmatrix}$$

with

$$(A.4) \quad W_{ij} = \begin{cases} \text{Area}(T_j) & \text{if the } j\text{th triangle } T_j \text{ contains the } i\text{th vertex,} \\ 0 & \text{otherwise.} \end{cases}$$

A.2. Solving the diffusion equation. To solve the diffusion equation (3.12), we discretize the Laplacian operator on the triangle mesh. Using the finite element method, the Laplacian operator Δ can be discretized as

$$(A.5) \quad \Delta \rho(v_i) = \frac{1}{A(v_i)} \sum_{j \in \mathcal{N}(v_i)} \frac{(\cot \alpha_{ij} + \cot \beta_{ij})}{2} (\rho(v_j) - \rho(v_i)),$$

where $\mathcal{N}(v_i)$ is the 1-ring vertex neighborhood of v_i , α_{ij} , and β_{ij} are the two angles opposite to the edge $[v_i, v_j]$, and $A(v_i)$ is the vertex area with

$$(A.6) \quad A(v_i) = \frac{1}{3} \sum_{\mathcal{T} \in \mathcal{N}^{\mathcal{F}}(v_i)} \text{Area}(\mathcal{T}),$$

where $\mathcal{N}^{\mathcal{F}}(v_i)$ is the 1-ring face neighborhood of v_i . Alternatively, the Laplacian operator Δ can be represented as a matrix multiplication

$$(A.7) \quad \Delta = A^{-1}L,$$

where A is a $|\mathcal{V}| \times |\mathcal{V}|$ diagonal matrix consisting of the vertex area and L is a $|\mathcal{V}| \times |\mathcal{V}|$ sparse matrix with

$$(A.8) \quad L_{ij} = \begin{cases} -\sum_{v_k \in \mathcal{N}(v_i)} \frac{(\cot \alpha_{ik} + \cot \beta_{ik})}{2} & \text{if } j = i, \\ \frac{(\cot \alpha_{ij} + \cot \beta_{ij})}{2} & \text{if } v_j \in \mathcal{N}(v_i), \\ 0 & \text{otherwise.} \end{cases}$$

With the densities defined on all vertices, the diffusion equation (3.12) can be solved by the semidiscrete backward Euler method

$$(A.9) \quad \frac{\rho_n^{\mathcal{V}} - \rho_{n-1}^{\mathcal{V}}}{\delta t} = \Delta_{n-1} \rho_n^{\mathcal{V}},$$

where $\rho_n^{\mathcal{V}}$ is the vertex density at the n th iteration, Δ_n is the Laplace operator, and δt is the time step size. Combining with the Laplacian operator, the vertex density can be updated by

$$(A.10) \quad \rho_n^{\mathcal{V}} = (I - \delta t \Delta_{n-1})^{-1} \rho_{n-1}^{\mathcal{V}} = (A_{n-1} - \delta t L_{n-1})^{-1} (A_{n-1} \rho_{n-1}^{\mathcal{V}}).$$

Then, we consider discretizing the gradient operator ∇ . Let $\mathcal{T} = [v_i, v_j, v_k]$ be a triangle element, where the three directed edges are denoted by $e_{ij} = [v_i, v_j]$, $e_{jk} = [v_j, v_k]$, and $e_{ki} = [v_k, v_i]$. Denote N as the unit normal vector of \mathcal{T} . For any point x lying on \mathcal{T} , $\rho_n^{\mathcal{F}}$ can be interpolated by

$$(A.11) \quad \rho_n^{\mathcal{F}}(x) = \rho_n^{\mathcal{V}}(v_i) \varphi_i(x) + \rho_n^{\mathcal{V}}(v_j) \varphi_j(x) + \rho_n^{\mathcal{V}}(v_k) \varphi_k(x),$$

where $\rho_n^{\mathcal{V}}(v_i)$, $\rho_n^{\mathcal{V}}(v_j)$, $\rho_n^{\mathcal{V}}(v_k)$ are vertex densities at v_i, v_j, v_k , and φ_i are hat functions defined on the vertices

$$(A.12) \quad \varphi_i(p) = \begin{cases} 1 & \text{if } p = v_i, \\ \text{affine} & \text{if } p \in \mathcal{N}^{\mathcal{F}}(v_i), \\ 0 & \text{otherwise.} \end{cases}$$

Using the property that $\nabla \varphi_i = \frac{N \times e_{jk}}{2 \text{Area}(\mathcal{T})}$, $(\nabla \rho_n^{\mathcal{F}})(\mathcal{T})$ can be expressed as

$$(A.13) \quad \begin{aligned} (\nabla \rho_n^{\mathcal{F}})(\mathcal{T}) &= \nabla (\rho_n^{\mathcal{V}}(v_i) \varphi_i(x) + \rho_n^{\mathcal{V}}(v_j) \varphi_j(x) + \rho_n^{\mathcal{V}}(v_k) \varphi_k(x)) \\ &= \rho_n^{\mathcal{V}}(v_i) \nabla \varphi_i(x) + \rho_n^{\mathcal{V}}(v_j) \nabla \varphi_j(x) + \rho_n^{\mathcal{V}}(v_k) \nabla \varphi_k(x) \\ &= \frac{1}{2 \text{Area}(\mathcal{T})} N \times (\rho_n^{\mathcal{V}}(v_i) e_{jk} + \rho_n^{\mathcal{V}}(v_j) e_{ki} + \rho_n^{\mathcal{V}}(v_k) e_{ij}). \end{aligned}$$

After obtaining the gradient $(\nabla\rho)_n^{\mathcal{F}}$ on the triangle element, we can calculate the gradient operator on the vertices by

$$(A.14) \quad (\nabla\rho)_n^{\mathcal{V}} = W^{\mathcal{FV}}(\nabla\rho)_n^{\mathcal{F}}.$$

As mentioned in the main text, we enforce the Neumann boundary condition $\nabla\rho \cdot \mathbf{n} = 0$ to preserve the circular shape of all boundaries. However, during the iterations, some numerical errors may occur, resulting in minor changes to the boundary shape as described in [11]. Hence, after every iteration, we add one step to project all boundary vertices onto the prescribed shape given by Algorithm 5.1. As we have enforced the Neumann boundary condition, this projection step will not significantly affect the consistency between the density field and the shape deformation.

Appendix B. Computing the boundary Beltrami coefficient in terms of the hole parameters. In this section, we describe the process of computing the boundary Beltrami coefficient in terms of the parameters of an inner circular hole on the circular domain. Specifically, consider the i th inner circular hole and a face $\mathcal{T}_i = [p_0, p_1, p_2]$ in the 1-ring face neighborhood of the hole. There are then two possibilities:

- (i) Exactly one vertex of \mathcal{T}_i (say, p_0) lies on the circular hole.
- (ii) Exactly two vertices of \mathcal{T}_i (say, p_0 and p_1) lie on the circular hole.

To simplify our discussion, we focus on the first case below and then briefly describe the second case at the end of this section.

In the first case, we start by further simplifying some notations as follows. We denote the boundary Beltrami coefficient ν_l as ν and its corresponding triangle element \mathcal{T}_l as \mathcal{T} . Additionally, we denote the hole parameters $(c_i, r_i, T_{i,j})$ for the boundary point p_0 as (c, r, T) .

Consider a piecewise linear function g . Let $w_k = g(p_k)$, where $p_k = m_k + in_k$ and $w_k = s_k + it_k$ for $k = 0, 1, 2$. On the triangle face \mathcal{T} , g can be written as

$$(B.1) \quad g|_{\mathcal{T}}(x, y) = \begin{pmatrix} a_{\mathcal{T}}x + b_{\mathcal{T}}y + p \\ c_{\mathcal{T}}x + d_{\mathcal{T}}y + q \end{pmatrix},$$

where $D_x u(\mathcal{T}) = a_{\mathcal{T}}$, $D_y u(\mathcal{T}) = b_{\mathcal{T}}$, $D_x v(\mathcal{T}) = c_{\mathcal{T}}$, and $D_y v(\mathcal{T}) = d_{\mathcal{T}}$.

Based on the Beltrami equation, the Beltrami coefficient $\nu(g)$ can be expressed as

$$(B.2) \quad \nu(g) = \frac{\partial g / \partial \bar{z}}{\partial g / \partial z} = \frac{(a_{\mathcal{T}} - d_{\mathcal{T}}) + i(c_{\mathcal{T}} + b_{\mathcal{T}})}{(a_{\mathcal{T}} + d_{\mathcal{T}}) + i(c_{\mathcal{T}} - b_{\mathcal{T}})}.$$

For simplicity, we divide ν into two parts:

$$(B.3) \quad \nu = \rho + i\tau = \frac{R}{Q} + i\frac{I}{Q},$$

where

$$(B.4) \quad \begin{aligned} R &= a_{\mathcal{T}}^2 - d_{\mathcal{T}}^2 + c_{\mathcal{T}}^2 - b_{\mathcal{T}}^2, \\ I &= 2a_{\mathcal{T}}b_{\mathcal{T}} + 2c_{\mathcal{T}}d_{\mathcal{T}}, \\ Q &= a_{\mathcal{T}}^2 + d_{\mathcal{T}}^2 + c_{\mathcal{T}}^2 + b_{\mathcal{T}}^2 + 2a_{\mathcal{T}}d_{\mathcal{T}} + 2c_{\mathcal{T}}b_{\mathcal{T}}. \end{aligned}$$

The gradient of g on every triangle face \mathcal{T} is given by

$$(B.5) \quad \nabla_{\mathcal{T}}g = (D_xg(\mathcal{T}), D_yg(\mathcal{T}))^T,$$

where $D_xg(\mathcal{T}) = (D_xu, D_xv)$ and $D_yg(\mathcal{T}) = (D_yu, D_yv)$. To compute $\nabla_{\mathcal{T}}g$, we solve the following linear system:

$$(B.6) \quad \begin{pmatrix} p_1 - p_0 \\ p_2 - p_0 \end{pmatrix} \nabla_{\mathcal{T}}g = \begin{pmatrix} \frac{g(p_1) - g(p_0)}{|p_1 - p_0|} \\ \frac{g(p_2) - g(p_0)}{|p_2 - p_0|} \end{pmatrix},$$

where $[p_0, p_1]$ and $[p_0, p_2]$ are two edges of \mathcal{T} . After solving the above equation, the coefficients $a_{\mathcal{T}}$, $b_{\mathcal{T}}$, $c_{\mathcal{T}}$, and $d_{\mathcal{T}}$ can be obtained as follows:

$$(B.7) \quad \begin{aligned} a_{\mathcal{T}} &= A_0s_0 + A_1s_1 + A_2s_2, \\ b_{\mathcal{T}} &= B_0s_0 + B_1s_1 + B_2s_2, \\ c_{\mathcal{T}} &= A_0t_0 + A_1t_1 + A_2t_2, \\ d_{\mathcal{T}} &= B_0t_0 + B_1t_1 + B_2t_2, \end{aligned}$$

where

$$(B.8) \quad \begin{aligned} A_0 &= (n_1 - n_2) / \text{Area}(\mathcal{T}), \\ A_1 &= (n_2 - n_0) / \text{Area}(\mathcal{T}), \\ A_2 &= (n_0 - n_1) / \text{Area}(\mathcal{T}), \\ B_0 &= (m_2 - m_1) / \text{Area}(\mathcal{T}), \\ B_1 &= (m_0 - m_2) / \text{Area}(\mathcal{T}), \\ B_2 &= (m_1 - m_0) / \text{Area}(\mathcal{T}). \end{aligned}$$

With a focus solely on the boundary points, the formula above can be expressed as follows,

$$(B.9) \quad \begin{aligned} a_{\mathcal{T}} &= \tilde{A}_1 + A_0s_0, \\ b_{\mathcal{T}} &= \tilde{B}_1 + B_0s_0, \\ c_{\mathcal{T}} &= \tilde{A}_2 + A_0t_0, \\ d_{\mathcal{T}} &= \tilde{B}_2 + B_0t_0. \end{aligned}$$

where $\tilde{A}_1, \tilde{B}_1, \tilde{A}_2, \tilde{B}_2$ are constants.

Substituting the above results into (B.3), the Beltrami coefficient at p_0 can be expressed as

$$(B.10) \quad \begin{aligned} \nu(s_0, t_0) &= \rho(s_0, t_0) + i\tau(s_0, t_0) \\ &= \frac{R(s_0, t_0)}{Q(s_0, t_0)} + i \frac{I(s_0, t_0)}{Q(s_0, t_0)}, \end{aligned}$$

where

$$(B.11) \quad \begin{aligned} R(s_0, t_0) &= (\tilde{A}_1 + A_0s_0)^2 - (\tilde{B}_2 + B_0t_0)^2 + (\tilde{A}_2 + A_0t_0)^2 - (\tilde{B}_1 + B_0s_0)^2, \\ I(s_0, t_0) &= 2(\tilde{A}_1 + A_0s_0)(\tilde{B}_1 + B_0s_0)^2 + 2(\tilde{A}_2 + A_0t_0)(\tilde{B}_2 + B_0t_0), \\ Q(s_0, t_0) &= (\tilde{A}_1 + A_0s_0)^2 + (\tilde{B}_2 + B_0t_0)^2 + (\tilde{A}_2 + A_0t_0)^2 + (\tilde{B}_1 + B_0s_0)^2 \\ &\quad + 2(\tilde{A}_1 + A_0s_0)(\tilde{B}_2 + B_0t_0) + 2(\tilde{A}_2 + A_0t_0)(\tilde{B}_1 + B_0s_0)^2. \end{aligned}$$

Using polar coordinates, we can express s_0 as $r \cos \theta - c^1$ and t_0 as $r \sin \theta - c^2$, where $c = (c^1, c^2)$ represents the center of the circular hole, r denotes the radius of the circular hole, and θ denotes the angular coordinate of the point. Now, note that the angular component of the point can be expressed in terms of the tangent vector T at it. Also, by an abuse of notation, r is interchangeable with the radial vector as in the main text. Therefore, the boundary Beltrami coefficient ν_l associated with \mathcal{T}_l can be expressed in terms of the hole parameters (c, r, T) :

$$(B.12) \quad \nu(s_0, t_0) = \nu(c, r, T) = \frac{R(c, r, T)}{Q(c, r, T)} + i \frac{I(c, r, T)}{Q(c, r, T)}.$$

We now briefly discuss the second case where two vertices p_0, p_1 of the triangle element lie on the circular hole. Analogously to (B.9) in the first case, we can obtain the following expressions for $a_{\mathcal{T}}, b_{\mathcal{T}}, c_{\mathcal{T}}$, and $d_{\mathcal{T}}$:

$$(B.13) \quad \begin{aligned} a_{\mathcal{T}} &= \hat{A}_1 + A_0 s_0 + A_1 s_1, \\ b_{\mathcal{T}} &= \hat{B}_1 + B_0 s_0 + B_1 s_1, \\ c_{\mathcal{T}} &= \hat{A}_2 + A_0 t_0 + A_1 t_1, \\ d_{\mathcal{T}} &= \hat{B}_2 + B_0 t_0 + B_1 t_1, \end{aligned}$$

where $\hat{A}_1, \hat{B}_1, \hat{A}_2, \hat{B}_2, A_0, B_0, A_1,$ and B_1 are some constants.

Using polar coordinates, we can express the coordinates of the two boundary points as follows: $s_0 = r \cos \theta^0 - c^1$, $s_1 = r \cos \theta^1 - c^1$, $t_0 = r \sin \theta^0 - c^2$, and $t_1 = r \sin \theta^1 - c^2$, where $c = (c^1, c^2)$ represents the center of the circular hole, r denotes the radius of the circular hole, and θ^0, θ^1 denote the angular coordinates of the two points. Similarly to the previous case, we can express s_0, s_1, t_0, t_1 using (c, r^0, T^0) and (c, r^1, T^1) , where r^0, r^1 are two radial vectors and T^0, T^1 are two tangent vectors at the two points. We can then express ν_l in terms of c, r^0, r^1, T^0, T^1 as follows:

$$(B.14) \quad \begin{aligned} \nu(s_0, t_0, s_1, t_1) &= \nu(c, r^0, r^1, T^0, T^1) \\ &= \frac{R(c, r^0, r^1, T^0, T^1)}{Q(c, r^0, r^1, T^0, T^1)} + i \frac{I(c, r^0, r^1, T^0, T^1)}{Q(c, r^0, r^1, T^0, T^1)}. \end{aligned}$$

Finally, note that in the domain alteration step of the GM method, all points on an inner boundary are enforced to rescale and rotate consistently. Therefore, we can further simplify the above expression as $\nu = \nu(c, r, T)$ without ambiguity.

REFERENCES

- [1] N. AIGERMAN, R. PORANNE, AND Y. LIPMAN, *Lifted bijections for low distortion surface mappings*, ACM Trans. Graph., 33 (2014), 69.
- [2] A. I. BOBENKO, S. SECHELMANN, AND B. SPRINGBORN, *Discrete conformal maps: Boundary value problems, circle domains, Fuchsian and Schottky uniformization*, in Advances in Discrete Differential Geometry, Springer, Berlin, 2016, pp. 1–56.
- [3] E. CHIEN, Z. LEVI, AND O. WEBER, *Bounded distortion parametrization in the space of metrics*, ACM Trans. Graph., 35 (2016), 215.
- [4] G. P. T. CHOI, *Efficient conformal parameterization of multiply-connected surfaces using quasi-conformal theory*, J. Sci. Comput., 87 (2021), 70.

- [5] G. P. T. CHOI, B. CHIU, AND C. H. RYCROFT, *Area-preserving mapping of 3D carotid ultrasound images using density-equalizing reference map*, IEEE Trans. Biomed. Eng., 67 (2020), pp. 2507–2517.
- [6] G. P. T. CHOI, Y. LEUNG-LIU, X. GU, AND L. M. LUI, *Parallelizable global conformal parameterization of simply-connected surfaces via partial welding*, SIAM J. Imaging Sci., 13 (2020), pp. 1049–1083.
- [7] G. P. T. CHOI, Y. LIU, AND L. M. LUI, *Free-boundary conformal parameterization of point clouds*, J. Sci. Comput., 90 (2022), pp. 1–26.
- [8] G. P.-T. CHOI AND L. M. LUI, *A linear formulation for disk conformal parameterization of simply-connected open surfaces*, Adv. Comput. Math., 44 (2018), pp. 87–114.
- [9] G. P. T. CHOI AND L. M. LUI, *Recent developments of surface parameterization methods using quasi-conformal geometry*, in Handbook of Mathematical Models and Algorithms in Computer Vision and Imaging, Springer, Cham, Switzerland, 2023, pp. 1483–1523.
- [10] G. P.-T. CHOI, M. H.-Y. MAN, AND L. M. LUI, *Fast spherical quasiconformal parameterization of genus-0 closed surfaces with application to adaptive remeshing*, Geom. Imaging Comput., 3 (2016), pp. 1–29.
- [11] G. P. T. CHOI AND C. H. RYCROFT, *Density-equalizing maps for simply connected open surfaces*, SIAM J. Imaging Sci., 11 (2018), pp. 1134–1178.
- [12] G. P. T. CHOI AND C. H. RYCROFT, *Volumetric density-equalizing reference map with applications*, J. Sci. Comput., 86 (2021), 41.
- [13] P. T. CHOI, K. C. LAM, AND L. M. LUI, *FLASH: Fast landmark aligned spherical harmonic parameterization for genus-0 closed brain surfaces*, SIAM J. Imaging Sci., 8 (2015), pp. 67–94.
- [14] P. T. CHOI AND L. M. LUI, *Fast disk conformal parameterization of simply-connected open surfaces*, J. Sci. Comput., 65 (2015), pp. 1065–1090.
- [15] D. CROWDY, *Solving Problems in Multiply Connected Domains*, NSF-CBMS Regional Conf. Ser. Appl. Math., SIAM, Philadelphia, 2020.
- [16] L. CUI, X. QI, C. WEN, N. LEI, X. LI, M. ZHANG, AND X. GU, *Spherical optimal transportation*, Comput. Aided Design, 115 (2019), pp. 181–193.
- [17] M. DESBRUN, M. MEYER, AND P. ALLIEZ, *Intrinsic parameterizations of surface meshes*, Comput. Graph. Forum, 21 (2002), pp. 209–218.
- [18] D. DORLING, M. NEWMAN, AND A. BARFORD, *The Atlas of the Real World*, Thames & Hudson, London, 2008.
- [19] M. S. FLOATER AND K. HORMANN, *Surface parameterization: A tutorial and survey*, in Advances in Multiresolution for Geometric Modelling, Springer, Berlin, 2005, pp. 157–186.
- [20] M. T. GASTNER AND M. E. J. NEWMAN, *Diffusion-based method for producing density-equalizing maps*, Proc. Nat. Acad. Sci. USA, 101 (2004), pp. 7499–7504.
- [21] X. GU AND S.-T. YAU, *Global conformal surface parameterization*, in Proceedings of the 2003 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing, ACM, New York, 2003, pp. 127–137.
- [22] X. D. GU AND S.-T. YAU, *Computational Conformal Geometry*, Vol. 1, International Press, Somerville, MA, 2008.
- [23] Y. GUO, Q. CHEN, G. P. T. CHOI, AND L. M. LUI, *Automatic landmark detection and registration of brain cortical surfaces via quasi-conformal geometry and convolutional neural networks*, Comput. Biol. Med., (2023), 107185.
- [24] P. HENRICI, *Applied and Computational Complex Analysis, Volume 3: Discrete Fourier Analysis, Cauchy Integrals, Construction of Conformal Maps, Univalent Functions*, Vol. 41, Wiley, New York, 1993.
- [25] K. T. HO AND L. M. LUI, *QCMC: Quasi-conformal parameterizations for multiply-connected domains*, Adv. Comput. Math., 42 (2016), pp. 279–312.
- [26] K. HORMANN AND G. GREINER, *MIPS: An efficient global parametrization method*, in Curve and Surface Design, Saint-Malo, France, 1999, Vanderbilt, University Press, Nashville, TN, 2000, pp. 153–162.
- [27] K. HORMANN, B. LÉVY, AND A. SHEFFER, *Mesh parameterization: Theory and practice*, ACM SIGGRAPH 2007 Courses, ACM, New York, 2007.
- [28] Z. JIANG, S. SCHAEFER, AND D. PANOZZO, *Simplicial complex augmentation framework for bijective maps*, ACM Trans. Graphics, 36 (2017), 186.
- [29] P. KOEBE, *Über die konforme abbildung mehrfach zusammenhängender bereiche*, Jahresber. Dtsch. Math.-Ver., 19 (1910), pp. 339–348.

- [30] E. KROPF, X. YIN, S.-T. YAU, AND X. D. GU, *Conformal parameterization for multiply connected domains: Combining finite elements and complex analysis*, Eng. Comput., 30 (2014), pp. 441–455.
- [31] K. C. LAM AND L. M. LUI, *Landmark-and intensity-based registration with large deformations via quasi-conformal maps*, SIAM J. Imaging Sci., 7 (2014), pp. 2364–2392.
- [32] K. C. LAM AND L. M. LUI, *Optimized quasiconformal parameterization with user-defined area distortions*, Commun. Math. Sci., 15 (2017), pp. 2027–2054.
- [33] B. LÉVY, S. PETITJEAN, N. RAY, AND J. MAILLOT, *Least squares conformal maps for automatic texture atlas generation*, ACM Trans. Graph., 21 (2002), pp. 362–371.
- [34] X. LI, Y. BAO, X. GUO, M. JIN, X. GU, AND H. QIN, *Globally optimal surface mapping for surfaces with arbitrary topology*, IEEE Trans. Vision Comput. Graph., 14 (2008), pp. 805–819.
- [35] Y. LIPMAN, *Bounded distortion mapping spaces for triangular meshes*, ACM Trans. Graph., 31 (2012), 108.
- [36] L. M. LUI, K. C. LAM, T. W. WONG, AND X. GU, *Texture map and video compression using Beltrami representation*, SIAM J. Imaging Sci., 6 (2013), pp. 1880–1902.
- [37] L. M. LUI, S. THIRUVENKADAM, Y. WANG, P. M. THOMPSON, AND T. F. CHAN, *Optimized conformal surface registration with shape-based landmark matching*, SIAM J. Imaging Sci., 3 (2010), pp. 52–78.
- [38] L. M. LUI, Y. WANG, T. F. CHAN, AND P. THOMPSON, *Landmark constrained genus zero surface conformal mapping and its application to brain mapping research*, Appl. Numer. Math., 57 (2007), pp. 847–858.
- [39] L. M. LUI, T. W. WONG, W. ZENG, X. GU, P. M. THOMPSON, T. F. CHAN, AND S.-T. YAU, *Optimization of surface registrations using Beltrami holomorphic flow*, J. Sci. Comput., 50 (2012), pp. 557–585.
- [40] F. LUO, *Combinatorial Yamabe flow on surfaces*, Commun. Contemp. Math., 6 (2004), pp. 765–780.
- [41] Z. LYU, Q. CHEN, AND L. M. LUI, *A two-stage algorithm for combined quasiconformal and optimal mass transportation spherical parameterization*, Math. Comput. Geom. Data, 3 (2023), pp. 29–57.
- [42] P. MULLEN, Y. TONG, P. ALLIEZ, AND M. DESBRUN, *Spectral Conformal Parameterization*, Comput. Graph. Forum, 27 (2008), pp. 1487–1494.
- [43] M. M. S. NASSER, *PlgCirMap: A MATLAB toolbox for computing conformal mappings from polygonal multiply connected domains onto circular domains*, SoftwareX, 11 (2020), 100464.
- [44] P.-O. PERSSON AND G. STRANG, *A simple mesh generator in MATLAB*, SIAM Rev., 46 (2004), pp. 329–345.
- [45] D. QIU, K.-C. LAM, AND L.-M. LUI, *Computing quasi-conformal folds*, SIAM J. Imaging Sci., 12 (2019), pp. 1392–1424.
- [46] M. RABINOVICH, R. PORANNE, D. PANOZZO, AND O. SORKINE-HORNUNG, *Scalable locally injective mappings*, ACM Trans. Graph., 36 (2017), 1.
- [47] M. SHAQFA, G. P. T. CHOI, G. ANCIAUX, AND K. BEYER, *Disk Harmonics for Analysing Curved and Flat Self-Affine Rough Surfaces and the Topological Reconstruction of Open Surfaces*, preprint, [arXiv:2403.07001](https://arxiv.org/abs/2403.07001).
- [48] M. S. M. SHAQFA, *Geometrical Treatise on the Modelling of 3D Particulate Inclusion-Matrix Microstructures with an Application to Historical Stone Masonry Walls*, EPFL thesis, EPFL, Lausanne, Switzerland, 2022.
- [49] A. SHEFFER, E. PRAUN, K. ROSE, ET AL., *Mesh parameterization methods and their applications*, Found. Trends Comput. Graph. Vision, 2 (2007), pp. 105–171.
- [50] A. SHTENGEL, R. PORANNE, O. SORKINE-HORNUNG, S. Z. KOVALSKY, AND Y. LIPMAN, *Geometric optimization via composite majorization*, ACM Trans. Graph., 36 (2017), 38.
- [51] J. SMITH AND S. SCHAEFER, *Bijective parameterization with free boundaries*, ACM Trans. Graph., 34 (2015), 70.
- [52] Y. WANG, L. M. LUI, T. F. CHAN, AND P. M. THOMPSON, *Optimization of brain conformal mapping with landmarks*, in Medical Image Computing and Computer-Assisted Intervention–MICCAI 2005: 8th International Conference, Palm Springs, CA, 2005, Springer, Berlin, 2005, pp. 675–683.
- [53] Y. WANG, L. M. LUI, X. GU, K. M. HAYASHI, T. F. CHAN, A. W. TOGA, P. M. THOMPSON, AND S.-T. YAU, *Brain surface conformal parameterization using Riemann surface structure*, IEEE Trans. Med. Imaging, 26 (2007), pp. 853–865.

- [54] O. WEBER, A. MYLES, AND D. ZORIN, *Computing extremal quasiconformal maps*, Comput. Graph. Forum, 31 (2012), pp. 1679–1689.
- [55] M.-H. YUEH, *Theoretical foundation of the stretch energy minimization for area-preserving mappings*, SIAM J. Imaging Sci., 16 (2023), pp. 1142–1176.
- [56] M.-H. YUEH, W.-W. LIN, C.-T. WU, AND S.-T. YAU, *An efficient energy minimization for conformal parameterizations*, J. Sci. Comput., 73 (2017), pp. 203–227.
- [57] M.-H. YUEH, W.-W. LIN, C.-T. WU, AND S.-T. YAU, *A novel stretch energy minimization algorithm for equiareal parameterizations*, J. Sci. Comput., 78 (2019), pp. 1353–1386.
- [58] W. ZENG, F. LUO, S.-T. YAU, AND X. D. GU, *Surface quasi-conformal mapping by solving Beltrami equations*, in, Mathematics of Surfaces XIII: 13th IMA International Conference York, UK, 2009, Springer, Berlin, 2009, pp. 391–408.
- [59] W. ZENG, X. YIN, M. ZHANG, F. LUO, AND X. GU, *Generalized Koebe’s method for conformal mapping multiply connected domains*, in 2009 SIAM/ACM Joint Conference on Geometric and Physical Modeling, SIAM, Philadelphia, 2009, pp. 89–100.
- [60] D. ZHANG, G. P. T. CHOI, J. ZHANG, AND L. M. LUI, *A unifying framework for n -dimensional quasi-conformal mappings*, SIAM J. Imaging Sci., 15 (2022), pp. 960–988.
- [61] X. ZHAO, Z. SU, X. D. GU, A. KAUFMAN, J. SUN, J. GAO, AND F. LUO, *Area-preservation mapping using optimal mass transport*, IEEE Trans. Vis. Comput. Graph., 19 (2013), pp. 2838–2847.
- [62] Z. ZHU, G. P. T. CHOI, AND L. M. LUI, *Parallelizable global quasi-conformal parameterization of multiply connected surfaces via partial welding*, SIAM J. Imaging Sci., 15 (2022), pp. 1765–1807.
- [63] G. ZOU, J. HU, X. GU, AND J. HUA, *Authalic parameterization of general surfaces using Lie advection*, IEEE Trans. Vis. Comput. Graph., 17 (2011), pp. 2005–2014.