Optimal Surface Quadrilateral Mesh Generation

Zhou Zhao* Siyu Fang† Na Lei‡ Yuanpeng Liu* Yiming Zhu†
Chander Sadasivan * Apostolos Tassiopoulos* Shikui Chen* Xianfeng Gu*

Abstract

Structured mesh generation has fundamental importance, but controlling the qualities of structured meshes remains a challenge. This work proposes a rigorous and practical algorithm to generate quadrilateral meshes on topological poly-annuli with the least number of singularities. It is shown that each quad-mesh with $4k$ vertex degree induces a holomorphic differential. All the holomorphic differentials form a finite-dimensional linear space. A geometric distortion energy is proposed to measure the local area distortion. One can achieve quad-meshes as uniformly as possible by optimizing the distortion energy in the linear space. Experimental results show the proposed algorithm is able to improve the uniformity of the quad-meshes, and the method has the potential to be generalized to handle quad-meshes with other constraints.

1 Introduction

Structured mesh generation plays a fundamental role in CAE fields such as CAD, FEA, CFD, and FSI. These fields also play a critical role in medicine to help understand the etiology of various diseases, to facilitate prognosis, and, especially, to optimize medical devices for improved patient outcomes. In engineering practice, it is critical to control the quality of the generated meshes, such as conformality and uniformity. Intuitively speaking, the conformality criteria require each quad-face on the mesh to be similar to a planar square, and the uniformity criteria require that the sizes of the quad-faces be as uniform as possible. Despite years of intensive research, controlling the quality of structured meshes remains a great challenge.

1.1 Optimization in Quad-mesh Space Recently, Lei et al. [4, 18, 33] introduced a novel theoretic framework for surface quad-mesh generation, which bridges the quad-meshes with meromorphic quatic differentials on the Riemann surface, namely a meromorphic section of a special holomorphic line bundle, such that the singularities of the quad-mesh are governed by the Abel-Jacobi theorem. This work can generate quad-meshes with high conformality. Furthermore, this framework shows that quad-meshes satisfying special constraints form a finite-dimensional linear space. Therefore, it is possible to optimize the mesh quality within this space using the standard energy minimization method. This work focuses on the simplest situation, the quad-meshes with the minimal number of singularities, namely the valence of all vertices of the quad-mesh are $4k$, $k \geq 1$ is a positive integer. In this situation, each quad-mesh induces a holomorphic differential (one-form).

Theorem 1.1. *Suppose $(S, \mathbf{g})$ is an oriented, closed surface with an Riemannian metric $\mathbf{g}$, Q is a quadrilateral mesh of S with genus $g > 1$, if all vertices have valence $4k$, $k \in \mathbb{Z}^+$, then Q induces a Riemann surface $\mathcal{R}_Q$, and a holomorphic one-form ω on the $\mathcal{R}_Q$.*

Proof. Given a quad-mesh Q on the surface, if we treat each face as the planar unit square, then Q induces a Riemannian metric $\mathbf{g}_Q$ on the surface, and a conformal structure of the surface as shown in the proofs of the theorems 4.6 and 4.7 in [18].

According to theorem 4.7 in [18], the quad-mesh Q induces a mermorphic quartic differential ω_Q. Suppose the singularities of Q are $p_1, p_2, \ldots, p_n$ with valences $4k_1, 4k_2, \ldots, 4k_n$ respectively, where $k_i \geq 2$ for $i = 1, 2, \ldots, n$. Then according to the theorem 4.11 in [18], the divisor of ω_Q is

$$(\omega_Q) = \sum_{i=1}^{n} (4k_i - 4)p_i.$$

Suppose ω_0 is an arbitrary holomorphic one-form on $\mathcal{R}_Q$, by Abel-Jacobi condition

$$\begin{aligned}(\omega_Q) - 4(\omega_0) &= \sum_{i=1}^{n} 4(k_i - 1)p_i - 4(\omega_0) \\ &= 4\left(\sum_{i=1}^{n}(k_i - 1)p_i - (\omega_0)\right) \\ &= 0,\end{aligned}$$

*Stony Brook University.
†Corresponding author, siyu.fang@stonybrook.edu, Stony Brook University.
‡Dalian University of Technology.

this shows there is a meromorphic 1-form ω'_Q whose divisor is

$$(\omega'_Q) = \sum_{i=1}^{n} (k_i - 1) p_i.$$

Since $k_i - 1 > 0$, hence all the singularities of ω'_Q are zeros, no poles, ω'_Q is a holomorphic one-form. ∎

According to the Riemann-Roch theorem [5], all holomorphic 1-forms form a g complex dimensional linear space on a closed genus g Riemann surface $\mathcal{R}$, denoted as $\Omega^1(\mathcal{R})$. Suppose $\{\omega_1, \omega_2, \ldots, \omega_g\}$ form a basis of $\Omega^1(\mathcal{R})$, any holomorphic one-form $\omega \in \Omega^1(\mathcal{R})$ is a complex linear combination of the bases,

$$\omega = \lambda_1 \omega_1 + \lambda_2 \omega_2 + \cdots + \lambda_g \omega_g.$$

The holomorphic 1-form ω induces a flat metric with cone singularities, denoted as $\mathbf{g}_\lambda$. We can define an energy to measure the distance between $\mathbf{g}_\lambda$ and the original metric $\mathbf{g}$ then find the optimal $(\lambda_1, \lambda_2, \ldots, \lambda_g)$ to minimize the energy. For example, let $\mathcal{T}$ be a triangulation of the input surface $(S, \mathbf{g})$, and we define the energy to measure the area distortions of the faces of $\mathcal{T}$,

$$\min_\lambda \sum_{\Delta \in \mathcal{T}} \left(|\Delta|_{\mathbf{g}} - |\Delta|_{\mathbf{g}_\lambda} \right)^2,$$

where Δ represents a triangular face in $\mathcal{T}$ and $|\Delta|_{\mathbf{g}}$ is the area of Δ under the metric $\mathbf{g}$.

1.2 Holomorphic Differential Basis In order to construct a basis of the space of all holomorphic differentials, we use the Hodge decomposition method [7]. Given a closed Riemannian surface $(S, \mathbf{g})$, we first compute the basis of the surface homology group $H_1(S, \mathbb{Z})$, $\{\gamma_1, \gamma_2, \ldots, \gamma_{2g}\}$; then dual basis of the cohomology group $H^1(S, \mathbb{Z})$, $\{eta_1, \eta_2, \ldots, \eta_{2g}\}$; thirdly, the basis of harmonic differential group $H_\Delta(S, \mathbb{R})$, $\{\omega_1, \omega_2, \ldots, \omega_{2g}\}$, such that ω_i is in the cohomology class of η_i, $\omega_i \in [\eta_i]$; finally, for each harmonic 1-form ω_i, we compute its conjugate harmonic 1-form ${}^\star\omega_i$, and pair them to form a holomorphic 1-form $\varphi_i = \omega_i + \sqrt{-1}{}^\star\omega_i$, $\{\varphi_1, \varphi_2, \ldots, \varphi_{2g}\}$ form a basis of $\Omega^1(S)$.

For surfaces with boundary components, the computational algorithm is more complicated. For the current work, we focus on an oriented, genus zero surface with multiple boundary components, namely topological poly-annulus S with boundary

$$\partial S = \gamma_0 - \gamma_1 - \cdots - \gamma_n,$$

where γ_0 is the exterior boundary component, γ_i, $i = 1, 2, \ldots, n$ are the interior ones. For harmonic 1-forms, we need to compute both exact harmonic 1-forms and non-exact harmonic 1-forms. For each interior boundary component γ_i, we compute a harmonic function $f_i : S \to \mathbb{R}$ with Dirichlet boundary condition, such that the restriction of f_i on γ_i is 1, and the restrictions on other γ_j's are 0. Then df_i gives us the exact harmonic 1-forms. Suppose the basis of $H^1(S, \mathbb{R})$ are $\{\eta_1, \eta_2, \ldots, \eta_n\}$, such that $\int_{\gamma_i} \eta_j = \delta_{ij}$. For each η_i we find a function $u_i : S \to \mathbb{R}$, such that $\Delta u_i = \delta \eta_i$ with Neumann boundary condition. Then $\omega_i = \eta_i + du_i$ is the non-exact harmonic 1-form. The conjugate harmonic 1-form ${}^\star\omega_i$ is also harmonic, conventional algorithm represent

$${}^\star\omega_i = \sum_{j=1}^{n} \lambda_i df_i + \sum_{k=1}^{n} \mu_k \omega_k.$$

We observe that the conjugate harmonic 1-form ${}^\star\omega_i$ is exact, therefore

$${}^\star\omega_i = \sum_{j=1}^{n} \lambda_i df_i,$$

this reduces the computational complexity by half.

THEOREM 1.2. *Suppose $(S, \mathbf{g})$ is a poly-annulus, $\partial S = \gamma_0 - \gamma_1 - \ldots \gamma_n$, where γ_0 is the exterior boundary component, γ_i's are interior boundary components, $i = 1, 2, \ldots, n$, suppose f_i's are harmonic functions with Dirichlet boundary condition $f_i|_{\gamma_j} = \delta_{ij}$; ω_i's are non-exact harmnoic 1-forms with Neumann boundary condition, and $\int_{\gamma_j} \omega_i = \delta_{ij}$. Then the harmonic 1-form ${}^\star\omega_i$ conjugate to ω_i is exact, therefore*

$${}^\star\omega_i = \lambda_1 df_1 + \lambda_2 df_2 + \cdots + \lambda_n df_n, \quad \lambda_i \in \mathbb{R}.$$

Proof. By our construction, ω_i is dual to a harmonic tangential vector field v_i, ${}^\star\omega_i$ is dual to ${}^\star v_i$, where

$${}^\star v_i(p) = n(p) \times v_i(p),$$

where $n(p)$ is normal to the surface at p. Along the boundary γ_j, the vector field v_i is parallel to the tangent direction of γ_j, hence ${}^\star v_i$ is orthogonal to the tangent direction of γ_j, therefore

$$\int_{\gamma_j} {}^\star v_i = 0, \quad 1 \le i, j \le n.$$

Since ω_i is harmonic, ${}^\star\omega_i$ is harmonic, therefore ${}^\star\omega_i$ is closed. The above equation shows ${}^\star\omega_i$ is an exact harmonic 1-form, hence can be represented as the linear combination of df_i's. ∎

1.3 Contributions This work proposes a novel method for generating quadrilateral meshes that are as uniform as possible on topological poly-annulus surfaces with the least number of singularities. In detail:

- Propose a framework that formulates the problem of controlling quad-mesh qualities as an optimization problem in a finite-dimensional linear subspace of the meromorphic quadratic differentials;
- Prove in theorem 1.1 that quadrilateral meshes with $4k$ degree vertices, $k \in \mathbb{Z}_+$ induce harmonic 1-forms;
- Prove in theorem 1.2 that the conjugate harmonic 1-form of non-exact harmonic 1-form on a poly-annulus must be exact, this reduces the computational complexity of Hodge star operator by half;
- Propose a quartic polynomial energy for optimization to generate the quad-mesh as uniform as possible.

2 Previous Work

Quadrilateral mesh generation holds a significant position in the fields of science and engineering owing to its attractive properties, including tensor-product nature and smooth surface approximation. There is a plethora of literature on quad meshing. For a more comprehensive and thorough literature review, we recommend readers to refer to [2]. In the following, we only review the main methods of quadrilateral mesh generation.

Converting Triangulation Method The first commonly used method is converting a triangle mesh into a quad mesh. The process generally includes steps such as edge matching, vertex insertion, and optimization to achieve desired quad mesh properties. The simplest way is that two neighboring triangles can be combined into a single quadrilateral, resulting in the formation of a quad mesh [8, 25, 28, 30]. This method can only produce unstructured quad-meshes, and the disadvantages of converting a triangle mesh to a quad mesh include complexity, potential quality degradation, loss of geometric detail, challenges with boundaries, and irregular shapes.

Patch-based Method Another method is the patch-based approach. This method involves dividing data or an area into smaller segments or patches for analysis and processing, commonly used in fields like image processing, computer graphics, and machine learning to manage complex data efficiently. The clustering method used to create the skeleton involves merging neighboring triangle faces into patches, employing techniques like normal-based and center-based methods [1, 3]. The computation of these patches is facilitated through the use of poly-cube maps [32, 31, 21, 9].

Parameterization Based Method Another common method is the quad-meshing algorithm based on parameterization. The Parameterization-Based Approach is a methodology that relies on creating parameterizations or mappings of geometric data to facilitate various tasks in computer graphics, computer-aided design, and related fields. The spectral surface quadrangulation method [6, 10] is applied to the input mesh. Techniques like global conformal parameterization [7], discrete harmonic forms [29], periodic global parameterization [23], branched covering methods [13] and discrete surface Ricci flow [12, 26, 27], all utilize parameterization as a foundational element for quad mesh generation.

Frame Field Method In addition to these, one of the most popular approaches is cross-field guided quad-mesh generation. In this method, cross fields are used to guide the generation of quadrilateral (quad) meshes on complex surfaces. By aligning or orienting the quads along the directions provided by the cross fields, it becomes possible to create structured quad meshes, which are valuable in various applications like finite element analysis, 3D modeling, and simulation. Each approach needs to initially decide on a method for representing a cross. Some examples are: N-RoSy representation [22, 15], period jump technique [20] and complex value representation [14]. Subsequently, these approaches commonly create a continuous and smooth cross field using energy minimization techniques. Field smoothness is typically measured using a discrete form of the Dirichlet energy [11]. Finally, relying on the established cross field, these methods produce quad meshes by employing either streamline tracing techniques [24] or parameterization methods [2]. Cross-field-based methods offer structured quad mesh generation and improved alignment but can be computationally complex, algorithm-sensitive, and may require manual intervention with output quality dependent on the input. Lei et al. [19] proved the sufficient and necessary conditions for the existence of a cross field in terms of singularity configurations. Furthermore, they pointed out that cross fields are not equivalent to quad-meshes and gave an explanation using fiber bundle theory.

Method Based On Abel-Jacobi Theory Chen et al. [4] gave the sufficient and necessary conditions for a Riemannian metric induced by a quad-mesh, including the Gauss-Bonnet condition for the curvatures, the holonomy condition, boundary alignment condition and the finite streamline condition. Lei et al. [18] proved that the holonomy condition can be formulated using the Abel-Jacobi equation in algebraic geometry. Zheng et al. [33] gave a practical algorithm to optimize the singularity configurations to satisfy the Abel-Jacobi condition. These works show that the surface quad-meshes are equivalent to meromorphic quartic differentials and can be treated as a meromorphic global section of a special holomorphic line bundle on the Riemann surface, and the singularities of the quad-meshes are the characteristic class of the line bundle. Therefore, the singularities and their indexes are governed by the Abel-Jacobi equations. These works laid down the theoretic foundation for structured mesh generation. Furthermore, Lei et al. [16, 17] also proposed to generalize the method to hexahedral mesh generation based on surface foliations, which are special cases of meromorphic quartic differentials.

Our current work is mainly based on this framework - the Abel-Jacobi theorem gives the conditions of the configurations of singularities of quad-meshes. Furthermore, given special constraints, the solutions to the Abel-Jacobi equations are not unique but form a finite-dimensional space. We propose a method to perform optimization within the solution space to improve the quad-mesh quality.

3 Theoretic Background

In this section, we briefly introduce the basic concepts and theorems in conformal geometry related to the holomorphic differential.

3.1 Riemann Surface

Definition 3.0.1. (Holomorphic Function) *Suppose a complex function $f : \mathbb{C} \to \mathbb{C}$, $f(x+iy) = u+iv$, satisfying the Cauchy-Riemann equation*

$$\frac{\partial u}{\partial x} = \frac{\partial v}{\partial y}, \quad \frac{\partial u}{\partial y} = -\frac{\partial v}{\partial x},$$

then f is called holomorphic. If f is invertible and the inverse f^{-1} is also holomorphic, then f is called biholomorphic.

Definition 3.0.2. (Riemann Surface) *Suppose S is a topological surface with an open covering $S \subset \bigcup_\alpha U_\alpha$, each open set has a local coordinates system $\varphi_\alpha : U_\alpha \to \mathbb{C}$, if $U_\alpha \cap U_\beta \neq \emptyset$, then the transition map*

$$\varphi_{\alpha\beta} : \varphi_\alpha(U_\alpha \cap U_\beta) \to \varphi_\beta(U_\alpha \cap U_\beta), \varphi_{\alpha\beta} = \varphi_\beta \circ \varphi_\alpha$$

is biholomorphic. Then the atlas $\mathcal{A} = \{(U_\alpha, \varphi_\alpha)\}$ is called a complex structure of the surface, and the surface S with the complex structure is called a Riemann surface.

Definition 3.0.3. (Isothermal Coordinates) *Suppose S is a topological surface with a Riemannian metric $\mathbf{g}$, and U is a neighborhood $U \subset S$ with the local coordinates (u,v), such that the Riemannian metric has a special form*

$$\mathbf{g}(u,v) = e^{2\lambda(u,v)}(du^2 + dv^2),$$

then (u,v) are called the isothermal coordinates *on U and $\lambda : U \to \mathbb{R}$ is called the* conformal factor *function.*

According to classical surface differential geometry, all oriented metric surfaces are Riemann surfaces.

Theorem 3.1. *Suppose S is an oriented surface with a Riemannian metric $\mathbf{g}$, then for each point $p \in S$, there is a neighborhood $p \in U(p)$, such that $U(p)$ has an isothermal coordinates system. All such isothermal coordinate systems form a complex structure. Hence, S is a Riemann surface.*

3.2 Hodge Theorem Suppose S is a surface with a conformal structure $\{U_\alpha, z_\alpha\}$. A real differential 0-form is a function $f : S \to \mathbb{R}$; a differential 1-form has local representation,

$$\omega = f_\alpha(x_\alpha, y_\alpha)dx_\alpha + g_\alpha(x_\alpha, y_\alpha)dy_\alpha,$$

on the local chart (U_β, z_β), it has representation $f_\beta dx_\beta + g_\beta dy_\beta$, satisfying

$$\begin{pmatrix} f_\alpha & g_\alpha \end{pmatrix} \begin{pmatrix} dx_\alpha \\ dy_\alpha \end{pmatrix} = \begin{pmatrix} f_\beta & g_\beta \end{pmatrix} \begin{pmatrix} \frac{\partial x_\beta}{\partial x_\alpha} & \frac{\partial x_\beta}{\partial y_\alpha} \\ \frac{\partial y_\beta}{\partial x_\alpha} & \frac{\partial y_\beta}{\partial y_\alpha} \end{pmatrix} \begin{pmatrix} dx_\alpha \\ dy_\alpha \end{pmatrix}$$

therefore

$$\begin{pmatrix} f_\alpha & g_\alpha \end{pmatrix} = \begin{pmatrix} f_\beta & g_\beta \end{pmatrix} D\varphi_{\alpha\beta}$$

where $D\varphi_{\alpha\beta}$ is the Jacobian matrix of the transition map $\varphi_{\alpha\beta}$. The differential 2-form has local representation,

$$h_\alpha(x_\alpha, y_\alpha)dx_\alpha \wedge dy_\alpha = h_\beta(x_\beta, y_\beta)dx_\beta \wedge dy_\beta,$$

$h_\alpha = h_\beta \det D\varphi_{\alpha\beta}$. The *exterior differential* operator d is defined as follows: for a 0-form f,

$$d^0 f_\alpha(x_\alpha, y_\alpha) := \frac{\partial f_\alpha}{\partial x_\alpha}dx_\alpha + \frac{\partial f_\alpha}{\partial y_\alpha}dy_\alpha,$$

for a 1-form ω,

$$d^1\omega = \left(\frac{\partial g}{\partial x_\alpha} - \frac{\partial f}{\partial y_\alpha}\right)dx_\alpha \wedge dy_\alpha.$$

for a 2-form,

$$d^2 h_\alpha dx_\alpha \wedge dy_\alpha = 0.$$

The k-th de Rham cohomology group is defined as:

$$H^k_{dR}(S, \mathbb{R}) := \frac{\text{Ker}d^k}{\text{Img}d^{k-1}}.$$

For a genus g closed surface, its first dimensional cohomology group $H^1_{dR}(S,\mathbb{R})$ is $2g$ dimensional.

On a metric surface $(S,\mathbf{g})$ with isothermal coordinates, the *Hodge star* operator is given by

$$\begin{aligned} {}^*dx_\alpha &= dy_\alpha, {}^*dy_\alpha = -dx_\alpha, \\ {}^*1 &= e^{2\lambda}dx_\alpha \wedge dy_\alpha, {}^*e^{2\lambda}dx_\alpha \wedge dy_\alpha = 1. \end{aligned}$$

The co-differential operator is defined as $\delta := {}^*d^*$. The Laplace-Beltrami operator is defined as $\Delta_\mathbf{g} = d\delta + \delta d$. For 0-forms, the operator has local representation:

$$\Delta_\mathbf{g} := \frac{1}{e^{2\lambda}}\left(\frac{\partial^2}{\partial x_\alpha^2} + \frac{\partial^2}{\partial y_\alpha^2}\right).$$

Definition 3.1.1. (Harmonic Differential) *Suppose ω is a differential k-form on $(S,\mathbf{g})$, then ω is* harmonic *if and only if $\Delta_\mathbf{g}\omega = 0$.*

If ω is a differential 1-form, then ω is harmonic if and only if

$$d\omega = 0, \quad \delta\omega = 0.$$

Theorem 3.2. (Hodge) *Each de Rham cohomology class has a unique harmonic form.*

The group of all harmonic k-forms is denoted as $H^k_\Delta(S,\mathbb{R})$, according to Hodge theory, the harmonic k-form group is isomorphic to the k-dimensional de Rham cohomology group,

$$H^k_\Delta(S,\mathbb{R}) \cong H^k_{dR}(S,\mathbb{R}).$$

Suppose ω is a harmonic 1-form, then ${}^*\omega$ is also a harmonic 1-form.

3.3 Holomorphic Differential

Definition 3.2.1. (Holomorphic Differential) *Suppose φ is a complex different form, on each local chart (U_α, z_α), φ has local representation $\varphi = f_\alpha(z_\alpha)dz_\alpha$, where f_α is a holomorphic function. On another intersecting chart (U_β, z_β), $U_\alpha \cap U_\beta \neq \emptyset$, $\varphi = f_\beta(z_\beta)dz_\beta$, such that*

$$f_\alpha = f_\beta(z_\beta(z_\alpha))\frac{dz_\beta}{dz_\alpha},$$

then φ is globally defined, and called a holomorphic differential.

Suppose at a point $p \in S$, $\varphi(p) = 0$ (locally, $f_\alpha(p) = 0$, for any local chart (U_α, z_α)), then p is called a *zero* of the holomorphic 1-form φ. The local representation of φ in the neighborhood of a zero point p is

$$\varphi = z_\alpha^{n_p} dz_\alpha,$$

where $z_\alpha(p) = 0$. n_p is called the *order* of φ at p, denoted as $\mu_p(\varphi)$. The total order of zeros equals the Euler characteristic number:

$$\sum_{\varphi(p)=0} \mu_p(\varphi) = \chi(S).$$

In general, there are $2g-2$ zeros for a holomorphic 1-form.

Each holomorphic 1-form φ can be decomposed into two conjugate harmonic 1-forms, namely,

$$\varphi = \omega + \sqrt{-1}{}^*\omega,$$

where ω is a real harmonic differential form.

Given a holomorphic 1-form φ, for any point on the surface $p \in S$, a tangent direction $v \in T_pS$ is called a *horizontal direction*, if $\varphi(v) \in \mathbb{R}$ is a real number; similarly if $\varphi(v)$ is an imaginary number, then v is called a vertical direction. A curve γ is called a *horizontal trajectory* of φ, if its tangent directions are horizontal everywhere except at the zeros of φ. The vertical trajectories of φ are defined similarly.

For any non-zero point $p \in S$ of φ, there is a unique horizontal (vertical) trajectory through it. For a zero point p with order n_p, there are $n_p + 1$ horizontal (vertical) trajectories through it. Such a kind of horizontal trajectory is called a *critical horizontal trajectory.*

4 Computational Algorithm

Data Structure The surfaces are represented as triangle meshes (simplicial complex) $M(V,E,F)$, where V,E,F are the set of vertices, edges and faces respectively. We use $[v_0, v_1, \ldots, v_n]$ to represent a simplex with vertices $v_0, v_1, \ldots, v_n$, the orientation of the simplex is given by the order of the vertices. A k-chain is a linear combination of k-dimensional simplexes,

$$\sigma = \sum_i \lambda_i \sigma_i, \quad \lambda_i \in \mathbb{Z},$$

σ_i is a k-dimensional simplex. The linear space of all k-dimensional chains is denoted as $C_k(M,\mathbb{Z})$. The boundary operator $\partial_k : C_k(M,\mathbb{Z}) \to C_{k-1}(M,\mathbb{Z})$ is defined as

$$\partial_k[v_0, v_1, \ldots, v_k] := \sum_{i=0}^k (-1)^i [v_0, \ldots, v_{i-1}, v_{i+1}, \ldots, v_k].$$

A k-dimensional simplicial σ is a linear map $\sigma : C_k(M,\mathbb{Z}) \to \mathbb{Z}$. The k-dimensional co-chain space

$C^k(M,\mathbb{Z})$ consists of all the k-dimensional co-chains. The coboundary operator $d_k : C^k(M,\mathbb{Z}) \to C^{k+1}(M,\mathbb{Z})$ is defined as the dual operator of ∂_{k+1}. For example, suppose ω is a 1-form, σ is a 2-chain, then

$$d\omega(\sigma) = \omega(\partial\sigma).$$

The simplicial integration is calculated using summation. Suppose given a 1-chain γ, $\gamma = \sum_k [v_{i_k}, v_{i_{k+1}}]$, then

$$\int_\gamma \omega = \sum_k \omega([v_{i_k}, v_{i_{k+1}}]).$$

The wedge product of two simplicial 1-forms ω_1, ω_2 on a 2-simplex σ is given by the following formula:

$$\omega_1 \wedge \omega_2(\sigma) = \frac{1}{6}\begin{vmatrix} \omega_1(e_i) & \omega_1(e_j) & \omega_1(e_k) \\ \omega_2(e_i) & \omega_2(e_j) & \omega_2(e_k) \\ 1 & 1 & 1 \end{vmatrix} \tag{4.1}$$

where σ is a face with three edges e_i, e_j, e_k sorted counter-clock-wisely.

The triangulation of M is dual to a cell decomposition $M^\star$ as follows:

1. Each face $f \in M$ is dual to a vertex $f^\star \in {}^\star M$, $f^\star$ is the circum-center of f;

2. Each edge $e \in M$ shared by two faces (from right to left) $f_i, f_j \in M$ is adjacent to an edge $e^\star = [f_i^\star, f_j{}^\star]$;

3. Each vertex $v \in M$ is adjacent to faces $f_0, f_1, \ldots, f_k \in M$, ordered counter-clock-wisely, dual to a cell

$$v^\star = [f_0^\star, f_1^\star, \ldots, f_k^\star].$$

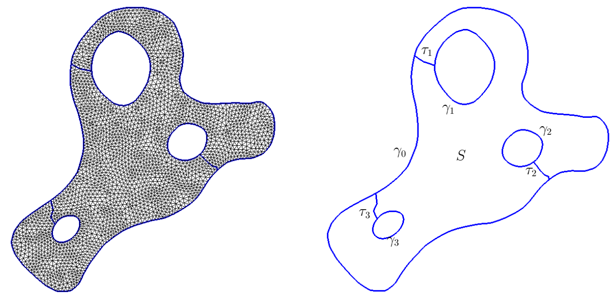
The Hodge star operator $^\star : C^k(M,\mathbb{Z} \to C^{2-k}(\bar{M},\mathbb{Z}$ is defined as follows, suppose ω is a k-form, σ is a k-simplex, then

$$\frac{{}^\star\omega(\sigma^\star)}{|\sigma^\star|} = \frac{\omega(\sigma)}{|\sigma|}.$$

Given two simplicial 1-forms ω_1 and ω_2, the 2-form $\omega_1 \wedge {}^*\omega_2$ on each face $\sigma = [v_i, v_j, v_k]$ is evaluated as

$$\begin{aligned} \omega_1 \wedge {}^*\omega_2(\sigma) = \frac{1}{2}[&\cot\theta_i \omega_1(e_i)\omega_2(e_i) + \\ &\cot\theta_j \omega_1(e_j)\omega_2(e_j) + \\ &\cot\theta_k \omega_1(e_k)\omega_2(e_k)] \end{aligned} \tag{4.2}$$

In the current project, all the differential forms are approximated as simplicial forms (simplicial co-chains). Given a 0-form $f : V \to \mathbb{R}$, the value of f at a vertex v_i, $f(v_i)$ is stored on the vertex; similarly, given a 1-form $\omega : E \to \mathbb{R}$, the value on each oriented edge e, $\omega(e)$ is stored on e. For a complex 1-form $\varphi : E \to \mathbb{C}$, the value on an oriented edge e, $\varphi(e)$ is stored on e.



(a). input mesh (b). cuts

Figure 1: The input mesh and the shortest paths from interior boundaries to the exterior boundary.

Input The input is a genus zero surface S with multiple boundaries, represented as a triangle mesh,

$$\partial S = \gamma_0 - \gamma_1 - \gamma_2 - \cdots - \gamma_n,$$

where γ_0 is the exterior boundary component, γ_i, $i = 1, 2, \ldots, n$, are interior boundary components, as shown in the Fig. 1 frame (a).

Cuts For each interior boundary component γ_i, $i = 1, \ldots, n$, we compute the shortest path τ_i from γ_i to the exterior boundary component γ_0. Basically, for each vertex $v_k \in \gamma_i$, we use breadth-first search to traverse the triangle mesh and find the shortest path to γ_0, then we choose the one with the minimal length as τ_i. Fig. 1 frame (b) shows the shortest paths (cuts) computed this way on the input mesh.

Exact Harmonic Forms For each interior boundary component γ_i, $i = 1, \ldots, n$, we compute a unique harmonic function f_i with Dirichlet boundary condition, the restriction of f_i on γ_i equals to 1, the restriction of f_i on other boundary components equal to zero. This boils down to solving the Laplace-Beltrami equation: for $i = 1, 2, \cdots, n$,

$$\left\{ \begin{array}{lcll} \Delta_{\mathbf{g}} f_i & = & 0 & \\ f_i|_{\gamma_i} & = & 1 & \\ f_i|_{\gamma_j} & = & 0 & j \neq i \end{array} \right. \tag{4.3}$$

The equation $\Delta_{\mathbf{g}} f_i = 0$ is equivalent to $\delta d f_i = 0$. By using the Finite Element Method, this equation is converted to a large sparse linear system.

For each edge $e_{ij} = [v_i, v_j]$, we compute the *cotangent edge weight* w_{ij} as follows: suppose e_{ij} is shared by two faces $[v_i, v_j, v_k]$ and $[v_j, v_i, v_l]$ the corner angles against e_{ij} are θ_k^{ij} and θ_l^{ji}, then the edge weight is the summation of the cotangent of both corner angles; if e_{ij} is on the surface boundary, and only adjacent to one face $[v_i, v_j, v_k]$, then the edge weight equals to cotangent of θ_k^{ij},

$$w_{ij} := \begin{cases} \cot\theta_k^{ij} + \cot\theta_l^{ji} & e_{ij} \notin \partial S \\ \cot\theta_k^{ij} & e_{ij} \in \partial S \end{cases}$$

For each interior vertex $v_j \notin \partial S$, we have a linear equation $\delta f_i(v_j) = 0$,

$$\delta f_i(v_j) = \sum_{v_k \sim v_j} w_{jk}(f_i(v_k) - f_i(v_j)) = 0,$$

where $v_k \sim v_j$ means the vertex v_k is connected with v_j via e_{jk}. For boundary vertex $v_j \in \partial S$, $f_i(v_j)$ is fixed. The stiffness matrix of the linear system is positive definite. The linear system can be solved using conjugate gradient method efficiently. Fig. 2 left columns visualize n exact harmonic 1-forms by using texture mapping.

Non-exact Harmonic Forms For each interior boundary component γ_i, $i = 1, 2, \ldots, n$, we compute a non-exact harmonic form ω_i, such that

$$\int_{\gamma_j} \omega_i = \delta_{ij}, \quad i, j = 1, 2, \ldots, n.$$

For each interior boundary component $\gamma_i \in \partial S$, we slice the surface S along the shortest cut τ_i to obtain $\bar{S}_i$, τ_i is split into two boundary components $\tau_i^+, \tau_i^- \in \partial\bar{S}_i$. We then define a function $g_i : \bar{S}_i \to \mathbb{R}$,

$$g_i(v_k) = \begin{cases} 1 & v_k \in \tau_i^+ \\ 0 & v_k \in \tau_i^- \\ \text{rand} & \text{otherwise} \end{cases} \tag{4.4}$$

For each edge $e \in \tau_i$, the corresponding edges $e^+ \in \tau_i^+$ and $e^- \in \tau_i^-$, and

$$dg_i(e_i^+) = dg_i(e_i^-) = 0,$$

hence dg_i is a 1-form defined on the original surface S, denoted as η_i. By the construction η_i is closed, and

$$\int_{\gamma_j} \eta_i = \delta_{ij},$$

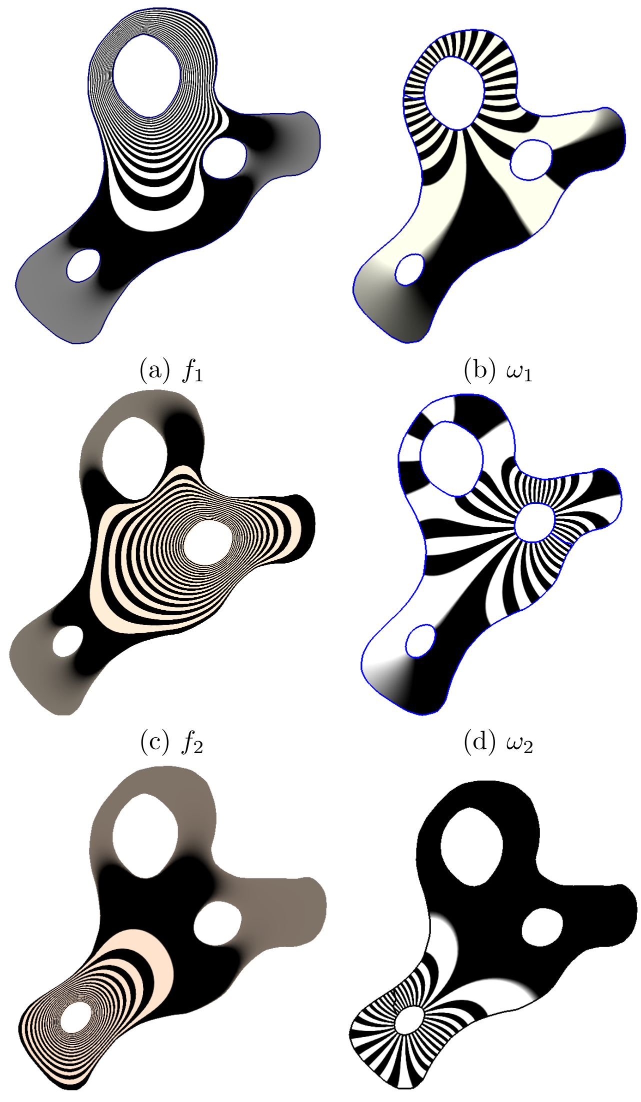
hence $\{\eta_1, \eta_2, \ldots, \eta_n\}$ form a basis of $H^1(S, \mathbb{R})$.

According to Hodge theorem 3.2, for each cohomology class, there is a unique harmonic form. We can find a function $h_i : S \to \mathbb{R}$, such that $\omega_i = \eta_i + dh_i$ is harmonic, and satisfies the equation $\Delta_{\mathbf{g}}\omega_i$ is zero, namely $d\omega_i = 0$ and $\delta\omega_i = 0$. By the construction of ω_i, the first closedness condition is satisfied automatically. We only need to ensure the second condition. This gives us the following equation:

$$\delta dh_i = \Delta_{\mathbf{g}h_i} = -\delta\omega_i, \tag{4.5}$$

with the Neumann boundary condition $\partial h_i/\partial n = 0$, where n is the exterior normal on the surface boundary. Namely, for each vertex $v_j \in S$, we have

$$\sum_{v_k \sim v_j} w_{jk}(h_i(v_k) - h_i(v_j)) = -\sum_{v_k \sim v_j} \eta_i([v_j, v_k]).$$



(a) f_1 (b) ω_1 (c) f_2 (d) ω_2 (e) f_3 (f) ω_3

Figure 2: Exact (left) and non-exact (right) harmonic differentials.

The linear coefficient matrix is positive definite on the linear subspace $\sum_k h_i(v_k) = 0$, therefore, it can be efficiently solved using the conjugate gradient method. The solutions give the non-exact harmonic 1-forms $\omega_i = \eta_i + dh_i$. Fig. 2 right columns visualize n non-exact harmonic 1-forms by using texture mapping.

Conjugate Harmonic Forms So far, we have computed n exact harmonic 1-forms $\{df_1, df_2, \cdots, df_n\}$ and n non-exact harmonic 1-forms $\{\omega_1, \omega_2, \cdots, \omega_n\}$, then we compute the conjugate their harmonic 1-forms. In theory, the conjugate harmonic 1-form of a non-exact harmonic 1-form is an exact harmonic 1-form, and the conjugate of an exact harmonic 1-form is a non-exact harmonic 1-form. Hence, we can construct the equations:

$$(4.6) \qquad {}^\star\omega_i = \lambda_{i1} df_1 + \lambda_{i2} df_2 + \cdots + \lambda_{in} df_n$$

therefore, we obtain the linear equation group:

$$(4.7) \qquad \begin{cases} \int_S \omega_1 \wedge {}^\star\omega_i = \lambda_{i1}\omega_1 \wedge df_1 + \cdots \lambda_{in}\omega_1 \wedge df_n \\ \int_S \omega_2 \wedge {}^\star\omega_i = \lambda_{i2}\omega_2 \wedge df_1 + \cdots \lambda_{in}\omega_2 \wedge df_n \\ \qquad \cdots\cdots\cdots \\ \int_S \omega_n \wedge {}^\star\omega_i = \lambda_{i1}\omega_n \wedge df_1 + \cdots \lambda_{in}\omega_n \wedge df_n \end{cases}$$

The left-hand side of Eqn. 4.7 can be calculated using the formula in Eqn. 4.2, the right-hand side can be evaluated using the formula in Eqn. 4.1. There are n equations for n unknowns, and the linear system is non-degenerated, so we can solve the coefficients λ_{ij}'s and obtain the conjugate harmonic 1-form ${}^\star\omega_i$. Then we obtain n holomorphic 1-forms:

$$(4.8) \qquad \varphi_i = \omega_i + \sqrt{-1}{}^\star\omega_i, \quad i = 1, 2, \ldots, n.$$

Fig. 3 show the basis of the holomorphic 1-form group φ_1, φ_2 and φ_3. The frame (d) shows the linear combination $\varphi_1 - \varphi_2 + \varphi_3$.

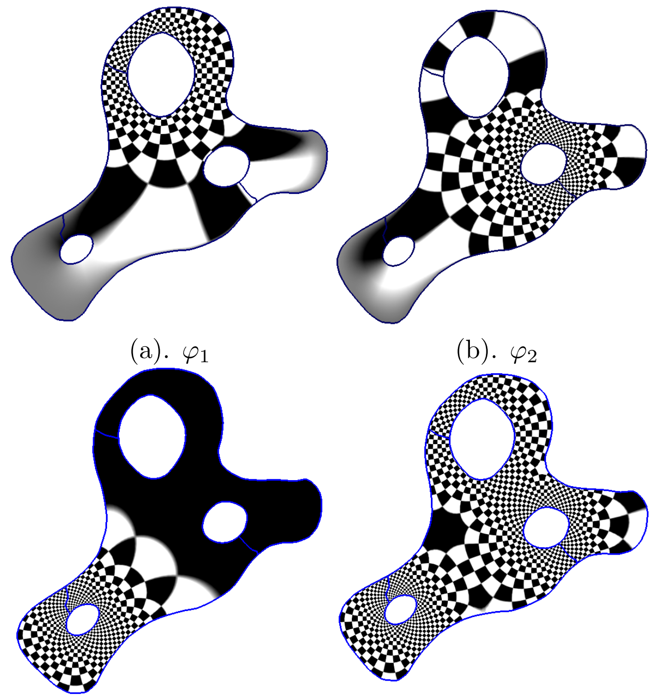
Another way to change the holomorphic 1-forms is to alter the boundary condition in Eqn. 4.3, then the same algorithm pipeline will produce different holomorphic 1-forms.

Optimization of Holomorphic 1-forms

Given a basis of the group of all holomorphic 1-forms ω_k, $k = 1, \ldots, n$, $\omega_k = \alpha_k + \sqrt{-1}\beta_k$, where α_k and β_k are the real and imaginary parts of ω_k respectively, we represent each ω_k as a simplicial 1-form defined on edges $\omega_k : E \to \mathbb{C}$. Any holomorphic 1-form ω is a linear combination of ω_k's,

$$\omega = \lambda_1\omega_1 + \lambda_2\omega_2 + \cdots + \lambda_n\omega_n,$$

where λ_k's are the coefficients. Applying ω to an edge $e \in E$, we have $\omega(e) = \sum_{k=1}^n \lambda_k(\alpha_k(e) + \sqrt{-1}\beta_k(e))$. The holomorphic 1-form ω induces a Riemannian metric $|\omega|^2$, denoted as $\mathbf{g}_\lambda$.



(a). φ_1 (b). φ_2 (c). φ_3 (d). $\varphi_1 - \varphi_2 + \varphi_3$

Figure 3: holomorphic 1-form basis.

For a triangle $\Delta_k \in F$ with edges e_1^k, e_2^k, e_3^3, its area under the metric induced by ω is given by

$$\begin{aligned} |\Delta_k|_{\mathbf{g}_\lambda} : &= \frac{1}{2}\omega(e_1^k) \times \omega(e_2^k) \\ &= \frac{1}{2}\sum_{i=1}^n \lambda_i\alpha_i(e_1^k)\sum_{j=1}^n \lambda_j\beta_j(e_2^k) \\ &- \frac{1}{2}\sum_{i=1}^n \lambda_i\beta_i(e_1^k)\sum_{j=1}^n \lambda_j\alpha_j(e_2^k) \\ &= \frac{1}{2}\sum_{i,j=1}^n \lambda_i\lambda_j \begin{vmatrix} \alpha_i(e_1^k) & \beta_i(e_1^k) \\ \alpha_j(e_2^k) & \beta_j(e_2^k) \end{vmatrix} \end{aligned}$$

which is a quadratic function in variables λ_k's. The total energy is defined as

$$(4.9) \qquad E(\lambda) := \sum_{\Delta_k \in F} (|\Delta_k|_{\mathbf{g}_\lambda} - |\Delta_k|_{\mathbf{g}})^2$$

after expansion

$$(4.10) \qquad E(\lambda) = \sum_{\Delta_k \in F}\left(\frac{1}{2}\sum_{i,j=1}^n c_k\lambda_i\lambda_j - |\Delta_k|_{\mathbf{g}}\right)^2$$

where

$$c_k := \begin{vmatrix} \alpha_i(e_1^k) & \beta_i(e_1^k) \\ \alpha_j(e_2^k) & \beta_j(e_2^k) \end{vmatrix}.$$

The partial derivative is

(4.11)

$$\frac{\partial E(\lambda)}{\partial \lambda_i} = 2 \sum_{\Delta_k \in F} \left(\frac{1}{2} \sum_{i,j=1}^{n} \lambda_i \lambda_j c_k - |\Delta_k|_{\mathbf{g}} \right) \left(\sum_{j=1}^{n} \lambda_j c_k \right)$$

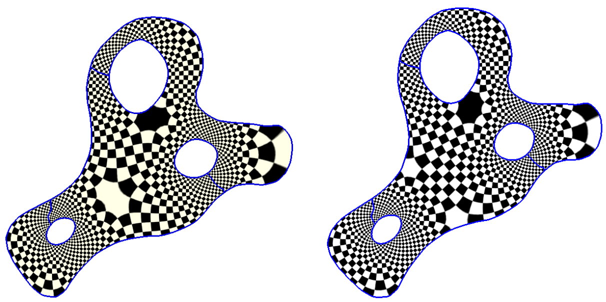


Figure 4: holomorphic 1-forms.

Quad-mesh generation Suppose we have obtained a holomorphic 1-form φ, we need to locate the zeros of the differential form. In general, there are $n-1$ zeros on the surface. For each vertex v_i, we estimate the conformal factor at v_i,

$$(4.12) \qquad u(v_i) := \frac{1}{n_i} \sum_{v_j \sim v_i} \frac{|\varphi([v_i, v_j])|^2}{|v_j - v_i|^2},$$

where n_i is the topological valence of v_i. Then we sort all the $u(v_i)$'s in the ascending order and choose the smallest $n-1$ vertices as the zeros.

Suppose v is a zero point of φ, its one right neighboring vertices $\{v_0, v_1, \dots, v_k\}$ sorted counter-clockwisely form a loop γ. We immerse the loop to the complex plane with $z(v) = 0$, $z(v_i) = \varphi([v, v_i])$, $i = 0, 1, \dots, k$, then the winding number of the image $z(\gamma)$ equals to the order of v plus one, $\mu_v(\varphi) + 1$. The real axis intersects $z(\gamma)$. Suppose the positive real axis intersects $z(\gamma)$ at the edge $[v_i, v_{i+1}]$, then we immerse the neighborhood of $[v_i, v_{i+1}]$ and extend the positive real axis to find the next intersection point. By repeating this procedure, we can extend the horizontal trajectory until it hits the boundary of the surface or returns to the zero point v again, then we have traced a critical horizontal trajectory through v. Similarly, we can also trace the critical vertical trajectory through v. All the critical horizontal and vertical trajectories partition the surface into patches $\{\Omega_1, \Omega_2, \dots, \Omega_k\}$, by integrating φ, we can embed each patch Ω_i on the complex plane. We tessellate the critical trajectories, then quadrangulate the planar image of each patch Ω_i with the tessellation of the critical trajectories as the boundary constraints to obtain a quad-mesh Q_i. The union of the quad-meshes Q_i's form the quad-mesh Q of the input surface S. Fig. 5 shows two quad-meshes induced by the holomorphic 1-forms in Fig. 4. The details of the algorithm pipeline can be found in Alg. 1.

Furthermore, the quad-element size can be treated as one of the input parameters. The algorithm computes the Jacobin matrix of the conformal mapping, and tessellated the parameter domain into quad-meshes, such that the mean quad edge length multiply the square root of the determinant of the Jacobin matrix equals to the input size parameter.

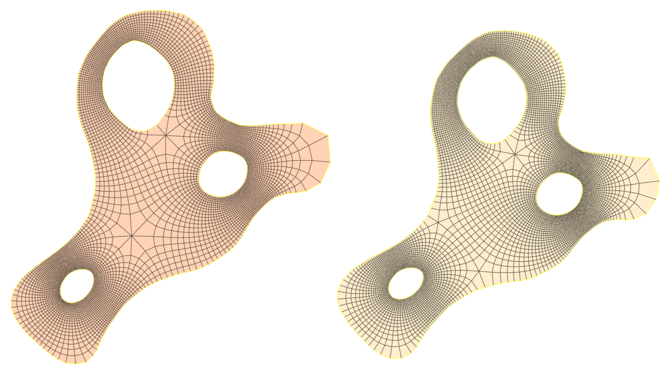


Figure 5: Quadrilateral meshes.

The final mesh quality depends on the initial triangular mesh. The computation of harmonic differentials is to solve the Laplace-Beltrami equation on the surface. According to the Finite element theory, the convergence, approximation accuracy of the discrete solutions depend on the triangle mesh quality.

5 Experimental Results

All algorithms have been developed using generic C++ under Visual Studio 2022 on the Windows platform. All the experiments are conducted on a laptop with Intel(R) Core(TM) i7-10750H CPU @2.60GHz with 6 cores and 64GB of memory.

We have tested our proposed method in medical imaging applications. Pretreatment Computed Tomography Angiography (CTA) scans of abdominal aortic aneurysm (AAA) patients treated with endovascular aneurysm repair (EVAR) were retrospectively and anonymously collected with Institutional Review Board approval. The scanned images were segmented, and the blood vessel surfaces were reconstructed. Each vessel surface is a topological poly-annulus with many branches.

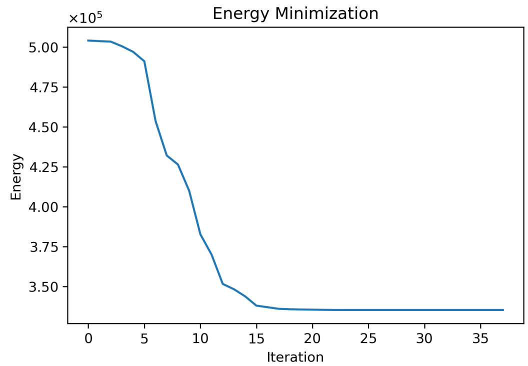


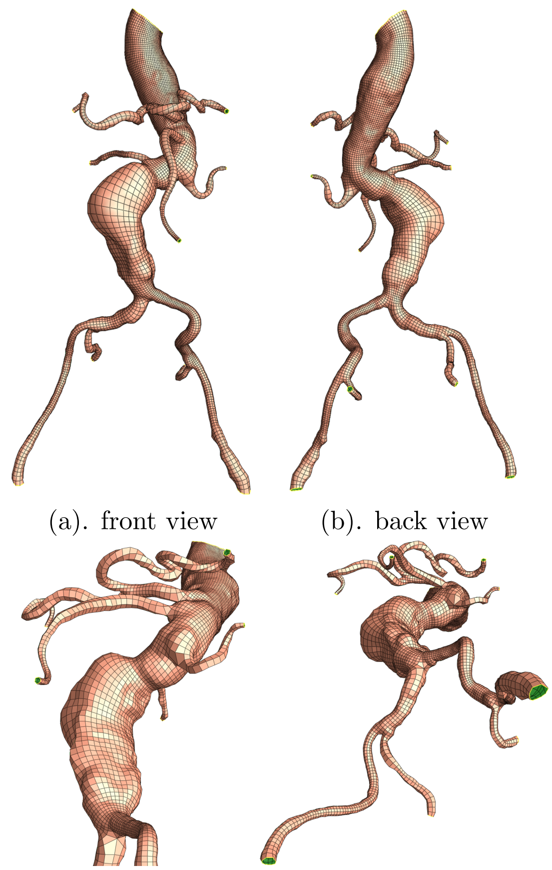
Figure 6: The energy $E(\lambda)$ monotonously decreases during the optimization.

Fig. 6 shows the optimization process for an abdominal aortic aneurysm model. The horizontal axis shows the number of iterations, and the vertical axis shows the energy. It can be seen clearly that the energy monotonously decreases and converges to the minimum during the process.

As shown in Fig. 10, the aneurysm model has 9 branches, 10 boundary components, 26165 vertices, and 50000 faces. The computation of holomorphic 1-form basis takes about 165.61 seconds, but the optimization takes only about $2144ms$. From the figure, it is easy to visually see that the uniformity of the checkers is improved prominently during the optimization. In the beginning, the checkers on thin branches are relatively sparse, while those on the top part of the main trunk are much denser. In the final stage, the checker density on the thin branches and that on the main trunk are much more uniform. Fig 7 shows the quad-mesh resulting from the optimization. It can be seen that the quad faces on both the thin branches and the main trunk are evenly distributed and suitable for simulation purposes. Fig. 8 shows the histograms of corner angles and the logarithms of the ratios between adjacent edge lengths of the quad-mesh in Fig. 7, which demonstrates the good quality of the mesh.

Fig. 11 and Fig. 9 show another blood vessel example, which has 7 branches and 8 boundary components. The vessel mesh is with 77242 vertices, 228481 edges, and 151242 faces. The computation of holomorphic differential basis takes about 303.74 seconds, and the optimization takes about $5510ms$. From the figures, we can see that during the optimization, the uniformity of the checkers is monotonously increasing, and the final quad-mesh has high quality.

We have tested about 8 blood vessel models with branches about $7-9$. The whole computational process is fully automatic without human intervention. The optimized quad-meshes are with high uniformity, and conformality to the input geometry, all the quad-faces are similar to the planar square. The quad-meshes are used to generate hex-meshes of the blood vessel wall as thin cells and applied for fluid dynamic simulation. The numerical computation process is stable and converges fast. These experiments demonstrate the practical value of the proposed quad-mesh optimization algorithms.



(a). front view (b). back view

(c). zoomed view (d). zoomed view

Figure 7: The quad-meshes obtained from the optimal result in Fig. 10.

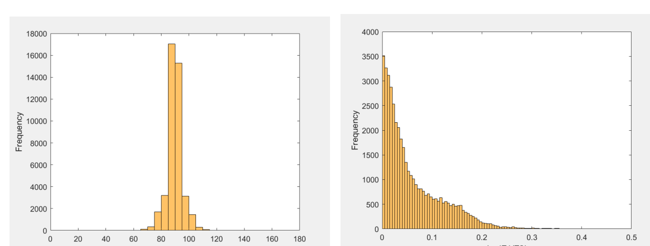


Figure 8: The histograms of the corner angles (left) and the logarithms of ratios between the adjacent edge lengths of the quad-mesh in Fig. 7.

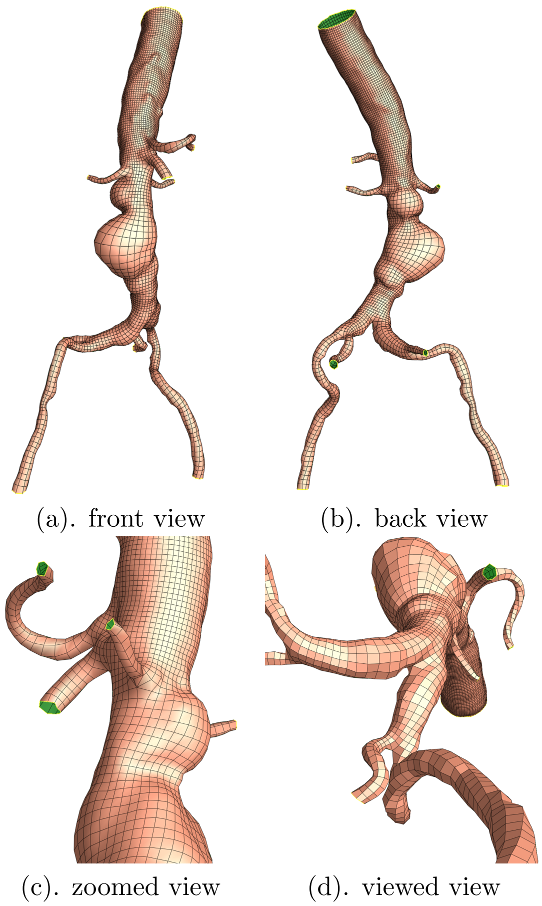


Figure 9: The quad-meshes obtained from the optimal result in Fig. 11.

6 Conclusion

This work proposes a practical algorithm for generating quadrilateral meshes as uniformly as possible on topological poly-annulus surfaces with the least number of singularities. We prove that quadrilateral meshes with $4k$ degree vertices, $k \in \mathbb{Z}_+$ induce holomorphic 1-forms, therefore the space of all such kinds of quad-meshes is finite dimensional. Hence in order to control the quad-mesh qualities, we propose to optimize a quartic polynomial energy to improve the uniformity. The experimental results demonstrate the efficiency and efficacy of the proposed method.

In our future works, we will improve the proposed method and validate its practical application by handling more complex engineering models with intricate constraints.

Acknowledgement

This work was supported by NIH grant# R21EB029733 and NSF grant# 2213852, NSFC T2225012, 61936002.

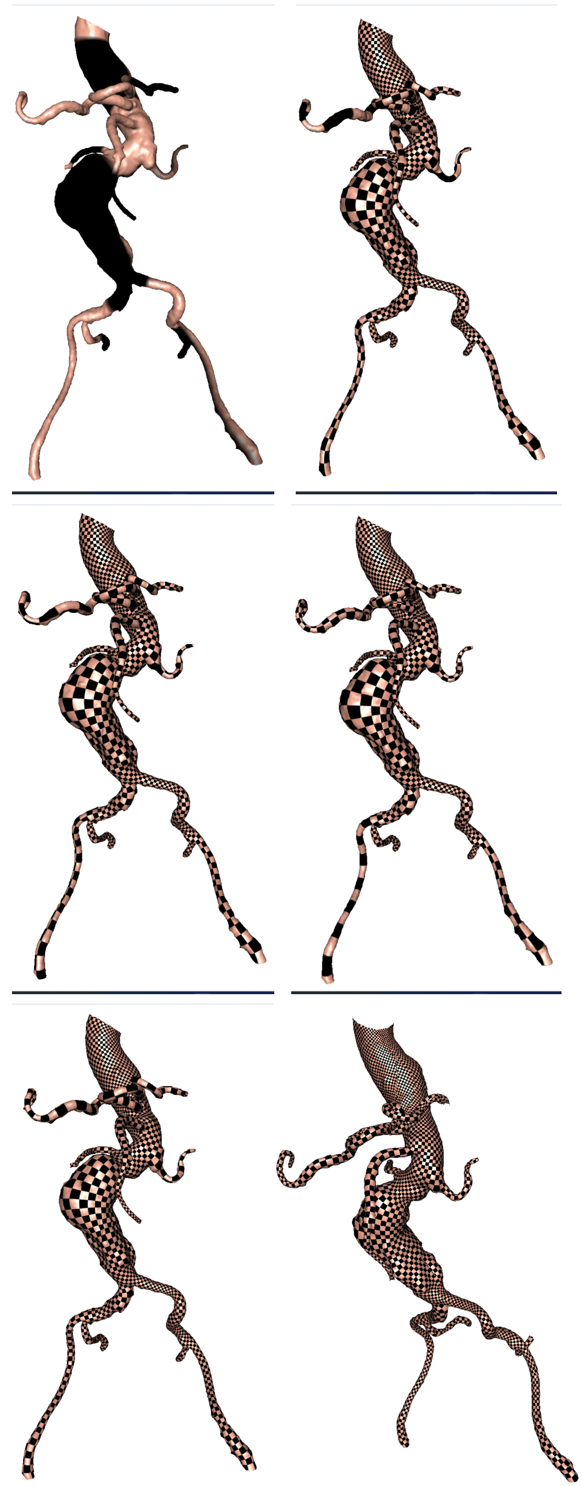


Figure 10: Optimization of the first blood vessel model, from top to bottom, left to right, during the process, the uniformity of the quadrilaterals increases.

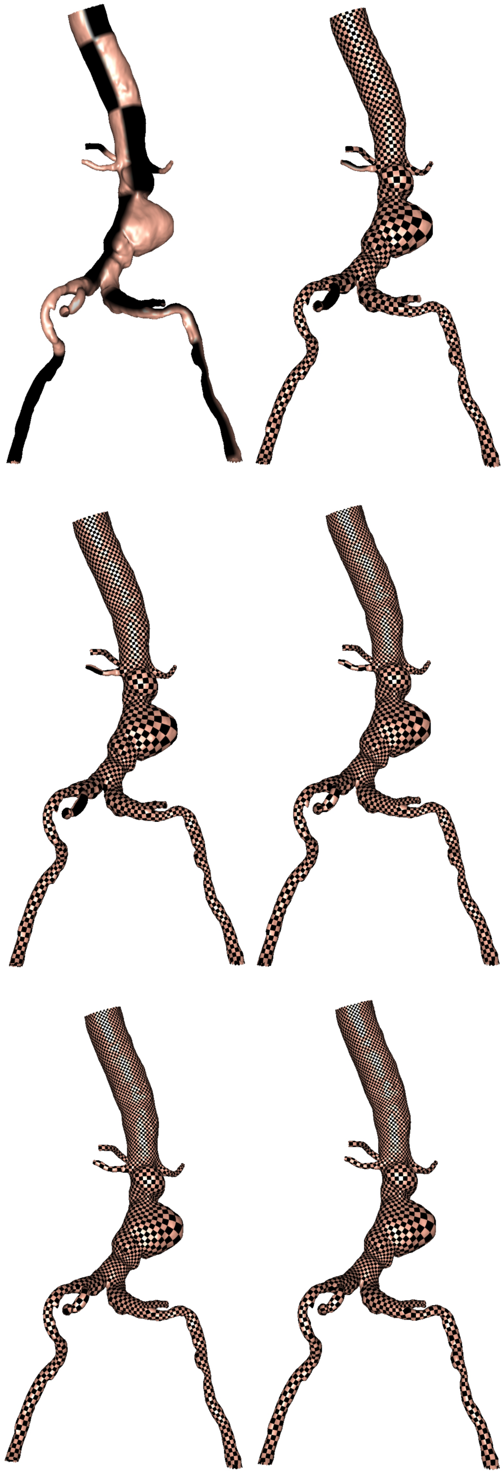


Figure 11: Optimization of the second blood vessel model, from top to bottom, left to right, during the process, the uniformity of the quadrilaterals increases.

Algorithm 1 Optimal Quad Mesh Generation

Require: A triangular mesh M with genus zero with $n+1$ boundary components;
Ensure: An quad-mesh of M with least distortion;
for all interior boundary loop γ_i **do**
  Compute the shortest path τ_i from γ_i to γ_0.
end for
for all interior boundary γ_i **do**
  Solve Eqn. 4.3 to obtain exact harmonic forms $\{df_1, \ldots, df_n\}$;
end for
for all cut τ_k **do**
  Slice M along τ_k to get $\bar{M}_k$, τ_k corresponds to τ_k^+ and τ_k^-;
  Construct random function g_k on $\bar{M}_k$ using Eqn. 4.4;
  Set the closed 1-form $\eta_k \leftarrow dg_k$
end for
for all closed 1-form η_k **do**
  Find the function $h_k : S \to \mathbb{R}$ by solving Eqn. 4.5;
  Set the non-exact harmonic 1-form $\omega_k \leftarrow \eta_k + dh_k$
end for
for all non-exact harmonic 1-form ω_k **do**
  Find the conjugate harmonic 1-form ${}^\star\omega_k$ by solving Eqn. 4.7;
  Set the holomorphic 1-form $\varphi_k \leftarrow \omega_k + \sqrt{-1}{}^\star\omega_k$
end for
Find the optimal holomorphic 1-form φ^* by minimizing the energy in Eqn. 4.10;
Locate the $n-1$ zeros $\{p_1, \ldots, p_{n-1}\}$ of φ^* by sorting the conformal factor Eqn. 4.12;
for all zero point p_i **do**
  Trace the critical horizontal trajectory through p_i;
  Trace the critical vertical trajectory through p_i;
end for
All the critical trajectories partition the surface S input patches $\{\Omega_1, \Omega_2, \ldots, \Omega_k\}$;
Tessellate the critical trajectories;
for all surface patch Ω_i **do**
  Map Ω_i onto a planar rectangle by integration φ^*;
  Quadrangulate the planar image of Ω_k with the tessellation of the critical trajectories as the boundary constraints to obtain a quad-mesh Q_k;
end for
Set the quad-mesh of M $Q \leftarrow \bigcup_k Q_k$;
return Q

References

[1] Boier-Martin, I., Rushmeier, H., Jin, J.: Parameterization of triangle meshes over quadrilateral domains. In: Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing. pp. 193–203 (2004)

[2] Bommes, D., Lévy, B., Pietroni, N., Puppo, E., Silva, C., Tarini, M., Zorin, D.: Quad-mesh generation and processing: A survey. In: Computer graphics forum. vol. 32, pp. 51–76. Wiley Online Library (2013)

[3] Carr, N.A., Hoberock, J., Crane, K., Hart, J.C.: Rectangular multi-chart geometry images. In: Symposium on geometry processing. pp. 181–190 (2006)

[4] Chen, W., Zheng, X., Ke, J., Lei, N., Luo, Z., Gu, X.: Quadrilateral mesh generation i: Metric based method. Computer Methods in Applied Mechanics and Engineering **356**, 652–668 (2019)

[5] Donaldson, S.: Riemann surfaces. Oxford University Press (2011)

[6] Dong, S., Bremer, P.T., Garland, M., Pascucci, V., Hart, J.C.: Spectral surface quadrangulation. In: ACM SIGGRAPH 2006 Papers, pp. 1057–1066 (2006)

[7] Gu, X., Yau, S.T.: Global conformal surface parameterization. In: First Eurographics Symposium on Geometry Processing (SGP03), Aachen, Germany. pp. 127–137 (June 23-25, 2003)

[8] Gurung, T., Laney, D., Lindstrom, P., Rossignac, J.: Squad: Compact representation for triangle meshes. In: Computer Graphics Forum. vol. 30, pp. 355–364. Wiley Online Library (2011)

[9] He, Y., Wang, H., Fu, C.W., Qin, H.: A divide-and-conquer approach for automatic polycube map construction. Computers & Graphics **33**(3), 369–380 (2009)

[10] Huang, J., Zhang, M., Ma, J., Liu, X., Kobbelt, L., Bao, H.: Spectral quadrangulation with orientation and alignment control. In: ACM SIGGRAPH Asia 2008 papers, pp. 1–9 (2008)

[11] Jiang, T., Fang, X., Huang, J., Bao, H., Tong, Y., Desbrun, M.: Frame field generation through metric customization. ACM Transactions on Graphics (TOG) **34**(4), 1–11 (2015)

[12] Jin, M., Kim, J., Luo, F., Gu, X.: Discrete surface ricci flow. IEEE Transaction on Visualization and Computer Graphics (TVCG) **14**(5), 1030–1043 (2008)

[13] Kälberer, F., Nieser, M., Polthier, K.: Quadcover-surface parameterization using branched coverings. In: Computer graphics forum. vol. 26, pp. 375–384. Wiley Online Library (2007)

[14] Kowalski, N., Ledoux, F., Frey, P.: A pde based approach to multidomain partitioning and quadrilateral meshing. In: Proceedings of the 21st international meshing roundtable. pp. 137–154. Springer (2013)

[15] Lai, Y.K., Jin, M., Xie, X., He, Y., Palacios, J., Zhang, E., Hu, S.M., Gu, X.: Metric driven rosy field design and remeshing. IEEE Transaction on Visualization and Computer Graphics (TVCG) **16**(1), 95–108 (2009)

[16] Lei, N., Zheng, X., Jiang, J., Lin, Y.Y., Gu, D.X.: Quadrilateral and hexahedral mesh generation based on surface foliation theory. Computer Methods in Applied Mechanics and Engineering **316**, 758–781 (2017)

[17] Lei, N., Zheng, X., Luo, Z., Gu, D.X.: Quadrilateral and hexahedral mesh generation based on surface foliation theory ii. Computer Methods in Applied Mechanics and Engineering **321**, 406–426 (2017)

[18] Lei, N., Zheng, X., Luo, Z., Luo, F., Gu, X.: Quadrilateral mesh generation ii: Meromorphic quartic differentials and abel–jacobi condition. Computer methods in applied mechanics and engineering **366**, 112980 (2020)

[19] Lei, N., Zhu, Y., Zheng, X., Si, H., Gu, X.: Why cross fields are not equivalent to quadrilateral meshes. CMAME (2023). https://doi.org/10.2139/ssrn.4460747, `http://dx.doi.org/10.2139/ssrn.4460747`

[20] Li, W.C., Vallet, B., Ray, N., Lévy, B.: Representing higher-order singularities in vector fields on piecewise linear surfaces. IEEE Transactions on Visualization and Computer Graphics **12**(5), 1315–1322 (2006)

[21] Lin, J., Jin, X., Fan, Z., Wang, C.C.: Automatic polycube-maps. In: Advances in Geometric Modeling and Processing: 5th International Conference, GMP 2008, Hangzhou, China, April 23-25, 2008. Proceedings 5. pp. 3–16. Springer (2008)

[22] Palacios, J., Zhang, E.: Rotational symmetry field design on surfaces. ACM Transactions on Graphics (TOG) **26**(3), 55–es (2007)

[23] Ray, N., Li, W.C., Lévy, B., Sheffer, A., Alliez, P.: Periodic global parameterization. ACM Transactions on Graphics (TOG) **25**(4), 1460–1485 (2006)

[24] Ray, N., Sokolov, D.: Robust polylines tracing for n-symmetry direction field on triangulated surfaces. ACM Transactions on Graphics (TOG) **33**(3), 1–11 (2014)

[25] Remacle, J.F., Lambrechts, J., Seny, B., Marchandise, E., Johnen, A., Geuzainet, C.: Blossom-quad: A non-uniform quadrilateral mesh generator using a minimum-cost perfect-matching algorithm. International journal for numerical methods in engineering **89**(9), 1102–1119 (2012)

[26] Shepherd, K.M., Gu, X.D., Hiemstra, R.R., Hughes, T.J.: Quadrilateral layout generation and optimization using equivalence classes of integral curves: theory and application to surfaces with boundaries. Journal of Mechanics **38**, 128–155 (2022)

[27] Shepherd, K.M., Gu, X.D., Hughes, T.J.: Feature-aware reconstruction of trimmed splines using ricci flow with metric optimization. Computer Methods in Applied Mechanics and Engineering **402**, 115555 (2022)

[28] Tarini, M., Pietroni, N., Cignoni, P., Panozzo, D., Puppo, E.: Practical quad mesh simplification. In: Computer Graphics Forum. vol. 29, pp. 407–418. Wiley Online Library (2010)

[29] Tong, Y., Alliez, P., Cohen-Steiner, D., Desbrun, M.: Designing quadrangulations with discrete harmonic

forms. In: Eurographics symposium on geometry processing (2006)

[30] Velho, L., Zorin, D.: 4–8 subdivision. Computer Aided Geometric Design **18**(5), 397–427 (2001)

[31] Wang, H., Jin, M., He, Y., Gu, X., Qin, H.: User-controllable polycube map for manifold spline construction. In: Proceedings of the 2008 ACM symposium on Solid and physical modeling. pp. 397–404 (2008)

[32] Xia, J., Garcia, I., He, Y., Xin, S.Q., Patow, G.: Editable polycube map for gpu-based subdivision surfaces. In: Symposium on interactive 3D graphics and games. pp. 151–158 (2011)

[33] Zheng, X., Zhu, Y., Chen, W., Lei, N., Luo, Z., Gu, X.: Quadrilateral mesh generation iii: Optimizing singularity configuration based on abel–jacobi theory. Computer Methods in Applied Mechanics and Engineering **387**, 114146 (2021). https://doi.org/10.1016/j.cma.2021.114146, `https://doi.org/10.1016/j.cma.2021.114146`