Spectral Quadrangulation with Feature Curve Alignment and Element Size Control

Ruotian Ling
The University of Hong Kong
and
Jin Huang*
State Key Lab of CAD&CG and Cyber Innovation Joint Research Center, Zhejiang University
and
Bert Jüttler
Johannes Kepler University
and
Feng Sun
The University of Hong Kong
and
Hujun Bao
State Key Lab of CAD&CG, Zhejiang University
and
Wenping Wang
The University of Hong Kong

Existing methods for surface quadrangulation cannot ensure accurate alignment with feature or boundary curves and tight control of local element size, which are important requirements in many numerical applications (e.g. FEA). Some methods rely on a prescribed direction field to guide quadrangulation for feature-alignment, but such a direction field may conflict with a desired density field, thus making it difficult to control the element size. We propose a new spectral method that achieves both accurate feature curve alignment and tight control of local element size according to a given density field. Specifically, the following three technical contributions are made. First, to make the quadrangulation align accurately with feature curves or surface boundary curves, we introduce novel boundary conditions for wave-like functions that satisfy the Helmholtz equation approximately in the least squares sense. Such functions, called *quasi-eigenfunctions*, are computed efficiently as the solutions to a variational problem. Second, the mesh element size is effectively controlled by locally modulating the Laplace operator in the Helmholtz equation according to a given density field. Third, to improve robustness, we propose a novel scheme to minimize the vibration difference of the quasi-eigenfunction in two orthogonal directions. It is demonstrated by extensive experiments that our method outperforms previous methods in generating feature-aligned quadrilateral meshes with tight control of local elememt size. We further present some preliminary results to show that our method can be extended to generating hex-dominant volume meshes.

Categories and Subject Descriptors: I.3.5 [**Computer Graphics**]: Computational Geometry and Object Modeling—*Quadrilateral mesh generation*

Additional Key Words and Phrases: boundary conditions, manifolds with boundaries, sharp features, spectral quadrangulation

* Corresponding author: hj@cad.zju.edu.cn

DOI 10.1145/XXXXXXX.YYYYYYY
http://doi.acm.org/10.1145/XXXXXXX.YYYYYYY

1. INTRODUCTION

Quadrilateral (*quad*) meshes are widely used in many applications, such as texture mapping, parameterization and finite element analysis (FEA) [Hormann et al. 2008]. In this paper we study the problem of computing a quad mesh (also called *quadrangulation*) of a given surface that needs to satisfy various constraints, such as feature alignment, element size and orientation. Currently, there is a lack of effective methods for surface quadrangulation that ensure accurate alignment with feature curves or boundary curves and tight control of local element size.

Feature alignment means that the resulting quadrilateral mesh preserves significant shape features, e.g. sharp edges and boundary curves, while element size control means that locally the size of

mesh elements follows closely a given density function to produce an adaptive mesh. To achieve feature alignment, previous methods [Kalberer et al. 2007; Bommes et al. 2009; Zhang et al. 2010] usually use a direction field to guide the orientation of mesh elements to align with feature curves. However, in this way, the size element will be heavily dictated by the guiding direction field, rather than following a prescribed density field. This incompatibility between the guiding directional field and the given density field is often responsible for undesired singularities and low-quality mesh elements in quadrangulation. Note that the control of element size is often as important as the orientation control in quad meshing for many numerical applications. In fact, commonly used meshing tools for FEA, such as SolidWorks, ANSYS, Abaqus, provide the density control rather than the orientation control.

We present a new spectral method for quadranulation that achieves both accurate feature curve alignment and tight control of local element size. The first spectral method for surface quadragulation [Dong et al. 2006] employs the Morse-Smale complex (MSC) extracted from the solution functions of Helmholtz equation on a given surface. It works well for closed smooth surfaces but does neither consider orientation alignment with boundary curves and sharp feature curves nor the control of element size. The method is later improved in [Huang et al. 2008] to achieve approximate feature alignment without emphasizing on element size control. In our method we propose new boundary conditions of the Helmholtz equation to make the Morse-Smale Complex (MSC) associated with its solution (to be called *quasi-eigenfunction* or *QE* for short) precisely align with domain boundaries or feature curves, without the need of a guiding direction field. Because these new boundary conditions impose over-constraints, the quasi-eigenfunction is defined as the minimizer of a variational problem and therefore satisfies the Helmholtz equation only approximately in the least squares sense. With this formulation, we are able to specify and enforce the desired element size using an isotropic Riemannian metric to locally modulate the Laplace operator in the Helmholtz equation. Overall, we formulate surface quadrangulation as a linearly constrained quadratic optimization problem and compute its solution (i.e. the resulting quasi-eigenfunction) by solving a sparse linear system. In addition, to improve the distribution of the critical points of the quasi-eigenfunction for robustly constructing its Morse-Smale complex, we propose a novel non-linear optimization scheme to make the vibration amplitude of the quasi-eigenfunction similar in two orthogonal directions.

2. RELATED WORKS

Many methods have been proposed on surface quadrangulation, or quad mesh generation, in recent decades. Comprehensive surveys are provided in [Hormann et al. 2008; Bommes et al. 2012]. Some of the methods [Tarini et al. 2011; Campen et al. 2012] aim at generating quad mesh with a simple coarse structure. We shall only review those works related to our concern of generating quad meshes with tight control of element size and accurate feature curve alignment.

The methods in [Alliez et al. 2003; Dong et al. 2005] trace two orthogonal direction fields on a given surface and generate a quad-dominant mesh. To get a pure quad mesh, a coarse quad domain is required in some works [Tong et al. 2006; Marinov and Kobbelt 2006]. In other works [Ray et al. 2006; Kalberer et al. 2007; Bommes et al. 2009], a direction field is used to extract a topological structure for parameterization. The direction field is often

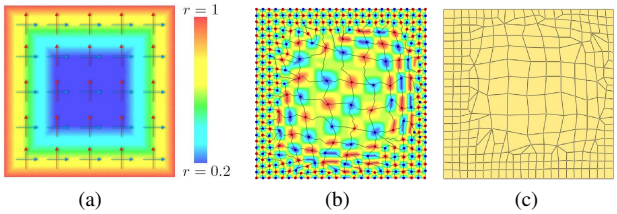


Fig. 1: *Incompatibility between element size control and direction control in [Zhang et al. 2010]. (a) The density function (in color) and the direction field (in arrows) on a square domain; (b) The generated scalar field and its Morse-Smale complex; (c) The resulting quad mesh.*

constructed based on sharp feature curves or curvature directions, without taking size control into consideration. Recently, Kovacs *et al.* [2011] proposed a method to improve approximation error of quadrangulations by introducing a curvature-dependent surface anisotropy metric. This method still relies on a direction field for feature alignment. As pointed out in [Zhang et al. 2010], there is often a conflict between applying the element size control and defining a direction field for feature alignment.

As shown in Fig. 1, a uniform direction field is derived from the boundaries of a square. This direction field is not compatible with the desired density field shown in pseudo-color, which leads to many unwanted singularities. Curl minimization proposed in [Zhang et al. 2010] could be used to adjust the size control according to the direction field. However, such an adjustment according to the uniform direction field will lead to a mesh with nearly uniform element size, instead of the desired result (Fig. 9).

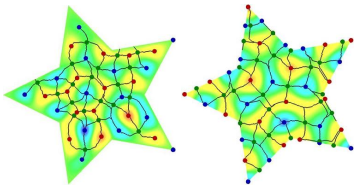


Fig. 2: *Eigenfunctions with Dirichlet boundary condition (left) and Neumann boundary condition (right), respectively.*

Different from the above methods, the spectral approach by Dong *et al.* [2006] uses the Morse-Smale complex from an eigenvalue function of the Helmholtz equation over a given surface to generate a quad mesh. The method is proposed for quandrangulation of closed surfaces and thus cannot handle surfaces with boundaries or feature lines. This difficulty is illustrated by the examples of two eigenfunctions in a planar domain (that has a boundary) in Fig. 2. Here neither the Dirichlet boundary condition nor the Neumann boundary condition can make the Morse-Smale complex align with domain boundary. Hence, the resulting quad elements do not conform with the boundary.

Huang *et al.* [2008] extended this method in [Dong et al. 2006] by adding orientation and alignment control of quads using a guiding direction field within an optimization framework. The improved method attempts to handle the boundary and feature curves with a set of symmetric constraints but it is effective only for surfaces with simple boundary curves and feature lines. The method proposed in [Zhang et al. 2010] for improving boundary and feature alignment adopts the same framework of Morse-Smale complex. It uses a feature-dependent direction field and thus also restricts the ability of element size control. Recently, a method [Pellenard et al. 2011] has been proposed for controlling the size and orientation of isotropic 2D quadrangulation that addresses the requirement of size control before orientation alignment. Because the method relies on

labeling the triangles of the background triangulation through local information, it is difficult to reduce mesh singularities; furthermore, it produces results that are sensitive to the initial tessellation. For example, the quad mesh has many singularities even in the case of a simple quad domain with a constant cross field and uniform sizing.

3. BOUNDARY ALIGNED QUASI-EIGENFUNCTION

Like the previous spectral approach (e.g. [Dong et al. 2006]), our method also uses the Morse-Smale complex (MSC) to build a quad mesh. An MSC of a function f on a two-manifold is a complex connecting the critical points (minimum/saddle/maximum) of f through the integral curves of the gradient field ∇f. The MSC is composed of a set of quad cells and therefore yields the structure of a pure quad mesh [Edelsbrunner et al. 2003; Ni et al. 2004]. Note that mesh orientation and element size of the resulting quad mesh are largely determined by the geometry of the MSC.

We first discuss how to achieve the alignment of boundary curve and feature curves by introducing some new boundary conditions, without resorting to a direction field. For simplicity, we will only discuss boundary curves since the method can be applied to feature curve alignment as well in a straightforward manner.

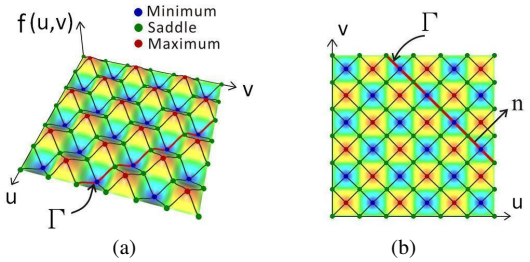


Fig. 3: *(a) Perspective view of the function* $f(u,v) = 0.075\sin(6\pi u)\sin(6\pi v)$*; (b) Top view of* $f(u,v)$.

3.1 Boundary Conditions

Consider a function $f(u,v)$ with its critical points (i.e., maxima, minima, and saddles) as illustrated in Fig. 3. A *minimal integral curve* is defined to be the integral curve of the gradient field ∇f that starts at a minimum and ends at a saddle. An *extended minimal integral curve*, denoted Γ, is defined to be a sequence of connected minimal integral curves (see the thick red curve in Fig. 3). Clearly, an extended minimal integral curve represents locally the orientation of the quad mesh represented by the Morse-Smale complex. Let $\mathbf{n}$ be the unit normal vector of the curve Γ. Because Γ is an integral curve of the gradient field ∇f, we have $\partial f/\partial \mathbf{n} = 0$. Furthermore, because Γ only goes from a minimum to a saddle, it can be shown that $\partial^2 f/\partial \mathbf{n}^2 > 0$ (except for high-order saddles). In fact, these two conditions characterize an extended minimal integral curve.

Given a 2D domain Ω, our idea for achieving boundary alignment is to construct a function f on Ω such that the domain boundary curves $\partial\Omega$ become extended minimal integral curves of f. To this end, we introduce two boundary conditions $\partial f/\partial \mathbf{n} = 0$ and $\partial^2 f/\partial \mathbf{n}^2 > 0$ on the domain boundary $\partial\Omega$. The sufficiency of these conditions can be seen as follows: Because $\partial f/\partial \mathbf{n}\big|_{\partial\Omega} = 0$, the boundary curve must be an integral curve. Because $\partial^2 f/\partial \mathbf{n}^2\big|_{\partial\Omega} > 0$, no maximum can appear on the boundary. Therefore there are only saddle points and minimum points, occurring alternatively, on the boundary $\partial\Omega$. Hence, the boundary curve $\partial\Omega$ is an extended minimal integral curve of f. For definiteness in computation, we replace $\partial^2 f/\partial \mathbf{n}^2 > 0$ by the condition $\partial^2 f/\partial \mathbf{n}^2 = \xi$ for some constant $\xi > 0$. That is, we will use in our computation the boundary conditions $\partial f/\partial \mathbf{n}|_{\partial\Omega} = 0$ and $\partial^2 f/\partial \mathbf{n}^2|_{\partial\Omega} = \xi > 0$.

The above boundary conditions are sufficient but not necessary for feature alignment, since it is possible that a boundary curve is aligned with maximal integral curve (which connects a saddle to a maximum) rather than with a minimal integral curve, and the second order directional derivative can be different. In other words, a more general formulation of this boundary condition could be $\sigma_\Gamma \cdot \partial^2 f/\partial \mathbf{n}^2|_{\partial\Gamma} \geq \xi > 0$, where $\sigma_\Gamma \in \{1,-1\}$ for each connected boundary curve $\Gamma \in \partial\Omega$. However, using the above equation leads to complex problem involving binary integer programming about σ_Γ and inequality constraints about ξ. Our assumption that all the boundary curves or feature curves are aligned with only minimal integral curves imposes the restriction that the number of columns of mesh elements between two "parallel" feature lines is even. In practice, this does not lead to any problem because we are primarily interested in quadrangulation with sufficiently small element size. Further more, setting ξ to a positive constant leads to a simple linearly constrained quadratic optimization problem, which favors efficiency and still yields high quality results. Different specific values of the constant $\xi > 0$ scale f globally and therefore do not affect the distribution of the critical points of f, as well as its Morse-Smale complex. Hence, we set to $\xi = 1$ in our computation; that is, the condition $\partial^2 f/\partial \mathbf{n}^2|_{\partial\Omega} = 1$ is used.

In a related note, Huang *et al.* [Huang et al. 2008] also introduced a boundary condition for feature alignment within the spectral framework. Their condition is that the directional derivative of f in the normal direction of the domain boundary vanishes. This tends to reduce the variance of the resulting scalar function near the feature rather than explicitly enforcing mesh nodes to align with the boundary curve. Therefore, the method cannot ensure satisfactory conformation of the resulting quad mesh with boundary curves and feature curves.

3.2 Definition of Quasi-Eigenfunction

If a function f satisfies the Helmholtz equation,

$$\nabla^2 f = \lambda f, \tag{1}$$

where $\lambda \leq 0$, f is called an eigenfunction of the Laplace operator. Due to the relatively even distribution of the critical points of f, the Morse-Smale complex of f has been used to construct quad meshes [Dong et al. 2006; Huang et al. 2008].

For a 2D domain with boundary curves, the Dirichlet boundary condition or the Neumann boundary condition are often used in the eigenfunction problem in Eq. (1). However, as shown in Fig. 2 neither of them ensures boundary curve alignment as required for quadrangulation.

If we apply our new boundary conditions proposed in Section 3.1 for boundary alignment, the eigenfunction problem in Eq. (1) becomes over-constrained and therefore cannot be solved. We circumvent this difficulty by introducing the quasi-eigenfunction via a variational formulation as follows.

To better illustrate this variational formulation, let us start by considering the 1D case, that is, a smooth functions defined on $[-1, 1]$. Let f denote the function that minimizes the following energy

$$E_L(f) = \int_{-1}^{1} \|\nabla^2 f - \lambda f\|^2, \tag{2}$$

subject to the boundary conditions

$$\frac{\mathrm{d}f}{\mathrm{d}u}\Big|_{u=\pm 1} = 0, \quad \frac{\mathrm{d}^2 f}{\mathrm{d}u^2}\Big|_{u=\pm 1} = 1, \tag{3}$$

where $\nabla^2 = \mathrm{d}^2/\mathrm{d}u^2$ is the Laplace operator, and λ a negative constant. The solution to this constrained optimization problem will be called a *quasi-eigenfunction* (QE).

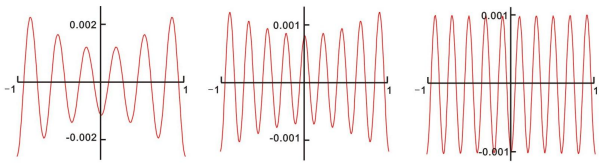
Eq. (2) and Eq. (3) define a typical problem in variational calculus. Its Euler-Lagrange equation is

$$(\nabla^2)^2 f - 2\lambda \nabla^2 f + \lambda^2 f = 0. \tag{4}$$

When f is an eigenfunction, the energy in Eq. (2) is 0, and it also satisfies Eq. (4). A quasi-eigenfunction approximates an eigenfunction in the least squares sense while respecting the boundary conditions in Eq. (3). Note that here λ does not have to be an eigenvalue of Eq. (1); indeed, a nontrivial solution to Eq. (4) subject to Eq. (3) is well defined for any $\lambda < 0$. Given a λ, the analytical form of the QE in $[-1, 1]$ can be shown to be

$$\begin{aligned} f_1^*(u) =& c_1 \sin(u\sqrt{-\lambda}) + c_2 \cos(u\sqrt{-\lambda}) + \\ & c_3 u \sin(u\sqrt{-\lambda}) + c_4 u \cos(u\sqrt{-\lambda}), \end{aligned} \tag{5}$$

where c_1, c_2, c_3, c_4 are constants. Fig. 4 shows the graphs of several QEs.



(a) (b) (c)

Fig. 4: *Quasi-eigenfunctions in* $[-1, 1]$ *with (a)* $\lambda = -400$*; (b)* $\lambda = -750$*; (c)* $\lambda = -1000$*.*

Now we consider the quasi-eigenfunction (QE) on an open 2-manifold Ω. The boundary conditions in Eq. (3) become

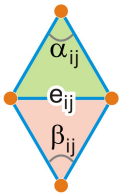
$$\frac{\partial f}{\partial \mathbf{n}}\Big|_{\partial\Omega} = 0, \quad \frac{\partial^2 f}{\partial \mathbf{n}^2}\Big|_{\partial\Omega} = 1, \tag{6}$$

where $\mathbf{n}$ is the outward unit normal vector of the boundary curve. It can be shown that the QE on the domain $[-1, 1]^2$ is

$$f_2^*(u, v) = f_1^*(u) + f_1^*(v). \tag{7}$$

So far we have shown how to use two new boundary conditions to ensure boundary alignment without using any direction field. Now, the element size can easily be controlled by integrating an isotropic Riemannian metric into the Laplace operator in the Helmholtz equation, that is, $\nabla_r^2 : (\partial^2/\partial u^2 + \partial^2/\partial v^2)/r, r > 0$, where the desired quad edge length is almost linearly proportional to $\frac{1}{\sqrt{r}}$. This treatment is similar to that in [Huang et al. 2008]; but, instead of using mass density, we adopt the metric description in [Kovacs et al. 2011]. Note that the metric related parameter r does not have to be continuous, because we are solving a least square problem Eq. (2). However, in practice, a smooth r is preferred for generating smoothly varying meshes.

3.3 Discrete Quasi-Eigenfunction (QE)



Now we show how to compute quasi-eigenfunctions efficiently on 2D triangulated domains. A well–known discretization of the Laplace operator on a triangulated surfaces is the cotangent formula [Pinkall and Polthier 1993; Meyer et al. 2002]. Let $\mathbf{f}$ be the vector of all function values defined on the vertices of a triangular mesh $\mathcal{M}$. The cotangent formula for computing the Laplacian on vertex i with respect to the density distribution r is

$$\nabla_r^2 \mathbf{f}_i = \frac{3}{2\mathrm{Area}_i \cdot r_i} \sum_{j \in N(i)} \big(\cot(\alpha_{ij}) + \cot(\beta_{ij})\big)(\mathbf{f}_j - \mathbf{f}_i), \tag{8}$$

where $N(i)$ is the set of vertices incident to vertex i, Area_i the total area of triangles incident to vertex i, and α_{ij}, β_{ij} the angles opposite to edge ij.

The boundary conditions in Eq. (6) involve the first order and second order derivatives of $\mathbf{f}$. We adopt the strategy in [Huang et al. 2008] to locally approximate $\mathbf{f}$ using a second order polynomial, and thus represent it as a linear combination of the values on its neighboring vertices. Specifically, the neighborhood $N(i)$ of vertex i is locally parameterized into a plane by the exponential map [Schmidt et al. 2006] (see Fig. 5(a)), and each vertex $j \in N(i)$ gets assigned some coordinates (u_j, v_j). Then the coefficients a_k, $k \in \{uu, uv, vv, u, v, c\}$, of a quadratic polynomial $\tilde{f}(u, v) = \frac{1}{2}\big(a_{uu}u^2 + 2a_{uv}uv + a_{vv}v^2\big) + a_u u + a_v v + a_c$ are solved for from the least squares problem

$$\min_{j \in N(i)} \|\tilde{f}(u_j, v_j) - \mathbf{f}_j\|^2. \tag{9}$$

It can be shown that the coefficients a_k obtained from Eq. (9) can be represented as a linear combination of $\mathbf{f}$, i.e., $a_k = \mathbf{a}_k^T \mathbf{f}$. Note that the $\mathbf{a}_k$ are independent of $\mathbf{f}$ and depend only on the discretization of the domain. Hence given a triangulation, all entries of the gradient vector and the Hessian matrix on a boundary vertex can be represented as a linear combination of $\mathbf{f}$ with pre-computed coefficients. For a boundary vertex with normal direction (n_u, n_v) in the local 2D frame of the tangent plane of the underlying surface domain, the directional derivatives in Eq. (6) are approximated by

$$\begin{aligned} \frac{\partial f}{\partial \mathbf{n}} &\approx (n_u \mathbf{a}_u + n_v \mathbf{a}_v)^T \mathbf{f} \\ \frac{\partial^2 f}{\partial \mathbf{n}^2} &\approx (n_u^2 \mathbf{a}_{uu} + 2n_u n_v \mathbf{a}_{uv} + n_v^2 \mathbf{a}_{vv})^T \mathbf{f}. \end{aligned} \tag{10}$$

Then, all the boundary conditions can be represented in a matrix form

$$\mathbf{B}\mathbf{f} = \begin{bmatrix} \mathbf{Y} \\ \mathbf{Z} \end{bmatrix} \mathbf{f} = \begin{bmatrix} \mathbf{0} \\ \mathbf{1} \end{bmatrix} = \mathbf{C}. \tag{11}$$

Note that using a two-ring neighborhood for all the vertices will make the $\mathbf{a}_k$ contain too many non-zero entries, thus leading to a high computational cost. Therefore, we adopt a "diameter-2" neighborhood, i.e. a two-ring neighborhood for open boundary vertices

and a one-ring neighborhood for interior vertices when solving for the coefficients $\mathbf{a}_k$ via fitting.

Clearly, the boundary conditions in Eq. (6) introduced for boundary curve alignment can be applied directly to feature curve alignment. The discretization of these conditions at a vertex on a feature line is similar to that at a boundary vertex, as shown in Fig. 5(b).

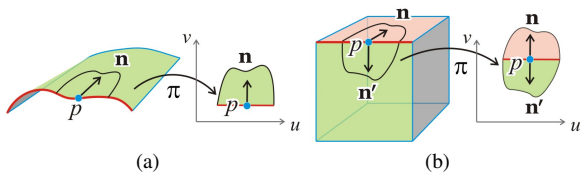


Fig. 5: *(a) A boundary node and its exponential map; (b) A node on a feature line and its exponential map.*

Now the minimization problem in Eq. (2) subject to Eq. (6) is formulated discretely: we minimize the energy E_{L_r}

$$E_{L_r}(\mathbf{f}) = ||\mathbf{L_r}\mathbf{f} - \lambda\mathbf{f}||^2, \tag{12}$$

subject to the constraints in Eq. (11), where $\mathbf{L_r}$ is the Laplacian matrix defined by Eq. (8). Unlike the optimization problem with non-linear constraints in [Huang et al. 2008], our boundary conditions lead to a quadratic minimization problem with linear constraint, therefore can be solved directly using its KKT matrix [Boyd and Vandenberghe 2004] with the solver UMFPACK [Davis 2004]:

$$\begin{bmatrix} \mathbf{H}_{L_r} & \mathbf{B}^T \\ \mathbf{B} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{f} \\ \nu \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{C} \end{bmatrix}, \tag{13}$$

where ν is the Lagrange multiplier, and

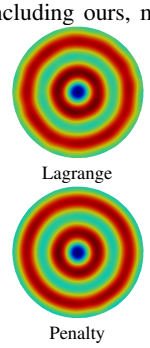
$$\mathbf{H}_{L_r} = \mathbf{L_r}^T\mathbf{L_r} - \lambda(\mathbf{L_r} + \mathbf{L_r^T}) + \lambda^2\mathbf{I}. \tag{14}$$

After solving the QE from the above sparse linear system, its Morse-Smale complex (MSC) can be extracted for quad mesh generation. The methods in [Edelsbrunner et al. 2003; Ni et al. 2004] can be used to construct the MSC of a function f on a closed triangular mesh surface. We adopt the mirrored boundary condition in [Gyulassy et al. 2011] to extract the MSC on a surface with open boundary. Then the positions of the MSC vertices are further refined by solving a global smooth parameterization problem [Dong et al. 2006]. Finally, as postprocessing, the Catmull-Clark subdivision can be used to refine the mesh.

While our method does not restrict λ to be an eigenvalue, not all the choices of λ lead to high quality results because of the possible incompatibility between the alignment constraints and the quad sizing, especially when using small $|\lambda|$ for coarse quadrangulation. When a large $|\lambda|$ is used, to make sure the critical points in the resulting QE are kept sufficiently apart from each other (desirably about the distance of the average edge length of the underlying triangle mesh), the input triangular mesh should be dense enough with respect to the desired quad density. To address this issue, after extracting the critical points, we locally estimate the average distance d between the neighboring critical points, and then subdivide the triangular mesh to make the edge length shorter than μd, and then re-solve the problem with the refined mesh as input to obtain a properly discretized QE. We set μ to 0.25 experimentally. Such a strategy works well in most cases, as long as the Hessian matrix $\mathbf{H}_{L_r}$ is well conditioned.

3.4 Vibration Enhancement

Like all the other spectral methods [Dong et al. 2006; Huang et al. 2008], our method also assumes the existence of periodically distributed significant critical points of the scalar function. But, in some case, either an eigenfunction or a quasi-eigenfunction function may not have their critical point distributed in a desirable manner. Thus, such methods, including ours, may suffer from lacking "vibration", as illustrated in the inset. For example, for a disk-like region with a strong rotational symmetry, the QE tends to vibrate only along the radial direction while not to vary much along the direction perpendicular to the radial direction. Consequently, the critical points are not easy to detect or not uniformly distributed, an issue also pointed out in [Dong et al. 2006; Huang et al. 2008]. In the following we will introduce a technique, called *vibration enhancement*, to address this issue.



As shown in [Huang et al. 2008], the principal directions of the Hessian matrix of the QE are locally aligned with the edges of the Morse-Smale complex. Using these directions as the local coordinate system, we locally approximate the QE by $f(u,v) \approx A_u\cos(u\sqrt{-\lambda}+\theta) + A_v\cos(v\sqrt{-\lambda}+\phi)$. It follows that

$$\nabla f \nabla f^T + \frac{H_f^2}{-\lambda} \approx -\lambda \begin{bmatrix} A_u^2 & A_u \cdot A_v \cdot s_u s_v \\ A_u \cdot A_v \cdot s_u s_v & A_v^2 \end{bmatrix}, \tag{15}$$

where $s_u = \sin(u\sqrt{-\lambda}+\theta), s_v = \sin(v\sqrt{-\lambda}+\phi)$. Again, applying the discretization techniques in Section 3.3 to compute the gradient ∇f and the Hessian matrix H_f, we have

$$A_u^2 = \mathbf{f}^T W^u \mathbf{f}, \quad \text{and } A_v^2 = \mathbf{f}^T W^v \mathbf{f}, \tag{16}$$

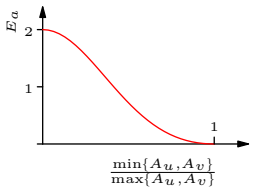
where:

$$\begin{aligned} W^u &= \frac{1}{-\lambda}\mathbf{a}_u\mathbf{a}_u^T + \frac{1}{\lambda^2}(\mathbf{a}_{uu}\mathbf{a}_{uu}^T + \mathbf{a}_{uv}\mathbf{a}_{uv}^T) \\ W^v &= \frac{1}{-\lambda}\mathbf{a}_v\mathbf{a}_v^T + \frac{1}{\lambda^2}(\mathbf{a}_{vv}\mathbf{a}_{vv}^T + \mathbf{a}_{uv}\mathbf{a}_{uv}^T). \end{aligned} \tag{17}$$

From Eq. (15), we can estimate the vibration amplitude $A_{u,i}, A_{v,i}$ for the vertex i, and measure the amplitude difference as:

$$E_{a,i}(\mathbf{f}) = (\mathbf{f}^T W_i^u \mathbf{f}/\bar{A}_i^2 - 1)^2 + (\mathbf{f}^T W_i^v \mathbf{f}/\bar{A}_i^2 - 1)^2, \tag{18}$$

where $\bar{A}_i = \sqrt{(A_{u,i}^2 + A_{v,i}^2)/2}$. The inset plots the relationship of the energy and the amplitude difference. The penalty terms are applied to all the vertices $\mathcal{V}$ to make local vibration amplitudes in the two orthogonal direction as similar as possible.

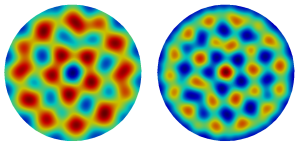


Using the solution of Eq. (13) as the initial value, we optimize the following energy

$$E(\mathbf{f}) = E_{L_r}(\mathbf{f}) + \omega\left(\sum_{i\in\mathcal{V}} E_{a,i}(\mathbf{f})\right) + \xi\|\mathbf{B}\mathbf{f} - \mathbf{C}\|_2^2, \tag{19}$$

where the weights $\omega = 0.1, \xi = 100$ are used in all our experiments. In the phase of vibration enhancement, Eq. (19) is optimized by the Gauss-Newton method, and CHOLMOD [Chen et al. 2008] is used in each iteration. The cost of each iteration is less than solving the KKT system in Eq. (13), and 10 iterations are performed in our experiments unless otherwise stated.

We use the penalty function with the boundary condition $\mathbf{Bf} = \mathbf{C}$, because using the Lagrange multipliers would introduce a large number of hard constraints, which prevents the scalar field near the boundary from adjusting for desired vibration. As shown in the inset, the critical points on the boundary are more prominent with the penalty scheme (left) than using the Lagrange multipliers scheme (right).



Vibration enhancement also improves the robustness of our method when the value of λ is away from an eigenvalue. For example, for the quadrangulation of the domain $[-1, 1]^2$ in Fig. 7, to generate quad meshes from the QEs with $n = 4$ and $n = 5$ periods along each side, the ideal values of λ should be $\lambda \approx -158$ and $\lambda \approx -247$, respectively. (Ideally, λ should be $-(n\pi)^2$.) If we set $\lambda = -200$, which lies between the two ideal values, the QE solved from Eq. (13) does not possess prominent critical points (see the region surrounded by the black rectangle in Fig. 7). However, the result improves significantly after applying a few iterations of vibration enhancement.

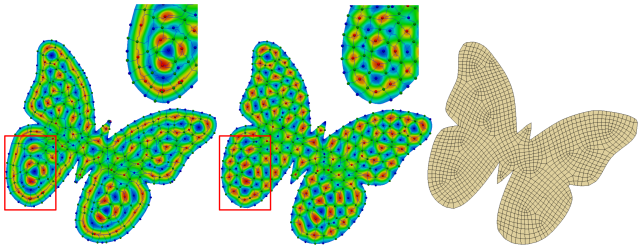


Fig. 6: *BUTTERFLY ($\lambda = -1700$): (a) QE without vibration enhancement; (b) QE with vibration enhancement; (c) Quad mesh from (b).*

4. EXTENSION TO HEX-DOMINANT REMESHING

Hexahedral meshing of a 3D domain is another important and challenging problem in mesh generation. Unlike a 2D manifold, which can be closed without no boundary, a compact 3D volume always has a boundary surface. That is, the boundary alignment issue is inevitable in hex meshing of *any* 3D volume. Hence, due to their lack of proper boundary treatment, the previous spectral remeshing methods [Dong et al. 2006; Huang et al. 2008] for surface quadrangulation cannot be extended to hex meshing of 3D volumes. In this section, we will briefly discuss how our spectral method with new boundary conditions can be extended Eq. (6) to hexahedral mesh generation, and present some preliminary results.

One major consideration in this extension is how to discrete a 3D domain and represent the Laplacian operator on it. We assume that the domain is represented by a sufficiently fine tetrahedral mesh. If we use the 3D counterpart of the cotangent formula in Eq. (8) for tetrahedral meshes [Wang et al. 2003], the obtuse angles in the tetrahedral mesh will result in the negative values of the cotangent formula, which in turn will cause numerical instability [Wardetzky et al. 2007]. Because completely eliminating obtuse angles in a tet mesh is still an open problem [Tournois et al. 2009], we compute quasi-eigenfunctions on a tetrahedral mesh using a quadratic finite element formulation, which is an extension of the linear finite element analysis on triangular mesh [Vallet and Lvy 2008; Reuter et al. 2009].

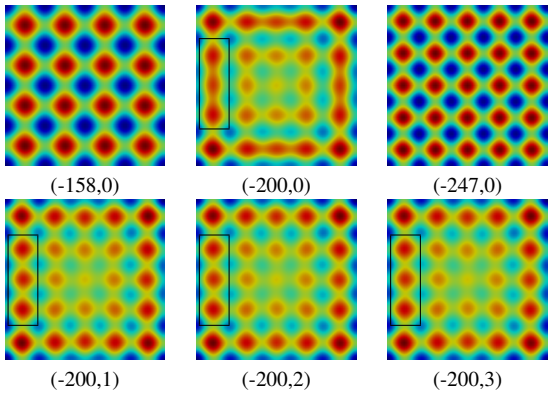


Fig. 7: *A few number of vibration enhancement iterations improves the distribution of the critical points when λ is not an eigenvalue. The numbers in parentheses indicate the value of λ and the number of iterations, respectively.*

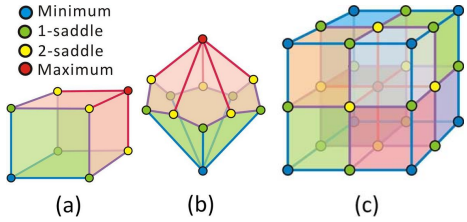


Fig. 8: *3D Morse-Smale complex: (a) A regular 3D Morse-Smale cell (3 pairs of saddles); (b) A general 3D Morse-Smale cell; (c) Eight neighboring Morse-Smale cells in a 3D Morse-Smale complex.*

We use the method in [Gyulassy et al. 2007] to extract the 3D Morse-Smale complex of a quasi-eigenfunction, and construct a hex mesh topologically by subdividing the Morse-Smale complex and geometrically smoothing it by iteratively updating its vertex positions to the centroid of neighboring hex elements. Unlike a 2D Morse-Smale complex, a 3D Morse-Smale complex contains two types of saddles, called 1-saddle and 2-saddle, respectively [Gyulassy et al. 2007] (see Fig. 8). An important point to note is that 3D Morse-Smale complex cells are not necessarily hexahedra (see Fig. 8(b)). Therefore a small portion of non-hex elements will remain in each non-hex MSC cell after subdivision. Hence, in general, our method generates a hex-dominant mesh, instead of a pure hex mesh.

5. RESULTS

We shall discuss experimental results in this section. To better appreciate the effect of the parameter λ on computing quasi-eigenfunction, we scale all two-dimensional (resp. three-dimensional) shapes to fit them in the box $[-1,1]^2$ (resp. $[-1,1]^3$). We also scale the density r so that its minimal value is 1. Note that our method does not restrict the λ to be an eigenvalue. Thus λ can be adjusted to cater for desired overall density.

5.1 Quadrangular Results

As the first example, we apply our method to computing a quad mesh of a square with the quad element size following the same density function as shown in Fig. 1. The quad mesh computed by our method, shown in (Fig. 9), has a much better singularity distribution than that produced by the method of [Zhang et al. 2010] (Fig. 1). Note that the direction field implied by our quad mesh in Fig. 9 is compatible with the given density field, and it would be difficult to compute such a direction field beforehand if it were needed for guiding the computation of the quad mesh as in the previous method [Zhang et al. 2010].

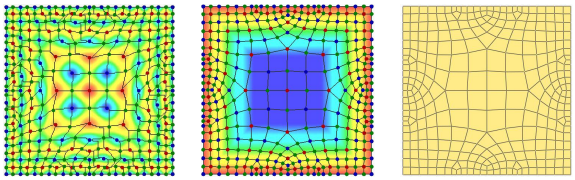
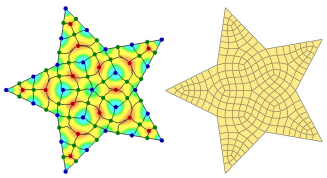


Fig. 9: *SQUARE. Left: QE ($\lambda = -1050$, no vibration enhancement) and its MSC. Middle: The density function; Right: The output quad mesh.*

The inset shows quasi-eigenfunction ($\lambda = -1200$, no vibration enhancement) and the resulting quad mesh computed by our method on the same star-shaped domain as Fig. 2. In comparison with the eigenfunctions shown Fig. 2, the Morse-Smale complex of this quasi-eigenfunction aligns well with the domain boundary.



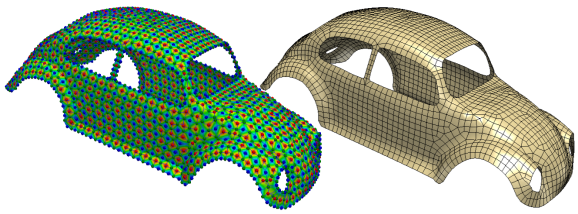
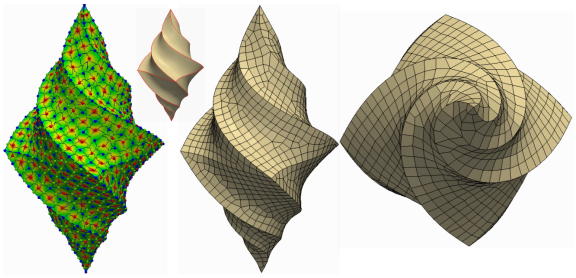


Fig. 10: *BEETLE. Left: QE ($\lambda = -8600$) and its MSC. Right: The output quad mesh.*

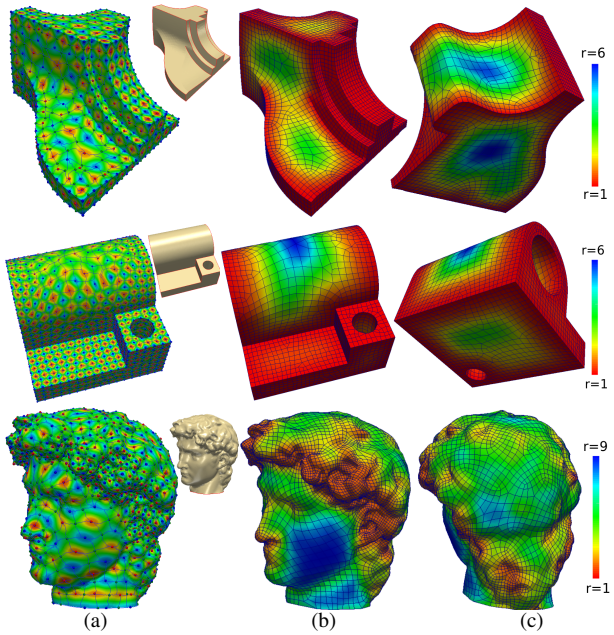
Fig. 10 shows the QE and its MSC computed by our method on an open surface with boundary curves. Fig. 11 shows the QE and its MSC computed by our method on the Spiral model with sharp feature curves. Fig. 12 demonstrates quad meshes computed by our method with adaptive element size control.



(a) (b) (c)

Fig. 11: *SPIRAL: (a) QE ($\lambda = -3600$) and its MSC with feature lines colored in red; (b) The output quad mesh; (c) The top view of (b).*

In the Fandisk model shown in Fig. 13, our method naturally introduces a valence-5 singularity at the meeting point of two feature curves that intersect at a small angle.



(a) (b) (c)

Fig. 12: *FANDISK (top, $\lambda = -2000$), JOINT (middle, $\lambda = -2400$) and David (bottom, $\lambda = -4600$) with feature lines colored in red: (a) QE and its MSC; (b) The output quad mesh and the density function; (c) The output quad mesh and the density function. The color coding of the density function is shown on the right.*

Although a guiding direction field is not needed for feature curve alignment, it can be integrated into our method to guide mesh orientation in regions away from boundary curves or features. Following the formulation of orientation energy E_{Orient} in [Huang et al. 2008], we minimize the combined energy $E(\mathbf{f}) = E_{Lr}(\mathbf{f}) + \gamma E_{Orient}(\mathbf{f})$ subject to Eq. (11). Fig. 14 shows the QE and its associated mesh with orientation control ($\gamma = 10^2$), which is clearly an improvement over the result by the method in [Huang et al. 2008] (Fig. 14(d)) in terms of boundary conformation. Here the guiding direction field is only applied to the back part of the CARHOOD model as shown in Fig. 14(a).

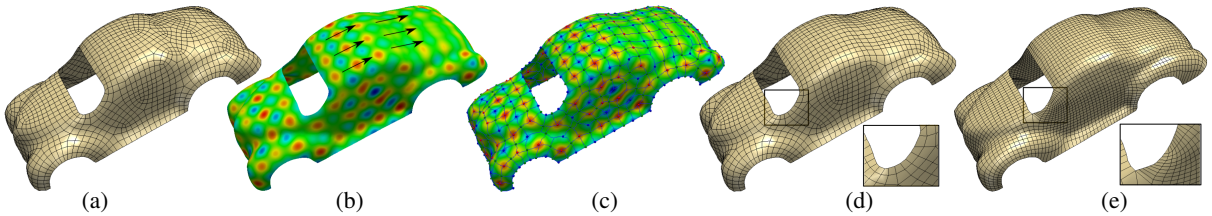


Fig. 14: *CARHOOD: (a) The quad mesh computed without using a guiding vector field; (b) The quasi-eigenfunction ($\lambda = -1400$) computed with the shown guiding direction field for improving mesh orientation; (c) The Morse-Smale complex from (b); (d) The output quad mesh of (b); (e) A quad mesh computed with the method in [Huang et al. 2008] for the same surface.*

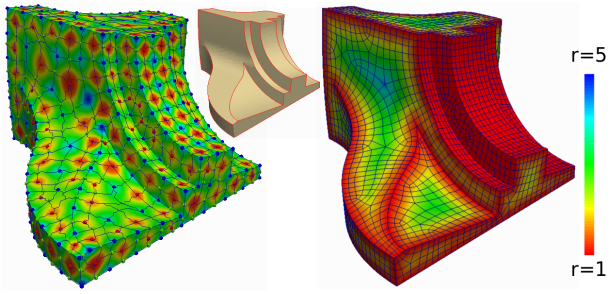


Fig. 13: *All sharp edges of the Fandisk are marked as feature curves. On the left is the quasi-eigenfunction and its MSC ($\lambda = -1900$) that we computed. On the right is the resulting quad mesh and the density field that is defined by the closed minimal distance of a point to the feature curves.*

Fig. 15 shows comparisons between our method and the method in [Pellenard et al. 2011]. For the uniform-density field on a quad domain, the quad mesh computed by our method contains no singularity (Fig. 15(a)), while their method produces unjustified singular points (Fig. 15(c)). With the same non-uniform density field, the quad mesh computed by our method has much fewer singular points (Fig. 15(b)) than that by their method (Fig. 15(d)).

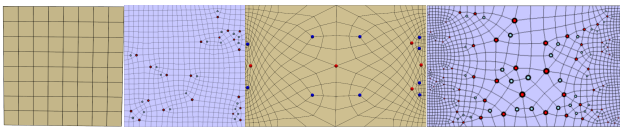


Fig. 15: *Our results ($\lambda = -280$ for a and c) contain fewer singularities than the results (b, d) from [Pellenard et al. 2011].*

Tab. I shows the timing data of the preceding experiments on a computer with an Intel Core i7-3770K CPU@3.50GHz CPU with 16GB RAM. The columns QE, MSC and Mesh show the time (in seconds) of computing the quasi-eigenfunctions, the Morse-Smale complexes, and the final quad meshes, respectively. Compared with cost of scalar field optimization in [Huang et al. 2008]. Because we need to solve a linear system based on the KKT condition and use non-linear optimization for vibration enhancement, our method is slower than the method in [Huang et al. 2008].

Model	Vert#	QE	MSC	Mesh
STAR (inset)	16k	2.4, NA	0.06	2.66
BUTTERFLY (Fig. 6 (b))	15k	1.5, 2.6	0.06	2.35
SQUARE (Fig. 9)	16k	3.0, NA	0.04	2.47
BEETLE (Fig. 10)	119k	56.4, 41.5	0.39	24.6
SPIRAL (Fig. 11)	26k	13.7, 6.6	0.12	3.11
FANDISK (Fig. 12)	13k	6.5, 4.1	0.05	1.17
JOINT (Fig. 12)	50k	31.7, 24.7	0.33	6.42
DAVID (Fig. 12)	84k	13.3, 50.8	0.40	15.3
CARHOOD (Fig. 14)	65k	11.7, 34.0	0.22	6.95

Table I. : *Quadrangular remeshing performance statistics (in seconds). The column of QE indicates the time of solving Eq.* (13) *for initial value and the time used for vibration enhancement.*

5.2 Robustness

To test the robustness of our method, we apply it to the CARHOOD model (with $\lambda = -1500$) with different density fields, as shown in Fig. 16. For the moderately varying density field in the first row, where the element size on the rear of the model is 10 times larger than that on the front, our method works well. For the more radically varying density field in the second row, where the element size on the rear of the model is 30 times larger than that on the front, our method fails because of the lacking of degree of freedom that can be provided by the relatively coarse input triangular mesh. In this case, refining input triangular mesh may alleviate the problem but that would also increase the condition number of the Laplace matrix, making the computation numerically unstable. Finally, the random density field on the third row leads to irregular distribution of the nodes in the Morse-Smale complex, from which we still extract a quad mesh robustly.

The global control of element size can be achieved by changing the value of λ, while fixing the density field. In Fig. 17, we apply different values λ to the sculpture model, with the same non-uniform density field. Note that, when the desired element size is too large or too small, the quad meshes extracted from the QEs may contain elements in bad shape.

Finally, we consider the limit of the complexity of feature curves that our method can accommodate. As shown by the examples in (Fig. 18), mesh singularities can be specified explicitly by manually introducing some feature curves (even in a smooth region of the surface). However, when such features curves get too complex or cluttered, no reasonable results can be expected because the discretization of the underlying input triangular mesh has no enough degree of freedom (DOF) for the quasi-eigenfunctions to meets all these constraints.

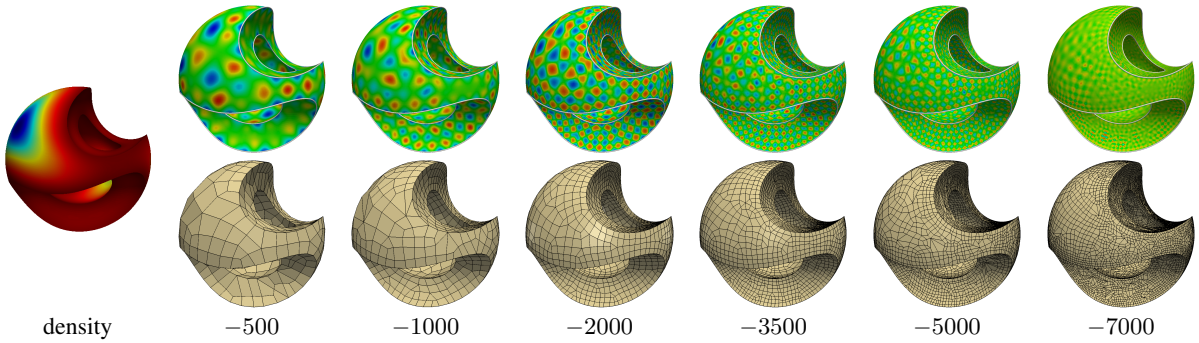


Fig. 17: *Quad meshes of the sculpture model with different values of λ and the same non-uniform density field.*

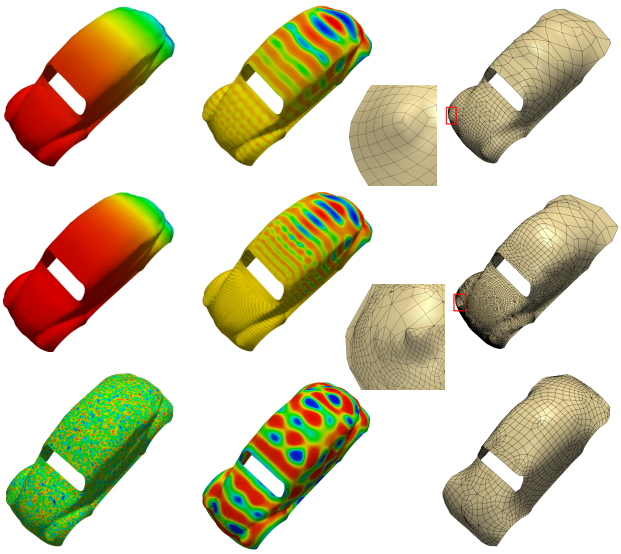


Fig. 16: *From left to right: density fields, quasi-eigenfunctions, and quad meshes. The elements on the rear of the model are* 10 *and* 30 *times larger than those on the front for the two density fields in the first and second rows, respectively. In the third row, the quad mesh is generated with a random density field with* $r \in [1, 5]$.

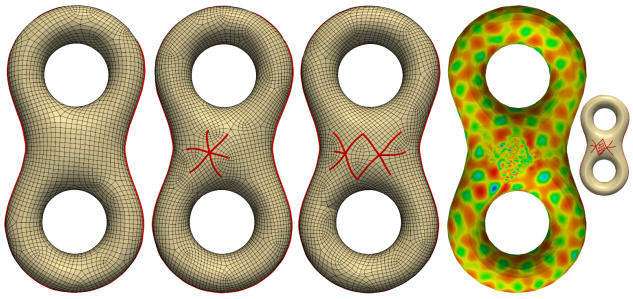


Fig. 18: *By manually specifying feature lines (in red), one can control the locations of some singular points explicitly (middle left, middle right). But when feature curves get too complex, no reasonable results can be produced due to the lack of DOF of the underlying input triangulation (the right most picture).*

5.3 Results on Hex-dominant Meshing

We now present some preliminary results of applying our spectral method to computing hex-dominant meshes of 3D volumes, with two examples shown in Fig. 19 and Fig. 20, respectively. For comparison, the meshing results by the octree method [Hexotic 2010] and the results by the advancing front method [Geompack++ 2010] are shown as well. Our method outperforms the other two methods in mesh quality in both examples. Here, the non-hex elements are shown in red for visualization. Our method can also process objects with genus larger than 0, as shown in Fig. 21. In all these examples, no sharp feature corner/lines are labeled a priori in the input tetrahedral mesh to assist hex mesh generation. These demonstrate the potential of our spectral method in automatic hex meshing.

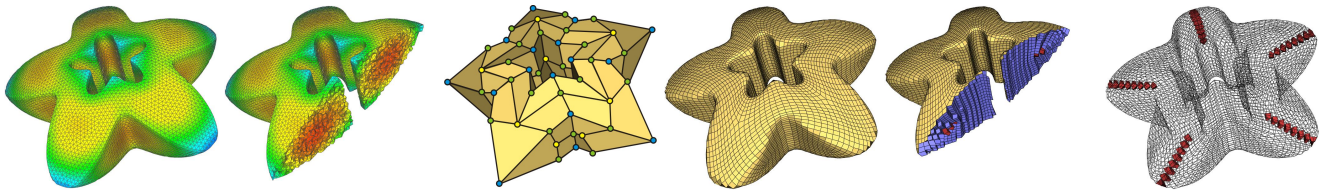
The major limitation of our spectral method for hex meshing is its high computational cost. The timing data for the three models are shown in Tab. II. The main cause for this inefficiency is that the linear system obtained from Eq. (13) based on the quadratic FEM formulation is much denser and larger than the linear system constructed with the cotangent formulation [Wang et al. 2003]. We used a 12-core 2.8GHz Intel server with 128GB RAM to solve these linear systems. Further research is needed to make the method more efficient to be practical for hex meshing.

Model	Vert#	QE	MSC	Mesh
TET	12k	21	0.4	0.1
TWISTED-CUBE	10k	15	0.3	0.1
DOUBLE-STAR	40k	179	1.2	0.8

Table II. : *Timing data for hexahedral meshing (in minutes).*

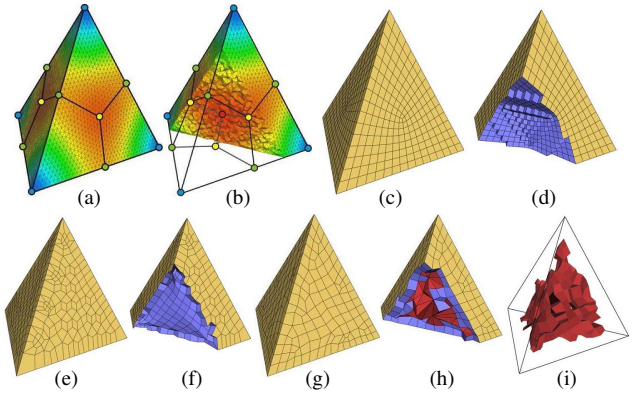
6. CONCLUSIONS

We have introduced a set of novel boundary conditions to ensure boundary conformation in the spectral approach to surface quadrangulation. Our boundary conditions eliminate the requirement of a guiding direction field, and thus provides more flexibility in designing mesh density functions to effective better element size control. Experiments showed that the quad meshes computed by our method capture domain boundaries and feature curves accurately and follow the given density field closely. Preliminary results are presented on extending our method to hex meshing.



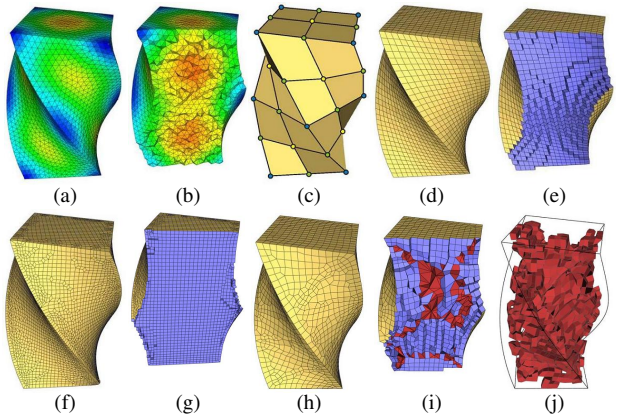
(a) (b) (c) (d) (e) (f)

Fig. 21: *DOUBLE-STAR: (a) QE ($\lambda = -450$); (b) Interior view of the QE; (c) The Morse-Smale complex; (d) Boundary mesh; (e) Interior mesh; (f) Non-hex elements.*



(a) (b) (c) (d) (e) (f) (g) (h) (i)

Fig. 19: *TET: (a) QE ($\lambda = -94$) with its MSC; (b) Interior view of the QE; (c) Boundary mesh; (d) Interior mesh; (e)-(f): The results by the octree method; and, (g)-(i): The results by the advancing front method.*



(a) (b) (c) (d) (e) (f) (g) (h) (i) (j)

Fig. 20: *TWISTED-CUBE: (a) QE ($\lambda = -140$); (b) Interior view of the QE; (c) The Morse-Smale complex; (d) Boundary mesh; (e) Interior meshing; (f)-(g): The results by the octree method; and, (h)-(j): The he results by the advancing front method.*

A major limitation of our method is its lack of explicit control of mesh singularities, which are implicitly determined by the quasi-eigenfunction. To properly capture the critical points in the quasi-eigenfunction, the element size of the input triangular mesh has to be smaller than the expected quad element size. This leads to the need for a highly refined input triangular mesh when a fine quad mesh is to be computed or there are a set of complex feature curves to accommodate. This is also the main cause of inefficiency when applying our method to hex meshing of 3D volumes. Future work is needed to improve this efficiency issue so the method can be used to compute quadrangulations of highly complex surface models and become practical for hexahedral meshing of 3D volumes.

Acknowledgements

We would like to thank Xinxin Shen and Tengfei Jiang for their help on implementing some of the experiments, and the anonymous reviewers for their valuable comments, and Dongming Yan, Stanford Scanning Repository for dataset.

The work of Wenping Wang was partially supported by the National Basic Research Program of China (2011CB302400), National Natural Science Foundation of China (61272019), and the Research Grant Council of Hong Kong (718209, 718010 and 718311).

The work of Jin Huang and Hujun Bao was partially supported by the NSFC (No.61170139, No.61210007), the National High Technology Research and Development (863) Program of China (No.2012AA011503) and the Fundamental Research Funds for the Central Universities (No. 2014XZZX006-08).

REFERENCES

ALLIEZ, P., COHEN-STEINER, D., DEVILLERS, O., LÉVY, B., AND DESBRUN, M. 2003. Anisotropic polygonal remeshing. *ACM Trans. Graph. 22,* 3, 485–493.

BOMMES, D., LÉVY, B., PIETRONI, N., PUPPO, E., SILVA, C., TARINI, M., AND ZORIN, D. 2012. State of the art in quad meshing. In *Eurographics STARS.*

BOMMES, D., ZIMMER, H., AND KOBBELT, L. 2009. Mixed-integer quadrangulation. *ACM Trans. Graph. 28,* 3 (July), 77:1–77:10.

BOYD, S. AND VANDENBERGHE, L. 2004. *Convex Optimization.* Cambridge University Press.

CAMPEN, M., BOMMES, D., AND KOBBELT, L. 2012. Dual loops meshing: quality quad layouts on manifolds. *ACM Trans. Graph. 31,* 4 (July), 110:1–110:11.

CHEN, Y., DAVIS, T. A., HAGER, W. W., AND RAJAMANICKAM, S. 2008. Algorithm 887: Cholmod, supernodal sparse cholesky factorization and update/downdate. *ACM Trans. Math. Softw. 35,* 3 (Oct.), 22:1–22:14.

DAVIS, T. A. 2004. Algorithm 832: UMFPACK V4.3—an unsymmetric-pattern multifrontal method. *ACM Transactions On Mathematical Software 30,* 2 (June), 196–199.

DONG, S., BREMER, P.-T., GARLAND, M., PASCUCCI, V., AND HART, J. C. 2006. Spectral surface quadrangulation. *ACM Trans. Graph. 25,* 3, 1057–1066.

DONG, S., KIRCHER, S., AND GARLAND, M. 2005. Harmonic functions for quadrilateral remeshing of arbitrary manifolds. *Comput. Aided Geom. Des. 22,* 5 (July), 392–423.

EDELSBRUNNER, H., HARER, J., AND ZOMORODIAN, A. 2003. Hierarchical morse-smale complexes for piecewise linear 2-manifolds. discrete and computational. *Discrete and Computational Geometry 30,* 1, 87–107.

GEOMPACK++. 2010. http://members.shaw.ca/bjoe/.

GREGSON, J., SHEFFER, A., AND ZHANG, E. 2011. All-hex mesh generation via volumetric polycube deformation. *Computer Graphics Forum (SGP) 30:5*, 1407–1416.

GYULASSY, A., BREMER, P.-T., HAMANN, B., AND PASCUCCI, V. 2011. Practical considerations in morse-smale complex computation. In *Topological Methods in Data Analysis and Visualization*. Springer, 67–78.

GYULASSY, A., NATARAJAN, V., PASCUCCI, V., AND HAMANN, B. 2007. Efficient computation of morse-smale complexes for three-dimensional scalar functions. *IEEE Trans. on Visualization and Computer Graphics 13,* 6, 1440–1447.

HEXOTIC. 2010. http://www-roc.inria.fr/gamma/gamma/logiciels/hexotic/index.html.

HORMANN, K., POLTHIER, K., AND SHEFFER, A. 2008. Mesh parameterization: theory and practice. In *ACM SIGGRAPH ASIA 2008 courses*. SIGGRAPH Asia '08. ACM, New York, NY, USA, 47:1–47:87.

HUANG, J., TONG, Y., WEI, H., AND BAO, H. 2011. Boundary aligned smooth 3d cross-frame field. *ACM Trans. Graph. 30,* 6 (Dec.), 143:1–143:8.

HUANG, J., ZHANG, M., MA, J., LIU, X., KOBBELT, L., AND BAO, H. 2008. Spectral quadrangulation with orientation and alignment control. *ACM Trans. Graph. 27,* 5 (Dec.), 147:1–147:9.

KALBERER, F., NIESER, M., AND POLTHIER, K. 2007. QuadCover-surface parameterization using branched coverings. *Computer Graphics Forum 26,* 3, 375–384.

KOVACS, D., MYLES, A., AND ZORIN, D. 2011. Anisotropic quadrangulation. *Computer Aided Geometric Design 28,* 8, 449–462.

LI, Y., LIU, Y., XU, W., WANG, W., AND GUO, B. 2012. All-hex meshing using singularity-restricted field. *ACM Trans. Graph. 31,* 6 (Nov.), 177:1–177:11.

LI, Y., WANG, W., LING, R., AND TU, C. 2011. Shape optimization of quad mesh elements. *Computers & Graphics 35,* 3, 444–451.

MARÉCHAL, L. 2009. Advances in octree-based all-hexahedral mesh generation: handling sharp features. In *Proceedings of the 18th International Meshing Roundtable*. 65–84.

MARINOV, M. AND KOBBELT, L. 2006. A robust two-step procedure for quad-dominant remeshing. *Computer Graphics Forum 25,* 3, 537–546.

MEYER, M., DESBRUN, M., SCHRÖDER, P., AND BARR, A. H. 2002. Discrete differential-geometry operators for triangulated 2-manifolds. In *Visualization and Mathematics III*. 35–57.

NI, X., GARLAND, M., AND HART, J. C. 2004. Fair morse functions for extracting the topological structure of a surface mesh. *ACM Trans. Graph. 23,* 3, 613–622.

NIESER, M., REITEBUCH, U., AND POLTHIER, K. 2011. CUBECOVER - parameterization of 3d volumes. *Computer Graphics Forum (SGP) 30:5*, 1397–1406.

PELLENARD, B., ALLIEZ, P., AND MORVAN, J.-M. 2011. Isotropic 2D Quadrangle Meshing with Size and Orientation Control. *Proceedings of the 20th International Meshing Roundtable*, 81–98. 18 pages.

PINKALL, U. AND POLTHIER, K. 1993. Computing discrete minimal surfaces and their conjugates. *Experimental Mathematics 2,* 1, 15–36.

RAY, N., LI, W. C., LÉVY, B., SHEFFER, A., AND ALLIEZ, P. 2006. Periodic global parameterization. *ACM Trans. on Graph. 25,* 4, 1460–1485.

REUTER, M., BIASOTTI, S., GIORGI, D., PATANE, G., AND SPAGNUOLO, M. 2009. Discrete laplace-beltrami operators for shape analysis and segmentation. *Computers & Graphics 33,* 3, 381–390.

SCHMIDT, R., GRIMM, C., AND WYVILL, B. 2006. Interactive decal compositing with discrete exponential maps. *ACM Trans. Graph. 25,* 3 (July), 605–613.

STATEN, M., OWEN, S., AND BLACKER, T. 2005. Unconstrained paving and plastering: A new idea for all hexahedral mesh generation. In *Proceedings of the 14th International Meshing Roundtable*, B. W. Hanks, Ed. 399–416.

TARINI, M., PUPPO, E., PANOZZO, D., PIETRONI, N., AND CIGNONI, P. 2011. Simple quad domains for field aligned mesh parametrization. *ACM Trans. Graph. 30,* 6 (Dec.), 142:1–142:12.

TONG, Y., ALLIEZ, P., COHEN-STEINER, D., AND DESBRUN, M. 2006. Designing quadrangulations with discrete harmonic forms. In *SGP '06: Proc. of the 4th Eurographics Symposium on Geometry Processing*. 201–210.

TOURNOIS, J., WORMSER, C., ALLIEZ, P., AND DESBRUN, M. 2009. Interleaving delaunay refinement and optimization for practical isotropic tetrahedron mesh generation. *ACM Trans. Graph. 28,* 3 (July), 75:1–75:9.

VALLET, B. AND LVY, B. 2008. Spectral geometry processing with manifold harmonics. *Computer Graphics Forum 27,* 2, 251–260.

WANG, Y., GU, X., AND YAU, S.-T. 2003. Volumetric harmonic map. *Communications in Information and Systems 3,* 3, 191–202.

WARDETZKY, M., MATHUR, S., KÄLBERER, F., AND GRINSPUN, E. 2007. Discrete laplace operators: no free lunch. In *SGP '07: Proc. of the 5th Eurographics Symposium on Geometry Processing*. 33–37.

YAN, D., LÉVY, B., LIU, Y., SUN, F., AND WANG, W. 2009. Isotropic remeshing with fast and exact computation of restricted voronoi diagram. *Computer Graphics Forums 28,* 5, 1445–1454.

ZHANG, M., HUANG, J., LIU, X., AND BAO, H. 2010. A wave-based anisotropic quadrangulation method. *ACM Trans. Graph. 29,* 4 (July), 118:1–118:8.