Constructing IGA-suitable planar parameterization from complex CAD boundary by domain partition and global/local optimization

Gang Xu, Ming Li, Bernard Mourrain, Timon Rabczuk, Jinlan Xu, Stéphane P.A. Bordas

Constructing IGA-suitable planar parameterization from complex CAD boundary by domain partition and global/local optimization

Gang Xu[a,b], Ming Li[c], Bernard Mourrain[d], Timon Rabczuk[e], Jinlan Xu[a], Stéphane P.A. Bordas[f]

[a] *School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou 310018, China*
[b] *Key Laboratory of Complex Systems Modeling and Simulation, Ministry of Education, Hangzhou 310018, China*
[c] *State Key Laboratory of CAD &CG, Zhejiang University, Hangzhou 310058, P.R China*
[d] *INRIA Sophia-Antipolis, 2004 Route des Lucioles, 06902 Cedex, France*
[e] *Institute of Structural Mechanics, Bauhaus-University Weimar, Marienstr. 15, D-99423 Weimar, Germany*
[f] *Research Unit in Engineering, University of Luxembourg, Luxembourg*

Abstract

In this paper, we propose a general framework for constructing IGA-suitable planar B-spline parameterizations from given complex CAD boundaries. Instead of the computational domain bounded by four B-spline curves, planar domains with high genus and more complex boundary curves are considered. Firstly, some pre-processing operations including Bézier extraction and subdivision are performed on each boundary curve in order to generate a high-quality planar parameterization; then a robust planar domain partition framework is proposed to construct high-quality patch-meshing results with few singularities from the discrete boundary formed by connecting the end points of the resulting boundary segments. After the topology information generation of quadrilateral decomposition, the optimal placement of interior Bézier curves corresponding to the interior edges of the quadrangulation is constructed by a global optimization method to achieve a patch-partition with high quality. Finally, after the imposition of C^1/G^1-continuity constraints on the interface of neighboring Bézier patches with respect to each quad in the quadrangulation, the high-quality Bézier patch parameterization is obtained by a local optimization method to achieve uniform and orthogonal iso-parametric structures while keeping the continuity conditions between patches. The efficiency and robustness of the proposed method are demonstrated by several examples which are compared to results obtained by the skeleton-based parameterization approach.

Keywords: isogeometric analysis, analysis-suitable planar parameterization, domain partition, global/local optimization

---

1. Introduction

In isogeometric analysis (IGA) [19], the parameterization of the computational domain corresponds to the mesh generation in finite element analysis. Consequently, it has a great effect on the subsequent accuracy and efficiency [8, 49, 38]. As proposed in [50], an analysis-suitable parameterization of the computational domain should satisfy three requirements: 1) it has no self-intersections, i.e, the mapping from the parametric domain to the physical domain should be injective; 2) the iso-parametric elements should be as uniform as possible; 3) the iso-parametric structure should be as orthogonal as possible. From the given boundary information with spline representations, several approaches such as constrained optimization methods [51], variational harmonic methods [52], divide-and-conquer techniques [47], the Teichmüller mapping method [32], skeleton-based decomposition method [54], multi-patch parameterization method [5], and parameterization with non-standard B-splines (i.e, T-splines [56], THB-splines [13], PHT/RHT-splines [34, 35, 36, 37], Powell-Sabin splines [43] and subdivision surfaces [39, 40]), have been proposed to address the parameterization problem of the computational domain. However, most of the above methods only focus on the computational domain with simple boundaries. The construction of analysis-suitable parameterizations of complex computational domains with standard B-spline boundaries remains one of the most significant challenges in IGA.

In this paper, a general framework for constructing IGA-suitable planar parameterization from complex CAD boundaries consisting of standard B-spline curves is proposed. Instead of the computational domain formed by four boundary curves, the planar domains with high genus and more complex boundary curves are considered. Our main contributions can be summarized as follows:

- We propose a general framework for IGA-suitable planar parameterizations from complex CAD boundaries by domain partition and global/local optimization.
- Given a complex planar domain bounded by B-spline curves, a novel method is proposed to construct four-sided patch partitions with high quality by global optimization and quad-meshing with few singularities.
- A C^1/G^1-constrained local optimization approach is proposed to construct the Bézier patch with respect to each quad in the quadrangulation achieving high-quality iso-parametric structures while keeping the continuity constraints between patches.

The rest of the paper is structured as follows. Some related work on parameterization of the computational domain are reviewed in Section 2. Preliminary properties of Bernstein polynomials and an overview of the proposed framework is given in Section 3. Several pre-processing operations, including Bézier extraction and Bézier subdivision, are presented in Section 4. Section 5 describes the topology information generation and interior Bézier boundary construction for the quadrilateral high-quality patch partition by the global optimization approach. A local optimization method for high-quality Bézier patch parameterizations with C^1/G^1 continuity constraints is proposed in Section 6. Some parameterization examples are presented in Section 7. To demsonstrate the effectiveness of the proposed method, the results are compared to results obtained by the skeleton-based approach. Finally, we conclude this paper and outline future work in Section 8.

2. Related work

Currently, the related work on parameterization of the computational domain in IGA can be classified into four categories: (1) analysis-aware optimal parameterization; (2) volumetric spline parameterization from boundary triangulation; (3) analysis-suitable planar parameterization; (4) analysis-suitable volumetric parameterization from spline boundaries .

Analysis-aware optimal parameterization: E. Cohen et al. [8] proposed the concept of *analysis-aware modeling*, in which the parameters of CAD models are selected to facilitate iso-geometric analysis. Xu et al. [49] showed that the quality of parameterization has a great impact on the analysis results and the efficiency. Pilgerstorfer and Jüttler [38] showed that the condition number of the stiffness matrix, which is a key factor for the stability of the linear system, depends strongly on the quality of the domain parameterization.

Volumetric spline parameterization from boundary triangulation: Using volumetric harmonic functions, Martin et al. [27] proposed a fitting method for triangular meshes by B-spline volumes. In [12], a method is proposed to construct trivariate T-spline parameterizations for genus-zero solids based on tetrahedral meshing and mesh untangling technique. Zhang et al. proposed an efficient approach to construct injective solid T-splines for genus-zero geometries from a boundary triangulation [55]. Chan et al. proposed a volumetric parameterization method with PHT-splines from the level-set boundary representation [7]. For meshes with arbitrary topology, volumetric parameterization methods are proposed from the Morse theory [46] and Boolean operations [25].

Analysis-suitable planar parameterization: The boundary in CAD is usually provided in spline form. Xu et al. proposed a constrained optimization method to construct injective planar parameterizations [50]. Gravessen et al. [15] investigated the planar parameterization problem within a general non-linear optimization framework, in which several objective functions related to the parameterization quality are introduced. Xu et al. proposed a skeleton-guided domain decomposition method for planar parameterization with C^0-continuity between patches [54]. Speleers and Manni proposed a parameterization method with C^1 Powell-Sabin splines defined on triangulation [43]; truncated hierarchical B-splines are used for the planar parameterization problems in [13]. Nian and Chen [32] proposed an approach for planar domain parameterization based on Teichmüller mapping, which can generate a bijective high-quality parameterization from four specified boundary curves. For the multi-patch parameterization problem, Buchegger and Jüttler [5] proposed a systematic method to explore the different possible parameterizations of a planar domain by collections of quadrilateral patches. For the geometrically continuous spline space over multi-patch domains, Kapl et al. proposed the construction of geometrically continuous isogeometric functions which are defined on two-patch domains [20]. Bases for bicubic and biquartic geometrically continuous isogeometric functions on bilinearly parameterized multi-patch domains are constructed in [21] ; Mourrain et al. analyzed the space of geometrically continuous piecewise polynomial splines for rectangular and triangular patches with arbitrary topology and general rational transition maps [28]; Buchegger et al. constructed the truncated hierarchical B-spline basis for the space of adaptively refined spline functions on multi-patch domains with enhanced smoothness across interfaces [4]. Overall, it is still an open problem how to construct analysis-suitable planar parameterization from complex CAD boundary.

Analysis-suitable volumetric parameterizations from spline boundary: A variational approach for constructing NURBS parameterizations of swept volumes is proposed by M. Aigner et al [1]. Xu et al. proposed a constrained optimization framework to construct analysis-suitable volume parameterizations [51]. Spline volume faring is proposed by Pettersen and Skytt to obtain high-quality volume parameterization [41]. The construction of conformal solid T-splines from boundary T-spline representations is studied by using octree structure and boundary offset [56]. In [52], a variational harmonic method is proposed to construct analysis-suitable parameterizations of computational domains from given CAD boundary information. Wang and Qian proposed an

efficient method by combining divide-and-conquer, constraint aggregation and the hierarchical optimization technique to obtain valid trivariate B-spline solids from six boundary B-spline surfaces [47]. Analysis-suitable trivariate NURBS representations of composite panels is constructed with a new curve/surface offset algorithm [31]. Xu et al. proposed a two-stage scheme to construct the analysis-suitable volumetric parameterization by a uniformity-improved boundary reparameterization method [53]. Recently, given a template domain, B-spline based consistent volumetric parameterization is proposed for a set of models with similar semantic features [48].

3. Preliminary properties of Bernstein polynomials and framework overview

3.1. Preliminary properties of Bernstein polynomials

Since some properties of Bernstein polynomials [17] will be applied in our framework, they are reviewed subsequently.

Lemma 3.1. *Product of Bernstein polynomials*

$$B_i^m(t)B_j^n(t) = \frac{\binom{m}{i}\binom{n}{j}}{\binom{m+n}{i+j}} B_{i+j}^{m+n}(t) \tag{1}$$

Lemma 3.2. *Integration of Bernstein polynomials*

$$\int_0^1 B_i^m(t)dt = \frac{1}{m+1} \tag{2}$$

Lemma 3.3. *Degree elevation of Bernstein polynomials*

$$B_i^{n-1}(t) = \frac{n-i}{n} B_i^n(t) + \frac{i+1}{n} B_{i+1}^n(t) \tag{3}$$

From Lemma 3.1, we have the following proposition [17].

Proposition 3.4. *Let $R(t)$ and $S(t)$ be a parametric function defined by*

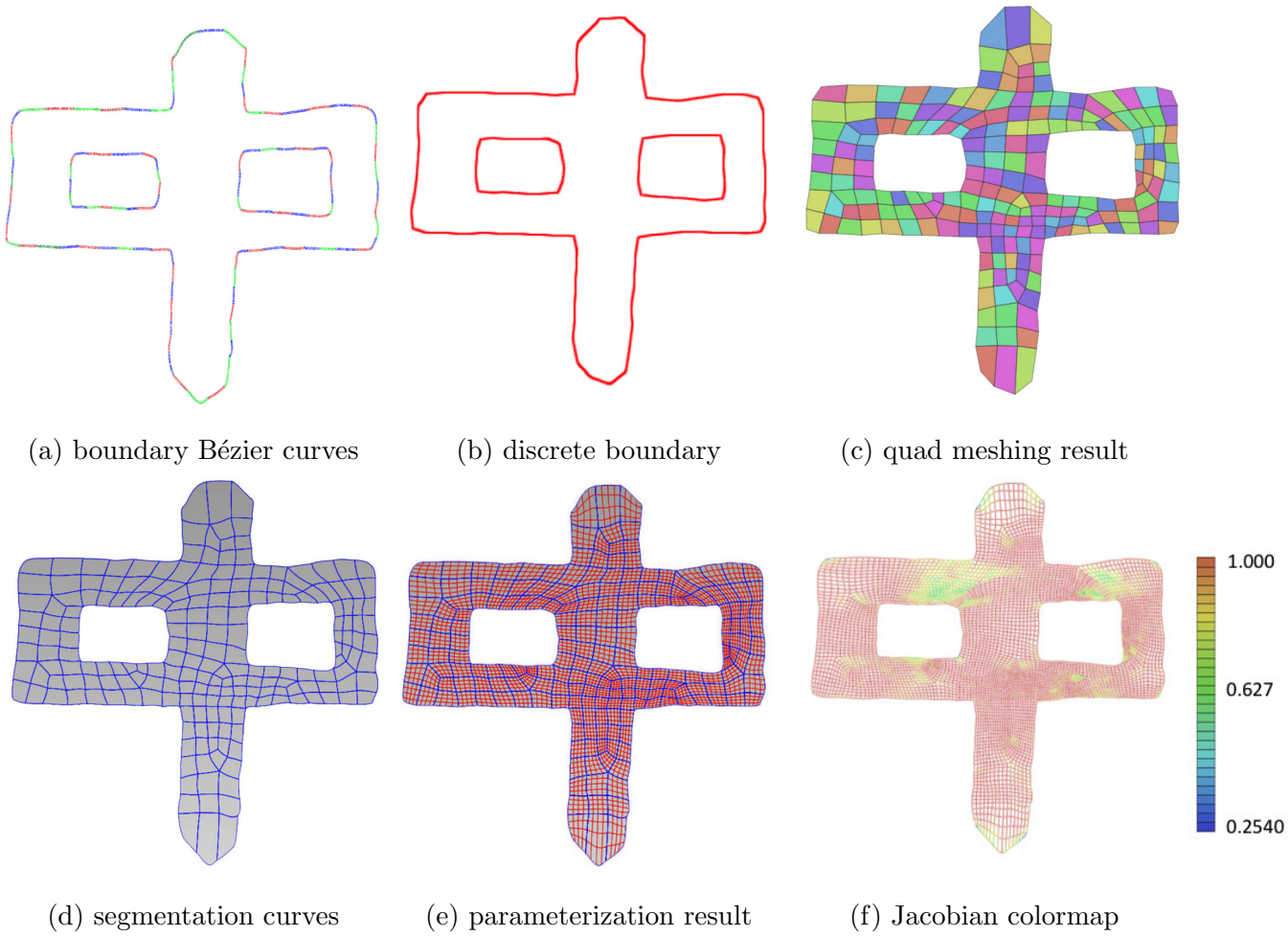
$$R(t) = \sum_{i=0}^{\ell_1} a_i B_i^{\ell_1}(t), \qquad S(t) = \sum_{i=0}^{\ell_2} b_i B_i^{\ell_2}(t),$$

where a_i and b_i are scalar values. Then the product of $R(t)$ and $S(t)$ can be defined as

$$R(t)S(t) = \sum_{i=0}^{\ell_1+\ell_2} c_i B_i^{\ell_1+\ell_2}(t), \tag{4}$$

where

$$c_i = \sum_{r=max(0,i-\ell_1)}^{min(i,\ell_2)} \frac{\binom{\ell_1}{r}\binom{\ell_2}{i-r}}{\binom{\ell_1+\ell_2}{i}} a_i b_{i-r}$$



(a) boundary Bézier curves (b) discrete boundary (c) quad meshing result

(d) segmentation curves (e) parameterization result (f) Jacobian colormap

Fig. 1. Illustration example I of the proposed parameterization framework. (a) shows the boundary Bézier curves after the pre-processing of input boundary B-spline curves; (b) presents the discrete boundary obtained by connecting the ending control points of each Bézier curve, and the corresponding quad meshing result is shown in (c). (d) shows the domain partition results with the construction of segmentation curves by global optimization method. After local optimization process for each sub-patch, the final planar parameterization result is illustrated in (e) with the iso-parametric curves; (f) presents the scaled Jacobian colormap of the parameterization to illustrate the quality of planar parameterization.

3.2. Framework overview

The problem investigated in this paper can be stated as follows: Given a planar region bounded by a set of B-spline curves, construct an analysis-suitable planar B-spline parameterization. In other words: How can several B-spline patches be constructued to fill the computational domain bounded by B-spline curves? In order to address the above problem, a general framework is proposed which consists of the following four steps illustrated in Fig. 1:

1. *Pre-processing for high-quality parameterization* (Section 4): In order to generate a high-quality planar parameterization, several pre-processing operations, including Bézier extraction and Bézier subdivision are performed on boundary curves (see Fig. 1(a)) ;
2. *Topology information generation of quadrilateral decomposition* (Section 5.1): From the discrete boundary obtained by connecting the ending control points of each Bézier curve as shown in Fig. 1(b), approximate convex decomposition and pattern based quad mesh generation are employed to generate the topology information of the quadrilateral decomposition of the computational domain;
3. *Construction of the quadrilateral patch partition by global optimization* (Section 5.2 and Section 5.3): After the topology partition is obtained, the Laplacian smoothing method is used to improve the quad mesh quality (Fig. 1(c)); then the optimal geometry of the interior B-spline curves corresponding to the interior edges of the quadrangulation are obtained by a global optimization method to achieve a high-quality patch-partition (Fig. 1(d)).
4. *High-quality patch parameterization by local optimization* (Section 6): The Bézier patch with respect to each quad in the quadrangulation is obtained by local optimization to achieve uniform and orthogonal iso-parametric structures while keeping the continuity conditions between patches; an approach for detecting and recovering of invalid Bézier patches is also proposed to guarantee the injectivity of the resulting parameterization (Fig. 1(e) and Fig. 1(f)).

4. Pre-processing of input boundary curves for high-quality parameterization

In order to achieve a high-quality parameterization of the computational domain, special treatment of input boundary curves are needed.

In IGA, the standard computational element is the sub-patch corresponding to the knot interval in the definition of the B-spline surface. In our parameterization framework, the input boundary B-spline curves will be segmented firstly with the Bézier extraction technique [3, 17], in which the piecewise B-spline representation is converted into Bézier form. The B-spline basis defined on a knot vector can be written as a linear combination of the Bernstein polynomials, that is,

$$\mathbf{N}(\mathbf{t}) = \mathbf{C}\mathbf{B}(\mathbf{t}) \tag{5}$$

where $\mathbf{C}$ denotes the Bézier extraction operator and $\mathbf{B}(\mathbf{t})$ are the Bernstein polynomials which are defined on $[0, 1]$. The conversion matrix $\mathbf{C}$ is sparse and its entries can be obtained by multiple knot insertion, which can be performed by Boehm's algorithm. Details on the Bézier extraction can be found in [3, 17].

With the conversion matrix $\mathbf{C}$, the Bézier extraction of the B-spline curves can be represented by

$$\mathbf{P} = \mathbf{CQ}, \tag{6}$$

in which $\mathbf{Q}$ are the control points of the B-spline curve, and $\mathbf{P}$ are the control points of the extracted Bézier curve.

After Bézier extraction, most of the resulting Bézier curves have an ideal shape for the parameterization. However, in some cases, there might be some Bézier curves with complex shapes, which require further division. In order to obtain satisfactory curve segments, we will perform the Bézier subdivision process based on the distance estimation between Bézier curves and its control polygon [45, 29]. Let L_{ave} be the average length of the lines connecting the starting control points and ending control points of all the extracted Bézier curves from the input boundary. The corresponding termination criterion for this Bézier subdivision process can be described as follows:

Termination criterion. The maximal distance D^k_{max} between the Bézier curve $S_k(t)$ and the straight line connecting its starting control point $(s^x_{0,k}, s^y_{0,k})$ and ending control point $(s^x_{n,k}, s^y_{n,k})$ is smaller than L_{ave}.

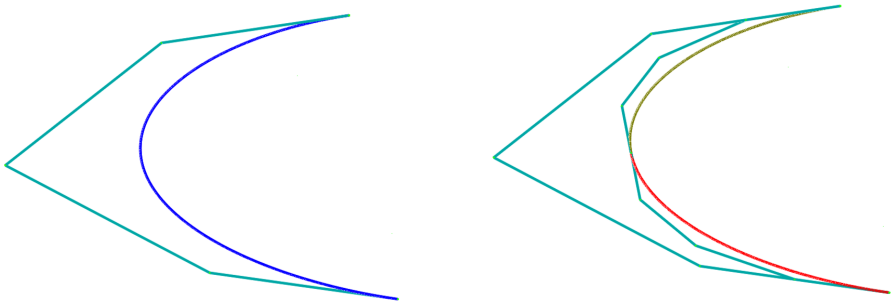
It should be mentioned that L_{ave} is determined by the input boundary and is kept unchanged during the subdivision process. From the above termination criterion, the Bézier curve will be subdivided into two Bézier segments at $t = 0.5$ if D^k_{max} is larger than L_{ave}. If $(s^x_{i,k}, s^y_{i,k})$, $i = 0, \ldots, n$ and

$$\eta = \max_{0 \le i \le n-2} \{|s^x_{i,k} - 2s^x_{i+1,k} + s^x_{i+2,k}|, |s^y_{i,k} - 2s^y_{i+1,k} + s^y_{i+2,k}|\},$$

then

$$\Gamma \ge \log_4 \frac{\sqrt{3}n(n-1)\eta}{8L_{ave}}, \tag{7}$$

where Γ is the number of times the Bézier curve must be subdivided in order to satisfy the termination criterion [45, 29]. Fig. 2 shows an example of the Bézier subdivision, in which a Bézier curve with concave shape is subdivided into two Bézier segments for the subsequent high-quality parameterization process.



(a) original Bézier curve (b) Bézier subdivision

Fig. 2. Subdivision of a Bézier curve with concave shape.

5. Four-sided partition of the computational domain by global optimization

After the pre-processing operation for the given boundary curves, we will now propose a global optimization method to construct the four-sided curved partition of the computational domain. Three steps are required to address this problem: Firstly, we generate the topology information of the quadrilateral decomposition. Afterwards, the Laplacian mesh smoothing method is employed to improve the quad mesh quality. Finally, the optimal shape of the interior B-spline curves corresponding to the interior edges of the quadrangulation are obtained by a global optimization method.

5.1. Topology generation of quadrilateral decomposition

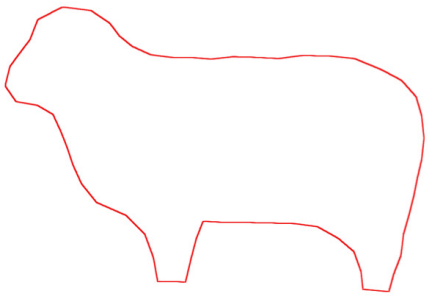
In this subsection, we will describe how to generate the topology information of quadrilateral decompositions from the boundary curves of the computational domain. The main steps include:

Step 1: Construct the discrete boundary by connecting the endpoints of the extracted Bézier curves obtained in Section 4 (see Fig. 3(a));

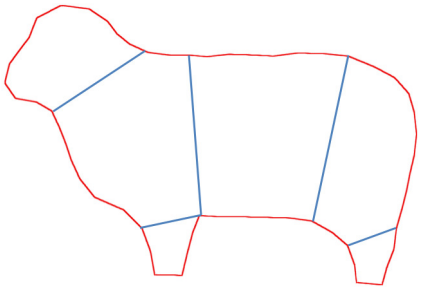
Step 2: Convert the multiply-connected regions into simply-connected regions as presented in Appendix I (see Fig. 4);

Step 3: Approximate the convex decomposition of the simply-connected regions by using the approach proposed in [24] (see Fig. 3(b));

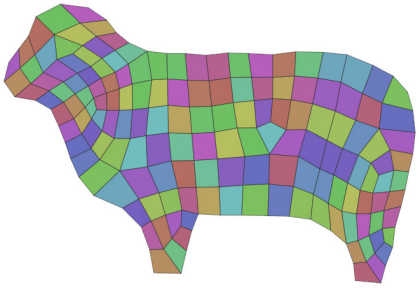
Step 4: For each quasi-convex polygon obtained in Step. 3, generate the quadrangulation topology information by the patterns proposed in [44] (see Fig. 3(c)).



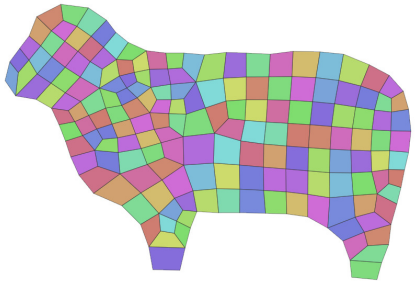
(a) input discrete boundary



(b) quasi-convex polygon decomposition



(c) quad-meshing result by our method with 147 elements and 14 irregular vertices



(d) meshing result by the method in [26] with 160 elements and 45 irregular vertices

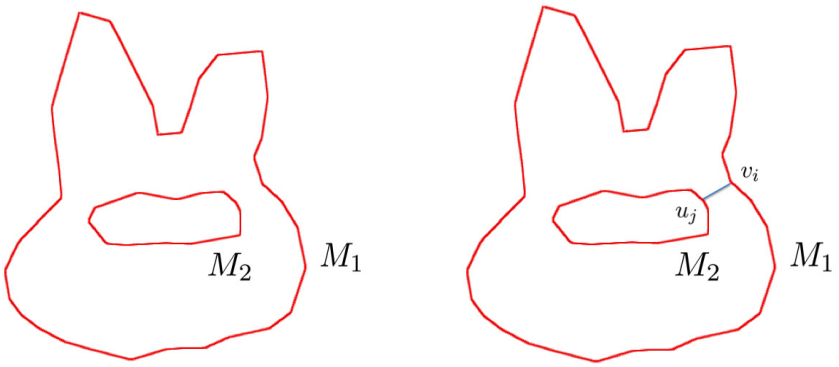
Fig. 3. Quad-meshing example.

The proposed method in [44] can produce patterns with minimal number of irregular vertices, which are constructed by solving a set of small integer linear programs. As shown in Fig. 3, our framework can generate high-quality quad-meshing results with fewer irregular vertices compared to the method in [26]. Furthermore, the patterns proposed in [44] only introduce irregular vertices with valence 3 or 5, which guarantees the solution existence for G^1 planar parameterization around the irregular vertex as shown in Section 6.1.3.

5.2. Laplacian smoothing method

Since the quad-meshing quality significantly affects the final parameterization result, we adapt an iterative Laplacian smoothing method to improve the quality of the quad mesh. In every iteration, each internal mesh vertex is moved to the centroid of its neighbor vertices, i.e.

$$x_i^k = \frac{\sum_{j=1}^{N_i} x_j^{k-1}}{N_i}, \qquad y_i^k = \frac{\sum_{j=1}^{N_i} y_j^{k-1}}{N_i} \tag{8}$$



(a) multiply-connected domain (b) simply-connected domain

Fig. 4. Convert a multiply-connected domain into simply-connected domain.

in which N_i is the number of neighbor vertices of the internal mesh vertex with location (x, y), and the superscript k is the iteration counter. This iteration is terminated according to the following termination rules:

$$\frac{[\sum_{i=1}^{m}[(x_i^k - x_i^{k-1})^2 + (y_i^k - y_i^{k-1})^2]]^{1/2}}{[\sum_{i=1}^{m}[(x_i^{k-1})^2 + (y_i^{k-1})^2]]^{1/2}} < \delta, \tag{9}$$

where m is the total number of mesh vertices, and δ is the specified tolerance value (0.001).

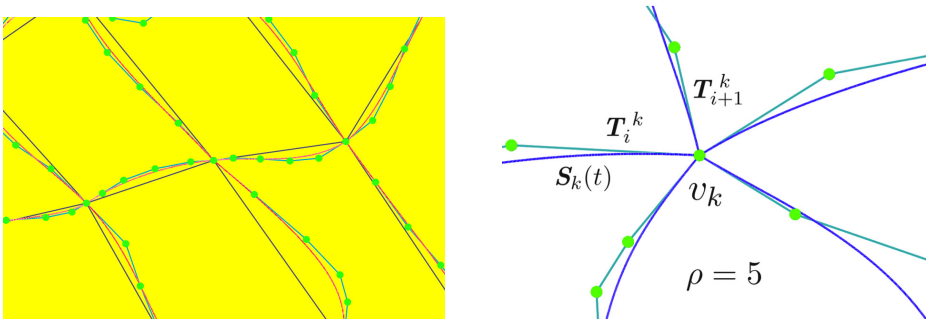
By the Laplacian smoothing method, we can improve the quad-mesh quality towards equilibrating the element size globally as shown in Fig. 3(c).

5.3. Construction of segmentation curves between patches

After constructing the quad mesh $Q(V, E)$ of the discrete computational domain, we construct the segmentation curves between patches corresponding to each quad. The segmentation curves interpolate two vertices on the quad mesh $Q(V, E)$ as shown in Fig. 5(a). Furthermore, for a planar computational domain, the shape of the segmentation curves has a great effect on the uniformity of the patch-size. Hence, we propose a global optimization method to construct the optimal shape of the segmentation curves.

There are three objective functions: The first term is related to the uniformity of the patch-size; the second term is related to the shape quality of the segmentation curves; the third term is related to the tangent constraints of the segmentation curves at quad-mesh vertices.

For the uniformity of the patch-size, we propose a new metric related to the patch-area. It is



(a) segmentation curves(red) w.r.t quad edges(black)

(b) segmentation curve at a singular vertex with $\rho = 5$

Fig. 5. Construction of segmentation curves.

obvious that the area of a planar region only depends on the shape of the boundary curves. In fact, for computing the area of the planar region bounded by Bézier curves, we have the following proposition.

Proposition 5.1. *For the planar region Ω bounded by N Bézier curves*

$$\boldsymbol{S}_k(t) = (S_k^x(t), S_k^y(t)) = \sum_{i=1}^{n} (s_{i,k}^x, s_{i,k}^y) B_i^n(t),$$

in which $(s_{i,k}^x, s_{i,k}^y)$ is the control point, $t \in [0,1]$, $k = 1, \cdots, N$, the area $A(\Omega)$ of the planar region Ω is the sum of integrals as follows:

$$A(\Omega) = \frac{1}{4n} \sum_{k=1}^{N} \sum_{j=0}^{2n-1} (c_j^k - d_j^k), \tag{10}$$

in which

$$c_j^k = \sum_{r=max(0,j-n)}^{min(j,n-1)} \frac{\binom{n}{r}\binom{n-1}{j-r}}{\binom{2n-1}{j}} s_{r,k}^x \, (s_{j-r+1,k}^y - s_{j-r,k}^y) \tag{11}$$

$$d_j^k = \sum_{r=max(0,j-n)}^{min(j,n-1)} \frac{\binom{n}{r}\binom{n-1}{j-r}}{\binom{2n-1}{j}} s_{r,k}^y \, (s_{j-r+1,k}^x - s_{j-r,k}^x) \tag{12}$$

Proposition 5.1 can be proved by Green's formula [6] and the properties of Bernstein polynomials, which are described in subsection 3.1.

From Proposition 5.1, we define the uniformity metric for the Bézier patches constructed from the quad mesh Q. The uniform patch structure over the quad mesh Q means that each patch

defined on each quad element in Q has the same area. In probability theory and statistics, the *variance* measures how far a set of numbers is spread out. A variance of zero indicates that all the values are identical. Hence, the uniformity of the patch structure requires the variance between each patch to be as small as possible. Suppose that A_i is the area of the i-th patch bounded by four Bézier curves, and A_{ave} is the average patch area in the patch structure Ω over the quad mesh Q, i.e.

$$A_{ave} = \frac{A(\Omega)}{L},$$

then the uniformity metric $F_{uniform}$ is defined as the variance of A_i,

$$F_{\text{uniform}} = \frac{1}{L}\sum_{i=0}^{L}(A_i - A_{ave})^2 \tag{13}$$

in which L is the number of Bézier patches in the patch structure Ω, $A(\Omega)$ is the area of the patch structure Ω bounded by the given set of B-spline curves. A_i and $A(\Omega)$ can be computed according to the area formula presented in Proposition 5.1.

For the term related to the shape quality of the segmentation curves, the corresponding objective function is defined as a combination of the stretch energy and the strain energy,

$$F_{\text{shape}} = \sum_{k=0}^{N}\int_0^1 \sigma_1 \|\boldsymbol{S}_k^{'}(t)\|^2 + \sigma_2 \|\boldsymbol{S}_k^{''}(t)\|^2 dt, \tag{14}$$

in which σ_1 and σ_2 are positive weights. If $\sigma_1 > \sigma_2$, then the resulting segmentation curves have smaller stretch energy, which measures the length of a curve. If $\sigma_2 > \sigma_1$, we obtain segmentation curves with smaller strain energy, which is a measures of the curve's bending.

For the term related to the tangent vector requirements of segmentation curves at the quad-mesh vertex v_k with valence ρ, we define the following objective function,

$$F_{\text{tangent}} = \sum_{k=0}^{N}\sum_{i=1}^{\rho}\left(\frac{\boldsymbol{T}_i^k \cdot \boldsymbol{T}_{i+1}^k}{\|\boldsymbol{T}_i^k\|\|\boldsymbol{T}_{i+1}^k\|} - \cos\frac{2\pi}{\rho}\right)^2 \tag{15}$$

where $\boldsymbol{T}_i^k$ is the tangent vector of the segmentation curve $\boldsymbol{S}_k(t)$ at the quad-mesh vertex v_k, and $\boldsymbol{T}_{\rho+1}^k = \boldsymbol{T}_1^k$ as shown in Fig. 5(b). At regular vertices with valence $\rho = 4$, the minimization of F_{tangent} will achieve quasi-C^1 and quasi-orthogonal segmentation curves, which is a basic requirement for analysis-suitable parameterization.

Combining the optimization terms defined in (13), (14) and (15), we construct an objective function F as follows,

$$F = \omega_1 F_{\text{uniform}} + \omega_2 F_{\text{shape}} + \omega_3 F_{\text{tangent}}, \tag{16}$$

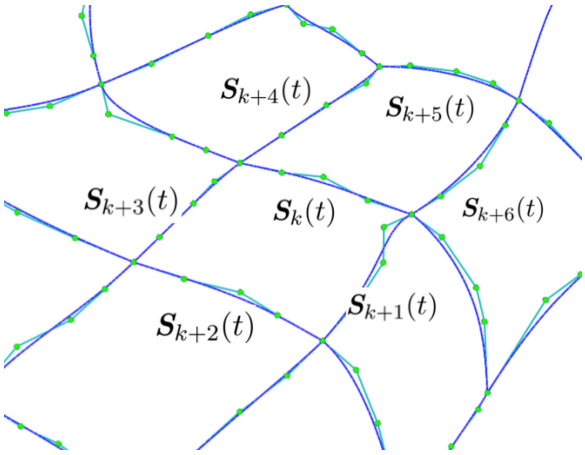


Fig. 6. Interior segmentation curve $\boldsymbol{S}_k(t)$ and its neighboring segmentation curves.

in which ω_1, ω_2 and ω_3 are positive weights for the balance between the shape quality metric, uniformity metric and the metric for tangent constraints. In order to achieve quasi-C^1 and quasi-orthogonal segmentation curves, ω_3 is usually much larger than ω_1 and ω_2.

The segmentation curves are obtained by solving the non-linear optimization problem

$$\arg\min_{\boldsymbol{s}_{i,k}} \quad F, \tag{17}$$

in which the design variables are the control points of the segmentation curves $\boldsymbol{S}_k(t)$.

The scale of the optimization problem in (17) depends on the number and the degree of the Bézier segmentation curves. The Limited memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) method [33] is adopted to obtain the optimal solution, which is a quasi-Newton method to solve unconstrained nonlinear minimization problems. In the L-BFGS method, we approximate the inverse Hessian matrix of the objective function in (17) by a sequence of gradient vectors from previous iterations. For more details, the reader can refer to [33].

Remark 5.1. The interior segmentation curves are called *valid* if there are no intersections between them except the connecting control points. That is, for each interior segmentation curve $\boldsymbol{S}_k(t)$, $k = 1, \cdots, N$, it has no intersections with its neighboring segmentation curves $\boldsymbol{S}_{k+1}(t)$, $\boldsymbol{S}_{k+2}(t)$, $\boldsymbol{S}_{k+3}(t)$, $\boldsymbol{S}_{k+4}(t)$, $\boldsymbol{S}_{k+5}(t)$ and $\boldsymbol{S}_{k+6}(t)$ as shown in Fig. 6. In particular, we can derive the following sufficient conditions such that $\boldsymbol{S}_k(t)$ has no intersections with $\boldsymbol{S}_{k+j}(t)$:

$$\triangle \boldsymbol{s}_{i-1}^k \boldsymbol{s}_i^k \boldsymbol{s}_{i+1}^k \cap \triangle \boldsymbol{s}_{l-1}^{k+j} \boldsymbol{s}_l^{k+j} \boldsymbol{s}_{l+1}^{k+j} = \emptyset, \quad i = 1, \cdots, n, \quad l = 1, \cdots, n, \quad j = 1, \cdots, 6.$$

in which $\boldsymbol{s}_i^k$ is the control point of segmentation curve $\boldsymbol{S}_k(t)$. That is, if the triangles $\triangle \boldsymbol{s}_{i-1}^k \boldsymbol{s}_i^k \boldsymbol{s}_{i+1}^k$ formed by the control points of $\boldsymbol{S}_k(t)$ do not intersect with the other triangles $\triangle \boldsymbol{s}_{l-1}^{k+j} \boldsymbol{s}_l^{k+j} \boldsymbol{s}_{l+1}^{k+j}$ formed by the control points $\boldsymbol{S}_{k+j}(t)$, then $\boldsymbol{S}_k(t)$ has no intersections with $\boldsymbol{S}_{k+j}(t)$. These sufficient conditions can be integrated with the optimization problem (17) to generate valid interior segmentation curves.

6. High-quality patch parameterization by local constrained optimization

After the four-sided patch partition is generated, we next construct the inner control points for each Bézier patch. As shown in [10], the high-order continuity of the basis function for parameterization and solution field is one of the main advantages of isogeometric analysis compared with the traditional finite element method, especially for thin shell analysis [34] and weakly nonlocal continuum models [2]. Moreover, for multi-patch parameterizations, G^k constructions yield C^k-continuous isogeometric elements [16], which can achieve fast convergence at the irregular vertex. From above motivations, in this paper, we propose an efficient patch-wise local optimization method to construct high-quality patch parameterizations with C^1/G^1-continuity constraints. The main framework for each local patch construction is summarized as follows:

Step 1: Construct the boundary control points on the second layer of the control mesh for each patch by orthogonality optimization and impose continuity constraints to achieve a near-orthogonal iso-parametric structure at common segmentation curves as described in subsection 6.1;

Step 2: Construct interior $(n-3)\times(n-3)$ control points for each patch by solving a linear system related to efficient C^1 energy-minimizing scheme as presented in subsection 6.2;

Step 3: Find the invalid patches on the parameterization and recover the patch validity by repositioning the control points by a local optimization method as described in subsection 6.3.

6.1. Construction of boundary second-layer control points with orthogonality optimization and continuity constraints

In this section, we will construct the boundary control points on the second layer of the control mesh, i.e. $\boldsymbol{P}_{n-1,j}$, $\boldsymbol{P}_{1,j}$, $\boldsymbol{P}_{i,1}$, and $\boldsymbol{P}_{i,n-1}$, to satisfy the orthogonality and continuity requirements. The construction procedures involves:

Step 1: *Initial construction*: Construct the initial control points on the second layer of the control mesh for each patch by orthogonality optimization as described in subsection 6.1.1;

Step 2: *Imposition of C^1-continuity*: In the regular region, adjust the boundary control points on the second layer of the control mesh for each patch to achieve C^1-continuity. Therefore, the Lagrange-multiplier method is employed as proposed in subsection 6.1.2 ;

Step 3: *Imposition of G^1-continuity*: In the irregular region, adjust the 1-neighbor control points around the irregular vertex to satisfy the general G^1-continuity constraints by solving linear system as described in subsection 6.1.3.

In the following subsections, some details will be described for each step.

6.1.1. Initial construction by orthogonality optimization

Firstly, we will describe the initial construction by orthogonality optimization. Without loss of generality, we firstly construct the initial position $\boldsymbol{P}^0_{n-1,j}$ of $\boldsymbol{P}_{n-1,j}$ for the Bézier patch $\boldsymbol{r}(u,v)$:

$$\boldsymbol{P}^0_{n-1,j} = \boldsymbol{P}_{n,j} + \frac{(\boldsymbol{P}_{0,j} - \boldsymbol{P}_{n,j})}{n},$$

In order to achieve a near-orthogonal isoparametric structure on the boundary $\boldsymbol{r}(1,v)$, the final position of $\boldsymbol{P}_{n-1,j}$ is obtained by solving the following optimization problem

$$\arg\min_{\boldsymbol{P}_{n-1,j}} \int_0^1 (< \boldsymbol{r}_{1,u}(1,v), \boldsymbol{r}_{1,v}(1,v) >)^2 dv \tag{18}$$

in which

$$\boldsymbol{r}_{1,u}(1,v) = n \sum_{j=0}^{n} B_l^n(v) \Delta^{1,0} \boldsymbol{P}_{n-1,l}, \tag{19}$$

$$\boldsymbol{r}_{1,v}(1,v) = n \sum_{j=0}^{n-1} B_l^{n-1}(v) \Delta^{0,1} \boldsymbol{P}_{n,l}, \tag{20}$$

Solving the similar optimization problems as (18) yields the position of $\boldsymbol{P}_{1,j}$, $\boldsymbol{P}_{i,1}$ and $\boldsymbol{P}_{i,n-1}$ for each patch, which requires adjustment to achieve C^1 continuity.

6.1.2. C^1 construction in the regular region

In order to ensure that the two joint Bézier patches $\boldsymbol{r}^k_1(u,v)$ and $\boldsymbol{r}^k_2(u,v)$ satisfy C^1 continuity in the regular region, the control mesh near the common segmentation curve $\boldsymbol{S}_k(t)$ with control points $\boldsymbol{s}^k_j$ should satisfy the following C^1 conditions:

$$\boldsymbol{s}^k_j - \boldsymbol{P}^k_j = \boldsymbol{Q}^k_j - \boldsymbol{s}^k_j, \qquad k = 0, \cdots, N, \quad j = 0, \cdots, n \tag{21}$$

in which N is the number of segmentation curves, $\boldsymbol{P}_j^k$ and $\boldsymbol{Q}_j^k$ are the one-neighbor control points of the joint Bézier patches along the segmentation curve $\boldsymbol{S}_k(t)$ (see Fig. 7).

The problem to impose C^1-continuity can be formulated as follows: Minimize the change of related control points along the segmentation curves (except the one-neighbor control points around the irregular vertex) such that they satisfy the C^1-constraints. In the following, we will solve this constrained optimization problem using the Lagrange multiplier method.

In order to minimize the change of all the related control points along the segmentation curves, the optimization term is defined as

$$\text{Min} \sum_{k=1}^{N} \sum_{j=0}^{n} (\| \boldsymbol{P}_j^k - \bar{\boldsymbol{P}}_j^k \|^2 + \| \boldsymbol{Q}_j^k - \bar{\boldsymbol{Q}}_j^k \|^2) \tag{22}$$

in which N is the number of segmentation curves, $\bar{\boldsymbol{P}}_j^k$ and $\bar{\boldsymbol{Q}}_j^k$ are the initial control points constructed by the approach in subsection 6.1.1.

Combining (21) with (22), the Lagrange function is defined as

$$L = \sum_{i=0}^{N} \sum_{j=0}^{n} (\| \boldsymbol{P}_j^k - \bar{\boldsymbol{P}}_j^k \|^2 + \| \boldsymbol{Q}_j^k - \bar{\boldsymbol{Q}}_j^k \|^2) + \sum_{i=0}^{N} \sum_{j=0}^{n} \lambda_{k,j} (2\boldsymbol{s}_j^k - \boldsymbol{P}_j^k - \boldsymbol{Q}_j^k) \tag{23}$$

where $\lambda_{k,j} = [\lambda_{k,j}^x, \lambda_{k,j}^y]^{\mathrm{T}}$ are Lagrange multipliers.

The unknown variables in the constrained optimization problem (23) are the control points $\boldsymbol{P}_j^k = [p_j^{k,x}, p_j^{k,y}]$ and $\boldsymbol{Q}_j^k = [q_j^{k,x}, q_j^{k,y}]$. A necessary condition for $\boldsymbol{P}_j^k$, $\boldsymbol{Q}_j^k$ and $\lambda_{k,j}$ to be a solution of (23) is that the corresponding partial derivatives vanish, that is,

$$\begin{cases} \dfrac{\partial L}{\partial \lambda_{k,j}} = 2\boldsymbol{s}_j^k - \boldsymbol{P}_j^k - \boldsymbol{Q}_j^k = 0, \quad k = 0, \cdots, N, \quad j = 0, \cdots, n; \\ \dfrac{\partial L}{\partial p_j^{k,r}} = 0, \quad r = x, y; \\ \dfrac{\partial L}{\partial q_j^{k,r}} = 0, \quad r = x, y \end{cases} \tag{24}$$

Methods such as Gauss elimination can be employed to solve the linear system (24).

The proposed least-square scheme leads to the boundary control points on the second layer of control mesh for each Bézier patch, which satisfies the C^1-continuity requirements.

6.1.3. G^1 construction around irregular vertex

For the irregular vertices $\boldsymbol{P}_{00}$ with non-four-valence on the quad mesh, some special treatments should be done in order to achieve G^1 continuity at $\boldsymbol{P}_{00}$; G^1 continuity means C^1-continuity of the

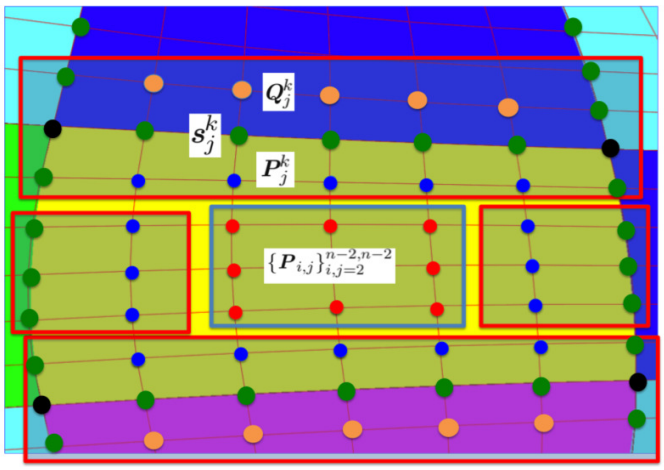


Fig. 7. Control points involved in the local optimization process: The blue control points $\boldsymbol{P}_j^k$ and the orange control points $\boldsymbol{Q}_j^k$ are determined by the C^1-continuity constraints with the green control points $\boldsymbol{s}_j^k$ of segmentation curves $\boldsymbol{S}_k(t)$ as proposed in subsection 6.1.2 ; the red control points $\{\boldsymbol{P}_{i,j}\}_{i,j=2}^{n-2,n-2}$ are determined by the local linear-energy-minimizing method as described in subsection 6.2.

parameterization maps through the composition with transition maps across the common edges of the patches. It differs from the G^1 continuity of the (planar) surfaces, which are the image of these parameterizations.

According to the pattern-based construction method of topology information described in Section 5.1, there are only irregular vertices of valence 3 or 5. Without loss of generality as shown in Fig. 8, suppose that there are M Bézier patches $\boldsymbol{r}_i(u,v)$ of degree $n \times n$ meeting at the common control points $\boldsymbol{P}_{00}$, $i = 1, 2, \cdots, M$ ($M = 3$ or $M = 5$). With the Bézier segmentation curves $\boldsymbol{S}_i(t)$ constructed in Section 5.3, we construct the control points $\boldsymbol{P}_{11}^i$, which are nearest to the irregular vertex $\boldsymbol{P}_{00}$ among the inner control points of $\boldsymbol{r}_i(u,v)$, to satisfy the general G^1 continuity constraints. Specifying the transition map in the G^1 conditions proposed in [28], the G^1 continuity constraints around the irregular vertex $\boldsymbol{P}_{00}$ can be obtained as follows,

$$(\boldsymbol{s}_1^i - \boldsymbol{P}_{00}) = \alpha_i(\boldsymbol{s}_1^{i+1} - \boldsymbol{P}_{00}) + \beta_i(\boldsymbol{s}_1^{i-1} - \boldsymbol{P}_{00}), \tag{25}$$

$$\boldsymbol{0} = n\alpha_i(\boldsymbol{P}_{11}^i - \boldsymbol{s}_1^i) + n\beta_i(\boldsymbol{P}_{11}^{i-1} - \boldsymbol{s}_1^i) - (n-1)(\boldsymbol{s}_2^i - \boldsymbol{s}_1^i) + (\boldsymbol{s}_1^i - \boldsymbol{P}_{00}), i = 1, 2, \ldots, M \tag{26}$$

Eq.(25) and Eq. (26) can be deduced from Eq. (13) and the system after Eq. (22) in [28], where the derivatives for the functions are expressed in terms of the control points of the Bézier patches. Similar G^1-continuity constraints can be found in [22, 23].

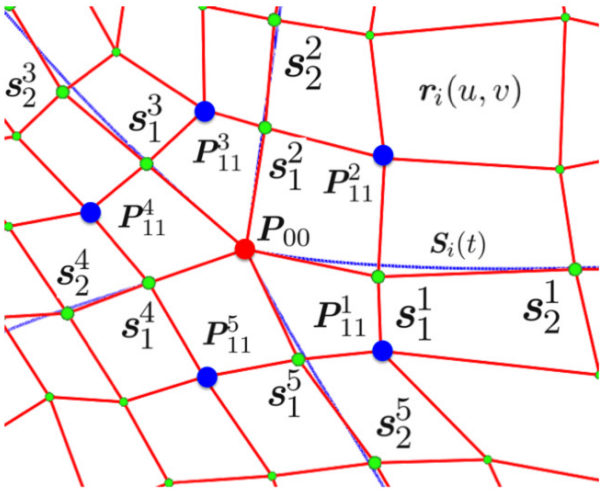


Fig. 8. Given the control points $\boldsymbol{s}_j^i$ of segmentation curves $\boldsymbol{S}_i(t)$, construct 1-neighbor control points $\boldsymbol{P}_{11}^i$, around irregular vertex $\boldsymbol{P}_{00}$ to achieve G^1-continuity on planar parameterization, $i = 1, 2, \cdots, 5, \quad j = 0, 1, \cdots, n$.

The control points $\boldsymbol{s}_j^i$ of the segmentation curves are fixed, $i = 1, 2, \cdots, M$, $j = 0, 1, \cdots, n$. In order to satisfy the first condition (25) of G^1 continuity, we firstly determine the values of α_i and β_i by solving a 2×2 linear system (25) for a specified i. Then, the unknown control points $\boldsymbol{P}_{11}^i$ are determined according to the second condition (26), which can be rewritten as a linear system,

$$\begin{pmatrix} \alpha_1 & 0 & \dots & 0 & \beta_1 \\ \beta_2 & \alpha_2 & \dots & 0 & 0 \\ 0 & \beta_3 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & \alpha_{M-1} & 0 \\ 0 & 0 & \dots & \beta_M & \alpha_M \end{pmatrix} \begin{pmatrix} \boldsymbol{P}_{11}^1 \\ \boldsymbol{P}_{11}^2 \\ \boldsymbol{P}_{11}^3 \\ \vdots \\ \boldsymbol{P}_{11}^{M-1} \\ \boldsymbol{P}_{11}^M \end{pmatrix} = \begin{pmatrix} \boldsymbol{H}_1 \\ \boldsymbol{H}_2 \\ \boldsymbol{H}_3 \\ \vdots \\ \boldsymbol{H}_{M-1} \\ \boldsymbol{H}_M \end{pmatrix} \tag{27}$$

in which $\boldsymbol{H}_i = (\alpha_i + \beta_i - 1)\boldsymbol{s}_1^i + (1 - \frac{1}{n})\boldsymbol{s}_2^i + \frac{1}{n}\boldsymbol{P}_{00}, \quad i = 1, 2, \cdots, M.$

Proposition 6.1. *The coefficient matrix involved in the linear system* (27) *is invertible for* $M = 3$ *and* $M = 5$*, i.e, there exists unique solution of the linear system* (27)*.*

Proof. The determinant of the $M \times M$ matrix in the linear system (27) is, for $M = 3$ and $M = 5$,

$$\prod_{i=1}^{M} \alpha_i + \prod_{i=1}^{M} \beta_i. \tag{28}$$

in which α_i and β_i are determined from (25).

From the vertex enclosure formulation with G^1-continuity [18], we have

$$\prod_{i=1}^{M}\alpha_i=\prod_{i=1}^{M}\beta_i.$$

From (28), the $M\times M$ matrix in the linear system (27) is invertible, i.e. the linear system (27) has a unique solution. □

6.2. Local C^1 linear-energy-minimizing method for constructing inner control points

In this section, we will propose a local C^1 linear-energy-minimizing method for constructing interior $(n-3)\times(m-3)$ control points of each patch with prescribed boundary two-layer control points as shown in Fig.7. This problem can be stated as follows: Given the boundary control points on the first two layers of the control mesh, i.e. $\boldsymbol{P}_{0,j}$,$\boldsymbol{P}_{1,j}$,$\boldsymbol{P}_{n-1,j}$ $\boldsymbol{P}_{n,j}$, $\boldsymbol{P}_{i,0}$, $\boldsymbol{P}_{i,1}$, $\boldsymbol{P}_{i,n-1}$ and $\boldsymbol{P}_{i,n}$ of a tensor Bézier patch $\boldsymbol{r}(u,v)$, find the remaining interior control points $\{\boldsymbol{P}_{i,j}\}_{i,j=2}^{n-2,n-2}$, such that the following energy function $E(\boldsymbol{r})$ is minimal,

$$E(\boldsymbol{r})=\int_{\Omega}\tau_1(\|\boldsymbol{r}_u\|^2+\|\boldsymbol{r}_v\|^2)+\tau_2(\|\boldsymbol{r}_{uu}\|^2+2\|\boldsymbol{r}_{uv}\|^2+\|\boldsymbol{r}_{vv}\|^2)dudv. \tag{29}$$

The energy function $E(\boldsymbol{r})$ is related to the orthogonality and uniformity of the iso-parametric structure on the Bézier surface [52]. τ_1 and τ_2 are positive weights to control the parameterization results: if τ_1 is big, then the resulting iso-parametric structure has better orthogonality; if τ_2 has a small value, then we can obtain an iso-parametric grid with better uniformity.

Different from the non-linear optimization method [52], with prescribed boundary two-layer control points, we will give the sufficient and necessary condition for the interior control points of the Bézier patch with minimal energy $E(\boldsymbol{r})$ in the following proposition.

Proposition 6.2. *Given the boundary control points $\boldsymbol{P}_{0,j}$,$\boldsymbol{P}_{1,j}$,$\boldsymbol{P}_{n-1,j}$ $\boldsymbol{P}_{n,j}$, $\boldsymbol{P}_{i,0}$, $\boldsymbol{P}_{i,1}$, $\boldsymbol{P}_{i,n-1}$ and $\boldsymbol{P}_{i,n}$ of a tensor product Bézier surface $\boldsymbol{r}(u,v)$, then $\boldsymbol{r}(u,v)$ has minimal energy $E(\boldsymbol{r})$ if and only if the remaining inner control points $\{\boldsymbol{P}_{i,j}\}_{i,j=2}^{n-2,n-2}$ satisfy*

$$\begin{aligned}\boldsymbol{0} \;=\;& \frac{\tau_1}{4(n-1)}\left(\sum_{k=0}^{n-1}\sum_{l=0}^{n}\frac{\binom{n}{l}}{\binom{2n}{l+j}}C_{n,i}^{k}\Delta^{1,0}\boldsymbol{P}_{kl}+\sum_{k=0}^{n}\sum_{l=0}^{n-1}\frac{\binom{n}{k}}{\binom{2n}{i+k}}C_{n,j}^{l}\Delta^{0,1}\boldsymbol{P}_{kl}\right)\\
&+\frac{2\tau_2}{(2n-1)^2}\sum_{k=0}^{n-1}\sum_{l=0}^{n-1}\frac{\binom{n-1}{k}\binom{n-1}{l}}{\binom{2n-2}{i+k-1}\binom{2n-2}{l+j-1}}B_{n,i}^{k}B_{n,j}^{l}\Delta^{1,1}\boldsymbol{P}_{kl}\\
&+\frac{\tau_2}{(2n-3)(2n+1)}\left(\sum_{k=0}^{n-2}\sum_{l=0}^{n}\frac{\binom{n-2}{k}\binom{n}{l}}{\binom{2n-4}{i+k-2}\binom{2n}{l+j}}A_{n,i}^{k}\Delta^{2,0}\boldsymbol{P}_{kl}+\sum_{k=0}^{n}\sum_{l=0}^{n-2}\frac{\binom{n}{k}\binom{n-2}{l}}{\binom{2n}{i+k}\binom{2n-4}{l+j-2}}A_{n,j}^{l}\Delta^{0,2}\boldsymbol{P}_{kl}\right)\end{aligned} \tag{30}$$

where

$$
\begin{aligned}
A_{n,i}^{k} &= \frac{nk(n-1)(k-1)+i^2(n-2)(n-3)-i(n-3)(2kn+n-2k-2)}{(2n-i-k-2)(2n-i-k-3)} \\
B_{n,i}^{k} &= \frac{[n(k-i)-k][n(k-i)-n+2i]-ni(2n-i-k-1)}{(2n-i-k)(2n-i-k-1)} \\
C_{n,i}^{k} &= \frac{ni-nk-i}{2n-i-k-1}\frac{\binom{n-1}{k}}{\binom{2n-2}{i+k-1}}
\end{aligned}
$$

A proof of Proposition 6.2 is given in Appendix II.

Since the energy functional $E(\mathbf{r})$ has a lower bound, the corresponding energy-minimizing Bézier surface always exists. From Proposition 6.2, for each unknown interior control point, an equation is obtained. Hence, we can get a linear system with $(n-3)\times(n-3)$ equations and $(n-3)\times(m-3)$ variables as follows,

$$\mathbf{MP}=\mathbf{B}$$

in which $\mathbf{M}$ is the coefficient matrix determined by Eq. (30), $\mathbf{P}=\{\boldsymbol{P}_{i,j}\}_{i,j=2}^{n-2,n-2}$ is the set of unknown interior control points, $\mathbf{B}$ is the right-hand side related to the specified control points on the first two layers of control mesh.

Solving this linear system, the unknown interior control points $\{\boldsymbol{P}_{i,j}\}_{i,j=2}^{n-2,n-2}$ can be represented as a linear combination of the specified control points on the first two layers of the control mesh. Furthermore, as each Bézier patch in the computational domain has the same coefficient matrix $\mathbf{M}$, we only compute the inverse of $\mathbf{M}$ once and reuse it for the remaining Bézier patches in the computational domain.

6.3. Local optimization approach for injective parameterization

A parameterization is valid if it has no self-intersection, meaning the mapping from the parametric domain to the computational domain is injective. After the local energy-minimizing patch construction described in Section 6.2, usually most of the Bézier patches are valid. However, for a few patches the resulting parameterization may still have self-intersections. In this section, we will firstly identify these invalid patches and subsequently reposition their control points to recover validity.

A parameterization is injective if its Jacobian is positive everywhere [15, 50]. For a planar Bézier surface $\boldsymbol{r}(u,v)=\sum_{i=0}^{n}\sum_{j=0}^{n}\boldsymbol{P}_{i,j}B_i^n(u)B_j^n(v)$, its Jacobian can be represented as a high-order

Bernstein polynomial [15, 50],

$$J(u,v) \quad = \quad \sum_{i=0}^{2n-1}\sum_{j=0}^{2n-1} \alpha_{ij} B_i^{2n-1}(u) B_j^{2n-1}(v). \tag{31}$$

Hence, the Jacobian $J(u,v)$ is bounded by

$$\min_{0\le i,j\le 2n-1} \alpha_{ij} \le J(u,v) \le \max_{0\le i,j\le 2n-1} \alpha_{ij} \tag{32}$$

Our invalid-patch-finding method can be described as follows: For each Bézier patch $\boldsymbol{r}(u,v)$ on the computational domain Ω, we compute its Jacobian coefficients α_{ij} according to Eq. (31); if $\min_{0\le i,j\le 2n-1} \alpha_{ij} > 0$, then $\boldsymbol{r}(u,v)$ is called *valid patch*; otherwise, $\boldsymbol{r}(u,v)$ is called *invalid patch*.

After finding all the invalid patches on the planar parameterization, we will repair the invalid patch locally by repositioning its internal control points such that all the Jacobian coefficients σ_{ij} are positive. Gravessen et al. [15] and Xu et al. [50] investigated this problem with a general non-linear constrained optimization framework,

$$\min \quad E(\boldsymbol{r}(u,v)) \quad \text{s.t.} \quad \alpha_{ij} > 0 \tag{33}$$

in which $E(\boldsymbol{r}(u,v))$ is defined as (29).

In contrast to the proposed optimization method in [15, 50], we solve this constrained optimization problem using the classical logarithmic-barrier method [11, 14, 42]:

$$\arg\min_{\boldsymbol{P}_{i,j}} \quad E(\boldsymbol{r}(u,v)) - \mu \sum_{i=0}^{2n-1}\sum_{j=0}^{2n-1} \ln(\alpha_{ij}) \tag{34}$$

where μ is a positive penalty parameter.

The results from our experiments indicate that the injective parameterization can be achieved by the log-barrier method, although without theoretical guarantee of injectivity. Since the corresponding non-linear optimization (34) is only performed on very few local invalid Bézier patches with limited number of design variables, we can obtain the final optimized parameterization efficiently.

Remark 6.1. For a general framework to obtain an injective parameterization, the optimization problem (33) can be generalized as follows :

$$\min \quad E(\boldsymbol{r}(u,v)) \quad \text{s.t.} \quad J(u,v) > 0$$

in which $J(u, v)$ is defined in (31).

Remark 6.2. Teichmüller mapping method proposed in [32], which has theoretical guarantees of injectivity, can be used to improve the proposed local optimization scheme.

Remark 6.3. Due to the C^1/G^1-constraints and the local optimization method in our framework, the minimum degree of the Bézier patches to guarantee a solution is 4. If the degree of the input boundary is smaller than 4, degree-elevation operation should be performed.

7. Examples and comparison

Starting from a set of B-spline curves as input, the proposed framework for analysis-suitable planar parameterization has been implemented as a plugin in the AXEL[1] platform. In this section, five parameterization examples are presented to show the effectiveness of the proposed method. The comparison with the skeleton-based parameterization method [54] is also performed for three parameterization examples.

7.1. Metrics for quality evaluation

In order to evaluate the quality of the planar parameterization results, we use the scaled Jacobians and the condition number of the Jacobian matrices as two important criteria. The *scaled Jacobian* at $\boldsymbol{r}(u, v)$ can be computed as follows,

$$J_s(u, v) = \frac{J(u, v)}{\|\boldsymbol{r}_u\|\|\boldsymbol{r}_v\|} \tag{35}$$

A parameterization is said to be *skew* if its Jacobian at some place is less or equal to zero. The minimal requirement for a planar parameterization to be suitable for isogeometric analysis is that all scaled Jacobians are as close as possible to 1.0.

The condition number $\kappa(\mathbf{J})$ of a Jacobian matrix $\mathbf{J}$ is another important evaluation criteria, given by

$$\kappa(\mathbf{J}) = |\mathbf{J}||\mathbf{J}^{-1}| \tag{36}$$

in which

$$\mathbf{J} = \begin{pmatrix} x_u & y_u \\ x_v & y_v \end{pmatrix}, \tag{37}$$

[1] http://axel.inria.fr/

The Frobenius norm of $\mathbf{J}$ is defined as $|\mathbf{J}| = (tr(\mathbf{J}^T\mathbf{J}))^{1/2}$. It can be easily proven that the minimum of $\kappa(\mathbf{J})$ is 2.0. For the quality evaluation of planar parameterization, the condition number $\kappa(\mathbf{J})$ is smaller, the optimization is better.

The parameterization quality with respect to the scaled Jacobians and condition number for the examples presented in this paper are summarized in Table. 1 and Table. 2.

7.2. Effect of quad meshing

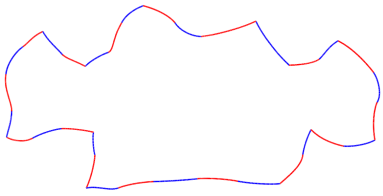
In the proposed framework, the quad-meshing step has a great effect on the final parameterization results. Furthermore, the quad-meshing results depend on the approximate convex decomposition of the input computational domain. For the approximate convex decomposition method in [24], if different non-concavity tolerances ϵ are specified, different number of approximate convex sub-domain can be obtained. Hence, from the same input boundary Fig. 9 (a), different quad-meshing results can be generated from different selections of non-concavity tolerance ϵ as shown in Fig. 9(b) with $\epsilon = 0.6$ and Fig. 9(e) with $\epsilon = 0.4$. For a smaller non-concavity tolerances ϵ, we can obtain a quad-meshing result with more quad elements and more irregular vertices. Moreover, the corresponding scaled Jacobian colormap of the parameterization in Fig. 9 (c) and Fig. 9 (g) are depicted in Fig. 9 (d) and Fig. 9 (h).

The number of Bézier segments for the same input boundary also influences the final parameterization results. For the same input boundary with different number of Bézier segments, different quad-meshing result will be obtained leading to different parameterizations. For instance, the same boundary with different number of Bézier segments is shown in Fig. 9 (i). The corresponding quad-meshing result is presented in Fig. 9 (j), and the final parameterization and the corresponding scaled Jacobian colormap are illustrated in Fig. 9 (k) and Fig. 9 (l) respectively. Quantitative data of three parameterization results presented in Fig. 9 (c), Fig. 9 (g) and Fig. 9 (k) are listed in Table 1.

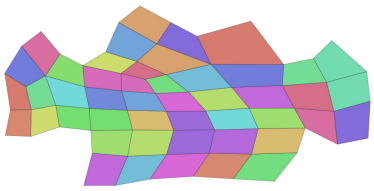
7.3. Comparison with skeleton-based parameterization method

In order to show the effectiveness of the proposed approach, three examples are presented to compare the proposed method with the skeleton-based domain decomposition method [54].

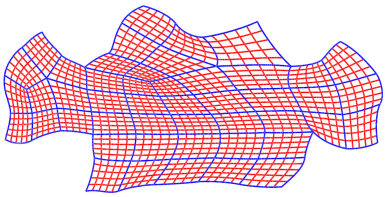
Fig. 10 (a) depicts the boundary Bézier curves after the pre-processing of input boundary B-spline curves; Fig. 10 (b) presents the discrete boundary obtained by connecting the ending



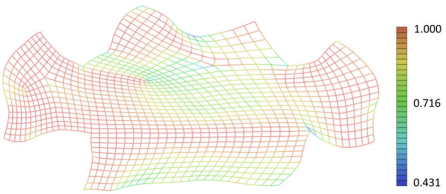
(a) boundary Bézier curves



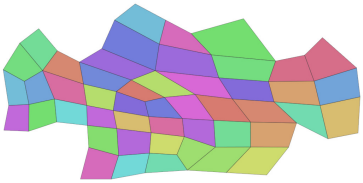
(b) quad meshing result I with $\epsilon = 0.6$



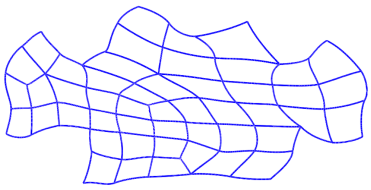
(c) parameterization result I



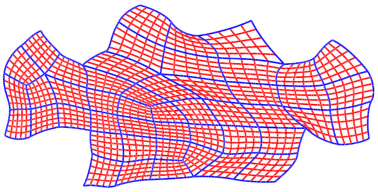
(d) Jacobian colormap I



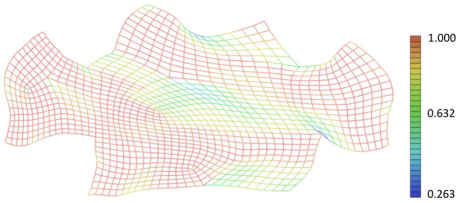
(e) quad meshing result II with $\epsilon = 0.4$



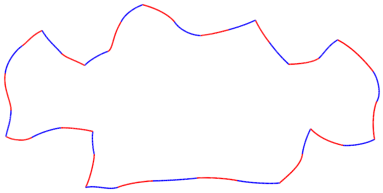
(f) segmentation curves II



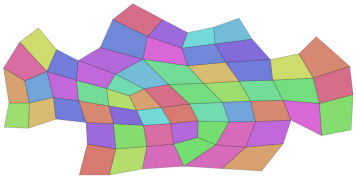
(g) parameterization result II



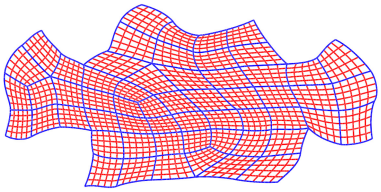
(h) Jacobian colormap



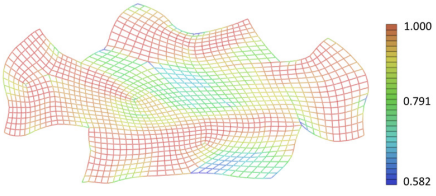
(i) boundary with more Bézier curves



(j) quad meshing result III

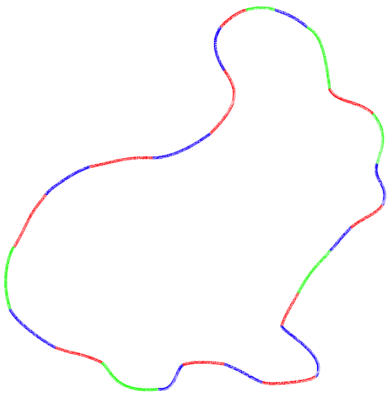


(k) parameterization result III

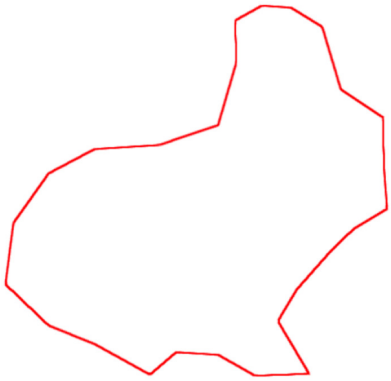


(l) Jacobian colormap III

Fig. 9. Example II to show the effect of quad meshing.

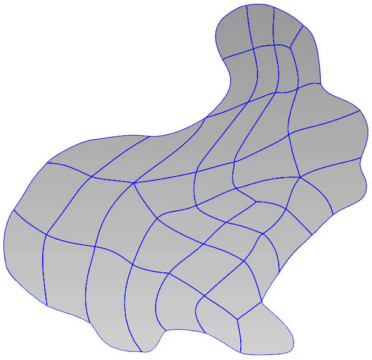


(a) boundary Bézier curves

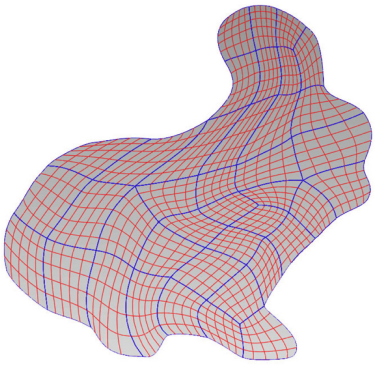


(b) discrete boundary

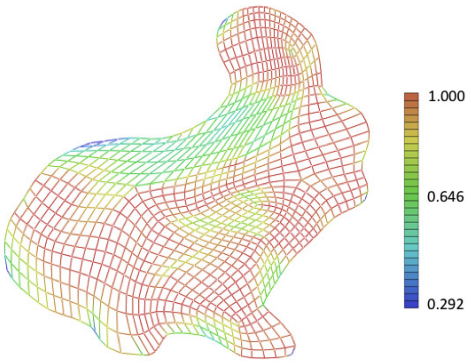
(c) quad meshing result



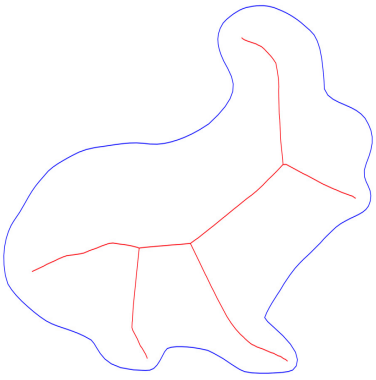
(d) segmentation curves



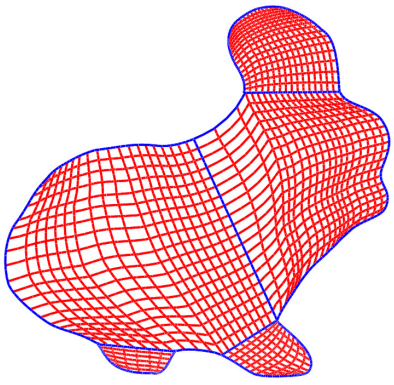
(e) parameterization result



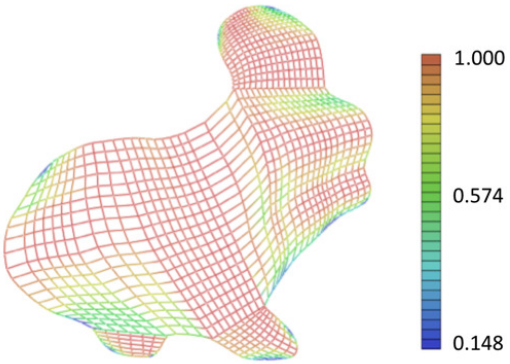
(f) Jacobian colormap



(g) extracted skeleton [54]

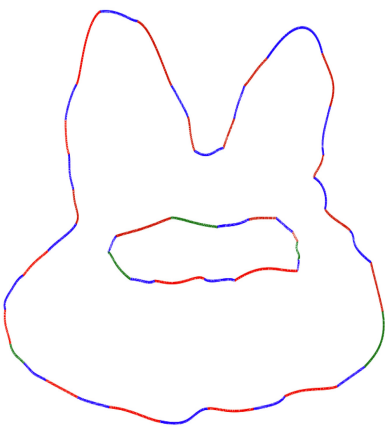


(h) skeleton-based parameterization [54]

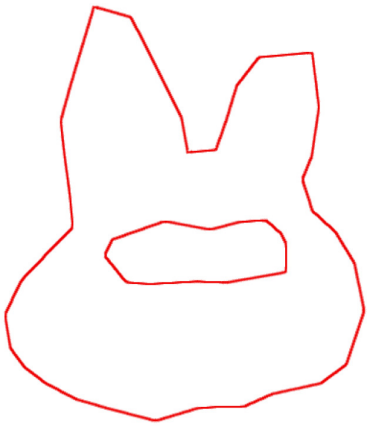


(i) Jacobian colormap of (h)

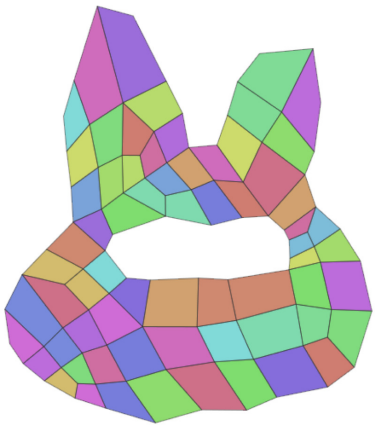
Fig. 10. Example III.



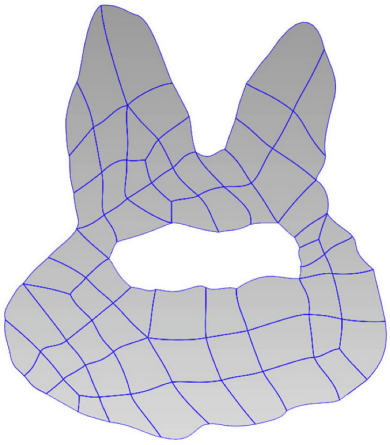
(a) boundary Bézier curves



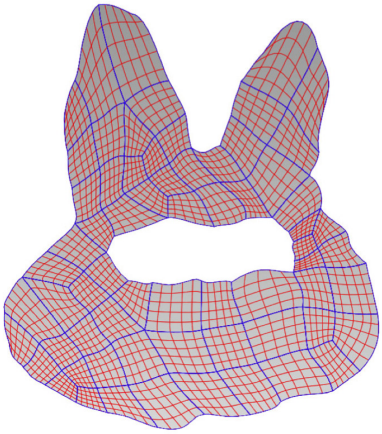
(b) discrete boundary



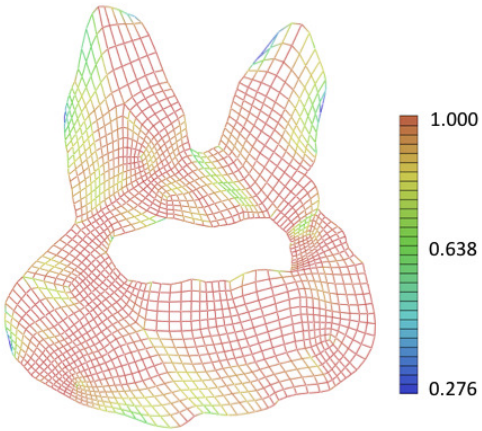
(c) quad meshing result



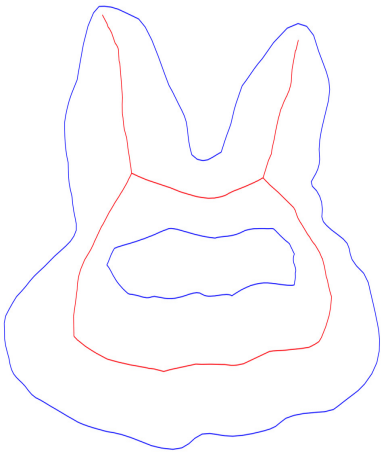
(d) segmentation curves



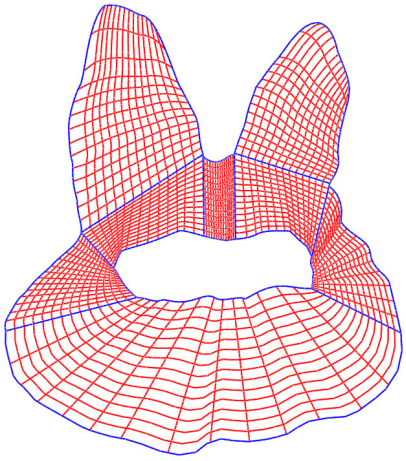
(e) parameterization result



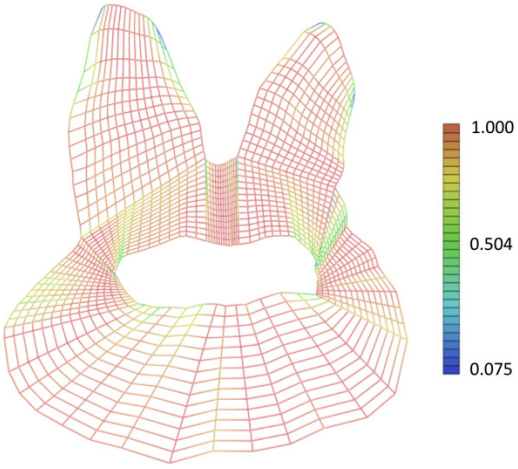
(f) Jacobian colormap



(g) extracted skeleton [54]

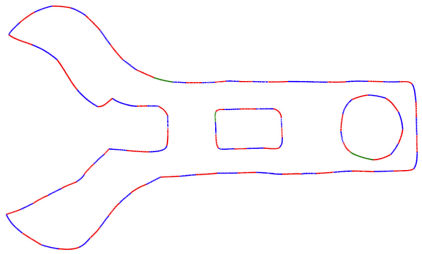


(h) skeleton-based parameterization [54]

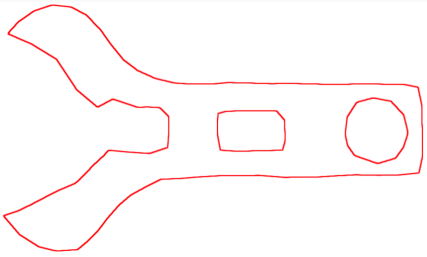


(i) Jacobian colormap of (h)

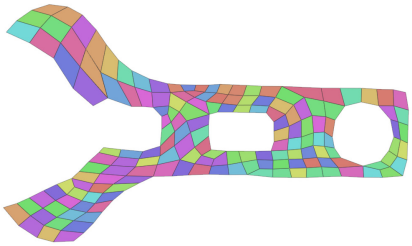
Fig. 11. Example IV.



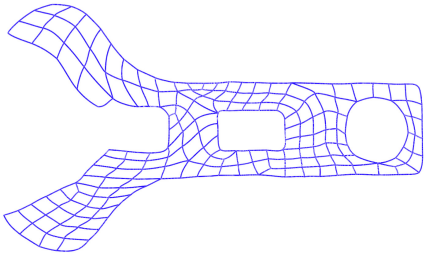
(a) boundary Bézier curves



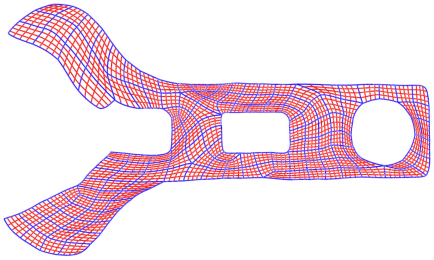
(b) discrete boundary



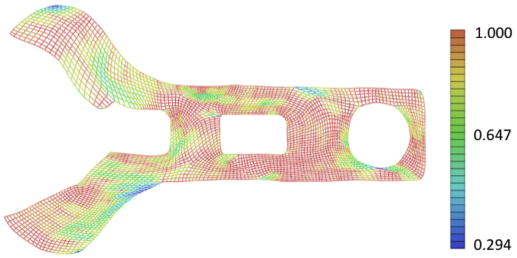
(c) quad meshing result



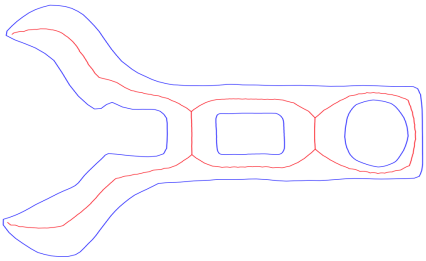
(d) segmentation curves



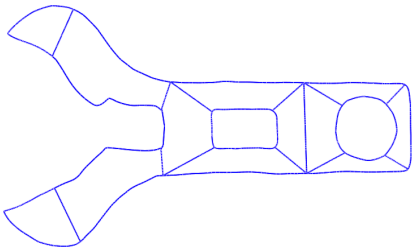
(e) parameterization result



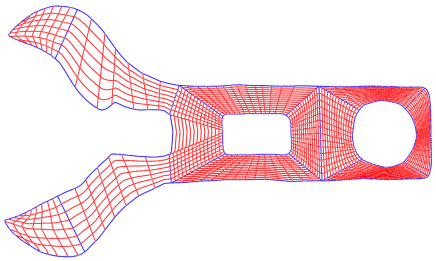
(f) Jacobian colormap



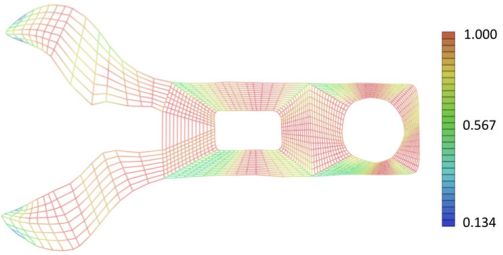
(g) extracted skeleton [54]



(h) skeleton-based domain partition[54]



(i) skeleton-based parameterization [54]



(j) Jacobian colormap of (i)

Fig. 12. Example V.

Table 1: Quantitative data for planar parameterization in Fig. 9 (c), Fig. 9 (g) and Fig. 9 (k). p: degree of planar parameterization; # Con.: number of control points; # Patch.: number of patches.

Example	p	#Con.	#Patch	Scaled Jacobian			Condition number		
				Max	Average	Min	Max	Average	Min
Fig. 9 (c)	4	817	47	1.000	0.9054	0.431	4.72	2.41	2.00
Fig. 9 (g)	4	841	49	1.000	0.9024	0.263	7.62	2.53	2.00
Fig. 9 (k)	4	923	57	1.000	0.9279	0.582	3.86	2.36	2.00

control points of each Bézier curve, and the corresponding quad meshing result can be found in Fig. 10 (c). Fig. 10 (d) shows the domain partition results with the construction of segmentation curves by the global optimization method. After the local optimization for each sub-patch, the final planar parameterization is illustrated in Fig. 10 (e). The iso-parametric curves show the quality of the planar parameterization. The corresponding scaled Jacobian colormap in Fig. 10 (f) proves that the parameterization is analysis suitable. Fig. 10 (g) presents the extracted skeleton from the input boundary, Fig. 10 (h) the skeleton-based parameterization results with blue segmentation curves, and Fig. 10 (i) the corresponding scaled Jacobian colormap. More comparison examples with complex geometry can be found in Fig. 11 and Fig. 12. Since global/local optimizations with C^1/G^1 constraints are applied for the construction of segmentation curves and parameterization of Bézier subdomains, our method achieves smooth parameterizations compared to the skeleton-based approach [54].

Quantitative data of the comparison examples including the parameterization quality metrics with respect to scaled Jacobians and condition numbers presented in Fig. 10, Fig. 11 and Fig. 12 are summarized in Table 2. We can see from Table 2 that the proposed approach achieves analysis-suitable parameterizations with bigger average scaled Jacobians and smaller average condition numbers compared to the skeleton-based method [54]. In the comparison examples, the input boundary curves are usually B-spline curves, hence after the domain partition by skeleton-based method [54], knot-insertion operations should be performed on each sub-domain to obtain extracted Bézier patches. Although the number of subdomains in the skeleton-based method [54] is much smaller than that in our proposed method, see Table 2, the number of Bézier patches and the number of control points in the skeleton-based approach [54] are similar to those in the proposed

Table 2: Quantitative data for planar parameterization in Fig. 10, Fig. 11 and Fig. 12. p: degree of planar parameterization; # SD: number of subdomains by domain decomposition;# Patch: number of Bézier patches; # Con.: number of control points.

Example	Method	p	#SD	#Patch	#Con.	Scaled Jacobian		Condition number	
						Average	Min	Average	Max
Fig. 10	Our method	6	39	39	1467	0.8843	0.292	2.76	8.06
	Xu et al.[54]	6	5	35	1309	0.5172	0.148	5.36	16.31
Fig. 11	Our method	5	66	66	1768	0.9194	0.276	2.42	10.18
	Xu et al.[54]	5	8	56	1507	0.7801	0.075	4.35	18.23
Fig. 12	Our method	5	155	155	3720	0.9017	0.294	2.57	7.86
	Xu et al.[54]	5	12	132	3282	0.7894	0.134	4.23	15.64

method. These extracted Bézier patches are used as the computational elements in IGA [3].

There are several parameters involved in the objective functions of the optimization framework. Different parameterization results can be obtained from different choices of these parameters. The parameters of the objective functions F_{shape} in (14), F in (16) and $E(\boldsymbol{r})$ in (29) are summarized in Table 3 for the presented examples in Fig. 1, Fig. 9, Fig. 10, Fig. 11 and Fig. 12. Furthermore, the associated CPU times for the global optimization and local optimization process are listed in Table 3. The computational cost of the proposed method depends on the number of patches and the number of control points in the planar parameterization.

In conclusion, the planar parameterization obtained by the proposed global/local optimization method has high-quality, and is suitable for isogeometric applications.

8. Conclusion and future work

The parameterization of the computational domain with complex CAD boundary is a key step in IGA. In this paper, a general framework for constructing IGA-suitable planar B-spline parameterizations of the computational domain with high genus and more complex boundary curves is proposed. High-quality patch-partition results with few singularities are achieved by a global optimization method, and the Bézier patch with respect to each quad in the quadrangulation is obtained by a local optimization method yielding uniform and orthogonal iso-parametric structures while keeping the C^1/G^1 continuity conditions between patches. The proposed framework can be

Table 3: Parameters of the objective functions and the computing times (in seconds) for the examples in Fig. 1, Fig. 9, Fig. 10, Fig. 11 and Fig. 12. # T_1: computing time for the global optimization ; # T_2: computing time for the local optimization; # T: total computing time.

Example	F_{shape} in (14)		F in (16)			$E(\boldsymbol{r})$ in (29)		$\#T_1$	$\#T_2$	$\#T$
	σ_1	σ_2	ω_1	ω_2	ω_3	τ_1	τ_2			
Fig. 1	2.0	1.0	2.0	1.0	50.0	2.0	1.5	73.22	177.86	251.08
Fig. 9(c)	1.0	1.0	1.0	1.0	50.0	1.0	1.5	22.68	26.64	49.32
Fig. 9(g)	1.0	1.0	1.0	1.0	50.0	1.0	1.5	22.96	27.18	50.14
Fig. 9(k)	1.0	1.0	1.0	1.0	50.0	1.0	1.5	27.68	36.32	63.90
Fig. 10	1.0	2.0	1.0	2.0	50.0	1.0	2.0	32.74	53.84	86.58
Fig. 11	2.0	1.0	1.0	2.0	50.0	2.0	1.0	41.02	61.36	102.38
Fig. 12	2.0	1.0	2.0	2.0	50.0	2.0	1.0	75.70	209.68	285.38

considered as a generalized Bézier extraction of a planar domain, and the resulting Bézier patches can be used as the computational elements for IGA [3]. The efficiency and robustness of the proposed approach are demonstrated by several examples.

In the future, we will focus on the following extensions:

1. Quad mesh generation significantly influences the parameterization quality. State-of-the-art quad meshing techniques will be investigated for IGA-suitable parameterization problem.
2. Several parameters are involved in the proposed optimization framework. Automatic selection of these parameters from the input data will be addressed.
3. The proposed local optimization scheme can be improved by using Teichmüller mapping [32] with theoretical guarantees of injectivity, and it can be accelerated by parallelized implementation exploiting GPU/OpenMP.
4. The proposed method can be extended to NURBS planar parameterization directly. Extension to three-dimensional volumetric parameterization problem with complex boundary representation will be a part of future work.

Acknowledgment

This research was supported by Zhejiang Provincial Natural Science Foundation of China under Grant Nos. LR16F020003, LQ16F020005, the National Nature Science Foundation of China under

Grant Nos. 61472111, 61602138, 61502130, and the Open Project Program of the State Key Lab of CAD&CG (A1703), Zhejiang University.

Stéphane Bordas also thanks partial funding for his time provided by the European Research Council Starting Independent Research Grant (ERC Stg grant agreement No. 279578) " RealTCut Towards real time multi-scale simulation of cutting in non-linear materials with applications to surgical simulation and computer guided surgery ". Stéphane Bordas is also grateful for the support of the Fonds National de la Recherche Luxembourg FWO-FNR grant INTER/FWO/15/10318764.

Appendix I. Conversion of multiply-connected region

In this part, an approach for converting multiply-connected region into simply-connected region will be described.

If the region bounded by the given boundary curves is not a simply-connected region, that is, it is a multiply-connected region, we have to convert it into a simply-connected region. Suppose that the given multiply-connected region has one outer boundary $\boldsymbol{M}_1 = (\boldsymbol{V}_1, \boldsymbol{E}_1)$ and one interior boundary $\boldsymbol{M}_2 = (\boldsymbol{V}_2, \boldsymbol{E}_2)$, then we find the vertex $\boldsymbol{v}_i \in \boldsymbol{V}_1 = \{\boldsymbol{v}_0, \boldsymbol{v}_1, \cdots, \boldsymbol{v}_{n_1-1}\}$, $\boldsymbol{u}_j \in \boldsymbol{U}_2 = \{\boldsymbol{u}_0, \boldsymbol{u}_1, \cdots, \boldsymbol{u}_{n_2-1}\}$ such that the distance ℓ_0 between $\boldsymbol{u}_j$ and $\boldsymbol{v}_i$ is minimal among all the distance between two points from the interior boundary to outer boundary. That is,

$$\ell_0 = \min\{\ell(\boldsymbol{v}_p, \boldsymbol{u}_q)\}, \boldsymbol{v}_p \in \boldsymbol{V}_1, \boldsymbol{u}_q \in \boldsymbol{V}_2. \tag{38}$$

in which $\ell(\boldsymbol{v}_p, \boldsymbol{u}_q)$ is the distance between the vertex $\boldsymbol{v}_p$ and $\boldsymbol{u}_q$. In order to generate a uniform quadrilateral decomposition, some extra points should be added between the vertex $\boldsymbol{u}_j$ and $\boldsymbol{v}_i$. The number ν of added vertex is computed as follows:

$$\nu = \lceil \frac{(n_1 + n_2) \times \ell_0}{\sum_{p=0}^{n_1-2} \ell(\boldsymbol{v}_p, \boldsymbol{v}_{p+1}) + \sum_{q=0}^{n_2-2} \ell(\boldsymbol{u}_q, \boldsymbol{u}_{q+1})} \rceil \tag{39}$$

then the added ν vertices $\boldsymbol{z}_k$ between $\boldsymbol{v}_i$ and $\boldsymbol{u}_j$ are obtained by linear interpolation. The final boundary vertex sequence $\boldsymbol{V}$ of the simply-connected region is

$$\boldsymbol{V} = \{\boldsymbol{v}_0, \boldsymbol{v}_1, \cdots, \boldsymbol{v}_i, \boldsymbol{z}_0, \cdots, \boldsymbol{z}_{\nu-1}, \boldsymbol{u}_i, \cdots, \boldsymbol{u}_j, \boldsymbol{z}_{\nu-1}, \cdots, \boldsymbol{z}_0, \boldsymbol{v}_i, \cdots, \boldsymbol{v}_{n_1-1}\}. \tag{40}$$

For the domain with multiple components, we can apply above method iteratively, that is, in every iteration, we insert edges between the vertices that have the shortest distance between the outer and interior boundaries.

Appendix II. Proof of Proposition 6.2

Proof. Set $\boldsymbol{P}_{ij} = (x_{ij}^1, x_{ij}^2)$. The gradient of the energy functional (29) with respect to the coordinates x_{ij}^{ω} of an unknown control points, $\omega \in \{1, 2\}$, can be computed as follows,

$$\begin{aligned}\frac{\partial E(\boldsymbol{r})}{\partial x_{ij}^{\omega}} &= \frac{1}{2}\int_U \tau_1(< \boldsymbol{r}_u, \frac{\partial \boldsymbol{r}_u}{\partial x_{ij}^{\omega}} > + < \boldsymbol{r}_v, \frac{\partial \boldsymbol{r}_v}{\partial x_{ij}^{\omega}} >) + \tau_2 < \frac{\partial \boldsymbol{r}_{uu}}{\partial x_{ij}^{\omega}}, \boldsymbol{r}_{uu} > \\ &\quad +2\tau_2 < \frac{\partial \boldsymbol{r}_{uv}}{\partial x_{ij}^{\omega}}, \boldsymbol{r}_{uv} > +\tau_2 < \frac{\partial \boldsymbol{r}_{vv}}{\partial x_{ij}^{\omega}}, \boldsymbol{r}_{vv} > dudv\end{aligned}$$

From the derivatives of the Bézier surface over rectangular domain, we have

$$\boldsymbol{r}_u(u,v) = n\sum_{k=0}^{n-1}\sum_{l=0}^{n} B_k^{n-1}(u)B_l^n(v)\Delta^{1,0}\boldsymbol{P}_{kl}, \tag{41}$$

$$\boldsymbol{r}_v(u,v) = n\sum_{k=0}^{n}\sum_{l=0}^{n-1} B_k^{n}(u)B_l^{n-1}(v)\Delta^{0,1}\boldsymbol{P}_{kl} \tag{42}$$

$$\boldsymbol{r}_{uu}(u,v) = n(n-1)\sum_{k=0}^{n-2}\sum_{l=0}^{n} B_k^{n-2}(u)B_l^n(v)\Delta^{2,0}\boldsymbol{P}_{kl}, \tag{43}$$

$$\boldsymbol{r}_{uv}(u,v) = n^2\sum_{k=0}^{n-1}\sum_{l=0}^{n-1} B_k^{n-1}(u)B_l^{n-1}(v)\Delta^{1,1}\boldsymbol{P}_{kl}, \tag{44}$$

$$\boldsymbol{r}_{vv}(u,v) = n(n-1)\sum_{k=0}^{n}\sum_{l=0}^{n-2} B_k^{n}(u)B_l^{n-2}(v)\Delta^{0,2}\boldsymbol{P}_{kl}, \tag{45}$$

Hence,

$$\frac{\partial \boldsymbol{r}_u}{\partial x_{ij}^{\omega}} = n\left(B_{i-1}^{n-1}(u) - B_i^{n-1}(u)\right)\boldsymbol{e}^{\omega}, \quad \frac{\partial \boldsymbol{r}_v}{\partial x_{ij}^{\omega}} = n\left(B_{j-1}^{n-1}(v) - B_j^{n-1}(v)\right)\boldsymbol{e}^{\omega}, \tag{46}$$

$$\frac{\partial \boldsymbol{r}_{uu}}{\partial x_{ij}^{\omega}} = n(n-1)\left(B_{i-2}^{n-2}(u) - 2B_{i-1}^{n-2}(u) + B_i^{n-2}(u)\right)B_j^m(v)\boldsymbol{e}^{\omega}, \tag{47}$$

$$\frac{\partial \boldsymbol{r}_{uv}}{\partial x_{ij}^{\omega}} = n^2\left(B_{i-1}^{n-1}(u) - B_i^{n-1}(u)\right)\left(B_{j-1}^{m-1}(v) - B_j^{m-1}(v)\right)\boldsymbol{e}^{\omega}, \tag{48}$$

$$\frac{\partial \boldsymbol{r}_{vv}}{\partial x_{ij}^{\omega}} = n(n-1)B_i^n(u)\left(B_{j-2}^{n-2}(v) - 2B_{j-1}^{n-2}(v) + B_j^{n-2}(v)\right)\boldsymbol{e}^{\omega}, \tag{49}$$

where $\boldsymbol{e}^1 = (1,0), \boldsymbol{e}^2 = (0,1)$.

From (41), (42), (43), (44), (45),(46), (47), (48) and (49), it follows that

$$\begin{aligned}\frac{\partial E(\boldsymbol{r})}{\partial x_{ij}^{\omega}} &= \frac{1}{2}\tau_1 \int_U n\left(B_{i-1}^{n-1}(u) - B_i^{n-1}(u)\right) < \boldsymbol{e}^{\omega}, \boldsymbol{r}_u > dudv \\ &+\frac{1}{2}\tau_1 \int_U n\left(B_{j-1}^{n-1}(v) - B_j^{n-1}(v)\right) < \boldsymbol{r}_v, \boldsymbol{e}^{\omega} > dudv \\ &+\frac{1}{2}\tau_2 \int_U n(n-1)\left(B_{i-2}^{n-2}(u) - 2B_{i-1}^{n-2}(u) + B_i^{n-2}(u)\right) B_j^m(v) < \boldsymbol{e}^{\omega}, \boldsymbol{r}_{uu} > dudv \\ &+\tau_2 \int_U n^2\left(B_{i-1}^{n-1}(u) - B_i^{n-1}(u)\right)\left(B_{j-1}^{m-1}(v) - B_j^{m-1}(v)\right) < \boldsymbol{r}_{uv}, \boldsymbol{e}^{\omega} > dudv \\ &+\frac{1}{2}\tau_2 \int_U n(n-1)B_i^n(u)\left(B_{j-2}^{n-2}(v) - 2B_{j-1}^{n-2}(v) + B_j^{n-2}(v)\right) < \boldsymbol{r}_{vv}, \boldsymbol{e}^{\omega} > dudv\end{aligned}$$

The energy functional (29) has an extreme if and only if $\frac{\partial E(\boldsymbol{r})}{\partial x_{ij}^{\omega}} = 0$, $\omega \in \{1, 2, 3\}$. After some simple computation by using the product formula of two Bernstein polynomials in Proposition. 3.4 and the integration formula of Bernstein polynomials in Eq. (2), Eq. (30) can be obtained. Thus the proof is completed. □

References

[1] M. Aigner, C. Heinrich, B. Jüttler, E. Pilgerstorfer, B. Simeon and A.-V. Vuong. Swept volume parametrization for isogeometric analysis. In E. Hancock and R. Martin (eds.), The Mathematics of Surfaces (MoS XIII 2009), LNCS 5654(2009) 19-44.

[2] R. Ansari, A. Norouzzadeh. Nonlocal and surface effects on the buckling behavior of functionally graded nanoplates: An isogeometric analysis. Physica E: Low-dimensional Systems and Nanostructures, 84(2016) 84-97.

[3] M. J. Borden, M. A. Scott, J. A. Evans, T. J. R. Hughes. Isogeometric finite element data structures based on Bézier extraction of NURBS. International Journal for Numerical Methods in Engineering 87(1-5)(2011) 15-47.

[4] F. Buchegger, B. Jüttler, A. Mantzaflaris. Adaptively refined multi-patch B-splines with enhanced smoothness. Applied Mathematics and Computation 272(2016) 159-172.

[5] F. Buchegger, B. Jüttler. Planar multi-patch domain parameterizaton via patch adjacency graphs. Computer-Aided Design 82(2017) 2-12.

[6] do Carmo, Manfredo P.. Differential geometry of curves and surfaces. Prentice Hall, 1976.

[7] C. Chan, C. Anitescu, T. Rabczuk. Volumetric parametrization from a level set boundary representation with PHT-splines. Computer-Aided Design 82 (2017) 29-41.

[8] E. Cohen, T. Martin, R.M. Kirby, T. Lyche and R.F. Riesenfeld, Analysis-aware modeling: understanding quality considerations in modeling for isogeometric analysis. Computer Methods in Applied Mechanics and Engineering 199(2010) 334-356.

[9] J.A. Cottrell, T.J.R. Hughes, Y. Bazilevs. Isogeometric Analysis: Toward Integration of CAD and FEA. Wiley, 2009.

[10] J.A. Cottrell, T.J.R. Hughes, A. Reali. Studies of refinement and continuity in isogeometric structural analysis. Computer Methods in Applied Mechanics and Engineering, 196(41)(2007) 4160-4183.

[11] D. Den Hertog, C. Roos, T. Terlaky. On the classical logarithmic barrier function method for a class of smooth convex programming problems. Journal of Optimization Theory and Applications 73(1)(1992) 1-25.

[12] J.M. Escobara, J.M. Cascónb, E. Rodrígueza, R. Montenegro. A new approach to solid modeling with trivariate T-spline based on mesh optimization. Computer Methods in Applied Mechanics and Engineering, 200(2011) 3210-3222.

[13] A. Falini, J. Speh, B. Jüttler. Planar domain parameterization with THB-splines. Computer Aided Geometric Design 35-36(2015) 95-108.

[14] A. V. Fiacco, G. P. McCormick. Nonlinear Programming: Sequential Unconstrained MinimizationTechniques. J. Wiley and Sons, New York, 1968. Reprinted as Classics in Applied Mathematics 4, SIAM, 1990. MR 39:5152.

[15] J. Gravesen, A. Evgrafov, D.M. Nguyen, P. Nørtoft. Planar parametrization in isogeometric analysis. In: Mathematical methods for curves and surfaces. Lecture Notes in Computer Science 8177(2014) 189-212.

[16] D. Groisser, J. Peters. Matched G^k-constructions always yield C^k-continuous isogeometric elements. Computer Aided Geometric Design 34(2015) 67-72.

[17] G. Farin. Curves and Surfaces for CAGD - A Practical Guide. 5th Edition, Morgan Kaufmann. Publishers, 2002.

[18] T. Hermann, J. Peters, T. Strotman, A geometric constraint on curve networks suitable for smooth interpolation, Computer-Aided Design, 43(2011) 741-746

[19] T.J.R. Hughes, J.A. Cottrell, Y. Bazilevs. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry, and mesh refinement. Computer Methods in Applied Mechanics and Engineering 194(2005) 4135-4195.

[20] M. Kapl, V. Vitrih, B. Jüttler, K. Birner. Isogeometric analysis with geometrically continuous functions on two-patch geometries. Computers & Mathematics with Applications 70(2015) 1518-1538.

[21] M. Kapl, F. Buchegger, M. Bercovier, B. Jüttler. Isogeometric analysis with geometrically continuous functions on planar multi-patch geometries. Computer Methods in Applied Mechanics and Engineering, 316(2017) 209-234

[22] K. Karčiauskas, J. Peters. Refinable G^1 functions on G^1 free-form surfaces. Computer Aided Geometric Design, 54(2017) 61-73.

[23] K. Karčiauskas, T. Nguyen, and J. Peters. Generalizing bicubic splines for modelling and IGA with irregular layout. Computer Aided Design, 70(2016) 23-35.

[24] J.M. Lien, N. M. Amato. Approximate convex decomposition of polygons. Computational Geometry 35(1-2)(2016) 100-123.

[25] L. Liu, Y. Zhang, T. J.R. Hughes, M. A. Scott, T. W. Sederberg. Volumetric T-Spline construction Using Boolean operations. Engineering with Computers, 30(4)(2014) 425-439

[26] X.W. Ma, G.Q Zhao, L. Sun. AUTOMESH2D/3D: robust automatic mesh generator for metal forming simulation. Materials Research Innovations 15(1)(2011) 482-486.

[27] T. Martin, E. Cohen, R.M. Kirby. Volumetric parameterization and trivariate B-spline fitting using harmonic functions. Computer Aided Geometric Design 26(2009) 648-664.

[28] B. Mourrain, R. Vidunas, N. Villamizar. Dimension and bases for geometrically continuous splines on surfaces of arbitrary topology. Computer Aided Geometric Design 45(2016) 108-133.

[29] D. Nairn, J. Peters, D. Lutterkort. Sharp, quantitative bounds on the distance between a polynomial piece and its Bézier control polygon. Computer Aided Geometric Design 16(7) (1999) 613-631.

[30] T. Nguyen, B. Jüttler. Parameterization of contractible domains using sequences of harmonic maps. Lecture Notes in Computer Science 6920(2012) 501-514.

[31] V.P Nguyen, P. Kerfriden, S.P.A. Bordas, T. Rabczuk. Isogeometric analysis suitable trivariate NURBS representation of composite panels with a new offset algorithm. Computer-Aided Design 55(2014) 49-63.

[32] X. Nian, F. Chen. Planar domain parameterization for isogeometric analysis based on Teichmüller mapping. Computer Methods in Applied Mechanics and Engineering 311 (2016) 41-55.

[33] J. Nocedal, S. J. Wright. Numerical Optimization, 2nd Edition, Springer Series in Operations Research and Financial Engineering, Springer, New York, 2006.

[34] N. Nguyen-Thanh, K. Zhou, X. Zhuang, P. Areias, H. Nguyen-Xuan, Y. Bazilevs , T. Rabczuk. Isogeometric analysis of large-deformation thin shells using RHT-splines for multiple-patch coupling. Computer Methods in Applied Mechanics and Engineering, 316(2017) 1157-1178.

[35] N. Nguyen-Thanh, J. Muthu, X. Zhuang, T. Rabczuk. An adaptive threedimensional RHT-spline formulation in linear elasto-statics and elasto-dynamics. Computational Mechanics, 53(2014) 369-385.

[36] N. Nguyen-Thanh, J. Kiendl, H. Nguyen-Xuan, R. Ẃuchner, K.U. Bletzinger, Y. Bazilevs, T. Rabczuk. Rotation free isogeometric thin shell analysis using PHT-splines. Computer Methods in Applied Mechanics and Engineering, 200(2011) 3410-3424.

[37] N. Nguyen-Thanh, H. Nguyen-Xuan, S. Bordas, T. Rabczuk. Isogeometric analysis using polynomial splines over hierarchical T-meshes for two-dimensional elastic solids. Computer Methods in Applied Mechanics and Engineering, 200(2011) 1892-1908.

[38] E. Pilgerstorfer, B. Jüttler. Bounding the influence of domain parameterization and knot spacing on numerical stability in Isogeometric Analysis. Computer Methods in Applied Mechanics and Engineering 268(2014) 589-613.

[39] Q. Pan, C. Chen, G. Xu. Isogeometric finite element approximation of minimal surfaces based on extended loop subdivision. Journal of Computational Physics, 343(2017) 1-16.

[40] Q. Pan, G. Xu, G. Xu, Y. Zhang. Isogeometric analysis based on extended Loop's subdivision. Journal of Computational Physics, 299(2015) 731-746.

[41] K.F. Pettersen, V. Skytt. Spline volume fairing. Lecture Notes in Computer Science 6920(2012) 553-561.

[42] S.P. Sastry, S.M. Shontz, and S.A. Vavasis. A log-barrier method for mesh quality improvement and untangling. Engineering with Computers 30 (2014) 315-329.

[43] H. Speleers, C. Manni. Optimizing domain parameterization in isogeometric analysis based on Powell-Sabin splines. Journal of Computational and Applied Mathematics 289(2015) 68-86.

[44] K. Takayama, D. Panozzo, O. Sorkine-Hornung Pattern-based quadrangulation for N-sided patches. Computer Graphics Forum, 33(5) (2014) 177-184.

[45] G. Wang. Subdivision method for finding the intersection between two Bézier curves or surfaces. Journal of Zhejiang University, Special Issue on Computational Geometry (1984) 108-119.

[46] W. Wang, Y. Zhang, L. Liu, T. J.R. Hughes. Trivariate solid T-spline construction from boundary triangulations with arbitrary genus topology. Computer-Aided Design 45(2013) 351-360.

[47] X. Wang, X. Qian. An optimization approach for constructing trivariate B-spline solids. Computer-Aided Design 46 (2014) 179 - 191.

[48] G. Xu, T. H. Kwok, and C. C. Wang. Isogeometric computation reuse method for complex objects with topology-consistent volumetric parameterization. Computer-Aided Design 91(2017) 1-13.

[49] G. Xu, B. Mourrain, R. Duvigneau, A. Galligo. Optimal analysis-aware parameterization of computational domain in 3D isogeometric analysis. Computer-Aided Design 45(2013) 812-821.

[50] G. Xu, B. Mourrain, R. Duvigneau, A. Galligo. Parameterization of computational domain in isogeometric analysis: methods and comparison. Computer Methods in Applied Mechanics and Engineering 200(2011) 2021-2031.

[51] G. Xu, B. Mourrain, R. Duvigneau, A. Galligo. Analysis-suitable volume parameterization of multi-block computational domain in isogeometric applications. Computer-Aided Design 45(2013) 395-404.

[52] G. Xu, B. Mourrain, R. Duvigneau, A. Galligo. Constructing analysis-suitable parameterization of computational domain from CAD boundary by variational harmonic method. Journal of Computational Physics 252(2013) 275-289.

[53] G. Xu, B. Mourrain, A. Galligo, T. Rabczuk. High-quality construction of analysis-suitable trivariate NURBS solids by reparameterization methods. Computational Mechanics 54 (2014) 1303-1313

[54] J. Xu, F. Chen, J. Deng. Two-dimensional domain decomposition based on skeleton computation for parameterization and isogeometric analysis. Computer Methods in Applied Mechanics and Engineering 284(2015) 541-555

[55] Y. Zhang, W. Wang, T. J.R. Hughes. Solid T-spline construction from boundary representations for genus-zero geometry. Computer Methods in Applied Mechanics and Engineering, 201(2012) 185-197.

[56] Y. Zhang, W. Wang, T. J.R. Hughes. Conformal solid T-spline construction from boundary T-spline representations. Computational Mechanics 51(2013) 1051-1059.