



HAL
open science

Parametrization of computational domain in isogeometric analysis: methods and comparison

Gang Xu, Bernard Mourrain, Régis Duvigneau, André Galligo

► **To cite this version:**

Gang Xu, Bernard Mourrain, Régis Duvigneau, André Galligo. Parametrization of computational domain in isogeometric analysis: methods and comparison. *Computer Methods in Applied Mechanics and Engineering*, 2010. inria-00530758v1

HAL Id: inria-00530758

<https://inria.hal.science/inria-00530758v1>

Submitted on 29 Oct 2010 (v1), last revised 4 Mar 2011 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Parametrization of computational domain in isogeometric analysis: methods and comparison

Gang Xu^a, Bernard Mourrain^a, Régis Duvigneau^b, André Galligo^c

^a*Galaad, INRIA Sophia-Antipolis, 2004 Route des Lucioles, 06902 Cedex, France*

^b*OPALE, INRIA Sophia-Antipolis, 2004 Route des Lucioles, 06902 Cedex, France*

^c*University of Nice Sophia-Antipolis, 06108 Nice Cedex 02, France*

Abstract

Parameterization of computational domain plays an important role in isogeometric analysis as mesh generation in finite element analysis. In this paper, we investigate this problem in the 2D case, i.e, how to parametrize the computational domains by planar B-spline surface from the given CAD objects (four boundary planar B-spline curves). Firstly, two kinds of sufficient conditions for injective B-spline parameterization are derived with respect to the control points. Then we show how to find good parameterization of computational domain by solving a constraint optimization problem, in which the constraint condition is the injectivity sufficient conditions of planar B-spline parameterization, and the optimization term is the minimization of quadratic energy functions related to the first and second derivatives of planar B-spline parameterization. By using this method, the resulted parameterization has no self-intersections, and the isoparametric net has good uniformity and orthogonality. After introducing a posteriori error estimation for isogeometric analysis, we propose r -refinement method to optimize the parameterization by repositioning the inner control points such that the estimated error is minimized. Several examples are tested on isogeometric heat conduction problem to show the effectiveness of the proposed methods and the impact of the parameterization on the quality of the approximation solution. Comparison examples with known exact solutions are also presented.

Keywords: isogeometric analysis; injectivity of B-spline parameterization; constraint optimization method; a posteriori error estimation; r -refinement

Email addresses: xugangzju@gmail.com (Gang Xu), Bernard.Mourrain@sophia.inria.fr (Bernard Mourrain), Regis.Duvigneau@sophia.inria.fr (Régis Duvigneau), galligo@unice.fr (André Galligo)

1. Introduction

The isogeometric analysis (IGA for short) approach originally proposed by Hughes et al. in [23] can be employed to overcome the gap between CAD and finite element analysis (FEA for short) by using the same geometric representation based on NURBS for all design and analysis tasks. This framework allows to compute the analysis results on the exact CAD geometry rather than a discretized mesh geometry, to obtain a more accurate solution by high-order approximation, to reduce spurious numerical sources of noise that deteriorate convergence, and to avoid data exchanges between the design and analysis phases. Furthermore, NURBS representation allows to perform refinement operations to improve the simulation result by knot insertion and degree elevation.

Mesh generation, which generates a discrete mesh of a computational domain from a given CAD object, is a key and the most time-consuming step in FEA. It consumes about 80% of the overall design and analysis process [7] in automotive, aerospace and ship industry. Parametrization of computational domain in IGA, which corresponds to the mesh generation in FEA, also has some impact on analysis result and efficiency. In particular, arbitrary refinements can be performed on the computational mesh in FEA, but in IGA if we compute with tensor product B-splines, we can only perform refinement operations in u direction and v direction by knot insertion or degree elevation. Hence, parameterization of computational domain is also being important for IGA.

In IGA, the parameterization of a computational domain is determined by control points, knot vectors and the degrees of B-spline objects. For IGA problem of two dimensions, the knot vectors and the degree of the computational domain are determined by the given boundary curves. That is, given boundary curves, the quality of parameterization of computational domain is determined by the positions of inner control points. Hence, finding a good placement of the inner control points inside the computational domain, is a key issue. A basic requirement of the resulting parameterization for IGA is that it doesn't have self-intersections, so that it is an injective map from the parametrization domain to the computational domain. In order to get more accurate simulation results, the isoparametric net in the computational domain should be as uniform as possible and have orthogonal isoparametric curves [8]. In this paper, we investigate this problem in 2D case, i.e. how to construct the computational domains (planar B-spline surface) from the given CAD objects (four boundary planar B-spline curves). Firstly, Two kinds of sufficient conditions for injective B-

spline parameterization are derived with respect to the control points. Then the parameterization of the computational domain is considered as a constraint optimization problem, in which the constraint condition is the injectivity sufficient conditions of planar B-spline parameterization, and the optimization term is the minimization of quadratic energy functions related to the first and second derivatives of the computational domain (B-spline surface). By using this method, the resulted parameterization has no self-intersection, and the isoparametric net has good uniformity and orthogonality.

In IGA, there are three kinds of methods to improve the simulation results by increasing degree of freedom: h -refinement by knot insertion, p -refinement by order elevation, and k -refinement combining order elevation and knot insertion [12, 18, 25]. The common feature of above three methods is that the degree of freedom (control points in IGA) is increased while keeping the geometry of computational domain. In FEA, r -refinement is a remesh operation on computational domain to minimize the cost function while keeping the number of elements as a constant [29]. In this paper, as the second method for optimizing the parametrization of computational domain, based on a residual-based a posteriori error estimation, an isogeometric r -refinement method is proposed to optimize the parameterization of the computational domain by minimizing the estimated error. It is a generalization of the optimal parameterization method proposed in [32], in which only the problem with a known exact solution is investigated. Several examples have been tested on isogeometric heat conduction problems to show the effectiveness of the proposed methods.

The remainder of the paper is organized as follows. Section 2 reviews the related work in IGA. Section 3 presents two kinds of sufficient conditions for injectivity of planar B-spline parameterization. Section 4 describes the constraint optimization method for parameterization of computational domain. r -refinement method and a posteriori error estimation is presented in 5. Some examples and comparisons based on the isogeometric heat conduction problem are presented in Section 6. Finally, we conclude this paper and outline future works in Section 7.

2. Related work

In this section, some related works in IGA and parameterization of computational domains will be reviewed.

IGA was firstly proposed by T.R Hughes et al. [23] in 2005 to achieve the seamless integration

of CAD and FEA. Since then, many researchers in the fields of computational mechanical and geometric computation were involved in this topic. The current work on isogeometric analysis can be classified into three categories: (1) application of IGA to various simulation and analysis problems [3, 6, 5, 13, 17, 22, 24, 33]; (2) application of various modeling tools in geometric computation to IGA [7, 15, 27, 9]; (3) accuracy and efficiency improvement of IGA framework by reparameterization and refinement operations [1, 4, 11, 12, 18, 25, 28, 32].

The topic of this paper belongs to the third field. As far as we know, there are few works on the parametrizations of computational domains for IGA. T. Martin et al. [28] proposed a method to fit a genus-0 triangular mesh by B-spline volume parameterization, based on discrete volumetric harmonic functions; this can be used to build computational domains for 3D IGA problems. A variational approach for constructing NURBS parameterization of swept volumes is proposed by M. Aigner et al [1]. Many free-form shapes in CAD systems, such as blades of turbines and propellers, are covered by this kind of volumes. E. Cohen et.al. [11] proposed the concept of *analysis-aware modeling*, in which the parameters of CAD models should be selected to facilitate isogeometric analysis. They also demonstrated the influence of parameterization of computational domains by several examples. Approximate implicitization technique is used for parametrization of computational domain in [31]. In [32], a method for generating optimal analysis-aware parameterization of computational domain is proposed based on shape optimization method. However, it only works for problems with exact solutions. In this paper, we propose a general method to generate analysis-suitable parameterization of computational domain in IGA.

3. Sufficient conditions for injective B-spline parameterization

The concept of isogeometric analysis consists in representing the physical quantities $\Phi \in \mathbb{R}^p$ on the geometry Ω using the same type of B-spline representation as for the geometry Ω . In other words, given a point $\mathbf{x} = \sigma(\xi, \eta) \in \Omega$ with $(\xi, \eta) \in \mathcal{P}$, we associate to it the physical quantity $\Phi(\xi, \eta)$ where $\Phi(\xi, \eta)$ is a B-spline function with nodes in \mathcal{P} and control points in \mathbb{R}^p . This means that the map $\mathbf{x} \in \Omega \mapsto \Phi \in \mathbb{R}^p$ is defined *implicitly* as $\mathbf{x} \mapsto \Phi \circ \sigma^{-1}(\mathbf{x})$.

Consequently, the framework of isogeometry is thus valid when the parameterization σ of the geometry is *injective* (or bijective on its image). We will consider this issue in the context of finding a “good” parameterization of a domain when its boundary is given. In [26], a general sufficient

condition is proposed for injective parameterization.

Proposition 3.1. *Suppose that σ is a C^1 parameterization from a compact domain $\mathcal{P} \subset \mathbb{R}^n$ with a connected boundary to a geometry $\Omega \subset \mathbb{R}^n$. If σ is injective on the boundary $\partial\mathcal{P}$ of \mathcal{P} and its Jacobian J_σ does not vanish on \mathcal{P} , then σ is injective.*

For a parameterization σ from $\mathcal{P} : [a, b] \times [c, d]$ to $\Omega \subset \mathbb{R}^2$, we define the boundary curves as the image of $\{a\} \times [c, d]$, $\{b\} \times [c, d]$, $[a, b] \times \{c\}$, $[a, b] \times \{d\}$ by σ . We say that σ defines a *regular boundary* if these curves do not intersect pairwise, except at their end points and if they have no self-intersection points.

As a consequence of the previous proposition, we get the following injectivity test for standard B-spline tensor product parameterization of a planar domain.

Proposition 3.2. *Let σ be a C^1 parameterization from $[a, b] \times [c, d]$ to $\Omega \subset \mathbb{R}^2$ which defines a regular boundary. If its Jacobian J_σ does not vanish on $[a, b] \times [c, d]$, then σ is injective.*

We consider first the case of a planar parameterization

$$\sigma : (\xi, \eta) \in \mathcal{P} := [a, b] \times [c, d] \mapsto \sigma(\xi, \eta) := \sum_{\substack{0 \leq i \leq n \\ 0 \leq j \leq m}} \mathbf{c}_{i,j} N_i^p(\xi) N_j^q(\eta),$$

where $\mathbf{c}_{i,j} \in \mathbb{R}^2$ are the control points, $N_i^p(\xi)$ and $N_j^q(\eta)$ are B-spline basis functions of degree p and q for a given knot vector on $[a, b]$ and $[c, d]$.

The derivative of $\sigma(\xi, \eta)$ with respect to ξ can be expressed in terms of the differences $\Delta_{i,j}^1 := (\Delta_{i,j}^{1,x}, \Delta_{i,j}^{1,y}) = \mathbf{c}_{i+1,j} - \mathbf{c}_{i,j}$:

$$\partial_\xi \sigma(\xi, \eta) := \sum_{\substack{0 \leq i \leq n-1 \\ 0 \leq j \leq m}} \omega_{i,j}^1 \Delta_{i,j}^1 N_i^{p-1}(\xi) N_j^q(\eta), \quad (1)$$

where $N_i^{p-1}(\xi)$ is the B-spline basis function with one degree less in u , $\omega_{i,j}^1$ is a positive factor.

Similarly, the derivative of $\sigma(\xi, \eta)$ with respect to η can be expressed in terms of the differences $\Delta_{i,j}^2 := (\Delta_{i,j}^{2,x}, \Delta_{i,j}^{2,y}) = \mathbf{c}_{i,j+1} - \mathbf{c}_{i,j}$:

$$\partial_\eta \sigma(\xi, \eta) := \sum_{\substack{0 \leq i \leq n \\ 0 \leq j \leq m-1}} \omega_{i,j}^2 \Delta_{i,j}^2 N_i^p(\xi) N_j^{q-1}(\eta), \quad (2)$$

where $N_j^{q-1}(\eta)$ is the B-spline basis with one degree less in v , $\omega_{i,j}^2$ is a positive factor.

3.1. Non-linear sufficient condition based on Jacobian computation

From Proposition 3.2, if the Jacobian determinant $\mathbf{J}(\sigma(\xi, \eta))$ of planar B-spline parameterization satisfies $\mathbf{J}(\sigma(\xi, \eta)) > 0$, $\sigma(\xi, \eta)$ has no self-intersections.

From (1), (2) and the product properties of B-splines [30], the Jacobian determinant $\mathbf{J}(\sigma(\xi, \eta))$ of B-spline surface can be computed as follows:

$$\begin{aligned} \mathbf{J}(\sigma(\xi, \eta)) &= \begin{vmatrix} \sigma_\xi^x & \sigma_\eta^x \\ \sigma_\xi^y & \sigma_\eta^y \end{vmatrix} = \sigma_\xi^x \sigma_\eta^y - \sigma_\xi^y \sigma_\eta^x \\ &= \sum_{\substack{0 \leq i \leq n-1 \\ 0 \leq j \leq m}} \sum_{\substack{0 \leq i' \leq n \\ 0 \leq j' \leq m-1}} N_i^{p-1}(\xi) N_j^q(\eta) N_{i'}^p(\xi) N_{j'}^{q-1}(\eta) \omega_{i,j}^1 \omega_{i',j'}^2 \begin{vmatrix} \Delta_{i,j}^{1,x} & \Delta_{i,j}^{2,x} \\ \Delta_{i',j'}^{1,y} & \Delta_{i',j'}^{2,y} \end{vmatrix} \quad (3) \\ &= \sum_{i=0}^{2n-1} \sum_{j=0}^{2m-1} G_{ij} N_i^{2p-1}(\xi) N_i^{2q-1}(\eta) \quad (4) \end{aligned}$$

Hence, the Jacobian of planar B-spline parameterization can be represented in the form of B-spline surface with higher degrees. From the convex hull property of B-spline surface [19], we have the following theorem.

Theorem 3.3. *If $G_{ij} > 0$ in (4), then $\mathbf{J}(\sigma(\xi, \eta)) > 0$, that is, $\sigma(\xi, \eta)$ has no self-intersections.*

This is a non-linear sufficient condition with respect to the inner control points. In Section 4, we will employ it as constraint term in the constraint optimization method for general case.

Notice that if the coefficients $\omega_{i,j}^1 \omega_{i',j'}^2 \begin{vmatrix} \Delta_{i,j}^{1,x} & \Delta_{i,j}^{2,x} \\ \Delta_{i',j'}^{1,y} & \Delta_{i',j'}^{2,y} \end{vmatrix}$ in the expansion (3), we also obtain a sufficient condition for $\mathbf{J}(\sigma(\xi, \eta)) \neq 0$. But the conditions $G_{ij} > 0$ are weaker than these conditions, since the coefficient matrix which expresses the elements $N_i^{p-1}(\xi) N_j^q(\eta) N_{i'}^p(\xi) N_{j'}^{q-1}(\eta)$ in terms of the elements $(N_i^{2p-1}(\xi) N_i^{2q-1}(\eta))_{1 \leq i \leq 2n-1, 1 \leq j \leq 2m-1}$, has positive coefficients.

3.2. Linear sufficient condition using injectivity cones

In this subsection, we propose a linear sufficient condition for injectivity of a B-spline parameterization.

We denote by $\mathcal{C}_1(\mathbf{c})$ the convex cone of \mathbb{R}^2 generated by the half rays $\mathbb{R}_+ \cdot \Delta_{i,j}^1$. We denote by $\mathcal{C}_2(\mathbf{c})$ the convex cone of \mathbb{R}^2 generated by the half rays $\mathbb{R}_+ \cdot \Delta_{i,j}^2$. If this cone is generated by two opposite vectors, which are on a straight line, we define $\mathcal{C}_i(\mathbf{c})$ as any half-plane.

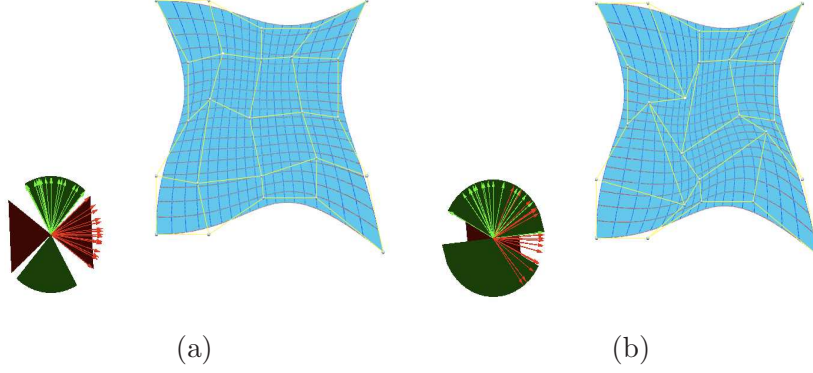


Fig. 1. Injectivity test using cones: (a) transverse case; (b) non-transverse case.

We say that two cones $\mathcal{C}_1, \mathcal{C}_2$ are *transverse* if $\mathbb{R} \cdot \mathcal{C}_1$ and $\mathbb{R} \cdot \mathcal{C}_2$ intersect only at $\{0\}$.

In [32], a sufficient and easy-to-check condition is proposed to ensure the local injectivity of σ .

Proposition 3.4. *Let σ be a B-spline parametrisation, which is at least C^1 from $\mathcal{P} := [a, b] \times [c, d]$ to $\Omega \subset \mathbb{R}^2$ given by the control points \mathbf{c} . If the boundary curves do not intersect and have no self-intersection point and the cones $\mathcal{C}_1(\mathbf{c}), \mathcal{C}_2(\mathbf{c})$ are transverse, then σ is injective on \mathcal{P} .*

Fig. 1 shows two examples of the injectivity testing method.

This linear condition can be used to devise an algorithm which constructs an injective parameterization from given boundary control points. Given four planar boundary curves described by the controls points $\mathbf{c}_{i,0}, \mathbf{c}_{i,m}, \mathbf{c}_{0,j}, \mathbf{c}_{n,j}$, with $0 \leq i \leq n, 0 \leq j \leq m$, we define the boundary cone $\mathcal{C}_1^0(\mathbf{c})$ (resp. $\mathcal{C}_2^0(\mathbf{c})$) as the cone generated by the vectors $\Delta_{i,0}^1(\mathbf{c}), \Delta_{i,m}^1(\mathbf{c})$ for $0 \leq i \leq n-1$ (resp. $\Delta_{0,j}^2(\mathbf{c}), \Delta_{m,j}^2(\mathbf{c})$ for $0 \leq j \leq m-1$). We assume that these two boundary cones $\mathcal{C}_1^0(\mathbf{c}), \mathcal{C}_2^0(\mathbf{c})$ are transverse and $\mathbb{R} \cdot \mathcal{C}_1^0(\mathbf{c})$ is the boundary cone defined by $F_1^+(\mathcal{C}_1^0(\mathbf{c})) \leq 0, F_1^-(\mathcal{C}_1^0(\mathbf{c})) \leq 0$, where F_1^+ and F_1^- are the linear equations defining the boundary of $\mathbb{R} \cdot \mathcal{C}_1^0(\mathbf{c})$. We defined similarly F_2^+, F_2^- for $\mathcal{C}_2^0(\mathbf{c})$.

In [32], the following linear sufficient condition for injective parameterization are proposed,

$$\begin{cases} F_2^+(\mathbf{c}_{i+1,j} - \mathbf{c}_{i,j}) + F_1^-(\mathbf{c}_{i+1,j} - \mathbf{c}_{i,j}) \leq 0, & F_2^-(\mathbf{c}_{i+1,j} - \mathbf{c}_{i,j}) + F_1^+(\mathbf{c}_{i+1,j} - \mathbf{c}_{i,j}) \leq 0, \\ F_2^+(\mathbf{c}_{i,j+1} - \mathbf{c}_{i,j}) + F_1^-(\mathbf{c}_{i,j+1} - \mathbf{c}_{i,j}) \leq 0, & F_2^-(\mathbf{c}_{i,j+1} - \mathbf{c}_{i,j}) + F_1^+(\mathbf{c}_{i,j+1} - \mathbf{c}_{i,j}) \leq 0, \end{cases} \quad (5)$$

where $0 < i < n, 0 \leq j < m$.

This set of conditions provides linear constraints for the injectivity of planar B-spline parameterization. In the following section, we will employ it as constraint term in the quadratic programming method for the case when the boundary cones are transverse.

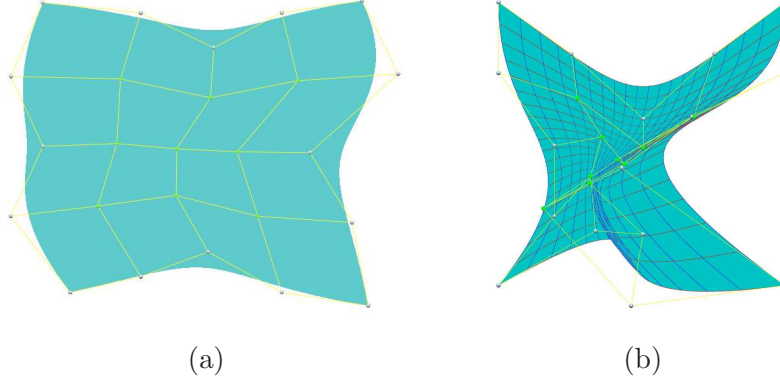


Fig. 2. Examples of Coons surface: (a) example without self-intersection ; (b) example with self-intersections. Green control points are inner control points generated by discrete Coons method.

4. Constraint optimization method for parametrization of computational domains

In this section, we aim at finding injective parameterization of the computational domain with a uniform and orthogonal isoparametric net.

4.1. Initial construction of inner control points

In order to solve this constraint optimization problem, an initial construction of inner control points is required. We rely on the discrete Coons method presented in [20] to generate inner control points as initial value from boundary control points. See Fig. 2 (a) for an example.

Given the boundary control points $\mathbf{c}_{0,j}, \mathbf{c}_{n,j}, \mathbf{c}_{i,0}, \mathbf{c}_{i,m}$, $i = 0, \dots, n, j = 0, \dots, m$, the inner control points $\mathbf{c}_{i,j}$ ($i = 1, \dots, n-1, j = 1, \dots, m-1$) can be constructed by the discrete Coons method as follows:

$$\begin{aligned} \mathbf{c}_{i,j} = & (1 - \frac{i}{n})\mathbf{c}_{0,j} + \frac{i}{n}\mathbf{c}_{n,j} + (1 - \frac{j}{m})\mathbf{c}_{i,0} + \frac{j}{m}\mathbf{c}_{i,m} \\ & - [1 - \frac{i}{n} \quad \frac{i}{n}] \begin{pmatrix} \mathbf{c}_{0,0} & \mathbf{c}_{0,m} \\ \mathbf{c}_{n,0} & \mathbf{c}_{n,m} \end{pmatrix} \begin{pmatrix} 1 - \frac{j}{m} \\ \frac{j}{m} \end{pmatrix} \end{aligned}$$

Remark 4.1. Since the sum of the coefficients equals 1, the resulting inner control points lie in the convex hull of the boundary control points.

Remark 4.2. For some given boundary curves, this construction may cause some self-intersections, and lead to an improper parameterization for IGA. See Fig. 2 (b) for such an example.

4.2. Non-linear constraint optimization method for uniform and orthogonal grid

An internal energy function of the computational domain will be used as an optimization term in a constraint optimization method to construct a computational domain with an uniform and orthogonal isoparametric net.

A parametric surface $\sigma(\xi, \eta)$ has orthogonal isoparametric curves if and only if it satisfies the following condition

$$\sigma_\xi \cdot \sigma_\eta = 0. \quad (6)$$

Hence by minimizing the following energy functions, one can achieve isoparametric net with orthogonal isoparametric curves,

$$\int (\sigma_\xi \cdot \sigma_\eta)^2 d\xi d\eta. \quad (7)$$

As $|\sigma_\xi \cdot \sigma_\eta| \leq \|\sigma_\xi\| \cdot \|\sigma_\eta\| \leq \frac{\|\sigma_\xi\|^2 + \|\sigma_\eta\|^2}{2}$, in order to get quadratic energy functions with respect to interior control points, we can minimize the following energy function

$$\iint \|\sigma_\xi\|^2 + \|\sigma_\eta\|^2 d\xi d\eta. \quad (8)$$

The uniformity of the iso-parametric net on the B-spline surface is related to the following quantity:

$$\iint (\|\sigma_{\xi\xi}\|^2 + \|\sigma_{\eta\eta}\|^2 + 2\|\sigma_{\xi\eta}\|^2) d\xi d\eta. \quad (9)$$

In [10], the above two energy functions (8) and (9) are called stretch energy and bend energy respectively, which depend only on intrinsic property of the surface. In summary, combining (8) and (9), we use the following optimization term as objective function,

$$\min \iint (\|\sigma_{\xi\xi}\|^2 + \|\sigma_{\eta\eta}\|^2 + 2\|\sigma_{\xi\eta}\|^2) + \omega(\|\sigma_\xi\|^2 + \|\sigma_\eta\|^2) d\xi d\eta$$

where ω is a positive constant.

This leads to the following constraint optimization algorithm based on the non-linear sufficient condition in subsection 3.1:

Input: four coplanar boundary B-spline curves

Output: inner control points and the corresponding planar B-spline surfaces

- Construct the initial inner control points as in subsection 4.1;
- Construct the constraints condition (5) from boundary B-spline curves as in Section 3;

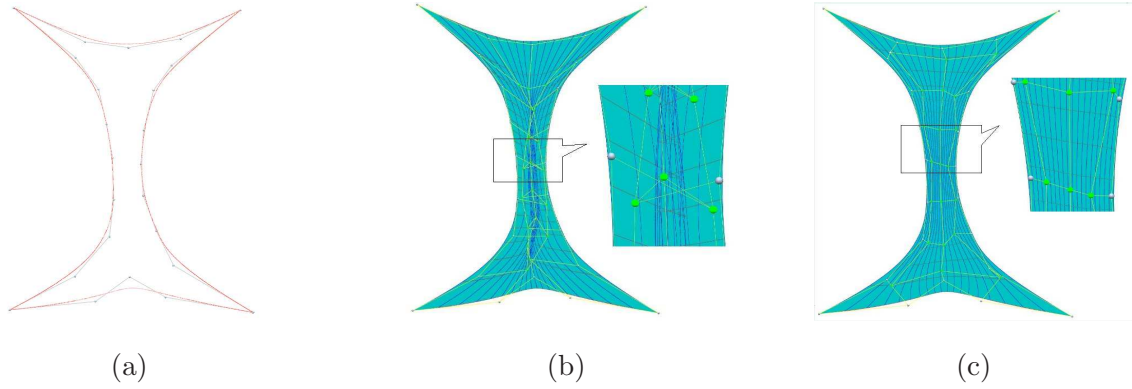


Fig. 3. Example of non-linear constraint optimization method: (a) given boundary B-spline curves; (b) initial Coons surface; (c) final optimization result;

- Solve the following constraint optimization problem by using SQP method

$$\begin{aligned} \min \quad & \iint (\| \sigma_{\xi\xi} \|^2 + \| \sigma_{\eta\eta} \|^2 + 2 \| \sigma_{\xi\eta} \|^2) + \omega(\| \sigma_{\xi} \|^2 + \| \sigma_{\eta} \|^2) d\xi d\eta \\ \text{s.t.} \quad & G_{ij} > 0 \end{aligned}$$

- Generate the corresponding planar B-spline surface $\sigma(\xi, \eta)$ as computational domain.

As this method is a non-linear constraint optimization algorithm, it can be used also for more general cases, including the case where the boundary injectivity cones are non-transverse. An example of the application of this method is illustrated in Fig. 3. Fig. 3 (a) presents the given boundary curves. Fig. 3 (b) shows the initial Coons surface with self-intersections. The optimization result by non-linear constraint optimization method is shown in Fig. 3 (c). The optimization result avoids self-intersection and gives more uniform iso-parametric grid.

4.3. Quadratic programming method for parametrization of computational domain

If the boundary injectivity cones are transverse, we can propose a quadratic programming method for parameterization of computational domain as follows:

Input: four coplanar boundary B-spline curves

Output: inner control points and the corresponding planar B-spline surfaces

- Construct the initial inner control points as in subsection 4.1;
- Construct the constraints condition (5) from boundary B-spline curves as in Section 3;

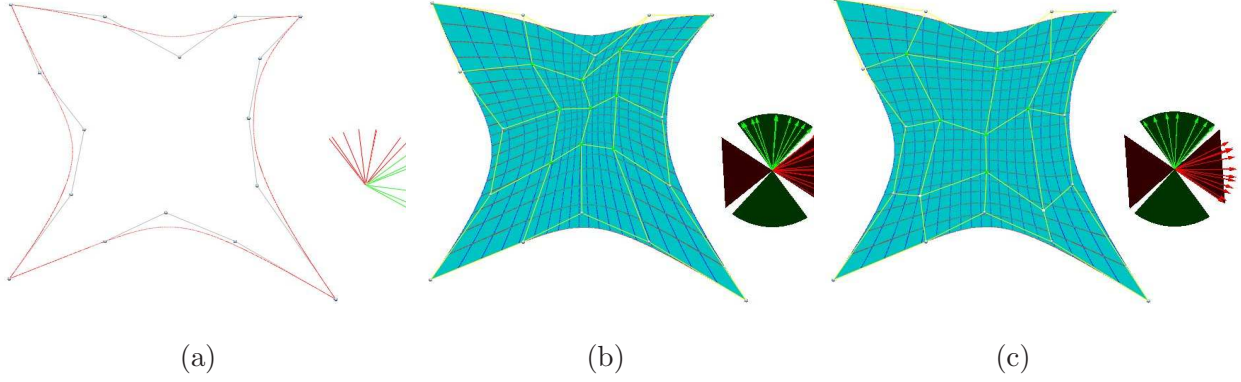


Fig. 4. Example of quadratic programming method: (a) given boundary B-spline curves and boundary cones; (b) initial Coons surface; (c) final optimization result.

- Solve the following constraint optimization problem by using a quadratic programming method,

$$\begin{aligned}
 \min \quad & \iint (\|\sigma_{\xi\xi}\|^2 + \|\sigma_{\eta\eta}\|^2 + 2\|\sigma_{\xi\eta}\|^2) + \omega(\|\sigma_{\xi}\|^2 + \|\sigma_{\eta}\|^2) d\xi d\eta \\
 \text{s.t.} \quad & \begin{cases} F_2^+(\mathbf{c}_{i+1,j} - \mathbf{c}_{i,j}) + F_1^-(\mathbf{c}_{i+1,j} - \mathbf{c}_{i,j}) \leq 0, F_2^-(\mathbf{c}_{i+1,j} - \mathbf{c}_{i,j}) + F_1^+(\mathbf{c}_{i+1,j} - \mathbf{c}_{i,j}) \leq 0, \\ F_2^+(\mathbf{c}_{i,j+1} - \mathbf{c}_{i,j}) + F_1^-(\mathbf{c}_{i,j+1} - \mathbf{c}_{i,j}) \leq 0, F_2^-(\mathbf{c}_{i,j+1} - \mathbf{c}_{i,j}) + F_1^+(\mathbf{c}_{i,j+1} - \mathbf{c}_{i,j}) \leq 0, \end{cases}
 \end{aligned}$$

- Generate the corresponding planar B-spline surface $\sigma(\xi, \eta)$ as computational domain.

Fig. 4 illustrates an example. The given boundary B-spline curves and boundary injectivity cones are shown in Fig. 4(a). The initial Coons surfaces is shown in Fig. 4 (b), and we can obtain a better parameterization as illustrated in Fig. 4 (c) by quadratic programming method.

Remark 4.3. *This second method is more efficient because the constraint conditions are linear and easy to compute.*

5. r -refinement in isogeometric analysis

In this section, we will propose the isogeometric version of r -refinement for problems with unknown exact solutions, which is a generalization of the method presented in [32]. We firstly propose a residual-based a posteriori error estimation for specified model problem, then present r -refinement method to optimize the placement of inner control points.

5.1. Model problem

For ease of presentation, we consider the two dimensional second order elliptic PDE with homogeneous Dirichlet boundary condition as an illustrative model problem :

$$\begin{aligned} -\Delta U(\mathbf{x}) &= f(\mathbf{x}) & \text{in } \Omega \\ U(\mathbf{x}) &= 0 & \text{on } \partial\Omega \end{aligned} \tag{10}$$

where \mathbf{x} are the Cartesian coordinates, Ω is a Lipschitz domain with boundary $\partial\Omega$, $f(\mathbf{x}) \in L^2(\Omega)$: $\Omega \mapsto \mathbb{R}$ is a given source term, and $U(\mathbf{x}) : \Omega \mapsto \mathbb{R}$ is the unknown solution.

The weak form of (10) is given by: find $U(\mathbf{x}) \in H_0^1(\Omega)$, such that

$$B(U, V) = (f, V), \quad \forall V \in H_0^1(\Omega) \tag{11}$$

where

$$\begin{aligned} B(U, V) &= \int_{\Omega} \nabla U(\mathbf{x}) \cdot \nabla V(\mathbf{x}) \, d\mathbf{x}, \\ (f, V) &= \int_{\Omega} f(\mathbf{x}) V(\mathbf{x}) \, d\mathbf{x}, \quad \forall U, V \in H_0^1(\Omega) \end{aligned}$$

The *energy norm* and $L^2(\Omega)$ *norm* of $U \in H_0^1(\Omega)$ are defined by $\|U\|_{E(\Omega)} = \sqrt{B(U, U)}$ and $\|U\|_{L^2(\Omega)} = \sqrt{(U, U)}$, respectively.

5.2. Residual-based a posteriori error estimate

The isogeometric solution $U_h \in \mathcal{U}(\sigma)$ of the problem (10) satisfies

$$B(U_h, V_h) = (f, V_h), \quad \forall V_h \in \mathcal{U}(\sigma), \tag{12}$$

where $\mathcal{U}(\sigma) \subset H_0^1(\Omega)$ is a suitable subspace of $H_0^1(\Omega)$ based on a parametrization σ of Ω .

We assume that U is the exact solution. Let $e_h = U - U_h$ be the error of the isogeometric approximation U_h . From (11) and (12), we have for arbitrary $V \in H_0^1(\Omega)$

$$\begin{aligned} B(e_h, V) &= B(U, V) - B(U_h, V) = (f, V) - B(U_h, V) \\ &= \int_{\Omega} f(\mathbf{x}) V(\mathbf{x}) \, d\mathbf{x} - \int_{\Omega} \nabla U_h(\mathbf{x}) \cdot \nabla V(\mathbf{x}) \, d\mathbf{x} \\ &= \sum_{K \in \sigma} \left\{ \int_K f(\mathbf{x}) V(\mathbf{x}) \, d\mathbf{x} - \int_K \nabla U_h(\mathbf{x}) \cdot \nabla V(\mathbf{x}) \, d\mathbf{x} \right\} \end{aligned} \tag{13}$$

where K is the sub-patch in the B-spline parameterization $\sigma(\xi, \eta)$ of Ω with respect to the knot elements in the parametric domain \mathcal{P} .

After integrating by parts and rearranging terms, we obtain

$$B(e_h, V) = \sum_{K \in \sigma} \int_K R_K(\mathbf{x}) V(\mathbf{x}) \, d\mathbf{x} + \sum_{\gamma \in \mathcal{Z}_h} \int_\gamma J_\gamma V(\mathbf{x}) \, d\mathbf{x} \quad (14)$$

where R_K is the interior sub-patch residual

$$R_K(\mathbf{x}) = f(\mathbf{x}) + \Delta U_h(\mathbf{x}) \quad \text{in } K, \quad (15)$$

and J_γ is the jump of the gradient across the curved edge $\gamma \in \mathcal{Z}_h$ between sub-patches,

$$J_\gamma = \begin{cases} \mathbf{n} \cdot \nabla U_h + \mathbf{n}' \cdot \nabla U_h', & \text{if } \gamma \notin \partial\Omega \\ 0, & \text{if } \gamma \in \partial\Omega \end{cases} \quad (16)$$

where on curved edges between interior patches, $\gamma \notin \partial\Omega$, the curved edge γ separates sub-patches K and K' .

Notice that $J_\gamma = 0$ in (16) due to the continuity of ∇U_h in isogeometric analysis. Hence, we have

$$B(e_h, V) = \sum_{K \in \sigma} \int_K R_K(\mathbf{x}) V(\mathbf{x}) \, d\mathbf{x} \quad (17)$$

For given $V \in H_0^1(\Omega)$, denote by $I_h V$ the interpolation of V on $\mathcal{U}(\sigma)$. Then the orthogonality property implies $B(e_h, I_h V) = 0$, and by (17), we obtain

$$B(e_h, V) = \sum_{K \in \sigma} \int_K R_K(\mathbf{x}) (V(\mathbf{x}) - I_h V(\mathbf{x})) \, d\mathbf{x}, \quad (18)$$

From Cauchy-Schwartz inequality,

$$B(e_h, V) \leq \sum_{K \in \sigma} \int_K \|R_K(\mathbf{x})\|_{L^2(K)} \|V(\mathbf{x}) - I_h V(\mathbf{x})\|_{L^2(K)} \, d\mathbf{x}, \quad (19)$$

Letting $V = e_h$, from the properties of bilinear form $B(\cdot, \cdot)$ and the approximation theory [2], we can obtain the residual-based a posteriori error estimate in isogeometric analysis:

$$\|e_h\|_{E(\Omega)}^2 \leq C \sum_{K \in \sigma} h_K^2 \|R_K\|_{L^2(K)}^2 \quad (20)$$

where C is a constant and h_K is the diameter of patch $K \in \sigma$.

From (15), the crucial point of a posteriori error estimation in (20) is the computation of $\Delta U_h(\mathbf{x})$. The following proposition shows how to compute the error estimation directly on the parametric domain:

Proposition 5.1. *Given the parameterization $\sigma(\xi, \eta) = (x(\xi, \eta), y(\xi, \eta))$ of the computational domain and the solution field $U_h(\mathbf{x}) = U_h(x(\xi, \eta), y(\xi, \eta)) = \mathcal{T}(\xi, \eta)$ over $\sigma(\xi, \eta)$, then $\Delta U_h(\mathbf{x})$ has the following form,*

$$\Delta U_h = \frac{\partial^2 U_h}{\partial^2 x} + \frac{\partial^2 U_h}{\partial^2 y} = \frac{JL_\xi \mathcal{T}_{\eta\eta} - JL_\eta \mathcal{T}_{\xi\xi} + L_{\xi\xi} \mathcal{T}_\eta - L_{\eta\eta} \mathcal{T}_\xi}{JK},$$

where

$$J = x_\xi y_\eta - x_\eta y_\xi, K = (x_\xi y_\eta)^2 - (x_\eta y_\xi)^2,$$

$$L_\xi = x_\xi^2 - y_\xi^2, L_\eta = x_\eta^2 - y_\eta^2,$$

$$L_{\xi\xi} = (L_\eta y_{\xi\xi} - L_\xi y_{\eta\eta})x_\xi - (L_\eta x_{\xi\xi} - L_\xi x_{\eta\eta})y_\xi,$$

$$L_{\eta\eta} = (L_\eta y_{\xi\xi} - L_\xi y_{\eta\eta})x_\eta - (L_\eta x_{\xi\xi} - L_\xi x_{\eta\eta})y_\eta.$$

The proof of Proposition 5.1 can be found in Appendix A. As we can see, there is no need of function expressions on the physical domain, which is consistent with the main idea of isogeometric analysis.

5.3. Optimization method

In the proposed approach, we minimize the estimated error from the IGA solution, by moving inner control points of the computational domain. Therefore, we consider as optimization variables the coordinates of the inner control points and as cost function the error of the IGA solution. The optimization algorithm used for this study is a classical steepest-descent method in conjunction with a back-tracking line-search. For this exercise, the gradient of the cost function is approximated using a centered finite-differencing scheme.

Each iteration k of the optimization algorithm can be summarized as follows, starting from a point x_k in the variable space:

1. Evaluation of perturbed points $x_k + \epsilon e_k$
2. Estimation of the gradient $\nabla f(x_k)$ by finite-difference
3. Define search direction $d_k = -\nabla f(x_k)$
4. Line search : find ρ such as $f(x_k + \rho d_k) < f(x_k)$

These steps are carried out until a stopping criterion is satisfied.

5.4. Overview of r -refinement method in IGA

Based on the proposed error estimation, the procedure of r -refinement for model problem (10) can be described as follows,

Input: initial planar B-spline parameterization $\sigma(\xi, \eta)$ of computational domain Ω

Output: final planar B-spline parametrization $\sigma_{final}(\xi, \eta)$ of Ω after r -refinement

- compute the isogeometric solution $U_h(\mathbf{x}) = U_h(x(\xi, \eta), y(\xi, \eta)) = \mathcal{T}(\xi, \eta)$ of model problem (10) over given parameterization $\sigma(\xi, \eta)$ of computational domain Ω ;
- compute $\sum_{K \in \sigma} h_K^2 \|f(\mathbf{x}) + \Delta U_h(\mathbf{x})\|_{L^2(K)}^2$ according to Proposition 5.1;
- reposition inner control points by minimizing $\sum_{K \in \sigma} h_K^2 \|f(\mathbf{x}) + \Delta U_h(\mathbf{x})\|_{L^2(K)}^2$ with optimization method in Subsection 5.3 ;
- output final placement of inner control points and corresponding planar B-spline parameterization.

Fig. 5 shows an example from [32], which has exact solution for problem (10) over the given computational domain in Fig. 5 (a). Initial error color map computed as difference between exact solution and IGA solution is shown in Fig. 5 (b), final computational domain by r -refinement method based on the estimated error is illustrated in Fig. 5 (c), Fig. 5 (d) shows the error color map obtained from computational domain in Fig. 5 (c); the final computational domain in [32] obtained by minimizing the exact error is shown Fig. 5 (e), Fig. 5 (f) illustrates the error color map obtained from computational domain in Fig. 5 (e). The color maps in Fig. 5 (d) and Fig. 5 (f) have the same scale as in Fig. 5 (b). From this example, we can find that r -refinement method can move the inner control points to the region with big errors, and based on the proposed error estimator, r -refinement can achieve a very similar result as the method in [32].

6. Examples and comparison

Starting from a planar B-spline surface as computational domain, a general framework of an isogeometric solver for heat conduction problem (10) has been implemented as a plugin in the AXEL¹ platform, yielding a B-spline surface as solution field. Additional details concerning the

¹<http://axel.inria.fr/>

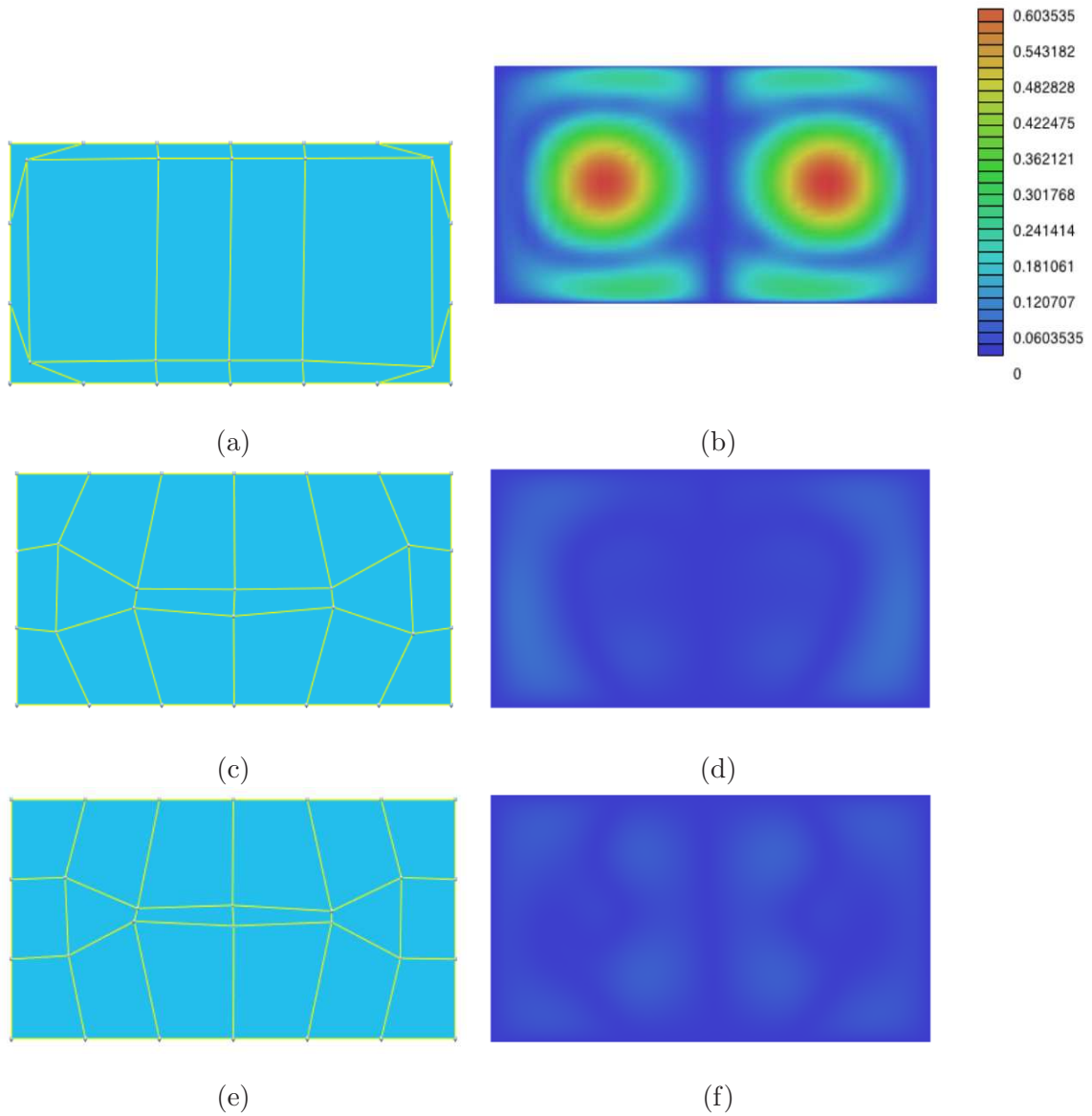


Fig. 5. Example of r -refinement method: (a) initial computational domain; (b) initial error color map; (c) final computational domain by r -refinement method based on the estimated error ; (d) error color map obtained from computational domain in (c); (e) final computational domain in [32] obtained by minimizing the exact error; (f) error color map obtained from computational domain in (e). The color map in (d) and (f) has the same scale as (b).

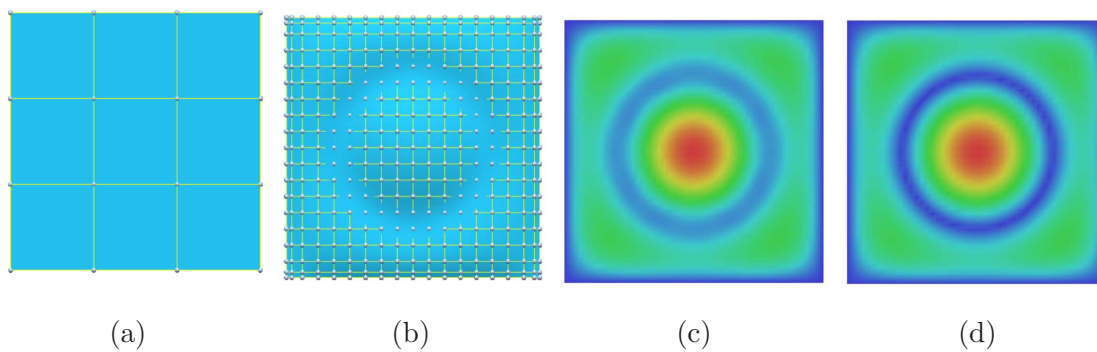


Fig. 6. Example of error assessment method: (a) computational domain; (b) error surface with control points obtained by solving (22); (c) color map of approximate error surface in (b); (d) exact error color map.

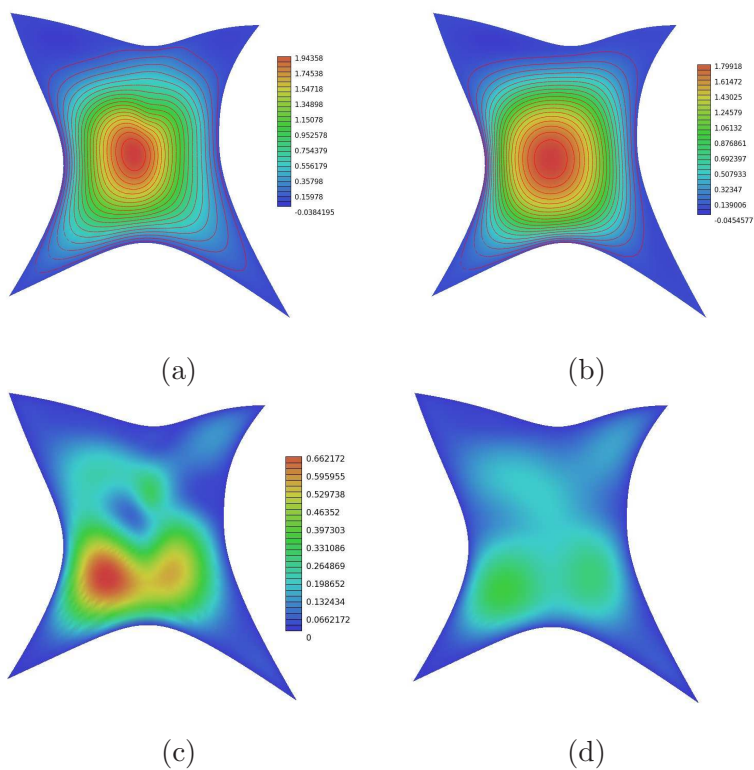


Fig. 7. Example of quadratic programming method: (a) initial simulation result; (b) final simulation result; (c) initial simulation error; (d) final simulation error with same scale.

isogeometric solver of problem (10) can be found in [16]. The proposed constraint optimization methods and r -refinement method are implemented as a part of the isogeometric toolbox of the project EXCITING².

In this paper, we test the different parameterizations of computational domains for the heat conduction problem (10) with source term

$$\mathbf{f}(x, y) = \frac{4\pi^2}{9} \sin\left(\frac{\pi x}{3}\right) \sin\left(\frac{\pi y}{3}\right). \quad (21)$$

For problems with unknown exact solution U , suppose that U_h is the approximation solution obtained by isogeometric method, then the discrete error $e = U - U_h$. Hence, a posteriori error assessment can be obtained by resolving the following problem,

$$\begin{aligned} \Delta e &= -f + \Delta U_h && \text{in } \Omega \\ e &= 0 && \text{on } \partial\Omega_D \end{aligned} \quad (22)$$

The approximation error e from (22) also has a B-spline form. In order to achieve more accurate results for above problem, some h-refinement operation should be performed. An example is shown in Fig. 6. Here $\Omega(x, y) = [0, 3] \times [0, 3]$. We can obtain a good approximation of error surface by four h-refinement as shown in 6 (b). Though it is much more expensive, we can use it as an error assessment method to show the effectiveness of the proposed construction method of computational domain.

Fig. 7 (a) presents the simulation result based on the initial parametrization of computational domain in Fig. 4 (b), the simulation result based on the final parametrization of computational domain in Fig. 4 (c) is shown in Fig. 7 (b). The corresponding simulation error obtained from (22) are illustrated in Fig. 7 (c) and 7 (d) with same scale. The final parametrization obtained by the quadratic programming method can achieve better simulation results than the initial Coons parametrization.

Two examples of r -refinement method are shown in Fig. 8 and Fig. 9. The initial B-spline parameterizations are presented in Fig. 8 (a) and Fig. 9 (a), Fig. 8 (b) and Fig. 9 (b) show the final parametrization obtained by r -refinement. The corresponding simulation error obtained from 22 for initial and final parametrization are illustrated in Fig. 8 (c), Fig. 9 (c), Fig. 8 (d) and Fig. 9 (d). From these two examples, we can find that by r -refinement method some inner control points are moved to the region with big error computed from (22).

²<http://exciting-project.eu/>

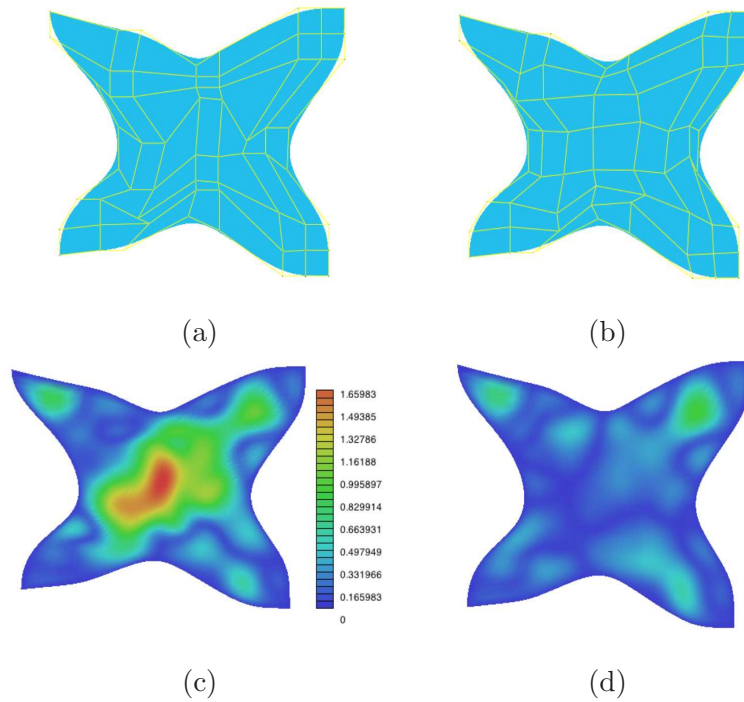


Fig. 8. Example II of r-refinement method: (a) initial computational domain; (b) final computational domain ; (c) initial simulation error; (d) final simulation error with same scale.

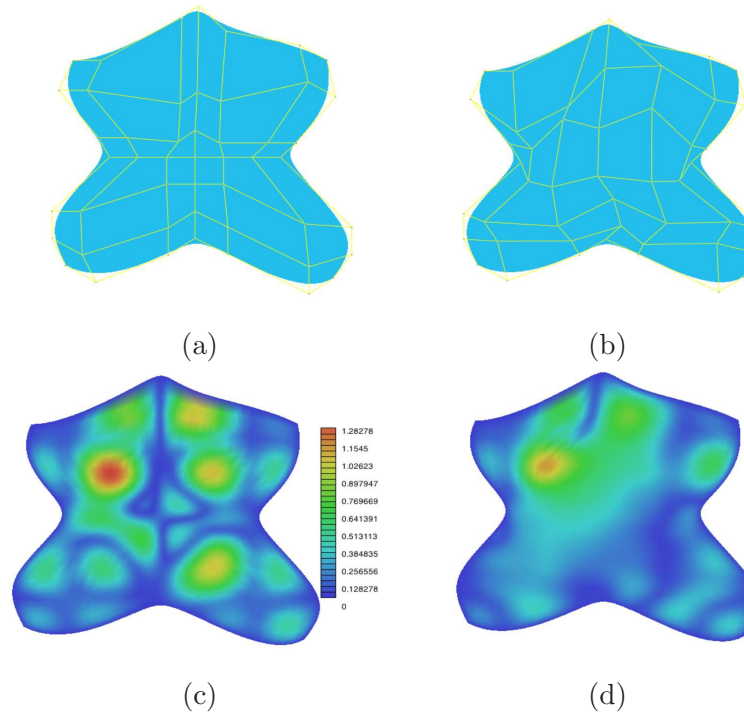


Fig. 9. Example III of r-refinement method: (a) initial parametrization of computational domain; (b) final parametrization of computational domain ; (c) initial simulation error; (d) final simulation error with same scale.

7. Conclusion and future work

Parameterization of computational domains is the first step in an IGA process, and has some effect on simulation results. In this paper, we firstly propose constraint optimization method to parameterize the computational domains with good uniformity and orthogonality from the given four boundary planar B-spline curves. Then r -refinement method in isogeometric version is proposed to optimize the parameterization by repositioning the inner control points based on a posteriori error estimation. Examples and comparison are presented to show that the proposed methods can produce analysis-suitable parameterization of computational domain for IGA.

As part of the future work, we will generalize the proposed methods to 3D case, which is more important in practice. In particular, local r -refinement in IGA is our ongoing work for computational domain with complex geometry.

Acknowledgements

The authors are supported by the 7th Framework Program of the European Union, project SCP8-218536 “EXCITING”. The first author is partially supported by the National Nature Science Foundation of China (No.61004117) and the Zhejiang Provincial Natural Science Foundation of China under Grant No. Y1090718.

Appendix A. Proof of Proposition 5.1

The idea of isogeometric analysis is to use the same mathematical representation for the computational domain and solution field. Suppose that the computational domain Ω is parameterized by the following planar B-spline surface:

$$\sigma(\xi, \eta) = (x(\xi, \eta), y(\xi, \eta)) = \sum_{i=1}^{n_i} \sum_{j=1}^{n_j} N_i^{d_i}(\xi) N_j^{d_j}(\eta) \mathbf{c}_{ij}, \quad (23)$$

In isogeometric analysis, the solution field of heat conduction problem (10) over the computational domain Ω has the following form,

$$U_h(x(\xi, \eta), y(\xi, \eta)) = \mathcal{T}(\xi, \eta) = \sum_{i=1}^{n_i} \sum_{j=1}^{n_j} N_i^{d_i}(\xi) N_j^{d_j}(\eta) T_{ij}, \quad (24)$$

Here T_{ij} are the unknown variables in isogeometric analysis to be solved from the boundary condition and equation (10).

From $U_h(x(\xi, \eta), y(\xi, \eta)) = \mathcal{T}(\xi, \eta)$, we have

$$\frac{\partial \mathcal{T}}{\partial \xi} = \frac{\partial U_h}{\partial x} \frac{\partial x}{\partial \xi} + \frac{\partial U_h}{\partial y} \frac{\partial y}{\partial \xi}, \quad (25)$$

$$\frac{\partial \mathcal{T}}{\partial \eta} = \frac{\partial U_h}{\partial x} \frac{\partial x}{\partial \eta} + \frac{\partial U_h}{\partial y} \frac{\partial y}{\partial \eta}, \quad (26)$$

Then we can obtain

$$\frac{\partial U_h}{\partial x} = \left(\frac{\partial \mathcal{T}}{\partial \xi} y_\eta - \frac{\partial \mathcal{T}}{\partial \eta} y_\xi \right) / J, \quad (27)$$

$$\frac{\partial U_h}{\partial y} = \left(\frac{\partial \mathcal{T}}{\partial \eta} x_\xi - \frac{\partial \mathcal{T}}{\partial \xi} x_\eta \right) / J, \quad (28)$$

where $J = x_\xi y_\eta - x_\eta y_\xi$.

Similarly,

$$\frac{\partial^2 \mathcal{T}}{\partial^2 \xi} = \frac{\partial^2 U_h}{\partial^2 x} \left(\frac{\partial x}{\partial \xi} \right)^2 + \frac{\partial U_h}{\partial x} \frac{\partial^2 x}{\partial^2 \xi} + \frac{\partial^2 U_h}{\partial^2 y} \left(\frac{\partial y}{\partial \xi} \right)^2 + \frac{\partial U_h}{\partial y} \frac{\partial^2 y}{\partial^2 \xi}, \quad (29)$$

$$\frac{\partial^2 \mathcal{T}}{\partial^2 \eta} = \frac{\partial^2 U_h}{\partial^2 x} \left(\frac{\partial x}{\partial \eta} \right)^2 + \frac{\partial U_h}{\partial x} \frac{\partial^2 x}{\partial^2 \eta} + \frac{\partial^2 U_h}{\partial^2 y} \left(\frac{\partial y}{\partial \eta} \right)^2 + \frac{\partial U_h}{\partial y} \frac{\partial^2 y}{\partial^2 \eta}, \quad (30)$$

From (29)(30), we have

$$\frac{\partial^2 U_h}{\partial^2 x} = [y_\eta^2 (\mathcal{T}_{\xi\xi} - \frac{\partial U_h}{\partial x} x_{\xi\xi} - \frac{\partial U_h}{\partial y} y_{\xi\xi}) - y_\xi^2 (\mathcal{T}_{\eta\eta} - \frac{\partial U_h}{\partial x} x_{\eta\eta} - \frac{\partial U_h}{\partial y} y_{\eta\eta})] / K, \quad (31)$$

$$\frac{\partial^2 U_h}{\partial^2 y} = [x_\xi^2 (\mathcal{T}_{\eta\eta} - \frac{\partial U_h}{\partial x} x_{\eta\eta} - \frac{\partial U_h}{\partial y} y_{\eta\eta}) - x_\eta^2 (\mathcal{T}_{\xi\xi} - \frac{\partial U_h}{\partial x} x_{\xi\xi} - \frac{\partial U_h}{\partial y} y_{\xi\xi})] / K, \quad (32)$$

where $K = (x_\xi y_\eta)^2 - (x_\eta y_\xi)^2$.

Hence, from (27)(28) (31)(32), we can obtain

$$\begin{aligned} \Delta U_h &= \frac{\partial^2 U_h}{\partial^2 x} + \frac{\partial^2 U_h}{\partial^2 y} \\ &= [(x_\xi^2 - y_\xi^2) (\mathcal{T}_{\eta\eta} - \frac{\partial U_h}{\partial x} x_{\eta\eta} - \frac{\partial U_h}{\partial y} y_{\eta\eta}) - (x_\eta^2 - y_\eta^2) (\mathcal{T}_{\xi\xi} - \frac{\partial U_h}{\partial x} x_{\xi\xi} - \frac{\partial U_h}{\partial y} y_{\xi\xi})] / K \\ &= (JL_\xi \mathcal{T}_{\eta\eta} - JL_\eta \mathcal{T}_{\xi\xi} + L_{\xi\xi} \mathcal{T}_\eta - L_{\eta\eta} \mathcal{T}_\xi) / (JK). \end{aligned}$$

where

$$L_\xi = x_\xi^2 - y_\xi^2, L_\eta = x_\eta^2 - y_\eta^2,$$

$$L_{\xi\xi} = (L_\eta y_{\xi\xi} - L_\xi y_{\eta\eta}) x_\xi - (L_\eta x_{\xi\xi} - L_\xi x_{\eta\eta}) y_\xi,$$

$$L_{\eta\eta} = (L_\eta y_{\xi\xi} - L_\xi y_{\eta\eta}) x_\eta - (L_\eta x_{\xi\xi} - L_\xi x_{\eta\eta}) y_\eta.$$

This completes the proof. \square

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 vol. 5654(2009), Springer, pp.19-44
- [2] M. Ainsworth and J. Oden. *A posteriori error estimation in finite element analysis*. Wiley- Interscience, New York, 2002.
- [3] F. Auricchio, L.B. da Veiga, A. Buffa, C. Lovadina, A. Reali, and G. Sangalli. A fully locking-free isogeometric approach for plane linear elasticity problems: A stream function formulation. *Computer Methods in Applied Mechanics and Engineering*, 197:160-172, 2007.
- [4] Y. Bazilevs, L. Beirao de Veiga, J.A. Cottrell, T.J.R. Hughes, and G. Sangalli. Isogeometric analysis: approximation, stability and error estimates for refined meshes. *Mathematical Models and Methods in Applied Sciences*, 6:1031-1090, 2006.
- [5] Y. Bazilevs, V.M. Calo, T.J.R. Hughes, and Y. Zhang. Isogeometric fluid structure interaction: Theory, algorithms, and computations. *Computational Mechanics*, 43:3-37, 2008.
- [6] Y. Bazilevs, V.M. Calo, Y. Zhang, and T.J.R. Hughes. Isogeometric fluid structure interaction analysis with applications to arterial blood flow. *Computational Mechanics*, 38:310-322, 2006.
- [7] Y. Bazilevs, V.M. Calo, J.A. Cottrell, J. Evans, T.J.R. Hughes, S. Lipton, M.A. Scott, and T.W. Sederberg. Isogeometric analysis using T-Splines. *Computer Methods in Applied Mechanics and Engineering*, 199(5-8): 229-263, 2010.
- [8] K.H. Brakhage and Ph. Lamby. Application of B-spline techniques to the modeling of airplane wings and numerical grid generation. *Computer Aided Geometric Design*, 25: 738-750, 2008.
- [9] D. Burkhart, B. Hamann and G. Umlauf. Iso-geometric analysis based on Catmull-Clark subdivision solids. *Computer Graphics Forum*, 29(5): 1575-1584, 2010.
- [10] G. Celniker and D. Gossard. Deformable curve and surface finite elements for free-form shape design. *Computer Graphics (Procs of SIGGRAPH'91)* 25(4): 257-266, 1991.
- [11] 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(5-8): 334-356, 2010.
- [12] J.A. Cottrell, T.J.R. Hughes, and A. Reali. Studies of refinement and continuity in isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 196:4160-4183, 2007.
- [13] J.A. Cottrell, A. Reali, Y. Bazilevs, and T.J.R. Hughes. Isogeometric analysis of structural vibrations. *Computer Methods in Applied Mechanics and Engineering*, 195:5257-5296, 2006.
- [14] T. Dokken, V. Skytt, J. Haenisch, K. Bengtsson. Isogeometric representation and analysis—bridging the gap between CAD and analysis. 47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace Exposition, 5-8 January 2009, Orlando, Florida.
- [15] M. Dörfel, B. Jüttler, and B. Simeon. Adaptive isogeometric analysis by local h-refinement with T-splines. *Computer Methods in Applied Mechanics and Engineering*, 199(5-8): 264-275, 2010.

- [16] R. Duvigneau. An introduction to isogeometric analysis with application to thermal conduction. INRIA Research Report RR-6957, June 2009
- [17] T. Elguedj, Y. Bazilevs, V.M. Calo, and T.J.R. Hughes. \bar{B} and \bar{F} projection methods for nearly incompressible linear and non-linear elasticity and plasticity using higher-order NURBS elements. *Computer methods in applied mechanics and engineering*, 197: 2732-2762, 2008.
- [18] J.A. Evans, Y. Bazilevs, I. Babuka, T.J.R. Hughes. n -Widths, supinfs, and optimality ratios for the k-version of the isogeometric finite element method, *Computer Methods in Applied Mechanics and Engineering*, 198: 1726-1741, 2009.
- [19] G. Farin. *Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide*, 5th Edition. Morgan Kaufmann, San Mateo, CA, 2001
- [20] G. Farin and D. Hansford. Discrete coons patches. *Computer Aided Geometric Design*, 16(7):691-700, 1999.
- [21] J. Gain and N. Dodgson. Preventing self-Intersection under free-form deformation. *IEEE Transactions on Visualization and Computer Graphics*, 7(4), pp. 289-298, Oct.-Dec. 2001.
- [22] H. Gomez, V.M. Calo, Y. Bazilevs, and T.J.R. Hughes. Isogeometric analysis of the Cahn-Hilliard phase-field model. *Computer Methods in Applied Mechanics and Engineering*, 197:4333-4352, 2008.
- [23] 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, 39-41, pp 4135-4195, 2005.
- [24] T.J.R. Hughes, A. Realli, and G. Sangalli. Duality and unified analysis of discrete approximations in structural dynamics and wave propagation: Comparison of p-method finite elements with k-method NURBS. *Computer methods in applied mechanics and engineering*, 197: 4104-4124, 2008.
- [25] T.J.R. Hughes, A. Realli, and G. Sangalli. Efficient quadrature for NURBS-based isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 199(5-8): 301-313, 2010.
- [26] H. Kestelman. Mappings with non-vanishing Jacobian. *Amer. Math. Monthly* 78, 662-663, 1971.
- [27] H.J. Kim, Y.D Seo and S.K Youn. Isogeometric analysis for trimmed CAD surfaces. *Computer Methods in Applied Mechanics and Engineering*, 198(37-40): 2982-2995, 2009.
- [28] T. Martin, E. Cohen, and R.M. Kirby. Volumetric parameterization and trivariate B-spline fitting using harmonic functions. *Computer Aided Geometric Design*, 26(6):648-664, 2009.
- [29] D.S. McRae. r -refinement grid adaptation algorithms and issues. *Computer Methods in Applied Mechanics and Engineering*, 189: 1161-1182, 2000.
- [30] K. M. Morken. Some identities for products and degree raising of splines. *Constructive Approximation*, 7: 195-208, 1991.
- [31] T. Nguyen, B. Jüttler. Using approximate implicitization for domain parameterization in isogeometric analysis. *International Conference on Curves and Surfaces*, Avignon, France, 2010.
- [32] G. Xu, B. Mourrain, R. Duvigneau, A. Galligo. Optimal analysis-aware parameterization of computational domain in isogeometric analysis. *Proc. of Geometric Modeling and Processing (GMP 2010)*, 2010, 236-254
- [33] Y. Zhang, Y. Bazilevs, S. Goswami, C. Bajaj, and T.J.R. Hughes. Patient-specific vascular NURBS modeling for isogeometric analysis of blood flow. *Computer Methods in Applied Mechanics and Engineering*, 196:2943-2959, 2007.