Journal of Computational Mathematics
Vol.xx, No.x, 2023, 1–29.

http://www.global-sci.org/jcm
doi:10.4208/jcm.2301-m2022-0116

DEGREE ELEVATION AND KNOT INSERTION FOR GENERALIZED BÉZIER SURFACES AND THEIR APPLICATION TO ISOGEOMETRIC ANALYSIS*

Mengyun Wang
School of Mathematical Sciences, Dalian University of Technology, Dalian 116024, China
Key Laboratory for Computational Mathematics and Data Intelligence of Liaoning Province, Dalian 116024, China
Email: wmyyy@mail.dlut.edu.cn
Ye Ji
School of Mathematical Sciences, Dalian University of Technology, Dalian 116024, China
Key Laboratory for Computational Mathematics and Data Intelligence of Liaoning Province, Dalian 116024, China
Email: jiye@mail.dlut.edu.cn
Chungang Zhu[1)]
School of Mathematical Sciences, Dalian University of Technology, Dalian 116024, China
Key Laboratory for Computational Mathematics and Data Intelligence of Liaoning Province, Dalian 116024, China
Email: cgzhu@dlut.edu.cn

Abstract

Generalized Bézier surfaces are a multi-sided generalization of classical tensor product Bézier surfaces with a simple control structure and inherit most of the appealing properties from Bézier surfaces. However, the original degree elevation changes the geometry of generalized Bézier surfaces such that it is undesirable in many applications, e.g. isogeometric analysis. In this paper, we propose an improved degree elevation algorithm for generalized Bézier surfaces preserving not only geometric consistency but also parametric consistency. Based on the knot insertion of B-splines, a novel knot insertion algorithm for generalized Bézier surfaces is also proposed. Then the proposed algorithms are employed to increase degrees of freedom for multi-sided computational domains parameterized by generalized Bézier surfaces in isogeometric analysis, corresponding to the traditional p-, h-, and k-refinements. Numerical examples demonstrate the effectiveness and superiority of our method.

Mathematics subject classification: 65D07, 65D17, 68U07.
Key words: Generalized Bézier surface, Degree elevation, Knot insertion, Isogeometric analysis, Refinement.

1. Introduction

The representation of surfaces, as one of the core research fields of computer-aided geometric design (CAGD) [7], has undergone tremendous advances in the past decades. Concerning geometric modeling, multiple powerful tools have been developed, such as Bézier surfaces and non-uniform rational B-spline (NURBS) surfaces [8,25]. Despite this advancement, the majority

* Received May 3, 2022 / Revised version received August 25, 2022 / Accepted January 28, 2023 / Published online October 11, 2023 /
1) Corresponding author

of these tools are inconvenient to represent n-sided ($n > 4$) surfaces required in some application scenarios. To overcome this problem, multi-sided surfaces, e.g. S-surfaces [21] and toric surfaces [16], are widely investigated [10, 18, 41, 42].

In 2016, Várady *et al.* [28] proposed a new polygonal parametric surface, named generalized Bézier (GB) surface, which is a generalization of the classical tensor product Bézier surface. The local coordinates of a GB surface are related to the generalized barycentric coordinates of the polygonal parametric domain. GB surfaces inherit the most of nice properties of Bézier surfaces. Based on [28], an enhanced version of GB surfaces was proposed in [29]. Salvi and Várady [30] proposed a new multi-sided surface scheme that permits domains with concave angles. Recently, Vaitkus *et al.* [27] introduce generalized B-spline (GBS) surfaces, which match B-spline surfaces with arbitrary geometric continuity. Compared with toric surfaces, GB surfaces are easier to generate high-quality parameterizations, which facilitates further analysis. In this paper, we focus on the first version of GB surfaces [28].

Although the GB surfaces is a powerful tool in multi-sided surface modeling, the degree elevation algorithm introduced in [28] may change the interior of the original surface. As the degree of a GB surface increases, the weight of the central control point increases, leading to the problem that the isoparametric curves move towards the central control point. The parameterization goes worse as a consequence. For this reason, we propose an improved degree elevation algorithm for GB surfaces that remains surfaces unchanged. Additionally, considering its capability of endowing local control, the locality plays a pivotal role in a surface. Based on the idea of deeming a Bézier surface as a B-spline surface with appropriate knot vectors, a novel knot insertion algorithm for GB surfaces is also proposed.

In 2005, Hughes *et al.* [5, 11] proposed the concept of isogeometric analysis (IGA), which has the potential for bridging the gap between finite element analysis (FEA) and computer-aid design (CAD). Given the boundary representation of a CAD model, constructing a spline-based mapping from its parametric domain to the computational domain is a crucial task in IGA. The quality of parameterization dramatically affects the accuracy and efficiency of the subsequent analysis [3, 26, 37]. To construct high-quality parameterizations, many methods have been proposed. Some methods are suitable for genus-0 domains, such as variational harmonic method [38] and Teichmüller mapping method [23]. For complex domains, especially for high-genus domains, single-patch parameterizations are not sufficient due to the topological flaw of general parametric surfaces. To this end, multi-patch configurations were extensively adopted [1, 36, 40]. Moreover, Lei *et al.* [17] proposed a novel automatic hexahedral mesh generation method, which lays a solid theoretic foundation for structured hex-meshing based on foliations. The above works mainly focus on isotropic parameterization methods that are independent of governing equations. With the same degrees of freedom (DOFs), anisotropic parameterizations customized for the governing equation may yield a more accurate numerical solution [14, 35]. Compared with the planar parameterization construction, the volumetric case is more challenging in both robustness and efficiency [13, 24, 39]. Xie *et al.* [33] handle volumetric modeling using interpolatory Catmull-Clark subdivision approaches. To improve computational efficiency, Xu *et al.* [34] proposed a framework for computation reuse in IGA. Recently, a deep-learning-based isogeometric analysis-reuse approach, called IGA-Reuse-Net, was proposed [32].

Though many methods are proposed, most of them focus on triangular or quadrilateral domains. However, triangular or quadrilateral representation is not suitable for multi-sided computational domains. As pointed out in [3, 18], using quadrilateral surfaces (e.g. Bézier

and B-spline surfaces) to parametrize multi-sided domains may cause mesh degeneration or decreased continuity. As far as we know, only a few papers discuss the polygonal toric surface techniques in the parameterization for IGA [12, 18]. However, this method is sometimes unsatisfactory in terms of the quality of the parameterization. In this paper, GB surfaces are employed to IGA. And it is easier to construct satisfactory parameterizations.

Besides, there are three common methods in IGA to improve the accuracy of numerical solutions by increasing DOFs: knot insertion (h-refinement), degree elevation (p-refinement), and the combination of these two algorithms (k-refinement) [6,11]. By using the proposed degree elevation and knot insertion algorithms, we present the concepts of p-, h-, and k-refinements in IGA based on GB surfaces. To verify the effectiveness and robustness of the proposed methods, Poisson's equations and linear elasticity problems over polygonal computational domains are considered. Comparisons with toric and NURBS methods are also given.

The rest of the paper is organized as follows. Section 2 devotes to reviewing the definition of GB surface and its original degree elevation. In Section 3, based on the properties of basis functions, we present an improved degree elevation and a novel knot insertion for GB surfaces. These proposed geometric algorithms remain the geometry and parameterization of the original surface unchanged. In Section 4, the improved degree elevation and knot insertion are adopted for the refinement of GB parameterizations of computational domains in IGA. Several numerical examples and comparisons with the toric and NURBS methods are in Section 5. Finally, the paper concludes with a summary and future work in Section 6.

2. Preliminaries

In this section, we briefly review the definition of GB surfaces and the original degree elevation algorithm proposed in [28].

2.1. Generalized Bézier surface

A generalized Bézier surface is a rational polygonal surface defined over n-sided convex domain $\mathcal{P}$ (see Fig. 2.1(a) for an example). The definition of GB surface needs local parameters $s_i = s_i(u, v)$ and $h_i = h_i(u, v)$ which are computed by Wachspress barycentric coordinates λ_i [31]. Alternatively, we follow an expression of Wachspress coordinates [9].

Definition 2.1 ([9]). *For a given n-sided convex polygonal domain $\mathcal{P}$ with vertices $\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_n$ ($n \geq 4$), denote its outward unit normal to the edge $e_i = \overline{\mathbf{v}_{i-1}\mathbf{v}_i}$ by $\mathbf{n}_i = (n_1^i, n_2^i)^{\mathrm{T}}$. Let $g_i^{\perp}(\mathbf{x})$ be the perpendicular distance of $\mathbf{x}$ to the edge e_i, i.e.*

$$g_i^{\perp}(\mathbf{x}) = (\mathbf{v}_i - \mathbf{x}) \cdot \mathbf{n}_i, \quad \forall \mathbf{x} \in \mathcal{P}. \tag{2.1}$$

Then Wachspress barycentric coordinates are expressed as

$$\lambda_i(\mathbf{x}) = \frac{w_i(\mathbf{x})}{\sum_{j=1}^{n} w_j(\mathbf{x})}, \quad i = 1, \ldots, n, \tag{2.2}$$

where

$$w_i(\mathbf{x}) = \frac{\mathbf{n}_{i-1} \times \mathbf{n}_i}{g_{i-1}^{\perp}(\mathbf{x}) g_i^{\perp}(\mathbf{x})}, \tag{2.3}$$

$$\mathbf{n}_{i-1} \times \mathbf{n}_i = n_1^{i-1} n_2^i - n_2^{i-1} n_1^i.$$

For the exceeded indexes, we set

$$\mathbf{v}_0 = \mathbf{v}_n, \quad g_0^{\perp}(\mathbf{x}) = g_n^{\perp}(\mathbf{x}), \quad \mathbf{n}_0 = \mathbf{n}_n.$$

From the above formulas (2.1)-(2.3), the gradients of Wachspress coordinates, which play a crucial role in many applications, e.g. Hermite interpolation and IGA, can be derived quite easily. Let

$$\mathbf{R}_i(\mathbf{x}) = \frac{\mathbf{n}_{i-1}}{g_{i-1}^{\perp}(\mathbf{x})} + \frac{\mathbf{n}_i}{g_i^{\perp}(\mathbf{x})},$$

then the gradients of Wachspress barycentric coordinates are computed by

$$\nabla\lambda_i = \lambda_i\left(\mathbf{R}_i - \sum_{j=1}^{n}\lambda_j\mathbf{R}_j\right). \tag{2.4}$$

With Wachspress barycentric coordinates in hand, we recall the following definition of GB surfaces.

Definition 2.2 ([28]). *Given an n-sided convex polygonal domain $\mathcal{P}$ with vertices $\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_n$, denote the Wachspress barycentric coordinates of $\mathcal{P}$ by λ_i, $i = 1, \ldots, n$. Let θ_i be the interior angle of $\mathcal{P}$ at $\mathbf{v}_i$. Given the control points $\mathbf{C}_{j,k}^{d,i}$, $j = 0, \ldots, d$, $k = 0, \ldots, l-1$, where d is the degree of surface, $l = \lceil d/2 \rceil$ is the number of control point layers. The GB surface is the image of the mapping $\mathbf{S}^d : \mathcal{P} \to \mathbb{R}^3$, $\forall (u, v) \in \mathcal{P}$,*

$$\mathbf{S}^d(u,v) = \sum_{i=1}^{n}\sum_{j=0}^{d}\sum_{k=0}^{l-1}\mu_{j,k}^{i}\mathbf{C}_{j,k}^{d,i}B_{j,k}^{d,d}\big(s_i(u,v), h_i(u,v)\big) + \mathbf{C}_0^d B_0^d(u,v), \tag{2.5}$$

where

$$s_i = \frac{\sin(\theta_i)g_{i-1}^{\perp}}{\sin(\theta_i)g_{i-1}^{\perp} + \sin(\theta_{i-1})g_{i+1}^{\perp}}, \quad h_i = 1 - \lambda_{i-1} - \lambda_i \tag{2.6}$$

are the local parameters,

$$B_{j,k}^{d,d}(s_i, h_i) = B_j^d(s_i)B_k^d(h_i) = \binom{d}{j}(1-s_i)^{d-j}s_i^j\binom{d}{k}(1-h_i)^{d-k}h_i^k$$

are Bernstein basis functions of (s_i, h_i),

$$\mathbf{C}_0^d = \frac{1}{n}\sum_{i=1}^{n}\mathbf{C}_{l,l-1}^{d,i} \tag{2.7}$$

is the central point, and its corresponding blending function

$$B_0^d(u,v) = 1 - \sum_{i=1}^{n}\sum_{j=0}^{d}\sum_{k=0}^{l-1}\mu_{j,k}^{i}B_{j,k}^{d,d}\big(s_i(u,v), h_i(u,v)\big), \tag{2.8}$$

and weights

$$\mu_{j,k\,j\le\frac{d}{2}}^{i} = \begin{cases} 0, & j < k, \quad k \ge 2, \\ \dfrac{1}{2}, & 2 \le j = k < l, \\ \dfrac{h_{i-1}}{h_{i-1}+h_i}, & j, k < 2, \\ 1, & \textit{otherwise}, \end{cases}$$

$$\mu^i_{j,k_{j>\frac{d}{2}}} = \begin{cases} 0, & j > d-k, \quad k \geq 2, \\ \dfrac{1}{2}, & j = d-k, \quad k \geq 2, \\ \dfrac{h_{i+1}}{h_{i+1}+h_i}, & j > d-2, \quad k < 2, \\ 1, & \textit{otherwise.} \end{cases}$$

Remark 2.1. Notice that, in Definition 2.2, the control points around the corners are shared by the two adjacent patches. That is to say, these control points must be marked twice, i.e.

$$\mathbf{C}^{d,i}_{p,q} = \mathbf{C}^{d,i-1}_{d-q,p}, \quad p,q = 0,\ldots,l-1.$$

Remark 2.2. The original definition

$$s_i = \frac{\lambda_i}{\lambda_{i-1}+\lambda_i}$$

is undefined for points on e_α, $\alpha \notin \{i-1, i, i+1\}$ where the sum $\lambda_{\alpha-1}+\lambda_\alpha$ in the denominator vanishes. This causes the basis functions and the GB surface to be discontinuous on the boundary. To circumvent this problem, we use the equivalent definition (2.6) proposed in [28]. The resulting basis functions and their gradients are continuous on the whole parametric domain $\mathcal{P}$, which is beneficial to further analysis and computation.

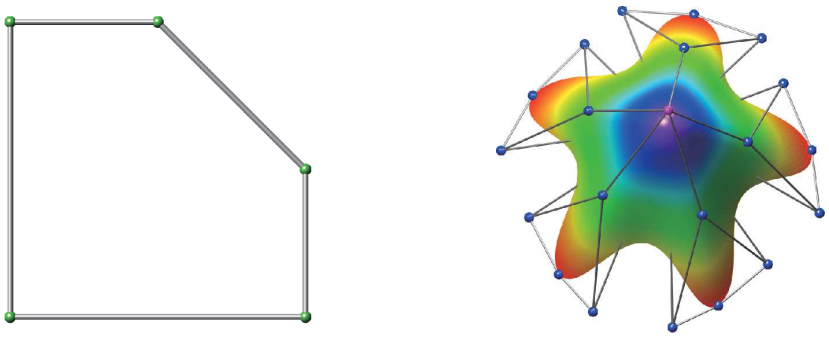
Similar to tensor product Bézier surfaces, GB surfaces have many nice properties, such as corner interpolation and affine invariance. And each boundary of $\mathcal{P}$ is mapped to the boundary of the GB surface, which is a Bézier curve of degree d. More importantly, GB surfaces are able to represent polygonal surfaces without mesh degeneration and/or decrease in continuity. These properties play an important role in surface modeling.

Example 2.1. Consider a pentagonal parametric domain

$$\mathcal{P} = Conv\left\{(0,0),(1,0),\left(1,\frac{1}{2}\right),\left(\frac{1}{2},1\right),(0,1)\right\},$$

as shown in Fig. 2.1(a), where $Conv$ means the convex hull of the vertices. Given $d = 4$, $l = 2, n = 5$ and the sets of control points

$$\begin{aligned}
\left\{\mathbf{C}^{d,1}_{j,k}\right\} &= \big\{(-8,-15,0),(-4,-18,0),(0,0,10),(4,-18,0),(8,-15,0),\\
&\quad (-12,-10,0),(-5,-8,10),(0,0,20),(5,-8,10),(12,-10,0)\big\},\\
\left\{\mathbf{C}^{d,2}_{j,k}\right\} &= \big\{(8,-15,0),(12,-10,0),(0,0,10),(16,-4,0),(14,2,0),\\
&\quad (4,-18,0),(5,-8,10),(0,0,20),(8,0,10),(10,8,0)\big\},\\
\left\{\mathbf{C}^{d,3}_{j,k}\right\} &= \big\{(14,2,0),(10,8,0),(0,0,10),(7,12,0),(0,13,0),\\
&\quad (16,-4,0),(8,0,10),(0,0,20),(0,8,10),(-7,12,0)\big\},\\
\left\{\mathbf{C}^{d,4}_{j,k}\right\} &= \big\{(0,13,0),(-7,12,0),(0,0,10),(-10,8,0),(-14,2,0),\\
&\quad (7,12,0),(0,8,10),(0,0,20),(-8,0,10),(-16,-4,0)\big\},\\
\left\{\mathbf{C}^{d,5}_{j,k}\right\} &= \big\{(-14,2,0),(-16,-4,0),(0,0,10),(-12,-10,0),(-8,-15,0),\\
&\quad (-10,8,0),(-8,0,10),(0,0,20),(-5,-8,10),(-4,-18,0)\big\}.
\end{aligned}$$



(a) Pentagonal parametric domain

(b) Pentagonal GB surface

Fig. 2.1. Pentagonal parametric domain and GB surface.

Specifically, $\mathbf{C}_{l,l-1}^{d,i}=(0,0,20), i=1,\ldots,n$, thus the central point is computed as $\mathbf{C}_0^d=(0,0,20)$. Fig. 2.1(b) shows the corresponding 5-sided GB surface.

More details about GB surfaces are available in [27, 28] and references therein.

2.2. Original degree elevation of GB surface

In this subsection, we briefly review the original degree elevation of GB surfaces in [28].

Assume that we have a GB surface of degree d with l layers, then the new control points for this GB surface of degree $d+1$ are computed as

$$\begin{aligned}\mathbf{C}_{j,k}^{d+1,i} &= \eta_j \upsilon_k \mathbf{C}_{j-1,k-1}^{d,i} + (1-\eta_j)\upsilon_k \mathbf{C}_{j,k-1}^{d,i} \\ &\quad + \eta_j(1-\upsilon_k)\mathbf{C}_{j-1,k}^{d,i} + (1-\eta_j)(1-\upsilon_k)\mathbf{C}_{j,k}^{d,i},\end{aligned} \tag{2.9}$$

where $\eta_j = j/(d+1), \upsilon_k = k/(d+1)$, $j=0,\ldots,d+1$, $k=0,\ldots,\lceil d+1/2\rceil$. And those control points $\mathbf{C}_{j,k}^{d,i}$ are set as $\mathbf{0}$, if $j\notin\{0,\ldots,d\}$ or $k\notin\{0,\ldots,l-1\}$. Besides, the central control point $\mathbf{C}_0^{d+1}$ is computed as

$$\mathbf{C}_0^{d+1} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{C}_{\lceil\frac{d+1}{2}\rceil,\lceil\frac{d+1}{2}\rceil-1}^{d+1,i}, \tag{2.10}$$

as described in (2.7). And its corresponding basis function is

$$B_0^{d+1}(u,v) = 1-\sum_{i=1}^{n}\sum_{j=0}^{d+1}\sum_{k=0}^{\lceil\frac{d+1}{2}\rceil-1}\mu_{j,k}^{i}B_{j,k}^{d+1,d+1}\big(s_i(u,v),h_i(u,v)\big). \tag{2.11}$$

Conclusively, the elevated GB surface of degree $d+1$ is written as

$$\mathbf{S}^{d+1}(u,v) = \sum_{i=1}^{n}\sum_{j=0}^{d+1}\sum_{k=0}^{\lceil\frac{d+1}{2}\rceil-1}\mu_{j,k}^{i}\mathbf{C}_{j,k}^{d+1,i}B_{j,k}^{d+1,d+1}\big(s_i(u,v),h_i(u,v)\big) + \mathbf{C}_0^{d+1}B_0^{d+1}(u,v).$$

Though boundaries and cross-derivatives preserve, one major drawback of this approach is that the geometry and parameterization of the interior of the elevated GB surface change. As the degree d increases, the weight of $\mathbf{C}_0^d$ also increases. As shown in Fig. 3.1(b), the elevated surface is distinct compared with the initial surface in Fig. 3.1(a). This drawback makes the original degree elevation not suitable for some applications, especially for IGA. In the following, therefore, we propose an improved degree elevation and a novel knot insertion of GB surfaces to improve their applicability in IGA.

3. Improved Degree Elevation and Knot Insertion of GB Surface

In this section, we propose an improved degree elevation, which increases the degree of GB surface while keeping the surface consistent. Besides, surfaces with locality are more favorable in practical applications. For example, the sparsity of mass and stiffness matrix will be introduced by using basis functions with the local support property in IGA. Therefore, we also propose a novel knot insertion for GB surfaces to introduce locality in Section 3.2.

3.1. Improved degree elevation

Our improved degree elevation is based on the fact that every Bernstein basis function of degree d can be written as linear combination of Bernstein basis functions of degree $d+1$, i.e.

$$B_j^d = \frac{d+1-j}{d+1} B_j^{d+1} + \frac{j+1}{d+1} B_{j+1}^{d+1}. \tag{3.1}$$

Then, we have

$$\begin{aligned} B_{j,k}^{d,d} = B_j^d B_k^d &= \left(\frac{d+1-j}{d+1} B_j^{d+1} + \frac{j+1}{d+1} B_{j+1}^{d+1}\right)\left(\frac{d+1-k}{d+1} B_k^{d+1} + \frac{k+1}{d+1} B_{k+1}^{d+1}\right) \\ &= \eta_{j+1}\upsilon_{k+1} B_{j+1,k+1}^{d+1,d+1} + (1-\eta_j)\upsilon_{k+1} B_{j,k+1}^{d+1,d+1} \\ &\quad + \eta_{j+1}(1-\upsilon_k) B_{j+1,k}^{d+1,d+1} + (1-\eta_j)(1-\upsilon_k) B_{j,k}^{d+1,d+1}, \end{aligned} \tag{3.2}$$

where $\eta_j = j/(d+1), \upsilon_k = k/(d+1)$, $j = 0, \ldots, d$, $k = 0, \ldots, d$. Besides, we treat each $\mu_{j,k}^i \mathbf{C}_{j,k}^{d,i}$ in form as a new control point $\tilde{\mathbf{C}}_{j,k}^{d,i}$. Then we obtain the following result.

Theorem 3.1. *A GB surface of degree d*

$$\mathbf{S}^d(u,v) = \sum_{i=1}^{n} \sum_{j=0}^{d} \sum_{k=0}^{l-1} \tilde{\mathbf{C}}_{j,k}^{d,i} B_{j,k}^{d,d}\big(s_i(u,v), h_i(u,v)\big) + \mathbf{C}_0^d B_0^d(u,v)$$

can be represented as a GB surface of degree $d+1$ as

$$\mathbf{S}^{d+1}(u,v) = \sum_{i=1}^{n} \sum_{j=0}^{d+1} \sum_{k=0}^{l} \tilde{\mathbf{C}}_{j,k}^{d+1,i} B_{j,k}^{d+1,d+1}\big(s_i(u,v), h_i(u,v)\big) + \mathbf{C}_0^d B_0^d(u,v), \tag{3.3}$$

where $\tilde{\mathbf{C}}_{j,k}^{d,i} = \mu_{j,k}^i \mathbf{C}_{j,k}^{d,i}$, and the control points $\tilde{\mathbf{C}}_{j,k}^{d+1,i}$ satisfy

$$\begin{aligned} \tilde{\mathbf{C}}_{j,k}^{d+1,i} &= \eta_j \upsilon_k \tilde{\mathbf{C}}_{j-1,k-1}^{d,i} + (1-\eta_j)\upsilon_k \tilde{\mathbf{C}}_{j,k-1}^{d,i} \\ &\quad + \eta_j(1-\upsilon_k)\tilde{\mathbf{C}}_{j-1,k}^{d,i} + (1-\eta_j)(1-\upsilon_k)\tilde{\mathbf{C}}_{j,k}^{d,i}, \end{aligned} \tag{3.4}$$

in which $\eta_j = j/(d+1), \upsilon_k = k/(d+1), j = 0, \ldots, d+1, k = 0, \ldots, l$. Besides, for $j \notin \{0, \ldots, d\}$ or $k \notin \{0, \ldots, l-1\}$, we set $\tilde{\mathbf{C}}_{j,k}^{d,i} = \mathbf{0}$.

Proof. From the definition of GB surface in (2.5) and the two-scale relation (3.2), we have

$$\sum_{i=1}^{n} \sum_{j=0}^{d+1} \sum_{k=0}^{l} \tilde{\mathbf{C}}_{j,k}^{d+1,i} B_{j,k}^{d+1,d+1}\big(s_i(u,v), h_i(u,v)\big)$$

$$
\begin{aligned}
&= \sum_{i=1}^{n}\sum_{j=0}^{d+1}\sum_{k=0}^{l}\Big(\eta_j\upsilon_k\tilde{\mathbf{C}}_{j-1,k-1}^{d,i} + (1-\eta_j)\upsilon_k\tilde{\mathbf{C}}_{j,k-1}^{d,i} + \eta_j(1-\upsilon_k)\tilde{\mathbf{C}}_{j-1,k}^{d,i} \\
&\qquad\qquad + (1-\eta_j)(1-\upsilon_k)\tilde{\mathbf{C}}_{j,k}^{d,i}\Big)B_{j,k}^{d+1,d+1} \\
&= \sum_{i=1}^{n}\sum_{j=0}^{d}\sum_{k=0}^{l-1}\eta_{j+1}\upsilon_{k+1}\tilde{\mathbf{C}}_{j,k}^{d,i}B_{j+1,k+1}^{d+1,d+1} + \sum_{i=1}^{n}\sum_{j=0}^{d}\sum_{k=0}^{l-1}(1-\eta_j)\upsilon_{k+1}\tilde{\mathbf{C}}_{j,k}^{d,i}B_{j,k+1}^{d+1,d+1} \\
&\quad + \sum_{i=1}^{n}\sum_{j=0}^{d}\sum_{k=0}^{l-1}\eta_{j+1}(1-\upsilon_k)\tilde{\mathbf{C}}_{j,k}^{d,i}B_{j+1,k}^{d+1,d+1} + \sum_{i=1}^{n}\sum_{j=0}^{d}\sum_{k=0}^{l-1}(1-\eta_j)(1-\upsilon_k)\tilde{\mathbf{C}}_{j,k}^{d,i}B_{j,k}^{d+1,d+1} \\
&= \sum_{i=1}^{n}\sum_{j=0}^{d}\sum_{k=0}^{l-1}\Big(\eta_{j+1}\upsilon_{k+1}B_{j+1,k+1}^{d+1,d+1} + (1-\eta_j)\upsilon_{k+1}B_{j,k+1}^{d+1,d+1} \\
&\qquad\qquad + \eta_{j+1}(1-\upsilon_k)B_{j+1,k}^{d+1,d+1} + (1-\eta_j)(1-\upsilon_k)B_{j,k}^{d+1,d+1}\Big)\tilde{\mathbf{C}}_{j,k}^{d,i} \\
&= \sum_{i=1}^{n}\sum_{j=0}^{d}\sum_{k=0}^{l-1}\tilde{\mathbf{C}}_{j,k}^{d,i}B_{j,k}^{d,d}\big(s_i(u,v),h_i(u,v)\big).
\end{aligned}
$$

This means a GB surface $\mathbf{S}^d$ of degree d is represented by a surface $\mathbf{S}^{d+1}$ of degree $d+1$ as (3.3). □

Theorem 3.1 presents the degree elevation for GB surface from degree d to degree $d+1$ while keeps geometry and parameterization consistent. For brevity and practical application, this degree elevation can be expressed in the form of matrixes.

Let

$$
\mathbf{P}^{d,i} = \Big[\tilde{\mathbf{C}}_{j,k}^{d,i}\Big]_{j=0,k=0}^{d,l-1}, \quad \mathbf{P}^{d+1,i} = \Big[\tilde{\mathbf{C}}_{j,k}^{d+1,i}\Big]_{j=0,k=0}^{d+1,l}
$$

denote the matrices composed of the parametric control points of GB surfaces $\mathbf{S}^d$ and $\mathbf{S}^{d+1}$, respectively. Then the result in Theorem 3.1 is written in the following matrix form:

$$
\mathbf{P}^{d+1,i} = \mathbf{T}_d^{d+1}\mathbf{P}^{d,i}\mathbf{R}_d^{d+1}, \tag{3.5}
$$

where

$$
\mathbf{T}_d^{d+1} = \frac{1}{d+1}\begin{pmatrix}
d+1 & & & & & & \\
1 & d & & & & & \\
 & 2 & d-1 & & & & \\
 & & \ddots & \ddots & & & \\
 & & & d-1 & 2 & & \\
 & & & & d & 1 & \\
 & & & & & d+1 &
\end{pmatrix}_{(d+2)\times(d+1)},
$$

$$
\mathbf{R}_d^{d+1} = \frac{1}{d+1}\begin{pmatrix}
d+1 & 1 & & & & \\
 & d & 2 & & & \\
 & & \ddots & \ddots & & \\
 & & & & l-1 & \\
 & & & & d-l+2 & l
\end{pmatrix}_{l\times(l+1)}.
$$

Successively applying the two-scale relation in (3.2) and (3.5), we obtain multi-degree elevation between $\mathbf{P}^{d+r,i} = [\tilde{\mathbf{C}}_{j,k}^{d+r,i}]_{j=0,k=0}^{d+r,l+r-1}$ of $(d+r)$-degree GB surface $\mathbf{S}^{d+r}$ and $\mathbf{P}^{d,i}$ of the initial GB surface $\mathbf{S}^d$

$$\mathbf{P}^{d+r,i} = \mathbf{T}_d^{d+r}\mathbf{P}^{d,i}\mathbf{R}_d^{d+r}, \tag{3.6}$$

where the matrix operators $\mathbf{T}_d^{d+r}$=$\mathbf{T}_{d+r-1}^{d+r}\mathbf{T}_{d+r-2}^{d+r-1}\cdots\mathbf{T}_d^{d+1}$, $\mathbf{R}_d^{d+r}$=$\mathbf{R}_d^{d+1}\cdots\mathbf{R}_{d+r-2}^{d+r-1}\mathbf{R}_{d+r-1}^{d+r}$, respectively.

3.2. Knot insertion

To introduce the locality of GB surfaces, we propose a knot insertion algorithm for GB surfaces in this section. The basic idea is that we deem a biparametric Bernstein basis function $B_{j,k}^{d,d}$ of degree d as a tensor product B-spline basis $N_{j,k}^{d,d}$ of the same degree with knot vectors

$$\mathbf{U} = \Big\{ \underbrace{0,\cdots,0}_{(d+1)\,times},\ \underbrace{1,\cdots,1}_{(d+1)\,times} \Big\}$$

in the u-direction and

$$\mathbf{V} = \Big\{ \underbrace{0,\cdots,0}_{(d+1)\,times},\ \underbrace{1,\cdots,1}_{(d+1)\,times} \Big\}$$

in the v-direction. Thus, the original GB surface (2.5) is expressed by

$$\mathbf{S}^d(u,v) = \sum_{i=1}^{n}\sum_{j=0}^{d}\sum_{k=0}^{l-1}\widetilde{\mathbf{C}}_{j,k}^{d,i}N_{j,k}^{d,d}\big(s_i(u,v),h_i(u,v)\big) + \mathbf{C}_0^d B_0^d(u,v). \tag{3.7}$$

We denote the spline spaces spanned by these B-spline basis $N_{j,k}^{d,d}$ by $\mathcal{S}_{\mathbf{U},\mathbf{V}}^{d,d}$, i.e., $\mathcal{S}_{\mathbf{U},\mathbf{V}}^{d,d} = \mathrm{span}(N_{j,k}^{d,d})$.

Denote the knot vector in the u-direction after inserting m_u new knots by

$$\bar{\mathbf{U}} = \big\{\bar{\xi}_1 = \xi_1, \bar{\xi}_2, \cdots, \bar{\xi}_{2d+m_u+2} = \xi_{2d+2}\big\} \supset \mathbf{U},$$

and denote the knot vector in the v-direction after inserting m_v new knots by

$$\bar{\mathbf{V}} = \{\bar{\eta}_1 = \eta_1, \bar{\eta}_2, \cdots, \bar{\eta}_{2d+m_v+2} = \eta_{2d+2}\} \supset \mathbf{V},$$

where $\xi_1, \xi_{2d+2}, \eta_1$ and η_{2d+2} are the start knots and end knots of $\mathbf{U}$ and $\mathbf{V}$ respectively, i.e., $\xi_1 = \eta_1 = 0,\ \xi_{2d+2} = \eta_{2d+2} = 1$. In addition, we emphasize that only the interior knots are inserted in our method and the maximum multiplicity of each interior knot value is limited to d. Then the spline space $\mathcal{S}_{\mathbf{U},\mathbf{V}}^{d,d}$ is a subspace of the enriched spline space $\mathcal{S}_{\bar{\mathbf{U}},\bar{\mathbf{V}}}^{d,d}$ since the relations between the knot vectors $\mathbf{U} \subset \bar{\mathbf{U}}$ and $\mathbf{V} \subset \bar{\mathbf{V}}$. Therefore, we have the following two-scale relation [25]:

$$N_{j,k}^{d,d} = \sum_{p=1}^{m_u+d+1}\sum_{q=1}^{m_v+d+1} c_{p,q}\big(N_{j,k}^{d,d}\big)\bar{N}_{p,q}^{d,d}, \tag{3.8}$$

where $c_{p,q}(N_{j,k}^{d,d})$ are non-negative coefficients and the B-spline basis $\bar{N}_{p,q}^{d,d} \in \mathcal{S}_{\bar{\mathbf{U}},\bar{\mathbf{V}}}^{d,d}$.

Finally, the following knot insertion for GB surfaces is derived.

Theorem 3.2. *Assume that* $\{\underbrace{0,\cdots,0}_{(d+1)\,times},\ \underbrace{1,\cdots,1}_{(d+1)\,times}\}\subset\bar{\mathbf{U}}$ *and* $\{\underbrace{0,\cdots,0}_{(d+1)\,times},\ \underbrace{1,\cdots,1}_{(d+1)\,times}\}\subset\bar{\mathbf{V}}$*, then the GB surface in* (3.7)

$$\mathbf{S}^d(u,v)=\sum_{i=1}^{n}\sum_{j=0}^{d}\sum_{k=0}^{l-1}\widetilde{\mathbf{C}}_{j,k}^{d,i}N_{j,k}^{d,d}\big(s_i(u,v),h_i(u,v)\big)+\mathbf{C}_0^dB_0^d(u,v) \tag{3.9}$$

can be represented as the following GB surface:

$$\bar{\mathbf{S}}^d(u,v)=\sum_{i=1}^{n}\sum_{j=0}^{m_u+d}\sum_{k=0}^{m_v+l-1}\bar{\mathbf{C}}_{j,k}^{d,i}\bar{N}_{j,k}^{d,d}\big(s_i(u,v),h_i(u,v)\big)+\mathbf{C}_0^dB_0^d(u,v), \tag{3.10}$$

where $\bar{N}_{j,k}^{d,d}$ *are B-spline basis functions of spline space* $\mathcal{S}_{\bar{\mathbf{U}},\bar{\mathbf{V}}}^{d,d}$*, and the collection of control points*

$$\mathbf{P}^{d,i}=\Big[\widetilde{\mathbf{C}}_{j,k}^{d,i}\Big]_{j=0,k=0}^{d,l-1},\quad \bar{\mathbf{P}}^{d,i}=\Big[\bar{\mathbf{C}}_{j,k}^{d,i}\Big]_{j=0,k=0}^{m_u+d,m_v+l-1}$$

satisfies

$$\bar{\mathbf{P}}^{d,i}=\mathbf{T}_{\xi}^{d}\mathbf{P}^{d,i}\mathbf{T}_{\eta}^{d}, \tag{3.11}$$

where the (α,β)*-th component of* $\mathbf{T}_{\xi}^{d}$ *are recursively computed by*

$$\mathbf{T}_{\xi,\alpha\beta}^{r+1}=\frac{\bar{\xi}_{\alpha+r}-\xi_{\beta}}{\xi_{\beta+r}-\xi_{\beta}}\mathbf{T}_{\xi,\alpha\beta}^{r}+\frac{\xi_{\beta+r+1}-\bar{\xi}_{\alpha+r}}{\xi_{\beta+r+1}-\xi_{\beta+1}}\mathbf{T}_{\xi,\alpha,\beta+1}^{r} \tag{3.12}$$

for $r=0,1,\ldots,d-1$*,* $\alpha=1,\ldots,d+m_u+1$*,* $\beta=1,\ldots,d+1$*, starting with*

$$\mathbf{T}_{\xi,\alpha\beta}^{0}=\begin{cases}1, & \bar{\xi}_{\alpha}\in[\xi_{\beta},\xi_{\beta+1}),\\ 0, & \textit{otherwise},\end{cases}$$

and the $(\alpha,\ \beta)$*-th component of* $\mathbf{T}_{\eta}^{d}$ *are recursively computed by*

$$\mathbf{T}_{\eta,\alpha\beta}^{s+1}=\frac{\bar{\eta}_{\alpha+s}-\eta_{\beta}}{\eta_{\beta+s}-\eta_{\beta}}\mathbf{T}_{\eta,\alpha\beta}^{s}+\frac{\eta_{\beta+s+1}-\bar{\eta}_{\alpha+s}}{\eta_{\beta+s+1}-\eta_{\beta+1}}\mathbf{T}_{\eta,\alpha,\beta+1}^{s} \tag{3.13}$$

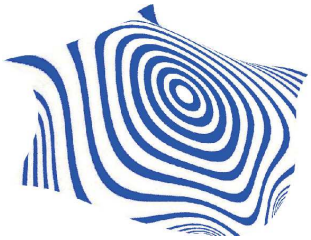
for $s=0,1,\ldots,d-1$*,* $\alpha=1,\ldots,l$*,* $\beta=1,\ldots,l+m_v$*, starting with*

$$\mathbf{T}_{\eta,\alpha\beta}^{0}=\begin{cases}1, & \bar{\eta}_{\alpha}\in[\eta_{\beta},\eta_{\beta+1}),\\ 0, & \textit{otherwise}.\end{cases}$$

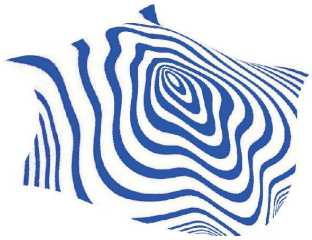
By Theorem 3.2, a GB surface is represented by a linear combination of B-spline basis and $B_0^d(u,v)$ after knot insertion. Then the locality of GB surface holds by the locality of B-spline basis functions.

3.3. Comparisons with original degree elevation

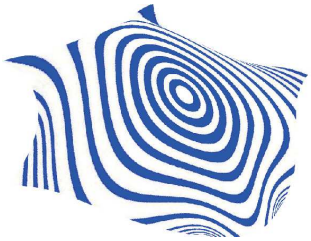
To compare the proposed algorithms with the original degree elevation in [28] and illustrate their difference clearly, we add texture mappings to GB surfaces after degree elevation and knot insertion, as shown in Figs. 3.1-3.2. The initial hexagonal and heptagonal GB surfaces of degree 4 are shown in Figs. 3.1(a) and 3.2(a). After elevating the degrees of initial surfaces to 24 by the original degree elevation, as shown in Figs. 3.1(b) and 3.2(b), one can observe that



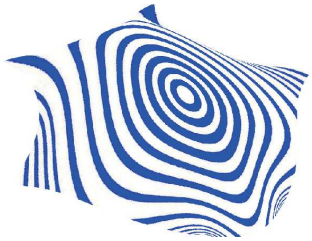
(a) Initial surface



(b) Original degree elevation [28]

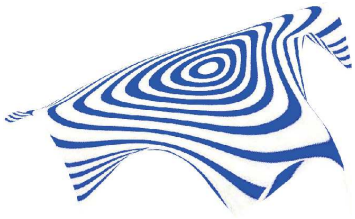


(c) Improved degree elevation (ours)



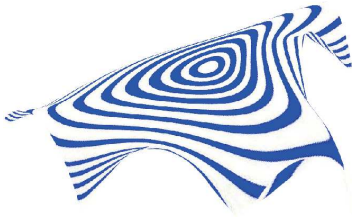
(d) Knot insertion (ours)

Fig. 3.1. Comparison with original degree elevation for hexagonal GB surface.

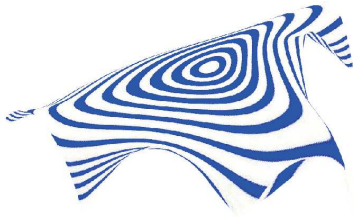


(a) Initial surface

(b) Original degree elevation [28]



(c) Improved degree elevation (ours)



(d) Knot insertion (ours)

Fig. 3.2. Comparison with original degree elevation for heptagonal GB surface.

the elevated surfaces have a distinct deformation. Meanwhile, the elevated surfaces obtained by the proposed degree elevation in Theorem 3.1 still keep consistent with the initial surfaces as shown in Figs. 3.1(c) and 3.2(c). Besides, the refined surfaces obtained by the proposed knot insertion in Theorem 3.2 share the same property of keeping GB surfaces consistently.

Remark 3.1. As pointed out in Section 2.2, the central control point $\mathbf{C}_0^d$ and its corresponding basis function need to be re-computed after the original degree elevation. In the proposed methods, however, we just keep $\mathbf{C}_0^d$ and its corresponding basis function the same as the original GB surface of degree d. This treatment is the key point to keeping the elevated/refined surface unchanged and avoiding redundant computation.

4. Applications to IGA

When NURBS basis functions are used for IGA, there are three main methods for increasing the DOFs: h-refinement by knot insertion, p-refinement by degree elevation, and k-refinement by the combination of the above. In this section, as an illustration, a brief review of Poisson's equation and linear elasticity problem is presented in Section 4.1. Then the proposed degree elevation is applied to the p-refinement method of parameterization in IGA using GB surfaces. And knot insertion of GB surfaces is applied to the h-refinement method. Finally, we combine these two methods to propose the k-refinement method. Note that the linearly independent of blending functions is of great importance for IGA. Otherwise, the stiffness matrix becomes singular. Li *et al.* firstly recognized this and identified a class of T-splines whose blending functions are guaranteed to be linearly independent [20], and then proposed analysis-suitable T-splines [19]. Actually, the linear independence of the blending functions of initial/refined GB surface can be obtained easily by their definitions and linear independence of Bernstein basis functions, and we will omit the proof in detail here.

4.1. Model problems

4.1.1. Poisson's equation

The Poisson's equation with homogeneous Dirichlet and Neumann boundary conditions over polygonal computational domain Ω is defined as

$$-\nabla^2 T(\mathbf{x}) = s(\mathbf{x}), \quad \mathbf{x} \in \Omega, \tag{4.1a}$$

$$T(\mathbf{x}) = 0, \qquad \mathbf{x} \in \partial\Omega_D, \tag{4.1b}$$

$$\frac{\partial T}{\partial \mathbf{n}} = \Phi(\mathbf{x}), \qquad \mathbf{x} \in \partial\Omega_N, \tag{4.1c}$$

where $T(\mathbf{x})$ is the unknown temperature field, $s(\mathbf{x})$ is a known source term, $\partial\Omega = \partial\Omega_D \bigcup \partial\Omega_N$, and $\mathbf{n}$ is the outward unit normal vector to the boundary $\partial\Omega_N$. To verify the effectiveness of the proposed methods, computational domain Ω is parameterized by GB surface.

Multiplying the Eq. (4.1a) by an arbitrary test function $\psi \in H^1_{\partial\Omega}(\Omega)$, we have

$$-\int_\Omega \nabla^2 T(\mathbf{x})\psi(\mathbf{x})d\Omega = \int_\Omega s(\mathbf{x})\psi(\mathbf{x})d\Omega. \tag{4.2}$$

According to the Dirichlet and Neumann boundary conditions (4.1b), (4.1c) and integrating (4.2) by parts, the resulting weak formulation of problem (4.1) is now: Seek for $T \in H^1_D(\Omega)$ such that for all $\psi \in H^1_{\partial\Omega}(\Omega)$

$$\int_\Omega \nabla T(\mathbf{x})\nabla\psi(\mathbf{x})d\Omega - \int_{\partial\Omega_N} \Phi(\mathbf{x})\psi(\mathbf{x})d\Gamma = \int_\Omega s(\mathbf{x})\psi(\mathbf{x})d\Omega. \tag{4.3}$$

According to the paradigm of IGA, the temperature field is approximated by the linear combination of the same basis functions used for geometry, which are also adopted as test functions. Denote the parameterization as $\sigma : \mathcal{P} \to \Omega$, i.e. the geometry mapping from the parametric domain $\mathcal{P}$ to the computational domain Ω. Then test functions are

$$\psi(\mathbf{x}) = \hat{B}_p(\mathbf{x}) = B_p \circ \sigma^{-1}(\mathbf{x}) = B_p(\mathbf{u}), \tag{4.4}$$

where $\mathbf{u} = (u, v)^{\mathrm{T}}, B_p(\mathbf{u})$ are the rearranged basis functions of GB surface through certain numbering rule. Thus, the temperature field is represented by

$$T(\mathbf{x}) = \sum_{p=1}^{n(d+1)l+1} T_p^h B_p(\mathbf{u}) = \sum_{p=1}^{n(d+1)l+1} T_p^h \hat{B}_p(\mathbf{x}). \tag{4.5}$$

Substituting (4.5) and (4.4) into Eq. (4.3), we obtain the corresponding linear system

$$\mathbf{K}\mathbf{T}^{\mathbf{h}} = \mathbf{f}, \tag{4.6}$$

where

$$\mathbf{K}_{p,q} = \int_{\Omega} \nabla \hat{B}_p(\mathbf{x}) \nabla \hat{B}_q(\mathbf{x}) d\Omega \tag{4.7}$$

are the components of stiffness matrix $\mathbf{K}$, and

$$\mathbf{f}_p = \int_{\Omega} s(\mathbf{x}) \hat{B}_p(\mathbf{x}) d\Omega + \int_{\partial\Omega_N} \Phi(\mathbf{x}) \hat{B}_p(\mathbf{x}) d\Gamma_{\Omega} \tag{4.8}$$

are the components of right-hand side $\mathbf{f}$. Eventually, the numerical solution $\mathbf{T}^{\mathbf{h}}$ is obtained by solving the linear system (4.6).

4.1.2. Linear elasticity problem

To demonstrate the robustness of the proposed scheme, the linear elasticity problem is also considered, which is a more complicated vector filed problem. The equations of the two-dimensional linear elasticity problem are given as

$$\begin{cases} \mathbf{L}^{\mathrm{T}} \boldsymbol{\sigma}(\mathbf{x}) = \mathbf{s}(\mathbf{x}), \\ \boldsymbol{\sigma}(\mathbf{x}) = \mathbf{D}\boldsymbol{\varepsilon}(\mathbf{x}) \quad \text{in } \Omega, \\ \boldsymbol{\varepsilon}(\mathbf{x}) = \mathbf{L}\mathbf{v}(\mathbf{x}), \end{cases} \tag{4.9}$$

where $\mathbf{x}=(x, y)^{\mathrm{T}} \in \Omega$, $\mathbf{v}(\mathbf{x})=(v_x(\mathbf{x}), v_y(\mathbf{x}))^{\mathrm{T}}$ are the displacements, $\boldsymbol{\varepsilon}(\mathbf{x})=(\varepsilon_x(\mathbf{x}), \varepsilon_y(\mathbf{x}), \gamma_{xy}(\mathbf{x}))^{\mathrm{T}}$ are the stains, $\boldsymbol{\sigma}(\mathbf{x}) = (\sigma_x(\mathbf{x}), \sigma_y(\mathbf{x}), \tau_{xy}(\mathbf{x}))^{\mathrm{T}}$ are the stresses, $\mathbf{s}(\mathbf{x}) = (s_x(\mathbf{x}), s_y(\mathbf{x}))^{\mathrm{T}}$ are the given body loads,

$$\mathbf{L} = \begin{pmatrix} \dfrac{\partial}{\partial x} & 0 & \dfrac{\partial}{\partial y} \\ 0 & \dfrac{\partial}{\partial y} & \dfrac{\partial}{\partial x} \end{pmatrix}^{\mathrm{T}}$$

is the linear differential operator,

$$\mathbf{D} = \frac{E}{1-\nu^2} \begin{pmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \dfrac{1-\nu}{2} \end{pmatrix}$$

is the elastic matrix, and E and ν are Young's modulus and Poisson's ratio, respectively.

For brevity, we consider Eqs. (4.9) with homogeneous Dirichlet boundary conditions in the following, i.e., $\mathbf{v} = \mathbf{0}$ along the boundary $\partial\Omega$. In this case, the weak formulation of (4.9) is: Find a $\mathbf{v} \in H^1(\Omega)$ such that for all $\mathbf{w} \in H^1_{\partial\Omega}(\Omega)$

$$\int_\Omega \boldsymbol{\varepsilon}(\mathbf{v})^{\mathrm{T}} \mathbf{D} \boldsymbol{\varepsilon}(\mathbf{w}) d\Omega = \int_\Omega \mathbf{s}\mathbf{w} d\Omega. \tag{4.10}$$

The components of the stiffness matrix and force vector are

$$\mathbf{K}_{p,q} = \int_\Omega \mathbf{R}_p^{\mathrm{T}} \mathbf{D} \mathbf{R}_q d\Omega, \tag{4.11}$$

$$\mathbf{f}_p = \int_\Omega \mathbf{s}(\mathbf{x}) \hat{B}_p(\mathbf{x}) d\Omega, \tag{4.12}$$

respectively, where

$$\mathbf{R}_p = \begin{pmatrix} \dfrac{\partial \hat{B}_p(\mathbf{x})}{\partial x} & 0 & \dfrac{\partial \hat{B}_p(\mathbf{x})}{\partial y} \\ 0 & \dfrac{\partial \hat{B}_p(\mathbf{x})}{\partial y} & \dfrac{\partial \hat{B}_p(\mathbf{x})}{\partial x} \end{pmatrix}^{\mathrm{T}}.$$

4.2. *p*-refinement method for generalized Bézier surfaces

Above all, we have the following theorem to reveal that the basis functions of GB surface can be used as test functions in IGA.

Theorem 4.1. *Given a GB surface defined on an convex polygonal domain $\mathcal{P}$. The basis functions of the surface are in $H^1(\mathcal{P})$.*

Proof. As mentioned in Remark 2.2, the basis functions and their gradients of the GB surface are continuous on a whole domain $\mathcal{P}$ which is a bounded closed domain. So they are integrable on $\mathcal{P}$. According to Lebesgue integrability criteria, it is easy to obtain that the basis functions and their gradients are square-integrable on $\mathcal{P}$. This means the basis functions of the GB surface are in $H^1(\mathcal{P})$. □

In the following, we denote the degree of the initial GB surface by d_0 and the degree after p-refinement by d, respectively. During the refinement procedure by the proposed p-refinement, the values and their gradients of basis functions need to be evaluated. Since $\nabla g_i^\perp = -\mathbf{n}_i$ and the gradient of Wachspress coordinates in (2.4), then we have

$$\begin{aligned} \nabla_{\mathbf{u}} s_i &= \frac{\sin(\theta_i)\sin(\theta_{i-1})\big(g_{i-1}^\perp \mathbf{n}_{i+1} - g_{i+1}^\perp \mathbf{n}_{i-1}\big)}{\big(\sin(\theta_i) g_{i-1}^\perp + \sin(\theta_{i-1}) g_{i+1}^\perp\big)^2}, \\ \nabla_{\mathbf{u}} h_i &= -\nabla_{\mathbf{u}} \lambda_{i-1} - \nabla_{\mathbf{u}} \lambda_i. \end{aligned} \tag{4.13}$$

Using the above formulas and chain rule, we obtain the gradient of $B_{j,k}^{d,d}(s_i(\mathbf{u}), h_i(\mathbf{u}))$ with respect to $\mathbf{u}$

$$\nabla_{\mathbf{u}} B_{j,k}^{d,d} = d\left(B_{j-1,k}^{d-1,d} - B_{j,k}^{d-1,d}\right) \nabla_{\mathbf{u}} s_i + d\Big(B_{j,k-1}^{d,d-1} - B_{j,k}^{d,d-1}\Big) \nabla_{\mathbf{u}} h_i.$$

Since the central control point $\mathbf{C}_0^{d_0}$ and its basis function $B_0^{d_0}$ remain unchanged during p-refinement procedure, the gradient of $B_0^{d_0}$ is evaluated as follows:

$$\nabla_{\mathbf{u}} B_0^{d_0} = -\sum_{i=1}^{n} \sum_{j=0}^{d_0} \sum_{k=0}^{\lceil \frac{d_0}{2} \rceil} \left(\nabla_{\mathbf{u}} \mu_{j,k}^i B_{j,k}^{d_0,d_0} + \mu_{j,k}^i \nabla_{\mathbf{u}} B_{j,k}^{d_0,d_0}\right), \tag{4.14}$$

where

$$\nabla_{\mathbf{u}}\mu_{j,k}^{i}=\begin{cases}\dfrac{h_i\nabla_{\mathbf{u}}h_{i-1}-h_{i-1}\nabla_{\mathbf{u}}h_i}{(h_{i-1}+h_i)^2}, & k<2,\quad j<2,\\ \dfrac{h_i\nabla_{\mathbf{u}}h_{i+1}-h_{i+1}\nabla_{\mathbf{u}}h_i}{(h_{i+1}+h_i)^2}, & k<2,\quad j>d_0-2,\\ 0, & \text{otherwise.}\end{cases}$$

4.3. h-refinement method for generalized Bézier surfaces

Due to the locality of GB surfaces after h-refinement, its treatment in IGA is different from p-refinement. Now, if the numerical integration is performed over the parametric domain $\mathcal{P}$, there will be many invalid integration points, which may increase error. Therefore, during h-refinement, we perform the numerical integration over the domain $\tilde{\mathcal{P}}=[0,1]^2$ consists of $\boldsymbol{s}_i=(s_i,h_i)^{\mathrm{T}}$.

As mentioned in [30], the mapping (2.6) between $\mathcal{P}$ and $\tilde{\mathcal{P}}$ is bijective. Hence, we denote its inverse mapping by

$$\mathbf{u}=\tau(\boldsymbol{s}_i). \tag{4.15}$$

Then the test function is rewritten as

$$A_p(\boldsymbol{s}_i)=B_p\big(\tau(\boldsymbol{s}_i)\big)=B_p(\mathbf{u})=\hat{B}_p(\mathbf{x}),$$

where $B_p(\mathbf{u})$ is the rearranged basis function of GB surface.

Similarly, the gradient of basis functions $N_{j,k}^{d,d}(s_q,h_q)$ with respect to $\boldsymbol{s}_i$ is calculated by chain rule, i.e.,

$$\begin{aligned}\nabla_{\boldsymbol{s}_i}N_{j,k}^{d,d}(s_q,h_q)&=(\nabla_{\boldsymbol{s}_i}u,\nabla_{\boldsymbol{s}_i}v)(\nabla_{\mathbf{u}}s_q,\nabla_{\mathbf{u}}h_q)\nabla_{\boldsymbol{s}_q}N_{j,k}^{d,d}(s_q,h_q)\\&=(\nabla_{\mathbf{u}}s_i,\nabla_{\mathbf{u}}h_i)^{-1}(\nabla_{\mathbf{u}}s_q,\nabla_{\mathbf{u}}h_q)\nabla_{\boldsymbol{s}_q}N_{j,k}^{d,d}(s_q,h_q),\end{aligned} \tag{4.16}$$

where $\nabla_{\boldsymbol{s}_q}N_{j,k}^{d,d}(s_q,h_q)$ is the gradient of B-spline basis function. To the central basis function $B_0^{d_0}$, its gradient is calculated as

$$\nabla_{\boldsymbol{s}_i}B_0^{d_0}=(\nabla_{\boldsymbol{s}_i}u,\nabla_{\boldsymbol{s}_i}v)\nabla_{\mathbf{u}}B_0^{d_0}=(\nabla_{\mathbf{u}}s_i,\nabla_{\mathbf{u}}h_i)^{-1}\nabla_{\mathbf{u}}B_0^{d_0}. \tag{4.17}$$

4.4. k-refinement method for generalized Bézier surfaces

By combining the improved degree elevation algorithm with the knot insertion algorithm, k-refinement is proposed as follows. First, the degree of GB surface is elevated by Theorem 3.1, and then Theorem 3.2 is used to insert knots. The calculation of gradients of basis functions in k-refinement is similar to that in h-refinement.

Compared with h-refinement, the k-refinement method has higher continuity in the same DOFs. However, we note that unlike p- and h-refinement, the sequence of solution spaces in k-refinement is not nested, i.e., each refinement should start with the initial surface [5].

5. Numerical Examples

To demonstrate the effectiveness and superiority of the proposed methods in refinement for IGA, several numerical examples are given in this section. A comparison with the original degree elevation in [28] and with other parameterization methods, including NURBS and toric surface methods, is also available.

5.1. Implementation details

We consider three types of parameterizations in the experiments, including GB, NURBS, and toric surfaces. For a fair comparison, all the parameterizations are constructed by the method in [15]. The NURBS and toric parameterizations are constructed by adjusting inner control points and weights, but the GB parameterizations are constructed by adjusting inner control points only.

For refinement, h-refinement and p-refinement [18] are adopted in NURBS and toric methods, respectively. For the h-refinement in NURBS and GB methods, we insert a new knot between every two distinct knots in two directions at a time. For each k-refinement, the degree of the GB surfaces is elevated $2r$ times in each direction first and then perform h-refinement r times.

5.2. Quality measures

As we know, the quality of parameterization seriously affects the accuracy of subsequent analysis. To characterize the quality of parameterizations, we use the following two quality measures.

The scaled Jacobian

$$J_s = \frac{\det \mathbf{J}}{\|\sigma_u\|_2 \|\sigma_v\|_2} \tag{5.1}$$

is used to describe the orthogonality of the parameterization σ, where $\mathbf{J}$ is the Jacobian matrix of σ and $\|\cdot\|_2$ is L_2 norm. It is clear that $J_s \in [-1, 1]$, and J_s is closer to 1, the better orthogonality the parameterization has.

The absolute relative area

$$S_a = \left| \frac{\det \mathbf{J} - S}{S} \right| \tag{5.2}$$

reflects the uniformity of the parameterization, where $S = Area(\Omega)/Area(\mathcal{P})$ is the ratio of the area of the computational domain Ω to the area of parametric domain $\mathcal{P}$. The smaller value of S_a means the uniformity of parameterization is better.

5.3. Poisson's equation

For Poisson's equation, the error is measured by

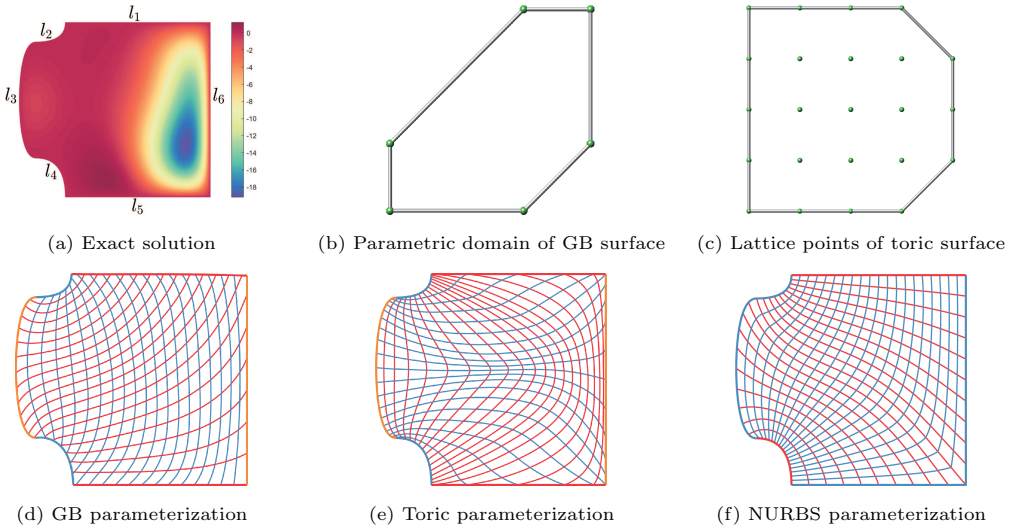
$$e = \sqrt{\frac{\int_\Omega (T - T^h)^{\mathrm{T}} (T - T^h) d\Omega}{\int_\Omega T^{\mathrm{T}} T d\Omega}}, \tag{5.3}$$

where $(\cdot)^{\mathrm{T}}$ means the transpose of a matrix, T and T^h are the exact solution and the numerical solution respectively.

Example 5.1 (Cup). First, we consider Poisson's equation with homogeneous Dirichlet boundary conditions (4.1) over a hexagonal computational domain (cup model), and the corresponding exact solution is shown in Fig. 5.1(a). Let us denote the boundary representation of the computational domain by l_i, $i = 1, \ldots, 6$, as shown in Fig. 5.1(a). To construct a benchmark test satisfying the boundary condition as (4.1b), we set the exact solution as $T = \prod_{i=1}^{6} l_i$. Accordingly, we obtain the source term $s(\mathbf{x}) = -\nabla^2 T(\mathbf{x})$ as (4.1a).

Figs. 5.1(b) and 5.1(c) show the parametric domain of GB parameterization, and the lattice points and parametric domain of toric parameterization, respectively. By using the parameterization technique in [15], various resulting parameterizations are shown in the second row of Fig. 5.1. One can observe a distinctly decreased continuity (C^0 continuity) around the lower right corner in Fig. 5.1(f), while good smoothness in the interior of the computational domain is obtained by the GB and toric parameterizations. Compared with toric parameterization, the smoothness and uniformity of the GB parameterization are much better.

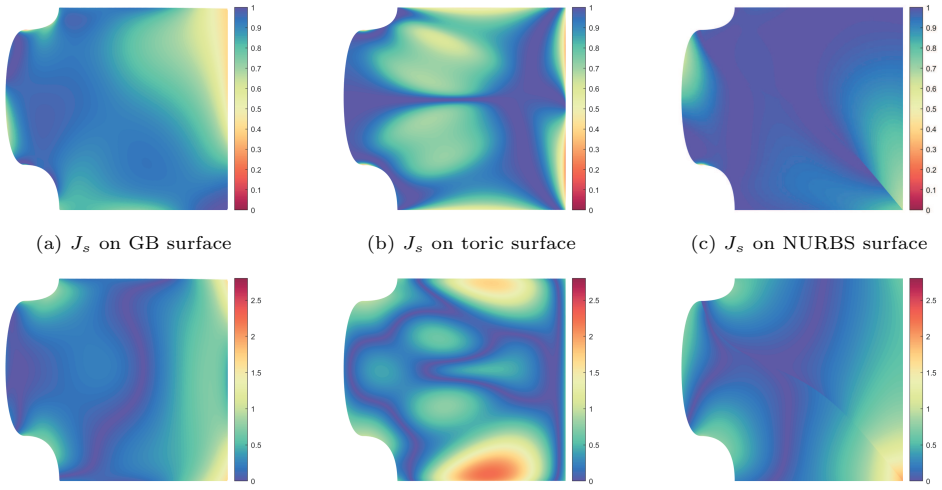
To quantitate the quality of parameterizations, the scaled Jacobian and absolute relative area of three different parameterizations are illustrated in Fig. 5.2. Compared with toric parameterization, the scaled Jacobian of GB parameterization is larger and the absolute relative area is smaller, that is, the orthogonality and uniformity of GB parameterization are better. Hence,



(a) Exact solution (b) Parametric domain of GB surface (c) Lattice points of toric surface

(d) GB parameterization (e) Toric parameterization (f) NURBS parameterization

Fig. 5.1. Hexagonal computational domain (cup).



(a) J_s on GB surface (b) J_s on toric surface (c) J_s on NURBS surface

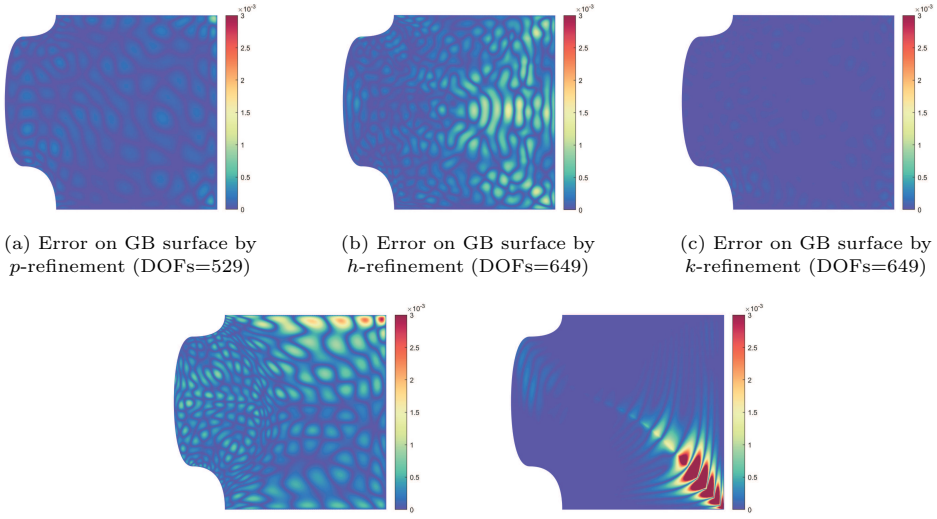
(d) S_a on GB surface (e) S_a on toric surface (f) S_a on NURBS surface

Fig. 5.2. Scaled Jacobian J_s and absolute relative area S_a for Example 5.1.

GB parameterization is able to maintain high quality while keeping high continuity within the domain.

The error colormaps of various methods are shown in Fig. 5.3. To facilitate comparison, the errors are plotted with the same colorbar. One can observe that the GB surface behaves better than the other two methods in terms of error. This is because the GB surface has good smoothness and high-quality parameterization. Due to the decreased continuity around the lower right corner, as shown in Fig. 5.3(e), NURBS is the worst performer. Besides, compared with h-refinement, p- and k-refinements are more effective, this is because they have higher continuity.

The error history during refinement is shown in Fig. 5.4. An oscillation behavior can be seen in the error curve of the original degree elevation. This is because the parameterization changes after the original degree elevation. Our approach effectively avoids the oscillation problem and the errors monotonically decrease.



(a) Error on GB surface by p-refinement (DOFs=529)

(b) Error on GB surface by h-refinement (DOFs=649)

(c) Error on GB surface by k-refinement (DOFs=649)

(d) Error on toric surface (DOFs=583)

(e) Error on NURBS surface (DOFs=780)

Fig. 5.3. Absolute errors for Example 5.1.

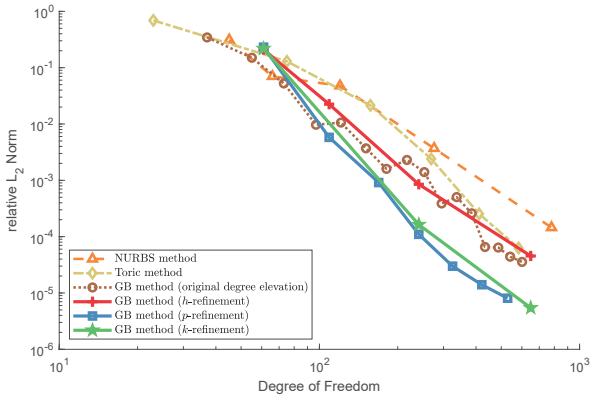
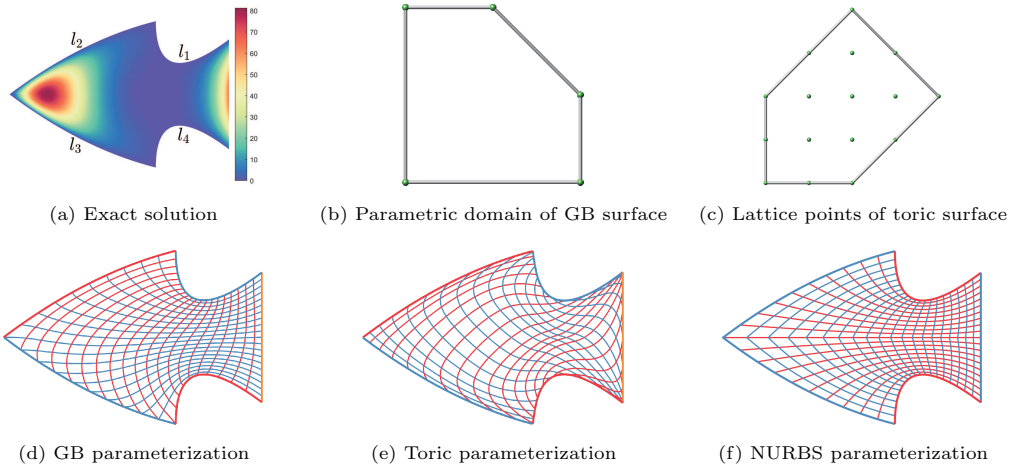


Fig. 5.4. Example 5.1: Error history of different methods.

Example 5.2 (Fish). In this example, we solve Poisson's equation in the pentagonal computational domain (fish model) with the exact solution $T = \prod_{i=1}^{4} l_i$, as shown in Fig. 5.5(a). The corresponding Neumann boundary condition along the right boundary is computed by $\Phi(\mathbf{x}) = \partial T/\partial \mathbf{n}$ as (4.1c). Neumann boundary condition is applied to the right edge (fishtail) and homogeneous Dirichlet boundary conditions are applied to the other edges. The pentagonal parametric domain of GB parameterization, and the lattice points and parametric domain of toric parameterization are provided in Figs. 5.5(b) and 5.5(c). The second row of Fig. 5.5 depicts the resulting parameterizations through the parameterization technique in [15]. As shown in Fig. 5.5(e), a toric parameterization with high distortion is obtained. By contrast, GB and NURBS parameterization perform better in terms of distortion. GB parameterization is smooth inside globally, while continuity degradations along the symmetric line occur in NURBS parameterization as shown in Fig. 5.5(f).

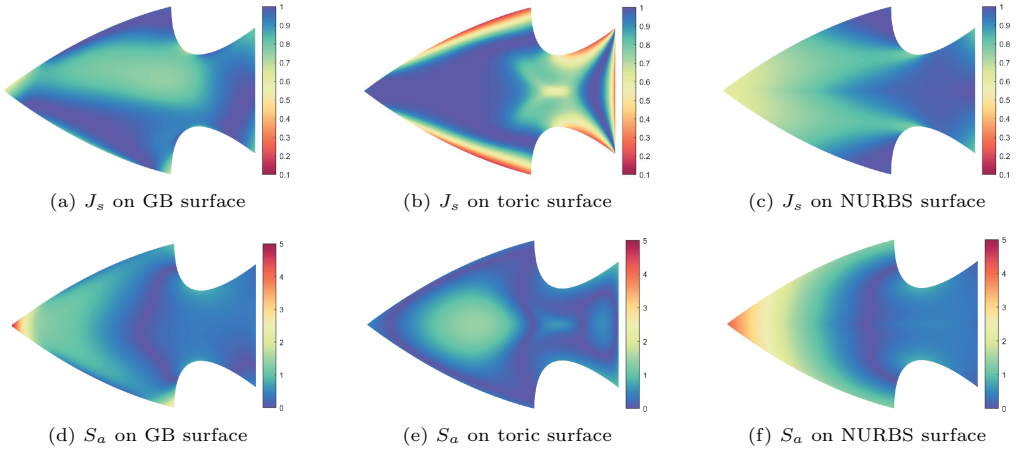
Fig. 5.6 shows the scaled Jacobian and absolute relative area of three different parameterizations. It is manifest that NURBS parameterization is partitioned along the axis of symmetry of the domain due to its tensor product structure.



(a) Exact solution (b) Parametric domain of GB surface (c) Lattice points of toric surface

(d) GB parameterization (e) Toric parameterization (f) NURBS parameterization

Fig. 5.5. Pentagonal computational domain (fish).



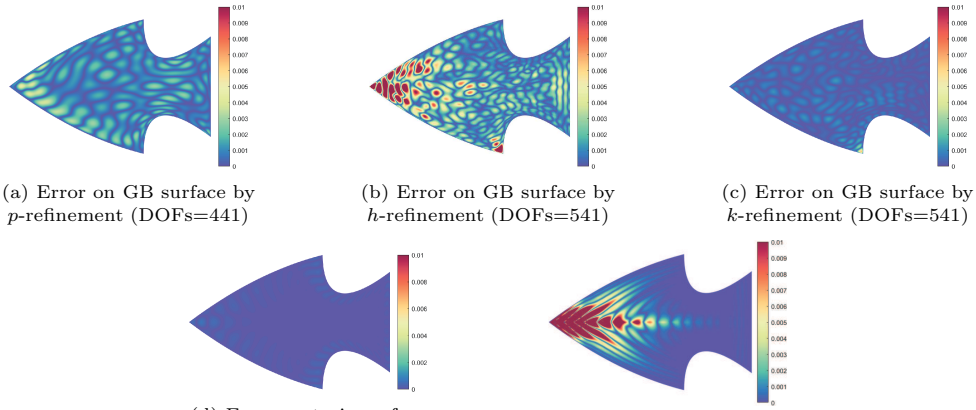
(a) J_s on GB surface (b) J_s on toric surface (c) J_s on NURBS surface

(d) S_a on GB surface (e) S_a on toric surface (f) S_a on NURBS surface

Fig. 5.6. Scaled Jacobian J_s and absolute relative area S_a for Example 5.2.

The colormaps of absolute error are illustrated in Fig. 5.7. To compare clearly, the errors are plotted on an identical scale. It can be seen that the error of the NURBS surface is much higher than the other two methods. This is caused by low continuity within the parameterization. Compared with the NURBS surface, GB and toric surfaces have a natural advantage in representing polygonal regions because of their properties. Therefore, both of them behave better than NURBS.

Fig. 5.8 shows the error history during refinement. In this example, the toric surface behaves slightly better than the GB surface. This is because toric parameterization has better orthogonality where the exact solution varies greatly (i.e. the fish head in Fig. 5.5(a)).



(a) Error on GB surface by p-refinement (DOFs=441)

(b) Error on GB surface by h-refinement (DOFs=541)

(c) Error on GB surface by k-refinement (DOFs=541)

(d) Error on toric surface (DOFs=391)

(e) Error on NURBS surface (DOFs=780)

Fig. 5.7. Absolute errors for Example 5.2.

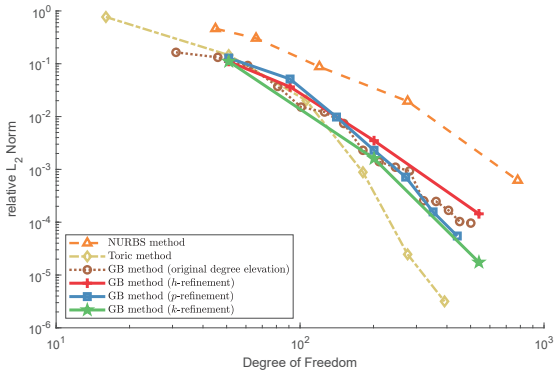


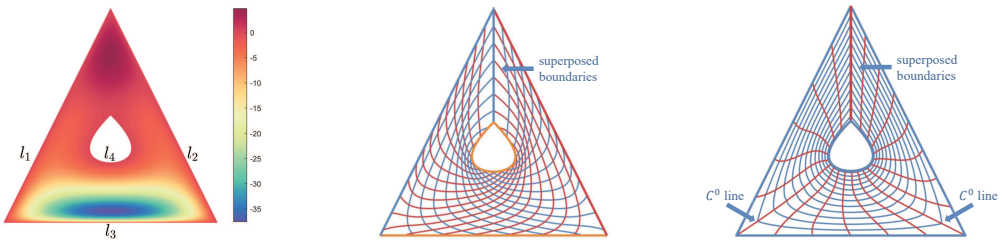
Fig. 5.8. Example 5.2: Error history of different methods.

5.3.1. High-genus domain

In this section, we consider Poisson's equations with homogeneous Dirichlet boundary conditions over two more complicated high-genus domains. The computational domains with hole(s) are parameterized in two different ways. In Example 5.3, a single GB patch representation is employed to represent genus-1 domain. On the contrast, a multi-patch configuration is adopted to generate a parameterization with low distortion in Example 5.4.

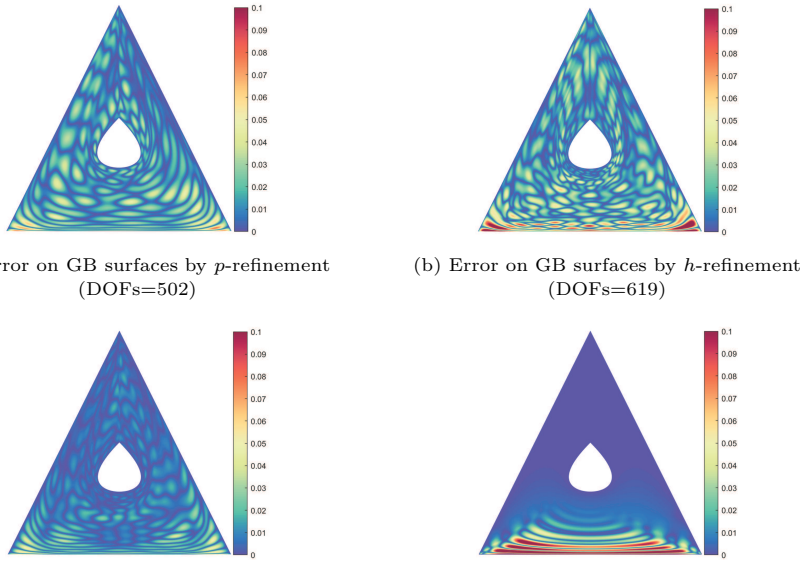
Example 5.3 (Single Patch Case). Consider a Poisson's equation with the exact solution $T = \prod_{i=1}^{4} l_i$, as shown in Fig. 5.9(a). As shown in Fig. 5.9(b), the computational domain is parameterized by a hexagonal GB patch, and two boundaries coincide to generate this genus-1 geometry. The GB parameterization is smooth apart from the superposed boundaries. Fig. 5.9(c) shows the NURBS parameterization. Two repetitive knots are needed to represent the computational domain restricted to the quadrilateral nature of NURBS. Therefore, three C^0 isoparametric lines exist in NURBS-based parameterization. Although the control points for internal repetitive knots serve as optimization variables in parameterization construction, distinct continuity degeneration is visible around the left bottom and right bottom corners.

The colormaps of the absolute errors under an identical scale are shown in Fig. 5.10. Compared with p-refinement and h-refinement schemes, k-refinement scheme shows a better error performance. Moreover, the absolute error of the NURBS method is higher around the bottom of the computational domain where the exact solution sharply changes. The error history during various refinement schemes is illustrated in Fig. 5.11. Compared with NURBS method, the GB method performs better in terms of numerical accuracy.



(a) Exact solution (b) GB parameterization (c) NURBS parameterization

Fig. 5.9. Genus-1 computational domain.



(a) Error on GB surfaces by p-refinement (DOFs=502)

(b) Error on GB surfaces by h-refinement (DOFs=619)

(c) Error on GB surfaces by k-refinement (DOFs=619)

(d) Error on NURBS surfaces (DOFs=1140)

Fig. 5.10. Absolute errors for Example 5.3.

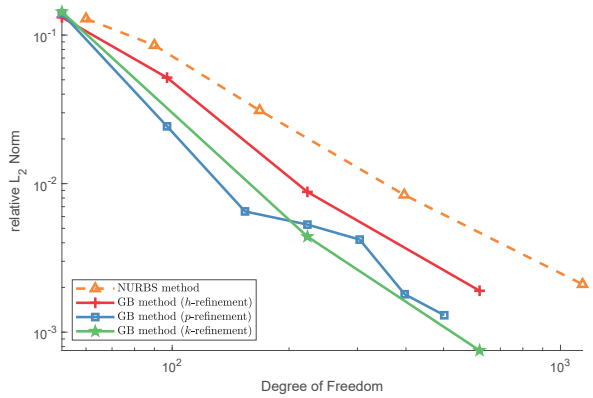
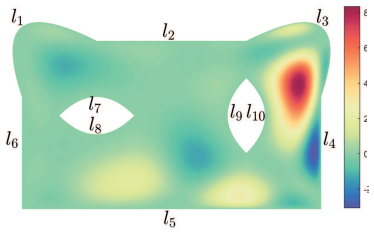


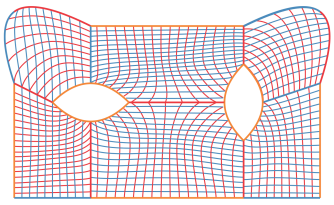
Fig. 5.11. Example 5.3: Error history of different methods.

Example 5.4 (Multi-Patch Case). For the computational domain with multi-holes inside, it is hard to ensure the quality of a single patch parameterization. To this end, multi-patch configurations are usually adopted [1, 36, 40]. In this example, the geometry of computational domain and the exact solution $T = \prod_{i=1}^{10} l_i$ are shown in Fig. 5.12(a). Thanks to the virtue of GB surface in representing multi-sided domains, it is possible to use fewer patches than NURBS-based multi-patch parameterization. As shown in Fig. 5.12(b), two hexagonal patches and four pentagonal patches are used for multi-patch GB parameterization. A NURBS-based multi-patch parameterization composed of twelve quadrilateral patches is shown in Fig. 5.12(c).

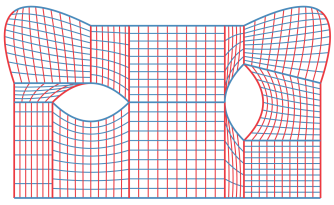
The colormaps of the absolute error distribution and the error history using various refinement schemes are shown in Figs. 5.13 and 5.14. Again, a lower numerical error of GB patches can be observed compared with NURBS patches. In this example, k-refinement and p-refinement schemes have a similar performance in terms of numerical error as shown in the error history, which are better than h-refinement scheme. It demonstrates that our approach can be applied to multi-patch configurations seamlessly.



(a) Exact solution



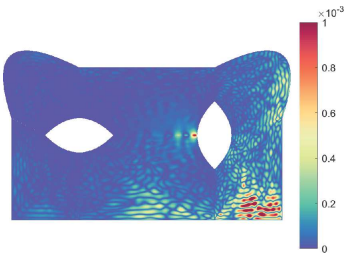
(b) GB parameterization



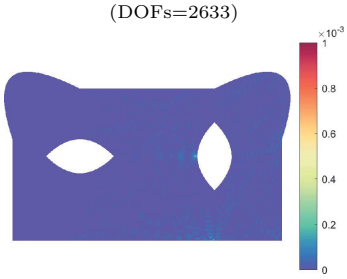
(c) NURBS parameterization

Fig. 5.12. Genus-2 computational domain.

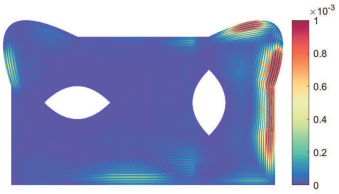
(a) Error on GB surfaces by p-refinement (DOFs=2633)



(b) Error on GB surfaces by h-refinement (DOFs=3252)



(c) Error on GB surfaces by k-refinement (DOFs=3252)



(d) Error on NURBS surfaces (DOFs=4086)

Fig. 5.13. Absolute errors for Example 5.4.

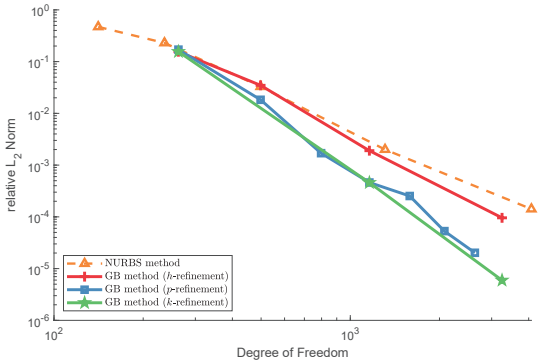


Fig. 5.14. Example 5.4: Error history of different methods.

5.4. Linear elasticity problem

In the following, the errors are computed by displacement error norm

$$e_1 = \sqrt{\int_\Omega (\mathbf{v} - \mathbf{v}^h)^{\mathrm{T}} (\mathbf{v} - \mathbf{v}^h) d\Omega}, \tag{5.4}$$

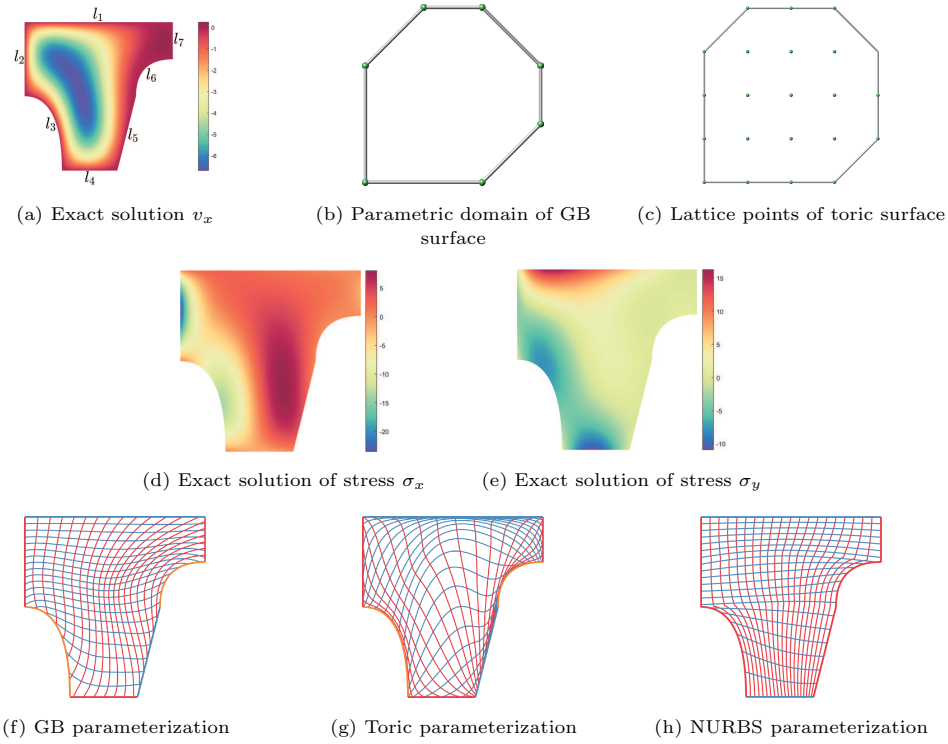
and energy error norm

$$e_2 = \sqrt{\int_\Omega (\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^h)^{\mathrm{T}} (\boldsymbol{\sigma} - \boldsymbol{\sigma}^h) d\Omega}, \tag{5.5}$$

where $\mathbf{v}, \boldsymbol{\sigma}$ and $\boldsymbol{\varepsilon}$ are the exact displacement, stress, and strain, and $\mathbf{v}^h, \boldsymbol{\sigma}^h$ and $\boldsymbol{\varepsilon}^h$ are the numerical displacement, stress, and strain, respectively.

Consider linear elasticity problem (4.9) over a heptagonal computational domain, the exact solutions of displacement in x- and y-directions are given as $v_x = v_y = 10^{-3} \prod_{i=1}^{7} l_i$ as shown

in Fig. 5.15(a). The body force is obtained by $\mathbf{s}(\mathbf{x}) = \mathbf{L}^{\mathrm{T}}\boldsymbol{\sigma}(\mathbf{x})$. The parametric domains of GB and toric parameterizations are provided in Figs. 5.15(b) and 5.15(c). Figs. 5.15(d)-5.15(e) illustrate the exact solutions of stress in x- and y-direction respectively. The last row of Fig. 5.15 shows the resulting parameterizations of three methods. Fig. 5.16 provides the consequence of scaled Jacobian and absolute relative area on three different surfaces. One can observe that GB parameterization has better uniformity and orthogonality compared with toric parameterization.

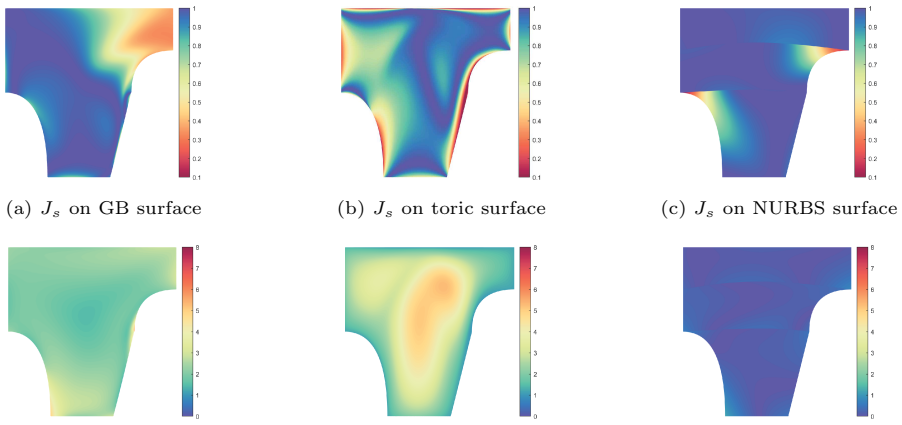


(a) Exact solution v_x (b) Parametric domain of GB surface (c) Lattice points of toric surface

(d) Exact solution of stress σ_x (e) Exact solution of stress σ_y

(f) GB parameterization (g) Toric parameterization (h) NURBS parameterization

Fig. 5.15. Pentagonal computational domain.



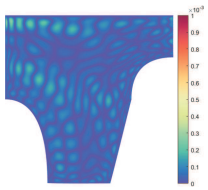
(a) J_s on GB surface (b) J_s on toric surface (c) J_s on NURBS surface

(d) S_a on GB surface (e) S_a on toric surface (f) S_a on NURBS surface

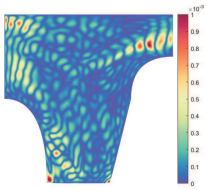
Fig. 5.16. Scaled Jacobian J_s and absolute relative area S_a for Example 5.3.

For comparison purposes, all the following figures of computational error are rendered with the same colormap. Fig. 5.17 provide the error colormaps of displacement v_x in the x-direction. The errors of stresses in the x-direction of various methods are shown in Fig. 5.18. The GB and toric parameterizations behave better than the NURBS parameterization due to the high continuity inside, which is similar to Poisson's equation in the previous examples. Fig. 5.19 displays the error history of displacement and energy norm. This example demonstrates the effectiveness and robustness of the proposed refinement methods.

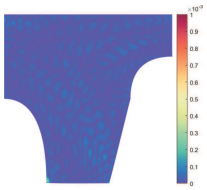
Most importantly, we emphasize that the numerical stress obtained by NURBS parameterization is discontinuous along the interfaces due to the reduced continuity for geometry requirements (please refer to Fig. 5.20). On the contrary, the proposed method produces continuous stress, which is one of the advantages of our method.



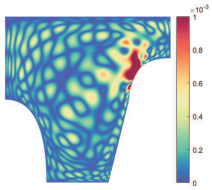
(a) $|v_x - v_x^h|$, GB surface, p-refinement (DOFs=1234)



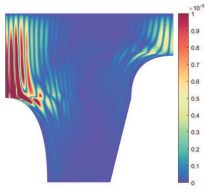
(b) $|v_x - v_x^h|$, GB surface, h-refinement (DOFs=1514)



(c) $|v_x - v_x^h|$, GB surface, k-refinement (DOFs=1514)

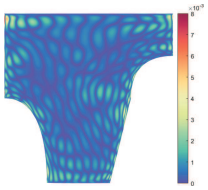


(d) $|v_x - v_x^h|$, toric surface (DOFs=1124)

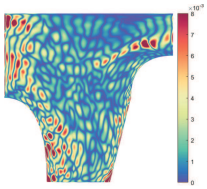


(e) $|v_x - v_x^h|$, NURBS surface (DOFs=2090)

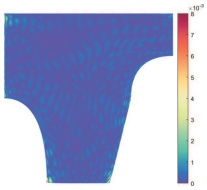
Fig. 5.17. Absolute errors of displacement in x-direction.



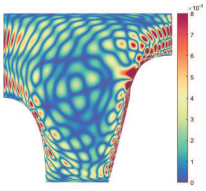
(a) $|\sigma_x - \sigma_x^h|$, GB surface, p-refinement (DOFs=1234)



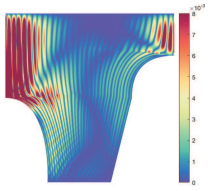
(b) $|\sigma_x - \sigma_x^h|$, GB surface, h-refinement (DOFs=1514)



(c) $|\sigma_x - \sigma_x^h|$, GB surface, k-refinement (DOFs=1514)



(d) $|\sigma_x - \sigma_x^h|$, toric surface (DOFs=1124)



(e) $|\sigma_x - \sigma_x^h|$, NURBS surface (DOFs=2090)

Fig. 5.18. Absolute errors of stress in x-direction.

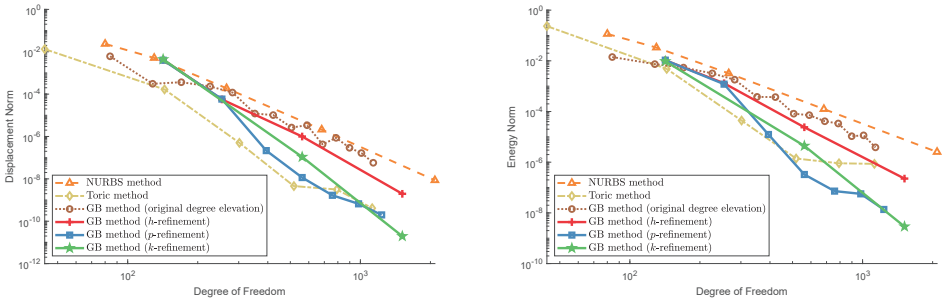
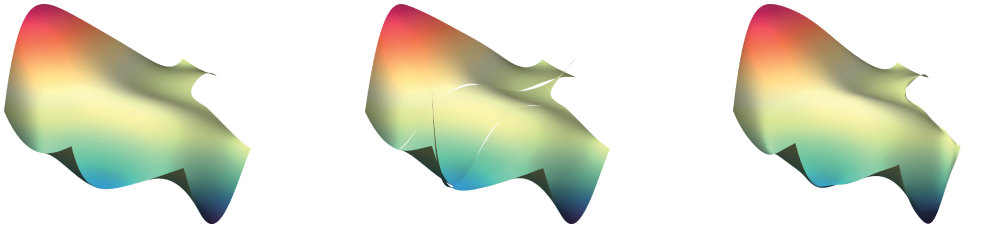


Fig. 5.19. History of displacement error norm (left) and energy error norm (right).



(a) Exact stress σ_y (b) σ_y^h, NURBS surface (DOFs=266) (c) σ_y^h, GB surface, p-refinement (DOFs=254)

Fig. 5.20. The exact and numerical stresses in y-direction.

6. Conclusions and Future Work

In this paper, an improved degree elevation and a novel knot insertion for GB surfaces are proposed, which keep surfaces unchanged. The control points and weights of the GB surface are evaluated together based on the degree elevation of tensor product Bézier surface and the knot insertion of tensor product B-spline surface. For IGA applications, GB surfaces are used to generate high-quality parameterizations of multi-sided computational domains. The applications to Poisson's equations and linear elasticity problems demonstrate the effectiveness and robustness of the proposed p-refinement using the proposed degree elevation, h-refinement using the knot insertion, and k-refinement combining the above two methods.

However, the control points obtained by the proposed methods are with parameters, and it is not friendly to some applications. Therefore, we hope to present a refinement method enriching control points without parameters in the future. Besides, the construction of high-quality volumetric parameterization is more challenging and of practical application significance. The difficulties are two-fold. First, 3D generalized barycentric coordinates are needed. As far as the authors know, some well-defined 3D generalized barycentric coordinates might be helpful [9]. Second, an appropriate definition of 3D local parameters, which maps a polyhedron to a unit cube and guarantees the bijectivity, is difficult to acquire. Extending the proposed method to 3D problems is one of our ongoing works.

Moreover, extending the proposed techniques to adaptive refinement/coarsening would be of theoretical and practical significance. A possible implementation might be replacing the B-spline basis functions in (3.10) with splines with local refinement ability, such as THB-splines [2, 4, 22]. Therefore, adaptive refinement/coarsening techniques for GB surfaces will be an interesting future direction.

Acknowledgments. We thank all the anonymous reviewers for their valuable comments and constructive suggestions. The authors would like to thank Dr. Péter Salvi, Budapest University of Technology and Economics, Hungary, for valuable discussions of GB surfaces.

This work was supported by the National Natural Science Foundation of China (Grant Nos. 12071057, 11671068, 12001327), and by the Fundamental Research Funds for the Central Universities. Y. Ji was also partially supported by the China Scholarship Council (Grant No. 202106060082).

References

[1] F. Buchegger and B. Jüttler, Planar multi-patch domain parameterizaton via patch adjacency graphs, *Comput. Aided Des.*, **82** (2017), 2–12.

[2] M. Carraturo, C. Giannelli, A. Reali, and R. Vázquez, Suitably graded THB-spline refinement and coarsening: Towards an adaptive isogeometric analysis of additive manufacturing processes, *Comput. Methods Appl. Mech. Engrg.*, **348** (2019), 660–679.

[3] E. Cohen, T. Martin, R.M. Kirby, T. Lyche, and R.F. Riesenfeld, Analysis-aware modeling: Understanding quality considerations in modeling for isogeometric analysis, *Comput. Methods Appl. Mech. Engrg.*, **199**:5–8 (2010), 334–356.

[4] L. Coradello, D.D. Angella, M. Carraturo, J. Kiendl, S. Kollmannsberger, E. Rank, and A. Reali, Hierarchically refined isogeometric analysis of trimmed shells, *Comput. Mech.*, **66** (2020), 431–447.

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

[6] J.A. Cottrell, A. Reali, Y. Bazilevs, and T.J.R. Hughes, Isogeometric analysis of structural vibrations, *Comput. Methods Appl. Mech. Engrg.*, **195**:41–43 (2006), 5257–5296.

[7] G. Farin, *Curves and Surfaces for CAGD: A Practical Guide*, Morgan Kaufmann, 2002.

[8] G. Farin, J. Hoschek, and M. Kim, *Handbook of Computer Aided Geometric Design*, Elsevier, 2002.

[9] M.S. Floater, *Wachspress and Mean Value Coordinates*, in: *Approximation Theory XIV: San Antonio 2013*, Springer, Vol. 83, 2014.

[10] L. García-Puente, F. Sottile, and C.G. Zhu, Toric degenerations of Bézier patches, *ACM Trans. Graph.*, **30**:5 (2011), 110.

[11] T.J.R. Hughes, J.A. Cottrell, and Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, *Comput. Methods Appl. Mech. Engrg.*, **194**:39-41 (2005), 4135–4195.

[12] Y. Ji, J.G. Li, Y.Y. Yu, and C.G. Zhu, *h*-Refinement method for toric parameterization of planar multi-sided computational domain in isogeometric analysis, *Comput. Aided Geom. Des.*, **93** (2022), 102065.

[13] Y. Ji, M.Y. Wang, M.D. Pan, Y. Zhang, and C.G. Zhu, Penalty function-based volumetric parameterization method for isogeometric analysis, *Comput. Aided Geom. Des.*, **94** (2022), 102081.

[14] Y. Ji, M.Y. Wang, Y. Wang, and C.G. Zhu, Curvature-based r-adaptive planar NURBS parameterization method for isogeometric analysis using bi-level approach, *Comput. Aided Des.*, **150** (2022), 103305.

[15] Y. Ji, Y.Y. Yu, M.Y. Wang, and C.G. Zhu, Constructing high-quality planar NURBS parameterization for isogeometric analysis by adjustment control points and weights, *J. Comput. Appl. Math.,* **396** (2021), 113615.

[16] R. Krasauskas, Toric surface patches, *Adv. Comput. Math.*, **17** (2002), 89–113.

[17] N. Lei, X.P. Zheng, Z.X. Luo, and X.F. Gu, Quadrilateral and hexahedral mesh generation based on surface foliation theory II, *Comput. Methods Appl. Mech. Engrg.*, **321** (2017), 406–426.

[18] J.G. Li, Y. Ji, and C.G. Zhu, De Casteljau algorithm and degree elevation of toric surface patches, *J. Syst. Sci. Complex.*, **34** (2021), 21–46.
[19] X. Li and M.A. Scott, Analysis-suitable T-splines: Characterization, refineability, and approximation, *Math. Models Methods Appl. Sci.*, **24**:6 (2014), 1141–1164.
[20] X. Li, J.M. Zheng, T.W. Sederberg, T. Hughes, and M.A. Scott, On linear independency of T-splines blending functions, *Comput. Aided Geom. Des.*, **29**:1 (2012), 63–76.
[21] C.T. Loop and T.D. Derose, A multisided generalization of Bézier surfaces, *ACM Trans. Graph.*, **8**:3 (1989), 204–234.
[22] G. Lorenzo, M.A. Scott, K. Tew, T.J.R. Hughes, and H. Gomez, Hierarchically refined and coarsened splines for moving interface problems, with particular application to phase-field models of prostate tumor growth, *Comput. Methods Appl. Mech. Engrg.*, **319** (2017), 515–548.
[23] X.S. Nian and F.L. Chen, Planar domain parameterization for isogeometric analysis based on Teichmüller mapping, *Comput. Methods Appl. Mech. Engrg.*, **311** (2016), 41–55.
[24] M.D. Pan, F.L. Chen, and W.H. Tong, Volumetric spline parameterization for isogeometric analysis, *Comput. Methods Appl. Mech. Engrg.*, **359** (2020), 112769.
[25] L. Piegl and W. Tiller, *The NURBS Book*, Springer Science & Business Media, 1997.
[26] E. Pilgerstorfer and B. Jüttler, Bounding the influence of domain parameterization and knot spacing on numerical stability in isogeometric analysis, *Comput. Methods Appl. Mech. Engrg.*, **268** (2014), 589–613.
[27] M. Vaitkus, T. Várady, P. Salvi, and Á. Sipos, Multi-sided B-spline surfaces over curved, multi-connected domains, *Comput. Aided Geom. Des.*, **89** (2021), 102019.
[28] T. Várady, P. Salvi, and G. Karikó, A multi-sided Bézier patch with a simple control structure, *Comput. Graph. Forum*, **35**:2 (2016), 307–317.
[29] T. Várady, P. Salvi, and I. Kovács, Enhancement of a multi-sided Bézier surface representation, *Comput. Aided Geom. Des.*, **55** (2017), 69–83.
[30] T. Várady, P. Salvi, M. Vaitkus, and Á. Sipos, Multi-sided Bézier surfaces over curved, multi-connected domains, *Comput. Aided Geom. Des.*, **78** (2020), 101828.
[31] E.L. Wachspress, *A Rational Finite Element Basis*, Academic Press, 1975.
[32] D.D. Wang, J.L. Xu, F. Gao, C.C.L. Wang, R.S. Gu, F. Lin, T. Rabczuk, and G. Xu, IGA-ReuseNET: A deep-learning-based isogeometric analysis-reuse approach with topology-consistent parameterization, *Comput. Aided Geom. Des.*, **95** (2022), 102087.
[33] J. Xie, J.L. Xu, Z.Y. Dong, G. Xu, C.Y. Deng, B. Mourrain, and Y.J. Zhang, Interpolatory Catmull-Clark volumetric subdivision over unstructured hexahedral meshes for modeling and simulation applications, *Comput. Aided Geom. Des.*, **80** (2020), 101867.
[34] G. Xu, T.H. Kwok, and C.C.L. Wang, Isogeometric computation reuse method for complex objects with topology-consistent volumetric parameterization, *Comput. Aided Des.*, **91** (2017), 1–13.
[35] G. Xu, B.J. Li, L.X. Shu, L. Chen, J.L. Xu, and T. Khajah, Efficient r-adaptive isogeometric analysis with Winslows mapping and monitor function approach, *J. Comput. Appl. Math.*, **351** (2019), 186–197.
[36] G. Xu, M. Li, B. Mourrain, T. Rabczuk, J.L. Xu, and S.P.A. Bordas, Constructing IGA-suitable planar parameterization from complex CAD boundary by domain partition and global/local optimization, *Comput. Methods Appl. Mech. Engrg.*, **328** (2018), 175–200.
[37] G. Xu, B. Mourrain, R. Duvigneau, and A. Galligo, Optimal analysis-aware parameterization of computational domain in 3D isogeometric analysis, *Comput. Aided Des.*, **45**:4 (2013), 812–821.
[38] G. Xu, B. Mourrain, R. Duvigneau, and A. Galligo, Constructing analysis-suitable parameterization of computational domain from CAD boundary by variational harmonic method, *J. Comput. Phys.*, **252** (2013), 275–289.
[39] G. Xu, B. Mourrain, R. Duvigneau, and A. Galligo, Analysis-suitable volume parameterization of multi-block computational domain in isogeometric applications, *Comput. Aided Des.*, **45**:2

(2013), 395–404.

[40] J.L. Xu, F.L. Chen, and J.S. Deng, Two-dimensional domain decomposition based on skeleton computation for parameterization and isogeometric analysis, *Comput. Methods Appl. Mech. Engrg.*, **284** (2015), 541–555.

[41] Y.Y. Yu, Y. Ji, and C.G. Zhu, An improved algorithm for checking the injectivity of 2D toric surface patches, *Comput. Math. with Appl.*, **79** (2020), 2973–2986.

[42] X.F. Zhu, Y. Ji, C.G. Zhu, P. Hu, and Z.D. Ma, Isogeometric analysis for trimmed CAD surfaces using multi-sided toric surface patches, *Comput. Aided Geom. Des.*, **79** (2020), 101847.