Mixed Bernstein-Bézier Construction from Unstructured Mesh for Higher-order Finite Element Analysis of Plates and Shells

Xiaoxiao Du[1] , Gang Zhao[2] , Wei Wang[3]

[1]Beihang University, duxiaoxiao@buaa.edu.cn
[2]Beihang University, zhaog@buaa.edu.cn
[3]Beihang University, jrrt@buaa.edu.cn

Corresponding author: Wei Wang, jrrt@buaa.edu.cn

Abstract. Bernstein polynomials possess several efficient and stable algorithms and have been widely used in engineering fields such as CAGD, computer graphics, and finite element analysis. In this paper, a mixed Bernstein basis function based higher-order finite element method is developed from a geometric view for the analysis and simulation of plate and shell structures represented by unstructured triangular and quadrilateral mesh. Triangular Bernstein-Bézier patches and tensor-product Bézier patches are constructed in a simple and intuitive way over triangular and quadrilateral elements, respectively. The h- and p-refinements can be easily implemented on the constructed mixed Bernstein-Bézier patches. Reissner-Mindlin theory is employed to deduce the governing equations and stiffness matrices of plates and shells. Several numerical examples including classical benchmark problems and engineering applications are studied to validate the accuracy, robustness, and convergence of the presented Bernstein-Bézier finite element method.

1 INTRODUCTION

Higher-order, also called *p*-version, finite element methods have been widely investigated in the industrial environment due to their superiorities on convergence rates and description of curved geometries. In addition, they possess particular advantages for some problems. For example, locking phenomena can be eliminated or greatly alleviated by using higher-order elements in the analysis of thin-walled structures [5, 6, 15]. In the simulation of lamb waves, higher-order finite element methods with polynomial degrees $p > 2$ are widely used to reduce the enormous computational costs [14, 40]. Therefore, a variety of higher-order elements and corresponding shape functions have been proposed [48]. Bernstein polynomials [18, 19] provide a series of

attractive properties and elegant algorithms e.g. *de Casteljau* algorithm for various computational applications besides its widespread use in fields of CAGD and computer graphics. Nevertheless, they have received virtually no attention in finite element approximation for many decades since that the interpolated nodes might be expected in finite element method instead of the control points away from physical elements [2, 21].

Recently Bernstein polynomials have been frequently used for finite element approximation and variables representation in FEA community. Bogdanovich [7] introduced the tensor-product Bernstein basis functions in three dimensions to approximate the displacement of hexahedral finite elements for the stress analysis of laminated plates with anisotropic layers. He [8] also gave a review about using Bernstein approximation polynomials in the hexahedral discrete element of composite structures for three-dimensional variational analysis and its application on crack propagation. Schumaker and Wang [35] firstly studied the spaces of Bernstein polynomials defined on mixed meshes with hanging vertices consisting of triangles and rectangles. Hanging vertices can provide much simpler local refinement algorithms for finite element method. Chan and Warburton [10] investigated the construction of pyramid based on Bernstein-Bézier basis. Ainsworth et al. [3] described the Bernstein-Bézier representation of finite elements defined on mixed three-dimensional partitions consisting of tetrahedra, hexahedra and pyramids. They [2] as well presented some elegant algorithms for computation and assembly of the element matrices including mass, stiffness and convective matrices in optimal complexity. Kirby and Thinh [28, 29] proposed fast algorithms for computing the mass, stiffness matrices and quadrature over simplices by using Bernstein polynomials. The foundation and motivation of these investigations are based on the well-known properties and algorithms of Bernstein basis functions.

Apart from the application of Bernstein polynomials to finite element methods, it also attracted much attention of practitioners in isogeometric analysis. Finite element data structures were proposed for isogeometric analysis based on the extraction of tensor-product Bézier elements from NURBS and T-splines [9, 36]. The greatest virtue of this method is that the extracted Bézier elements can be treated as the standard elements used in the classical FEA, which means the existing finite element program can easily incorporate Bézier elements only with a slight modification. Speleers et al. [38, 37] employed triangular Powell-Sabin splines based isogeometric analysis for parameter optimization and numerical simulation of advection-diffusion-reaction problems. Da Veiga et al. [12] investigated isogeometric analysis on Reissner-Mindlin plate problems with quadratic triangular NURPS to preclude the well-known "shear locking" phenomenon. Jaxon and Qian [25] presented a new approach to rebuild NURBS curve-bounded domains with rational triangular Bézier splines (rTBS) for isogeometric analysis. Afterward, Xia et al. [42] from the same group validated that the optimal convergence rates could be achieved in Poisson and linear elasticity problems with rTBS-based IGA. Motivated by the problem of surface-to-volume parametrization in IGA, Engvall and Evans [16, 17] proposed to automatically generate rational triangular Bézier patches from a curve-bounded domain in two dimension and rational Bézier hexahedra, tetrahedra, wedges and pyramids from NURBS surface-bounded domain in three dimensions. Zeng and Cohen [45] employed trivariate B-spline elements and Bézier tetrahedral elements to represent a 2-manifold volume. Xia and Qian [41] used C^r rational trivariate Bernstein-Bézier polynomials to represent solid models built with enclosed NURBS surfaces. It can be found that Bernstein-Bézier polynomials over triangles or tetrahedrons have been successfully applied in the isogeometric analysis.

Thin-walled structures like plates and shells are commonplace in engineering applications including vehicle, vessel, aircraft and architecture. Since plates and shells simplify the complex three-dimensional configurations and greatly save the computation time in finite element methods, they have been widely investigated and are still a significant topic nowadays [6]. In this paper, we only focus on the unstructured mesh consisting of both rectangular elements and triangular elements, and introduce Bernstein-Bézier polynomials of arbitrary order to represent unstructured mesh from a geometric view for analysis and simulation of the plate and shell structures. Reissner-Mindlin theory is introduced to deduce the governing equations and matrix formulations of the plates and shells. It assumes that any cross-section of plates and shells can remain straight but not perpendicular to the middle surface during the deformation [24, 47]. Meanwhile, it can be used to analyze both thin and thick plates and shells and only C^0 continuity should be satisfied across the common boundaries of the adjacent

elements. Since we have an exact representation of the Bernstein-Bézier elements, the normal vector of the elements can be computed exactly. The normal vectors defined at *Greville abscissa* were considered as a good compromise between accuracy and computation time [1]. Note that the shear locking is not a distracting problem because higher order Bézier elements are used. To verify the proposed method, We investigate the static bending and free vibration of the plates and shells, and compare the obtained numerical results with analytical solutions and that from software ABAQUS.

The remainder of this paper is organized as follows. In section 2, we review the concepts and formulations of triangular Bernstein-Bézier patch and tensor-product Bézier patch. The construction of mixed Bézier patches over triangular and quadrilateral mesh is described in detail in section 3 and section 4 demonstrates the hp-refinement techniques of the mixed Bézier patches. Section 5 deduces the finite element formulations for Reissner-Mindlin plate and shell problems. Several numerical examples are investigated in Section 7.

2   PRELIMINARIES

In this section, we briefly reviewed the Bernstein basis, triangular Bézier patch and tensor-product Bézier patch. Readers are referred to [18, 19] for more details on Bernstein-Bézier patches.

2.1   Bernstein Basis

Bernstein polynomial basis was firstly proposed by Sergei Natanovich Bernstein in 1912 to approximate the continuous functions by polynomials, even though it has been proved to be versatile in interactive design of polynomial functions after several decades [21]. The Bernstein basis functions of degree n on $t \in [0,1]$ are given as

$$B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i}, \quad i = 0, 1, \cdots, n. \tag{1}$$

By defining $u = t, v = 1 - t$, it also can be rewritten as

$$B_{ij}^n(u,v) = \frac{n!}{i!j!} u^i v^j, \quad 0 \leq i, j \leq n, i + j = n, \tag{2}$$

where u, v are called *barycentric coordinates*. A polynomial of Bernstein form associated with arbitrary coefficients $c_i, i = 0, 1, 2, \cdots, n$ is defined by

$$f^n(t) = \sum_{i=0}^{n} c_i B_i^n(t), \quad t \in [0,1]. \tag{3}$$

The Bernstein-form polynomials inherit some attractive properties from Bernstein basis functions such as symmetry, non-negativity, the partition of unity, recursion and numerical stability. Meanwhile, some concise and efficient algorithms of Bernstein basis like degree elevation, derivatives, integrals and arithmetic operations will bring great convenience of calculation to polynomials in Bernstein form, and thus have been widely applied in CAGD, computer graphics and finite element analysis. Here we simply review the derivatives algorithm and degree elevation algorithm of Bernstein basis functions for its computational requirements in FEA. The first-order derivatives of $B_i^n(t)$ are defined by

$$\frac{d}{dt}B_i^n(t) = n\big[B_{i-1}^{n-1}(t) - B_i^{n-1}(t)\big] \tag{4}$$

with $B_{-1}^{n-1} = B_n^{n-1} \equiv 0$. The formula of degree elevation can be expressed as

$$B_i^n(t) = \left(1 - \frac{i}{n+1}\right)B_i^{n+1}(t) + \frac{i+1}{n+1}B_{i+1}^{n+1}(t). \tag{5}$$
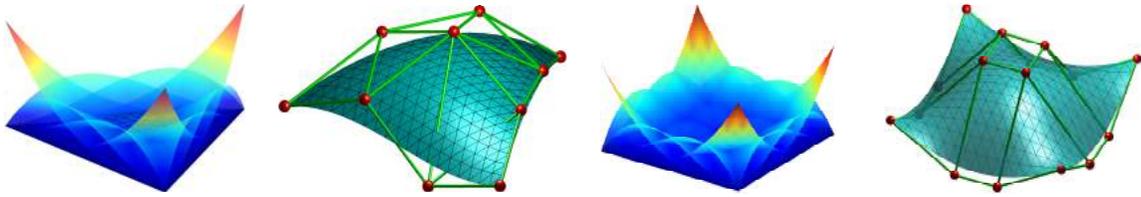
(a) Triangular Bézier basis   (b) Triangular Bézier patch   (c) Rectangular Bézier basis   (d) Rectangular Bézier patch

Figure 1: Cubic triangular and bi-cubic rectangular Bézier patches. (a)-(b) Ten Bernstein basis functions and the control net of a cubic triangular Bézier surface; (c)-(d) Sixteen Bernstein basis functions and the control net of a bi-cubic rectangular Bézier surface.

The formulae of derivatives and degree elevation of Bernstein-form polynomials can be deduced based on Eqs. 4-5.

2.2 Triangular Bézier Patch

The multivariate extension is obvious and natural with the definition of univariate Bernstein basis in Eq. 1. It is commonplace to define bivariate Bernstein basis over simplex domains such as triangles and quadrangles. Given a non-degenerate triangle $\mathcal{T}$ with three vertices $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$, the barycentric coordinates (u, v, w) of any point $\mathbf{v}$ in the triangle plane are coincided with area coordinates and can be written as

$$u = \frac{S(\mathbf{v}, \mathbf{v}_1, \mathbf{v}_2)}{S(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)}, v = \frac{S(\mathbf{v}_1, \mathbf{v}, \mathbf{v}_3)}{S(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)}, w = \frac{S(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v})}{S(\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3)},$$

where $S(\cdot)$ denote the area of the corresponding triangles. It is obvious that $0 \leq u, v, w \leq 1, u + v + w = 1$. The bivariate Bernstein basis of degree n functions over the triangle $\mathcal{T}$ can be extended from Eq. 2 as

$$B_{ijk}^n(u, v, w) = \frac{n!}{i!j!k!} u^i v^j w^k \tag{6}$$

with $0 \leq i, j, k \leq n, i + j + k = n$. Then an n-th bivariate polynomial can be expressed as

$$f^n(u, v, w) = \sum_{i+j+k=n} c_{ijk} B_{ijk}^n(u, v, w), \tag{7}$$

where the c_{ijk} are called *Bézier ordinates* or *B-coefficients* of f^n [18, 30]. Substituting the c_{ijk} in Eq. 7 by a set of vector-valued *control points* $\mathbf{b}_{ijk}$, we can obtain a *triangular Bézier patch* **T** in physical domain mapping from the parametric domain $\mathcal{T}$,

$$\mathbf{T}(u, v, w) = \sum_{i+j+k=n} \mathbf{b}_{ijk} B_{ijk}^n(u, v, w). \tag{8}$$

The *control net* of the triangular Bézier patch **T** consists of its ordered control points $\mathbf{b}_{ijk}$.

The construction of a triangular Bézier patch was initially proposed by *de Casteljau* in the late 1950s, who extended the concept of Bézier curve to triangular surface before defining tensor product Bézier surface [18]. As an example, Figures 1c-1d illustrate the ten Bernstein basis functions and the control net of a cubic triangular Bézier patch.

Assuming that u and v are two independent variables in Bernstein basis functions, the first-order partial derivatives of the B_{ijk}^n in u-direction are given by

$$\frac{\partial}{\partial u} B_{i,j,k}^n(u, v, w) = n \big[B_{i-1,j,k}^{n-1}(u, v, w) - B_{i,j-1,k}^{n-1}(u, v, w) \big]. \tag{9}$$

Here the B_{ijk}^n is substituted by the $B_{i,j,k}^n(u, v, w)$ for the better demonstration. The partial derivatives in v-direction have the similar form.

The once degree elevation of an n-th degree triangular Bézier patch can be expressed by

$$\mathbf{T}(u, v, w) = \sum_{i+j+k=n} \mathbf{b}_{ijk} B_{ijk}^n(u, v, w) = \sum_{i+j+k=n+1} \hat{\mathbf{b}}_{ijk} B_{ijk}^{n+1}(u, v, w), \tag{10}$$

where $\hat{\mathbf{b}}_{ijk}$ denote the control points of the triangular Bézier patch after once degree elevation and are deduced as

$$\hat{\mathbf{b}}_{i,j,k} = \frac{i\mathbf{b}_{i-1,j,k} + j\mathbf{b}_{i,j-1,k} + k\mathbf{b}_{i,j,k-1}}{n+1}, \quad i+j+k = n+1. \tag{11}$$

2.3 Tensor-product Bézier Patch

Tensor-product Bézier patch has been maturely developed in the field of CAGD and widely used in different CAD systems. It can be easily extracted from the B-spline surface with knot insertion operation [9]. A tensor-product Bézier patch is defined by the products of its bidirectional control points and univariate Bernstein basis functions

$$\mathbf{S}(u, v) = \sum_{i=0}^{n} \sum_{j=0}^{m} B_i^n(u) B_j^m(v) \mathbf{b}_{i,j}, \quad 0 \le u, v \le 1, \tag{12}$$

where $B_i^n(u)$ and $B_j^m(v)$ are the univariate Bernstein basis functions in two directions, m, n denote the degrees, $\mathbf{b}_{i,j}$ are the surface control points. A bi-cubic tensor-product Bézier patch and its 16 bivariate Bernstein basis functions are depicted in Figs. 1a-1b. The derivatives and degree elevation of a tensor product Bézier patch can be easily extended from univariate Bernstein-form polynomials [19] and will not be reviewed here.

It should be noticed that *de Casteljau* recursive algorithm provides an intuitive, stable and effective approach to calculate the points on rectangular Bézier patch as well as triangular Bézier patch. Meanwhile, the degree elevation and surface subdivision algorithms are two fundamental methods in terms of tensor-product Bézier surface which are similar to p and h strategies in classical finite element method and are beneficial for presented Bernstein-Bézier finite element method.

3 BÉZIER PATCHES CONSTRUCTION

Triangular and quadrilateral elements are the frequently-used and important elements for analysis of surface geometries in the finite element method. For purpose of employing the virtues of Bézier elements based high-order FEA, we build tensor-product Bézier surfaces over quadrilateral elements and triangular Bézier surfaces over triangular elements. There are two main steps: the first step is to construct edge Bézier control points on each edge in mesh model and the next is to build inner control points for each element according to the obtained boundary Bézier control points. In this section, bi-cubic tensor-product Bézier patches and cubic triangular Bézier patches are built over mesh elements.

(a) Tangent vector   (b) Plane mesh edge points   (c) Surface mesh edge points
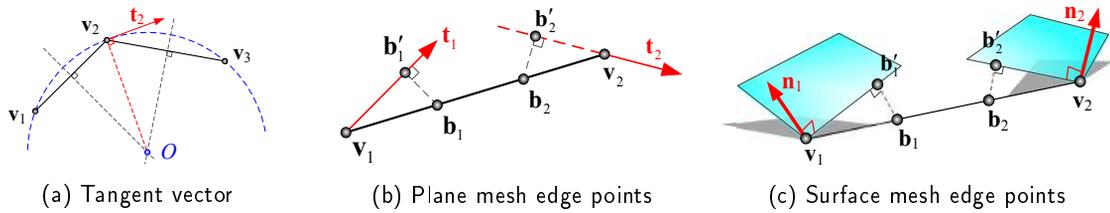
Figure 2: Edge Bézier control points construction. (a) and (b): the tangent vector and Bézier control points construction for plane mesh boundary edges; (c): Bézier control points construction for 3D surface mesh edges.

3.1 Edge Control Points Construction

For 2D plane mesh models, edges are divided into two parts: boundary edges and inner edges. We construct the Bézier control points for mesh boundary edges with a fairing strategy and for inner edges with a shape-preserving strategy.

Tangent vectors of the boundary vertices are defined firstly. Assuming that $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ are three successive boundary vertices as depicted in Fig. 2a, tangent vector $\mathbf{t}_2$ of the vertex $\mathbf{v}_2$ are defined by the circumcircle of vertices $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ with $\mathbf{O}\mathbf{v}_2 \cdot \mathbf{t}_2 = 0$ where point $\mathbf{O}$ is the center of the circumcircle. Tangent vector attached to each boundary vertex can be defined after iterating through all boundary vertices.

when obtaining the tangent vector of each boundary vertex, we can choose suitable points as Bézier control points for each boundary edges. As shown in Fig. 2b, e_{12} is a boundary edge and $\mathbf{t}_1, \mathbf{t}_2$ are two tangent vectors for vertices $\mathbf{v}_1$ and $\mathbf{v}_2$, where e_{ij} denote the edge built with the vertices $\mathbf{v}_i$ and $\mathbf{v}_j$. $\mathbf{b}_1, \mathbf{b}_2$ are two trisection points of the edge e_{12} with $\mathbf{b}_1 = 2/3\mathbf{v}_1 + 1/3\mathbf{v}_2$, $\mathbf{b}_2 = 1/3\mathbf{v}_1 + 2/3\mathbf{v}_2$. Bézier control points $\mathbf{b}_1'$ and $\mathbf{b}_2'$ are defined by the projection of two vertices $\mathbf{v}_1, \mathbf{v}_2$ onto the tangent vectors $\mathbf{t}_1, \mathbf{t}_2$. In formulae for implementation, control points $\mathbf{b}_1', \mathbf{b}_2'$ are given by:

$$
\begin{aligned}
\mathbf{b}_1' &= \mathbf{v}_1 + \frac{1}{3}[(\mathbf{v}_2 - \mathbf{v}_1) \cdot \mathbf{t}_1] \cdot \mathbf{t}_1, \\
\mathbf{b}_2' &= \mathbf{v}_2 + \frac{1}{3}[(\mathbf{v}_1 - \mathbf{v}_2) \cdot \mathbf{t}_2] \cdot \mathbf{t}_2.
\end{aligned}
\tag{13}
$$

The points $\mathbf{v}_1, \mathbf{b}_1', \mathbf{b}_2', \mathbf{v}_2$ are chosen as control points for each boundary edge and $\mathbf{v}_1, \mathbf{b}_1, \mathbf{b}_2, \mathbf{v}_2$ for each inner edges. A threshold value φ for the angle $\angle \mathbf{v}_1 \mathbf{v}_2 \mathbf{v}_3$ is provided to preserve boundary shape features. If the angle $\angle \mathbf{v}_1 \mathbf{v}_2 \mathbf{v}_3 \leq \varphi$, fairing will not be implemented. For example, Given an angle threshold $\varphi = 90°$, control points $\mathbf{v}_1, \mathbf{b}_1, \mathbf{b}_2', \mathbf{v}_2$ are chosen for the boundary edge e_{12} if the angle at vertex $\mathbf{v}_1 \leq 90°$. Similarly, $\mathbf{v}_1, \mathbf{b}_1', \mathbf{b}_2, \mathbf{v}_2$ are chosen if the angle at vertex $\mathbf{v}_2 \leq 90°$. Iterating each edge in a mesh model, edge control points for each Bézier patch can be obtained. Notice that the angle threshold φ here can be adjusted to different mesh models to obtain better results.

For 3D surface mesh models, vertices and its normals are employed to construct edge control points as proposed in [39]. As demonstrated in Fig. 2c, e_{12} is an arbitrary edge and $\mathbf{n}_1, \mathbf{n}_2$ are the normals for the vertices $\mathbf{v}_1, \mathbf{v}_2$. Control points $\mathbf{b}_1', \mathbf{b}_2'$ are defined by the projection of trisection points $\mathbf{b}_1, \mathbf{b}_2$ of the edge e_{12} onto the corresponding normal plane and can be formulated as:

$$
\begin{aligned}
\mathbf{b}_1' &= \frac{1}{3}\{2\mathbf{v}_1 + \mathbf{v}_2 - [(\mathbf{v}_2 - \mathbf{v}_1) \cdot \mathbf{n}_1] \cdot \mathbf{n}_1\}, \\
\mathbf{b}_2' &= \frac{1}{3}\{\mathbf{v}_1 + 2\mathbf{v}_2 - [(\mathbf{v}_1 - \mathbf{v}_2) \cdot \mathbf{n}_2] \cdot \mathbf{n}_2\}.
\end{aligned}
\tag{14}
$$

The Bézier control points for each mesh edge are obtained after the implementation of Eqs. 13-14. For arbitrary triangular or quadrilateral elements, we can find all their edge Bézier control points which are regarded as the boundary control points of triangular Bézier patch or tensor-product Bézier patch. Next, the inner control points of different Bézier patches could be constructed based on the known boundary control points.

3.2 Triangular Bézier Inner Control Points Construction

While three boundary control polygons of a triangular Bézier patch are given, the inner control points can be constructed by utilizing a mask of the form [20]:

$$
\begin{array}{ccccccc}
 & & & \alpha & & & \\
 & & \beta & & \beta & & \\
 & \beta & & \bullet & & \beta & \\
\alpha & & \beta & & \beta & & \alpha
\end{array}
\tag{15}
$$

with $3\alpha + 6\beta = 1$. This kind of triangular Bézier patches is also called triangular permanence patches [20]. The inner control point $\mathbf{b}_{111}$ of a cubic triangular Bézier patch could be defined by utilizing the above mask as:

$$
\mathbf{b}_{111} = \alpha(\mathbf{b}_{300} + \mathbf{b}_{030} + \mathbf{b}_{003}) + \beta(\mathbf{b}_{210} + \mathbf{b}_{120} + \mathbf{b}_{021} + \mathbf{b}_{012} + \mathbf{b}_{102} + \mathbf{b}_{201}).
\tag{16}
$$

A quadratic precision property could be obtained by choosing $\alpha = -\frac{1}{6}$ [20]. Then the above formula can be rewritten as:

$$
\begin{aligned}
V &= \frac{1}{3}(\mathbf{b}_{300} + \mathbf{b}_{030} + \mathbf{b}_{003}), \\
E &= \frac{1}{6}(\mathbf{b}_{210} + \mathbf{b}_{120} + \mathbf{b}_{021} + \mathbf{b}_{012} + \mathbf{b}_{102} + \mathbf{b}_{201}), \\
\mathbf{b}_{111} &= E + (E - V)/2.
\end{aligned}
\tag{17}
$$

It is noticed that the above equation is equal to that given in [39]. Therefore, the cubic triangular Bézier patch built with $\alpha = -\frac{1}{6}$ is a curved PN triangle. $\alpha = -\frac{1}{6}$ is also used in this paper.

3.3 Tensor-product Bézier Inner Control Points Construction

Tensor-product Bézier patches are established over quadrilateral elements. Given boundary control points of a tensor-product Bézier patch, the inner control points could be generated by introducing a discrete Coons method presented in [20], which was also employed as the first step to build good parametrization of a computational domain for isogeometric analysis [43].

Given four boundary control points $\mathbf{b}_{0j}, \mathbf{b}_{nj}, \mathbf{b}_{i0}, \mathbf{b}_{im}, i = 0, 1, \cdots, n, j = 0, 1, \cdots, m$, the interior control points $\mathbf{b}_{ij}, 0 < i < n, 0 < j < m$ are defined by the discrete Coons method as:

$$
\mathbf{b}_{ij} = (1 - \frac{i}{n})\mathbf{b}_{0j} + \frac{i}{n}\mathbf{b}_{nj} + (1 - \frac{j}{m})\mathbf{b}_{i0} + \frac{j}{m}\mathbf{b}_{im} - \begin{bmatrix} 1 - i/n & i/n \end{bmatrix} \begin{bmatrix} \mathbf{b}_{00} & \mathbf{b}_{0m} \\ \mathbf{b}_{n0} & \mathbf{b}_{nm} \end{bmatrix} \begin{bmatrix} 1 - j/m \\ j/m \end{bmatrix}
\tag{18}
$$

There also exists a mask to construct permanence patches [20]. The discrete Coons method is used to generate inner control points of a tensor-product Bézier patch in this work. Figures 3a and 3d illustrate a rotated letter 'g' plane mesh model and a car body 3D surface mesh model, both of which are constituted with few triangular elements and a bunch of quadrilateral elements. The mixed Bézier models are constructed

(a) Unstructured mesh | (b) Mixed Bézier mesh | (c) Bézier control points

(d) Unstructured mesh | (e) Mixed Bézier mesh | (f) Bézier control points
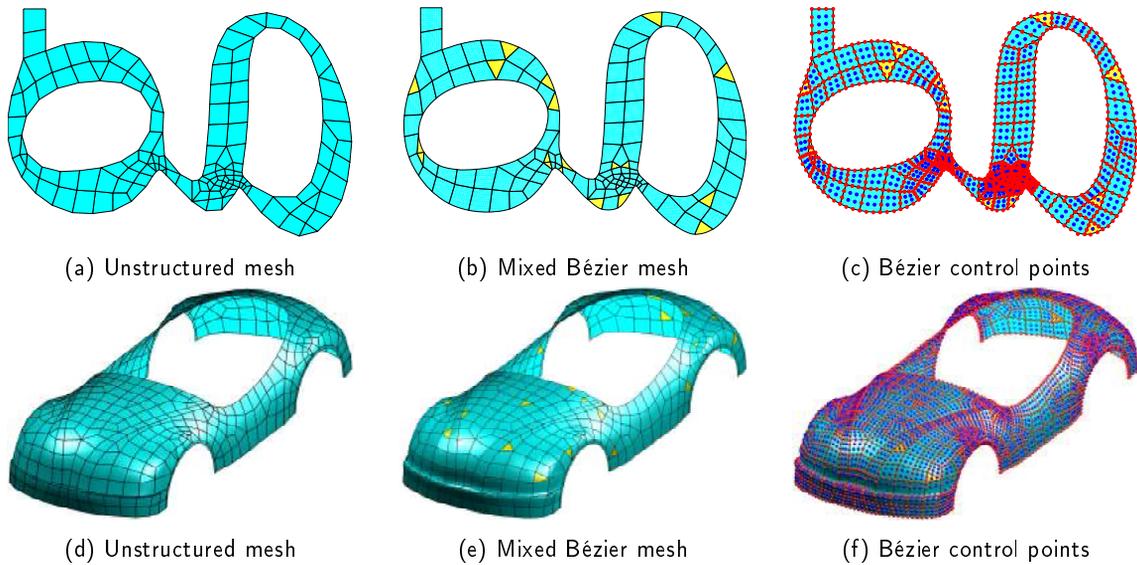
Figure 3: Rotated letter 'g' plane model and three-dimensional car body surface model. (a) and (d) denote plane mesh model and three-dimensional surface mesh model; (b) and (e) denote bi-cubic rectangular Bézier elements and cubic triangular Bézier elements built from the mesh models; (c) and (f) denote the Bézier control nets of the corresponding Bézier models.

from the unstructured meshes as shown in Figs. 3b and 3e. Figures 3c and 3f present the distributions of the constructed Bernstein-Bézier mesh. Yellow and cyan elements represent triangular and rectangular Bézier elements. Red and blue solid dots (spheres) represent boundary control points and inner control points, respectively. As illustrated in Fig 3, it's easy to see the boundaries of plane mixed Bézier mesh model are smooth than that of the initial unstructured mesh model. Note that for plane Bézier mesh the boundaries are G^1 continuous except on some corner points, and for surface Bézier mesh we can achieve C^0 continuity across the common boundaries and G^1 continuity on the shared corner vertices of different Bernstein-Bézier elements.

4   REFINEMENT

The accuracy of the finite element methods is directly associated with the mesh generation. Mesh refinement is a crucial step to improve the accuracy of the numerical results and verify the finite element models. Meanwhile, it is an important tool to investigate the convergence behavior of the models. There are two main mesh refinement techniques: h-refinement and p-refinement that are normally used. The so-called h-refinement is the process of reducing the element size successively to get finer meshes and p-refinement is increasing the element order without changing the mesh structure. Next, we will show that h- and p-refinement strategies can be easily achieved on the mixed Bernstein-Bézier patches.

We first consider the uniform h-refinement technique. For tensor-product Bézier patches, it can be exactly divided into four sub-patches by using knot insertion algorithms [19]. Given the control points $\{\mathbf{b}_{i,j}\}_{i,j=0}^n$ of a tensor-product Bézier patch, the surface point at the parameter (u_0, v_0) can be calculated with *de Casteljau*

(a) Refined parametric domain

(b) Refined sub-patches
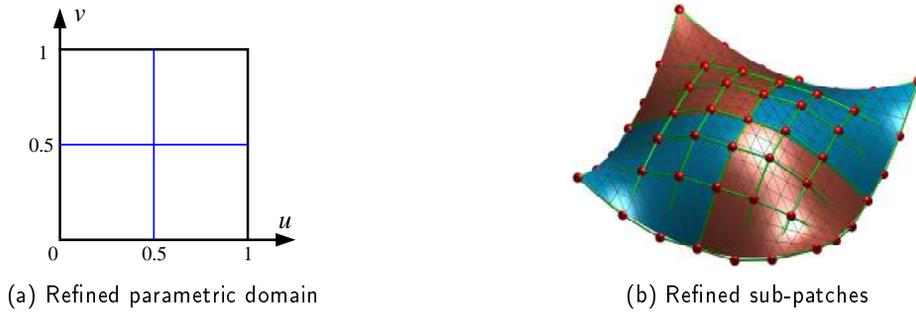
Figure 4: h-refinement of a tensor-product Bézier patch. (a) The refined parametric domain and (b) the corresponding sub-patches.

(a) Parameter subdivision 1   (b) Refined sub-patches   (c) Parameter subdivision 2   (d) Refined sub-patches
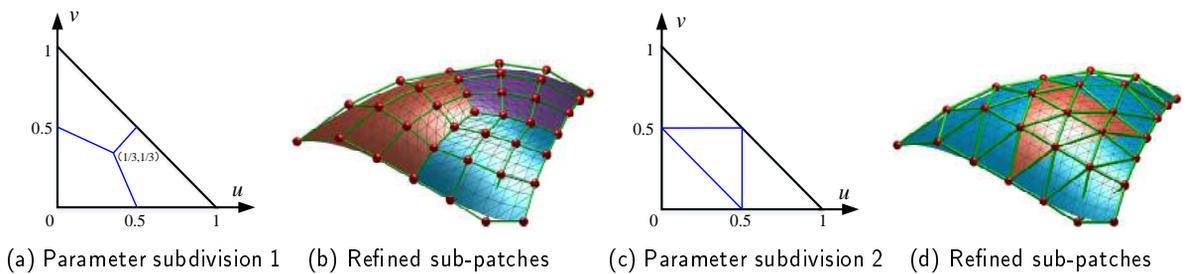
Figure 5: Two approaches in the h-refinement of a triangular Bézier patch. (a) and (c) The subdivision in parametric domain, (b) and (d) the obtained sub-patches from the initial triangular Bézier patch.

algorithm as

$$\mathbf{b}_{i,j}^{k,l} = \begin{cases} \mathbf{b}_{i,j}, & k = l = 0 \\ (1-u)\mathbf{b}_{i,j}^{k-1,0} + u\mathbf{b}_{i+1,j}^{k-1,0}, & k = 1, 2, \cdots, n; l = 0 \\ (1-v)\mathbf{b}_{0,j}^{m,l-1} + v\mathbf{b}_{0,j+1}^{m,l-1}, & k = n; l = 1, 2, \cdots, n. \end{cases}$$

The recursive vertex $\mathbf{b}_{0,j}^{n,n}$ denotes the calculated surface point. This algorithm also presents the control points of four sub-patches generated by the isoparametric curves $u = u_0, v = v_0$. $\mathbf{b}_{0,0}^{k,l}, \mathbf{b}_{0,k}^{n-k,l}, \mathbf{b}_{k,0}^{l,n-k}$ and $\mathbf{b}_{k,l}^{n-l,n-k}$ denote the control points of the four sub-patches with $k, l = 1, 2, \cdots, n$. As shown in Fig. 4, we use parameter $(0.5, 0.5)$ to subdivide the tensor-product Bézier patch, given in Fig. 1, into four sub-patches.

Two strategies are employed to refine the triangular Bézier patch. In the first strategy, we exactly converted the triangular Bézier patch into three tensor-product Bézier patches. Let D^l denote the domain of the l-th sub-rectangular Bézier patch, the control points $\mathbf{b}_{i,j}$ of which can be deduced by

$$\mathbf{T}_{u,v,w \in D^l}(u,v,w) = \sum_{i=0}^{n} \sum_{j=0}^{n} \mathbf{b}_{i,j} B_i^n(s) B_j^n(t). \tag{19}$$

This conversion technique is provided by Hu [23] and also verified by Lasser [31]. In the second strategy, a triangular Bézier patch is divided into four triangular Bézier patches by enforcing the coincidence at the selected collocation points [16]. Let $\triangle^l \subset \triangle$ denote the l-th sub-triangular Bézier patch. Enforcing the

(a) Cubic mixed Bézier   (b) Quartic mixed Bézier   (c) Refined mesh 1   (d) Refined mesh 2
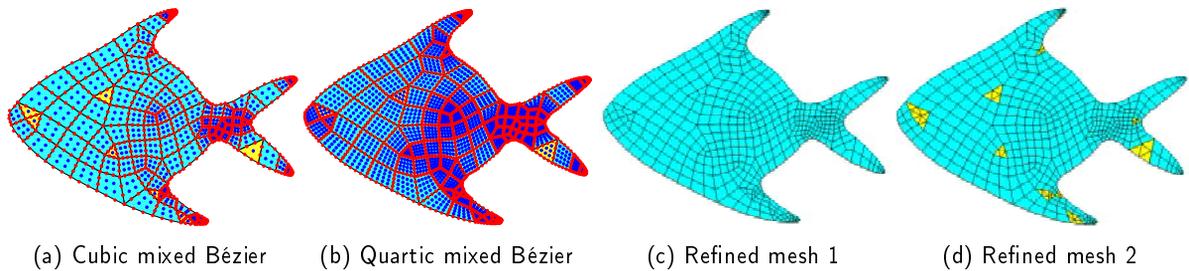
Figure 6: h- and p-refinement of a mixed Bézier model with fish shape. (a)Fish-shaped model consisting of 14 triangular Bézier patches and 106 rectangular Bézier patches, (b)p-refinement of the initial model with twice degree elevation operations, (c) and (d)h-refinement with different strategies on the subdivision of triangular Bézier patches.

surface coincidence at the selected collocation points (u, v, w), yields

$$\sum_{i+j+k=n} \mathbf{b}_{ijk} B_{ijk}^n(u, v, w) = \sum_{i+j+k=n} \mathbf{b}_{ijk}^l B_{ijk}^n(u^l, v^l, w^l), \tag{20}$$

where (u, v, w) and (u^l, v^l, w^l) denote the parameters of the same physical points associated with $\triangle$ and $\triangle^l$. Fig. 5 illustrates the subdivision of the triangular Bézier patch given in fig. 1 with these two techniques. It should be noticed that the adaptive h-refinement of a mixed Bézier model is completely possible with the conversion algorithms between rectangular and triangular Bézier patch [44, 31, 22].

The p-refinement can be easily achieved by using the degree elevation algorithms of rectangular and triangular Bézier patches which are the fundamental algorithms of Bernstein-Bézier patches and are not described here [19]. It is noteworthy that the p-refinement strategy shows great superiority of Bernstein-Bézier elements on flexibly elevating degrees without changing the geometries compared with other higher order finite elements.

As an example, Figure 6 describes the uniform h-refinement and p-refinement of a fish-shaped model built with mixed cubic Bézier patches as given in 6a. Figure 6b shows the surface and control points of the model after twice degree elevation from the first figure. Figures 6c-6d present the uniform h-refinement of the model with two different strategies on the subdivision of triangular Bézier patches.

5   FINITE ELEMENT FORMULATIONS

Reissner-Mindlin theory assumes that any cross section of the undeformed structures remains straight but not perpendicular to the middle surface of the plate or shell during the deformation. It can be applied to analyze both thick and thin plate and shell and only C^0-continuity should be satisfied, thus becomes a common hypothesis in structural finite element analysis. In this section, we present a brief deduction of the finite element formulations based on the Reissner-Mindlin plate and shell theory in the context of elastomechanics. Here the control points of the Bézier elements are treated as the nodes in the finite element method. With the solutions on control points, we can easily compute the corresponding fields solutions on the surfaces based on the Eqs. 8 and 12.

5.1 Plate Model

In Mindlin plate theory [47], the displacement can be expressed by using three independent variables, transverse deflection w and rotations θ_x, θ_y. Therefore, displacement field can be directly written as:

$$\mathbf{u} = \begin{bmatrix} w & \theta_x & \theta_y \end{bmatrix}^T. \tag{21}$$

Then the nodal displacements $\mathbf{d}^e$ of any Bézier elements can be expressed by:

$$\mathbf{d}^e = \begin{bmatrix} w_1 & \theta_{x1} & \theta_{y1} & \cdots & w_{ncp} & \theta_{xncp} & \theta_{yncp} \end{bmatrix}^T, \tag{22}$$

where ncp denotes the number of the control points of a Bézier element. Motivated by isogeometric analysis scenario, Bernstein-Bézier basis functions are used to be the shape functions in FEA. By inserting the basis functions in Eqs. 8 and 12 into the Eq. 21, then the displacement fields in each Bézier element can be rewritten as:

$$\mathbf{u}^e = \begin{bmatrix} w \\ \theta_x \\ \theta_y \end{bmatrix} = \sum_{i=1}^{ncp} \begin{bmatrix} B_i & 0 & 0 \\ 0 & B_i & 0 \\ 0 & 0 & B_i \end{bmatrix} \begin{bmatrix} w_i \\ \theta_{xi} \\ \theta_{yi} \end{bmatrix} = \mathbf{R}^e \mathbf{d}^e, \tag{23}$$

where $\mathbf{R}$ denotes the shape function matrix. For the concisely writing, B_i are used to represent $B_{ijk}^n(u, v, w)$ in the Eq. 8 and $B_{i,n}(u)B_{j,m}(v)$ in the Eq. 12.

Pseudo-strain ϵ_p could be computed by the differential of displacement $\mathbf{u}$ and given as:

$$\epsilon_p = \begin{bmatrix} \partial\theta_y/\partial x \\ -\partial\theta_x/\partial y \\ \partial\theta_y/\partial y - \partial\theta_x/\partial x \\ \theta_x - \partial w/\partial y \\ \theta_y + \partial w/\partial x \end{bmatrix} = [\partial]\mathbf{u} = \mathbf{B}\mathbf{d}^e, \tag{24}$$

where the matrix $\mathbf{B}$ is deduced as $\mathbf{B} = [\mathbf{B}_1 \ \mathbf{B}_2 \ \cdots \ \mathbf{B}_i \ \cdots \ \mathbf{B}_{ncp}]$ with

$$\mathbf{B}_i = \begin{bmatrix} 0 & 0 & B_{i,x} \\ 0 & -B_{i,y} & 0 \\ 0 & -B_{i,x} & B_{i,y} \\ -B_{i,y} & B_i & 0 \\ B_{i,x} & 0 & B_i \end{bmatrix}. \tag{25}$$

The pseudo-stress $\boldsymbol{\sigma}_p$, defined through the Hooke's law, can be expressed as:

$$\boldsymbol{\sigma}_p = \mathbf{D}\epsilon_p = \mathbf{D}\mathbf{B}\mathbf{d}^e, \tag{26}$$

where the matrix $\mathbf{D}$ is defined by

$$\mathbf{D} = \begin{bmatrix} \mathbf{D}_b & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_s \end{bmatrix} \tag{27}$$

with

$$\mathbf{D}_b = \frac{Eh^3}{12(1-\nu^2)} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix}, \mathbf{D}_s = \frac{5Eh}{12(1+\nu)} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix},$$

in which h denotes the thickness of the plate. E and ν are Young's modulus and Poisson ratio.

Considering the static bending and free vibration of the Reissner-Mindlin plate, the variational form of the equilibrium equation in the context of elastodynamics can be written as:

$$\int_\Omega \delta\boldsymbol{\epsilon}_p^T \boldsymbol{\sigma}_p d\Omega + \int_\Omega \delta\mathbf{u}^T \rho\ddot{\mathbf{u}} d\Omega = \int_\Omega \delta\mathbf{u}^T \mathbf{b} d\Omega - \int_\Gamma \delta\mathbf{u}^T \bar{\mathbf{t}} d\Gamma \tag{28}$$

where $\mathbf{b}, \bar{\mathbf{t}}$ and ρ denote the body force, traction force and density. Substituting the Eqs. 21-26 into the Eq. 27, the finite element equations derived for static bending problems are

$$\mathbf{Ku} = \mathbf{F}, \tag{29}$$

and for free vibration problems are

$$\mathbf{M\ddot{u}} + \mathbf{Ku} = \mathbf{0}, \tag{30}$$

where

$$\mathbf{K} = \int_\Omega \mathbf{B}^T \mathbf{D} \mathbf{B} d\Omega, \mathbf{M} = \int_\Omega \mathbf{R}^T \mathbf{m} \mathbf{R} d\Omega,$$

$$\mathbf{m} = \rho \begin{bmatrix} h & 0 & 0 \\ 0 & h^3/12 & 0 \\ 0 & 0 & h^3/12 \end{bmatrix}.$$

To calculate the integration over elements, Gauss-Legendre quadrature rule [34] is used for tensor-product Bézier patches and a collapsed Gaussian quadrature rule is used for triangular Bézier patches by collapsing the square to a triangle [33].

5.2 Shell Model

In this section, finite element formulations of Reissner-Mindlin shell are derived from degenerated three-dimensional solid and for more shell theory details we refer to [6]. It is much easier for us to impose rotational constraints based on Reissner-Mindlin theory.

The position vector of a given point on the shell is defined by:

$$\mathbf{S}(\xi, \eta, \zeta) = \sum_{i=1}^{ncp} B_i(\xi, \eta)\mathbf{P}_i + \frac{h}{2}\zeta\mathbf{n}, \tag{31}$$

where $\mathbf{P}_i$ denote the control points of the shell middle surface. $\mathbf{n}$ denote the normal vector of the surface point projected from the corresponding control point. ζ is the parameter along the thickness direction. In contrast with Mindlin plate problem, there are 6 DOFs of each control point in Mindlin shell elements, thus the nodal displacement can be expressed by $\mathbf{d}^e = [\mathbf{d}_1^e \ \mathbf{d}_2^e \ \cdots \mathbf{d}_i^e \cdots \ \mathbf{d}_{ncp}^e]^T$ where $\mathbf{d}_i^e$ is given as:

$$\mathbf{d}_i^e = \begin{bmatrix} u_i & v_i & w_i & \theta_{xi} & \theta_{yi} & \theta_{zi} \end{bmatrix}. \tag{32}$$

The displacement vector can be expressed by the interpolation of Bernstein polynomials as:

$$\mathbf{u}^e(\xi, \eta, \zeta) = \sum_{i=1}^{ncp} B_i(\xi, \eta)(\mathbf{u}_i^e + \frac{h}{2}\zeta\theta_i \times \mathbf{n}^e) = \mathbf{R}^e\mathbf{d}^e. \tag{33}$$

The definition of the normal vectors is crucial in the analysis of shell. There are two main methods to describe the normal vectors: exact normal and interpolated normal. Exact normal vectors are used in this paper. It is helpful to notice that using Greville abscissae coordinates to calculate normal vectors is considered to be an advisable way to compromise between accuracy and efficiency [1]. A local orthonormal basis is constructed to transform the variables between global basis and local basis. For any point on the shell, normal vector can be defined as:

$$\mathbf{n} = \frac{\mathbf{S},_\xi \times \mathbf{S},_\eta}{||\mathbf{S},_\xi \times \mathbf{S},_\eta||}. \tag{34}$$

Defining two vectors $\mathbf{t}_1$ and $\mathbf{t}_2$ by

$$\mathbf{t}_1 = \frac{\mathbf{S},_\xi}{||\mathbf{S},_\xi||}, \quad \mathbf{t}_2 = \mathbf{n} \times \mathbf{t}_1, \tag{35}$$

then the transformation matrix from global to local basis is given by $\mathbf{q} = [\mathbf{t}_1 \ \mathbf{t}_1 \ \mathbf{n}]$. The quantities in the local system can be converted into a global system, e.g. $[\epsilon^g] = \mathbf{q}[\epsilon^l]\mathbf{q}^T$, where $[\epsilon]$ denotes the tensor form of the strain ϵ. The superscript g, l represent the global and local systems. Using Voigt notation, the global strain matrix is given by

$$\epsilon^g = \mathbf{B}\mathbf{d}^e \tag{36}$$

with matrix $\mathbf{B}$ defined by

$$\mathbf{B} = \begin{bmatrix} \mathbf{B}_1^u & \mathbf{B}_1^\theta & \cdots & \mathbf{B}_i^u & \mathbf{B}_i^\theta & \cdots & \mathbf{B}_{ncp}^u & \mathbf{B}_{ncp}^\theta \end{bmatrix},$$

$$\mathbf{B}_i^u = \begin{bmatrix} B_{i,x} & 0 & 0 \\ 0 & B_{i,y} & 0 \\ 0 & 0 & B_{i,z} \\ B_{i,y} & B_{i,x} & 0 \\ 0 & B_{i,y} & B_{i,y} \\ B_{i,z} & 0 & B_{i,x} \end{bmatrix}, \mathbf{B}_i^\theta = \begin{bmatrix} 0 & (\bar{B}_i n_{zi}),_x & -(\bar{B}_i n_{yi}),_x \\ -(\bar{B}_i n_{zi}),_y & 0 & (\bar{B}_i n_{xi}),_y \\ (\bar{B}_i n_{yi}),_z & -(\bar{B}_i n_{xi}),_z & 0 \\ -(\bar{B}_i n_{zi}),_x & (\bar{B}_i n_{zi}),_y & (\bar{B}_i n_{xi}),_x - (\bar{B}_i n_{yi}),_y \\ (\bar{B}_i n_{yi}),_y - (\bar{B}_i n_{zi}),_z & -(\bar{B}_i n_{xi}),_y & (\bar{B}_i n_{xi}),_z \\ (\bar{B}_i n_{yi}),_x & (\bar{B}_i n_{zi}),_z - (\bar{B}_i n_{xi}),_x & -(\bar{B}_i n_{yi}),_z \end{bmatrix}$$

with $\bar{B}_i = \frac{h}{2}\zeta B_i$. n_{xi}, n_{yi}, n_{zi} denote three components of the normal vector $\mathbf{n}_i$. The global stress is defined by

$$\sigma^g = \mathbf{D}^g\epsilon^g. \tag{37}$$

Substituting the Eqs. 32-36 into the Eq. 28, the general finite element equations as presented in the Eqs. 29-30 can be obtained. Two Gaussian quadrature points are selected along the thickness direction for integration.

6 NUMERICAL EXAMPLES

6.1 Clamped Square Plate Under a Transverse Load

In the plate bending problem, a unit square plate $[0, 1]^2$ with all sides fully clamped is investigated as shown in Fig. 7. The square plate is submitted to a prescribed transverse load $f(x, y)$ expressed as:

$$f(x, y) = \frac{E}{12(1 - \nu^2)}[12y(y - 1)(5x^2 - 5x + 1) \times (2y^2(y - 1)^2 + x(x - 1)(5y^2 - 5y + 1))$$
$$+ 12x(x - 1)(5y^2 - 5y + 1)(2x^2(x - 1)^2 + y(y - 1)(5x^2 - 5x + 1))],$$

(a) Mixed Bézier model (b) Deflection w (c) Rotation θ_x (d) Rotation θ_y

(e) Bending moment M_x (f) Bending moment M_{xy} (g) Shear force Q_x (h) Shear force Q_y
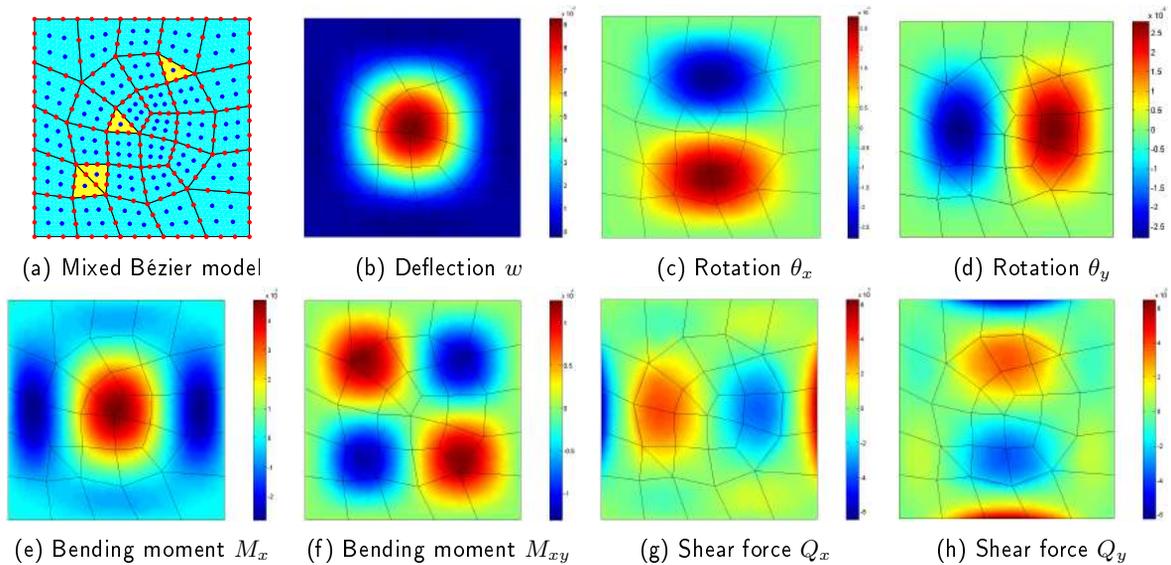
Figure 7: Clamped unit square plate built with 30 bi-cubic rectangular Bézier elements and 4 cubic triangular Bézier elements. (a) The constructed mixed Bézier model and the visualized field solutions of (b)-(e) the deflection w, bending moment M_x, rotation θ_x and shear force Q_x.

where Young's modulus $E = 10.92 \times 10^6$, Poisson ratio $\nu = 0.3$. The thickness-span ratio of the plate is 0.1. The analytical solution of deflection $w(x,y)$ and rotations $\theta_x(x,y), \theta_y(x,y)$ is given in [11] as follows:

$$
\begin{aligned}
w(x,y) =& \frac{1}{3}x^3(x-1)^3 y^3(y-1)^3 - \frac{2t^2}{5(1-\nu)} \\
& \times [y^3(y-1)^3 x(x-1)(5x^2-5x+1) + x^3(x-1)^3 y(y-1)(5y^2-5y+1)], \\
\theta_x(x,y) =& x^3(x-1)^3 y^2(y-1)^2(2y-1), \\
\theta_y(x,y) =& y^3(y-1)^3 x^2(x-1)^2(2x-1).
\end{aligned}
$$

The exact solutions of bending moments and shear forces can be derived from the above formulations as [27]:

$$
M_x(x,y) = \frac{Et^3}{6(1-\nu^2)}[y^3(y-1)^3(x-x^2)(5x^2-5x+1) + \nu(x^3(x-1)^3(y-y^2)(5y^2-5y+1))],
$$

$$
Q_x(x,y) = -\frac{Et^3}{6(1-\nu^2)}[y^3(y-1)^3(20x^3-30x^2+12x-1) + 3y(y-1)(5y^2-5y+1)x^2(x-1)^2(2x-1)].
$$

The studies of this problem can also be found in [13, 12] with the isogeometric method. The unstructured mesh of square plate consists of 30 quadrilateral elements and 4 triangular elements, over which we first build bi-cubic tensor-product Bézier patches and cubic triangular Bézier patches as presented in Fig. 7a. The deflection w, bending moment M_x, rotation θ_x and shear force Q_x are plotted based on the constructed mixed-Bézier patches as shown in Figs. 7b-7h.

To study the behavior of convergence, the approximation error of deflection w is defined as

$$
E_w = \frac{||w_{ex} - w_h||_2}{||w_{ex}||_2} = \sqrt{\frac{\sum_{i=1}^{elem}\sum_{j=1}^{nq}(w_{ex}^{i,j} - w_h^{i,j})^2}{\sum_{i=1}^{elem}\sum_{j=1}^{nq}(w_{ex}^{i,j})^2}},
$$

(a)

(b)

(c)

(d)
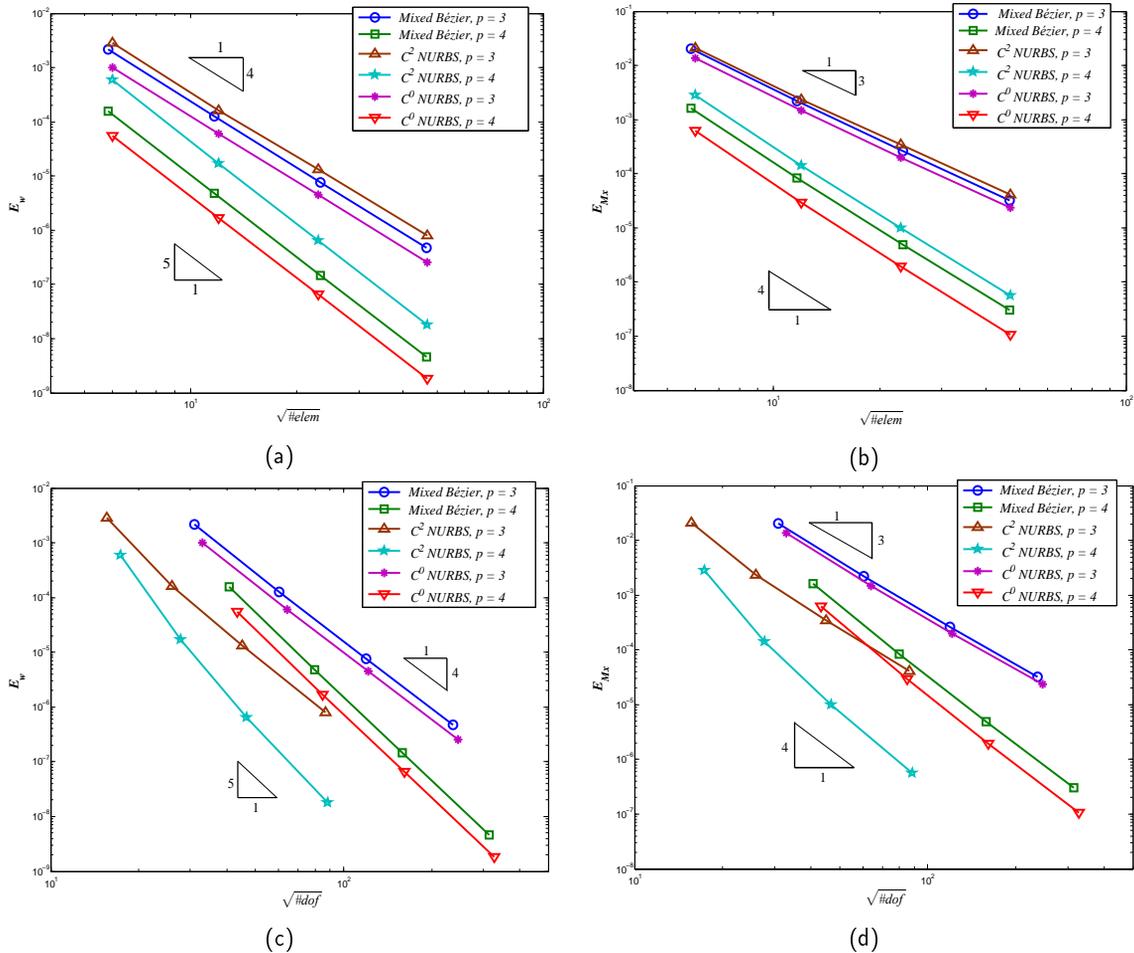
Figure 8: Clamped unit square plate under a transverse load. The approximation error of deflection and bending moment M_x is plotted against the square root of the number of elements ($\sqrt{\#elem}$) in (a)-(b) and the square root of the number of DOFs ($\sqrt{\#dof}$) in (c)-(d).

where w_{ex} and w_h denote the exact results and numerical results. $elem$ and nq denote the number of elements and the number of quadrature points on each element. Similarly, the approximation error of other variables can be defined. As depicted in Fig. 8, we investigate the approximation error of deflection w and bending moment M_x plotted against the square root of the total number of elements ($\sqrt{\#elem}$) and that of degrees of freedom ($\sqrt{\#dof}$). Bi-cubic and bi-quartic NURBS models with C^2-continuity and C^0-continuity across the inner element boundaries are analyzed by using the isogeometric method for comparison. The h- and p-refinement strategies described in Section 4 are introduced here to refine the mesh and elevate the degree of the constructed mixed-Bézier mesh. It can be found that all these three models can converge with the rate of $p+1$ on deflection and the rate of p on bending moment M_x. As given in Figs. 8a and 8b, plotted against with $\sqrt{\#elem}$, C^0-continuity NURBS models offer most accurate results and the following is the mixed Bézier mesh. In this case the C^2-continuity NURBS models only occupy a weak position because it contains a small number of control points compared with another two models. In contrast, the C^2-continuity NURBS models
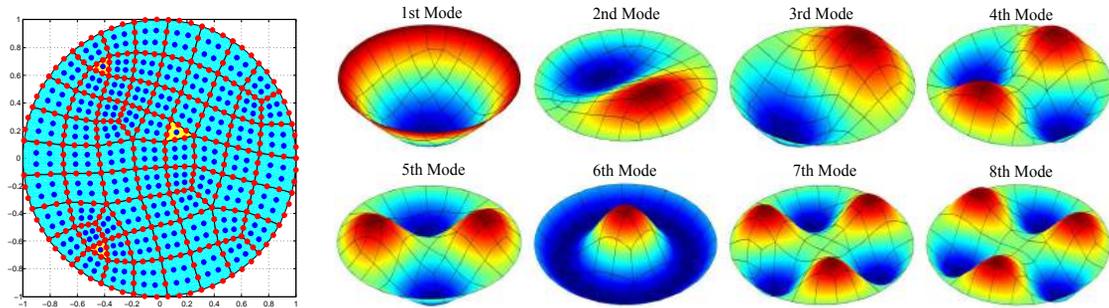
Figure 9: The vibration analysis of simply supported circular plate. Left: mixed Bézier mesh and control points of the plate with 3 triangular Bézier patches and 67 rectangular Bézier patches; Right: the first eighth mode shapes of the plate.

present a great superiority on accuracy when we plot the convergence against the degrees of freedom $\sqrt{\#dof}$ as illustrated in Figs. 8c and 8d, because the higher continuity can significantly reduce the total number of the control points and this is the greatest virtue of IGA.

6.2 Free Vibration of a Simply Supported Circular Plate

The free vibration of a simply supported circular plate with radius $R = 1m$ is studied in this problem. The material parameters Young's modulus $E = 200 \times 10^9 N/m^2$, Poisson ratio $\nu = 0.3$, and mass density $\rho = 8000 kg/m^3$.

Figure 9 shows the cubic mixed Bézier mesh and the first eighth mode shapes of the plate. Here the mixed Bézier mesh consists of 3 triangular Bézier patches and 67 rectangular Bézier patches. The first 10th frequency parameters $\varpi = (\omega R^2)\sqrt{\rho h/D_0}$ for this simply supported circular plate are calculated as listed in Table 1. For the comparison, the exact solutions and the results from NURBS-based IGA with 64 bi-cubic elements are also listed here [32, 46]. It is obvious that the results of frequency parameters agree very well with exact solutions and NURBS-based IGA results.

Table 1: Comparison of frequency parameters, $\varpi = (\omega R^2)\sqrt{\rho h/D_0}$, for simply supported circular plate.

h/a	Method	Mode number									
		1	2	3	4	5	6	7	8	9	10
0.1	Our method	4.895	13.518	13.518	24.340	24.337	28.260	37.017	37.019	44.744	44.751
	NURBS-IGA [46]	4.894	13.519	13.519	24.355	24.362	28.296	37.129	37.129	45.036	45.036
	Exact [32]	4.894	13.510	13.510	24.313	24.313	28.240	36.962	36.962	44.691	44.691
0.2	Our method	4.779	12.632	12.632	21.721	21.722	25.035	31.612	31.612	37.621	37.624
	NURBS-IGA [46]	4.779	12.631	12.631	21.723	21.727	25.043	31.640	31.640	37.704	37.704
	Exact [32]	4.777	12.620	12.620	21.687	21.687	24.994	31.547	31.547	37.537	37.537

(a) Description and dimensions   (b) Mixed Bézier mesh   (c) Vertical displacement
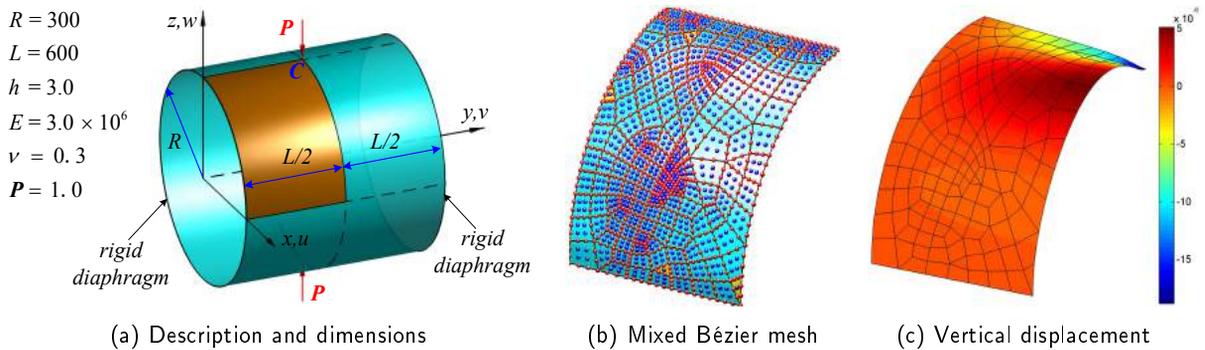
Figure 10: Pinched cylinder. (a)The description and dimensions of the pinched cylinder, (b) the mixed Bézier mesh of one-eighth of the pinched cylinder and (c) the vertical displacement.

6.3 Pinched Cylinder Subjected to a Concentrated Load

Pinched cylinder from the so-called "shell obstacle course" [4] with rigid diaphragm ($u_x = u_z = \theta_y = 0$) is subjected to a pair of radial load **P** at the top and bottom middle position as shown in Fig. 10a. Due to the symmetry property, only one-eighth of the cylinder should be modeled and analyzed. Figure 10b shows the mixed Bézier mesh consisting of 9 triangular Bézier patches and 150 rectangular Bézier patches. The vertical displacement of the shell is presented in Fig. 10c.

The convergence of the vertical displacement at point **C**, against $\sqrt{\#elem}$ and $\sqrt{\#dof}$, is investigated among mixed Bézier meshes, C^2-continuity and C^0-continuity NURBS mesh with different degrees as depicted in Fig. 11. These results are compared with the reference solution $w_C = -1.8248 \times 10^{-5}$ [4]. It can be found that for the same order the results obtained from the mixed Bézier mesh converge slightly faster than that from C^2-continuity NURBS-based IGA in the displacement-$\sqrt{\#elem}$ plot as presented in Fig. 11a, while the trend is inverted in the displacement-$\sqrt{\#dof}$ plot as given in Fig. 11b where the abscissa $\sqrt{\#dof}$ is plotted into log-form for better observation. Our convergence results are slightly higher than the reference results and the similar phenomena are also reported in [1, 26].

6.4 Sink Shell Model Under Uniform Pressure

In this example, we extend the presented method to an industrial application, dealing with a complicated sink shell model. Figure 12 gives the problem description and presents the top view, front view and axonometric view of the model. The top surface of the sink is subjected to a uniformly distributed load with the magnitude of $P = 10N$. Fixed Dirichlet boundary conditions are enforced on the bottom surface. The basic dimensions include length $L = 140cm$, width $W = 80cm$, height $H = 30cm$ and thickness $h = 0.5cm$. The material of the sink is aluminum alloy with $E = 70 \times 10^9 N/m^2, \nu = 0.3$.

Mixed Bézier model is constructed on the FEA mesh with 23 triangles and 1274 quadrangles. For comparison, the FEA solution of z-displacement is obtained in ABAQUS with 23 quadratic triangular elements and 1274 quadratic rectangular elements as shown in Fig. 13a. Figure 13b shows the result of isogeometric analysis based on the constructed Bézier patches. It is easy to find that the result of the our method on the mixed Bézier mesh agrees well with the FEA result on quadratic elements. As presented in Fig. 13c, we also plot the deformation along the section **A-A** defined in Fig. 12. The black dashed line denotes the undeformed section. Red triangles and blue line denote the deformation results obtained by using ABAQUS and presented method. A scaled factor $\alpha = 3 \times 10^5$ is chosen for better visualization. This example shows that the prescribed method in this paper can be applied to the model with the geometric complexity of the industrial level, and
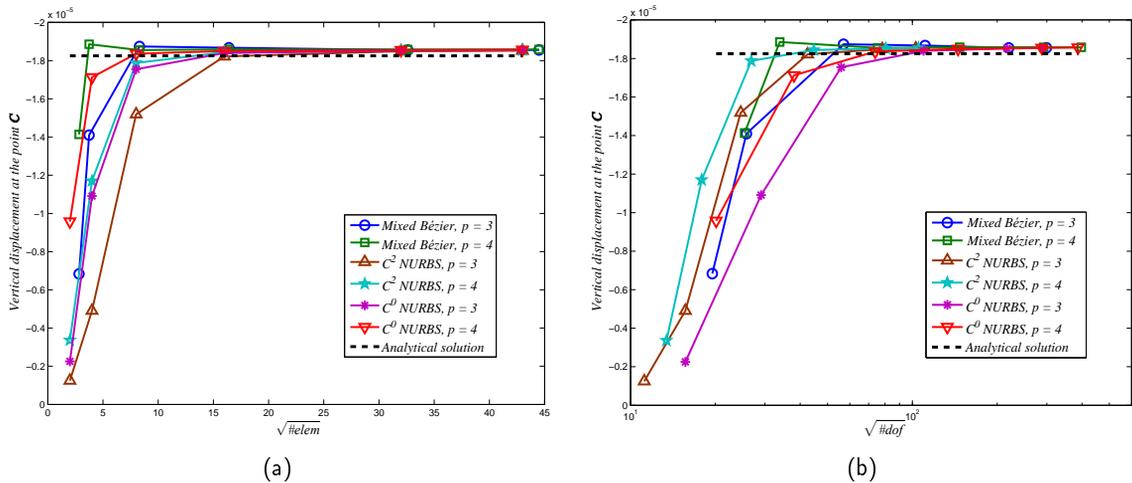
Figure 11: Pinched cylinder: the convergence comparison of the vertical displacement at point C against (a) the square root of the number of the elements ($\sqrt{\#elem}$) and (b) the square root of the number of the DOFs ($\sqrt{\#dof}$).

can acquire similar results in accuracy compared with classical higher-order finite element method.

7   CONCLUSIONS

A novel method has been proposed to realize finite element analysis on unstructured mesh surface by the construction of triangular Bézier patches over triangular elements and tensor-product Bézier patches over quadrilateral elements. The procedures of construction are simple and intuitive. To validate the optimal convergence of the proposed method, we can use general mesh generation software to discretize the initial CAD models to unstructured mesh models which are used to build mixed Bézier patches. The results of the numerical examples including static bending and free vibration of Reissner-Mindlin plate and shell show that this method is feasible for its near-equivalent accuracy and convergence with isogeometric analysis so it provides an alternative for engineering scenarios when higher-order finite elements are required. Also for incomplete mesh models that are often too coarse to give a fair result but need to be further refined, this method can be especially helpful and give more accuracy compared with resorting to the classical linear finite element method. Compared with other higher-order finite elements method, several attractive properties and elegant algorithms like de Casteljau and degree elevation algorithms, can be introduced into FEA and bring competitive advantages on robustness and efficiency when mixed Bernstein-Bézier elements are used.

It should be noticed that only C^0 continuity remains between Bézier patches which means that more control points will be produced after the reconstruction compared with NURBS-based reconstruction which is definitely more complicated and time-consuming. But the mixed Bézier method proposed here shows its powerful adaptability for complex geometry representation. This entitles it the possibility of being applied to extract rectangular and triangular Bézier elements from trimming NURBS or T-spline models and comprise a hybrid as well as pro-calculation model. An even bolder proposition is to generate CAD models directly on the basis of mixed Bézier representation for tight integration between CAD and CAE. This will be interesting and a worthy topic since the trimmed model problem is still a great challenge in isogeometric analysis. Therefore, future works will focus on the trimming problem in IGA and high-continuity mixed Bézier patches construction over existing mesh surface models.
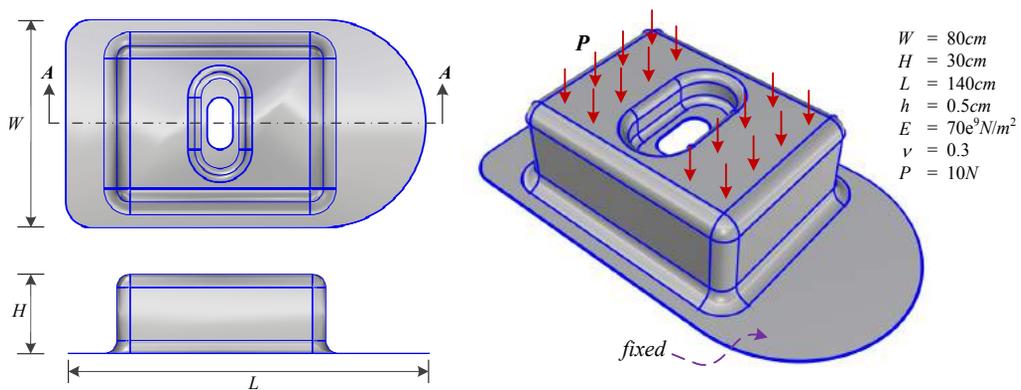
Figure 12: Problem definition of the sink shell model by giving dimensions, material properties and three views of the model: top view, front view, and axonometric view.

(a) ABAQUS results

(b) Our results

(c) Deformation curves
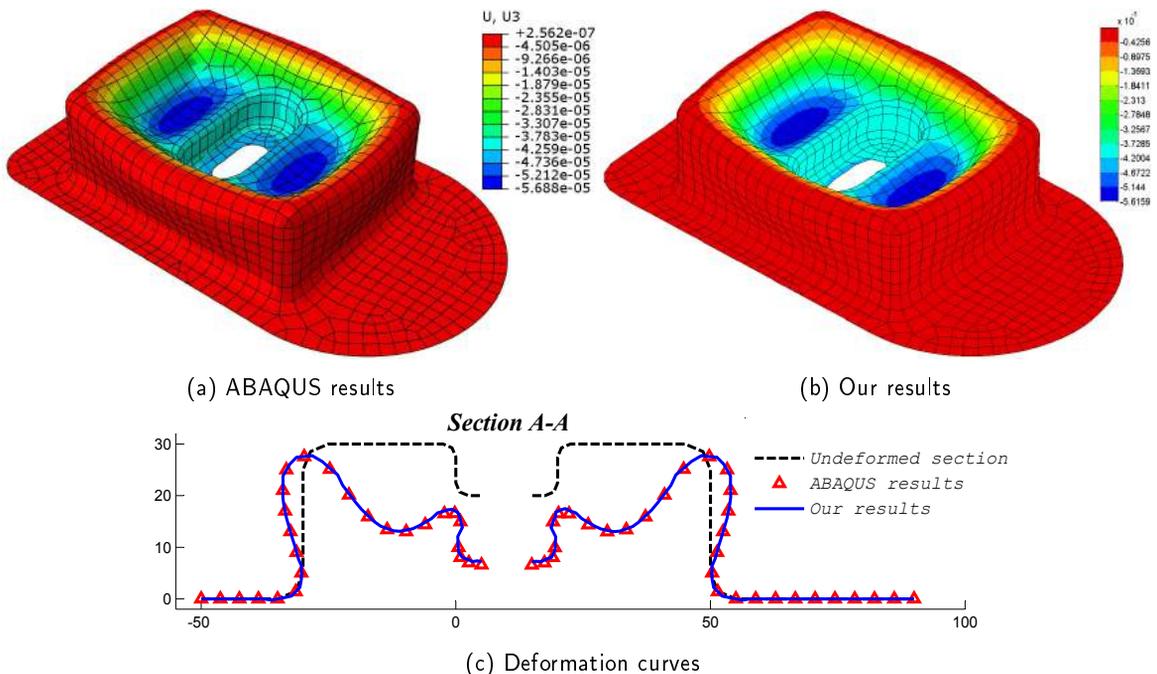
Figure 13: Deflection of the sink shell model. (a) and (b) FEA solution in ABAQUS with 1297 quadratic elements, and our solution with the mesh of 23 cubic triangular Bézier elements and 1274 bi-cubic tensor-product Bézier elements; (c) the comparison of deformation along the section A-A with a scaled factor $\alpha = 3 \times 10^5$.

ACKNOWLEDGEMENTS

ORCID

Xiaoxiao Du http://orcid.org/0000-0002-2324-1005

REFERENCES

[1] Adam, C.; Bouabdallah, S.; Zarroug, M.; Maitournam, H.: Improved numerical integration for locking treatment in isogeometric structural elements. Part II: Plates and shells. Computer Methods in Applied Mechanics and Engineering, 284(279), 106–137, 2015. http://doi.org/10.1016/j.cma.2014.07.020.

[2] Ainsworth, M.; Andriamaro, G.; Davydov, O.: Bernstein-Bézier finite elements of arbitrary order and optimal assembly procedures. SIAM Journal on Scientific Computing, 33(6), 3087–3109, 2011. http://doi.org/10.1137/11082539X.

[3] Ainsworth, M.; Davydov, O.; Schumaker, L.L.: Bernstein-Bézier finite elements on tetrahedral-hexahedral-pyramidal partitions. Computer Methods in Applied Mechanics and Engineering, 304, 140–170, 2016. http://doi.org/10.1016/j.cma.2016.01.021.

[4] Belytschko, T.; Stolarski, H.; Liu, W.K.; Carpenter, N.; Ong, J.S.J.: Stress projection for membrane and shear locking in shell finite elements. Computer Methods in Applied Mechanics and Engineering, 51(1-3), 221–258, 1985. http://doi.org/10.1016/0045-7825(85)90035-0.

[5] Benson, D.; Bazilevs, Y.; Hsu, M.C.; Hughes, T.: Isogeometric shell analysis: the Reissner–Mindlin shell. Computer Methods in Applied Mechanics and Engineering, 199(5-8), 276–289, 2010. http://doi.org/10.1016/j.cma.2009.05.011.

[6] Bischoff, M.; Bletzinger, K.; Wall, W.A.; Ramm, E.: Models and finite elements for thin-walled structures, chap. 3. John Wiley and Sons, Ltd, 2004. http://doi.org/10.1002/9781119176817.ecm2026.

[7] Bogdanovich, A.E.: Three-dimensional variational theory of laminated composite plates and its implementation with Bernstein basis functions. Computer Methods in Applied Mechanics and Engineering, 185(2-4), 279–304, 2000. http://doi.org/10.1016/S0045-7825(99)00263-7.

[8] Bogdanovich, A.E.: Three-Dimensional variational analysis of composite structures using Bernstein polynomial approximations. Report 1. Mechanics of Composite Materials, 37(5-6), 525–538, 2001. http://doi.org/10.1023/A:1014229600159.

[9] Borden, M.J.; Scott, M.A.; Evans, J.A.; Hughes, T.J.R.: Isogeometric finite element data structures based on Bézier extraction of NURBS. International Journal for Numerical Methods in Engineering, 87(1-5, SI), 15–47, 2011. http://doi.org/10.1002/nme.2968.

[10] Chan, J.; Warburton, T.: A short note on a Bernstein-Bézier basis for the pyramid. SIAM Journal on Scientific Computing, 38(4), A2162–A2172, 2015. http://doi.org/10.1137/15M1036397.

[11] Chinosi, C.; Lovadina, C.: Numerical analysis of some mixed finite element methods for Reissner-Mindlin plates. Computational Mechanics, 16(1), 36–44, 1995. http://doi.org/10.1007/BF00369883.

[12] Da Veiga, L.B.; Hughes, T.J.R.; Kiendl, J.; Lovadina, C.; Niiranen, J.; Reali, A.; Speleers, H.: A locking-free model for Reissner-Mindlin plates: Analysis and isogeometric implementation via NURBS and triangular NURPS. Mathematical Models and Methods in Applied Sciences, 25(8), 1519–1551, 2015. http://doi.org/10.1142/S0218202515500402.

[13] Du, X.; Zhao, G.; Wang, W.: Nitsche method for isogeometric analysis of Reissner-Mindlin plate with non-conforming multi-patches. Computer Aided Geometric Design, 35-36, 121–136, 2015. http://doi.org/10.1016/j.cagd.2015.03.005.

[14] Duczek, S.: Numerical analysis of lamb waves using the finite and spectral cell method. International Journal for Numerical Methods in Engineering, 99(1), 26–53, 2014. http://doi.org/10.1002/nme.4663.

[15] Düster, A.; Bröker, H.; Rank, E.: The p-version of the finite element method for three-dimensional curved thin walled structures. International Journal for Numerical Methods in Engineering, 52(7), 673–703, 2001. http://doi.org/10.1002/nme.222.

[16] Engvall, L.; Evans, J.A.: Isogeometric triangular Bernstein-Bézier discretizations: Automatic mesh generation and geometrically exact finite element analysis. Computer Methods in Applied Mechanics and Engineering, 304, 378–407, 2016. http://doi.org/10.1016/j.cma.2016.02.012.

[17] Engvall, L.; Evans, J.A.: Isogeometric unstructured tetrahedral and mixed-element Bernstein-Bézier discretizations. Computer Methods in Applied Mechanics and Engineering, 319, 83–123, 2017. http://doi.org/10.1016/j.cma.2017.02.017.

[18] Farin, G.: Triangular Bernstein-Bézier patches. Computer Aided Geometric Design, 3(2), 83–127, 1986. http://doi.org/10.2514/6.2015-1370.

[19] Farin, G.: Curves and surfaces for CAGD: a practical guide. Morgan Kaufmann Publishers Inc., 2001.

[20] Farin, G.; Hansford, D.: Discrete Coons patches. Computer Aided Geometric Design, 16(7), 691–700, 1999. http://doi.org/10.1016/S0167-8396(99)00031-X.

[21] Farouki, R.T.: The Bernstein polynomial basis: A centennial retrospective. Computer Aided Geometric Design, 29(6), 379–419, 2012. http://doi.org/10.1016/j.cagd.2012.03.001.

[22] Filip, D.J.: Adaptive subdivision algorithms for a set of Bézier triangles. Computer-Aided Design, 18(2), 74–78, 1986. http://doi.org/10.1016/0010-4485(86)90153-3.

[23] Hu, S.M.: Conversion of a triangular Bézier patch into three rectangular Bézier patches. Computer Aided Geometric Design, 13(3), 219–226, 1996. http://doi.org/10.1016/0167-8396(95)00023-2.

[24] Hughes, T.J.R.: The finite element method: linear static and dynamic finite element analysis. Dover Publishcations, 2000. http://doi.org/10.1111/j.1467-8667.1989.tb00025.x.

[25] Jaxon, N.; Qian, X.: Isogeometric analysis on triangulations. Computer-Aided Design, 46(1), 45–57, 2014. http://doi.org/10.1016/j.cad.2013.08.017.

[26] Kang, P.; Youn, S.K.: Isogeometric analysis of topologically complex shell structures. Finite Elements in Analysis and Design, 99, 68–81, 2015. http://doi.org/10.1016/j.finel.2015.02.002.

[27] Kiendl, J.; Auricchio, F.; Da Veiga, L.B.; Lovadina, C.; Reali, A.: Isogeometric collocation methods for the Reissner-Mindlin plate problem. Computer Methods in Applied Mechanics and Engineering, 284, 489–507, 2015. http://doi.org/10.1016/j.cma.2014.09.011.

[28] Kirby, R.C.: Fast simplicial finite element algorithms using Bernstein polynomials. Numerische Mathematik, 117(4), 631–652, 2011. http://doi.org/10.1007/s00211-010-0327-2.

[29] Kirby, R.C.; Thinh, K.T.: Fast simplicial quadrature-based finite element operators using Bernstein polynomials. Numerische Mathematik, 121(2), 261–279, 2012. http://doi.org/10.1007/s00211-011-0431-y.

[30] Lai, M.J.; Schumaker, L.L.: Spline functions on triangulations. Cambridge University Press, 2007. http://doi.org/10.1017/CBO9780511721588.

[31] Lasser, D.: Tensor product Bézier surfaces on triangle Bézier surfaces. Computer Aided Geometric Design, 19(8), 625–643, 2002. http://doi.org/10.1016/S0167-8396(02)00145-0.

[32] Liew, K.M.; Wang, C.; Xiang, Y.; Kitipornchai, S.: Vibration of Mindlin plates: programming the p-version Ritz Method. Amsterdam: Elsevier, 1998. http://doi.org/10.1002/bate.199903680.

[33] Lyness, J.N.; Cools, R.: A survey of numerical cubature over triangles. In Proceedings of Symposia in Applied Mathematics, 127–150. American Mathematical Society, 1994. http://doi.org/10.1090/psapm/048/1314845.

[34] Press, W.H.; Flannery, B.P.; Teukolsky, S.A.; Vetterling, W.T.: Numerical recipes in C: The art of scientific computing, third edition. Cambridge University Press, 2007. http://doi.org/10.1137/1031025.

[35] Schumaker, L.L.; Wang, L.: Spline spaces on TR-meshes with hanging vertices. Numerische Mathematik, 118, 531–548, 2011. http://doi.org/10.1007/s00211-010-0353-0.

[36] Scott, M.A.; Borden, M.J.; Verhoosel, C.V.; Sederberg, T.W.; Hughes, T.J.R.: Isogeometric finite element data structures based on Bézier extraction of T-splines. International Journal for Numerical Methods in Engineering, 88(2), 126–156, 2011. http://doi.org/10.1002/nme.3167.

[37] Speleers, H.; Manni, C.: Optimizing domain parameterization in isogeometric analysis based on Powell-Sabin splines. Journal of Computational and Applied Mathematics, 289, 68–86, 2015. http://doi.org/10.1016/j.cam.2015.03.024.

[38] Speleers, H.; Manni, C.; Pelosi, F.; Sampoli, M.L.: Isogeometric analysis with Powell-Sabin splines for advection-diffusion-reaction problems. Computer Methods in Applied Mechanics and Engineering, 221-222(1), 132–148, 2012. http://doi.org/10.1016/j.cma.2012.02.009.

[39] Vlachos, A.; Peters, J.; Boyd, C.; Mitchell, J.L.: Curved PN triangles. In Proceedings of the 2001 symposium on Interactive 3D graphics, 159–166, 2001. http://doi.org/10.1145/364338.364387.

[40] Willberg, C.; Duczek, S.; Perez, J.V.; Schmicker, D.; Gabbert, U.: Comparison of different higher order finite element schemes for the simulation of lamb waves. Computer Methods in Applied Mechanics and Engineering, 241, 246–261, 2012. http://doi.org/10.1016/j.cma.2012.06.011.

[41] Xia, S.; Qian, X.: Isogeometric analysis with Bézier tetrahedra. Computer Methods in Applied Mechanics and Engineering, 316, 782–816, 2017. http://doi.org/10.1016/j.cma.2016.09.045.

[42] Xia, S.; Wang, X.; Qian, X.: Continuity and convergence in rational triangular Bézier spline based isogeometric analysis. Computer Methods in Applied Mechanics and Engineering, 297, 292–324, 2015. http://doi.org/10.1016/j.cma.2015.09.001.

[43] Xu, G.; Mourrain, B.; Duvigneau, R.; Galligo, A.: Parameterization of computational domain in isogeometric analysis: Methods and comparison. Computer Methods in Applied Mechanics and Engineering, 200(23-24), 2021–2031, 2011. http://doi.org/10.1016/j.cma.2011.03.005.

[44] Yan, L.; Han, X.; Liang, J.: Conversion between triangular Bézier patches and rectangular Bézier patches. Appl. Math. Comput., 232(232), 469–478, 2014. http://doi.org/10.1016/S0167-8396(01)00060-7.

[45] Zeng, S.; Cohen, E.: Hybrid volume completion with higher-order Bézier elements. Computer Aided Geometric Design, 35-36, 180–191, 2015. http://doi.org/10.1016/j.cagd.2015.03.008.

[46] Zhao, G.; Du, X.; Wang, W.; Liu, B.; Fang, H.: Application of isogeometric method to free vibration of Reissner-Mindlin plates with non-conforming multi-patch. Computer-Aided Design, 2016. http://doi.org/10.1016/j.cad.2016.04.006.

[47] Zienkiewicz, O.C.; Taylor, R.L.: The finite element method for solid and structural mechanics, 6th edition. Butterworth-Heinemann, 2005. http://doi.org/10.1016/C2009-0-26332-X.

[48] Zienkiewicz, O.C.; Taylor, R.L.; Zhu, J.Z.: The Finite Element Method: Its Basis and Fundamentals (Seventh Edition), 2005. http://doi.org/10.1016/C2009-0-24909-9.