Seamless integration of design and Kirchhoff-Love shell analysis using analysis-suitable unstructured T-splines

Hugo Casquero[a,*], Xiaodong Wei[a,e,*], Deepesh Toshniwal[b], Angran Li[a], Thomas J.R. Hughes[c], Josef Kiendl[d], Yongjie Jessica Zhang[a]

[a] *Department of Mechanical Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, U.S.A.*
[b] *Delft Institute of Applied Mathematics, Delft University of Technology, Van Mourik Broekmanweg 6, XE Delft 2628, The Netherlands.*
[c] *Oden Institute for Computational Engineering and Sciences, 201 East 24th Street, C0200, Austin, TX 78712-1229, U.S.A.*
[d] *Department of Marine Technology, Norwegian University of Science and Technology, O. Nielsens veg 10, 7052 Trondheim, Norway.*
[e] *Institute of Mathematics, École Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland.*

Abstract

Analysis-suitable T-splines (ASTS) including both extraordinary points and T-junctions are used to solve Kirchhoff-Love shell problems. Extraordinary points are required to represent surfaces with arbitrary topological genus. T-junctions enable local refinement of regions where increased resolution is needed. The benefits of using ASTS to define shell geometries are at least two-fold: (1) The manual and time-consuming task of building a new mesh from scratch using the CAD geometry as an input is avoided and (2) C^1 or higher inter-element continuity enables the discretization of shell formulations in primal form defined by fourth-order partial differential equations. A complete and state-of-the-art description of the development of ASTS, including extraordinary points and T-junctions, is presented. In particular, we improve the construction of C^1-continuous non-negative spline basis functions near extraordinary points to obtain optimal convergence rates with respect to the square root of the number of degrees of freedom when solving linear elliptic problems. The applicability of the proposed technology to shell analysis is exemplified by performing geometrically nonlinear Kirchhoff-Love shell simulations of a pinched hemisphere, an oil sump of a car, a pipe junction, and a B-pillar of a car with 15 holes. Building ASTS for these examples involves using T-junctions and extraordinary points with valences 3, 5, and 6, which often suffice for the design of free-form surfaces. Our analysis results are compared with data from the literature using either a seven-parameter shell formulation or Kirchhoff-Love shells. We have also imported both finite element meshes and ASTS meshes into the commercial software LS-DYNA, used Reissner-Mindlin shells, and compared the result with our Kirchhoff-Love shell results. Excellent agreement is found in all cases. The

*Corresponding authors.
Email addresses: hugocp@andrew.cmu.edu (Hugo Casquero), xiaodong.wei@epfl.ch (Xiaodong Wei)

complexity of the shell geometries considered in this paper shows that ASTS are applicable to real-world industrial problems.

Keywords:
Isogeometric analysis, Analysis-suitable T-splines, Extraordinary points, Optimal approximation, T-junctions, Geometrically nonlinear Kirchhoff-Love shells

1. Introduction

In engineering applications, shell geometries are often created in computer-aided-design (CAD) programs using different types of spline functions, such as non-uniform rational B-splines (NURBS) [1] and T-splines [2]. In order to generate complex geometries, the tensor-product structure of NURBS requires gluing together many different NURBS surfaces called patches. This procedure has at least two shortcomings: (1) Superfluous control points are needed due to the tensor-product structure of each patch [3] and (2) watertight geometries are difficult to obtain in most cases [4]. T-splines, through the use of T-junctions and extraordinary points, overcome the aforementioned shortcomings of NURBS [3, 4]. T-junctions remove the tensor-product constraint of NURBS patches, thus enabling the placement of control points only where needed. Extraordinary points enable the representation of geometries with arbitrary topology using only one T-spline surface. The unstructured nature of T-splines also provides an alternative to the use of trimmed NURBS surfaces [5, 6].

When it comes to performing numerical simulations of shell structures in computer-aided-engineering (CAE) programs, the current spline representations used in CAD programs are not necessarily suitable for analysis. As a result, a new mesh needs to be built from scratch using the original CAD geometry as an input. This process is time consuming, taking up to 80% of the "design to analysis" cycle in certain engineering applications [7, 8]. Moreover, significant geometry modifications are often inevitable, thus jeopardizing the reliability of the analysis results. C^0 Lagrange polynomials, the standard basis functions in the finite element method [9], are frequently used to create the computational mesh. This reduces the smoothness of the original spline surface.

Isogeometric analysis (IGA) was introduced with the goal of developing a seamless integration between CAD and CAE programs when it comes to geometry representation. As a first step, a NURBS patch was shown to be a suitable basis for analysis in [10, 11, 12, 13]. In contrast with the C^0 inter-element continuity of Lagrange polynomials, the higher inter-element continuity of a NURBS patch brings the following advantages: higher-order partial differential equations can be solved in primal form [14, 15, 16, 17, 18], enhanced robustness in solid mechanics is obtained [19], enhanced accuracy in spectrum analysis is achieved [20], H^1-conforming discretizations that are either divergence-conforming or curl-conforming can be derived in a straightforward manner [21, 22, 23, 24], and partial differential equations can be collocated in strong form [25, 26]. As opposed to a NURBS patch, which is directly analysis-suitable, T-junctions are not necessarily analysis-suitable [27]. To remedy this, a subset of T-splines called analysis-suitable T-splines (ASTS), including T-junctions but

not extraordinary points, was defined which maintains all the important geometric and mathematical properties of a NURBS patch [28, 29, 30, 31, 32, 33, 34, 35]. When at least C^1 inter-element continuity is imposed, the use of multi-patch NURBS or T-splines with extraordinary points does not necessarily lead to spaces with optimal approximation properties in analysis [36, 37, 38]. A construction of extraordinary points with at least C^1 inter-element continuity that results in optimal convergence rates with respect to the mesh size h for second- and fourth-order linear elliptic problems was recently developed in [39], thus enabling the extension of the ASTS definition to include both T-junctions and extraordinary points.

NURBS-based IGA was applied to Kirchhoff-Love shells in [15, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50], Reissner-Mindlin shells in [51, 52, 53, 54, 55], solid-shell elements in [56, 57, 58, 59, 60], a hierarchic family of linear shells [61], and a shell formulation that blends Kirchhoff-Love theory with Reissner-Mindlin theory [62]. ASTS-based IGA, including T-junctions but not extraordinary points, was recently applied to Kirchhoff-Love shells in [63]. Triangular Loop and quadrilateral Catmull-Clark subdivision surfaces are another appealing alternative to integrate geometric modeling and shell analysis [64, 65, 66, 67, 68, 69, 70]. In the neighborhood of extraordinary points, however, the resulting basis functions obtained with subdivision surfaces are non-polynomial, which (1) complicates the numerical integration [71, 72] and (2) harms the approximation order of the spaces [73, 72].

In this work, we describe a detailed blueprint for the construction of smooth non-negative bi-cubic ASTS on unstructured meshes. This description is self-contained, complete, and general enough to handle all cases of interest in practice where one needs to work with meshes containing both extraordinary points and T-junctions. Extraordinary points enable generation of quadrilateral meshes for geometries with arbitrary topologies, while T-junctions allow us to locally increase the spline space resolution to one required for the purpose of analysis. Augmenting the classical ASTS construction (on locally structured meshes with T-junctions) with a novel extraordinary point treatment, we are able to achieve optimal convergence rates with respect to not only the mesh size but also the square root of the number of degrees of freedom for second- and fourth-order linear elliptic problems. We believe our description of the unstructured T-spline technology with extraordinary points improves upon and supersedes previous presentations in the literature, such as [74], which was deficient in its inability to achieve optimal convergence rates in the presence of extraordinary points. The performance and flexibility of the construction is further exemplified using a test-suite of geometrically nonlinear Kirchhoff-Love shells. We consider geometries of increasing complexity, including an automotive B-pillar. Although we focus on Kirchhoff-Love shells in this paper, our C^1-continuous ASTS meshes can be used to discretize other shell formulations such as Reissner-Mindlin shells.

The paper is outlined as follows. In Section 2, the construction of ASTS including both extraordinary points and T-junctions is described. In Section 3, geometrically nonlinear Kirchhoff-Love shell theory is presented in curvilinear coordinates. Section 4.1 tests the formulation on the pinched hemisphere benchmark problem. Section 4.2 considers the automotive oil-sump geometry presented in [75]. Section 4.3 deals with a pipe junction and our

results are compared with the commercial software LS-DYNA that uses the isogeometric Reissner-Mindlin formulation proposed in [51]. An automotive B-pillar geometry is considered in Section 4.4. The B-pillar geometry is also meshed using Lagrange polynomials for comparison with the well-established finite element discretization of Reissner-Mindlin shell theory proposed in [76]. Conclusions and future research directions are drawn in Section 5.

2. Analysis-suitable T-splines

Building upon theoretical studies of extraordinary points [77, 39], truncation [78, 79], and T-junctions [28, 80, 31], we explain in this section the construction of bi-cubic ASTS of arbitrary topology satisfying the following properties:

(1) Linear independence of the blending functions, that is, the blending functions constitute a basis.

(2) Partition of unity of the polynomial basis functions.

(3) Each basis function is pointwise non-negative.

(4) At least C^1 continuity everywhere while having a finite representation.

(5) Local support of the basis functions.

(6) Local h-refinement capabilities.

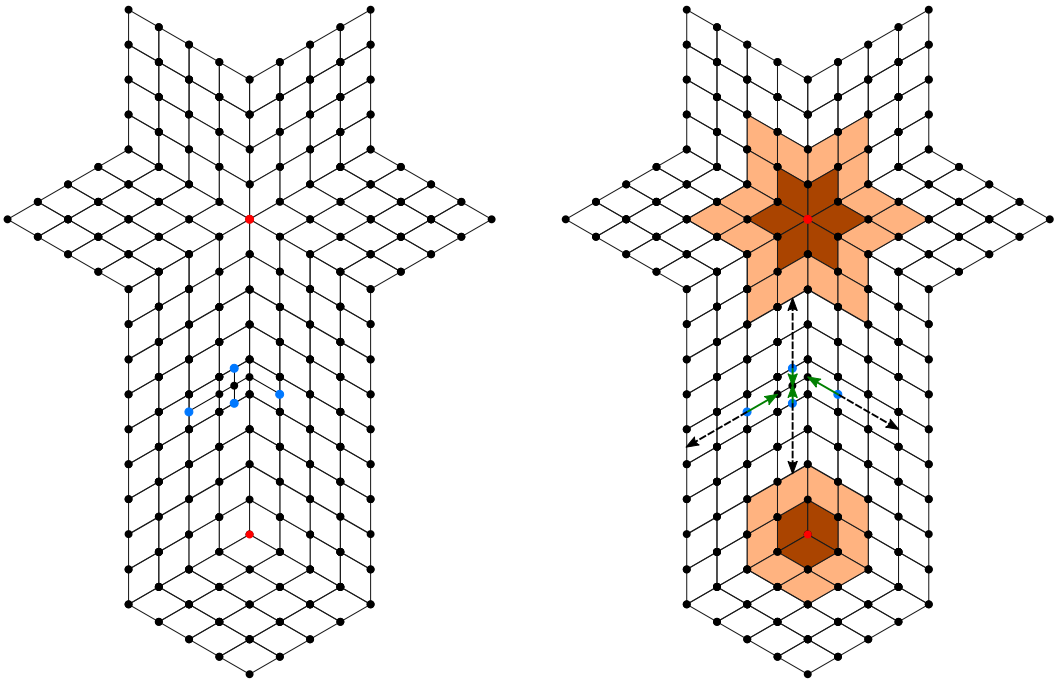
(7) Optimal convergence rates with respect to both the mesh size h and the square root of the number of degrees of freedom when solving second- and fourth-order linear elliptic partial differential equations.

Properties (2) and (3) guarantee that ASTS satisfy the convex-hull property. Property (1) implies that an affine transformation of an ASTS surface is obtained by applying the transformation to the control points. In this section, we assume that the reader is familiar with NURBS [1, 81] and Bézier extraction [82, 83].

2.1. T-mesh

The *T-mesh* defines the topological information of a T-spline surface* and it plays a key role in defining the subset of ASTS. An example of a T-mesh is shown in Fig. 1 (a). The *vertices* are marked with circles. The *edges* are closed line segments that connect two vertices without passing through any other vertex. The *faces* are the white regions delimited by edges. The *valence* of a vertex, denoted by μ, is the number of edges that emanate from that vertex. *T-junctions* are vertices playing the same role as hanging nodes in classical finite elements. *Extraordinary points* are either interior vertices with $\mu \neq 4$

*Here "topological information of a T-spline surface" must be understood as the way in which the different constituent parts of the T-spline technology are interrelated or arranged.



(a) T-mesh (b) Extended T-mesh

Figure 1: (a) T-mesh with two extraordinary points and four T-junctions. The extraordinary points and T-junctions are marked with red and blue circles, respectively. (b) Extended T-mesh. Face extensions and edge extensions are represented with dashed black lines and green solid lines, respectively.

that are not T-junctions or boundary vertices with $\mu > 3$. The edges emanating from an extraordinary point are called *spoke edges*. In Fig. 1 (a), T-junctions and extraordinary points are marked with blue and red circles, respectively. Important paraphernalia for dealing with T-junctions and extraordinary points within the subset of ASTS is detailed below.

A *face extension* is a closed directed line segment that originates at a T-junction and moves in the direction of the missing edge until two orthogonal edges are encountered. A *one-bay* face extension is the part of a face extension that lies on the face adjacent to the T-junction. An *edge extension* is a closed directed line segment that originates at a T-junction and moves in the opposite direction of the face extension until one orthogonal edge is encountered. A *T-junction extension* is composed of a face and an edge extension. Since T-junction extensions are closed line segments, a T-junction extension can intersect with other T-junction extension either in its interior or at its endpoints. The *extended T-mesh* is obtained adding the T-junctions extensions to the T-mesh. Fig. 1 (b) plots the extended T-mesh associated with the T-mesh shown in Fig. 1 (a).

The *1-ring faces* of an extraordinary point are the faces that are in contact with the extraordinary point. For $m > 1$, the *m-ring faces* of an extraordinary point are all faces that touch the (m-1)-ring faces and are not a part of the (m-2)-ring faces. The *m-disk faces* of an extraordinary point are defined as the set containing all its 1-, 2-, ... , m-ring faces.

In Fig. 1 (b), the 1- and 2-rings faces around extraordinary points are darkly and lightly shaded, respectively. The set of *0-ring vertices* of an extraordinary point contains only the extraordinary point itself. For $m > 0$, the *m-ring vertices* of an extraordinary point contains all the vertices that lie on the m-ring faces but are not a part of the (m-1)-ring vertices. The *m-disk vertices* of an extraordinary point are the union of all its 0-, 1-, 2-, ... , m-ring vertices.

The set of *1-layer* faces around the T-mesh boundary contains the faces that are in contact with the T-mesh boundary. For $m > 1$, the set of *m-layer* faces around the T-mesh boundary are all faces that touch the (m-1)-layer faces and are not a part of the (m-2)-layer faces. The set of *0-layer vertices* around the T-mesh boundary contains the vertices at the T-mesh boundary. For $m > 0$, the *m-layer vertices* around the T-mesh boundary contains all the vertices that lie on the m-layer faces but are not a part of the (m-1)-layer vertices.

We consider a T-mesh to be *admissible* when it satisfies the following topological conditions:

- No extraordinary point belongs to the 3-disk vertices of any other extraordinary point.
- No one-bay face extension subdivides a 3-disk face of an extraordinary point.
- No topologically perpendicular T-junction extensions intersect.
- No T-junction topologically parallel to the boundary belongs to the 0-, 1-layer vertices around the T-mesh boundary.
- No extraordinary point belongs to the 0-, 1-, 2-layer vertices around the T-mesh boundary.

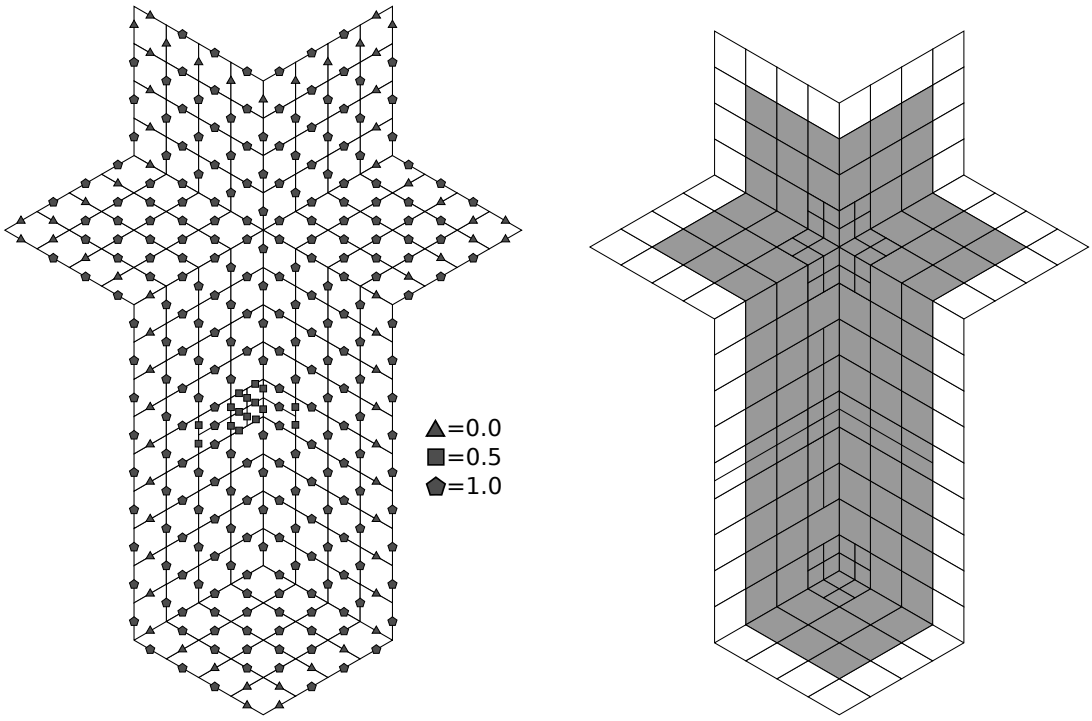
ASTS are T-splines defined over an admissible T-mesh. These topological conditions are not restrictive and complex geometries can be built satisfying these conditions as we will show in the examples of this paper.

2.2. Knot spans

Knot spans define the parametric information of a T-spline surface. A knot span is a non-negative real number assigned to a T-mesh edge. A *valid* knot span configuration must satisfy two conditions: (1) knot spans on opposite sides of every face are required to sum to the same value and (2) all the edges that emanate from the T-mesh boundary are assigned with zero knot spans. Fig. 2 (a) plots a possible knot span configuration for the T-mesh shown in Fig. 1 (a).

For simplicity of exposition and since it does not pose a restriction on the complexity of the geometries that can be generated, we will henceforth focus on T-splines with the same nonzero knot span a assigned to all the edges that delimit faces within the 3-disk faces of extraordinary points. All the T-splines built in this paper satisfy this condition.

The *elemental T-mesh* defines regions inside which all basis functions are C^∞. These regions, called *elements*, are suitable for performing both Bézier extraction and numerical integration in engineering analysis. The elements of the elemental T-mesh are obtained by modifying the T-mesh as follows:



(a) Knot spans (b) Elemental T-mesh

Figure 2: (a) A possible knot span configuration for the T-mesh represented in Fig. 1 (a). The pentagons, squares, and triangles correspond to knot spans with values 1, 1/2, and 0, respectively. (b) Elemental T-mesh associated with the T-mesh and the knot span configuration represented in Fig. 1 (a) and Fig. 2 (a), respectively. The elements of the elemental T-mesh are the shaded regions delimited by black lines.

- Adding the face extension to each T-junction, which subdivides faces into two elements.
- Subdividing each face within the 1-ring faces of an extraordinary point into four elements.
- Eliminating faces with zero parametric measure.

The number of elements in the elemental T-mesh is denoted by n_{el}. Fig. 2 (b) plots the elemental T-mesh associated with the T-mesh shown in Fig. 1 (a) and the knot span configuration shown in Fig. 2 (a).

2.3. Bézier extraction

A convenient way of dealing with splines in CAE software as well as even defining splines is Bézier extraction. Bézier extraction is built upon the fact that a spline basis function within each element of the elemental T-mesh is simply a bi-cubic polynomial. Thus, a spline basis function N_A restricted to an element e can be expressed as a linear combination of the 16 bi-cubic tensor-product Bernstein polynomial basis functions, viz.,

$$N_A|_e = N_a^e(\boldsymbol{\xi}) = \sum_{j=1}^{16} C_{aj}^e b_j(\boldsymbol{\xi}), \quad \boldsymbol{\xi} \in \square, \tag{1}$$

where $\square$ is the parent element domain[†], b_j is the j-th bi-cubic Bernstein polynomial, A is a global spline basis function index, and a is a local-to-element spline basis function index. Following [9], we use the array IEN to establish a correspondence between local and global numbering of basis functions, namely, $A = \text{IEN}(a, e)$. Let us collect all the spline basis functions with support on element e and the 16 Bernstein polynomials in column vectors $\boldsymbol{N}^e = (N_1^e, N_2^e, ..., N_{n^e}^e)^T$, $\boldsymbol{b} = (b_1, b_2, ..., b_{16})^T$, respectively. The *spline extraction operator* $\boldsymbol{C}^e$ is the linear map from $\boldsymbol{b}$ to $\boldsymbol{N}^e$, viz.,

$$\boldsymbol{N}^e(\boldsymbol{\xi}) = \boldsymbol{C}^e \boldsymbol{b}(\boldsymbol{\xi}), \quad \boldsymbol{\xi} \in \square. \tag{2}$$

Note that $\boldsymbol{C}^e$ is a matrix of dimension $n^e \times 16$ where n^e is the number of spline basis functions with support on element e. Within the subset of ASTS, for elements affected by T-junctions, $n^e = 16$ as for the case of NURBS, but for elements affected by extraordinary points, n^e may be higher than 16. In an analogous way to how Bernstein polynomials can be related to spline basis functions, spline control points $\boldsymbol{P}^e = (\boldsymbol{P}_1^e, \boldsymbol{P}_2^e, ..., \boldsymbol{P}_{n^e}^e)^T$ can be related to Bézier control points $\boldsymbol{B}^e = (\boldsymbol{B}_1^e, \boldsymbol{B}_2^e, ..., \boldsymbol{B}_{16}^e)^T$ as follows

$$\boldsymbol{B}^e = (\boldsymbol{C}^e)^T \boldsymbol{P}^e. \tag{3}$$

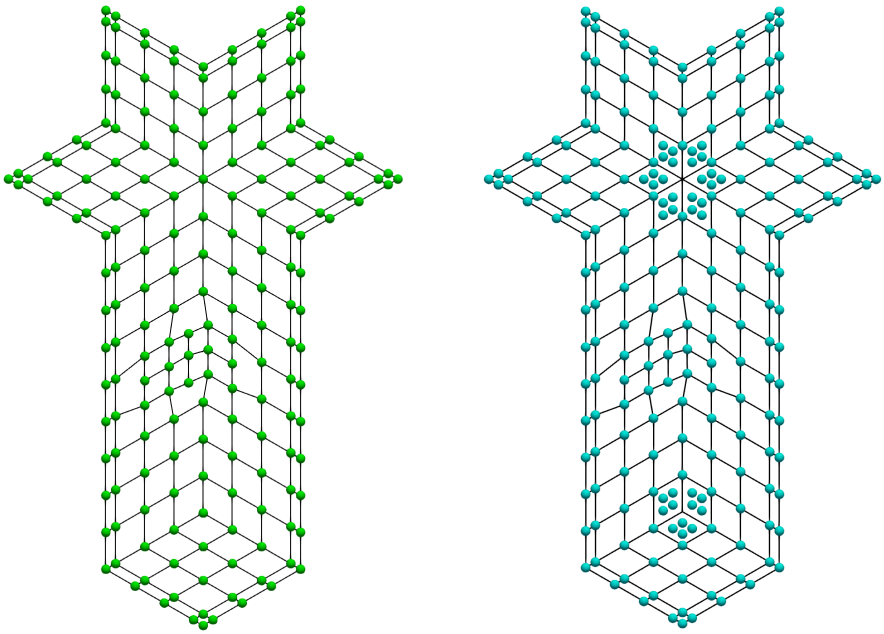
Note that $\boldsymbol{P}^e$ and $\boldsymbol{B}^e$ are matrices of dimension $n^e \times 3$ and 16×3, respectively. $\boldsymbol{E}^e = (\boldsymbol{C}^e)^T$ is the *Bézier extraction operator*, however, $\boldsymbol{C}^e$ and $\boldsymbol{E}^e$ will both be referred to as *extraction operators*.

2.4. Control points

Control points define the geometric information of a T-spline surface. A control point is a vector that stores the coordinates of a point in the three-dimensional Euclidean space. Control points play the same role as nodes in classical finite elements, but do not interpolate the T-spline surface in general. In CAD, a control net with simple connectivity is required so that moving its constituent control points modifies the associated T-spline surface in an intuitive manner. This is accomplished by assigning a control point to each T-mesh vertex[‡], thus leading to a control net with the same connectivity as the T-mesh. Using control nets and T-meshes with matching connectivity, it is known how to build bi-cubic T-spline spaces satisfying properties (1)-(6), but how to satisfy property (7) as well remains

[†]The parent element domain is usually chosen to be $[0, 1]^2$ and $[-1, 1]^2$ in CAD and CAE, respectively. The extraction operators do not depend on this choice, but the particular expressions for the Bernstains polynomials do.

[‡]Besides assigning a control point $\boldsymbol{P}_L \in \mathbb{R}^3$ to each T-mesh vertex, a weight $w_L \in \mathbb{R}^+$ can also be assigned to each vertex in order to define rational splines. For brevity, we focus on polynomial splines in this paper, that is, no weights are assigned to the vertices.



(a) Geometric modeling (b) Engineering analysis

Figure 3: (a) The control points of $\mathbb{S}_D$ are represented with green circles and the control net of $\mathbb{S}_D$ is represented by solid black lines. (b) The control points of $\mathbb{S}_A$ are represented with blue circles. The control net of $\mathbb{S}_D$ is also plotted.

an open problem. In order to circumvent this issue, as delineated in [39], two different T-spline spaces with different sets of control points are defined as follows:

- A T-spline space for CAD ($\mathbb{S}_D$), which satisfies properties (1)-(6). This space has vertex-based control points. From now on, the control points of $\mathbb{S}_D$ will be denoted by $\boldsymbol{P}_L$, where $L \in \{1, ..., n\}$ and n is the number of T-mesh vertices. Fig. 3 (a) plots a possible control net for CAD associated with the T-mesh shown in Fig. 1 (a).

- A T-spline space for CAE ($\mathbb{S}_A$), which satisfies properties (1)-(7). T-mesh vertices that are not extraordinary points are assigned a control point. In addition, four face-based control points are assigned to each 1-ring face of an extraordinary point. From now on, the vertex-based control points of $\mathbb{S}_A$, the face-based control points of $\mathbb{S}_A$, and all the control points of $\mathbb{S}_A$ will be denoted by $\hat{\boldsymbol{Q}}_V$, $\tilde{\boldsymbol{Q}}_F$, and $\boldsymbol{Q}_O$, respectively, where $F \in \{1, ..., n - n_{ep}\}$, $V \in \{1, ..., \sum_{j=1}^{n_{ep}} 4\mu_j\}$, $O \in \{1, ..., n_{cp}\}$, n_{ep} is the number of extraordinary points, μ_j is the valence of the jth extraordinary point, and $n_{cp} = n - n_{ep} + \sum_{j=1}^{n_{ep}} 4\mu_j$ is the number of control points in $\mathbb{S}_A$.

The spaces $\mathbb{S}_D$ and $\mathbb{S}_A$ are constructed in such a way that $\mathbb{S}_D \subseteq \mathbb{S}_A$. As a result, once we have obtained a satisfactory geometry by moving control points in $\mathbb{S}_D$, it is possible to obtain a set of control points in $\mathbb{S}_A$ that preserves the same geometry. This is accomplished

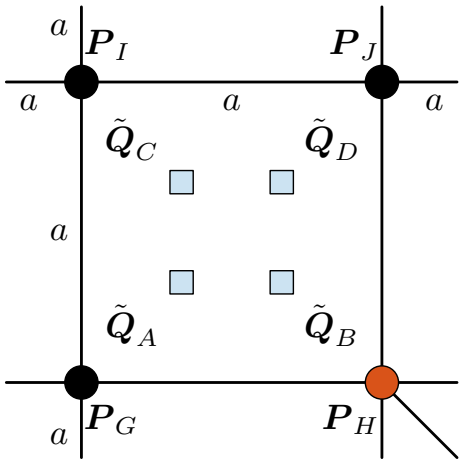


Figure 4: (Color online) The face-based spline control points of $\mathbb{S}_A$ are obtained as linear combinations of the vertex-based spline control points of $\mathbb{S}_D$ using Eqs. (4)-(7).

by keeping the vertex-based control points not associated with extraordinary points unchanged, discarding the vertex-based control points associated with extraordinary points, and computing the face-based control points using the following formulas:

$$\tilde{\boldsymbol{Q}}_A = \frac{4}{9}\boldsymbol{P}_G + \frac{2}{9}\boldsymbol{P}_H + \frac{2}{9}\boldsymbol{P}_I + \frac{1}{9}\boldsymbol{P}_J, \tag{4}$$

$$\tilde{\boldsymbol{Q}}_B = \frac{2}{9}\boldsymbol{P}_G + \frac{4}{9}\boldsymbol{P}_H + \frac{1}{9}\boldsymbol{P}_I + \frac{2}{9}\boldsymbol{P}_J, \tag{5}$$

$$\tilde{\boldsymbol{Q}}_C = \frac{2}{9}\boldsymbol{P}_G + \frac{1}{9}\boldsymbol{P}_H + \frac{4}{9}\boldsymbol{P}_I + \frac{2}{9}\boldsymbol{P}_J, \tag{6}$$

$$\tilde{\boldsymbol{Q}}_D = \frac{1}{9}\boldsymbol{P}_G + \frac{2}{9}\boldsymbol{P}_H + \frac{2}{9}\boldsymbol{P}_I + \frac{4}{9}\boldsymbol{P}_J, \tag{7}$$

where the labels used correspond to Fig. 4. Fig. 3 (b) plots the control points of $\mathbb{S}_A$ associated with the control points of $\mathbb{S}_D$ shown in Fig. 3 (a). Eqs. (4)-(7) are obtained by imposing the Bézier control points of $\mathbb{S}_D$ and $\mathbb{S}_A$ to be the same in the 1-ring faces of extraordinary points.

2.5. Spline basis functions

In both $\mathbb{S}_D$ and $\mathbb{S}_A$, a spline basis function is associated with each spline control point. From now on, the basis functions of $\mathbb{S}_D$ and $\mathbb{S}_A$ will be denoted by N_L and M_O, respectively, where $L \in \{1, ..., n\}$ and $O \in \{1, ..., n_{cp}\}$. In order to define these basis functions, we will classify the faces and vertices of the T-mesh as follows:

- *Irregular faces* are the 1-ring faces of extraordinary points. *Transition faces* are the 2-ring faces of extraordinary points. The remaining faces are *regular faces*. Element boundaries within regular faces are C^2-continuous while element boundaries within irregular and transition faces may be only C^1-continuous. The four face-based basis functions associated with each irregular face in $\mathbb{S}_A$ are globally C^1-continuous.

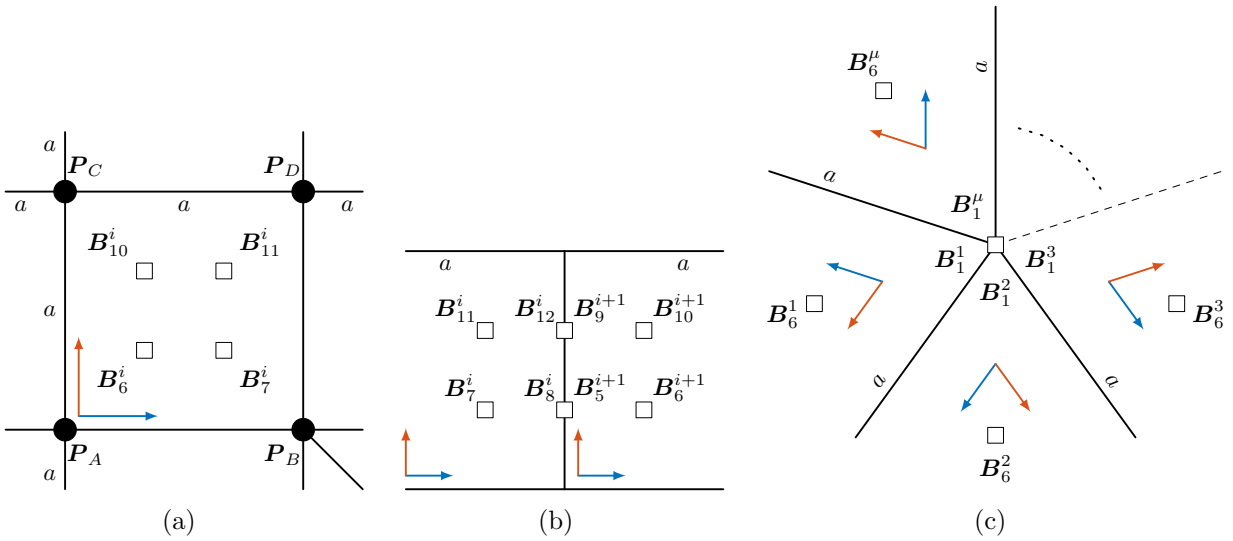


Figure 5: (a)-(c) For the 2-disk faces of extraordinary points, the Bézier control points can be obtained from the spline control points using Eqs. (8)-(14).

- *Irregular vertices* are the 0-ring vertices of extraordinary points. *Transition vertices* are the 1-ring vertices of extraordinary points. The remaining vertices are *regular vertices*. The spline basis functions associated with regular vertices are globally C^2-continuous while the spline basis functions associated with irregular and transition vertices are globally C^1-continuous. The control points and basis functions are also classified as irregular, transition, and regular according to the vertex with which they are associated.

The basis functions are defined through the extraction operators in each element of the elemental T-mesh. The extraction operators of elements that are within regular faces are the same for geometric modeling ($\mathbb{S}_D$) and engineering analysis ($\mathbb{S}_A$), but for irregular and transition faces are different.

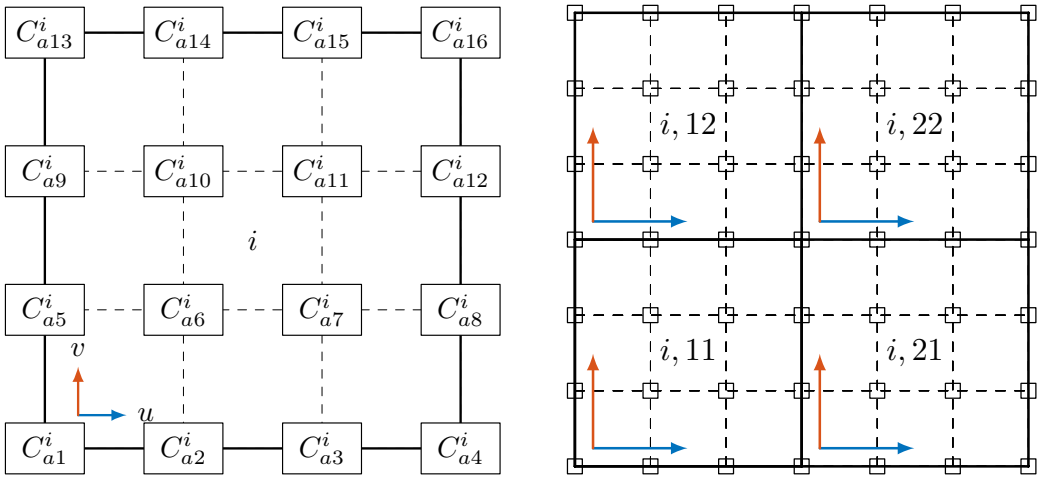
2.5.1. Irregular and transition faces in geometric modeling

We start associating just one element to each face and distinguish three types of Bézier control points, namely, face, edge, and vertex Bézier control points. Following [74, 39], face Bézier control points are defined in terms of spline control points as

$$\boldsymbol{B}_6^i = \frac{4}{9}\boldsymbol{P}_A + \frac{2}{9}\boldsymbol{P}_B + \frac{2}{9}\boldsymbol{P}_C + \frac{1}{9}\boldsymbol{P}_D, \tag{8}$$

$$\boldsymbol{B}_7^i = \frac{2}{9}\boldsymbol{P}_A + \frac{4}{9}\boldsymbol{P}_B + \frac{1}{9}\boldsymbol{P}_C + \frac{2}{9}\boldsymbol{P}_D, \tag{9}$$

$$\boldsymbol{B}_{10}^i = \frac{2}{9}\boldsymbol{P}_A + \frac{1}{9}\boldsymbol{P}_B + \frac{4}{9}\boldsymbol{P}_C + \frac{2}{9}\boldsymbol{P}_D, \tag{10}$$



(a) Before 2×2 split (b) After 2×2 split

Figure 6: (a) Extraction coefficients and local coordinate system before the 2×2 split. (b) Extraction coefficients and local coordinate system after the 2×2 split.

$$\boldsymbol{B}^i_{11} = \frac{1}{9}\boldsymbol{P}_A + \frac{2}{9}\boldsymbol{P}_B + \frac{2}{9}\boldsymbol{P}_C + \frac{4}{9}\boldsymbol{P}_D, \tag{11}$$

edge Bézier control points are defined in terms of adjacent face Bézier control points as

$$\boldsymbol{B}^i_8 = \boldsymbol{B}^{i+1}_5 = \frac{1}{2}\boldsymbol{B}^i_7 + \frac{1}{2}\boldsymbol{B}^{i+1}_6, \tag{12}$$

$$\boldsymbol{B}^i_{12} = \boldsymbol{B}^{i+1}_9 = \frac{1}{2}\boldsymbol{B}^i_{11} + \frac{1}{2}\boldsymbol{B}^{i+1}_{10}, \tag{13}$$

and vertex Bézier control points are defined in terms of adjacent face Bézier control points as

$$\boldsymbol{B}^1_1 = \boldsymbol{B}^2_1 = ... = \boldsymbol{B}^\mu_1 = \frac{1}{\mu}\sum_{j=1}^{\mu} \boldsymbol{B}^j_6, \tag{14}$$

where the labels used correspond to Fig. 5. Eqs. (8)-(14) define initial expressions for the extraction operators of the elements within irregular and transition faces. At this point in the process, all edges are C^2-continuous with the exception of the spoke edges, which are only C^0-continuous. In order to obtain C^1 continuity at the spoke edges, we will modify the extraction operators of the elements within irregular faces using the split-then-smoothen approach [39, 84, 85]. The main reason why we do the split before the smoothing is to maintain the linear independence of the spline basis functions. This approach is applied to each basis function that has support on the irregular faces as follows:

(a) The extraction coefficients that define the basis function on each irregular face are collected in row vectors $\boldsymbol{C}^i_a = (C^i_{a1}, C^i_{a2}, ..., C^i_{a16})$ with i being cyclic in $\{1, 2, ..., \mu\}$.

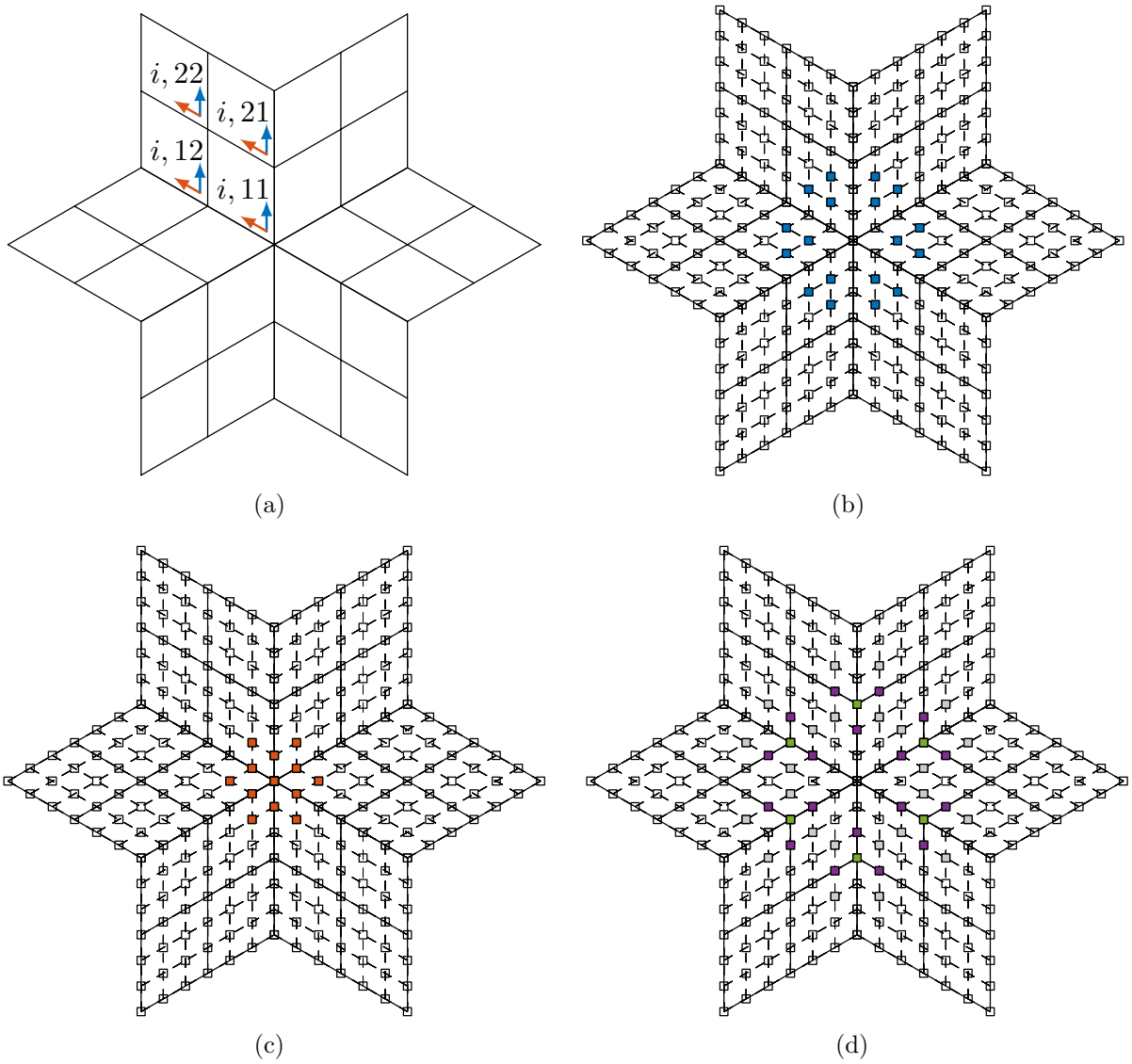


Figure 7: (a) 1-ring faces of an extraordinary point with valence 6 after the 2×2 split using the Casteljau algorithm. (b) The extraction coefficients that are changed by the smoothing matrix are plotted in blue. (c) The extraction coefficients that end up having the same value in the D-patch framework are plotted in red. (d) The edge and vertex extraction coefficients that need to be recomputed after application of the smoothing matrix are plotted in violet and green, respectively. The face extraction coefficients that intervene in the aforementioned recomputations are plotted in gray.

Given a local coordinate system, we arrange these coefficients in a 4×4 grid following the numbering shown in Fig. 6 (a).

(b) We split each irregular face at parameter lines $u = a/2$ and $v = a/2$ using the Casteljau algorithm [81]. Thus, each irregular face has four elements now. Each element has its own local coordinate system and the spline basis function is defined

in each of these elements by 16 extraction coefficients as shown in Fig. 6 (b). The extraction coefficients that define the basis function on these four elements, denoted by $\boldsymbol{C}_a^{i,pq}$ with $p, q \in \{1, 2\}$, are obtained as follows

$$\boldsymbol{C}_a^{i,pq} = \boldsymbol{C}_a^i \left(\boldsymbol{S}_q \otimes \boldsymbol{S}_p\right)^T, \tag{15}$$

with

$$\boldsymbol{S}_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} & 0 \\ \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \end{pmatrix}, \quad \boldsymbol{S}_2 = \begin{pmatrix} \frac{1}{8} & \frac{3}{8} & \frac{3}{8} & \frac{1}{8} \\ 0 & \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{16}$$

where $\otimes$ is defined for two matrices $\boldsymbol{G}$ and $\boldsymbol{H}$ as

$$\boldsymbol{G} \otimes \boldsymbol{H} = \begin{pmatrix} G_{11}\boldsymbol{H} & G_{12}\boldsymbol{H} & \dots \\ G_{21}\boldsymbol{H} & G_{22}\boldsymbol{H} & \\ \vdots & & \ddots \end{pmatrix}. \tag{17}$$

This split leaves the basis function unchanged since it is performed using the Casteljau algorithm and does not change the number of spline control points.

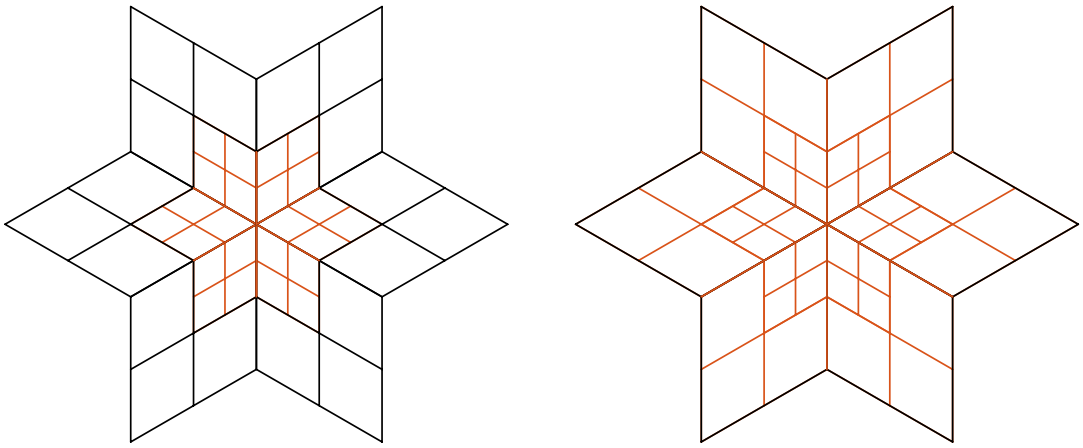
(c) Using the D-patch framework [85] and based on the local axes shown in Fig. 7 (a), the extraction coefficients $C_{a6}^{i,11}$, $C_{a7}^{i,11}$, and $C_{a10}^{i,11}$ with i being cyclic in $\{1, 2, ..., \mu\}$ are modified using a smoothing matrix $\boldsymbol{\Pi}$ with dimension $3\mu \times 3\mu$ as follows

$$\begin{pmatrix} \boldsymbol{a}_6 \\ \boldsymbol{a}_7 \\ \boldsymbol{a}_{10} \end{pmatrix} = \boldsymbol{\Pi} \begin{pmatrix} \boldsymbol{A}_6 \\ \boldsymbol{A}_7 \\ \boldsymbol{A}_{10} \end{pmatrix}, \tag{18}$$

with

$$\boldsymbol{A}_6 = \begin{pmatrix} C_{a6}^{1,11} \\ C_{a6}^{2,11} \\ \vdots \\ C_{a6}^{\mu,11} \end{pmatrix}, \quad \boldsymbol{A}_7 = \begin{pmatrix} C_{a7}^{1,11} \\ C_{a7}^{2,11} \\ \vdots \\ C_{a7}^{\mu,11} \end{pmatrix}, \quad \boldsymbol{A}_{10} = \begin{pmatrix} C_{a10}^{1,11} \\ C_{a10}^{2,11} \\ \vdots \\ C_{a10}^{\mu,11} \end{pmatrix}. \tag{19}$$

The extraction coefficients $C_{a6}^{i,11}$, $C_{a7}^{i,11}$, and $C_{a10}^{i,11}$ with $i \in \{1, 2, ..., \mu\}$ are replaced by their modified values obtained from the column vectors $\boldsymbol{a}_6$, $\boldsymbol{a}_7$, and $\boldsymbol{a}_{10}$, respectively. Fig. 7 (b) plots in blue the extraction coefficients that are changed by $\boldsymbol{\Pi}$. A choice needs to be made between picking a smoothing matrix $\boldsymbol{\Pi}^+$ with non-negative entries that leads to non-negative basis functions and an idempotent matrix $\boldsymbol{\Pi}^\circ$ that leads to nested spaces [85]. After running all the examples in this paper with both $\boldsymbol{\Pi}^+$



(a) $\mathbb{S}_D$ (b) $\mathbb{S}_A$

Figure 8: (a)-(b) Continuity of the basis at element boundaries in the 2-disk faces of an extraordinary point for $\mathbb{S}_D$ and $\mathbb{S}_A$, respectively. Black and orange lines represent C^2 and C^1 continuity lines, respectively.

and $\mathbf{\Pi}^{\circ}$, we have found either indistinguishable results with both smoothing matrices or increased accuracy with $\mathbf{\Pi}^{+}$ as we refine. Therefore, we favor the use of $\mathbf{\Pi}^{+}$ since having non-negative basis functions is a necessary requirement in CAD and an appealing additional property in CAE. $\mathbf{\Pi}^{+}$ is defined as follows

$$\mathbf{\Pi}^{+} = \begin{pmatrix} \mathbf{\Pi}_1^{+} & \mathbf{\Pi}_2^{+} & \mathbf{\Pi}_3^{+} \\ \mathbf{\Pi}_4^{+} & \mathbf{\Pi}_5^{+} & \mathbf{\Pi}_6^{+} \\ \mathbf{\Pi}_7^{+} & \mathbf{\Pi}_8^{+} & \mathbf{\Pi}_9^{+} \end{pmatrix}, \tag{20}$$

$$(\mathbf{\Pi}_i^{+})_{jk} = (\boldsymbol{p}_i)_{\mathrm{mod}(j-k,\mu)}, \tag{21}$$

$$\begin{aligned} (\boldsymbol{p}_1)_j &= (\boldsymbol{p}_4)_j = (\boldsymbol{p}_7)_j = 0, \\ (\boldsymbol{p}_2)_j &= (\boldsymbol{p}_3)_j = \frac{1}{2\mu}, \\ (\boldsymbol{p}_5)_j &= (\boldsymbol{p}_9)_j = \frac{1}{2\mu}\left(1+\cos(j\phi_\mu)\right), \\ (\boldsymbol{p}_6)_j &= \frac{1}{2\mu}\left(1+\cos(2\psi+j\phi_\mu)\right), \\ (\boldsymbol{p}_8)_j &= \frac{1}{2\mu}\left(1+\cos(2\psi-j\phi_\mu)\right), \end{aligned} \tag{22}$$

where $i \in \{1,2,...,9\}$, $j,k \in \{0,1,...,\mu-1\}$, $\mathbf{\Pi}_i^{+}$ is a circulant matrix with dimension $\mu \times \mu$, $\boldsymbol{p}_i$ is the vector of length μ that defines the circulant matrix $\mathbf{\Pi}_i^{+}$, $\mathrm{mod}(a,b)$ returns the remainder after division of a by b, $\phi_\mu = 2\pi/\mu$, $\psi = \arg\left((1+\iota\beta\sin(\phi_\mu))e^{-\iota\phi_\mu/2}\right)$, $\iota = \sqrt{-1}$, and we choose $\beta = 0.4$.

(d) As a consequence of the face extraction coefficients that have been changed by the smoothing matrix, certain edge and vertex extraction coefficients need to be updated.

First, the extraction coefficients $C^{i,11}_{a1}$, $C^{i,11}_{a2}$, and $C^{i,11}_{a5}$ with i being cyclic in $\{1, 2, ..., \mu\}$ are made equal to $C^{i,11}_{a6}$, which stores the same value for any $i \in \{1, 2, ..., \mu\}$ after application of the smoothing matrix. $C^{i,11}_{a1}$, $C^{i,11}_{a2}$, $C^{i,11}_{a5}$, and $C^{i,11}_{a6}$ with $i \in \{1, 2, ..., \mu\}$ are plotted in red in Fig. 7 (c). Second, edge extraction coefficients $C^{i,11}_{a3}$, $C^{i,11}_{a8}$, $C^{i,11}_{a9}$, and $C^{i,11}_{a14}$ with $i \in \{1, 2, ..., \mu\}$ and vertex extraction coefficients $C^{i,11}_{a4}$ and $C^{i,11}_{a13}$ with $i \in \{1, 2, ..., \mu\}$ need to be updated following Eqs. (12)-(13) and Eq. (14), respectively. In Fig. 7 (d), the aforementioned edge extraction coefficients, the aforementioned vertex extraction coefficients, and the face extraction coefficients that intervene in their recomputation are plotted in violet, green, and gray, respectively.

The final continuity of each element in $\mathbb{S}_D$ is indicated in Fig. 8 (a). Note that spline basis functions associated with 2-ring vertices of an extraordinary point satisfy that $C^{i,11}_{a6} = C^{i,11}_{a7} = C^{i,11}_{a10} = 0$ with i being cyclic in $\{1, 2, ..., \mu\}$ and, therefore, steps (c) and (d) are not needed for these basis functions.

2.5.2. Irregular and transition faces in engineering analysis

Since spline basis functions associated with both vertex-based and face-based control points are included in the basis of $\mathbb{S}_A$, it is needed to perform certain modifications to the basis functions in order to maintain partition of unity. In [39], the face-based basis functions are scaled and certain extraction coefficients of the transition basis functions are discarded to maintain partition of unity. This leads to elements with up to $8+5\mu$ basis functions with support on them. In order to decrease that number, we propose here an alternative. It consists on keeping the face-based basis functions unchanged and *truncating* the transition basis functions. This leads to elements with only up to $13+3\mu$ basis functions. These lower values of n^e near extraordinary points decrease the time spent in reading the extraction operators in the CAE program as well as assemblying and solving systems of equations due to the increased sparsity of the obtained matrices.

Truncation was first introduced in the context of hierarchical B-splines in order to recover partition of unity [78]. After that, truncation was generalized to hierarchical Catmull-Clark subdivision surfaces [86, 87], handling of T-junctions [35], hierarchical tricubic unstructured splines [88], and blended B-spines [79]. Truncation consists on discarding the contributions of basis functions that are already included in other basis functions. Here, we use the fact that a transition basis function M can be expressed as a linear combination of 16 C^1-continuous splines m_i

$$M = \sum_{i=1}^{16} c_i m_i. \tag{23}$$

The functions m_i are called the children of M and the values of the coefficients c_i are indicated in Fig. 9 (a). The children m_i associated with irregular faces are already basis functions of $\mathbb{S}_A$. These children are called *active* children and the others are called *pasive* children. The truncated basis function M^t is obtained by discarding its active children

$$M^t = \sum_{i \in \mathcal{F}_p} c_i m_i, \tag{24}$$

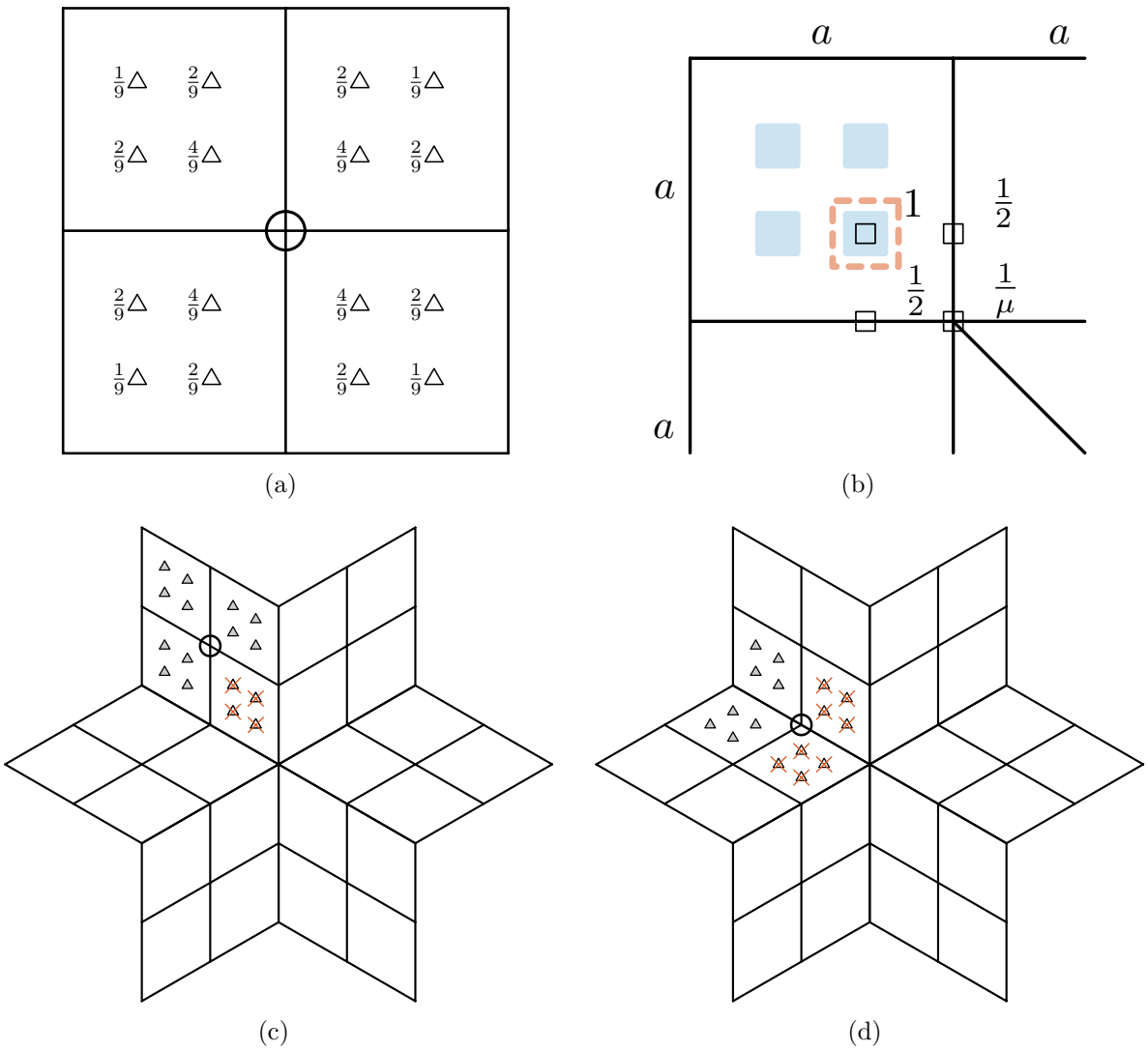


Figure 9: (a) The children m_i of a transition function M are indicated with triangles along with the coefficients c_i needed to obtain the transition function M as a linear combination of m_i. (b) The four nonzero extraction coefficients associated with each face-based spline basis function are indicated. When the vertex extraction coefficient is not an extraordinary point, μ equals 4. (c)-(d) For the two cases of transition functions, the children that are kept after truncation are plotted in gray. The crossed out children are discarded to maintain partition of unity.

where $\mathcal{F}_p$ represents the index set of pasive children.

The extraction operators are obtained following the next steps:

- Initial extraction operators for vertex-based spline functions are obtained through Eqs. (8)-(14), but discarding the contributions from irregular and transition control points.
- Face-based spline basis functions are C^1-continuous and have only four nonzero ex-

traction coefficients as shown in Fig. 9 (b). When, the vertex extraction coefficient of one of the face-based spline basis functions is not an extraordinary point, that function is just a C^1-continuous bi-cubic B-spline.

- Irregular and transition functions are truncated, that is, their active children, which are the face-based basis functions, are discarded. All the children of irregular functions are active and this is the reason why irregular functions are not included in the basis of $\mathbb{S}_A$. Two cases need to be distinguished for truncated transition functions, namely, transition basis functions associated with vertices shared by three transition faces and one irregular face and transition basis functions associated with vertices shared by two transition faces and two irregular faces. The children that need to be discarded for the two aforementioned cases are crossed out in Figs. 9 (c) and (d), respectively. The extraction coefficients of truncated transition functions are obtained multiplying the coefficients c_i with $i \in \mathcal{F}_p$ (see Eqs. (23)-(24) and Fig. 9 (a)) by the coefficients in Fig. 9 (b).

- The split-then-smoothen approach explained in steps (a)-(d) is applied to each vertex-based and face-based spline basis function with support on the 1-ring faces of the extraordinary point.

The final continuity of each element in $\mathbb{S}_A$ is indicated in Fig. 8 (b).

2.5.3. Regular faces

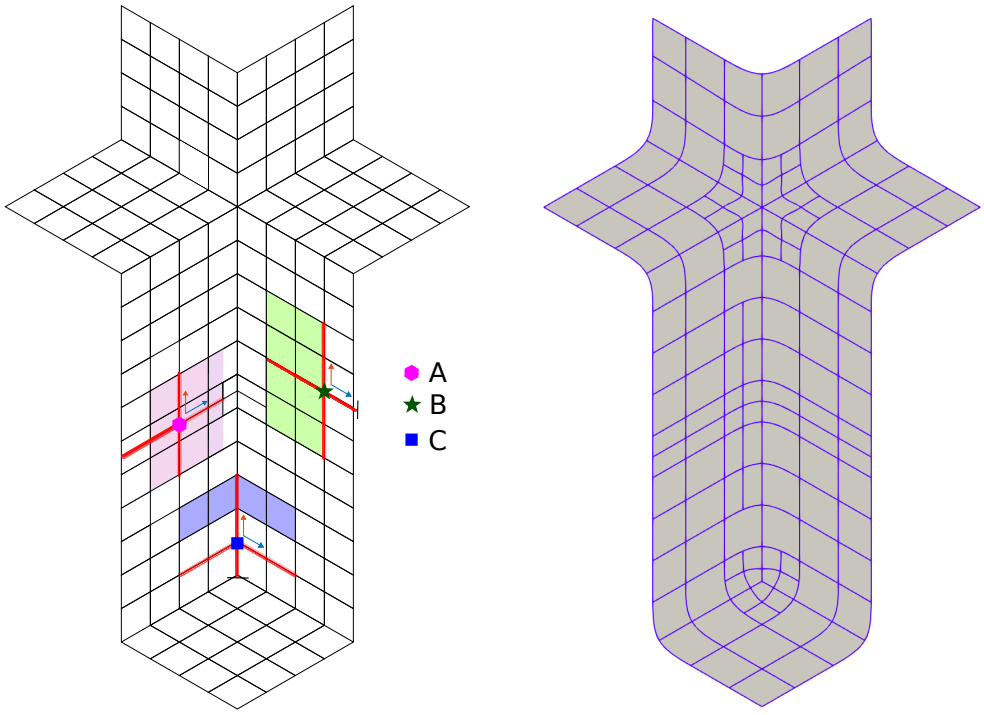
Face-based basis functions and vertex-based basis functions associated with irregular vertices do not have support on any element within regular faces. All other vertex-based basis functions have support on certain elements within regular faces. Within regular faces, each basis function is just a C^2-continuous bi-cubic B-spline defined using two local knot vectors. The support and the two local knot vectors of each basis function are obtained by shooting rays from the vertex associated with the basis function [2]. Rays are shot from each T-mesh vertex with the exception of irregular vertices. First, a local coordinate system (u, v) is assigned to the basis function, which has its origin at the vertex associated with that basis function. Then, rays are shot in the positive and negative directions defined by u and v ($\pm u$ and $\pm v$) until

- the ray hits two T-mesh edges (Case 1).

- the ray reaches the T-mesh boundary (Case 2).

- the ray reaches an extraordinary point (Case 3).

Once the rays have been shot, the local knot span vectors $\Delta\Xi = \{\Delta u_{-2}, \Delta u_{-1}, \Delta u_1, \Delta u_2\}$ and $\Delta\Theta = \{\Delta v_{-2}, \Delta v_{-1}, \Delta v_1, \Delta v_2\}$ are filled by using the knot span configuration associated with the T-mesh. Let us assume that we are filling the knot spans associated with the ray shot in the direction $-u$, namely, Δu_{-1} and Δu_{-2}. In Case 1, Δu_{-1} and Δu_{-2} are the first and second knot spans associated with the two edges traversed by the ray. In Case 2,



(a) Ray shooting (b) Bézier mesh

Figure 10: (a) The local axes assigned to basis functions A, B, and C are plotted. The rays associated with these three basis functions are represented by thick, semi-transparent red lines. Black lines orthogonal to the rays are plotted when a ray reaches the T-mesh boundary or an extraordinary point to indicate that additional knot spans need to be added in those cases. The shaded regions indicate the elements within regular faces in which the basis functions A, B, and C have support. Taking into account the knot span configuration shown in Fig. 2 (a), the local knot span vectors associated with the basis functions A, B, and C are $\Delta\Xi_A = \{0, 1, 1, 0.5\}$, $\Delta\Theta_A = \{1, 0.5, 0.5, 1\}$, $\Delta\Xi_B = \{1, 1, 0, 0\}$, $\Delta\Theta_B = \{1, 1, 1, 1\}$, and $\Delta\Xi_C = \{1, 1, 1, 1\}$, $\Delta\Theta_C = \{1, 1, 1, 1\}$. The local knot vectors associated with the basis functions A, B, and C are $\Xi_A = \{-1, -1, 0, 1, 1.5\}$, $\Theta_A = \{-1.5, -0.5, 0, 0.5, 1.5\}$, $\Xi_B = \{-2, -1, 0, 0, 0\}$, $\Theta_B = \{-2, -1, 0, 1, 2\}$, $\Xi_C = \{-2, -1, 0, 1, 2\}$, and $\Theta_C = \{-2, -1, 0, 1, 2\}$. (b) Bézier mesh associated with the T-mesh and the knot span configuration shown in Fig. 1 (a) and Fig. 2 (a), respectively.

Δu_{-2} is zero when the ray only traverses one edge and both Δu_{-1} and Δu_{-2} are zero when the ray does not traverse any edge. The zero knot spans are added to obtain an analogous boundary to the one obtained using NURBS and open knot vectors, which eases the imposition of Dirichlet boundary conditions. In Case 3, both Δu_{-1} and Δu_{-2} are equal to the knot span associated with the spoke edge traversed by the ray[§]. The knot spans associated with the rays shot in the directions $+u$, $-v$, and $+v$ are filled in an analogous manner.

[§]Since we are only using the local knot vectors to obtain the extraction operators of elements within regular faces, the particular value of Δu_{-2} has no impact on the basis function inside of regular faces. In other words, any value could be assigned to Δu_{-2} and the basis function restricted to regular faces would not change.

Examples of Cases 1, 2, and 3 are drawn in Fig. 10 (a). Fig. 10 (a) also shades the elements within regular faces where each basis function has support. Once the local knot span vectors are filled, the local knot vectors Ξ and Θ are defined as $\Xi = \{u_{-2}, u_{-1}, u_0, u_1, u_2\}$ and $\Theta = \{v_{-2}, v_{-1}, v_0, v_1, v_2\}$, where $u_0 = v_0 = 0$, $u_k - u_{k-1} = \Delta u_k$ and $v_k - v_{k-1} = \Delta v_k$ for $k \in \{1, 2\}$, and $u_{k+1} - u_k = \Delta u_k$ and $v_{k+1} - v_k = \Delta v_k$ for $k \in \{-1, -2\}$. For each element e where the basis function has support and that is within regular faces, the extraction coefficients $(C^e_{a1}, C^e_{a2}, ..., C^e_{a16})$ that define the basis function in terms of the Bernstein polynomials can be obtained by applying the knot insertion algorithm to the local knot vectors [83].

2.6. Geometry and displacement representation

The same T-spline surface (both geometrically and parametrically) is obtained using either the control points and the basis functions of $\mathbb{S}_D$ or the control points and the basis functions of $\mathbb{S}_A$. This surface is obtained by mapping each element of the elemental T-mesh into the Eucledian space as follows

$$\boldsymbol{x}^e(\boldsymbol{\xi}) = \sum_{a=1}^{n^e_D} \boldsymbol{P}^e_a N^e_a(\boldsymbol{\xi}) = \sum_{a=1}^{n^e_A} \boldsymbol{Q}^e_a M^e_a(\boldsymbol{\xi}) \quad \forall e \in \{1, 2, ..., n_{el}\}, \quad \boldsymbol{\xi} \in \square, \tag{25}$$

where n^e_D and n^e_A are the number of basis functions with support on element e in $\mathbb{S}_D$ and $\mathbb{S}_A$, respectively. The Bézier mesh is obtained by plotting the element boundaries over the T-spline surface. Fig. 10 (b) plots the Bézier mesh associated with the T-mesh and the knot span configuration shown in Fig. 1 (a) and Fig. 2 (a), respectively.

The displacement field is represented as follows

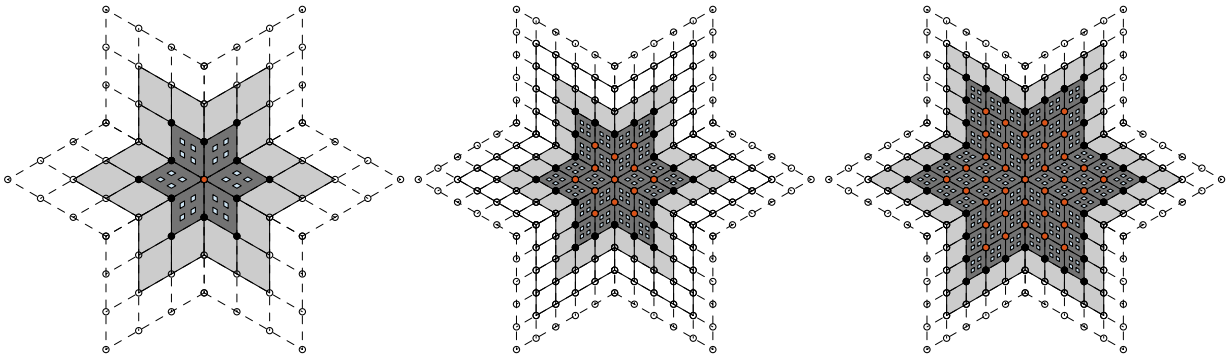
$$\boldsymbol{u}^e(\boldsymbol{\xi}) = \sum_{a=1}^{n^e_A} \boldsymbol{U}^e_a M^e_a(\boldsymbol{\xi}) \quad \forall e \in \{1, 2, ..., n_{el}\}, \quad \boldsymbol{\xi} \in \square, \tag{26}$$

where $\boldsymbol{u}^e$ is the displacement vector on the element e and $\boldsymbol{U}^e_a$ are the control variables of the displacement field that contribute to the element e. Therefore, the number of degrees of freedom (n_{dof}) is equal to three times the number of control points in $\mathbb{S}_A$.

2.7. Refinement

Refinement begins with adding vertices and edges to the T-mesh and assigning knot spans to the new edges. In order to stay within the subset of ASTS, the refinement must be done in such a way that the final T-mesh is admissible and the final knot span configuration is valid. After that, refinement involves defining the basis functions and control points on the refined T-mesh. *Global uniform refinement* is obtained when each face is bisected along its directions of nonzero parametric measure and the nonzero knot spans are divided in half. We will use global uniform refinement in the examples of this paper to show that we reach converged results.

The refinement within regular faces is the same for geometric modeling ($\mathbb{S}_D$) and engineering analysis ($\mathbb{S}_A$), but for irregular and transition faces is different. Note that when



(a) Level 0 (b) Level 1 (c) Level 1 in [39]

Figure 11: Refinement of irregular and transition faces in engineering analysis. Irregular and transition faces are shaded dark and light gray, respectively. Irregular and transition vertices are marked in red and black, respectively. Face-based control points are marked with blue squares.

either an irregular face or a transition face of an extraordinary point is refined, all the other irregular and transition faces of that extraordinary point need to be refined as well in order to stay within the subset of ASTS. As a result, this will be the only kind of refinement considered for irregular and transition faces in this section.

2.7.1. Irregular and transition faces in geometric modeling

In $\mathbb{S}_D$, the classification of faces and vertices in irregular and transition that is performed around an extraordinary point does not depend on the refinement level k around the extraordinary point (with $k = 0, 1, 2, ...$) and it is done as explained in the beginning of Section 2.5.

For the refinement level $k + 1$, the basis functions with support on irregular and/or transition faces are obtained as explained in Section 2.5.1. The control points that are not extraordinary points are obtained applying tensor products of univariate knot-insertion rules [89, 90]. When the smoothing matrix $\mathbf{\Pi}^+$ is used, an extraordinary point $\boldsymbol{P}_{ep}^{k+1}$ is obtained as follows

$$\boldsymbol{P}_{ep}^{k+1} = \frac{51}{112}\boldsymbol{P}_{ep}^{k} + \frac{3}{7\mu}\sum_{l\in\tilde{\mathcal{T}}^{v,ep}}\boldsymbol{P}_{l}^{k} + \frac{13}{112\mu}\sum_{l\in\hat{\mathcal{T}}^{v,ep}}\boldsymbol{P}_{l}^{k}, \tag{27}$$

where $\tilde{\mathcal{T}}^{v,ep}$ is the index set corresponding to transition control points that share an edge with $\boldsymbol{P}_{ep}^{k}$ and $\hat{\mathcal{T}}^{v,ep}$ is the index set corresponding to transition control points that do not share an edge with $\boldsymbol{P}_{ep}^{k}$. As shown in [39], when the control points are obtained as above, the surface stays invariant both geometrically and parametrically in transition faces and the geometric changes in irregular faces are kept to a minimum. For any refinement level k, properties (1)-(5) have been proven in [39].

2.7.2. Irregular and transition faces in engineering analysis

In $\mathbb{S}_A$, the classification of faces and vertices in irregular and transition around an extraordinary point depends on the refinement level k that is performed around the ex-

traordinary point, namely,

- Irregular faces are the 2^k-disk faces of the extraordinary point. Transition faces are the $(2^k + 1)$-ring faces of the extraordinary point. Four face-based spline control points are assigned to each irregular face.

- Irregular vertices are the $(2^k - 1)$-disk vertices of the extraordinary point. Transition vertices are the (2^k)-ring vertices of the extraordinary point. Irregular vertices are not assigned a spline control point in $\mathbb{S}_A$.

Here, we use 2^k rings of irregular faces and $2^k - 1$ rings of irregular vertices so that as we refine, the surface stays invariant both geometrically and parametrically in the transition faces of the unrefined mesh. In [39], one more ring of irregular faces and vertices was used so that as the mesh is refined, nestedness could be obtained. These two alternatives are plotted in Fig. 11. In Section 2.8, we will show how the labeling of irregular faces and vertices proposed in this work and the labeling proposed in [39] results in the same accuracy for any refinement level k while the former leads to fewer degrees of freedom.

For refinement level $k+1$, the basis functions with support on irregular and/or transition faces are obtained as explained in Section 2.5.2. The vertex-based control points that have influence on transition and/or irregular faces are obtained applying tensor products of univariate knot-insertion rules on the transition faces, where the spline control points used as input are only the vertex-based control points with influence on the transition faces of refinement level k. The face-based control points are obtained applying the Casteljau's algorithm on the irregular faces, where the Bézier control points used as input are the Bézier control points before the split-then-smoothen approach is applied. On each refined irregular face, the face-based control points are equal to the face Bézier control points. For any refinement level k, properties (1)-(5) can be proved using the same reasoning as in [39].

2.7.3. Regular faces

Within the subset of ASTS, adding T-junctions within regular faces leads to nested spaces and the surface is not changed either geometrically or parametrically [30, 33]. Algorithms to add T-junctions within regular faces are based on knot insertion and described in detail in [29]. Properties (1)-(7) have been proven in [80, 31].

2.8. Convergence rates

When defining C^1-continuous basis functions in the neighborhood of extraordinary points, one of the main challenges is to do it in such a way that the resulting spaces have optimal approximation properties [36, 91]. In this section, we numerically compute the convergence rates for the following cases:

- Truncation to maintain partition of unity, 2^k rings of irregular faces in refinement level k, and $\mathbf{\Pi}^+$ as smoothing matrix. Henceforth referred to as Case1.

- Truncation to maintain partition of unity, 2^k rings of irregular faces in refinement level k, and $\mathbf{\Pi}^\circ$ as smoothing matrix. Henceforth referred to as Case2.

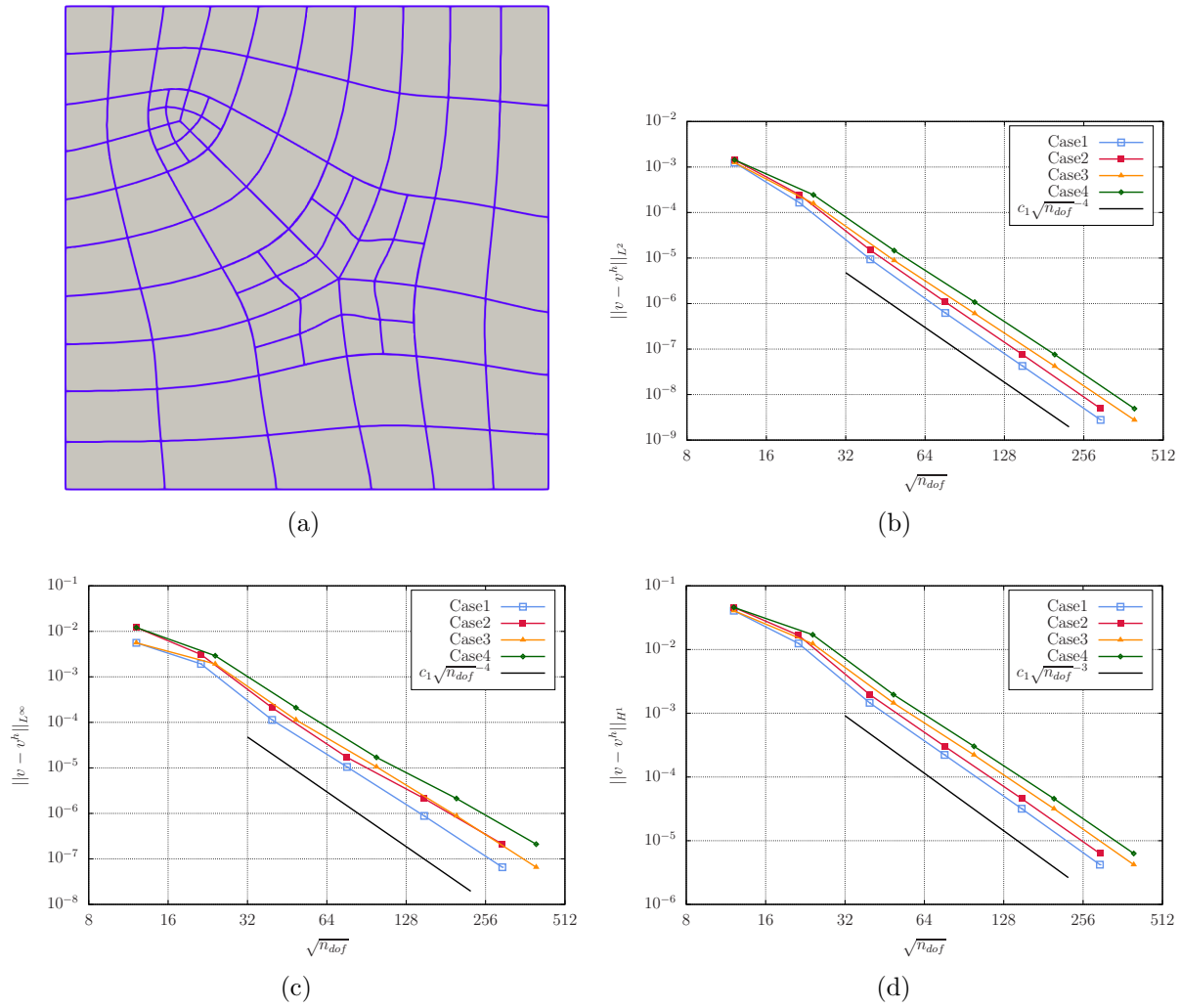


Figure 12: (a) The Bézier mesh for level 0 and Case1. (b)-(d) Convergence rates in L^2, L^∞, and H^1 norms for the Poisson equation. In (b)-(d), for a given smoothing matrix and a given refinement level k, the data point for truncation together with 2^k rings of irregular faces and the data point for scaling together with $2^{k+1}-1$ rings of irregular faces essentially lie on the same horizontal line. Therefore, Case1 and Case3 (as well as Case2 and Case4) would lead to overlapping convergence curves with respect to the mesh size h.

- Scaling to maintain partition of unity, $2^{k+1}-1$ rings of irregular faces in refinement level k, and $\mathbf{\Pi}^+$ as smoothing matrix. Henceforth referred to as Case3.

- Scaling to maintain partition of unity, $2^{k+1}-1$ rings of irregular faces in refinement level k, and $\mathbf{\Pi}^\circ$ as smoothing matrix. Henceforth referred to as Case4.

Case1 and Case2 are proposed in this work while Case3 and Case4 were proposed in [39].

We consider a unit square ($\Omega = [0,1]^2$) in which two extraordinary points with valances 3 and 5 are introduced. The Bézier mesh for level 0 and Case1 is plotted in Fig. 12

(a). After that, five levels of global uniform refinement are performed. We solve both the Poisson equation (second-order linear elliptic problem) and the biharmonic equation (fourth-order linear elliptic problem) using the approach of manufactured solutions.

The benchmark problem for the Poisson equation is defined as

$$\Delta v = g \quad \text{in} \quad \Omega, \tag{28}$$
$$v = 0 \quad \text{in} \quad \partial\Omega, \tag{29}$$
$$g = -2\pi^2 \sin(\pi x)\sin(\pi y) \quad \text{in} \quad \Omega, \tag{30}$$

for which the exact solution is $v = \sin(\pi x)\sin(\pi y)$. The convergence rates in L^2, L^∞, and H^1 norms for the four cases considered here are plotted in Figs. 12 (b),(c), and (d), respectively.

The benchmark problem for the biharmonic equation is defined as

$$\Delta^2 v = g \quad \text{in} \quad \Omega, \tag{31}$$
$$v = 0 \quad \text{in} \quad \partial\Omega, \tag{32}$$
$$\nabla v \cdot \boldsymbol{n} = 0 \quad \text{in} \quad \partial\Omega, \tag{33}$$
$$g = -16\pi^4(\cos(2\pi x) - 4\cos(2\pi x)\cos(2\pi y) + \cos(2\pi y)) \quad \text{in} \quad \Omega, \tag{34}$$

for which the exact solution is $v = (1 - \cos(2\pi x))(1 - \cos(2\pi x))$. The convergence rates in L^2, L^∞, H^1, and H^2 norms for the four cases considered here are plotted in Figs. 13 (a), (b),(c), and (d), respectively.

The main conclusions that can be taken from the convergence rates in Figs. 12 and 13 are

- For level 0, truncation and scaling lead to the same errors up to machine precision, which suggests that the spaces spanned by both sets of basis functions are the same. However, truncation leads to lower values of n^e near the extraordinary points which speeds up reading the extraction operators in a CAE program as well as assembling and solving systems of equations.

- For any refinement level k, truncation together with 2^k rings of irregular faces and scaling together with $2^{k+1}-1$ rings of irregular faces lead to indistinguishable levels of accuracy at the scale of the plots, but truncation together with 2^k rings of irregular faces has fewer degrees of freedom. Namely, for level 5, truncation together with 2^k rings of irregular faces has fewer degrees of freedom has 80.76% fewer degrees of freedom. As a result, the convergence rates with respect to the mesh size h of truncation together with 2^k rings of irregular faces and scaling together with $2^{k+1}-1$ rings of irregular faces are the same, but the convergence rates with respect to the square root of the number of degrees of freedom of truncation together with 2^k rings of irregular faces are higher.

- The nonnegative smoothing matrix leads to lower errors than the idempotent smoothing matrix.

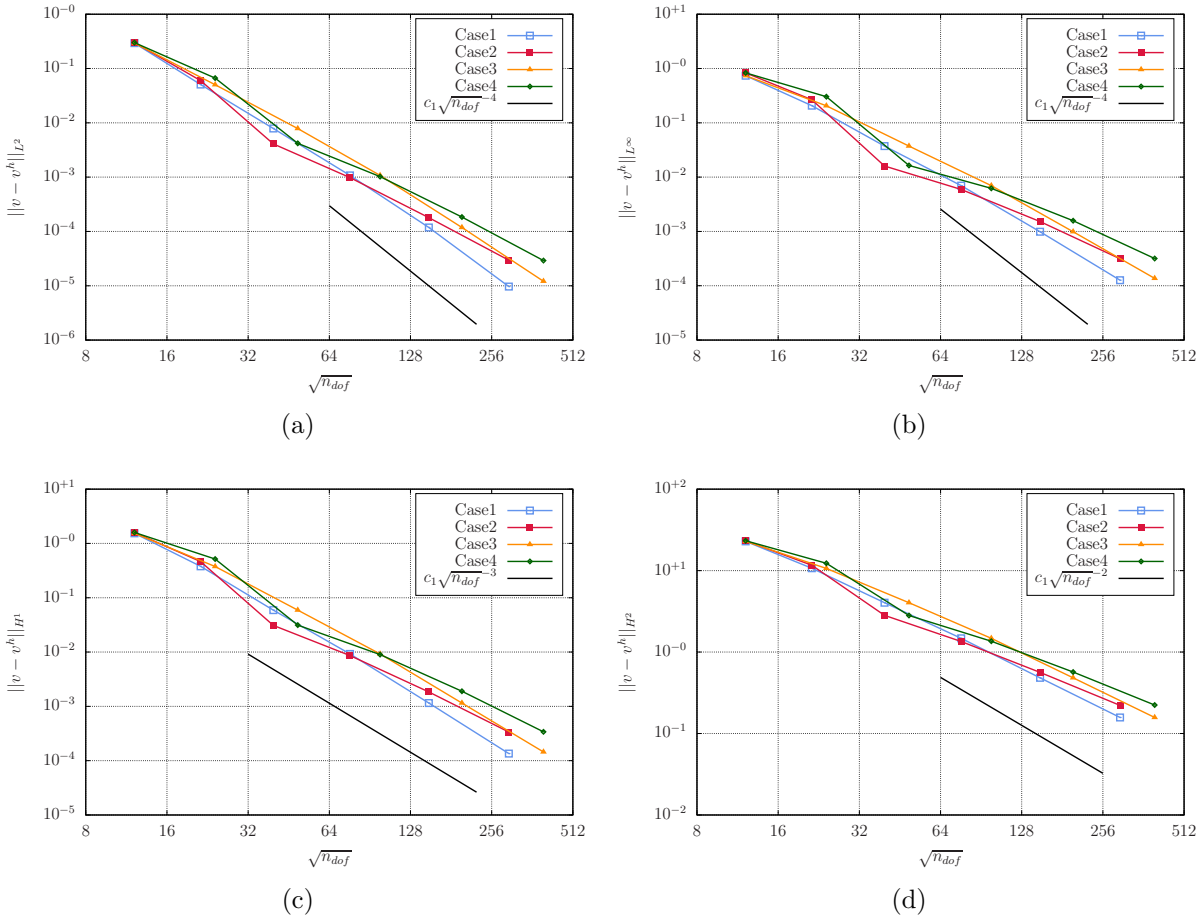


Figure 13: (a)-(d) Convergence rates in L^2, L^∞, H^1, and H^2 norms for the biharmonic equation. In (a)-(d), for a given smoothing matrix and a given refinement level k, the data point for truncation together with 2^k rings of irregular faces and the data point for scaling together with $2^{k+1}-1$ rings of irregular faces essentially lie on the same horizontal line. Therefore, Case1 and Case3 (as well as Case2 and Case4) would lead to overlapping convergence curves with respect to the mesh size h.

- Case1 performs the best in all convergence rates for both second- and fourth-order linear elliptic problems.

As a result, we favor the use of Case1 in simulations. The same conclusion will be reached in Section 4.2 when comparing the four cases in geometrically nonlinear Kirchhoff-Love shells.

Remark 2.1. As in [39], the results in Figs. 12 and 13 have been created picking $\beta = 0.4$ for all refinement levels when using $\mathbf{\Pi}^+$ and starting with $\beta = 0.4$ and dividing β by two in each refinement level when using $\mathbf{\Pi}^\circ$. Nevertheless, we have tried $\beta = 0.4$ for all refinement levels when using $\mathbf{\Pi}^\circ$ and starting with $\beta = 0.4$ and dividing β by two in each refinement level when using $\mathbf{\Pi}^+$ and the differences were completely negligible.

3. Kirchhoff-Love shells

Kirchhoff-Love shells rely on the following assumptions:

- Transverse normal stresses are neglected.
- Cross sections stay straight throughout deformation.
- A linear strain field is considered through the thickness.
- Cross sections remain normal to the midsurface during deformation. In other words, transverse shear strains are neglected. This assumption is considered to be acceptable for shells satisfying that their radii of curvature are greater than twenty times their thickness [92]. The majority of shell structures used in engineering applications satisfy this condition.

As a result of these assumptions, the kinematics of the shell can be entirely represented by its midsurface and its thickness.

In this section, we consider Kirchhoff-Love shells with large displacements and small strains, that is, we consider geometric nonlinearities, but not material nonlinearities. We derive the Kirchhoff-Love shell formulation using a Lagrangian description and curvilinear coordinates.

3.1. Kinematics

The geometry of the midsurface is expressed in the curvilinear coordinate system defined by θ^1 and θ^2. In the reference and deformed configurations, the position vector of a material point in the midsurface is denoted by $\mathring{\boldsymbol{r}}(\theta^1, \theta^2)$ and $\boldsymbol{r}(\theta^1, \theta^2)$, respectively. Therefore, the displacement vector of a material point in the midsurface is defined as

$$\boldsymbol{u}(\theta^1, \theta^2) = \boldsymbol{r}(\theta^1, \theta^2) - \mathring{\boldsymbol{r}}(\theta^1, \theta^2). \tag{35}$$

In the following, indices in Greek letters take the values $\{1, 2\}$, indices in Latin letters take the values $\{1, 2, 3\}$, and repeated indices are summed. When needed, we distinguish quantities in the reference and the deformed configurations by adding $\mathring{(\cdot)}$ to the quantity in the reference configuration. $(\cdot) \cdot (\cdot)$ and $(\cdot) \times (\cdot)$ denote the dot and cross products of vectors, respectively. $||\cdot||$ denotes the length of a vector. The determinant of a tensor is denoted by $\det(\cdot)$, whereas the determinant of a matrix is denoted by $|\cdot|$.

Non-unit tangent vectors to the midsurface in the reference and deformed configurations are obtained by

$$\mathring{\boldsymbol{a}}_\alpha = \frac{\partial \mathring{\boldsymbol{r}}}{\partial \theta^\alpha}, \qquad \boldsymbol{a}_\alpha = \frac{\partial \boldsymbol{r}}{\partial \theta^\alpha}, \tag{36}$$

respectively. Unit normal vectors to the midsurface in the reference and deformed configurations are obtained by

$$\mathring{\boldsymbol{a}}_3 = \frac{\mathring{\boldsymbol{a}}_1 \times \mathring{\boldsymbol{a}}_2}{||\mathring{\boldsymbol{a}}_1 \times \mathring{\boldsymbol{a}}_2||}, \qquad \boldsymbol{a}_3 = \frac{\boldsymbol{a}_1 \times \boldsymbol{a}_2}{||\boldsymbol{a}_1 \times \boldsymbol{a}_2||}, \tag{37}$$

respectively. Local covariant bases in the reference and deformed configuration are defined as $(\mathring{\boldsymbol{a}}_1, \mathring{\boldsymbol{a}}_2, \mathring{\boldsymbol{a}}_3)$ and $(\boldsymbol{a}_1, \boldsymbol{a}_2, \boldsymbol{a}_3)$, respectively.

The covariant metric coefficients of the midsurface in the reference and deformed configuration are defined as

$$\mathring{a}_{\alpha\beta} = \mathring{\boldsymbol{a}}_\alpha \cdot \mathring{\boldsymbol{a}}_\beta, \qquad a_{\alpha\beta} = \boldsymbol{a}_\alpha \cdot \boldsymbol{a}_\beta, \tag{38}$$

respectively. The contravariant metric coefficients can be computed as the inverse matrix of the covariant coefficients, i.e., $\left[\mathring{a}^{\alpha\beta}\right] = \left[\mathring{a}_{\alpha\beta}\right]^{-1}$ and $\left[a^{\alpha\beta}\right] = \left[a_{\alpha\beta}\right]^{-1}$.

The covariant curvature coefficients of the midsurface in the reference and deformed configuration are defined as

$$\mathring{b}_{\alpha\beta} = \frac{\partial \mathring{\boldsymbol{a}}_\alpha}{\partial \theta^\beta} \cdot \mathring{\boldsymbol{a}}_3, \qquad b_{\alpha\beta} = \frac{\partial \boldsymbol{a}_\alpha}{\partial \theta^\beta} \cdot \boldsymbol{a}_3, \tag{39}$$

respectively.

The covariant Green-Lagrange strain coefficients are obtained as follows

$$E_{\alpha\beta} = \varepsilon_{\alpha\beta} + \theta^3 \kappa_{\alpha\beta} = \frac{1}{2}(a_{\alpha\beta} - \mathring{a}_{\alpha\beta}) + \theta^3(\mathring{b}_{\alpha\beta} - b_{\alpha\beta}), \tag{40}$$

where $\varepsilon_{\alpha\beta}$ are the covariant membrane strain coefficients, $\kappa_{\alpha\beta}$ are the covariant change-of-curvature coefficients, and θ^3 is the coordinate in the thickness direction ($-0.5\,\mathring{t}(\theta^1, \theta^2) \leq \theta^3 \leq 0.5\,\mathring{t}(\theta^1, \theta^2)$ with $\mathring{t}(\theta^1, \theta^2)$ being a function that defines the shell thickness at each material point of the midsurface in the reference configuration).

3.2. St. Venant-Kirchhoff material

The St. Venant-Kirchhoff material model is an isotropic and linear elastic material that may be used when large displacements and small strains are assumed. The contravariant coefficients of the second Piola-Kirchhoff stress tensor for this material are defined as

$$S^{\alpha\beta} = \frac{E}{1-\nu^2}\left[(1-\nu)\mathring{a}^{\alpha\lambda}\,\mathring{a}^{\mu\beta}\,(\varepsilon_{\lambda\mu} + \theta^3\kappa_{\lambda\mu}) + \nu\mathring{a}^{\alpha\beta}\,\mathring{a}^{\mu\lambda}\,(\varepsilon_{\lambda\mu} + \theta^3\kappa_{\lambda\mu})\right]. \tag{41}$$

where E is the Young's modulus and ν is the Poisson's ratio.

The contravariant coefficients of the membrane forces in the reference configuration are computed as follows

$$\mathring{n}^{\alpha\beta} = \int_{-\mathring{t}/2}^{\mathring{t}/2} S^{\alpha\beta}\mathrm{d}\theta^3 = \frac{E\mathring{t}}{1-\nu^2}\left[(1-\nu)\mathring{a}^{\alpha\lambda}\,\mathring{a}^{\mu\beta}\,\varepsilon_{\lambda\mu} + \nu\mathring{a}^{\alpha\beta}\,\mathring{a}^{\mu\lambda}\,\varepsilon_{\lambda\mu}\right]. \tag{42}$$

The contravariant coefficients of the shell moments in the reference configuration are computed as follows

$$\mathring{m}^{\alpha\beta} = \int_{-\mathring{t}/2}^{\mathring{t}/2} S^{\alpha\beta}\theta^3\mathrm{d}\theta^3 = \frac{E\mathring{t}^3}{12(1-\nu^2)}\left[(1-\nu)\mathring{a}^{\alpha\lambda}\mathring{a}^{\mu\beta}\kappa_{\lambda\mu} + \nu\mathring{a}^{\alpha\beta}\mathring{a}^{\mu\lambda}\kappa_{\lambda\mu}\right]. \tag{43}$$

3.3. Variational form

The variational form can be obtained from the principle of virtual work which states that the internal virtual work (δW^{int}) must be equal to the external virtual work (δW^{ext}) for any virtual displacement ($\delta \boldsymbol{u}$), i.e.,

$$\delta W^{int} = \delta W^{ext} \quad \forall \delta \boldsymbol{u}, \tag{44}$$

with

$$\delta W^{int} = \int_{\mathring{A}} \left(\mathring{\boldsymbol{n}} : \delta \boldsymbol{\varepsilon} + \mathring{\boldsymbol{m}} : \delta \boldsymbol{\kappa} + \mathring{\rho} \mathring{t} \frac{\partial^2 \boldsymbol{u}}{\partial t^2} \cdot \delta \boldsymbol{u} \right) \mathrm{d}\mathring{A}, \tag{45}$$

$$\delta W^{ext} = \int_{\mathring{A}} \boldsymbol{f} \cdot \delta \boldsymbol{u} \, \mathrm{d}\mathring{A}, \tag{46}$$

where : represents the classical double contraction, $\delta \boldsymbol{\varepsilon}$ and $\delta \boldsymbol{\kappa}$ are the virtual membrane strain and the virtual curvature change, respectively, $\boldsymbol{f}$ is the external load applied to the shell, $\mathring{\rho}$ is the mass density in the reference configuration, and $\frac{\partial^2 \boldsymbol{u}}{\partial t^2}$ is the acceleration, $\mathring{A}$ and $\mathrm{d}\mathring{A} = \sqrt{|\mathring{a}_{\alpha\beta}|}\mathrm{d}\theta^1\mathrm{d}\theta^2$ denote the midsurface and the differential area in the reference configuration, respectively. In this section, we focus on static analysis in which the acceleration term vanishes.

3.4. Galerkin form

Invoking the isoparametric concept, we discretize and linearize Eq. (44) using the space $\mathbb{S}_A$ for both the shell geometry and the displacement field. Taking into account that the same basis functions are used for the test and trial spaces, the following contributions of element e to the residual vector $\boldsymbol{R} = \boldsymbol{R}^{int} - \boldsymbol{R}^{ext}$ and the stiffness matrix $\boldsymbol{K} = \boldsymbol{K}^{int} - \boldsymbol{K}^{ext}$ are obtained

$$R_r^{int} = \int_{\mathring{A}} \left(\mathring{n}^{\alpha\beta} \frac{\partial \varepsilon_{\alpha\beta}}{\partial U_r} + \mathring{m}^{\alpha\beta} \frac{\partial \kappa_{\alpha\beta}}{\partial U_r} \right) \mathrm{d}\mathring{A}, \tag{47}$$

$$R_r^{ext} = \int_{\mathring{A}} f_i \frac{\partial u_i}{\partial U_r} \, \mathrm{d}\mathring{A}, \tag{48}$$

$$K_{rs}^{int} = \int_{\mathring{A}} \left(\frac{\partial \mathring{n}^{\alpha\beta}}{\partial U_s} \frac{\partial \varepsilon_{\alpha\beta}}{\partial U_r} + \mathring{n}^{\alpha\beta} \frac{\partial^2 \varepsilon_{\alpha\beta}}{\partial U_r \partial U_s} + \frac{\partial \mathring{m}^{\alpha\beta}}{\partial U_s} \frac{\partial \kappa_{\alpha\beta}}{\partial U_r} + \mathring{m}^{\alpha\beta} \frac{\partial^2 \kappa_{\alpha\beta}}{\partial U_r \partial U_s} \right) \mathrm{d}\mathring{A}, \tag{49}$$

$$K_{rs}^{ext} = \int_{\mathring{A}} \frac{\partial f_i}{\partial U_s} \frac{\partial u_i}{\partial U_r} \, \mathrm{d}\mathring{A}, \tag{50}$$

where the vector $\boldsymbol{U}$ collects the control variables of the displacement field by changing first the index of the spatial dimension and then the basis function number, r and s are global degree-of-freedom numbers given by $r = 3(\mathrm{IEN}_A(a,e) - 1) + i$ and $s = 3(\mathrm{IEN}_A(b,e) - 1) + j$ with $i, j = 1, 2, 3$ referring to the global x, y, and z components, respectively, IEN_A establishes a correspondence between local and global numbering of basis functions in $\mathbb{S}_A$. Full Gauss integration rule is employed for all the integrals in Eqs. (47)-(50). Note that K_{rs}^{ext} vanishes in the case of displacement-independent loads.

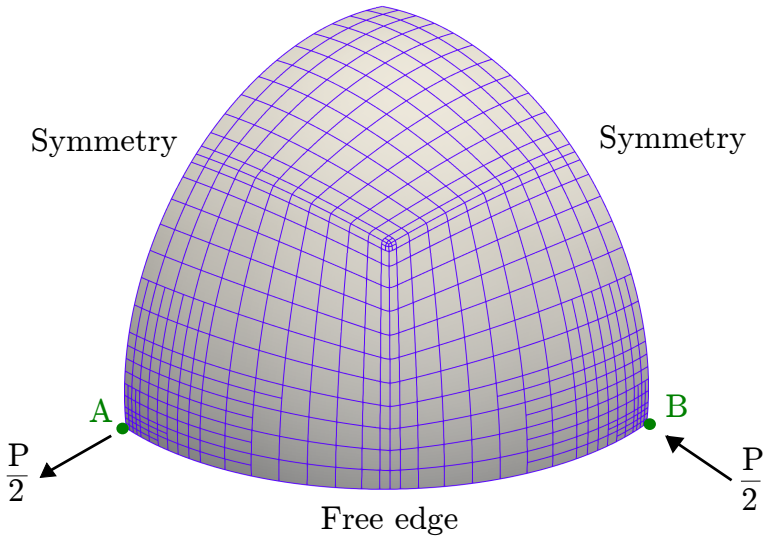


Figure 14: (Color online) Geometry, boundary conditions, and point loads applied for the pinched hemisphere. The mesh ASTS2 with 613 elements and 730 control points is also shown.

Using Eqs. (47)-(50), we solve for the linearized equation system

$$\boldsymbol{K}\,\Delta\boldsymbol{U} = -\boldsymbol{R}. \tag{51}$$

The vector $\Delta\boldsymbol{U}$ is used for the nonlinear updates of the displacement field until convergence is achieved.

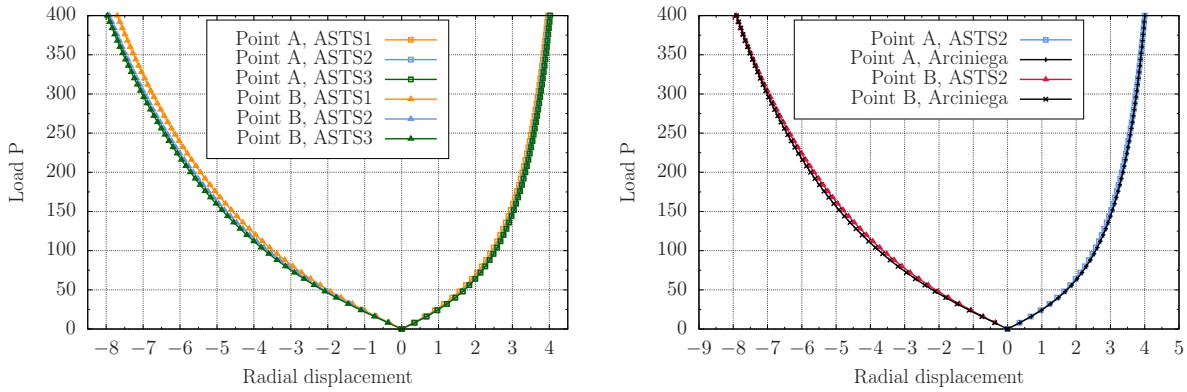
4. Numerical examples

The in-house code used to perform our simulations has been developed on top of the scientific library PETSc [93, 94]. Regarding the nonlinear and linear solvers used in our simulations, we have used the Newton-Raphson method with a critical point line search [95] and a direct solver based on LU factorization, respectively.

We solved all the examples considered in this Section using the four cases described in Section 2.8 to handle the extraordinary points. For brevity and since the conclusions are the same for the four examples considered in this Section, we only include the comparisons among the four cases in one of the four examples. For the other three examples, the results using Case1 are shown.

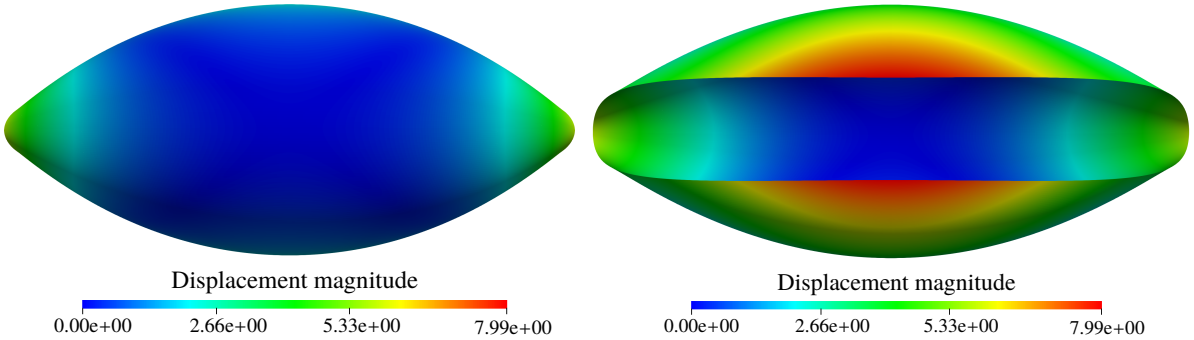
4.1. Pinched hemisphere

This benchmark problem consists in a hemisphere under two inward and two outward opposite point loads of magnitude P. The bottom circumferential edge of the hemisphere is free while it is fixed at the pole. Taking advantage of the two planes of symmetry, only a quarter of the geometry is considered for the numerical simulation. Considering one quarter of the geometry also removes rigid-body motions from this problem. This example



(a) Mesh-independence study (b) Comparison with the literature

Figure 15: (Color online) Load-deflection curves at points A and B of the pinched hemisphere. The meshes ASTS1, ASTS2, and ASTS3 have 714, 2,190, and 7,626 degrees of freedom, respectively. [96] uses a mesh with 2,268 degrees of freedom and degree 8.



(a) Top view (b) Bottom view

Figure 16: (Color online) Deformed hemisphere colored by the displacement magnitude.

has been solved in [96] using a seven parameter shell formulation that takes into account both transverse shear deformation and thickness stretching. The parameters that define this problem are

$$R = 10, \quad \mathring{t} = 0.04, \quad E = 6.825 \times 10^7, \quad \nu = 0.3, \quad P = 400, \tag{52}$$

where R is the radius of the hemisphere. The point loads are applied in 50 equal load steps.

In order to avoid having a singular point at the pole of the hemisphere, we introduce an extraordinary point with valence 3 in the ASTS used in this problem. Through the use of T-junctions, we introduce two levels of local refinement to increase the resolution near the point loads. Following that meshing procedure, we create three ASTS with 714, 2,190, and 7,626 degrees of freedom, denoted by ASTS1, ASTS2, and ASTS3, respectively. ASTS2 is plotted in Fig. 14. We monitor the radial displacement under the two point loads for each load step. In Fig. 15 a), we plot the load-deflection curves obtained with

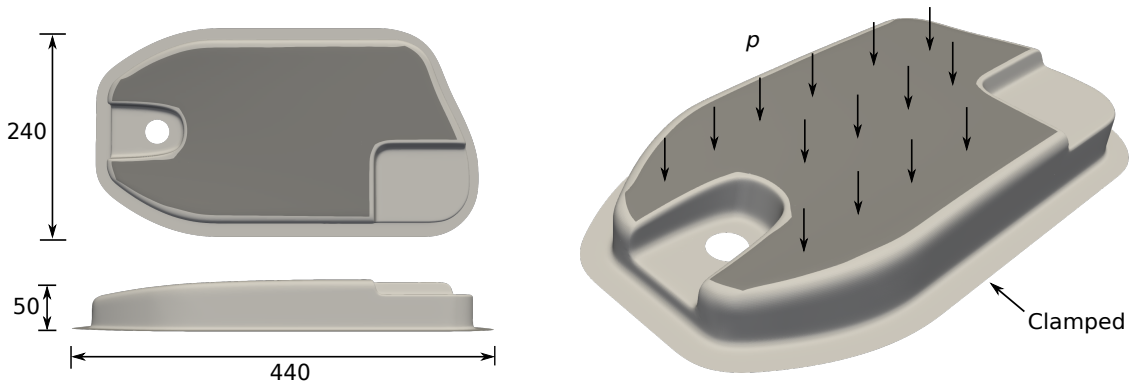


Figure 17: (Color online) Geometry, boundary conditions, and distributed load applied for the oil sump.

each of our three meshes. The results with ASTS2 and ASTS3 are nearly indistinguishable at the scale of the plot, which suggests that we have reached a converged result. Fig. 15 b) shows that the load-deflection curve obtained with ASTS2 matches the one obtained in [96]. This result shows that for a slenderness $R/t = 250$ the effects of both the transverse shear deformation and the thickness stretching, which are not considered in our Kirchhoff-Love shell formulation, are negligible. Note that ASTS2 has 2,190 degrees of freedom and polynomial degree 3 and the mesh used in [96] has 2,268 degrees of freedom and polynomial degree 8. Figs. 16 (a)-(b) plot the displacement magnitude from top and bottom views, respectively.

4.2. Oil sump

This example was defined in [75]. It considers the geometry from an oil sump of a car. Then, it clamps the external boundary and applies a distributed load on the top surface as indicated in Fig. 17. The parameters that define this problem are

$$\mathring{t} = 5.0, \quad E = 2.1 \times 10^5, \quad \nu = 0.2, \quad p = 10. \tag{53}$$

In [75], this example is solved using geometrically nonlinear Kirchhoff-Love shells and isogeometric B-Rep analysis (IBRA). IBRA is an immersed method that performs analysis on trimmed and non-watertight multipatch NURBS geometries. By introducing 9 extraordinary points with valence 3 and another 9 extraordinary points with valence 5, we build watertight and untrimmed ASTS from the original CAD geometry composed of 50 trimmed and non-watertight NURBS patches[¶]. Namely, we build one ASTS and perform global uniform refinement three times afterwards. This leads to four ASTS with 2,091, 7,716, 30,216,

[¶]Note that building watertight and untrimmed ASTS from an existent trimmed NURBS CAD geometry is done in this work to show that the current ASTS space is flexible enough to handle complex geometries. Once ASTS move to industrial settings, ASTS should be created within CAD software by designers. Currently, the commercial software Autodesk Fusion 360 enables to create watertight and untrimmed T-splines. A similar CAD capability that operates within the subset of ASTS is essential for the future of ASTS as a technology to achieve seamless integration between design and structural analysis.

Table 1: Number of degrees of freedom ($n_{dof} = 3n_{cp}$) for the four refinement levels using the four cases described in Section 2.8 to handle extraordinary points.

	Level 0	Level 1	Level 2	Level 3
Case1	6,993	26,586	103,410	407,610
Case2	6,993	26,586	103,410	407,610
Case3	6,993	30,042	125,442	513,450
Case4	6,993	30,042	125,442	513,450

Table 2: Maximum displacement magnitude obtained with the four refinement levels using the four cases described in Section 2.8 to handle extraordinary points. Case1 and Case2 lead to the same accuracy as Case3 and Case4, respectively, but with fewer degrees of freedom. Case1 has pointwise non-negative basis functions while Case2 leads to slightly negative basis functions near extraordinary points. In conclusion, we favor the use of Case1 to deal with extraordinary points.

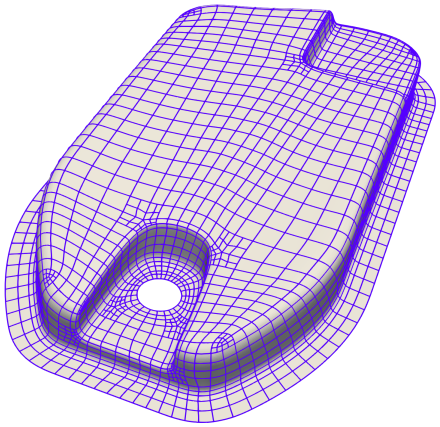
	Level 0	Level 1	Level 2	Level 3
Case1	15.101686	15.761504	15.847792	15.858248
Case2	15.056899	15.770749	15.849768	15.857084
Case3	15.101686	15.762917	15.847913	15.858225
Case4	15.056899	15.769962	15.850033	15.857087

and 120,216 elements. The Bézier mesh for the coarsest level is plotted in Fig. 18 (a). Table 1 and 2 contain the number of degrees of freedom and the maximum displacement magnitude obtained for each refinement level with the four cases described in Section 2.8 to handle extraordinary points, respectively. Out of all the examples considered in Section 4, the largest relative difference in maximum displacement magnitude between using either $\mathbf{\Pi}^+$ or $\mathbf{\Pi}^o$ is obtained for the coarsest level of this example. This relative difference is just 0.296%, which is completely negligible. Therefore, we favor the use of $\mathbf{\Pi}^+$ since it leads to pointwise non-negative basis functions. For any refinement level, Case1 and Case3 lead to the same accuracy, but Case1 has fewer degrees of freedom. Namely, for level 3, Case1 has 25.96% fewer degrees of freedom than Case3. As a result, we favor the use of truncation together with 2^k rings of irregular faces in refinement level k instead of scaling together with $2^{k+1}-1$ rings of irregular faces in refinement level k. In [75], only one simulation was considered and the maximum displacement obtained was 15.1, which is within the range of values that we obtain for the different mesh resolutions considered here.

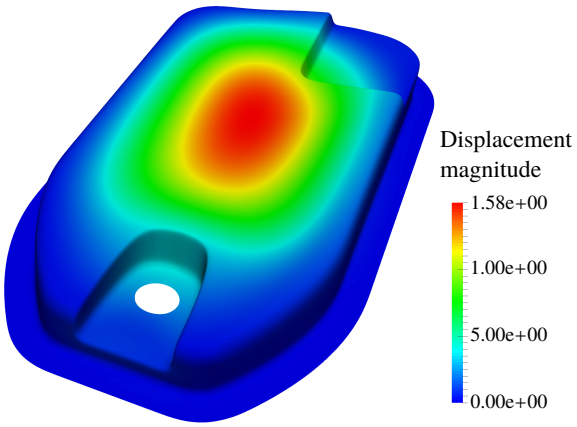
Fig. 18 (b) plots the displacement magnitude. In Figs. 18 (c)-(d), we plot the principal membrane forces (n_{max} and n_{min}). The principal membrane forces are computed as follows

$$n_{max} = \frac{\widehat{n}^{11} + \widehat{n}^{22}}{2} + \sqrt{\frac{(\widehat{n}^{11} - \widehat{n}^{22})^2}{4} + \widehat{n}^{12}\widehat{n}^{21}}, \tag{54}$$

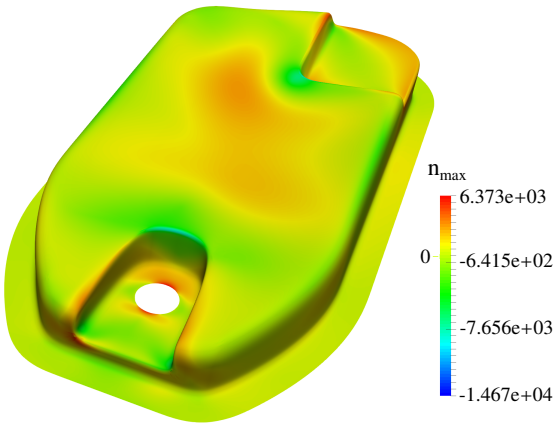
$$n_{min} = \frac{\widehat{n}^{11} + \widehat{n}^{22}}{2} - \sqrt{\frac{(\widehat{n}^{11} - \widehat{n}^{22})^2}{4} + \widehat{n}^{12}\widehat{n}^{21}}, \tag{55}$$



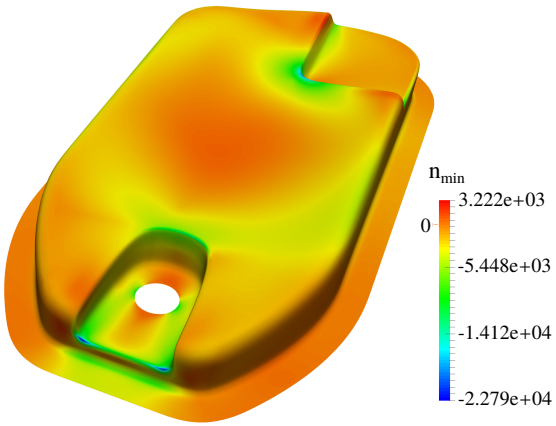
(a) The coarsest Bézier mesh



(b) Displacement Magnitude



(c) Maximum principal membrane force



(d) Minimum principal membrane force

Figure 18: (Color online) Oil sump. (a) The Bézier mesh for the coarsest level and Case1 is plotted on top of the undeformed geometry. (b) Deformed geometry colored by the displacement magnitude. (c)-(d) Deformed geometry colored by the maximum and minimum principal membrane forces, respectively.

with

$$\widehat{n}^{\alpha\beta} = \widehat{n}_{\alpha\beta} = n^{\gamma\mu}(\boldsymbol{e}^{\alpha} \cdot \boldsymbol{a}_{\gamma})(\boldsymbol{a}_{\mu} \cdot \boldsymbol{e}^{\beta}), \tag{56}$$

$$n^{\alpha\beta} = \frac{\mathring{n}^{\alpha\beta}}{J_0}, \tag{57}$$

$$J_o = \sqrt{|a_{\alpha\beta}|/|\mathring{a}_{\alpha\beta}|}, \tag{58}$$

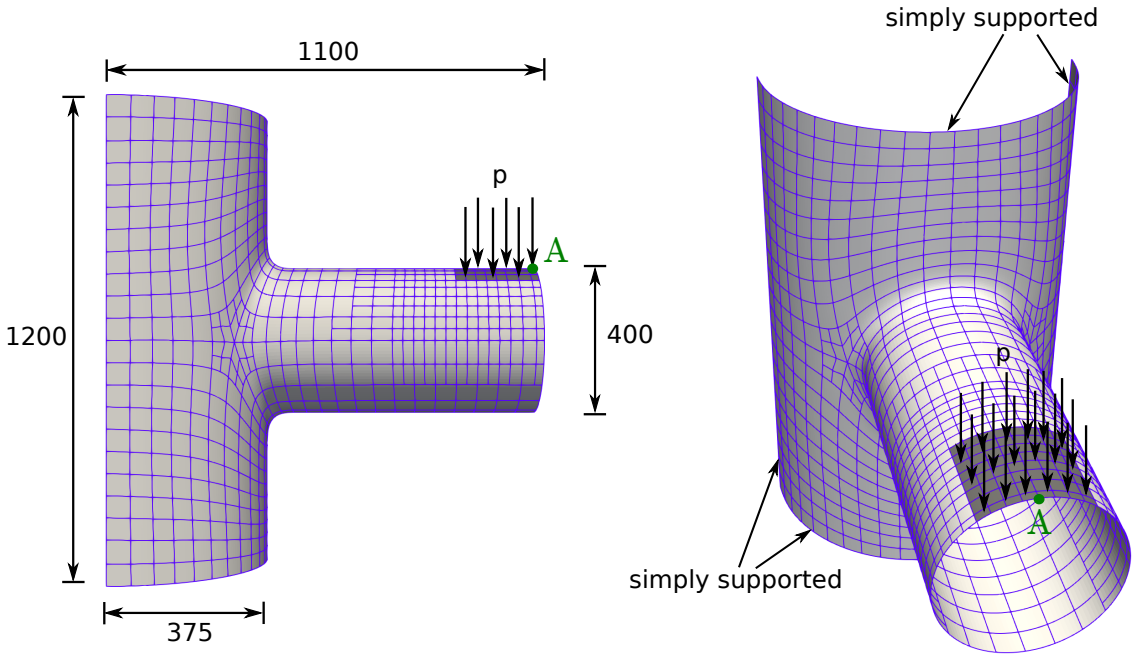


Figure 19: (Color online) Geometry, boundary conditions, and distributed load applied for the pipe junction. The mesh ASTS-1 with 1,100 elements and 1,259 control points is also shown.

$$\boldsymbol{e}_1 = \boldsymbol{e}^1 = \frac{\boldsymbol{a}_1}{||\boldsymbol{a}_1||}, \tag{59}$$

$$\boldsymbol{e}_2 = \boldsymbol{e}^2 = \frac{\boldsymbol{a}_2 - (\boldsymbol{a}_2 \cdot \boldsymbol{e}_1)\boldsymbol{e}_1}{||\boldsymbol{a}_2 - (\boldsymbol{a}_2 \cdot \boldsymbol{e}_1)\boldsymbol{e}_1||}, \tag{60}$$

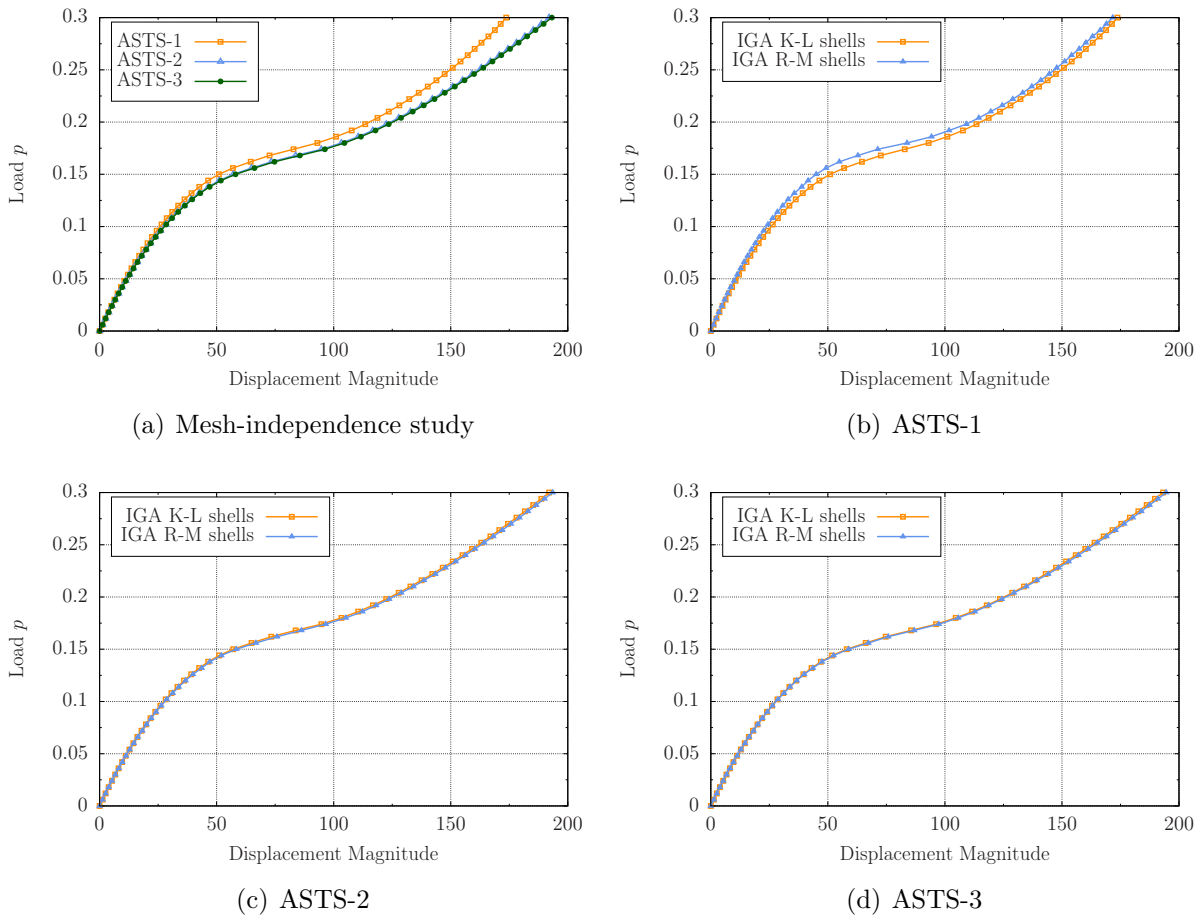
where J_o is the in-plane Jacobian determinant of the mapping from the reference to the deformed configuration, $\boldsymbol{e}_\alpha$ is the α-th base vector of a local in-plane Cartesian basis, $n^{\alpha\beta}$ are the contravariant coefficients of the membrane forces in the deformed configuration, $\widehat{n}^{\alpha\beta} = \widehat{n}_{\alpha\beta}$ are the membrane force coefficients with respect to the local in-plane Cartesian basis $\boldsymbol{e}_\alpha$.

4.3. Pipe junction

This example considers a pipe junction. The external boundary is simply supported and a symmetric distributed load is applied as indicated in Fig. 19. The parameters that define this problem are

$$\mathring{t} = 2.0, \quad E = 2.1 \times 10^5, \quad \nu = 0.25, \quad p = 0.3. \tag{61}$$

The distributed load is applied in 50 equal load steps. Two extraordinary points with valence 6 are used to mesh this geometry. By using T-junctions, we introduce one level of local refinement to increase the resolution in the region where the largest deformations



(a) Mesh-independence study

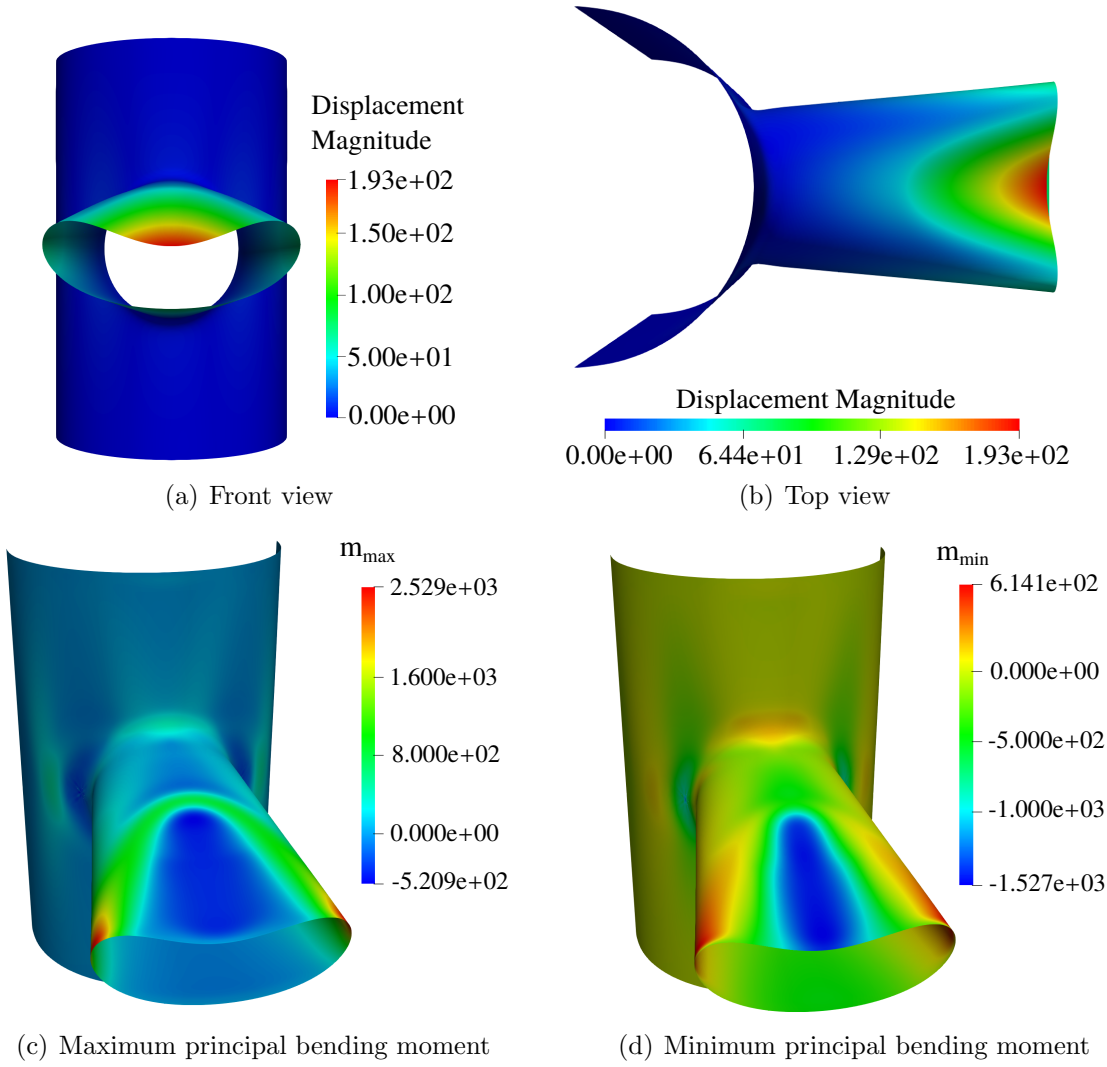
(b) ASTS-1

(c) ASTS-2

(d) ASTS-3

Figure 20: (Color online) Load-deflection curves at point A for the pipe junction. (a) As we refine our ASTS meshes, a converged result is obtained for Kirchhoff-Love shells. (b)-(d) We compare IGA discretizations of Kirchhoff-Love and Reissner-Mindlin shells for meshes ASTS-1, ASTS-2, and ASTS-3, respectively. The IGA Reissner-Mindlin simulations are performed in the commercial software LS-DYNA giving the Bézier extraction information of our ASTS meshes as an input. ASTS-1, ASTS-2, and ASTS-3 have 1,259, 4,610, and 17,612 control points, respectively. Even though, for IGA Kirchoff-Love shells, $n_{dof} = 3n_{cp}$, but for IGA Reissner-Mindlin shells, $n_{dof} = 6n_{cp}$, both shell theories lead to similar accuracy for a given ASTS mesh.

are going to take place. The ASTS that we obtained has 1,100 elements. This ASTS, denoted by ASTS-1, is plotted in Fig. 19. After that, we performed two levels of global uniform refinement obtaining ASTS with 4,190, and 16,454 elements, denoted by ASTS-2, and ASTS-3, respectively. We monitor the displacement magnitude at point A for each load step. In Fig. 20 (a), we plot the load-deflection curves obtained with each of our three meshes. ASTS-2 and ASTS-3 give indistinguishable load-deflection curves at the scale of the plot, indicating that a converged result has been reached. Besides of solving this problem with our code for Kirchhoff-Love shells, we also solved it using the commercial software LS-DYNA [97]. We imported our three ASTS meshes into LS-DYNA and compute the



(a) Front view

(b) Top view

(c) Maximum principal bending moment

(d) Minimum principal bending moment

Figure 21: (Color online) Pipe junction. (a)-(b) Displacement magnitude from front and top views, respectively. (c)-(d) Maximum and minimum principal bending moments, respectively.

load-deflection curves using the IGA-based Reissner-Mindlin formulation defined in [51]. As can be seen in Fig. 20 b)-d), good agreement is obtained with our Kirchhoff-Love code. These results show that for shells that are thin enough, Kirchhoff-Love and Reissner-Mindlin IGA discretizations results in similar levels of accuracy for a given ASTS mesh, but the Kirchhoff-Love discretization cuts the number of degrees of freedom in half. As shown in Fig. 20 b), for ASTS-1 and a given load step, IGA Kirchhoff-Love shells result in slightly higher displacement than IGA Reissner-Mindlin shells, that is, the load-deflection curve obtained with IGA Kirchhoff-Love shells is closer to the converged load-deflection curve.

Figs. 21 (a)-(b) plot the displacement magnitude from front and top views, respectively. In Figs. 21 (c)-(d), we plot the principal bending moments (m_{max} and m_{min}). The principal

bending moments are computed as follows

$$m_{max} = \frac{\widehat{m}^{11} + \widehat{m}^{22}}{2} + \sqrt{\frac{(\widehat{m}^{11} - \widehat{m}^{22})^2}{4} + \widehat{m}^{12}\widehat{m}^{21}}, \tag{62}$$

$$m_{min} = \frac{\widehat{m}^{11} + \widehat{m}^{22}}{2} - \sqrt{\frac{(\widehat{m}^{11} - \widehat{m}^{22})^2}{4} + \widehat{m}^{12}\widehat{m}^{21}}, \tag{63}$$

with

$$\widehat{m}^{\alpha\beta} = \widehat{m}_{\alpha\beta} = m^{\gamma\mu}(\boldsymbol{e}^{\alpha} \cdot \boldsymbol{a}_{\gamma})(\boldsymbol{a}_{\mu} \cdot \boldsymbol{e}^{\beta}), \tag{64}$$

$$m^{\alpha\beta} = \frac{\mathring{m}^{\alpha\beta}}{J_0}, \tag{65}$$

where $m^{\alpha\beta}$ are the contravariant coefficients of the shell moments in the deformed configuration, $\widehat{m}^{\alpha\beta} = \widehat{m}_{\alpha\beta}$ are the shell moment coefficients with respect to the local in-plane Cartesian basis $\boldsymbol{e}_{\alpha}$.

4.4. B-pillar

This example considers the B-pillar geometry of a car. The boundary conditions and the distributed load applied are indicated in Fig. 22. The parameters that define this problem are

$$\mathring{t} = 1.6, \quad E = 2.1 \times 10^5, \quad \nu = 0.25, \quad p = 18. \tag{66}$$

The distributed load is applied in 50 equal load steps. Besides of building ASTS meshes, we construct a piecewise linear quadrilateral mesh. This overkill FEM mesh is used to study whether or not our ASTS-based Kirchhoff-Love discretization gives the same result as the finite element discretization for Reissner-Mindlin shells proposed in [76], which is a workhorse shell discretization in commercial software. The original CAD geometry that we used as a reference to build our meshes was composed by 848 trimmed NURBS patches. In order to represent the 15 holes of this geometry, 31 extraordinary points with valence 5 and 16 extraordinary points with valence 6 are used in our ASTS. We also introduce T-junctions to locally refine the region near the load. The ASTS that we obtained has 26,805 degrees of freedom. This ASTS, denoted by ASTS-A, is plotted in Fig. 22. After that, we performed two levels of global uniform refinement obtaining ASTS with 99,315 and 381,063 degrees of freedom, denoted by ASTS-B and ASTS-C, respectively. For each mesh and load step, we measure the displacement magnitude under the center of the distributed load. As shown in Fig. 23 (a), the load-deflection curve obtained with ASTS-B and ASTS-C overlap completely, which indicates that a converged result has been reached. We also created a piecewise linear quadrilateral mesh with 186,876 degrees of freedom. We imported this finite-element mesh into LS-DYNA and computed the load-deflection curve using the Reissner-Mindlin discretization defined in [76]. As can be seen in Fig. 23 (b),

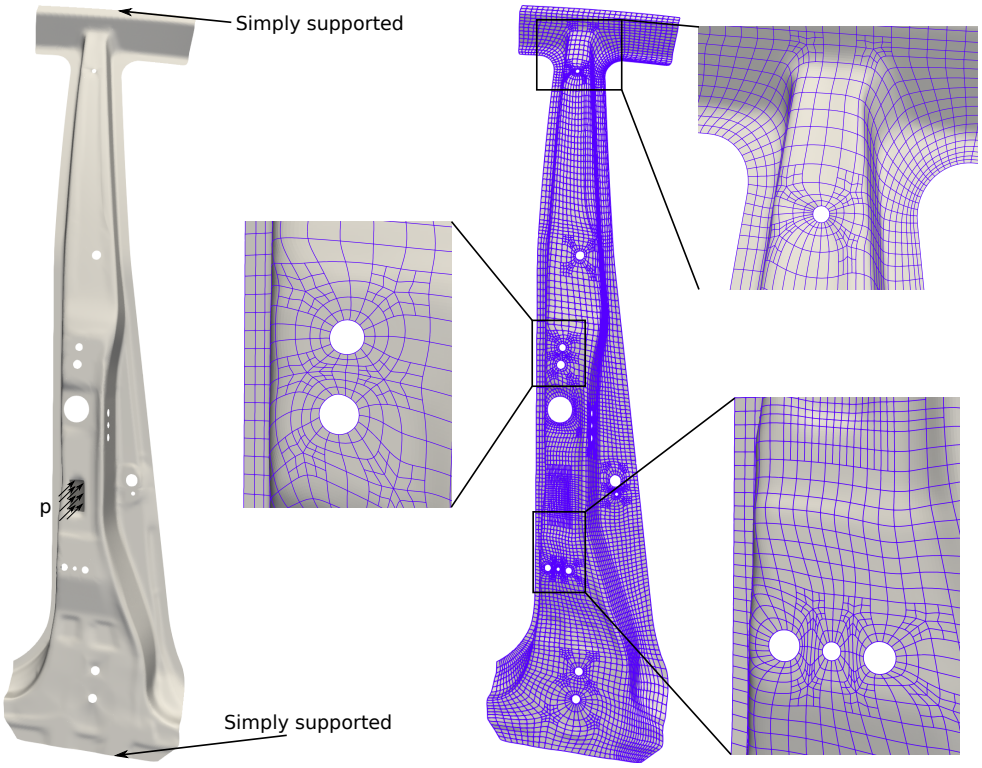
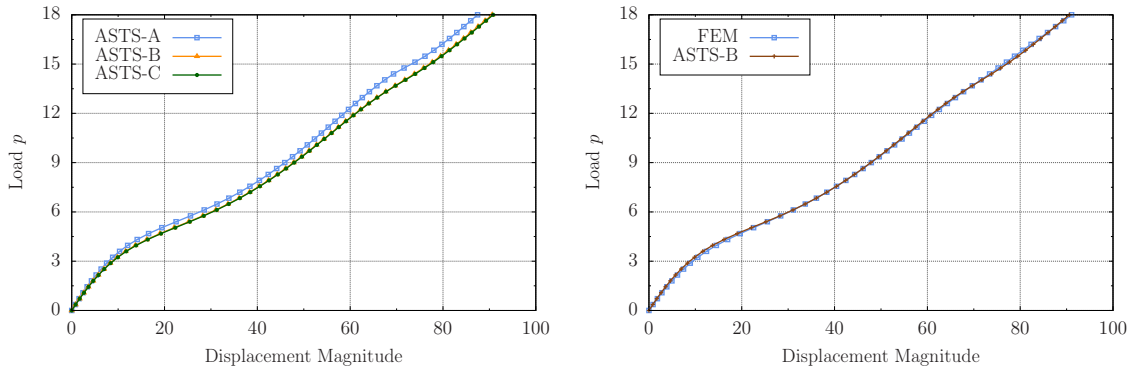


Figure 22: (Color online) Geometry, boundary conditions, and distributed load applied for the B-pillar. The coarsest Bézier mesh with 7,672 elements and 8,935 control points is shown as well.

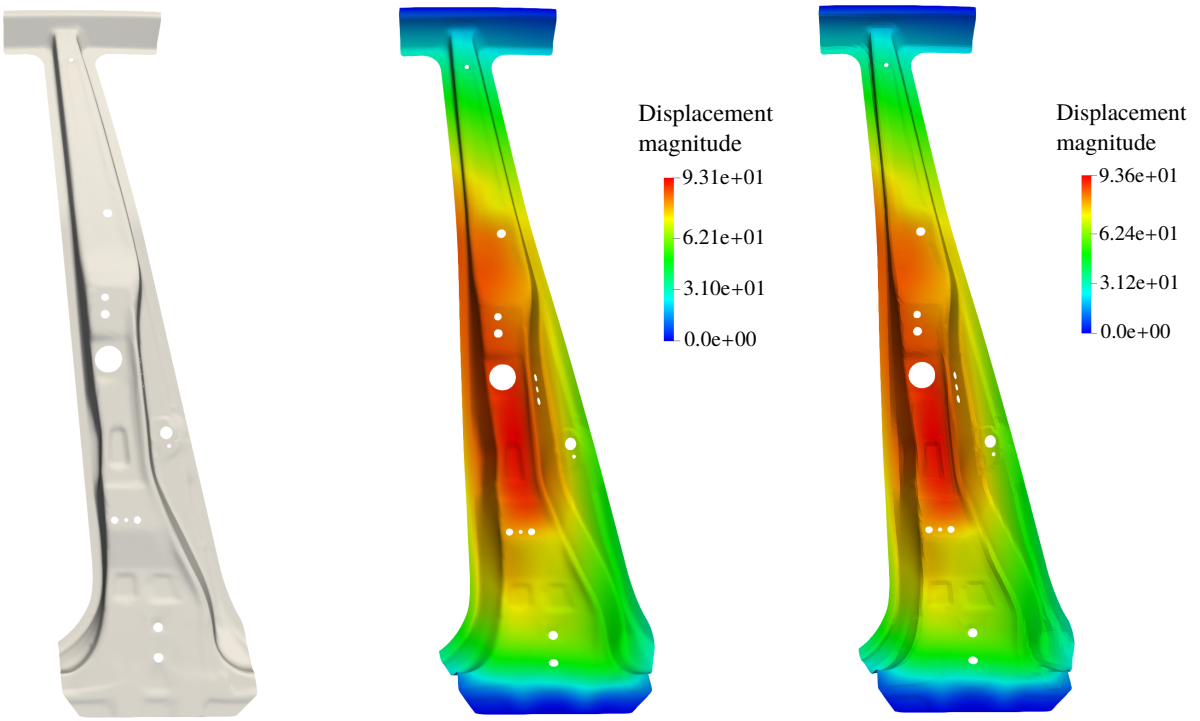


(a) Mesh-independence study

(b) Comparison with FEM

Figure 23: (Color online) Load-deflection curves under the center of the distributed load for the B-pillar. The meshes ASTS-A, ASTS-B, and ASTS-C have 26,805, 99,315, and 381,063 degrees of freedom, respectively. The FEM mesh has 186,876 degrees of freedom.

good agreement is obtained between our ASTS-based discretization of Kirchhoff-Love shells and the well-established FEM-based discretization of Reissner-Mindlin shells developed in [76].



(a) Geometry (b) IGA result (c) FEM result

Figure 24: (Color online) B-pillar. (a) Undeformed geometry front view. (b) Deformed geometry front view using ASTS-based IGA discretization of Kirchhoff-Love shells. (c) Deformed geometry front view using the finite-element discretization of Reissner-Mindlin shells proposed in [76]. Both the IGA and FEM results are plotted without using any kind of smoothing in the post-processing.

In Figs. 24 (a)-(c), we plot front views of the underformed geometry, the deformed geometry colored by the displacement magnitude using Kirchhoff-Love shells and ASTS discretization, and the deformed geometry colored by the displacement magnitude using Reissner-Mindlin shells and the finite element discretization detailed in [76], respectively.

5. Conclusions and future work

We presented a complete description of the construction of C^1-continuous bi-cubic, non-negative, T-splines on unstructured quadrilateral meshes, including extraordinary points, that delivers optimal convergence rates with respect to both mesh size and the square root of the number of degrees of freedom. This is the primary technical contribution of this work. These were applied to several doubly curved shell configurations, including geometries from real world engineering applications. For the hemisphere and the oil sump, ASTS-based IGA simulations of Kirchhoff-Love shells are compared with the literature. For the pipe junction and the B-pillar, we used LS-DYNA to perform simulations of Reissner-Mindlin shells to compare with our ASTS-based IGA simulations of Kirchhoff-Love shells. The results coincided in all cases, which seems to confirm that transverse shear deformations

are negligible in thin shells. These examples show the potential of ASTS including both extraordinary points and T-junctions to define shell geometries with arbitrary topology. The philosophy behind our work is to build spline spaces that possess all the important mathematical and geometric properties needed in either design or analysis in such a way that transferring geometric information from one space to the other is an automatic process for which explicit formulas are available [39]. This vision would conduce to a much more efficient and reliable "design to analysis" cycle in complex engineering applications.

Four interesting directions of future research are:

- Developing a smoothing matrix that has non-negative entries and is idempotent at the same time. This smoothing matrix would lead to non-negative basis functions and refinement algorithms in $\mathbb{S}_A$ for which the surface stays unchanged both parametrically and geometrically (currently, slight changes take place in the 1-ring of extraordinary points). In case this smoothing matrix was combined with $2^{k+1}-1$ rings of irregular faces (instead of 2^k rings of irregular faces) in refinement level k, nestedness would also be obtained. However, our convergence studies in Section 2.8 advise against adding this additional ring of irregular faces since the number of degrees of freedom in refinement level k increases, but the accuracy with which the solution is approximated stays the same.

- Extruding our current ASTS surfaces in the through-thickness direction to obtain C^1-continuous analysis-suitable volumetric spline meshes that can be used in solid simulations. Note that constructing C^1-continuous volumetric spline meshes with optimal convergence rates for unstructured meshes that cannot be obtained by extruding a surface is an open problem and there is no direct way to extend the algorithms included in this paper to that scenario.

- Developing a construction of extraordinary points based on geometric smoothness (G^1) [98] instead of parametric smoothness (C^1) that satisfies all the geometric and mathematical properties obtained by the construction proposed in this paper.

- The primal formulation based on the minimum potential energy theorem and the Galerkin method detailed in Eqs. (44)-(50) is not intrinsically free from locking. As a result, this Kirchhoff-Love shell formulation may undergo membrane locking when very coarse discretizations are used for highly slender shells [61]. In [99], the mixed displacement (MD) method, a mixed formulation that is intrinsically free from locking, was recently proposed. Coming up with constraint conditions that enable the application of the MD method to unstructured ASTS meshes so as to compare its performance with the primal formulation used in this paper is worth pursuing.

Acknowledgements

H. Casquero, X. Wei, A. Li, and Y. Zhang were supported in part by the PECASE Award N00014-16-1-2254, NSF CAREER Award OCI-1149591, and Honda funds. T.J.R.

Hughes and D. Toshniwal were partially supported by the Office of Naval Research, USA (Grant Nos. N00014-17-1-2119 and N00014-13-1-0500). J. Kiendl was supported by the Onsager fellowship program of NTNU. This work used the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation grant number OCI-1053575. Specifically, it used the Bridges system, which is supported by NSF award number ACI-1445606, at the Pittsburgh Supercomputing Center (PSC). We also acknowledge the open source scientific library PETSc and its developers.

The authors would like to thank M. Breitenberger and K. Takada for giving us the CAD geometries of the oil sump and the B-pillar, respectively. The authors would also like to thank A. P. Nagy and D. J. Benson for various fruitful discussions about the commercial software LS-DYNA.

References

[1] D. F. Rogers, An introduction to NURBS: With historical perspective, Morgan Kaufmann, San Francisco, 2001.

[2] T. W. Sederberg, J. Zheng, A. Bakenov, A. Nasri, T-splines and T-NURCCs, ACM Trans. Graph. 22 (2003) 477–484.

[3] T. W. Sederberg, D. L. Cardon, G. T. Finnigan, N. S. North, J. Zheng, T. Lyche, T-spline simplification and local refinement, in: ACM Transactions on Graphics (TOG), Vol. 23, ACM, 2004, pp. 276–283.

[4] T. W. Sederberg, G. T. Finnigan, X. Li, H. Lin, H. Ipson, Watertight trimmed NURBS, in: ACM SIGGRAPH 2008 Papers, New York, NY, USA, 2008, pp. 79:1–79:8.

[5] B. Marussig, T. J. Hughes, A review of trimming in isogeometric analysis: challenges, data exchange and simulation aspects, Archives of computational methods in engineering 25 (4) (2018) 1059–1127.

[6] L. Liu, Y. J. Zhang, X. Wei, Weighted T-splines with application in reparameterizing trimmed NURBS surfaces, Computer Methods in Applied Mechanics and Engineering 295 (2015) 108 – 126.

[7] J. A. Cottrell, T. J. R. Hughes, Y. Bazilevs, Isogeometric analysis toward integration of CAD and FEA, Wiley, 2009.

[8] M. Hardwick, R. Clay, P. Boggs, E. Walsh, A. Larzelere, A. Altshuler, Dart system analysis, Sandia National Laboratories, Albuquerque, New Mexico 87185.

[9] T. J. R. Hughes, The finite element method: linear static and dynamic finite element analysis, Courier Corporation, 2012.

[10] Y. Bazilevs, L. Beirao da Veiga, J. A. Cottrell, T. J. Hughes, G. Sangalli, Isogeometric analysis: approximation, stability and error estimates for h-refined meshes, Mathematical Models and Methods in Applied Sciences 16 (07) (2006) 1031–1090.

[11] T. J. Hughes, A. Reali, G. Sangalli, Duality and unified analysis of discrete approximations in structural dynamics and wave propagation: comparison of p-method finite elements with k-method NURBS, Computer methods in applied mechanics and engineering 197 (49-50) (2008) 4104–4124.

[12] J. A. Evans, Y. Bazilevs, I. Babuška, T. J. Hughes, n-widths, sup-infs, and optimality ratios for the k-version of the isogeometric finite element method, Computer Methods in Applied Mechanics and Engineering 198 (2009) 1726–1741.

[13] L. B. Da Veiga, A. Buffa, J. Rivas, G. Sangalli, Some estimates for h-p-k-refinement in isogeometric analysis, Numerische Mathematik 118 (2) (2011) 271–305.

[14] H. Gomez, V. M. Calo, Y. Bazilevs, T. J. R. Hughes, Isogeometric analysis of the Cahn-Hilliard phase-field model, Computer Methods in Applied Mechanics and Engineering 197 (49-50) (2008) 4333–4352.

[15] J. Kiendl, K.-U. Bletzinger, J. Linhard, R. Wuchner, Isogeometric shell analysis with Kirchhoff-Love elements, Computer Methods in Applied Mechanics and Engineering 198 (2009) 3902–3914.

[16] H. Gomez, T. J. R. Hughes, X. Nogueira, V. M. Calo, Isogeometric analysis of the isothermal Navier–Stokes–Korteweg equations, Computer Methods in Applied Mechanics and Engineering 199 (2010) 1828–1840.

[17] J. Liu, C. M. Landis, H. Gomez, T. J. R. Hughes, Liquid-vapor phase transition: Thermomechanical theory, entropy stable numerical formulation, and boiling simulations, Computer Methods in Applied Mechanics and Engineering.

[18] J. Bueno, H. Casquero, Y. Bazilevs, H. Gomez, Three-dimensional dynamic simulation of elastocapillarity, Meccanica 53 (6) (2018) 1221–1237.

[19] S. Lipton, J. Evans, Y. Bazilevs, T. Elguedj, T. J. R. Hughes, Robustness of isogeometric structural discretizations under severe mesh distortion, Computer Methods in Applied Mechanics and Engineering 199 (2010) 357–373.

[20] J. A. Cottrell, A. Reali, Y. Bazilevs, T. J. R. Hughes, Isogeometric analysis of structural vibrations, Computer Methods in Applied Mechanics and Engineering 195 (2006) 5257–5296.

[21] A. Buffa, C. De Falco, G. Sangalli, Isogeometric analysis: stable elements for the 2D Stokes equation, International Journal for Numerical Methods in Fluids 65 (11-12) (2011) 1407–1422.

[22] A. Buffa, G. Sangalli, R. Vázquez, Isogeometric analysis in electromagnetics: B-splines approximation, Computer Methods in Applied Mechanics and Engineering 199 (17-20) (2010) 1143–1152.

[23] J. A. Evans, T. J. Hughes, Isogeometric divergence-conforming B-splines for the unsteady Navier–Stokes equations, Journal of Computational Physics 241 (2013) 141–167.

[24] H. Casquero, Y. J. Zhang, C. Bona-Casas, L. Dalcin, H. Gomez, Non-body-fitted fluid–structure interaction: Divergence-conforming B-splines, fully-implicit dynamics, and variational formulation, Journal of Computational Physics 374 (2018) 625–653.

[25] F. Auricchio, L. Beirao Da Veiga, T. J. R. Hughes, A. Reali, G. Sangalli, Isogeometric collocation methods, Mathematical Models and Methods in Applied Sciences 20 (2010) 2075–2107.

[26] H. Gomez, L. De Lorenzis, The variational collocation method, Computer Methods in Applied Mechanics and Engineering 309 (2016) 152–181.

[27] A. Buffa, D. Cho, G. Sangalli, Linear independence of the T-spline blending functions associated with some particular T-meshes, Computer Methods in Applied Mechanics and Engineering 199 (23-24) (2010) 1437–1445.

[28] X. Li, J. Zheng, T. W. Sederberg, T. J. R. Hughes, M. A. Scott, On linear independence of T-spline blending functions, Computer Aided Geometric Design 29 (2012) 63–76.

[29] M. Scott, X. Li, T. Sederberg, T. J. R. Hughes, Local refinement of analysis-suitable T-splines, Computer Methods in Applied Mechanics and Engineering 213-216 (2012) 206–222.

[30] X. Li, M. A. Scott, Analysis-suitable T-splines: characterization, refineability, and approximation, Mathematical Models and Methods in Applied Sciences 24 (06) (2014) 1141–1164.

[31] L. Beirao da Veiga, A. Buffa, G. Sangalli, R. Vazquez, Analysis suitable T-splines of arbitrary degree: Definition, linear independence, and approximation properties, Mathematical Models and Methods in Applied Sciences 23 (11) (2013) 1979–2003.

[32] L. Beirao da Veiga, A. Buffa, D. Cho, G. Sangalli, Analysis-suitable T-splines are dual-compatible, Computer methods in applied mechanics and engineering 249 (2012) 42–51.

[33] A. Bressan, A. Buffa, G. Sangalli, Characterization of analysis-suitable T-splines, Computer Aided Geometric Design 39 (2015) 17–49.

[34] H. Casquero, L. Liu, Y. Zhang, A. Reali, H. Gomez, Isogeometric collocation using analysis-suitable T-splines of arbitrary degree, Computer Methods in Applied Mechanics and Engineering 301 (2016) 164 – 186.

[35] X. Wei, Y. Zhang, L. Liu, T. J. Hughes, Truncated T-splines: Fundamentals and methods, Computer Methods in Applied Mechanics and Engineering 316 (2017) 349–372.

[36] A. Collin, G. Sangalli, T. Takacs, Analysis-suitable G^1 multi-patch parametrizations for C^1 isogeometric spaces, Computer Aided Geometric Design 47 (2016) 93–113.

[37] M. Kapl, G. Sangalli, T. Takacs, Construction of analysis-suitable G^1 planar multi-patch parameterizations, Computer-Aided Design 97 (2018) 41–55.

[38] M. Kapl, G. Sangalli, T. Takacs, The Argyris isogeometric space on unstructured multi-patch planar domains, arXiv preprint arXiv:1711.05161.

[39] D. Toshniwal, H. Speleers, T. J. Hughes, Smooth cubic spline spaces on unstructured quadrilateral meshes with particular emphasis on extraordinary points: Geometric design and isogeometric analysis considerations, Computer Methods in Applied Mechanics and Engineering 327 (2017) 411–458.

[40] J. Kiendl, Y. Bazilevs, M.-C. Hsu, R. Wüchner, K.-U. Bletzinger, The bending strip method for isogeometric analysis of Kirchhoff–Love shell structures comprised of multiple patches, Computer Methods in Applied Mechanics and Engineering 199 (2010) 2403–2416.

[41] D. Benson, Y. Bazilevs, M.-C. Hsu, T. J. R. Hughes, A large deformation, rotation-free, isogeometric shell, Computer Methods in Applied Mechanics and Engineering 200 (13) (2011) 1367–1378.

[42] A. P. Nagy, S. T. IJsselmuiden, M. M. Abdalla, Isogeometric design of anisotropic shells: optimal form and material distribution, Computer Methods in Applied Mechanics and Engineering 264 (2013) 145–162.

[43] J. Kiendl, M.-C. Hsu, M. C. Wu, A. Reali, Isogeometric Kirchhoff–Love shell formulations for general hyperelastic materials, Computer Methods in Applied Mechanics and Engineering 291 (2015) 280–303.

[44] A. B. Tepole, H. Kabaria, K.-U. Bletzinger, E. Kuhl, Isogeometric kirchhoff–love shell formulations for biological membranes, Computer methods in applied mechanics and engineering 293 (2015) 328–347.

[45] T. X. Duong, F. Roohbakhshan, R. A. Sauer, A new rotation-free isogeometric thin shell formulation and a corresponding continuity constraint for patch boundaries, Computer Methods in Applied Mechanics and Engineering 316 (2017) 43–83.

[46] F. Roohbakhshan, R. A. Sauer, Efficient isogeometric thin shell formulations for soft biological materials, Biomechanics and modeling in mechanobiology 16 (5) (2017) 1569–1597.

[47] N. Nguyen-Thanh, N. Valizadeh, M. Nguyen, H. Nguyen-Xuan, X. Zhuang, P. Areias, G. Zi, Y. Bazilevs, L. De Lorenzis, T. Rabczuk, An extended isogeometric thin shell analysis based on Kirchhoff–Love theory, Computer Methods in Applied Mechanics and Engineering 284 (2015) 265–291.

[48] X. Deng, A. Korobenko, J. Yan, Y. Bazilevs, Isogeometric analysis of continuum damage in rotation-free composite shells, Computer Methods in Applied Mechanics and Engineering 284 (2015) 349–372.

[49] Y. Bazilevs, M. Pigazzini, A. Ellison, H. Kim, A new multi-layer approach for progressive damage simulation in composite laminates based on isogeometric analysis and kirchhoff–love shells. Part I: basic theory and modeling of delamination and transverse shear, Computational Mechanics 62 (3) (2018) 563–585.

[50] M. Ambati, J. Kiendl, L. De Lorenzis, Isogeometric Kirchhoff–Love shell formulation for elasto-plasticity, Computer Methods in Applied Mechanics and Engineering.

[51] D. Benson, Y. Bazilevs, M. Hsu, T. J. R. Hughes, Isogeometric shell analysis: The Reissner-Mindlin shell, Computer Methods in Applied Mechanics and Engineering 199 (2010) 276–289.

[52] C. H. Thai, H. Nguyen-Xuan, N. Nguyen-Thanh, T.-H. Le, T. Nguyen-Thoi, T. Rabczuk, Static, free vibration, and buckling analysis of laminated composite Reissner–Mindlin plates using NURBS-based isogeometric approach, International Journal for Numerical Methods in Engineering 91 (6) (2012) 571–603.

[53] W. Dornisch, S. Klinkel, B. Simeon, Isogeometric Reissner–Mindlin shell analysis with exactly calculated director vectors, Computer Methods in Applied Mechanics and Engineering 253 (2013) 491–504.

[54] B. Oesterle, E. Ramm, M. Bischoff, A shear deformable, rotation-free isogeometric shell formulation, Computer Methods in Applied Mechanics and Engineering 307 (2016) 235–255.

[55] B. Oesterle, R. Sachse, E. Ramm, M. Bischoff, Hierarchic isogeometric large rotation shell elements including linearized transverse shear parametrization, Computer Methods in Applied Mechanics and Engineering 321 (2017) 383–405.

[56] R. Bouclier, T. Elguedj, A. Combescure, Efficient isogeometric NURBS-based solid-shell elements: mixed formulation and B-method, Computer Methods in Applied Mechanics and Engineering 267 (2013) 86–110.

[57] R. Bouclier, T. Elguedj, A. Combescure, On the development of NURBS-based isogeometric solid shell elements: 2D problems and preliminary extension to 3D, Computational Mechanics 52 (5) (2013) 1085–1112.

[58] S. Hosseini, J. J. Remmers, C. V. Verhoosel, R. De Borst, An isogeometric solid-like shell element for nonlinear analysis, International Journal for Numerical Methods in Engineering 95 (3) (2013) 238–256.

[59] S. Hosseini, J. J. Remmers, C. V. Verhoosel, R. De Borst, An isogeometric continuum shell element for non-linear analysis, Computer Methods in Applied Mechanics and Engineering 271 (2014) 1–22.

[60] R. Bouclier, T. Elguedj, A. Combescure, An isogeometric locking-free NURBS-based solid-shell element for geometrically nonlinear analysis, International Journal for Numerical Methods in Engineering 101 (10) (2015) 774–808.

[61] R. Echter, B. Oesterle, M. Bischoff, A hierarchic family of isogeometric shell finite elements, Computer Methods in Applied Mechanics and Engineering 254 (2013) 170–180.

[62] D. Benson, S. Hartmann, Y. Bazilevs, M.-C. Hsu, T. J. R. Hughes, Blended isogeometric shells, Computer Methods in Applied Mechanics and Engineering 255 (2013) 133 – 146.

[63] H. Casquero, L. Liu, Y. Zhang, A. Reali, J. Kiendl, H. Gomez, Arbitrary-degree T-splines for isogeometric analysis of fully nonlinear Kirchhoff-Love shells, Computer-Aided Design 82 (2017) 140–153.

[64] F. Cirak, M. Ortiz, P. Schröder, Subdivision surfaces: a new paradigm for thin-shell finite-element analysis, International Journal for Numerical Methods in Engineering 47 (12) (2000) 2039–2072.

[65] F. Cirak, M. Ortiz, Fully C^1-conforming subdivision elements for finite deformation thin-shell analysis, International Journal for Numerical Methods in Engineering 51 (7) (2001) 813–833.

[66] F. Cirak, M. J. Scott, E. K. Antonsson, M. Ortiz, P. Schröder, Integrated modeling, finite-element analysis, and engineering design for thin-shell structures using subdivision, Computer-Aided Design 34 (2) (2002) 137–148.

[67] F. Cirak, M. Ortiz, A. Pandolfi, A cohesive approach to thin-shell fracture and fragmentation, Computer Methods in Applied Mechanics and Engineering 194 (21) (2005) 2604 – 2618.

[68] F. Cirak, Q. Long, Subdivision shells with exact boundary control and non-manifold geometry, International Journal for Numerical Methods in Engineering 88 (9) (2011) 897–923.

[69] Q. Long, P. Burkhard Bornemann, F. Cirak, Shear-flexible subdivision shells, International Journal for Numerical Methods in Engineering 90 (13) (2012) 1549–1577.

[70] K. Bandara, F. Cirak, Isogeometric shape optimisation of shell structures using multiresolution subdivision surfaces, Computer-Aided Design 95 (2018) 62 – 71.

[71] A. Wawrzinek, K. Polthier, Integration of generalized B-spline functions on CatmullClark surfaces at singularities, Computer-Aided Design 78 (2016) 60 – 70.

[72] B. Jüttler, A. Mantzaflaris, R. Perl, M. Rumpf, On numerical integration in isogeometric subdivision methods for PDEs on surfaces, Computer Methods in Applied Mechanics and Engineering 302 (2016) 131–146.

[73] G. Arden, Approximation properties of subdivision surfaces, Ph.D. thesis, University of Washington (2001).

[74] M. Scott, R. Simpson, J. Evans, S. Lipton, S. Bordas, T. J. R. Hughes, T. Sederberg, Isogeometric boundary element analysis using unstructured T-splines, Computer Methods in Applied Mechanics and Engineering 254 (2013) 197–221.

[75] M. Breitenberger, A. Apostolatos, B. Philipp, R. Wüchner, K.-U. Bletzinger, Analysis in computer aided design: Nonlinear isogeometric B-Rep analysis of shell structures, Computer Methods in Applied Mechanics and Engineering 284 (2015) 401–457.

[76] T. Belytschko, J. I. Lin, T. Chen-Shyh, Explicit algorithms for the nonlinear dynamics of shells, Computer methods in applied mechanics and engineering 42 (2) (1984) 225–251.

[77] X. Li, Some properties for analysis-suitable T-splines, Journal of Computational Mathematics 33 (2015) 428–442.

[78] C. Giannelli, B. Jüttler, H. Speleers, THB-splines: The truncated basis for hierarchical splines, Computer Aided Geometric Design 29 (7) (2012) 485–498.

[79] X. Wei, Y. Zhang, D. Toshniwal, H. Speleers, X. Li, C. Manni, J. A. Evans, T. J. R. Hughes, Blended B-spline construction on unstructured quadrilateral and hexahedral meshes with optimal convergence rates in isogeometric analysis, Computer Methods in Applied Mechanics and Engineering 341 (2018) 609–639.

[80] J. Zhang, X. Li, On the linear independence and partition of unity of arbitrary degree analysis-suitable T-splines, Communications in Mathematics and Statistics 3 (3) (2015) 353–364.

[81] L. Piegl, W. Tiller, The NURBS book, Springer Science & Business Media, 2012.

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

[83] M. A. Scott, M. J. Borden, C. V. Verhoosel, T. W. Sederberg, T. J. R. Hughes, Isogeometric finite element data structures based on Bézier extraction of T-splines, International Journal for Numerical Methods in Engineering 88 (2011) 126–156.

[84] T. Nguyen, J. Peters, Refinable C^1 spline elements for irregular quad layout, Computer aided geometric design 43 (2016) 123–130.

[85] U. Reif, A refineable space of smooth spline surfaces of arbitrary topological genus, Journal of Approximation Theory 90 (2) (1997) 174–199.

[86] X. Wei, Y. Zhang, T. J. R. Hughes, M. A. Scott, Truncated hierarchical Catmull-Clark subdivision with local refinement, Computer Methods in Applied Mechanics and Engineering 291 (2015) 1–20.

[87] X. Wei, Y. Zhang, T. J. R. Hughes, M. A. Scott, Extended truncated hierarchical Catmull-Clark subdivision, Computer Methods in Applied Mechanics and Engineering 299 (2016) 316–336.

[88] X. Wei, Y. Zhang, T. J. R. Hughes, Truncated hierarchical tricubic C^0 spline construction on unstructured hexahedral meshes for isogeometric analysis applications, Computers and Mathematics with Applications 74 (9) (2017) 2203–2220.

[89] W. Boehm, Inserting new knots into b-spline curves, Computer-Aided Design 12 (4) (1980) 199–201.

[90] T. J. Cashman, N. A. Dodgson, M. A. Sabin, A symmetric, non-uniform, refine and smooth subdivision algorithm for general degree b-splines, Computer Aided Geometric Design 26 (1) (2009) 94–104.

[91] T. Nguyen, K. Karčiauskas, J. Peters, A comparative study of several classical, discrete differential and isogeometric methods for solving Poissons equation on the disk, Axioms 3 (2) (2014) 280–299.

[92] M. Bischoff, K.-U. Bletzinger, W. A. Wall, E. Ramm, Models and finite elements for thin-walled structures, John Wiley & Sons, Ltd, 2004.

[93] S. Balay, M. F. Adams, J. Brown, P. Brune, K. Buschelman, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, K. Rupp, B. F. Smith, H. Zhang, PETSc Web page, `http://www.mcs.anl.gov/petsc` (2014).

[94] S. Balay, M. F. Adams, J. Brown, P. Brune, K. Buschelman, V. Eijkhout, W. D. Gropp, D. Kaushik, M. G. Knepley, L. C. McInnes, K. Rupp, B. F. Smith, H. Zhang, PETSc users manual, Tech. Rep. ANL-95/11 - Revision 3.4, Argonne National Laboratory (2013).

[95] P. R. Brune, M. G. Knepley, B. F. Smith, X. Tu, Composing scalable nonlinear algebraic solvers, SIAM Review 57 (4) (2015) 535–565.

[96] R. Arciniega, J. Reddy, Tensor-based finite element formulation for geometrically nonlinear analysis of shell structures, Computer Methods in Applied Mechanics and Engineering 196 (2007) 1048 – 1073.

[97] J. O. Hallquist, LS-DYNA theoretical manual. Livermore Software Technology Corporation.

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

[99] S. Bieber, B. Oesterle, E. Ramm, M. Bischoff, A variational method to avoid locking–independent of the discretization scheme, International Journal for Numerical Methods in Engineering 114 (8) (2018) 801–827.