Arbitrary-degree T-splines for isogeometric analysis of fully nonlinear Kirchhoff-Love shells

Hugo Casquero[a,*], Lei Liu[b], Yongjie Zhang[b], Alessandro Reali[c,d], Josef Kiendl[e], Hector Gomez[a]

[a] *Departamento de Métodos Matemáticos, Universidade da Coruña, Campus de A Coruña, 15071, A Coruña, Spain.*
[b] *Department of Mechanical Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, U.S.A.*
[c] *Department of Civil Engineering and Architecture, University of Pavia, via Ferrata 3, 27100, Pavia, Italy.*
[d] *Technische Universität München – Institute for Advanced Study, Lichtenbergstraße 2a, 85748, Garching, Germany.*
[e] *Institute of Applied Mechanics – TU Braunschweig, Bienroder Weg 87, 38106 Braunschweig, Germany.*

Abstract

This paper focuses on the employment of analysis-suitable T-spline surfaces of arbitrary degree for solving fully nonlinear thin shells. Our aim is to bring closer a seamless and flexible integration of design and analysis for shell applications. The local refinement capability of T-splines together with the Kirchhoff-Love shell discretization, which does not use rotational degrees of freedom, leads to a highly efficient and accurate formulation. Trimmed NURBS surfaces, which are ubiquitous in CAD programs, cannot be directly applied in analysis, however, T-splines can reparameterize these surfaces leading to analysis-suitable untrimmed T-spline representations. We consider various classical nonlinear benchmark problems where the cylindrical and spherical geometries are exactly represented and point loads are accurately captured through local h-refinement. Finally, we construct various trimmed NURBS surfaces with Rhino, an industrial and general-purpose CAD program, convert them to T-spline surfaces, and directly use them in analysis.

Keywords:
Isogeometric analysis, Analysis-suitable T-splines, Arbitrary-degree T-splines, Trimmed surfaces, Nonlinear Kirchhoff-Love shells, Incompressibility

1. Introduction

It is largely widespread in engineering to first generate a geometric model with a CAD program and then construct an analysis-suitable mesh to perform finite element analysis.

*Corresponding author.
Email address: hugo.casquero@udc.es (Hugo Casquero)

The process that goes from the CAD file to the analysis-suitable mesh can be highly time consuming. Moreover, in most cases, the geometry is slightly modified along the process, which may compromise the accuracy of the solution given by the finite element program. For complex designs, this process may need a lot of manual interactions and take much longer than the simulations that are going to be performed over the mesh afterwards [1]. Isogeometric analysis (IGA) was introduced with the goal of simplifying the above-mentioned process [2, 3]. IGA's main idea is to replace the conventional basis functions used in finite elements (e.g., Lagrange polynomials) with the blending functions employed in CAD, such as, for example, Non-Uniform Rational B-Splines (NURBS) and T-splines. Up to now, CAD programs model objects using surface parameterizations instead of volume representations. This is a significant obstacle for the integration of CAD and analysis, since most finite element computations need volume parameterizations. Therefore, to achieve full integration, a methodology that produces an analysis-suitable volume parameterization from a surface representation in a compatible and automatic way is needed. Although this is an open problem as far as we are aware, initial work on this direction has already been done [4, 5, 6, 7, 8]. Nevertheless, the study of shell mechanics is one of the few applications where surface parameterizations are used in analysis and this greatly simplifies the communication between CAD and analysis when the IGA paradigm is invoked.

One of the main differences between conventional finite-element basis functions and CAD blending functions is the global higher-order continuity of the latter. Within the analysis realm, the global smoothness of the basis functions has given rise to the proliferation of analysis applications based on NURBS or analysis-suitable T-splines (ASTS) including fields where CAD programs play no role. Some of the important benefits of the higher-order continuity of splines are: fourth-order and sixth-order partial differential equations can be solved in primal form [9, 10, 11, 12, 13, 14, 15, 16, 17], the strong form of partial differential equations can be collocated in a straightforward manner [18, 19, 20, 21, 22, 23, 24], enhanced robustness in solid mechanics and immersed fluid-structure interaction can be achieved compared to standard finite elements [25, 26], and others [1].

The analysis of Kirchhoff-Love shells, whose motion is governed by a fourth-order PDE posed on a surface, can be greatly simplified by using at least two of the aforementioned IGA's advantages, namely, the possibility of seamless integration between CAD and analysis as well as the global smoothness of the basis functions, which avoids the use of rotational degrees of freedom. These advantages have motivated the development of NURBS-based Kirchhoff-Love shell elements [27, 28, 29, 30, 31] and these formulations have been successfully applied to fluid-structure interaction [32, 33, 34, 35]. The Kirchhoff-Love shell theory, also referred to as "thin shell theory", assumes that transverse shear deformations are negligible. This theory is appropriate for shells that satisfy the geometrical constraint $R/t \geq \sim 20$, where R is the radius of curvature of the shell and t is its thickness [36]. Most shell structures of engineering interest satisfy this condition. However, Kirchhoff-Love shells are not currently widespread in commercial finite element programs due to the complexity of constructing globally $\mathcal{C}^1$-continuous spaces in traditional finite elements, which in addition, requires derivative degrees of freedom. Both NURBS and ASTS overcome this

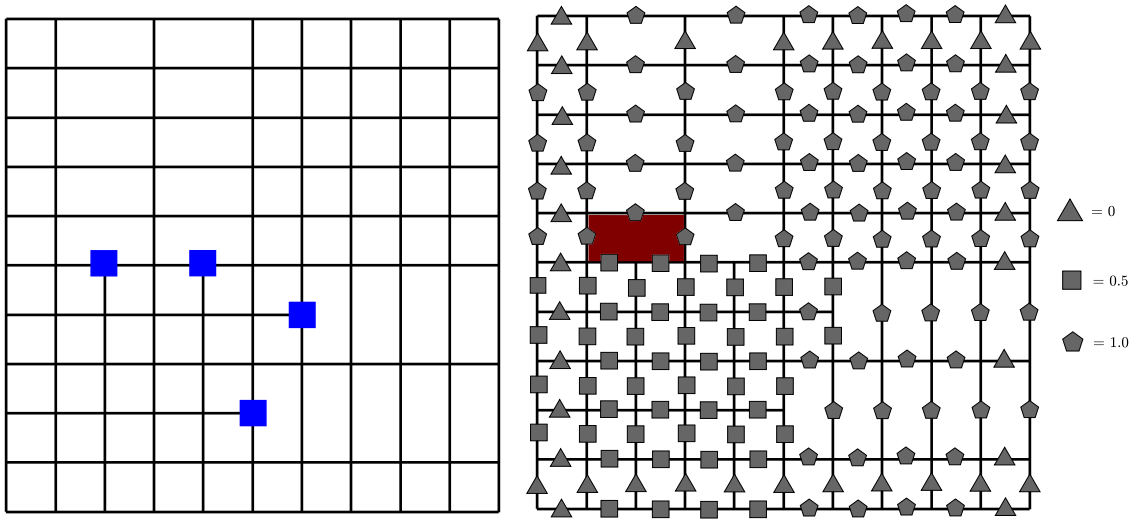
issue and we believe that Kirchhoff-Love shells are now in a position to compete against the Reissner-Mindlin shell theory, also known as "thick shell theory". The main reason why Reissner-Mindlin shells are currently the workhorse of commercial finite element programs is that their analysis requires only $\mathcal{C}^0$-conforming finite element spaces. However, they have significant downsides when compared with Kirchhoff-Love shells. For example, Reissner-Mindlin shell formulations require rotational degrees of freedom even at the continuum level, which after discretization increases both the total size of the system and the bandwidth of the tangent matrix by a factor of two. Reissner-Mindlin formulations also require methodologies that vanquish or alleviate sufficiently diverse "locking" phenomena.

In this work, we develop ASTS-based nonlinear Kirchhoff-Love shell formulations. The lack of local refinement capabilities of NURBS compromises the flexibility and efficiency of the computational methods developed with them. In contrast, ASTS support local h-refinement while maintaining all the important mathematical properties of NURBS [37, 38, 39]. Moreover, the local p-refinement of ASTS has recently been proposed [40]. Local refinement is needed, for example, in order to efficiently and accurately handle trimmed surfaces, which are largely widespread in CAD programs. Trimmed surfaces cannot be applied automatically to analysis. There are two main alternatives, namely, using an immersed method in analysis [41] or using an algorithm in order to reparameterize the surface [42]. We adopt the latter alternative in this article. In both cases, some type of local refinement is needed in order to capture the geometry of the trimming curve accurately. Another common situation where local h-refinement makes a difference is so as to capture accurately point loads.

The paper is outlined as follows: in Section 2 we give a brief review of ASTS of arbitrary degree and summarize the reparametrization algorithm for trimmed surfaces that we use. In Section 3 we describe the details of the fully nonlinear Kirchhoff-Love formulations that we utilize. In Section 4 we compare local against global refinement, study the performance of ASTS of different degrees, solve several nonlinear benchmark problems, and consider various examples that involve trimmed surfaces. We draw conclusions and outline future research directions in Section 5.

2. Analysis-suitable T-spline surfaces of arbitrary degree

A decade ago, T-splines were introduced in the CAD community as a generalization of NURBS [43]. T-splines were developed so as to overcome the main drawbacks of NURBS in design such as, for example, the high amount of superfluous control points needed to represent complex geometries [44] and the difficulty to create watertight surfaces in general cases [45]. In most instances, the number of degrees of freedom needed in order to capture the geometry of a certain surface is much lower than the amount of degrees of freedom needed to perform shell analysis over the surface. Therefore, when we receive a surface coming from a CAD program, the first step is typically to perform h-refinement. This refinement process can now be done locally thanks to ASTS, which increases the flexibility and efficiency of NURBS-based isogeometric analysis. The mathematical study of T-splines has been an active field in the previous years. As a consequence of this study, the subset



(a) T-mesh in index space (b) Knot interval configuration

Figure 1: (Color online) (a) T-mesh in index space with four T-junctions marked with blue squares. (b) Valid knot interval configuration for $p = 2$ and $p = 3$ associated with the T-mesh represented in Fig. 1(a). The pentagons, squares and triangles correspond to knot intervals of 1, 1/2, and 0, respectively. The knot intervals assigned to opposite sides of a given T-mesh face must sum to the same value as it is highlighted with the brown face.

of ASTS was defined, first for cubics [46, 47, 48] and then for arbitrary-degree splines [37, 38, 39]. ASTS are simply a subset (to be defined later) of T-splines that satisfies the following important mathematical properties:

(1) Linear independence of the blending functions
(2) Polynomial partition of unity of the blending functions
(3) Local h-refinement is available
(4) Optimal convergence rates in analysis under h-refinement
(5) Each blending function is pointwise non-negativity
(6) Affine covariance
(7) Convex hull property
(8) Dual compatibility [38]

Another important feature of ASTS is that they are forward and backward compatible with NURBS. This is a significant advantage of ASTS over other candidates that allow local refinement (e.g., subdivision surfaces [49, 50, 51]), but are not compatible with NURBS.

In this Section, we describe ASTS surfaces of arbitrary degree. We anticipate that ASTS of even and odd degree are constructed in a somewhat different manner. Here, we assume that the reader is familiarized with NURBS [52] and with the concept of Bézier extraction [53, 54]. All through the paper, we consider the same degree in the whole surface for sake of brevity and denote it by p.

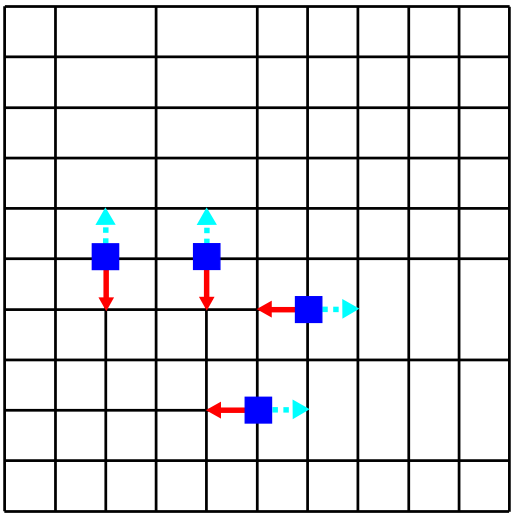


Figure 2: (Color online) Extended T-mesh associated to the T-mesh represented in Fig. 1(a) for $p = 2$. Face extensions are represented by light blue dashed lines and edge extensions are marked with solid red lines. Note that the red arrows hide the T-mesh edges located underneath them in the plot.

2.1. The T-mesh

For surfaces, the *T-mesh in index space* is a polygonal tiling of a two-dimensional region which defines the topology of an ASTS. For the sake of brevity, we will use the term T-mesh instead of T-mesh in index space when no confusion arises. The polygons, the corners of the polygons, and the line segments joining (exactly) two corners of the polygons in a T-mesh are called *faces*, *vertices*, and *edges*, respectively. Fig. 1(a) shows an example of T-mesh in index space.

A key difference between a T-mesh in index space and its NURBS counterpart is the existence of *T-junctions* [see the blue squares in Fig. 1(a)]. T-junctions play an analogous role to that of hanging nodes in conventional finite elements. The parametric information of an ASTS is given by the *knot interval configuration*, which is defined by assigning a non-negative real number to each T-mesh edge. We require the knot interval configuration to verify two conditions: (1) the knot intervals assigned to opposite sides of a given T-mesh face must sum to the same value [see brown face in Fig. 1(b)]; (2) the knot interval configuration has $\mathrm{int}(p/2)$ outer rings of zero-length knot intervals, where $\mathrm{int}(\cdot)$ represents the integer part of a real number. Assuming $p = 2$ or $p = 3$, a valid knot interval configuration for the T-mesh of Fig. 1(a) is shown in Fig. 1(b).

Let us now proceed to define the concepts of *T-junction extension* and *extended T-mesh*, which are helpful so as to set forth the topological condition that a T-mesh in index space needs to satisfy in order to be within the subset of ASTS. A T-junction extension is composed of a *face extension* and an *edge extension*. A face extension is a closed directed line segment obtained by marching from the T-junction in the direction of the missing edge until $\mathrm{int}((p+1)/2)$ orthogonal edges are encountered. An edge extension is a closed directed line segment obtained by marching from the T-junction in the opposite direction of

the face extension until int$(p/2)$ orthogonal edges are encountered. The extended T-mesh is obtained when T-junction extensions are plotted on top of the T-mesh. Fig. 2 shows the extended T-mesh for $p = 2$ corresponding to the T-mesh of Fig. 1(a).

A T-mesh in index space leads to an *ASTS space* when no vertical T-junction extension intersects with an horizontal face extension. Since face and edge extensions are closed line segments, vertical and horizontal T-junction extensions can intersect either on the interior of both extensions or at the endpoint of one or both extensions. As we will see later, an ASTS space is completely defined once we give a degree p, a valid T-mesh in index space, and a valid knot interval configuration.

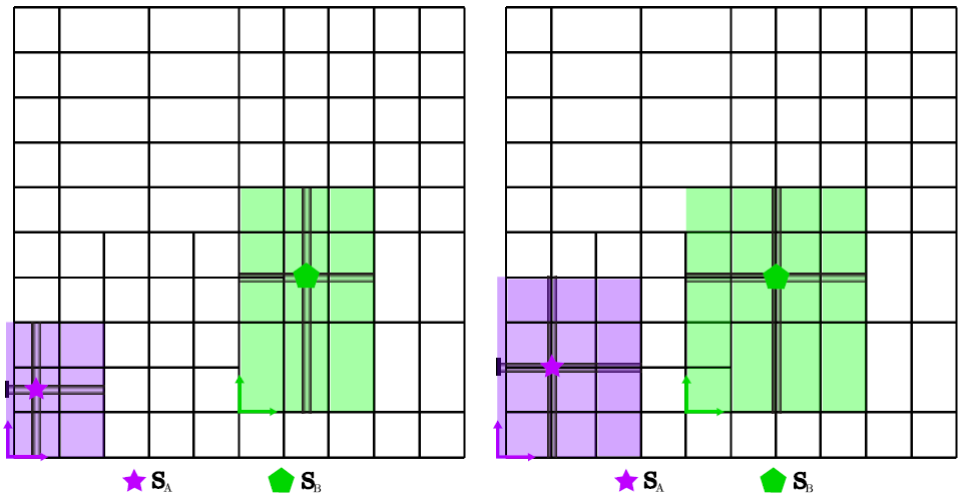
2.2. The ASTS basis

Let us start by defining an *anchor*, which is a point associated to each ASTS function. With the aforementioned definition of T-mesh in index space, an anchor in index space is located at every vertex if p is odd or at every face center if p is even. The anchor associated to the A-th global ASTS basis function will be denoted by $\boldsymbol{s}_A$. Fig. 3 shows two examples of anchors for both even- and odd-degree ASTS corresponding to the T-mesh of Fig. 1(a).

The ASTS basis functions are constructed from knot interval sequences inferred from their anchors. These knot interval sequences are called *local knot interval vectors* and we denote them by $\Delta\widehat{\Xi}_A^i$, where A is the global index of a basis function and i is one of the two directions of the index space ($i = 1$ represents the horizontal direction and $i = 2$ the vertical one). The vector $\Delta\widehat{\Xi}_A^i$ has always length $p + 1$ and can be written in component notation as $\Delta\widehat{\Xi}_A^i = \{\Delta\widehat{\xi}_{A,1}^i, \Delta\widehat{\xi}_{A,2}^i, ..., \Delta\widehat{\xi}_{A,p+1}^i\}$. In order to fill the two local knot interval vectors associated with a particular ASTS basis function, let us place horizontal and vertical line segments centered in the anchor that traverse exactly $p + 2$ orthogonal edges (see Fig. 3, where these segments are represented as thick semitransparent lines). Note that "centered in the anchor" means that the segment crosses the same number of orthogonal edges on the left- and right-hand sides of the anchor. These two line segments define the rectangular support of the ASTS basis function associated to that anchor (see the shaded areas in Fig. 3). Then, the horizontal local knot interval vector is filled by the knot intervals associated with the $p + 1$ edges spanned by the horizontal line segment going from the left to the right. Analogously, the vertical local knot interval vector is filled by the knot intervals associated with the $p+1$ edges spanned by the vertical line segment going from the bottom to the top. When an anchor is sufficiently close to the T-mesh boundary so that we cannot define a large enough line segment in a certain direction, zero-length edges should be added accordingly (see the anchors $\boldsymbol{s}_A$ in Fig. 3, where zero-length intervals were added in the horizontal direction).

For each function, we will define a *local basis coordinate system* with origin in the bottom-left corner of its support (see the arrows in Fig. 3). The local coordinate system of function A will be denoted by $\widehat{\boldsymbol{\xi}}_A = (\widehat{\xi}_A^1, \widehat{\xi}_A^2)$. In this reference system, we use the local knot interval vectors to define a *local basis function domain*, $\widehat{\Omega}_A \subset \mathbb{R}^2$, as

$$\widehat{\Omega}_A = \widehat{\Omega}_A^1 \times \widehat{\Omega}_A^2, \tag{1}$$



(a) $p = 2$ (b) $p = 3$

Figure 3: (Color online) Anchors, local knot interval vectors, and basis coordinate systems associated to the T-mesh and knot interval configuration represented in Fig. 1 for $p = 2$ and $p = 3$. (a) The local knot interval vectors associated with the A-th and B-th global ASTS basis functions for $p = 2$ are $\Delta\widehat{\Xi}^1_A = \{0, 0, 1/2\}$, $\Delta\widehat{\Xi}^2_A = \{0, 1/2, 1/2\}$, $\Delta\widehat{\Xi}^1_B = \{1, 1, 1\}$, $\Delta\widehat{\Xi}^2_B = \{1, 1, 1\}$. (b) The local knot interval vectors associated with the A-th and B-th global ASTS basis functions for $p = 3$ are $\Delta\widehat{\Xi}^1_A = \{0, 0, 1/2, 1/2\}$, $\Delta\widehat{\Xi}^2_A = \{0, 1/2, 1/2, 1/2\}$, $\Delta\widehat{\Xi}^1_B = \{1/2, 1, 1, 1\}$, $\Delta\widehat{\Xi}^2_B = \{1, 1/2, 1/2, 1\}$.

where $\widehat{\Omega}^i_A = \left[0, \sum_{j=1}^{p+1} \Delta\widehat{\xi}^i_{A,j}\right] \subset \mathbb{R}$. Then, *local knot vectors*, $\widehat{\boldsymbol{\Xi}}_A = \left\{\widehat{\boldsymbol{\Xi}}^i_A\right\}_{i=1}^2$, where $\widehat{\boldsymbol{\Xi}}^i_A = \{\widehat{\xi}^i_{A,1}, \widehat{\xi}^i_{A,2}, ..., \widehat{\xi}^i_{A,p+2}\}$, can be derived from $\Delta\widehat{\boldsymbol{\Xi}}_A$ by selecting an arbitrary origin $\boldsymbol{O} = \{O^1, O^2\} \in \mathbb{R}^2$ and setting $\widehat{\xi}^i_{A,j} = O^i + \sum_{k=1}^{j-1} \Delta\widehat{\xi}^i_{A,k}$. Now, we can define the basis function associated to the anchor $\boldsymbol{s}_A$, namely, $N_A : \widehat{\Omega}_A \mapsto \mathbb{R}^+ \cup \{0\}$. This is accomplished by forming the tensor product of the proper univariate B-spline basis functions as

$$N_A(\widehat{\boldsymbol{\xi}}_A|\widehat{\boldsymbol{\Xi}}_A) = \prod_{i=1}^{2} N^{i,p}_A(\widehat{\boldsymbol{\xi}}^i_A|\widehat{\boldsymbol{\Xi}}^i_A), \tag{2}$$

where $N^{i,p}_A : \widehat{\Omega}^i_A \mapsto \mathbb{R}^+ \cup \{0\}$ is an univariate B-spline basis function of degree p. The function $N^{i,p}_A$ may be defined using a recurrence relation, starting with the piecewise constant $(p = 0)$ basis function

$$N^{i,0}_A(\widehat{\xi}^i_A|\widehat{\xi}^i_{A,1}, \widehat{\xi}^i_{A,2}) = \begin{cases} 1 & \text{if } \widehat{\xi}^i_{A,1} \leq \widehat{\xi}^i_A < \widehat{\xi}^i_{A,2} \\ 0 & \text{otherwise,} \end{cases} \tag{3}$$

For $p > 0$, the basis function is defined using the Cox-de Boor recursion formula:

$$N_A^{i,p}(\widehat{\xi}_A^i|\widehat{\xi}_{A,1}^i,\widehat{\xi}_{A,2}^i,...,\widehat{\xi}_{A,p+2}^i) = \frac{\widehat{\xi}_A^i - \widehat{\xi}_{A,1}^i}{\widehat{\xi}_{A,p+1}^i - \widehat{\xi}_{A,1}^i} N_A^{i,p-1}(\widehat{\xi}_A^i|\widehat{\xi}_{A,1}^i,...,\widehat{\xi}_{A,p+1}^i) + \frac{\widehat{\xi}_{A,p+2}^i - \widehat{\xi}_A^i}{\widehat{\xi}_{A,p+2}^i - \widehat{\xi}_{A,2}^i} N_A^{i,p-1}(\widehat{\xi}_A^i|\widehat{\xi}_{A,2}^i,...,\widehat{\xi}_{A,p+2}^i). \quad (4)$$

Note that the ASTS basis function has been defined over its own local basis function domain.

2.3. The ASTS geometric map

The *elemental T-mesh* is obtained from the T-mesh in index space as follows: First, we plot the face extensions over the T-mesh. Second, we remove the polygons for which the knot interval sum is zero on at least one side. Fig. 4(a) shows, as shaded areas delimited by black lines, the polygons that compose the elemental T-mesh associated with the T-mesh and the knot interval configuration of Fig. 1(a) for $p = 2$. The polygons of the elemental T-mesh are important objects for analysis since they delimit areas in which all ASTS basis functions are $\mathcal{C}^\infty$. Therefore, these polygons will be called *elements*. When the elements of the elemental T-mesh are pushed forward to physical space, they will be called *Bézier elements* and the collection of Bézier elements will be referred to as the Bézier mesh [see Fig. 4(b)]. The Bézier elements are suitable regions to perform numerical integration in an isogeometric analysis program.

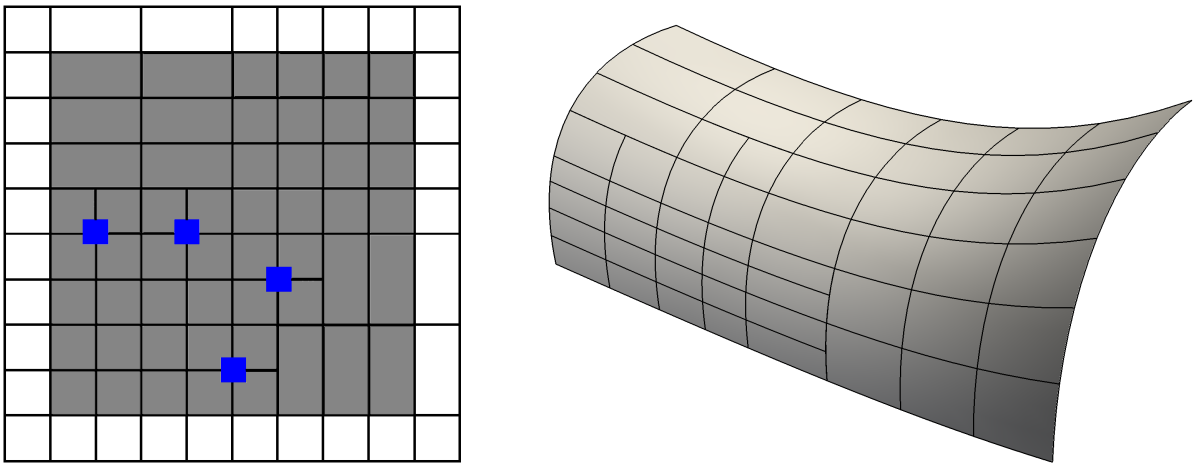
For a certain element e, Bézier extraction enables to represent the n^e ASTS basis functions that have support on that element as a linear combination of the canonical tensor product Bernstein polynomial basis defined on a fixed parent element denoted by $\square = [-1, +1]^2$. This can be expressed mathematically as

$$\boldsymbol{N}^e(\boldsymbol{\xi}) = \boldsymbol{C}^e \boldsymbol{B}(\boldsymbol{\xi}) \quad \boldsymbol{\xi} \in \square\,, \quad (5)$$

where $\boldsymbol{N}^e = \{N_a^e\}_{a=1}^{n^e}$ collects the ASTS basis functions with support on the element e, the index a is a local-to-element ASTS basis function counter so that the global basis function number is $A = \mathrm{IEN}(a, e)$. The array IEN represents the usual map from local to global unknowns [55]. $\boldsymbol{B} = \{B_i^p\}_{i=1}^{(p+1)^2}$ collects the two-dimensional Bernstein polynomials of degree p in the domain $\square$ (see [56] for a precise description) and $\boldsymbol{C}^e$ is a linear operator called the element Bézier extraction operator. The computation of the rectangular matrix $\boldsymbol{C}^e$ is based on knot insertion and [54] describes in detail how to efficiently compute this operator for ASTS.

In certain situations, it is useful to employ rational basis functions, typically to reproduce exactly a particular geometry. Given a set of global weights $\{w_A\}_{A=1}^n$ and their local-to-element counterparts, namely, $\{w_a^e\}$, we can rationalize the ASTS basis functions in $\boldsymbol{N}^e$ as

$$R_a^e(\boldsymbol{\xi}) = \frac{w_a^e N_a^e(\boldsymbol{\xi})}{\sum_{b=1}^{n^e} w_b^e N_b^e(\boldsymbol{\xi})}\,, \quad (6)$$



(a) Elemental T-mesh (b) Bézier mesh

Figure 4: (Color online) (a) Elemental T-mesh for $p = 2$ associated with the T-mesh and the knot interval configuration of Fig. 1. (b) Bézier mesh for $p = 2$ associated with the T-mesh and the knot interval configuration of Fig. 1.

where R_a^e is the a-th rational ASTS basis function over the element e. This concludes the process of constructing the ASTS basis functions in parameter space.

In order to perform computations on non-trivial surfaces, we need to map the parent element to physical space using a geometric map $\boldsymbol{x}^e : \square \mapsto \Omega^e$, which goes from the parent element to the Bézier element Ω^e. A set of geometry control points $\{\boldsymbol{P}_A\}_{A=1}^n$ is needed in order to define this map. A given geometry control point $\boldsymbol{P}_A$ will have local indices associated such that $\boldsymbol{P}_A = \boldsymbol{P}_a^e$, as it was the case with the basis functions and the weights. Thus, the geometric map local to element e can be defined as

$$\boldsymbol{x}^e(\boldsymbol{\xi}) = \sum_{a=1}^{n^e} \boldsymbol{P}_a^e R_a^e(\boldsymbol{\xi}) \quad \boldsymbol{\xi} \in \square \,. \tag{7}$$

Using the ASTS geometric map given by Eq. (7), we can get the ASTS basis functions in physical space. Eq. (7) is also used to map the elemental T-mesh to physical space, that is, to obtain the Bézier mesh.

Invoking the isoparametric concept, we will represent our displacement field as follows

$$\boldsymbol{u}^e(\boldsymbol{\xi}) = \sum_{a=1}^{n^e} \boldsymbol{U}_a^e R_a^e(\boldsymbol{\xi}) \quad \boldsymbol{\xi} \in \square \,, \tag{8}$$

where $\boldsymbol{u}^e$ is the displacement vector on the element e and $\boldsymbol{U}_a^e$ are the control variables of the displacement field that contribute on the element e.

In terms of implementation, Bézier extraction is highly appealing since it enables to use ASTS in a finite element program by simply modifying the shape function subroutine.

2.4. Local h-refinement

Local h-refinement of ASTS boils down to adding vertices and edges on the T-mesh in index space and modifying the knot intervals of the edges which are split in such a way that we maintain both a valid T-mesh topology and a valid knot interval configuration. As long as the initial and the final T-meshes in index space and knot interval configurations are within the subset of ASTS, the initial and the final ASTS are nested [48, 39]. As in the case of the global h-refinement of NURBS, there are algorithms in order to compute the new physical positions of the control points after local h-refinement so that the surface is not changed either geometrically or parametrically. These algorithms are based on knot insertion and are described in detail in [47].

2.5. Converting a trimmed NURBS surface into an untrimmed T-spline surface

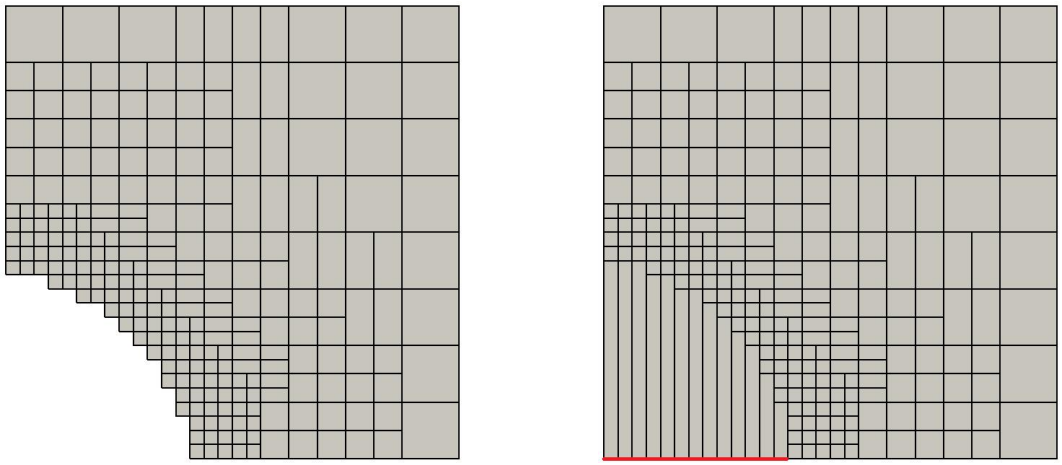
In order to reconstruct a trimmed NURBS surface by means of ASTS, we employ the method introduced in [42] for the necessary reparameterization. The reparameterization algorithm has three main steps. The first step consists in performing local h-refinement on the initial NURBS surface close to the trimming curve, which is straightforward due to the compatibility between NURBS and ASTS. The second step consists in deleting the elements that are completely beyond the trimming curve. Fig. 5(a) shows an example of resulting T-mesh in parametric space after these two steps for a squared NURBS surface, which is trimmed by a circular arc (not shown) in the bottom-left corner. After these two steps, the T-mesh in parametric space is not a square and this needs to be fixed. Therefore, the zigzag of Fig. 5(a) is removed by modifying the values of the knot intervals properly as it is shown in Fig. 5(b). The third step consists in using the modified polygons in order to capture the geometry of the trimming curve accurately. For additional details, the reader is remitted to [42], where the final untrimmed ASTS meshes were also validated solving various classical benchmark problems of 2D linear elasticity. The final Bézier mesh associated with Fig. 5 will be later on used to perform computations and may be found in Fig. 14(b).

3. Shell formulation

3.1. Kinematics

In this section, we will use basic concepts of differential geometry. The reader not familiarized with these concepts is referred to, e.g., [57] for an introduction. The shell geometry can be represented by its middle surface and the thickness t. The geometry of the middle surface is expressed with respect to the coordinates ξ^α. Here, α takes the values $\{1, 2\}$ and this will be the case for any Greek index utilized in this section. Latin indices, however, take values $\{1, 2, 3\}$. If $\boldsymbol{r}$ is the position vector of a point on the middle surface, we can compute the tangent and normal base vectors as follows:

$$\boldsymbol{a}_\alpha = \boldsymbol{r}_{,\alpha}, \qquad \boldsymbol{a}_3 = \frac{\boldsymbol{a}_1 \times \boldsymbol{a}_2}{||\boldsymbol{a}_1 \times \boldsymbol{a}_2||}, \tag{9}$$



(a) (b)

Figure 5: (Color online) Reparameterization of a trimmed NURBS surface with ASTS. (a) T-mesh in parametric space after abandoning elements beyond a circular trimming curve (not shown). (b) T-mesh in parametric space after knot interval modification, where the control points associated to the vertices falling on the red parametric line are used to reconstruct the trimming curve on the ASTS surface.

where $(\cdot)_{,\alpha} = \partial(\cdot)/\partial\xi^\alpha$ denotes the partial derivative with respect to a surface coordinate ξ^α, $||\cdot||$ denotes the length of a vector and $\times$ the classical cross product. In what follows, we will also use the Einstein summation convention of repeated indices. With the base vectors (9), the first and second fundamental forms can be computed, representing the metric and curvature properties of the surface:

$$a_{\alpha\beta} = \boldsymbol{a}_\alpha \cdot \boldsymbol{a}_\beta, \tag{10}$$

$$b_{\alpha\beta} = \boldsymbol{a}_{\alpha,\beta} \cdot \boldsymbol{a}_3. \tag{11}$$

Due to the Kirchhoff hypothesis of normal cross sections, any point $\boldsymbol{x}$ in the shell off the midsurface can be described by the position vector of the midsurface $\boldsymbol{r}$ and the normal vector $\boldsymbol{a}_3$:

$$\boldsymbol{x} = \boldsymbol{r} + \xi^3 \, \boldsymbol{a}_3 \ , \tag{12}$$

where $\xi^3 \in [-t/2, t/2]$ is the thickness coordinate. From (12), we can compute covariant base vectors and metric coefficients off the midsurface as $\boldsymbol{g}_i = \boldsymbol{x}_{,i}$ and $g_{ij} = \boldsymbol{g}_i \cdot \boldsymbol{g}_j$, respectively. We obtain $g_{\alpha 3} = g_{3\alpha} = 0$ according to the Kirchhoff constraint, and $g_{33} = a_{33} \equiv 1$. Thus, only in-plane components $g_{\alpha\beta}$ need to be computed. Neglecting terms that appear quadratic in ξ^3 and using (10) and (11) we obtain

$$g_{\alpha\beta} = a_{\alpha\beta} - 2\,\xi^3 b_{\alpha\beta} \ . \tag{13}$$

Contravariant base vectors $\boldsymbol{g}^i$ are defined by $\boldsymbol{g}^i \cdot \boldsymbol{g}_j = \delta^i_j$, where δ^i_j is the Kronecker delta. The contravariant metric coefficients can be obtained using the condition $g^{ik} g_{kj} = \delta^i_j$. All

of the above equations are valid for both the deformed and undeformed configurations. In the following, quantities that are associated to the undeformed configuration will be indicated by a symbol $\mathring{(\cdot)}$. For example, $\mathring{\boldsymbol{x}}$ is a point in the undeformed shell. Taking this into account, we can define the displacement vector as $\boldsymbol{u} = \boldsymbol{r} - \mathring{\boldsymbol{r}}$.

Let us now introduce the in-plane Jacobian determinant of the mapping from the undeformed to the deformed configuration

$$J_o = \sqrt{|g_{\alpha\beta}|/|\mathring{g}_{\alpha\beta}|} \ , \tag{14}$$

where $|g_{\alpha\beta}|$ denotes the determinant of the matrix formed by the metric coefficients $g_{\alpha\beta}$. In the following, we introduce some important tensors. Unless we indicate otherwise, we will write all tensors in the curvilinear basis of the undeformed geometry. Accordingly, the left Cauchy–Green deformation tensor $\boldsymbol{C} = C_{ij}\, \mathring{\boldsymbol{g}}^i \otimes \mathring{\boldsymbol{g}}^j$ can be represented as:

$$C_{ij} = \begin{pmatrix} g_{11} & g_{12} & 0 \\ g_{21} & g_{22} & 0 \\ 0 & 0 & C_{33} \end{pmatrix} \ , \tag{15}$$

where the in-plane components $C_{\alpha\beta}$ are obtained as the covariant metric coefficients in the deformed configuration $g_{\alpha\beta}$ and $C_{\alpha 3} = C_{3\alpha} = 0$ according to zero transverse shear deformation, while the thickness deformation C_{33}, which cannot be neglected in the case of large strains, needs to be determined through the constitutive equations as will be shown below. Before doing that, let us note that the volumetric Jacobian determinant may be expressed as $J = J_o\sqrt{C_{33}}$.

Finally, the Green-Lagrange strain tensor $\boldsymbol{E} = E_{ij}\, \mathring{\boldsymbol{g}}^i \otimes \mathring{\boldsymbol{g}}^j$ is obtained by

$$E_{ij} = \frac{1}{2}(C_{ij} - \mathring{g}_{ij}) \ . \tag{16}$$

3.2. Compressible materials

Assuming a hyperelastic material, the second Piola–Kirchhoff stress tensor $\boldsymbol{S} = S^{ij}\, \mathring{\boldsymbol{g}}_i \otimes \mathring{\boldsymbol{g}}_j$ is obtained from a strain-energy density function ψ as

$$S^{ij} = 2\frac{\partial \psi}{\partial C_{ij}} \ , \tag{17}$$

and the tangent material tensor $\mathbb{C} = \mathbb{C}^{ijkl} \mathring{\boldsymbol{g}}_i \otimes \mathring{\boldsymbol{g}}_j \otimes \mathring{\boldsymbol{g}}_k \otimes \mathring{\boldsymbol{g}}_l$ as

$$\mathbb{C}^{ijkl} = 4\frac{\partial^2 \psi}{\partial C_{ij} \partial C_{kl}} \ . \tag{18}$$

As mentioned above, the thickness stretch C_{33} cannot be determined directly from the kinematic equations. However, we can use the plane stress condition, $S^{33} = 0$, which must be satisfied, in order to compute C_{33}. Starting with an initial value $C_{33} = 1$, we simply

determine S^{ij} and $\mathbb{C}^{ijkl}$ according to (17) and (18), and then compute an incremental correction of C_{33} as

$$\Delta C_{33}^{(I)} = -2\frac{S_{(I)}^{33}}{\mathbb{C}_{(I)}^{3333}} \ , \tag{19}$$

$$C_{33}^{(I+1)} = C_{33}^{(I)} + \Delta C_{33}^{(I)} \ , \tag{20}$$

where I indicates the iteration step. Note that Eqns. (19)–(20) are simply the result of applying Newton's method to the equation $S^{33} = 0$. With the updated $\boldsymbol{C}$, we compute the updates of $\boldsymbol{S}(\boldsymbol{C})$ and $\mathbb{C}(\boldsymbol{C})$, and the procedure is repeated until the plane stress condition is satisfied within a defined tolerance. Once the plane stress condition is enforced, it can be used to determine the transverse normal strain E_{33} as:

$$E_{33} = -\frac{\mathbb{C}^{33\alpha\beta}}{\mathbb{C}^{3333}} E_{\alpha\beta} \ , \tag{21}$$

which is then eliminated from the equations by a static condensation of the material tensor:

$$\hat{\mathbb{C}}^{\alpha\beta\gamma\delta} = \mathbb{C}^{\alpha\beta\gamma\delta} - \frac{\mathbb{C}^{\alpha\beta 33}\mathbb{C}^{33\gamma\delta}}{\mathbb{C}^{3333}} \ , \tag{22}$$

with $\hat{\mathbb{C}}^{\alpha\beta\gamma\delta}$ denoting the statically condensed material tensor. The procedure shown above can be applied to arbitrary 3D compressible materials being described by a strain energy function ψ.

3.3. Incompressible materials

In the case of incompressible materials, we get the following augmented strain energy

$$\psi = \psi_{el} - p(J-1) \ , \tag{23}$$

where ψ_{el} is the elastic strain energy function, $(J-1)$ is the term that is constrained to be zero, enforcing incompressibility, and p is a Lagrange multiplier, which can be identified as the hydrostatic pressure [58]. As shown in detail in [31], we can use the plane stress condition, $S^{33} = 0$, in order to analytically determine and eliminate the Lagrange multiplier p. Eventually, we obtain the following equations for the stress tensor and the statically condensed material tensor:

$$S^{\alpha\beta} = 2\frac{\partial \psi_{el}}{\partial C_{\alpha\beta}} - 2\frac{\partial \psi_{el}}{\partial C_{33}} J_o^{-2} g^{\alpha\beta} \ , \tag{24}$$

$$\begin{aligned}\hat{\mathbb{C}}^{\alpha\beta\gamma\delta} = {} & 4\frac{\partial^2 \psi_{el}}{\partial C_{\alpha\beta}\partial C_{\gamma\delta}} + 4\frac{\partial^2 \psi_{el}}{\partial C_{33}^2} J_o^{-4} g^{\alpha\beta} g^{\gamma\delta} - 4\frac{\partial^2 \psi_{el}}{\partial C_{33}\partial C_{\alpha\beta}} J_o^{-2} g^{\gamma\delta} - 4\frac{\partial^2 \psi_{el}}{\partial C_{33}\partial C_{\gamma\delta}} J_o^{-2} g^{\alpha\beta} \\ & + 2\frac{\partial \psi_{el}}{\partial C_{33}} J_o^{-2} (2g^{\alpha\beta} g^{\gamma\delta} + g^{\alpha\gamma} g^{\beta\delta} + g^{\alpha\delta} g^{\beta\gamma}) \ .\end{aligned} \tag{25}$$

With Eqs. (24) and (25) the incompressibility and plane stress constraints are satisfied, and no iterative procedure as in (20) is necessary. We note that the approach shown in (19)-(20) applies to compressible materials, while (24)-(25) holds for the case of incompressibility, and we highlight that both approaches allow for a direct use of arbitrary (hyperelastic) 3D constitutive models for shell analysis.

3.4. Variational formulation

As usual for shells, the Green-Lagrange strains $E_{\alpha\beta}$ are obtained by a combination of membrane stains $\varepsilon_{\alpha\beta}$ and curvature changes $\kappa_{\alpha\beta}$:

$$E_{\alpha\beta} = \varepsilon_{\alpha\beta} + \xi^3 \kappa_{\alpha\beta} \ , \tag{26}$$

$$\varepsilon_{\alpha\beta} = \frac{1}{2}(a_{\alpha\beta} - \mathring{a}_{\alpha\beta}) \ , \tag{27}$$

$$\kappa_{\alpha\beta} = \mathring{b}_{\alpha\beta} - b_{\alpha\beta} \ , \tag{28}$$

while stresses are integrated through the thickness and represented by normal forces $n^{\alpha\beta}$ and bending moments $m^{\alpha\beta}$:

$$n^{\alpha\beta} = \int_{-t/2}^{t/2} S^{\alpha\beta} \mathrm{d}\xi^3 \ , \tag{29}$$

$$m^{\alpha\beta} = \int_{-t/2}^{t/2} S^{\alpha\beta} \xi^3 \mathrm{d}\xi^3 \ , \tag{30}$$

with their linearizations obtained by:

$$\mathrm{d}n^{\alpha\beta} = \left(\int_{-t/2}^{t/2} \hat{\mathbb{C}}^{\alpha\beta\gamma\delta} \mathrm{d}\xi^3 \right) \mathrm{d}\varepsilon_{\gamma\delta} + \left(\int_{-t/2}^{t/2} \hat{\mathbb{C}}^{\alpha\beta\gamma\delta} \, \xi^3 \mathrm{d}\xi^3 \right) \mathrm{d}\kappa_{\gamma\delta} \ , \tag{31}$$

$$\mathrm{d}m^{\alpha\beta} = \left(\int_{-t/2}^{t/2} \hat{\mathbb{C}}^{\alpha\beta\gamma\delta} \xi^3 \mathrm{d}\xi^3 \right) \mathrm{d}\varepsilon_{\gamma\delta} + \left(\int_{-t/2}^{t/2} \hat{\mathbb{C}}^{\alpha\beta\gamma\delta} (\xi^3)^2 \, \mathrm{d}\xi^3 \right) \mathrm{d}\kappa_{\gamma\delta} \ . \tag{32}$$

In statics, the variational formulation is derived from the equilibrium of internal and external virtual work, $\delta W = \delta W^{int} - \delta W^{ext} = 0$ with the internal and external virtual work defined as:

$$\delta W^{int} = \int_A (\boldsymbol{n} : \delta\boldsymbol{\varepsilon} + \boldsymbol{m} : \delta\boldsymbol{\kappa}) \, \mathrm{d}A \ , \tag{33}$$

$$\delta W^{ext} = \int_A \boldsymbol{f} \cdot \delta\boldsymbol{u} \, \mathrm{d}A \ , \tag{34}$$

where : represents the classical double contraction, $\boldsymbol{f}$ denotes the external load, $\delta\boldsymbol{u}$ is a virtual displacement, $\delta\boldsymbol{\varepsilon}$ and $\delta\boldsymbol{\kappa}$ are the corresponding virtual membrane strain and curvature change, respectively, A denotes the midsurface and $\mathrm{d}A = \sqrt{|\mathring{a}_{\alpha\beta}|}\mathrm{d}\xi^1\mathrm{d}\xi^2$ the differential area, both in the reference configuration.

We discretize and linearize Eqs. (33)-(34) using an ASTS discretization for both the shell's geometry and the displacement field. The displacement field is assumed to be expressed as a combination of control variables, each with three coordinates corresponding to the spatial dimensions. All the components of the control variables are collected on a bigger vector $\boldsymbol{U}$ by changing first the index of the spatial dimension and then, the basis

function number. Doing so, we obtain the following expressions for the contributions of element e to the vector of internal nodal forces $\boldsymbol{F}^{int}$ and the stiffness matrix $\boldsymbol{K}$

$$F_r^{int} = \int_A \left(\boldsymbol{n} : \frac{\partial \boldsymbol{\varepsilon}}{\partial U_r} + \boldsymbol{m} : \frac{\partial \boldsymbol{\kappa}}{\partial U_r} \right) \mathrm{d}A \ , \tag{35}$$

$$K_{rs} = \int_A \left(\frac{\partial \boldsymbol{n}}{\partial U_s} : \frac{\partial \boldsymbol{\varepsilon}}{\partial U_r} + \boldsymbol{n} : \frac{\partial^2 \boldsymbol{\varepsilon}}{\partial U_r \partial U_s} + \frac{\partial \boldsymbol{m}}{\partial U_s} : \frac{\partial \boldsymbol{\kappa}}{\partial U_r} + \boldsymbol{m} : \frac{\partial^2 \boldsymbol{\kappa}}{\partial U_r \partial U_s} \right) \mathrm{d}A \ , \tag{36}$$

where r and s are global degree-of-freedom numbers given by $r = 3(\mathrm{IEN}(a,e) - 1) + i$ and $s = 3(\mathrm{IEN}(b,e) - 1) + j$ with $i, j = 1, 2, 3$ referring to the global x, y, and z components. The detailed expression of each term appearing in the above two equations can be found in [31]. With (35)-(36), we solve for the linearized equation system

$$\boldsymbol{K}\, \Delta \boldsymbol{U} = -\boldsymbol{R} \ . \tag{37}$$

with $\boldsymbol{R} = \boldsymbol{F}^{int} - \boldsymbol{F}^{ext}$ being the residual, where the vector of external loads $\boldsymbol{F}^{ext}$ is computed in a standard way by integrating over the product of external loads and ASTS basis functions. The vector $\Delta \boldsymbol{U}$ is used for the nonlinear updates of the displacement field until convergence is achieved.

4. Numerical examples

In this Section, we present several numerical examples that illustrate the main advantages that ASTS of arbitrary degree bring to shell applications. First of all, we show the enhanced efficiency and flexibility that the local h-refinement capabilities of ASTS give to the formulation in comparison with the global h-refinement of NURBS. Second of all, we compare the accuracy per degree-of-freedom for different orders p. We observe that the accuracy increases with the order and higher order ASTS elements are stable under large deformations, in contrast to the instabilities that are known to appear with higher order Lagrange elements in these situations. Then, we solve two widespread nonlinear benchmark problems, which involve cylindrical and spherical geometries, represented exactly by ASTS. Finally, we consider two examples that illustrate how trimmed NURBS surfaces can be handled in shell analysis.

The analysis code used to perform all these simulations has been developed on top of the scientific library PETSc [59, 60]. Regarding the nonlinear and linear solvers used in the following examples, we have employed the Newton-Raphson algorithm with line search capabilities and direct solvers based on LU factorization, respectively.

4.1. Pinched cylindrical shell with rigid diaphragms

This is a common benchmark problem defined by a pinched cylinder under two inward opposite forces. The lateral boundaries of the cylinder have rigid diaphragms. The diaphragms preclude movements in their own plane and produce stress-free solutions in the orthogonal direction. This problem has been solved previously in [61, 62, 63]. Due to the three planes of symmetry, only one octant of the cylinder is modeled for analysis purposes

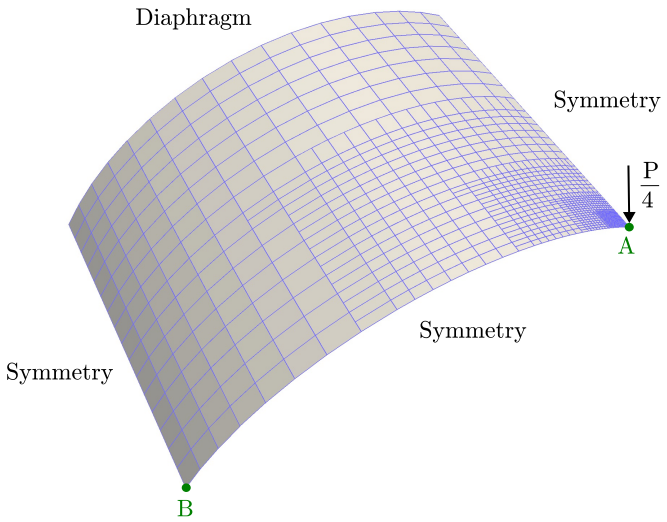


Figure 6: (Color online) Five-level quadratic Bézier mesh for the pinched cylinder along with the boundary conditions and the point load applied.

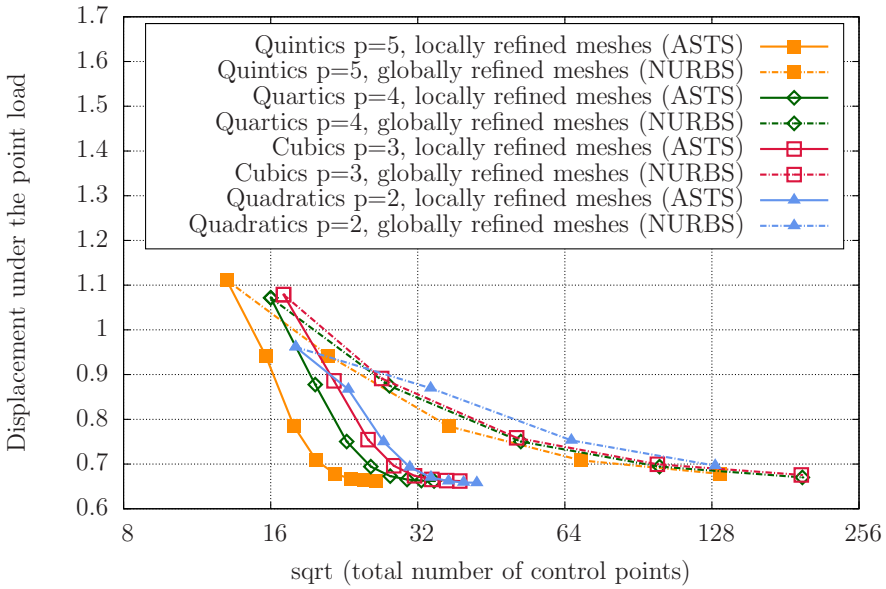


Figure 7: (Color online) Comparison of locally h-refined ASTS meshes against globally h-refined NURBS meshes for $p = 2$, 3, 4, and 5 for a point load of $P = 100$. For all orders, the local refinement capability of ASTS allows to get the same level of accuracy with a lot fewer degrees of freedom, which significantly improves the efficiency of our shell formulation based on ASTS.

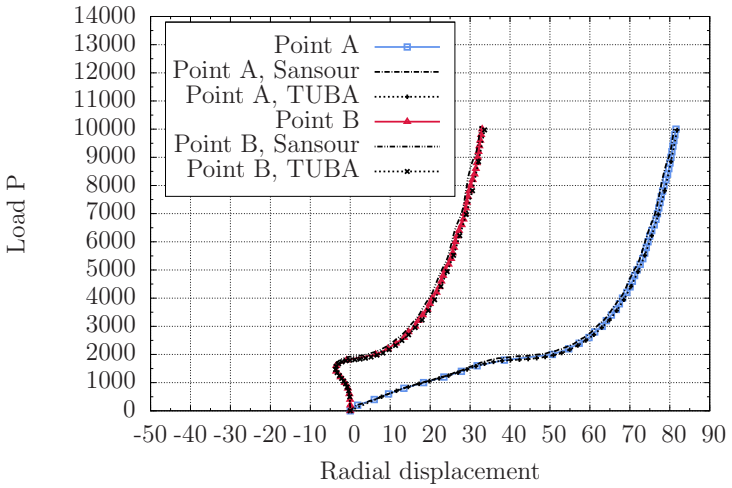


Figure 8: (Color online) Load-deflection curves at points A and B of the pinched cylindrical shell with a point load of value $P = 10,000$. We note that 11,767, 8,136, and 6,543 are the total number of degrees of freedom used in [61], [62], and our simulation, respectively.

(see Fig. 6). The geometry of the problem is defined by the parameters $R = 100$, $L = 200$ and $t = 1$, where R and L are, respectively, the radius and the length of the cylinder. Regarding the hyperelastic material, we considered the following compressible Neo-Hookean strain energy function [64]

$$\psi = \frac{1}{2}\mu\left(J^{-2/3}\mathrm{tr}(\boldsymbol{C}) - 3\right) + \frac{1}{4}K\,\left(J^2 - 1 - 2\ln J\right)\,, \tag{38}$$

where $\mathrm{tr}(\cdot)$ denotes the trace of a tensor, while μ and K are the shear and bulk moduli, respectively. We obtain μ and K from the Young's modulus $E = 3.0 \times 10^4$ and the Poisson ratio $\nu = 0.3$. Applying Eqs. (17) and (18) to the above energy function, we obtain the following 3D stress and material tensors

$$S^{ij} = \mu\, J^{-2/3}\left(\mathring{g}^{ij} - \frac{1}{3}\,\mathrm{tr}(\boldsymbol{C})\,\bar{C}^{ij}\right) + \frac{1}{2}K\,\left(J^2 - 1\right)\,\bar{C}^{ij}\,, \tag{39}$$

$$\begin{aligned}\mathbb{C}^{ijkl} = {} & \frac{1}{9}\,\mu\, J^{-2/3}\left(\mathrm{tr}(\boldsymbol{C})\left(2\bar{C}^{ij}\bar{C}^{kl} + 3\bar{C}^{ik}\bar{C}^{jl} + 3\bar{C}^{il}\bar{C}^{jk}\right) - 6\left(\mathring{g}^{ij}\bar{C}^{kl} + \bar{C}^{ij}\mathring{g}^{kl}\right)\right) \\ & + K\left(J^2\,\bar{C}^{ij}\bar{C}^{kl} - \frac{1}{2}(J^2 - 1)\left(\bar{C}^{ik}\bar{C}^{jl} + \bar{C}^{il}\bar{C}^{jk}\right)\right)\,.\end{aligned} \tag{40}$$

where the coefficients $\bar{C}^{ij}$ are defined by the expression $\boldsymbol{C}^{-1} = \bar{C}^{ij}\mathring{\boldsymbol{g}}_i \otimes \mathring{\boldsymbol{g}}_j$. We note that the procedure indicated in Eqns. (19)–(22) needs to be applied to (39)–(40).

We start our analysis by fixing the value of the point load to $P = 100$ and studying the performance of locally h-refined ASTS and globally h-refined NURBS for $p = 2$, 3, 4, and 5. For each order, the ASTS and NURBS meshes start with the same uniform Bézier mesh, then local and global h-refinement is performed, respectively. For the locally refined ASTS

meshes, we recursively add one more level of refinement around the point load by splitting half of the Bézier elements on the currently finest level in each parametric direction. Fig. 6 displays the five-level ASTS mesh for $p = 2$ along with the boundary conditions and the point load applied.

Fig. 7 shows the absolute value of the radial displacement under the point load versus the square root of the total number of control points for the different meshes considered. The solution seems to converge to a displacement of ~ 0.66 for all p. Nevertheless, the aim of this plot is to compare local refinement against global refinement. It is clear that the local refinement capability of ASTS meshes significantly enhances the accuracy per degree-of-freedom in comparison with their NURBS counterparts for all the orders considered. For example, if we refine the initial quintic mesh four times using both ASTS and NURBS, we end up obtaining essentially the same value for the displacement. However, the ASTS mesh has 469 control points and the NURBS mesh has 17,689 control points, that is, the ASTS mesh has 97.35% fewer control points than its NURBS counterpart. This shows the advantage of ASTS against NURBS when it comes to solve shell problems with point loads.

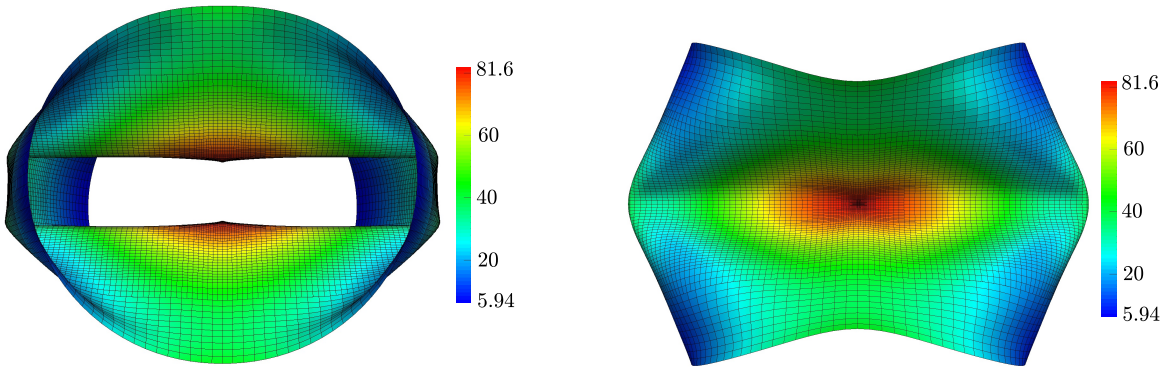
As a second test for this benchmark problem, we now increase the value of the point load to $P = 10,000$ in order to compare our load-deflection curves with previous results from the literature. The point load is applied in 50 equal load steps. We use a quintic ASTS mesh with five levels of refinement in this case. The mesh has 6,543 degrees of freedom and 1,846 elements. The total number of degrees of freedom used in our simulation is significantly lower than in references [61, 62] where 11,767 and 8,136 degrees of freedom are used, respectively. Fig. 8 shows the radial displacement of the points A and B (see Fig. 6) at the end of each load step. As it can be seen in Fig. 8, our curves match with the results given in [61, 62]. Note that, TUBA's solution (reference [62]) and our solution, which are based on Kirchhoff-Love shells, are slightly more flexible than Sansour's solution (reference [61]). Although we do not have conclusive data, the reason could be a residual locking in the Reissner-Mindlin formulation used in [61]. Finally, contour plots of the displacement magnitude for the final load are shown in Fig. 9 over the deformed cylinder together with the Bézier mesh utilized.

4.2. Pinched hemispherical shell

This example focuses on the classical pinched hemisphere under two inward and two outward opposite forces of magnitude P. We consider a hemisphere with no hole close to the pole. The bottom circumferential edge of the hemisphere is free. Due to the two planes of symmetry, only a quarter of the geometry is considered for the numerical simulation. This problem has been solved previously in [65]. We use the same material model as in the previous example [see Eq. (38)]. The parameters that define this problem are

$$R = 10, \quad t = 0.04, \quad E = 6.825 \times 10^7, \quad \nu = 0.3, \quad P = 400, \tag{41}$$

where R is the radius of the hemisphere. The point loads are applied in 50 equal load steps. We use a quartic ASTS mesh with 4,032 degrees of freedom and 1,168 elements. The local refinement capability of T-splines is used in order to avoid very small elements



(a) Displacement magnitude in front view (b) Displacement magnitude in top view

Figure 9: (Color online) Deformed cylinder for $P = 10,000$ along with the Bézier mesh from two different points of view. (a) Contour plot of the displacement magnitude from a front view. (b) Contour plot of displacement magnitude from a top view.

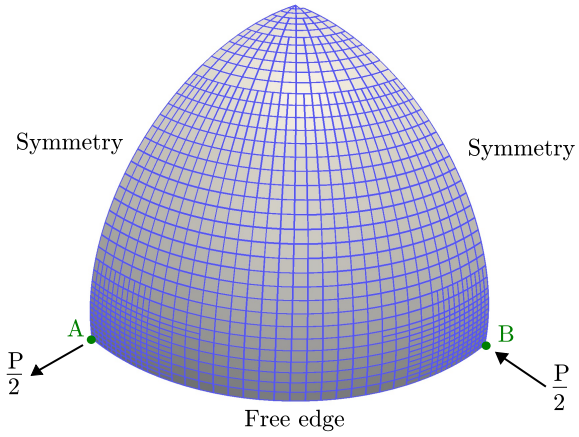


Figure 10: (Color online) Bézier mesh for the pinched hemisphere along with the boundary conditions and the point loads applied. Analysis-suitable T-splines of fourth order are used in this case.

close to the pole and to increase the resolution close to the point loads, leading to a final mesh with five levels of refinement as it can be seen in Fig. 10. Note that the Bézier elements located at the sphere's pole have one of their edges in parametric space mapped into a point in physical space. However, the presence of this degenerated point does not have a visible impact in the solution field since all quadrature points are located on the interior of the Bézier elements. We monitor the radial displacement under the two point loads through the load steps and plot them in Fig. 11 together with the data obtained by Arciniega and Reddy (reference [65]). As in Section 4.1, our solution is slightly more flexible, which may be due to the existence of a residual locking in the Reissner-Mindlin formulation used in [65]. In Fig. 12, we plot the x and y components of the displacement (see the axes orientation in the plot) over the deformed hemisphere under the total load.

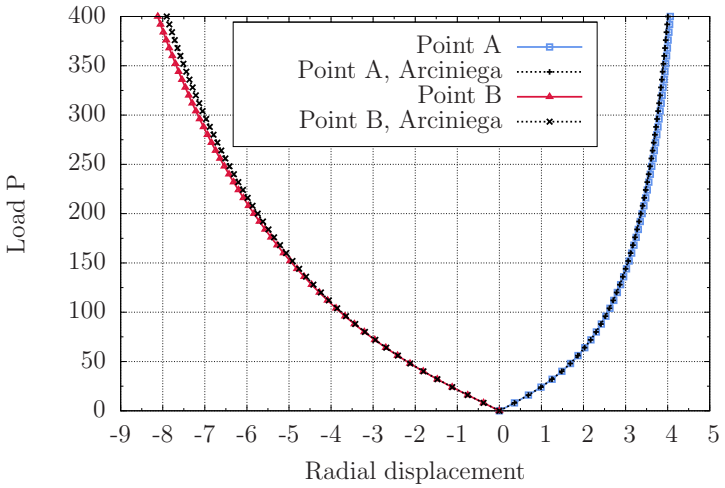
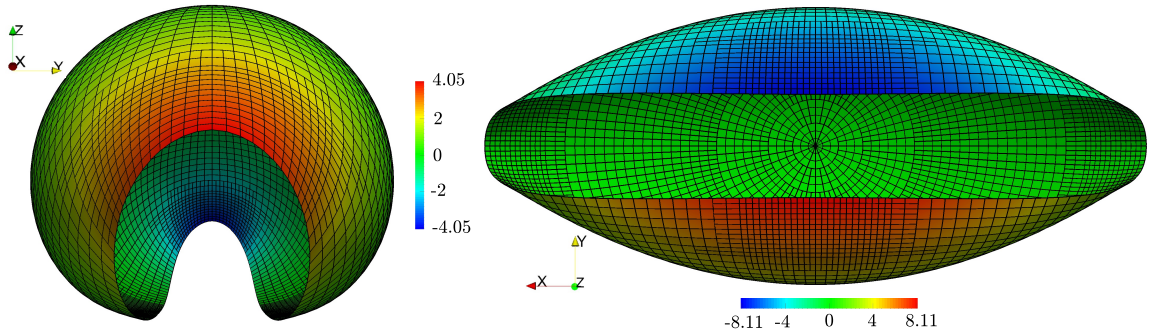


Figure 11: (Color online) Load-deflection curves at points A and B of the pinched hemispherical shell.



(a) Displacement in x direction

(b) Displacement in y direction

Figure 12: (Color online) Deformed hemisphere for $P = 400$ along with the Bézier mesh from two different points of view. (a) Contour plot of the displacement in x direction from a side view. (b) Contour plot of displacement in y direction from a bottom view.

4.3. Handling trimmed NURBS surfaces in analysis

In this Section, we illustrate our proposal to handle trimmed NURBS surfaces in shell analysis. We start with an academic example that may be modeled without trimming curves. We use this example to show the feasibility of our method. Then, we will move to a problem of practical interest. The geometry of the academic problem is defined by a square with a circular hole in its center. The shell is simply supported on both the circular edge and its four corners, and has six point loads applied (see Fig. 13 for details). The value of the point loads is given by $P = 100$. The geometry of the problem is defined by $R = 3$, $L = 20$, and $t = 0.1$, where R is the radius of circle and L is the side of the square. Due to the existing symmetry, we only model a quarter of the domain. As anticipated, this geometry may be modeled without using trimming curves. We proceed this way to generate an initial mesh that we call ASTS mesh without reparameterization [see Fig. 14(a)]. This

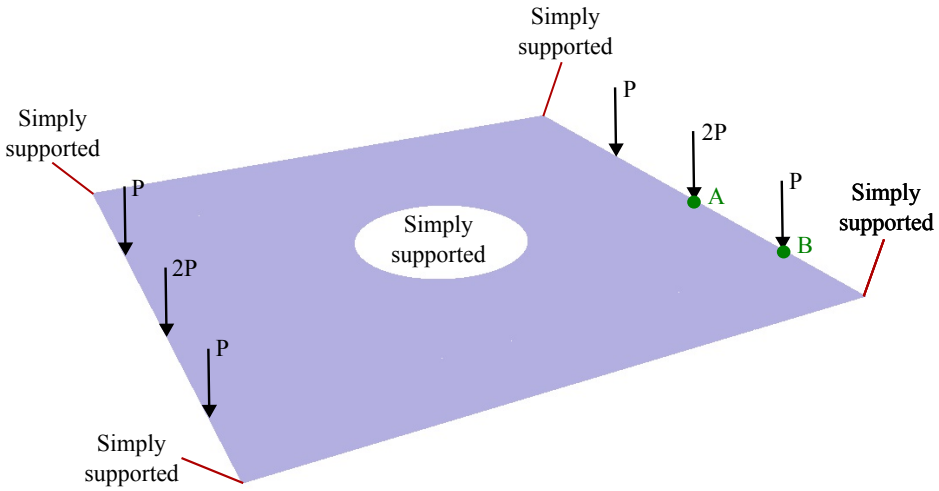
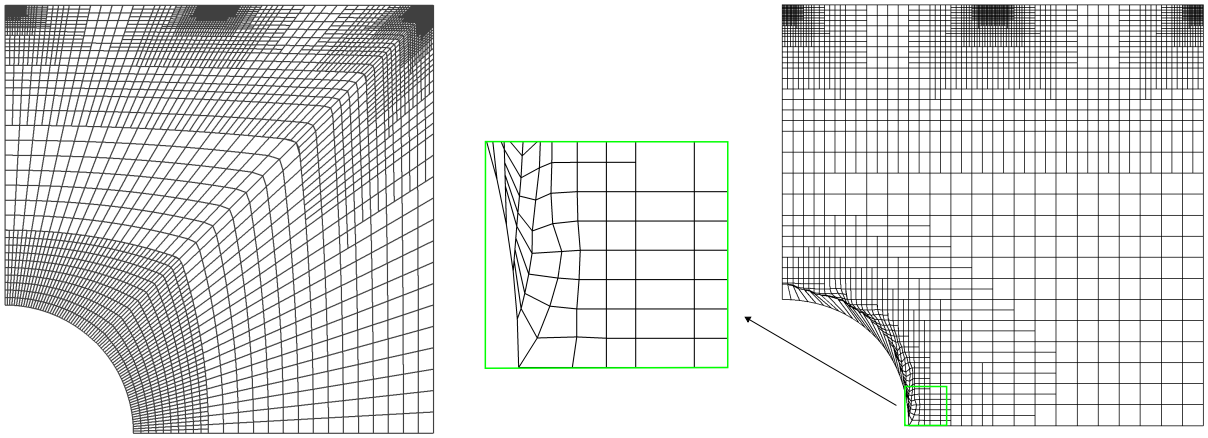


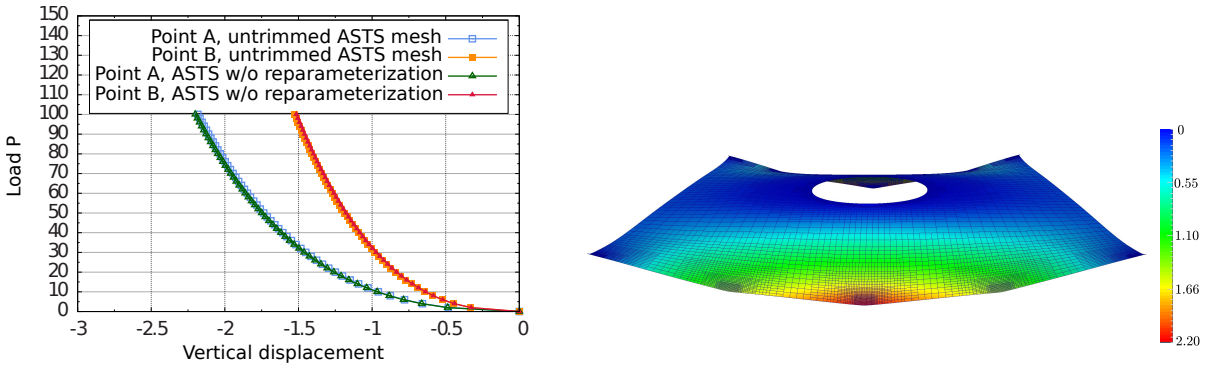
Figure 13: (Color online) Problem definition. The shell is simply supported at its four corners and at the circular edge. The three point loads are evenly distributed along the side.



(a) ASTS without reparameterization

(b) Untrimmed ASTS mesh

Figure 14: (Color online) Bézier meshes employed in the computations. (a) ASTS mesh without reparameterization. In this case, the geometry was generated without using trimming curves. (b) Untrimmed ASTS mesh. The geometry was generated by trimming a NURBS surface.



(a) Load-deflection curves

(b) Deformed geometry

Figure 15: (Color online) Analysis result for the square with a circular hole. (a) Load-displacement curves for the two mesh generation techniques. The results show the feasibility of the proposed algorithm to handle trimmed NURBS surfaces in analysis. (b) Final deformed geometry along with the displacement magnitude.

mesh is composed by 3,928 elements and 12,384 degrees of freedom. Then, we generate another mesh by representing the geometry exactly by a trimmed NURBS surface using the commercial software Rhino. Subsequently, we use the algorithm described in Section 2.5 in order to obtain an untrimmed ASTS mesh which has 2,781 elements and 8,409 degrees of freedom. The untrimmed mesh is shown in Fig. 14(b).

Here, we investigate an incompressible Neo-Hookean material characterized by the strain energy defined in Eq. (23) with

$$\psi_{el} = \frac{\mu}{2}(\mathrm{tr}(\boldsymbol{C}) - 3), \tag{42}$$

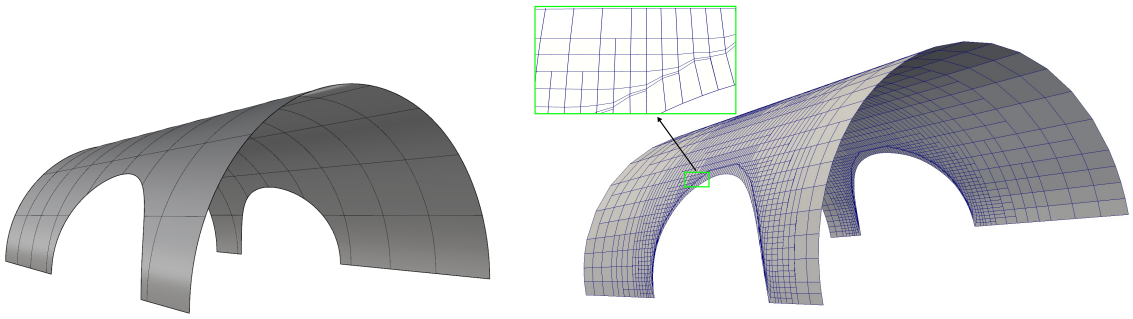
where μ is the shear modulus. Applying Eqs. (24) and (25) to the energy function (42), we obtain the following 3D stress and material tensors

$$S^{\alpha\beta} = \mu \left(\mathring{g}^{\alpha\beta} - J_o^{-2} g^{\alpha\beta} \right), \tag{43}$$

$$\hat{\mathbb{C}}^{\alpha\beta\gamma\delta} = \mu J_o^{-2} \left(2g^{\alpha\beta} g^{\gamma\delta} + g^{\alpha\gamma} g^{\beta\delta} + g^{\alpha\delta} g^{\beta\gamma} \right). \tag{44}$$

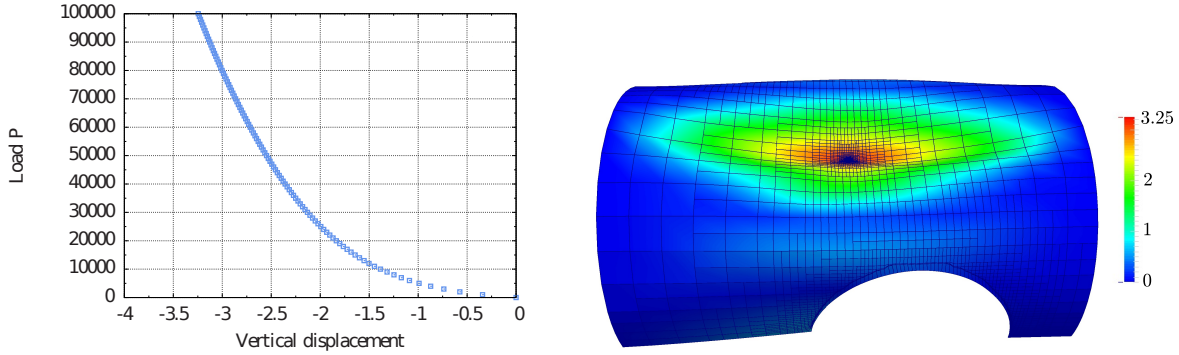
For our computations, we use the material parameter $\mu = 2 \times 10^5$. Fig. 15(a) shows the load-deflection curve for both meshes. As expected, the curves are indistinguishable at the scale of the plot, which shows that the use of the algorithm described in Section 2.5 as a way to handle trimmed NURBS surfaces does not influence the accuracy of the solution. Fig. 15(b) shows the final configuration of the shell under the total applied load.

Once the feasibility of our method to handle trimmed NURBS has been shown, we move to a case of practical significance. We start by constructing a NURBS patch with the software Rhino, which defines half of a cylinder with parameters $R = 10$, $L = 30$, and $t = 0.05$. Then, we cut it with two arbitrary B-splines curves. The final geometry is shown in Fig. 16(a). Subsequently, we output all the information related with the NURBS patch, the two trimming curves and the trimming operation in a IGES file (this



(a) Trimmed NURBS surface (b) Untrimmed ASTS surface

Figure 16: (Color online) From the trimmed NURBS surface to the untrimmed ASTS surface. (a) Initial trimmed NURBS surface. (b) Bézier mesh representation of the final untrimmed ASTS surface.



(a) Load-deflection curve (b) Deformed geometry

Figure 17: (Color online) Analysis result for the cylinder cut by two trimming curves. (a) Load-displacement curve at the location where the point load is applied. The data shows how the structure stiffens as it deforms. (b) Deformed geometry along with the displacement magnitude.

file is provided as Supplemental Material for reproducibility purposes). Then, we apply the algorithm described in Section 2.5 so as to obtain an untrimmed ASTS surface. The resulting surface, which has 2,906 Bézier elements and 7,686 control points, is represented in Fig. 16(b). Finally, we consider that the shell is simply supported at the two (circular) edges that have not been cut by trimming curves and apply a vertical point load $P = 10^5$ (pointing downwards) at the center of the surface. Regarding the constitutive theory, we use the Neo-Hookean material presented in Eq. (38) with parameters $E = 5 \times 10^7$ and $\nu = 0.3$. Fig. 17(a) shows the vertical displacement under the load for each of the 100 load steps and Fig. 17(b) displays the deformed Bézier mesh of the shell.

5. Conclusions and future work

We believe that ASTS are a key ingredient toward a seamless integration of CAD and analysis for shell-mechanics applications. The local refinement capabilities of ASTS provide flexibility in order to have increased spatial resolution where it is needed, e.g., close to point

loads. Moreover, we use the local refinement capabilities of ASTS to deal with trimmed surfaces coming from the CAD programs. It is also important to take into account that, in contrast to NURBS, ASTS are a novel technology and, therefore, they have not been implemented or are currently under implementation in some of the major CAD programs. Finally, ASTS constitute one of the most promising ways to represent surfaces with arbitrary topology in a compact format. Currently, watertight models, that is, $\mathcal{C}^0$-continuous models are always achievable through the use of extraordinary points. This property solved one of the most important issues in CAD industry, namely, the gap/overlap problem when it comes to model complex geometries with a combination of NURBS patches. Nevertheless, global $\mathcal{C}^0$-continuous spaces are not enough for some analysis applications as Kirchhoff-Love shells. It is possible to impose $\mathcal{G}^1$-continuity of both the geometry and the solution parametrization using the isoparametric concept at the spoke edges of an extraordinary point [56], which leads to $\mathcal{C}^1$ continuity in physical space of the solution field [66]. However, the current methods for handling extraordinary points do not necessarily guarantee optimal convergence rates under h-refinement [67, 68]. In our opinion, developing new methods for handling extraordinary points which lead to both $\mathcal{C}^1$-continuity in physical space and optimal convergence rates is the main last step toward a real ASTS-based seamless integration of CAD and analysis for Kirchhoff-Love applications and we hope to address it in the near future.

Acknowledgements

HG and HC were partially supported by the European Research Council through the FP7 Ideas Starting Grant (project # 307201). HG was also partially supported by Consellería de Cultura, Educación e Ordenación Universiaria, Ministerio de Economía y Competitividad (project # DPI2013-44406-R), cofinanced with FEDER funds. YZ and LL were partially supported by the PECASE Award N00014-14-10234. AR and JK were partially supported by the European Research Council through the FP7 Ideas Starting Grant (project # 259229). This support is gratefully acknowledged. Also, we would like to thank Dr. Reddy, Dr. Tiago, and Dr. Ivannikov for facilitating us the data that we have reproduced in this article in order to make comparisons, as well as Dr. Sangalli for some fruitful discussions about $\mathcal{C}^1$-continuity and extraordinary points. Finally, we acknowledge the open source scientific library PETSc and their developers.

References

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

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

[3] Y. Bazilevs, V. Calo, J. Cottrell, J. Evans, T. Hughes, S. Lipton, M. Scott, T. Sederberg, Isogeometric analysis using T-splines, Computer Methods in Applied Mechanics and Engineering 199 (2010) 229–263.

[4] Y. Zhang, W. Wang, T. J. Hughes, Solid T-spline construction from boundary representations for genus-zero geometry, Computer Methods in Applied Mechanics and Engineering 249-252 (2012) 185–197.

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

[6] W. Wang, Y. Zhang, L. Liu, T. J. Hughes, Trivariate solid T-spline construction from boundary triangulations with arbitrary genus topology, Computer-Aided Design 45 (2) (2013) 351–360.

[7] L. Liu, Y. Zhang, T. J. Hughes, M. A. Scott, T. W. Sederberg, Volumetric T-spline construction using Boolean operations, Engineering with Computers 30 (2014) 425–439.

[8] L. Liu, Y. Zhang, Y. Liu, W. Wang, Feature-preserving T-mesh construction using skeleton-based polycubes, Computer-Aided Design 58 (2015) 162–172.

[9] 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.

[10] H. Gomez, X. Nogueira, An unconditionally energy-stable method for the phase field crystal equation, Computer Methods in Applied Mechanics and Engineering 249 (2012) 52–61.

[11] H. Gomez, T. J. 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.

[12] J. Bueno, C. Bona-Casas, Y. Bazilevs, H. Gomez, Interaction of complex fluids and solids: theory, algorithms and application to phase-change-driven implosion, Computational Mechanics 55 (2015) 1105–1118.

[13] H. Gomez, L. Cueto-Felgueroso, R. Juanes, Three-dimensional simulation of unstable gravity-driven infiltration of water into a porous medium, Journal of Computational Physics 238 (2013) 217–239.

[14] G. Vilanova, I. Colominas, H. Gomez, Capillary networks in tumor angiogenesis: From discrete endothelial cells to phase-field averaged descriptions via isogeometric analysis, International Journal for Numerical Methods in Biomedical Engineering 29 (2013) 1015–1037.

[15] G. Vilanova, I. Colominas, H. Gomez, Coupling of discrete random walks and continuous modeling for three-dimensional tumor-induced angiogenesis, Computational Mechanics 53 (2013) 449–464.

[16] R. Dhote, H. Gomez, R. Melnik, J. Zu, 3D coupled thermo-mechanical phase-field modeling of shape memory alloy dynamics via isogeometric analysis, Computers & Structures.

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

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

[19] H. Casquero, L. Lei, J. Zhang, A. Reali, H. Gomez, Isogeometric collocation using analysis-suitable T-splines of arbitrary degree, submitted for publication.

[20] D. Schillinger, J. Evans, A. Reali, M. Scott, T. J. R. Hughes, Isogeometric collocation: Cost comparison with Galerkin methods and extension to adaptive hierarchical NURBS discretizations, Computer Methods in Applied Mechanics and Engineering 267 (2013) 170–232.

[21] H. Gomez, A. Reali, G. Sangalli, Accurate, efficient, and (iso)geometrically flexible collocation methods for phase-field models, Journal of Computational Physics 262 (2014) 153–171.

[22] H. Casquero, L. Lei, C. Bona-Casas, Y. Zhang, H. Gomez, A hybrid variational-collocation immersed method for fluid-structure interaction using unstructured T-splines, International Journal for Numerical Methods in Engineering (2015). DOI: 10.1002/nme.5004.

[23] A. Reali, H. Gomez, An isogeometric collocation approach for Bernoulli-Euler beams and Kirchhoff plates, Computer Methods in Applied Mechanics and Engineering 284 (2015) 623–636.

[24] J. Kiendl, F. Auricchio, T. Hughes, A. Reali, Single-variable formulations and isogeometric discretizations for shear deformable beams, Computer Methods in Applied Mechanics and Engineering 284 (2015) 988 – 1004.

[25] 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.

[26] H. Casquero, C. Bona-Casas, H. Gomez, A NURBS-based immersed methodology for fluid-structure interaction, Computer Methods in Applied Mechanics and Engineering 284 (2015) 943–970.

[27] 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.

[28] 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.

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

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

[31] 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.

[32] Y. Bazilevs, M.-C. Hsu, J. Kiendl, R. Wuchner, K.-U. Bletzinger, 3D simulation of wind turbine rotors at full scale. part ii: Fluid-structure interaction modeling with composite blades, International Journal for Numerical Methods in Fluids 65 (2011) 236–253.

[33] D. Kamensky, M.-C. Hsu, D. Schillinger, J. A. Evans, A. Aggarwal, Y. Bazilevs, M. S. Sacks, T. J. Hughes, An immersogeometric variational framework for fluid-structure interaction: Application to bioprosthetic heart valves, Computer Methods in Applied Mechanics and Engineering 284 (2015) 1005–1053.

[34] M.-C. Hsu, D. Kamensky, Y. Bazilevs, M. Sacks, T. Hughes, Fluid-structure interaction analysis of bioprosthetic heart valves: Significance of arterial wall deformation, Computational Mechanics 54 (2014) 1055–1071.

[35] M.-C. Hsu, D. Kamensky, F. Xu, J. Kiendl, C. Wang, M. Wu, J. Mineroff, A. Reali, Y. Bazilevs, M. Sacks, Dynamic and fluid-structure interaction simulations of bioprosthetic heart valves using parametric design with T-splines and Fung-type material models, Computational Mechanics 55 (6) (2015) 1211–1225.

[36] M. Bischoff, K.-U. Bletzinger, W. A. Wall, E. Ramm, Models and Finite Elements for Thin-Walled Structures, John Wiley & Sons, Ltd, 2004.

[37] 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.

[38] 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.

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

[40] L. Lei, H. Casquero, H. Gomez, Y. Zhang, Hybrid-degree weighted T-splines and their application in isogeometric analysis, submitted for publication.

[41] M. Ruess, D. Schillinger, Y. Bazilevs, V. Varduhn, E. Rank, Weakly enforced essential boundary conditions for NURBS-embedded and trimmed NURBS geometries on the basis of the finite cell method, International Journal for Numerical Methods in Engineering 95 (10) (2013) 811–846.

[42] L. Liu, Y. Zhang, X. Wei, Weighted T-spline and its application in reparameterizing trimmed NURBS surfaces, Computer Methods in Applied Mechanics and EngineeringUnder review.

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

[44] 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.

[45] T. W. Sederberg, G. T. Finnigan, X. Li, H. Lin, H. Ipson, Watertight trimmed nurbs, in: ACM SIGGRAPH 2008 Papers, SIGGRAPH '08, ACM, New York, NY, USA, 2008, pp. 79:1–79:8.

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

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

[48] 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.

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

[50] F. Cirak, M. Ortiz, Fully C1-conforming subdivision elements for finite deformation thin-shell analysis, International Journal for Numerical Methods in Engineering 51 (7) (2001) 813–833.

[51] 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.

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

[53] 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.

[54] 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.

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

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

[57] Y. Basar, D. Weichert, Nonlinear continuum mechanics of solids: fundamental mathematical and physical concepts, Springer Science & Business Media, 2013.

[58] G. A. Holzapfel, Nonlinear Solid Mechanics: A Continuum Approach for Engineering, Wiley, Chichester, 2000.

[59] 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).

[60] 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).

[61] C. Sansour, F. G. Kollmann, Families of 4-node and 9-node finite elements for a finite deformation shell theory. An assesment of hybrid stress, hybrid strain and enhanced strain elements, Computational Mechanics 24 (2000) 435–447.

[62] V. Ivannikov, C. Tiago, P. Pimenta, Generalization of the TUBA plate finite elements to the geometrically exact kirchhoff–love shell model, Computer Methods in Applied Mechanics and Engineering 294 (2015) 210 – 244.

[63] K. Sze, X. Liu, S. Lo, Popular benchmark problems for geometric nonlinear analysis of shells, Finite Elements in Analysis and Design 40 (2004) 1551 – 1569.

[64] J. Simo, C. Miehe, Associative coupled thermoplasticity at finite strains: Formulation, numerical analysis and implementation, Computer Methods in Applied Mechanics and Engineering 98 (1) (1992) 41 – 104.

[65] 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.

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

[67] T. Nguyen, K. Karciauskas, J. Peters, A comparative study of several classical, discrete differential and isogeometric methods for solving Poisson's equation on the disk, Axioms 3 (2) (2014) 280.

[68] A. Collin, G. Sangalli, T. Takacs, Approximation properties of multi-patch C^1 isogeometric spaces, arXiv preprint arXiv:1509.07619.