Bézier Segmentation of T-spline Solids in Parametric Domain

Mayi Guo[1], Wei Wang[2] , Gang Zhao[3] and Xiaoxiao Du[4]

[1] Beihang University, windowsgmy@126.com
[2] Beihang University, jrrt@buaa.edu.cn
[3] Beihang University, zhaog@buaa.edu.cn
[4] Beihang University, duxiaoxiao@buaa.edu.cn

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

Abstract. In order to solve the problem of Bézier segmentation of T-spline solids in parametric domains, a T-spline solid Bézier segmentation algorithm based on dimension decomposition is proposed. Firstly, the algorithm simplifies the problem by decomposing the three-dimensional parametric domain into three one-dimensional parametric domains. Then, the definition domain of a T-spline solid element is segmented according to the knot vector of the T-spline vertex. Finally, the segmentation results are combined with the Bézier extraction matrix to complete the extraction of the T-spline solid. The effectiveness of the proposed algorithm is verified by examples.

Keywords: T-splines Solid; Isogeometric Analysis; Bézier Extraction; Bézier Segmentation;
DOI: https://doi.org/10.14733/cadaps.2020.502-512

1 INTRODUCTION

Isogeometric analysis, a numerical method based on spline function proposed in 2005[7], uses the same geometric representation for both engineering design model and analysis model, thus eliminating the huge model difference between design and analysis, and improving the efficiency and accuracy of the overall product design process [13]. However, the traditional geometry model still has limitations in expressing three-dimensional objects. This is due to the fact that all the current CAD systems use NURBS or T-spline bounding surface set to represent solid objects, which lack the numerical parameterization of the internal domain. This kind of defect has become the bottleneck of the integration of design and analysis in many engineering fields. [2],[3],[5] Several papers have studied isogeometric analysis using non-uniform rational B-splines (NURBS) solids [4],[7],[9],[15]. However, NURBS has some drawbacks which limit its application in isogeometric analysis, for example, NURBS [9] does not support local refinement and gaps often occur between two neighboring NURBS surface patches. To overcome these limitations, Sederberg invented T-splines [11], which naturally support local refinement [12] by introducing T-junctions. In [7],[10], T-spline is introduced into isogeometric analysis.

The T-spline solid inherits the advantages of T-spline surface, which circumvent the shortcomings of NURBS in expressing complex geometries, such as the need for trimming, and the inability of local refinement [11-12]. T-spline solid construction is one of the key technologies to support isogeometric analysis philosophy to achieve practical level endeavors. A few works have been devoted to constructing T-spline solid and applying T-spline solid in isogeometric analysis. Zhang et al. proposed a method of transforming genus-zero boundary mesh models into T-spline solid models [17]. Wang et al. proposed a method of transforming arbitrary genus boundary mesh models into the T-spline solid models and performed isogeometric analysis of the T-spline solids obtained [13]. Zhang et al. completed the work of transforming special T-spline surfaces into T-spline solids [16]. The above research on T-spline solid focuses mainly on the construction process of T-spline solid. In [13], isogeometric analysis of T-spline solids is performed, but the emphasis is on visualizing the analysis results and the detailed steps in the analysis process are not introduced. As a key algorithm for T-spline solids applied to isogeometric analysis, Bézier extraction was proposed by Scott et al. [10]. When this algorithm is applied to isogeometric analysis of T-spline solids, there are two main steps: (1) Bézier segmentation of three-dimensional T-mesh in the parametric domain. (2) Get the Bézier extraction matrix corresponding to each blending function of the T-spline solid.

Scott et al. [10] gives a detailed description of the implementation process of step (2), but the specific implementation process of step (1) is not given. Other literatures on T-spline solid also lack specific algorithms. Although there is some research on the algorithm of Bézier segmentation of two-dimensional T-meshes, due to the intrinsic complexity of three-dimensional T-mesh the algorithm cannot be directly extended to a T-spline solid. In this paper, the problem of Bézier segmentation of the T-spline solid is studied and the parametric domain Bézier segmentation algorithm of a T-spline solid was proposed. Some examples are demonstrated to verify the algorithm.

The rest of paper is organized as follows. In section 2, we give a brief introduction to T-splines and T-spline solids. The Bézier extraction of T-spline solids is briefly explained in section 3. Section 4 is dedicated to describing the Bézier segmentation algorithm of T-spline solids in the parametric domain. Section 5 shows some numerical examples to confirm the algorithm's effectiveness. Conclusions and future work are presented in section 6.

2 T-SPLINES AND T-SPLINE SOLIDS

NURBS has been widely used in the field of CAD due to its excellent mathematical and algorithmic properties. However, it is to a certain extent limited by its tensor product topological structure. T-splines were introduced into the CAD community as a generalization of NURBS to overcome their limitations. Flexible topology helps T-splines to model complex design as one single watertight geometry without superfluous control points. In addition, while preserving the exact geometry of T-splines, a local refinement is feasible. All of these superiorities make T-splines ideal for isogeometric analysis. An example of a T-spline surface with T-junction (red node), is shown in Figure 1.

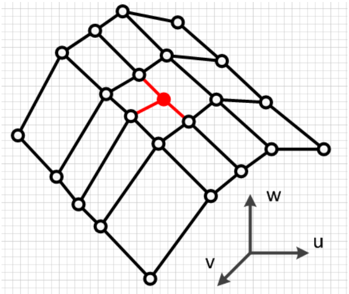


Figure 1: A T-spline surface and its T-junction.

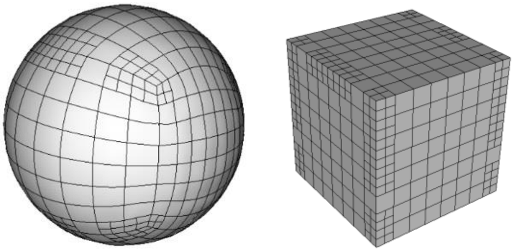
T-spline solid is a trivariate T-spline structure constructed in three-dimensional space. T-spline solid inherits the advantages of flexible topology and local refinement of bivariate T-splines, which makes it very attractive in isogeometric analysis. The basis function of rational T-spline solids is defined as follows:

$$\boldsymbol{T}(u,v,w)=\sum_{i=0}^{n}P_iR_i(u,v,w),(u,v,w)\in\Omega \tag{2.1}$$

$$R_i(u,v,w)=\frac{w_iN_i^{\,k}(u)N_i^{\,l}(v)N_i^{\,m}(w)}{\sum_{j=0}^{n}w_iN_j^{\,k}(u)N_j^{\,l}(v)N_j^{\,m}(w)} \tag{2.2}$$

$R_i(u,v,w)$ is the blending function corresponding to the control point $\boldsymbol{P}_i$, ω_i is the weight of control point $\boldsymbol{P}_i$. $N_i^{\,k}(u)$ is the B-spline basis function corresponding to the control point $\boldsymbol{P}_i$ with k degree in u-direction. The definition of T-splines is presented in detail in [11].

The three-dimensional T-mesh of a T-spline solid is called a spatial T-mesh. It contains the spatial coordinates and topology structure of all control points of the T-spline solid, as shown in Figure 2(a). Each control point in the T-spline solid corresponds to a parametric point. The topology structure of these parametric points is the same as that of control points, and is called parametric domain T-mesh, as shown in Figure 2(b).



(a) (b)

Figure 2: (a) Spatial T-mesh, (b) Parametric domain T-mesh.

The parametric domain T-mesh of the T-spline solid consists of two kinds of basic geometric information, hexahedron element (cube), which is segmented by isoparametric lines, and parametric point element (vertex) on T-mesh. In Figure 3 is a simple demonstration of cube and vertex of parametric domain T-mesh.

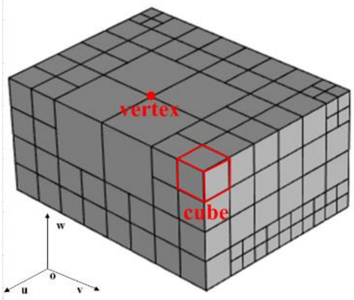


Figure 3: Cube and vertex.

Each cube has minimum and maximum values (u_min, u_max, v_min, v_max, w_min, w_max) in three directions of the parametric domain. Each vertex in the parametric domain of the T-mesh contains knot vectors in three directions (u_knot, v_knot, w_knot). An example of a cubic T-spline solid is shown in Figure 4. In this paper, the cubic T-spline solid is selected as the research object, but this algorithm is not limited to the cubic T-spline solid.

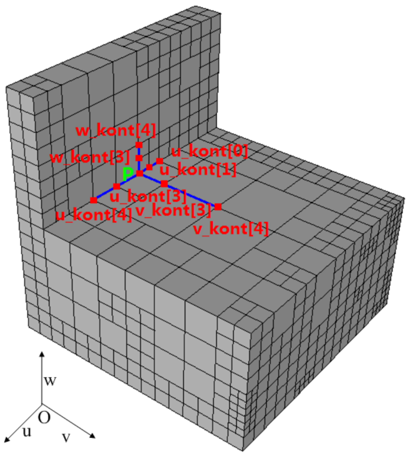


Figure 4: Knot vectors of vertex P.

There are many ways to generate T-spline solid, such as transforming boundary T-splines to T-spline solid [16], transforming boundary surface triangulations to T-spline solid [13], and creating T-spline solid according to the definition of trivariate T-spline. Once a T-spline solid is constructed, it is already equipped with parameterization data and has its solid data structures that store topological and geometrical information, such as the definition domains of cubes and the knot vectors of vertices mentioned above. But such T-spline solid cannot be directly used for isogeometric analysis, because not all cubes segmented by isoparametric lines in T-spline solid can be treated equivalently as one computation element in FEA [10]. Therefore, before performing isogeometric analysis of T-spline solid, Bézier extraction should be carried out first.

3 BÉZIER SEGMENTATION OF T-SPLINE SOLIDS

The Bézier extraction helps us develop T-splines from the finite element point-of-view. The operation for extracting the linear operator which maps the Bernstein polynomial basis on Bézier solids to the global T-spline basis is referred to as Bézier extraction. The Bézier extraction operation provides an element structure for isogeometric analysis which can be easily incorporated into existing finite element codes.

As mentioned in section 1, there are two main steps when the Bézier extraction is applied to the isogeometric analysis of T-spline solid: (1) Bézier segmentation of three-dimensional T-mesh in parametric domain. (2) Get the Bézier extraction matrix corresponding to each blending function of three-dimensional T-spline. Step (1) divides parametric domain T-mesh into Bézier solids. Step (2) associates vertices on T-mesh with nodes on Bézier solids by Bézier extraction matrix. Relevant studies have detailed descriptions of step (2), in this paper we will focus on the specific implementation process of step (1).

The essential function of Bézier extraction is to transform T-spline solids into Bézier solids, but not each cube on parametric T-mesh can be represented by a single rational Bézier element. This phenomenon is illustrated by a common example, as shown in Figure 5. Figure 5 is a parametric domain T-mesh of a cubic T-spline solid. V is a parametric vertex, C is a cube in the parametric domain T-mesh, and the red region is the definition domain of the blending function corresponding to the parametric vertex V. According to the differentiability of B-spline basis function, the internal C^{∞} continuity of T-spline solid corresponding to cube C cannot be achieved because of the existence of the dash line wherein the continuity is reduced [10], while in a rational Bézier solid C^{∞} continuity can be satisfied. Therefore, the T-spline solid corresponding to cube C should be segmented further into different Bézier solids by which the isogeometric analysis could be implemented. Because the T-junctions may appear frequently in a parametric T-spline solid, this step can be very delicate based on how complex the solid will be and the local details of the parametric T-spline solid. But in order to perform Bézier extraction of T-spline solid, the T-spline solid must be segmented first.

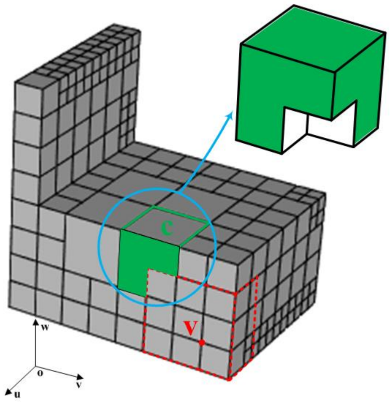


Figure 5: L-shaped region generated by Bézier segmentation.

If the brute force algorithm is used to traverse every vertex for Bézier segmentation, as shown in Figure 5, when the vertex V is reached, the segmentation will result in the green L-shaped volume which needs a different data structure to store from the regular cube volume. And this L-shape is not the final result so when encounter a new vertex that has knot interval interferes with it, it will be further segmented and some of the data related with it will be recalculated. This will make the algorithm clumsy and difficult to check so some craft should be invented to upgrade it. This will be done in next section of this paper.

After the Bézier segmentation, the extraction operators can be obtained by the method proposed in [10]. With the basis functions defined by Equations (2.1) (2.2), we now define the element geometric map $\boldsymbol{T}^e$ from the parent element domain onto the physical domain as:

$$\boldsymbol{T}^e(u,v,w)=\sum_{i=0}^{n}P_i^e R_i^e(u,v,w),(u,v,w)\in\Omega^e \tag{3.1}$$

$$R_i^e(u,v,w)=\frac{w_i^e N_i^e(u,v,w)}{\sum_{j=0}^{n} w_i^e N_i^e(u,v,w)} \tag{3.2}$$

where $N_i^e(u,v,w)$ is the ith T-spline basis function over element e, and P_i^e and w_i^e are the control point and weight, respectively. The Bézier extraction operator for T-splines can exactly represent the localized T-spline basis $N^e(u,v,w)$, over each element e in terms of a set of Bernstein polynomials $B^e(u,v,w)$. In matrix-vector form, the T-spline basis function $N^e(u,v,w)$ can be written as

$$N^e(u,v,w) = C^e B^e(u,v,w) \tag{3.3}$$

Through Bézier extraction, each cube in T-spline solids is transformed into Bézier solids and the internal C^∞ continuity of each element is obtained, so that isogeometric analysis can be carried out. More details about the isogeometric finite element data structures based on Bézier extraction of T-splines are presented in [10].

4 T-SPLINE SOLID BÉZIER SEGMENTATION ALGORITHM

In order to perform Bézier segmentation of the parametric domain T-mesh efficiently and avoid the difficulty of storing the intermediate segmentation results, this paper presents a Bézier segmentation algorithm for parametric domain T-mesh of T-spline solids. There is a flow chart of the algorithm, as shown in the Figure 6.

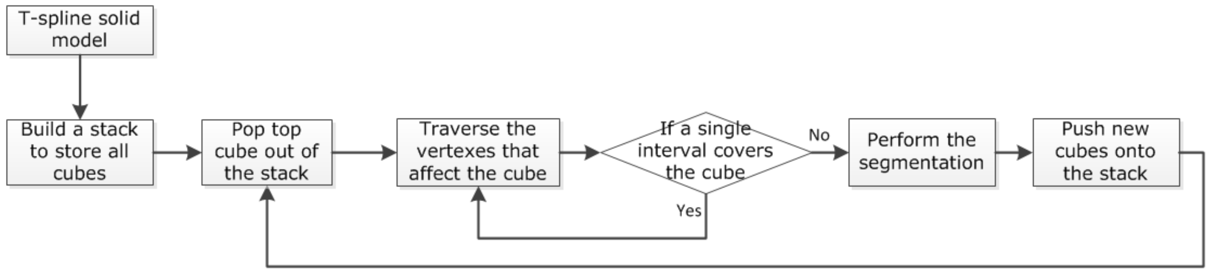


Figure 6: An overview of the T-spline solid Bézier segmentation algorithm.

Algorithms 1: T-spline Solid Parametric Domain T-mesh Bézier Segmentation

STEP1: Build a stack to store all cubes in the parametric domain T-mesh.

STEP2: If the stack is not empty, the cube on the top of the stack is popped out of the stack. Traverse the vertices that affect the cube and execute STEP3 for the current vertex.

STEP3: Calculate the knot vector of the current vertex. If a single interval of the knot vector completely covers the cube, go to the next vertex. Otherwise, execute STEP4.

STEP4: Perform the segmentation of the current cube (using *Algorithm2*), all the new cubes generated by segmentation are pushed onto the stack and return to STEP2.

An important step of the algorithm is to determine whether a single interval of the knot vector completely covers the cube. The specific method is to judge the overlap relation of the cube's parametric domain and the vertex's knot vector in three parametric directions respectively. We realize the algorithm by taking all the three possible situations into consideration, as shown in Figure 7, wherein the u-direction is taken as an example, without loss of generality.

Algorithm2: Parametric domain T-mesh cube segmentation algorithm

STEP1: The segmentation of the cube by the current vertex is decomposed into three directions: u, v and w. STEP2 is executed in three directions respectively.

STEP2: Taking the u-direction as an example, the u-direction of the cube is segmented by the u-direction knot vector (u_knot) of the vertex. There are three possible situations as shown in Figure 8 (u_min, u_max are the minimum and maximum values of the cube's parametric coordinates in u-direction). Based on these situations the cube could be segmented along the dash line as Figure 8 shows.

STEP3: The segmentations in u, v and w directions are superimposed to obtain the segmentation results.

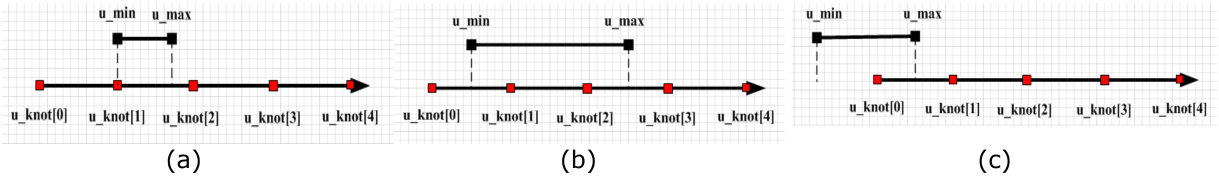


Figure 7: (a) A single interval of the knot vector completely covers the cube in u-direction, (b) A single interval of the knot vector cannot completely cover the cube in u-direction, (c) Another situation in which single interval of the knot vector cannot completely cover the cube in u-direction.

The segmentation results generated by the *Algorithm2* are regular hexahedrons in parametric space, thus solving the problem of producing L-shaped or more complex spatial volumes in the process of parametric space segmentation. In addition, this process only addresses each original cube once, and can complete the segmentation of T-mesh in the parametric domain without repetition or omission. There is an example to illustrate the segmentation of cube C by the knot vectors of vertex V, as shown in Figure 9.

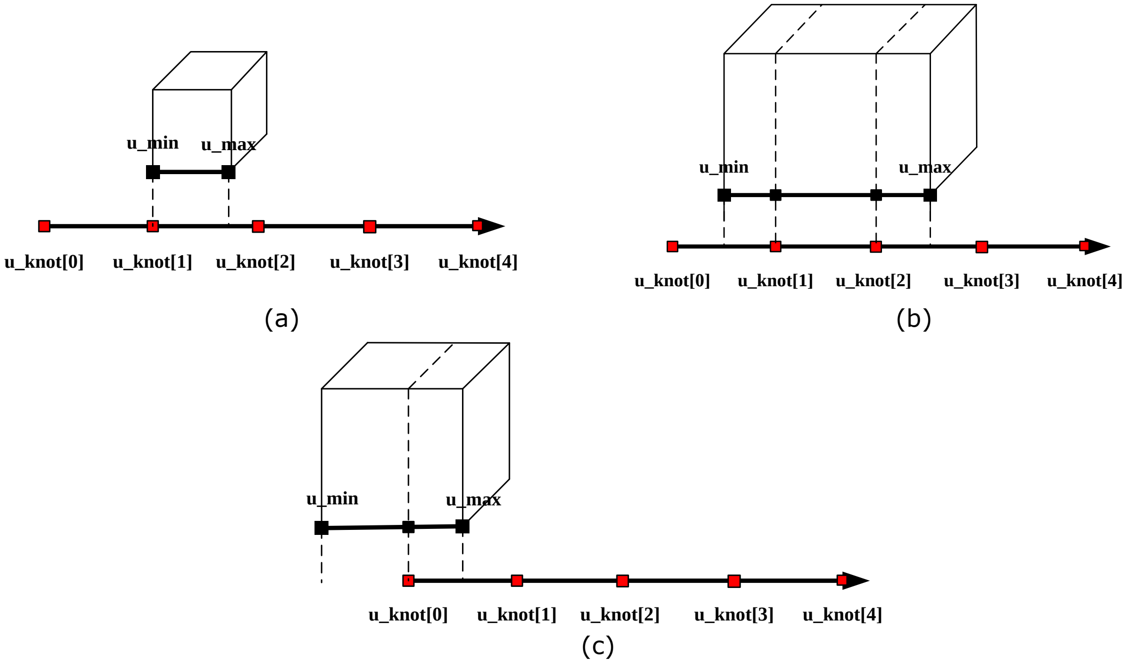
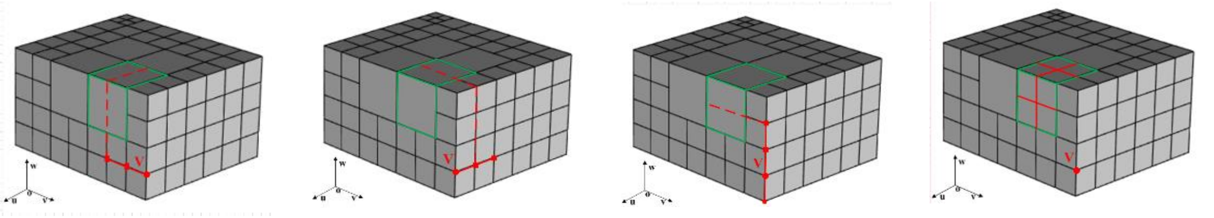


Figure 8: (a) Cube segmentation in u-direction (remain unchanged), (b) Cube segmentation in u-direction (segmented by u_knot[1] and u_knot[2]), (c) Cube segmentation in u-direction (segmented by u_knot[0]).



(a) (b) (c) (d)

Figure 9: (a) Segmentation in v-direction, (b) Segmentation in u-direction, (c) Segmentation in w-direction, (d) Segmentation result.

5 NUMERICAL EXAMPLES AND ISOGEOMETRIC ANALYSIS

In order to verify the validity of the method proposed in this paper, the Bézier extraction algorithm of T-spline solids is implemented by C++ programming language on a PC equipped with an Intel i5-6500 processor and 8 GB main memory. Firstly, the parametric domain T-mesh of T-spline model is processed by the Bézier segmentation algorithm proposed in this paper. After that, combined with the algorithm of extracting Bézier operator proposed in reference [6], the Bézier extraction of T-spline solids model is completed. Finally, the isogeometric analysis of T-spline solids can be performed. Statistics for all the tested models in the Bézier segmentation process are shown in Table 1. Basically, the time consumed in Bézier segmentation is dependent on the scale of the model, but as shown here, the result is highly related with the complexity of the model's geometry, as in complex situation, more segmentation will be executed so resulting in more time consumption. This is the case for the sphere model and the head model, which have similar number of T-mesh vertices but because the head model has more details it needs more segmentation operations and generate more Bézier elements, so the time consumed is sharply increased. When compare the sphere model with the bunny model, we can come to know that when a model is more complex, it has more vertices and cubes, so the number of vertices and cubes affects the time of segmentation directly.

Model	*T-mesh vertices*	*T-mesh cubes*	*Bézier elements*	*Time of segmentation (s)*
sphere	1229	793	1324	5.28
bunny	3433	1105	1788	21.37
head	1624	977	3415	19.25

Table 1: Statistics of the tested models.

We have developed a 3D isogeometric analysis solver for static mechanics analysis, which uses rational T-splines as the basis, and we used it to test the Bézier segmentation results of T-spline solids. For all the models, we fix all the control points on the bottom and apply uniformly distributed load on the top. The Young's modulus $E = 200GPa$, and the Poisson's ratio $\nu = 0.3$ are used for analysis. Bézier segmentation results and displacement results of z-direction from the isogeometric analysis are given in Figures 10-12.

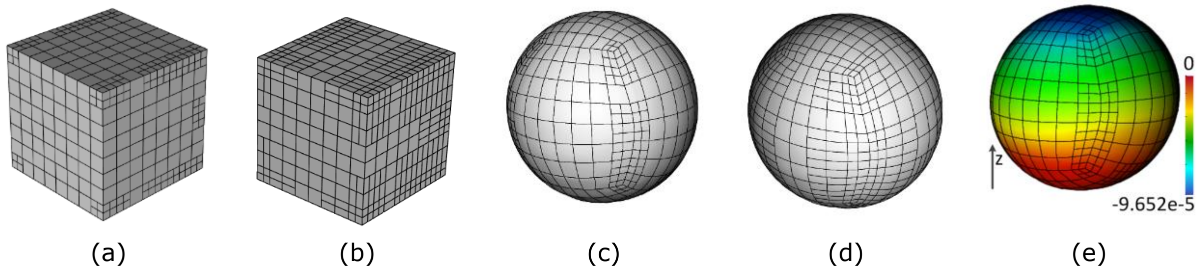


Figure 10: (a) Parametric domain T-mesh of sphere model, (b) Bézier segmentation of sphere model in parametric domain, (c) T-spline solid of sphere model, (d) Bézier segmentation of sphere model, (e) Isogeometric analysis result (displacement of z-direction) of sphere model.

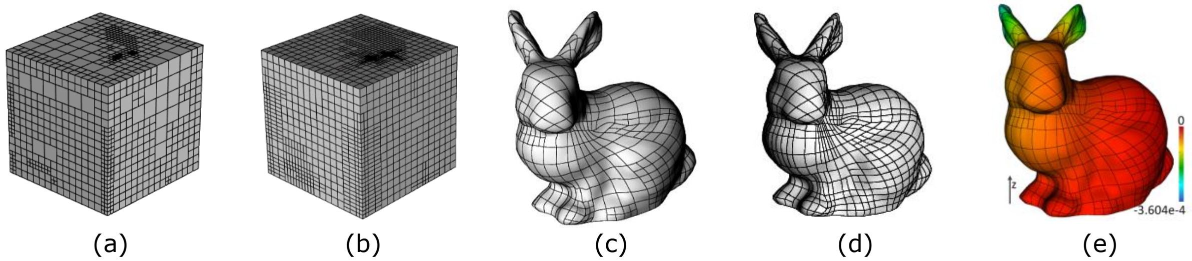


Figure 11: (a) Parametric domain T-mesh of bunny model, (b) Bézier segmentation of bunny model in parametric domain, (c) T-spline solid of bunny model, (d) Bézier segmentation of bunny model, (e) Isogeometric analysis result (displacement of z-direction) of bunny model.

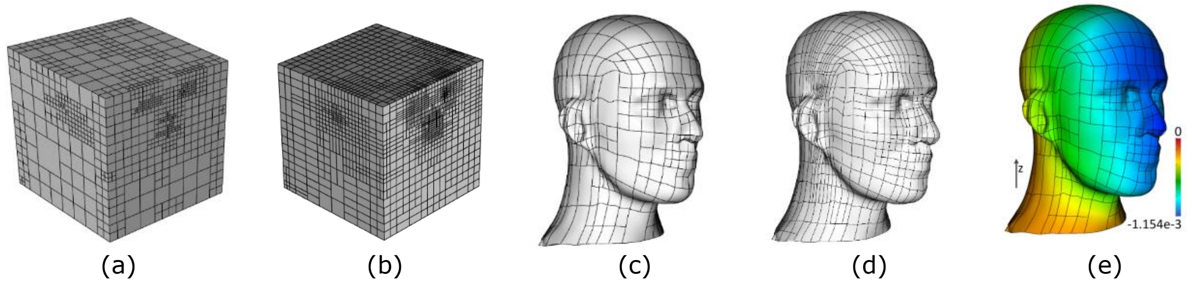


Figure 12: (a) Parametric domain T-mesh of head model, (b) Bézier segmentation of head model in parametric domain, (c) T-spline solid of head model, (d) Bézier segmentation of head model, (e) Isogeometric analysis result (displacement of z-direction) of head model.

6 CONCLUSIONS

By far NURBS is still the mathematical basis for prevailing CAD software, though T-splines have shown obvious potentials in certain technical branches, especially those wherein watertight models are heavily dependent on. But the simple and clear algorithms as well as the easy-to-do data structure in NURBS are also invaluable for quick and high-efficiency development activities, which the T-splines still can't match. Maybe in the future a mixed approach combining both NURBS and T-splines is possible, and the most reasonable pattern of coordinating the two will be a valuable topic. In this paper, we propose a Bézier segmentation algorithm in the parametric domain of T-spline solids for the purpose of employing T-spline in isogeometric analysis, for which basically only watertight models can be utilized. This algorithm can achieve Bézier segmentation of T-spline solids only by means of simple data structure and process. Our research supplements the brief description of Bézier segmentation in relevant papers. Combining with relevant literatures, we can achieve complete Bézier extraction of T-spline solid. On this basis, isogeometric analysis of T-spline solids is realized.

However, some complex T-spline solids may have extraordinary nodes (the nodes that have valence more than 6 in 3 dimensions, resulting in a much more complicated parametrization scheme). The above Bézier segmentation algorithm is incompatible with T-spline solids with such extraordinary nodes. As a difficult problem in the study of T-splines, the extraction of Bézier for extraordinary nodes in T-spline solid should be further explored. The examples shown here are also limited occasions because only zero genus solids are tested, which is only a small class of objects in mechanical engineering. In theory the main algorithms in this paper can be used for solids of arbitrary genus but such models need more complex parametrization method and it is still an ongoing job in our lab so no such examples can be demonstrated now. Once entitled with the ability to handle extraordinary nodes as well as arbitrary genus models, the isogeometric analysis based on trivariate T-spline models may fully unfold its potentials for integrated CAD/CAE scenarios, and this will be the work we devote the next few years to.

ACKNOWLEDGMENT

The work is financially supported by the Natural Science Foundation of China (Project Nos. 61572056).

ORCID

Mayi Guo, https://orcid.org/0000-0001-9380-4910

REFERENCES

[1] Aigner, M.; Heinrich, C.; Juttler, B.; Pilgerstorfer, E.; Simeon, B.; Vuong, A. V.: Swept volume parameterization for isogeometric analysis, IMA International conference on mathematics of surfaces XIII, 2009, 19–44. https://doi.org/10.1007/978-3-642-03596-8_2

[2] Akkerman, I.; Bazilevs, Y.; Calo, V. M.: The role of continuity in residual-based variational multiscale modeling of turbulence, Computational Mechanics, 41(3), 2008, 371-378. https://doi.org/10.1007/s00466-007-0193-7

[3] Bazilevs, Y.; Calo, V. M.; Cottrell, J. A.: Variational multiscale residual-based turbulence modeling for large eddy simulation of in compressible flows, Computer Methods in Applied Mechanics and Engineering, 197(1), 2007, 173-201. https://doi.org/10.1016/j.cma.2007.07.016

[4] Bazilevs, Y.; Calo, V. M.; Cottrell, J. A.; Evans, J. A.; Hughes, T. J. R.; Lipton. S.; Scott. M. A.; Sederberg, T. W.: Isogeometric analysis using T-splines, Computer Methods in Applied Mechanics and Engineering, 199(5-8), 2010, 229–263. https://doi.org/10.1007/978-3-642-59223-2

[5] Bazilevs, Y.; Calo, V. M.; Zhang, Y.: Isogeometric fluid–structure interaction analysis with applications to arterial blood flow, Computational Mechanics, 38(4-5), 2006, 310-322. https://doi.org/10.1007/s00466-006-0084-3

[6] Borden, M. J.; Scott, M. A.; Evans, J. A.; Hughes, T. J. R.: Isogeometric finite element data structures based on Bézier extraction of NURBS, International Journal for Numerical Methods in Engineering, 87, 2011, 15–36. https://doi.org/10.1002/nme.2968

[7] Hughes, T. J. R.; Cottrell, J. A.; Bazilevs, Y.: Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, Computer Methods in Applied Mechanics and Engineering, 194(39), 2005, 4135-4195. https://doi.org/10.1016/j.cma.2004.10.008

[8] Luke, Engvall.; John, A. E.: Isogeometric triangular Bernstein–Bézier discretizations: Automatic mesh generation and geometrically exact finite element analysis, Computer Methods in Applied Mechanics and Engineering, 304, 2016, 378–407. https://doi.org/10.1016/j.cma.2016.02.012

[9] Piegl, LA.; Tiller, W.: The NURBS Book (monographs in visual communication), 2nd ed. Springer, New York,1997.

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

[11] Sederberg, T. W.; Cardon, D. L.; Finnigan, G. T.: T-spline simplification and local refinement, ACM Transactions on Graphics, 23(3), 2004, 276-283. https://doi.org/10.1145/1015706.1015715

[12] Sederberg, T. W.; Zheng, J.; Bakenov, A.: T-splines and T-NURCCs, ACM Transactions on Graphics, 22(3), 2003, 477-484. https://doi.org/10.1145/882262.882295

[13] Wang, W.; Zhang, Y.; Hughes, T. J. R.: Trivariate solid t-spline construction from boundary triangulations with arbitrary genus topology, Computer-Aided Design, 45, 2013, 351–360. https://doi.org/10.1016/j.cad.2012.10.018

[14] Xu, G.; Mourrain, B.; Duvigneau, R.; Galligo, A.: Analysissuitable volume parameterization of multi-block computational domain in isogeometric applications, Computer-Aided Design 45(2),2013,395–404. https://doi.org/10.1016/j.cad.2012.10.022

[15] Zhang, Y.; Bazilevs, Y.; Goswami, S.; Bajaj, CL.; Hughes, T. J. R.: Patient-specific vascular NURBS modeling for isogeometric analysis of blood flow, Computer Methods in Applied Mechanics and Engineering 196(29-30), 2007, 2943–2959. https://doi.org/10.1016/j.cma.2007.02.009

[16] Zhang, Y.; Wang, W.; Hughes, T. J. R.: Conformal solid T-spline construction from boundary T-spline representations, Computational Mechanics, 51(6), 2013, 1051–1059. https://doi.org/10.1007/s00466-012-0787-6

[17] Zhang, Y.; Wang, W.; Hughes, T. J. R.: Solid T-spline construction from boundary representations for genus-zero geometry, Computer Methods in Applied Mechanics and Engineering, 249-252, 2012, 185-197.https://doi.org/10.1016/j.cma.2012.01.014