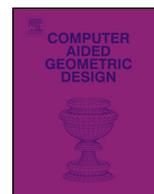




ELSEVIER

Contents lists available at ScienceDirect

Computer Aided Geometric Design

www.elsevier.com/locate/cagd


Modified T-splines [☆]


 Hongmei Kang, Falai Chen ^{*}, Jiansong Deng

School of Mathematical Sciences, University of Science and Technology of China, Hefei 230026, PR China

ARTICLE INFO

Article history:

Received 6 January 2013

Received in revised form 13 June 2013

Accepted 30 September 2013

Available online 17 October 2013

Keywords:

T-mesh

T-spline

Knot deletion

Surface fitting

Surface simplification

ABSTRACT

T-splines are a generalization of NURBS surfaces, the control meshes of which allow T-junctions. T-splines can significantly reduce the number of superfluous control points in NURBS surfaces, and provide valuable operations such as local refinement and merging of several B-splines surfaces in a consistent framework. In this paper, we propose a variant of T-splines called Modified T-splines. The basic idea is to construct a set of basis functions for a given T-mesh that have the following nice properties: non-negativity, linear independence, partition of unity and compact support. Due to the good properties of the basis functions, the Modified T-splines are favorable both in adaptive geometric modeling and isogeometric analysis.

© 2013 Elsevier B.V. All rights reserved.

1. Introduction

T-splines were introduced by Sederberg et al. (2003, 2004) and have been studied extensively in the last ten years. T-splines are a generalization of NURBS surfaces, the control meshes of which permit T-junctions. Unlike NURBS, T-junctions allow T-splines to be locally refinable without propagating entire columns or rows. This property makes T-splines an ideal technology for removing superfluous control points in NURBS surfaces and for adaptive isogeometric analysis (Hughes et al., 2005; Cottrell et al., 2009). Initial investigations using T-splines as a basis for isogeometric analysis demonstrate that T-splines possess similar convergence properties as NURBS with far fewer degrees of freedom (Dörfler et al., 2009; Bazilevs et al., 2010). However, the blending functions of T-splines are not always linearly independent. Buffa et al. (2010) gave an example of a T-spline with linearly dependent blending functions. This causes concerns about the linear independence of T-splines. A solution to this problem is the so-called *analysis-suitable T-splines* (AST-splines for short) (Li et al., 2012; Scott et al., 2012). AST-splines are a subset of T-splines defined over a restricted T-mesh whose T-junction extensions do not intersect, and the blending functions are always linearly independent and thus are suitable for isogeometric analysis. However, the topology of the meshes of AST-splines is relatively restrict. For example, the mesh for common local refinement in isogeometric analysis as shown in Fig. 1 is not an AST-mesh. Algorithm exists for modifying a non-AST-mesh into an AST-mesh (Scott et al., 2012).

In Deng et al. (2008), the authors introduced the concept of splines over T-meshes, and specifically PHT-splines were proposed. PHT-splines are polynomial splines defined over a hierarchical T-mesh, and the basis functions of PHT-splines are linearly independent, form a partition of unity and have compact supports. The local refinement algorithm of PHT-splines is local and very simple. Furthermore, since a PHT-spline is a polynomial (instead of a piecewise polynomial) over each cell of the T-mesh, it holds a good approximation property. These properties make PHT-splines an ideal tool for isogeometric

[☆] This paper has been recommended for acceptance by B. Jüttler.

^{*} Corresponding author.

 E-mail address: chenfl@ustc.edu.cn (F. Chen).

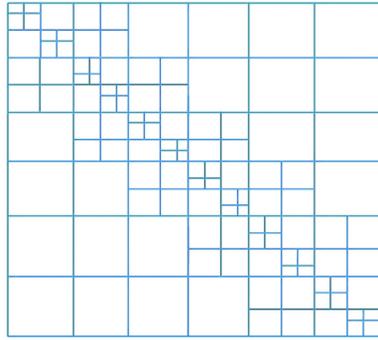


Fig. 1. A non-AST-mesh.

analysis (Nguyen-Thanh et al., 2011a, 2011b). However, PHT-splines are only C^1 continuous, which is a disadvantage for geometric modeling.

Another type of local refinement splines is LR-splines introduced by Dokken et al. (2013). LR-splines are defined on a μ -extended LR-mesh which is constructed by inserting line segments starting from a tensor product mesh according to certain rules. LR-spline also forms a non-negative partition of unity and spans the complete piecewise polynomial space on the mesh when the mesh construction follows certain rules. Different strategies can be employed to construct linearly independent LR B-splines by mesh modification. However, unlike T-splines and tensor product B-splines, there is no one-to-one correspondence between the 3D control mesh and the LR B-spline functions.

Hierarchical B-splines were introduced by Forsey and Bartels (1988), which have been recently further elaborated by Giannelli et al. (2012). The idea is to suitably truncate hierarchical B-spline functions according to finer levels in the hierarchy, which are called THB-splines. A THB-spline is a linear combination of B-splines, and THB-splines form a partition of unity and are linearly independent.

In this paper, we introduce a new type of local refinement splines called Modified T-splines, which inherit some good properties of the above splines while preventing some undesirable properties. The intuitive idea is to construct a set of basis functions which have good properties, such as non-negativity, partition of unity, linear independence and compact support. With the help of an auxiliary T-mesh T' (defined in Lemma 3.2), the basis functions are constructed as a linear combination of T-splines defined over T' . In this sense, Modified T-splines have some similarity with THB-splines.

The remainder of the current paper is organized as follows. In Section 2, we recall some preliminary knowledge about knot deletion of B-splines and T-splines. In Section 3, the construction details of Modified T-splines are described, and some properties, especially approximation property of Modified T-splines are discussed. Section 4 demonstrates some applications of Modified T-splines in surface fitting. Section 5 concludes the paper with a summary and future work.

2. Preliminaries

In this section, we recall some preliminary knowledge about knot deletion of B-splines in one dimension which is useful in the construction of Modified T-splines. Then the basic concepts about T-meshes and T-splines are reviewed.

2.1. Knot deletion of univariate B-splines

For simplicity, we only consider degree one and degree three B-splines for illustration.

Given a knot vector $\mathbf{t} = [t_0, t_1, \dots, t_n]$ with $t_0 \leq t_1 \leq \dots \leq t_n$, the associated B-spline basis functions are defined recursively as follows

$$N_i^0(t) = \begin{cases} 1, & t \in [t_i, t_{i+1}), \\ 0, & \text{otherwise,} \end{cases} \quad (1)$$

$$N_i^k(t) = \frac{t - t_i}{t_{i+k} - t_i} N_i^{k-1}(t) + \frac{t_{i+k+1} - t}{t_{i+k+1} - t_{i+1}} N_{i+1}^{k-1}(t), \quad k \geq 1. \quad (2)$$

We use $N^1[t_{i-1}, t_i, t_{i+1}](t)$ to denote the degree one B-spline basis function $N_{i-1}^1(t)$ and call it the linear B-spline basis function at knot t_i . Similarly, for cubic B-spline functions, $N^3[t_{i-2}, t_{i-1}, t_i, t_{i+1}, t_{i+2}](t) = N_{i-2}^3(t)$ is called the cubic B-spline basis function at knot t_i .

Now insert a knot \hat{t} into the knot vector \mathbf{t} to get a new knot vector $\tilde{\mathbf{t}}$. Without loss of generality, we assume $\hat{t} \in [t_3, t_4)$. Then the B-spline basis functions associated with \mathbf{t} either can be written as a linear combination of the B-spline basis functions associated with $\tilde{\mathbf{t}}$ or remain unchanged. The relationship between these basis functions can be written as

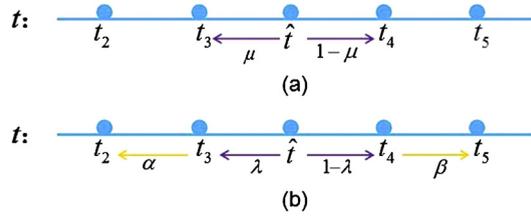


Fig. 2. Distribution process of univariate B-splines for knot deletion. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

$$\begin{pmatrix} N^1[t_2, t_3, t_4](t) \\ N^1[t_3, t_4, t_5](t) \end{pmatrix} = \begin{pmatrix} 1 & \frac{t_4 - \hat{t}}{t_4 - t_3} & 0 \\ 0 & \frac{\hat{t} - t_3}{t_4 - t_3} & 1 \end{pmatrix} \begin{pmatrix} N^1[t_2, t_3, \hat{t}](t) \\ N^1[t_3, \hat{t}, t_4](t) \\ N^1[\hat{t}, t_4, t_5](t) \end{pmatrix}, \tag{3}$$

$$\begin{pmatrix} N^3[t_0, t_1, t_2, t_3, t_4](t) \\ N^3[t_1, t_2, t_3, t_4, t_5](t) \\ N^3[t_2, t_3, t_4, t_5, t_6](t) \\ N^3[t_3, t_4, t_5, t_6, t_7](t) \end{pmatrix} = \begin{pmatrix} 1 & \frac{t_4 - \hat{t}}{t_4 - t_1} & 0 & 0 & 0 \\ 0 & \frac{\hat{t} - t_1}{t_4 - t_1} & \frac{t_5 - \hat{t}}{t_5 - t_2} & 0 & 0 \\ 0 & 0 & \frac{\hat{t} - t_2}{t_5 - t_2} & \frac{t_6 - \hat{t}}{t_6 - t_3} & 0 \\ 0 & 0 & 0 & \frac{\hat{t} - t_3}{t_6 - t_3} & 1 \end{pmatrix} \begin{pmatrix} N^3[t_0, t_1, t_2, t_3, \hat{t}](t) \\ N^3[t_1, t_2, t_3, \hat{t}, t_4](t) \\ N^3[t_2, t_3, \hat{t}, t_4, t_5](t) \\ N^3[t_3, \hat{t}, t_4, t_5, t_6](t) \\ N^3[\hat{t}, t_4, t_5, t_6, t_7](t) \end{pmatrix}. \tag{4}$$

Alternatively, the knot vector \mathbf{t} can be considered as deleting a knot \hat{t} from the knot vector $\tilde{\mathbf{t}}$. After knot deletion, the B-spline basis function at \hat{t} is distributed to the B-spline basis functions at the neighboring knots of \hat{t} . The distribution process is demonstrated in Fig. 2. Fig. 2(a) illustrates the distribution process of degree one B-splines for knot deletion. The B-spline basis function at knot \hat{t} is distributed to the B-spline basis functions at the knots t_3 and t_4 with a ratio of μ and $1 - \mu$ respectively, where $\mu = \frac{t_4 - \hat{t}}{t_4 - t_3}$. Fig. 2(b) shows the distribution process of cubic B-splines. There are two levels of distribution in this case. The distribution indicated by the orange arrows is firstly carried out. Then the distribution indicated by the purple arrows is performed, where the distribution coefficients are $\alpha = \frac{t_4 - \hat{t}}{t_4 - t_1}$, $\beta = \frac{\hat{t} - t_3}{t_6 - t_3}$ and $\lambda = \frac{t_5 - \hat{t}}{t_5 - t_2}$.

Note that the coefficient matrices in formulas (3) and (4) have the following nice properties: the column sums of each matrix are equal to one, the elements of each matrix are non-negative and each matrix has full rank. These properties induce the following important result.

Lemma 2.1. Let $\{N_i\}_{i=1}^m$ be a set of non-negative and linearly independent B-spline functions which are assumed to form a partition of unity in the defined domain D , $D \subseteq \mathbb{R}^d$. Let $\{B_j\}_{j=1}^n$ be a set of functions, each of which is a linear combination of $\{N_i\}_{i=1}^m$, or in matrix form

$$\begin{pmatrix} B_1 \\ \vdots \\ B_n \end{pmatrix} = M \begin{pmatrix} N_1 \\ \vdots \\ N_m \end{pmatrix}$$

where M is an $n \times m$ matrix with $n < m$. If the matrix M has full rank, then $\{B_j\}_{j=1}^n$ are linearly independent. Furthermore, if the column sums of M are all equal to one and the elements of M are all non-negative, then $\{B_j\}_{j=1}^n$ are non-negative and form a partition of unity over D .

Proof. Suppose there exist constants $\{c_i\}_{i=1}^n$ such that

$$\sum_{i=1}^n c_i B_i = (c_1, \dots, c_n)(B_1, \dots, B_n)^T \equiv \mathbf{0}.$$

Since $(B_1, \dots, B_n)^T = M(N_1, \dots, N_m)^T$, then

$$(c_1, \dots, c_n)M(N_1, \dots, N_m)^T = (d_1, \dots, d_m)(N_1, \dots, N_m)^T \equiv \mathbf{0},$$

where $(d_1, \dots, d_m) = (c_1, \dots, c_n)M$. Because of the linear independence of $\{N_i\}_{i=1}^m$, we have $d_i = 0$, $i = 1, 2, \dots, m$. Therefore $c_i = 0$, $i = 1, 2, \dots, n$, follows as the matrix M has full rank.

For the next part, notice that

$$\sum_{i=1}^n B_i = (1, \dots, 1)(B_1, \dots, B_n)^T = (1, \dots, 1)M(N_1, \dots, N_m)^T.$$

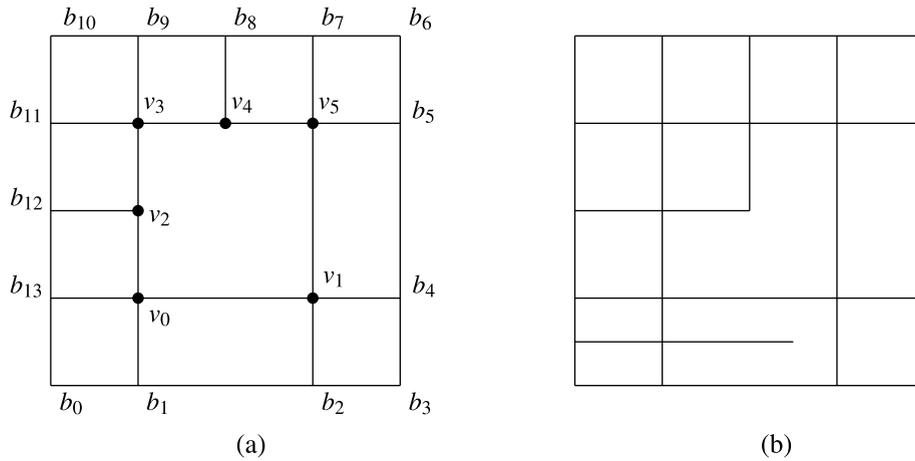


Fig. 3. Examples of T-mesh and non-T-mesh in a plane.

Since the column sums of the matrix M are all one, $(1, \dots, 1)M = (1, \dots, 1)$. Therefore

$$\sum_{i=1}^n B_i = (1, \dots, 1)(N_1, \dots, N_m)^T = \sum_{i=1}^m N_i \equiv 1.$$

The non-negativity of $\{B_j\}_{j=1}^n$ is obviously true. The lemma is thus proved. \square

2.2. T-mesh

A T-mesh is a rectangular grid that allows T-junctions. It is assumed that the end points of each grid line in the T-mesh must be on two other grid lines and each cell or facet in the grid must be a rectangle. If a T-junction on one edge of a face can be connected to a T-junction on an opposing edge of the face thereby splitting the face into two rectangles, that edge must be included in the T-mesh. Fig. 3(a) illustrates a T-mesh while the mesh in Fig. 3(b) is not a T-mesh. A T-mesh can lie either in a 2D plane or in a 3D space. Generally, a 2D T-mesh is served as the parametric space of T-splines while a 3D T-mesh is served as the control grid of T-splines. For a 3D T-mesh, each facet doesn't have to be a rectangle or planar.

A grid point in a T-mesh is called a vertex of the T-mesh. If the vertex is on the boundary grid line, then it is called a boundary vertex, otherwise it is called an interior vertex. For example, $b_0, b_1, \dots, b_{12}, b_{13}$ in Fig. 3(a) are boundary vertices, while $v_0, v_1, \dots, v_4, v_5$ are interior vertices. Interior vertices have two types: one is crossing vertices, for example, v_0, v_1, v_3, v_5 in Fig. 3(a); and the other one is T-vertices, for example, v_2, v_4 in Fig. 3(a). T-vertices are classified into two classes: horizontal T-vertices, for example, v_2 in Fig. 3(a) and vertical T-vertices, for example, v_4 in Fig. 3(a).

A T-vertex can be extended by inserting edges starting from this T-vertex, in the direction of a missing edge. A T-vertex can be extended as far as to the mesh boundary. All the edges created by such extension are called T-vertex extensions. Fig. 4 shows an example where dotted purple line segments are T-vertex extensions. For a given T-mesh T , let T' be the new mesh produced by some T-vertex extensions. Those vertices produced by the T-vertex extensions are called new vertices relative to the mesh T , that is, new vertices are the vertices that are in T' but not in the origin mesh T . The vertices in T are called old vertices. There are three types of new vertices: one is H-type, if the new vertex is produced by the extension of horizontal T-vertex; one is V-type, if the new vertex is produced by the extension of vertical T-vertex; the last one is C-type, if the new vertex is produced by both extensions of horizontal T-vertex and vertical T-vertex. As illustrated in Fig. 4, A is H-type produced by v_1 , B is V-type produced by v_4 and C is C-type produced by v_2 and v_3 .

Alternatively, one can delete vertices from a T-mesh to create a new T-mesh. When a vertex is deleted, we mean the edges connecting the vertex in a specific direction are also deleted. This will be made clear in later context.

2.3. T-splines

Given a control T-mesh in 3D space with control points $\mathbf{P}_i, i = 1, 2, \dots, n$, a T-spline is defined as

$$\mathbf{T}(s, t) = \frac{\sum_{i=1}^n \mathbf{P}_i T_i(s, t)}{\sum_{i=1}^n T_i(s, t)}, \quad (s, t) \in \Omega, \tag{5}$$

where $T_i(s, t)$ is the blending function

$$T_i(s, t) = N_i^3(s)N_i^3(t) \tag{6}$$

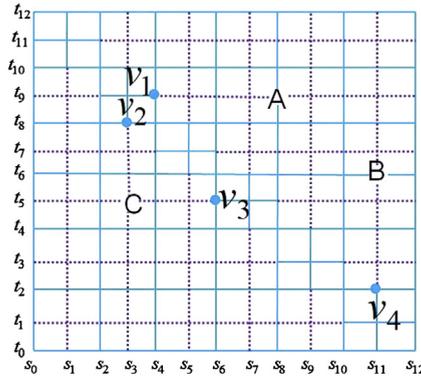


Fig. 4. T-vertex extensions, A is H-type, B is V-type and C is C-type. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

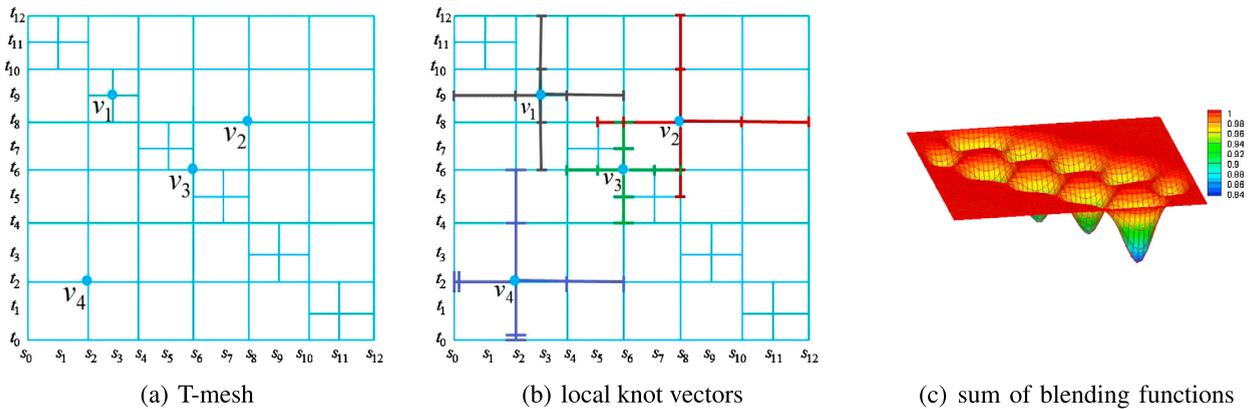


Fig. 5. Knot vectors for blending functions.

with $N_i^3(s)$, $N_i^3(t)$ being the cubic B-spline basis functions associated with two local knot vectors \mathbf{s}_i and \mathbf{t}_i respectively. The two local knot vectors \mathbf{s}_i and \mathbf{t}_i come from two global knot vectors $\mathbf{s} = [s_{-2}, s_{-1}, s_0, \dots, s_{p+2}]$ and $\mathbf{t} = [t_{-2}, t_{-1}, t_0, \dots, t_{q+2}]$ which are inferred from the control T-mesh. Let T be the preimage of the control mesh, which is a T-mesh in 2D parametric space. Ω is the 2D rectangular domain that is occupied by T .

To each control point \mathbf{P}_i corresponds a blending function $T_i(s, t) = N_i^3(s)N_i^3(t)$ defined in terms of knot vectors $\mathbf{s}_i = [s_{i_0}, s_{i_1}, s_{i_2}, s_{i_3}, s_{i_4}]$ and $\mathbf{t}_i = [t_{i_0}, t_{i_1}, t_{i_2}, t_{i_3}, t_{i_4}]$. The local knot vectors \mathbf{s}_i and \mathbf{t}_i are inferred as follows. Assume that the parameter coordinates of the preimage v_i of \mathbf{P}_i are (s_{i_2}, t_{i_2}) . The knots s_{i_3} and s_{i_4} are found by considering a ray in parameter space $R(\alpha) = (s_{i_2} + \alpha, t_{i_2})$. Then s_{i_3} and s_{i_4} are the s coordinates of the first two vertical edges (which are called s -edges) intersected by the ray (not including the initial (s_{i_2}, t_{i_2})). The other knots in \mathbf{s}_i and \mathbf{t}_i are found in like manner. For the details, the reader is referred to Sederberg et al. (2003).

Figs. 5(a) and (b) illustrate a T-mesh in (s, t) parameter space. For the vertex v_1 , the associated knot vectors are $\mathbf{s}_1 = [s_0, s_2, s_3, s_4, s_6]$ and $\mathbf{t}_1 = [t_6, t_8, t_9, t_{10}, t_{12}]$; likewise for v_2 , $\mathbf{s}_2 = [s_5, s_6, s_8, s_{10}, s_{12}]$ and $\mathbf{t}_2 = [t_5, t_6, t_8, t_{10}, t_{12}]$.

Specifically, when the control T-mesh is a tensor product mesh, T-splines reduce to tensor product B-splines.

Note that the blending functions of T-splines may be linearly dependent, and they generally do not form a partition of unity, i.e., $\sum_{i=1}^n T_i(s, t) \neq 1$ (see Fig. 5(c)). However, by imposing some conditions on the T-mesh (for example, AST-mesh), the blending functions of the corresponding T-splines form a partition of unity. Unfortunately, most common T-meshes in local refinement are not AST-meshes, and generally it is not a trivial task to modify an existing T-mesh to an AST-mesh.

3. Construction of modified T-splines

In this section, we are going to construct a new type of local refinement splines over T-meshes which inherit the major properties of current local refinement splines. We call such splines *Modified T-splines*. We start with an outline of the construction process. Then the detailed construction approach is described, and the properties of Modified T-splines are presented.

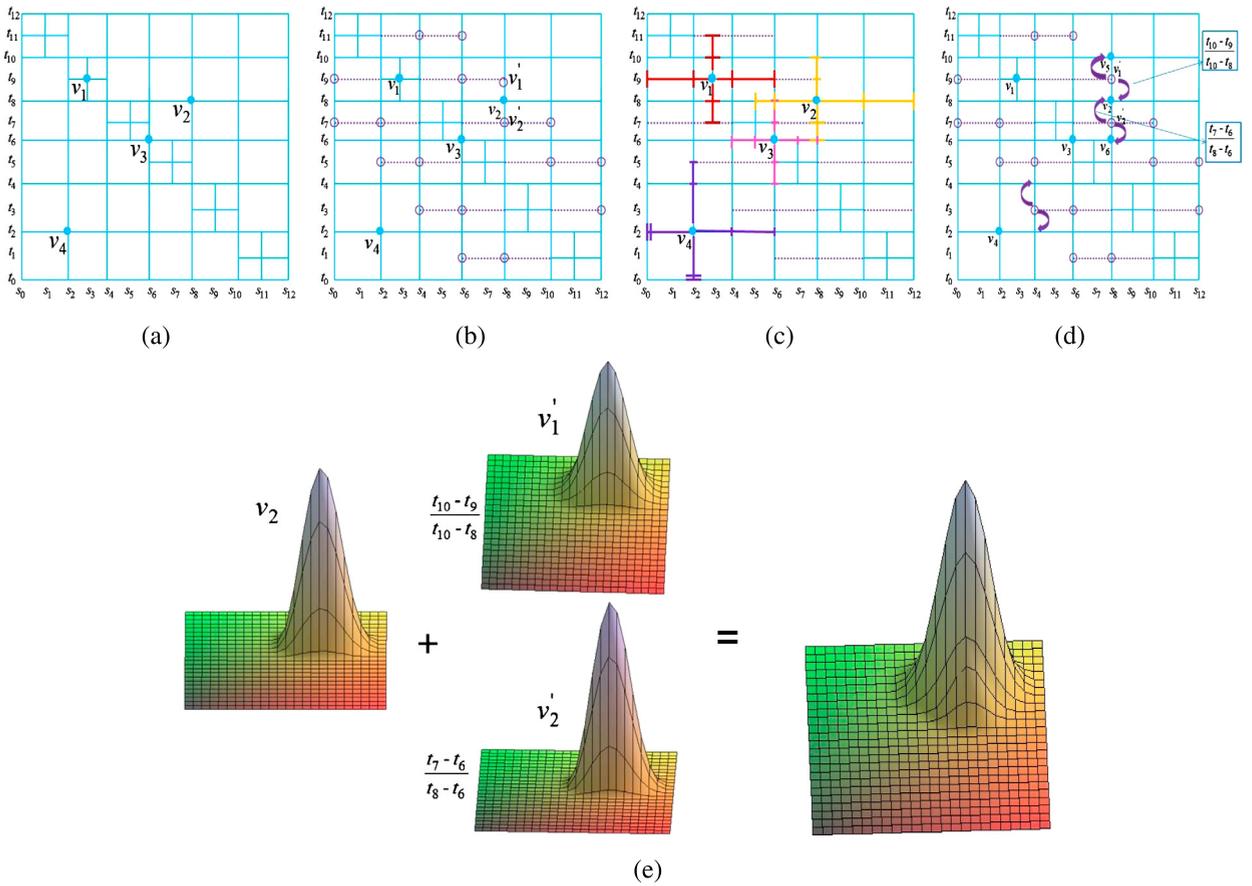


Fig. 6. Workflow of our algorithm. (a) Origin T-mesh T . (b) Extended T-mesh T' . Purple circles are new vertices. (c) Knot vectors of basis functions over T' . (d) Basis functions at new vertices are distributed. (e) Basis function at v_2 is constructed. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

3.1. Outline of the construction process

Given a T-mesh T in 2D parametric space, let $v_i, i = 1, 2, \dots, n$, be its vertices. For each vertex v_i , we are going to construct a piecewise bicubic basis function $B_i(s, t)$ over T with the following good properties:

- Non-negativity: $B_i(s, t) \geq 0, i = 1, 2, \dots, n$.
- Partition of unity: $\sum_{i=1}^n B_i(s, t) \equiv 1$.
- Linear independence: $B_i(s, t), i = 1, 2, \dots, n$, are linearly independent.
- Compact support: $B_i(s, t)$ has a compact support.

Then a Modified T-spline is defined

$$\mathbf{P}(s, t) = \sum_{i=1}^n \mathbf{P}_i B_i(s, t), \quad (s, t) \in \Omega, \tag{7}$$

where $\mathbf{P}_i, i = 1, 2, \dots, n$, are the control points which form a 3D control T-mesh. Ω is the 2D parametric domain that is occupied by T . Note that there is a one-to-one correspondence between the 3D control mesh and the mesh T in 2D parametric space.

The construction of the basis functions $B_i(s, t)$ consists of the following two major steps (see Fig. 6).

For the first step, we extend T to a new mesh T' by T-vertex extensions such that a set of basis functions $N_i(s, t)$ over T' with the above good properties can be easily constructed. The new mesh T' is called the *extended T-mesh* of T . For example, we can extend all the T-vertices of T to the mesh boundaries to form a tensor product mesh T' . Then we choose $N_i(s, t)$ to be the tensor product B-spline basis functions which of course satisfy the above properties. Fig. 6(b) demonstrates another choice of T' which is a T-mesh, and the corresponding basis functions are defined over T' as shown in Fig. 6(c).

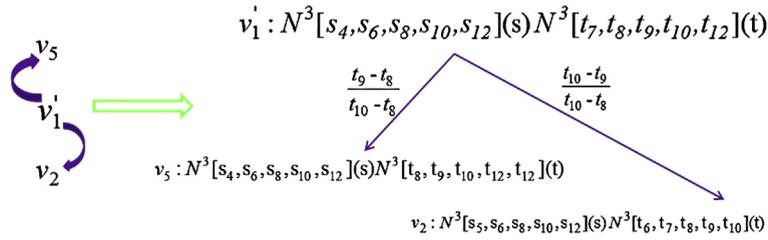


Fig. 7. Basis function at v'_1 is distributed to basis functions at its two receiving vertices v_2 and v_5 .

For the second step, let $\{v'_i\}_{i=1}^l$ be the new vertices generated by the T-vertex extensions. We distribute the basis function $N_i(s, t)$ at v'_i to the basis functions at the neighboring old vertices of T' . In Fig. 6(d), the basis function at v'_1 is distributed to the basis functions at v_2 and v_5 , and the basis function at v'_2 is distributed to the basis functions at v_2 and v_6 . Finally, the basis function $B_i(s, t)$ over T is constructed as a linear combination of the basis functions $N_i(s, t)$ at the neighboring vertices of v_i . Fig. 6(e) illustrates the basis function at v_2 .

The distribution of a basis function at a new vertex is composed of two steps: finding receiving vertices in T and computing distribution coefficients. Each new vertex is one of the three types: H-type, V-type and C-type. For H-type and V-type new vertices, the rule of distribution is simple: we distribute the basis function along one direction and perform the distribution as knot deletion in one dimension. However, for C-type new vertices, the situation is much harder. In our construction, we will avoid such situation.

For example, as illustrated in Fig. 6(d), v'_1 is a H-type vertex, the basis function at v'_1 is distributed along t (vertical) direction, and the distribution coefficients are computed by knot deletion in t direction as shown in formula (3). That is v_2 and v_5 are the receiving vertices of v'_1 . The distribution coefficient of v'_1 to v_2 is the ratio of distance $v'_1 v_5$ over $v_2 v_5$ ($= \frac{t_{10}-t_9}{t_{10}-t_8}$), and the one of v'_1 to v_5 is the ratio of distance $v'_1 v_2$ over $v_5 v_2$ ($= \frac{t_9-t_8}{t_{10}-t_8}$). Fig. 7 illustrates the distribution process of v'_1 by formula (3).

3.2. Construction method

In this section, we present details to construct bicubic Modified T-splines.

Given an original T-mesh T , we extend all the horizontal (or vertical) T-vertices to the boundary of T to obtain a new T-mesh. Now the new T-mesh has only vertical (or horizontal) T-vertices. For such T-meshes, we have the following important observation.

Lemma 3.1. *If there are only vertical (horizontal) T-vertices in a mesh T , then the blending functions of the T-splines defined over T are linearly independent and form a partition of unity.*

Proof. Without loss of generality, we assume T has only vertical T-vertices. Let T' be the tensor product mesh by extending all the T-vertices in T to the boundary of T . Because there are only vertical T-vertices in T , all the new vertices in T' are V-type. Let $B_i(s, t) = N_i^3(s)N_i^3(t)$ be the tensor product B-spline basis function at each vertex v_i of T' , $i = 1, 2, \dots, m$. Let $T_i(s, t)$ be the T-spline blending function at the vertex of T , $i = 1, 2, \dots, n$. We are going to show that: (1) $T_i(s, t)$ is a linear combination of $B_i(s, t)$, that is, there exists a matrix $M_{n \times m}$ such that

$$\begin{pmatrix} T_1 \\ \vdots \\ T_n \end{pmatrix} = M \begin{pmatrix} B_1 \\ \vdots \\ B_m \end{pmatrix}. \tag{8}$$

(2) the matrix M has full rank and the column sums of the matrix M are all one. Then by Lemma 2.1, the assertion of the lemma follows.

Assume there are p horizontal grid lines and q vertical grid lines in T' . For each horizontal grid line $t = t_i$, there are q vertices on the grid line, among which we assume r_i vertices ($r_i \leq q$) are old vertices. Since the T-splines at all the vertices on the grid line have the same local knot vector $\mathbf{t}_i = [t_{i-2}, t_{i-1}, t_i, t_{i+1}, t_{i+2}]$ in t direction, by the knot insertion algorithm of univariate cubic B-splines, the T-spline blending function $T_{ij}(s, t)$ at each old vertex on the grid line is a linear combination of the blending functions $B_{ij}(s, t)$ at all the vertices on the grid line, that is,

$$\begin{pmatrix} T_{i1} \\ \vdots \\ T_{ir_i} \end{pmatrix} = M_i \begin{pmatrix} B_{i1} \\ \vdots \\ B_{iq} \end{pmatrix}$$

where the matrix M_i has full rank and the column sums of M_i are all one by the property of knot insertion. Now (8) follows with $M = \text{diag}(M_1, M_2, \dots, M_p)$. Obviously, M has full rank and the column sums of M are all one. \square

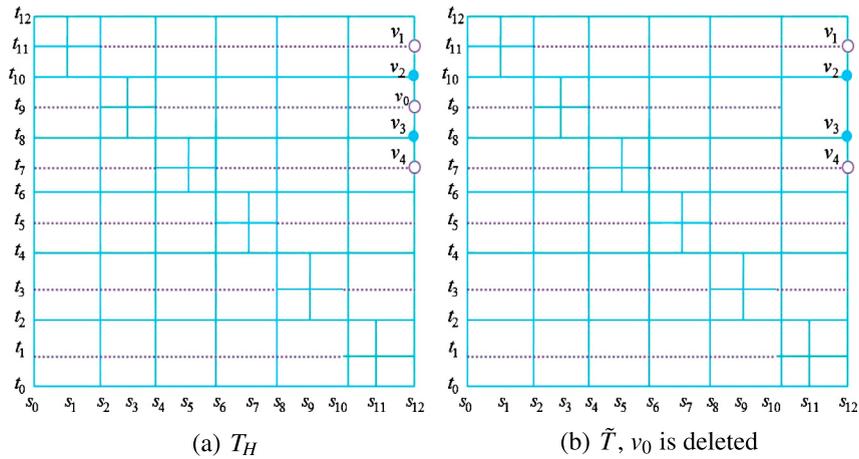


Fig. 8. T-mesh with only horizontal T-vertices.

For a given T-mesh T , let T_H be the mesh by extending all the horizontal T-vertices of T to the mesh boundary. T_H is called the *horizontal extension mesh* of T . Similarly, let T_V be the mesh by extending all the vertical T-vertices of T to the mesh boundary, and T_V is called the *vertical extension mesh* of T . Obviously T_H and T_V satisfy the condition of Lemma 3.1. If T_H is taken as the extended T-mesh of T , then there are only H-type new vertices which are needed to be distributed. Similarly, if we take T_V to be the extended T-mesh, then only V-type new vertices need to be distributed.

Before outlining the detailed algorithm to construct Modified T-splines, we can make some further simplification for extended T-meshes. In fact, among all the new vertices in T_H or T_V , many can be deleted without distribution process. We call such vertices *deletable vertices*. The key observation is as follows.

Assume v_0 is a H-type vertex on a vertical grid line $s = s_i$ in T_H , and its 2-neighboring vertices on the grid line (two above v_0 and two beneath v_0) are denoted as v_1, v_2, v_3, v_4 (see Fig. 8(a) as a reference). Now suppose that the T-spline blending functions at $v_i, i = 0, 1, \dots, 4$, possess the same s local knot vector \mathbf{s}_i . When v_0 (and the corresponding edge) is deleted from T_H , the blending function at v_0 is distributed to the blending functions at the four neighboring vertices $v_i, i = 1, 2, 3, 4$, by knot deletion of univariate cubic B-splines, and thus a T-spline is obtained over a new T-mesh \tilde{T} after deleting v_0 (and the corresponding edge) from T_H , see Fig. 8(b). It is easy to see that the blending functions of the T-spline over \tilde{T} are linearly independent and form a partition of unity. We conclude this in the following definition.

Definition 3.1. Let v_0 be a H-type new vertex on a vertical grid line in T , and let v_1, v_2, v_3, v_4 be the 2-neighboring vertices of v_0 on the grid line. If the T-spline blending functions at $v_i, i = 0, 1, \dots, 4$, have the same local s knot vector, then v_0 is called a h-deletable vertex. Similarly, one can define v-deletable vertex. Both h-deletable vertex and v-deletable vertex are called deletable vertex.

Similar to the proof of Lemma 3.1, we have

Lemma 3.2. Let T' be the T-mesh after deleting all the deletable vertices of T_H (or T_V). Then the T-spline blending functions over T' are linearly independent and form a partition of unity. T' is called the extended T-mesh of T .

Now we are ready to describe the details of the algorithm to construct Modified T-splines.

Input A T-mesh T in a 2D parametric plane.

Output A set of basis functions $B_i(s, t)$ over T .

Step 1 Extend all the horizontal T-vertices to the mesh boundary of T and denote the mesh by T_H .

Step 2 Find all the h-deletable vertices in T_H and delete them (and the corresponding edges), and the result T-mesh is denoted by T' .

Step 3 Construct T-spline blending functions $\{T_i(s, t)\}$ over T' .

Step 4 Scan each vertical grid line, and for each new vertex on the grid line, distribute the blending function at the new vertex to the blending functions at the neighboring old vertices on the grid line. This can be accomplished by knot deletion of univariate B-splines.

Step 5 Let v_i be a vertex in T . Assume $\{v_{ij}\}_{j=1}^l$ are all the new vertices whose blending functions are distributed to v_i , and $\{c_{ij}\}$ are the corresponding distribution coefficients, then the basis function at v_i is constructed as $B_i(s, t) = T_i(s, t) + \sum_{j=1}^l c_{ij} T_{ij}(s, t)$.

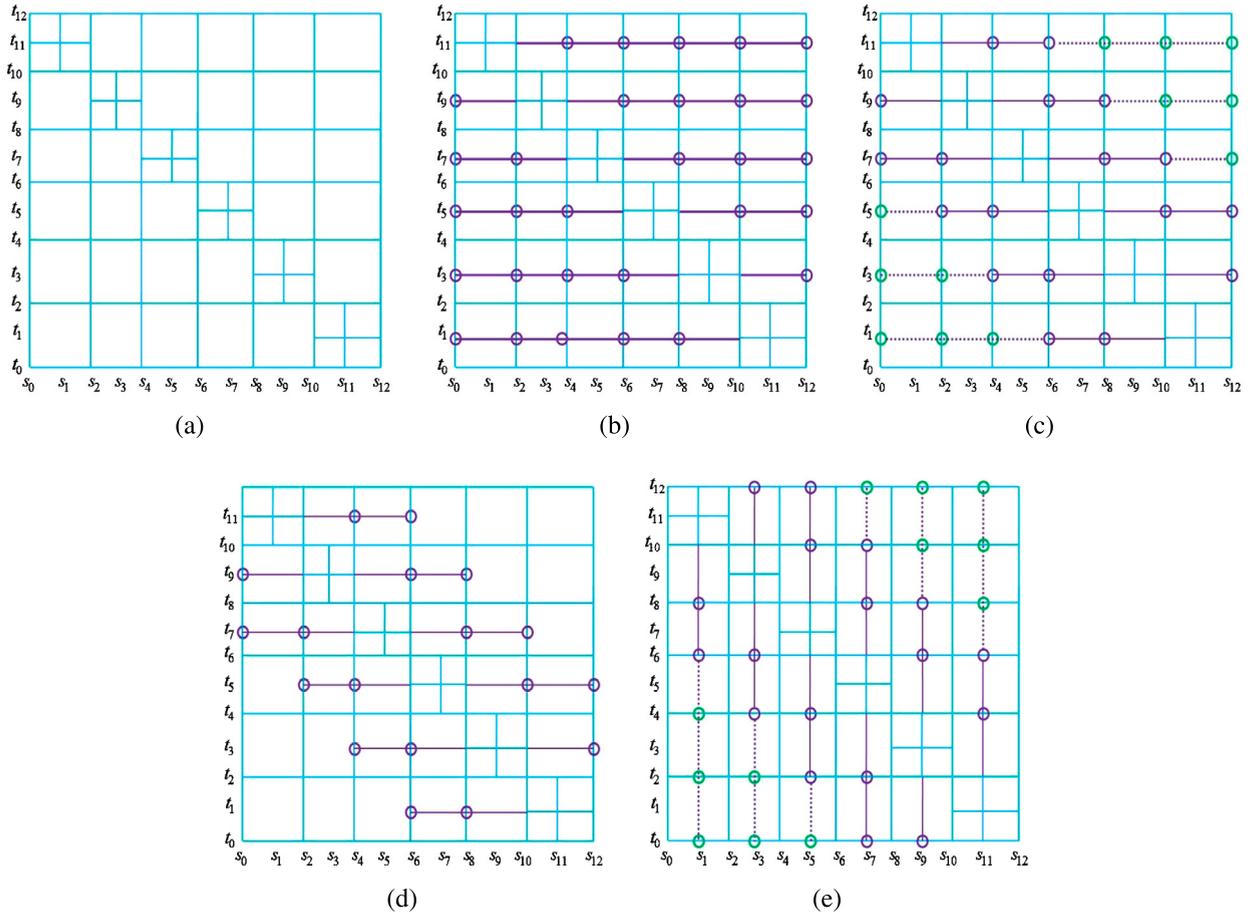


Fig. 9. Construction of the basis functions. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

Now we illustrate an example to explain the above algorithm. The origin T-mesh T is shown in Fig. 9(a), and T_H is shown in Fig. 9(b), where the purple circles are marked for H-type new vertices. According to Lemma 3.1, the blending functions of the T-spline defined over T_H are linearly independent and form a partition of unity. Scan every vertical grid line segments, find all h-deletable vertices, as marked by green circles in Fig. 9(c). Fig. 9(d) is the extended T-mesh T' after removing the h-deletable vertices from T_H . The blending functions at the remaining vertices (purple circles) are distributed as univariate knot deletion of B-spline to the neighboring old vertices.

We can also start with T_V . Fig. 9(e) is the T_V mesh, where purple circles are V-type new vertices needed to be distributed and green circles are v-deletable vertices.

Fig. 10 shows the comparison between the basis functions of Modified T-splines and T-splines, and how a basis function changes after knot deletion. From the examples, one can see that while the supports of Modified T-spline basis functions enlarge after knot deletion, they are comparable with the supports of T-spline basis functions.

We would like to make some remarks about the above construction method.

Remark 3.1.

1. Instead of using T_H , we can also use T_V as the starting mesh. The algorithm is similar, and one only has to change horizontal to vertical, and vertical to horizontal. In practice, we choose T_H or T_V based on which mesh would result in less number of vertices that need to be distributed.
2. In step 4, the blending functions at the new vertices are distributed to the blending functions at the old vertices by knot deletion of univariate cubic B-splines. However, it is favorable to simply distribute the blending functions at the new vertices by knot deletion of degree one B-splines, since knot deletion of linear B-splines makes the support of the basis functions smaller.
3. The extended T-mesh T' in our algorithm is an AST-mesh. The spline space spanned by Modified T-splines is a subspace of the spline space spanned by AST-splines over T' . The local refinement algorithm in Scott et al. (2012) can be applied

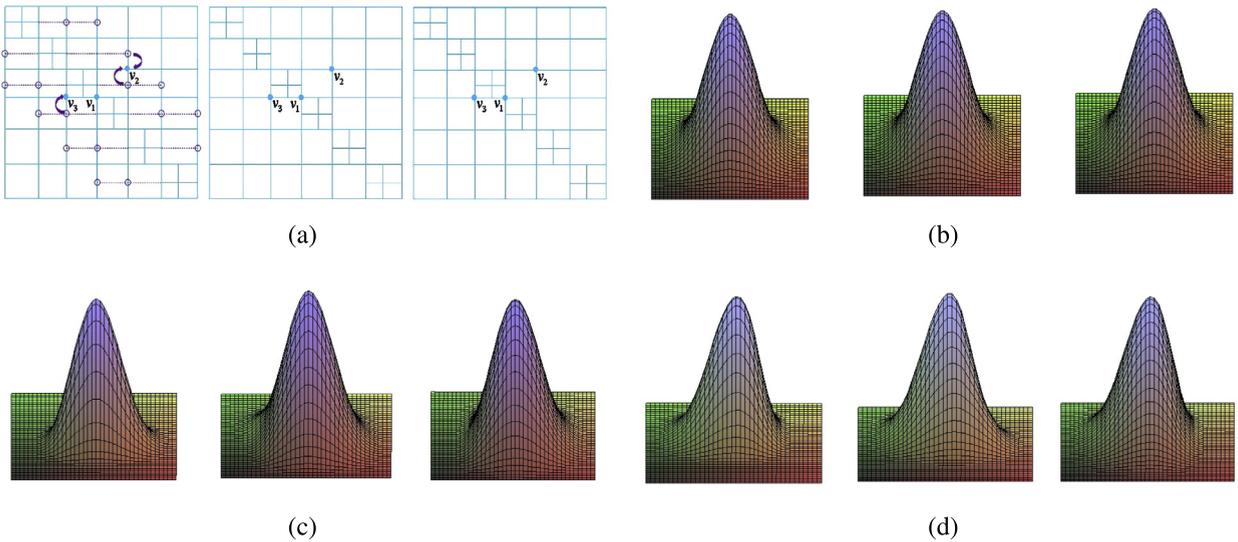


Fig. 10. (a) Left to right: the extended T-mesh T' , the original T-mesh T and the original T-mesh T . For (b), (c) and (d), left to right: basis functions of T-splines over T' , Modified T-splines and T-splines over T at three vertices v_1 , v_2 and v_3 .

to obtain an AST-mesh from a given T-mesh, so AST-mesh can be used as a starting point instead of T_H or T_V . But as stated previously, it may introduce C-type new vertices which is hard to handle.

- The construction algorithm involves three T-meshes: the origin T-mesh T and two auxiliary T-meshes: T_H and the extended T-mesh T' . T_H is served for constructing T' and T' is served for defining the T-splines, from which Modified T-splines are defined on T . At every vertex in T , a Modified T-spline basis function is defined as a linear combination of T-spline basis functions on T' , and thus the knot vectors and the distribution coefficients at the new vertices in T' are enough for further use. So it is not necessary to store the whole information of T' along with the original T-mesh. Both T_H and T' are discarded after the algorithm.

3.3. Algorithm analysis

While it is generally very hard to give a complete analysis of the algorithm complexity for constructing Modified T-splines, we present several typical examples of meshes to get a hint about the algorithm complexity.

The construction algorithm in Section 3.2 mainly includes two parts: extended T-mesh T' construction and non-deletable vertices distribution. Let $R = \frac{N}{M}$, where N is the total number of new vertices in T' , and M is the total number of old vertices in T' (i.e., M is the total number of vertices in T). The ratio R gives a rough idea about the computational complexity of the algorithm.

Fig. 11 shows four typical hierarchical T-meshes. Denote the maximum level of hierarchy by k and the refinement domain at level i by Ω_i . The refinement domain Ω_i in Fig. 11(a) is a rectangle with $\Omega_i \subset \Omega_{i-1}$ (assume that the boundaries of Ω_i and Ω_{i-1} do not overlap). There are $4k$ non-deletable vertices in T_H and at least $25 + 16k$ old vertices in T_H . So the ratio $R \leq \frac{4k}{25+16k} < 1/4$. The T-mesh in Fig. 11(b) is similar to the one in Fig. 11(a), where the refinement domains are polygons. In this case, there are $7k - 1$ non-deletable vertices in T_H and $30k + 36$ old vertices in T_H . So the ratio $R = \frac{7k-1}{30k+36} < \frac{7}{30}$.

For the mesh in Fig. 11(c), suppose that there are $2(n + 2)$ vertical T-vertices at level 1. Then there are $n_l = 2^l n + 4$ vertical T-vertices and $\sum_{j=1}^{0.5n_l} 6j + 3$ old vertices at level $l, l = 1, 2, \dots, k$. We assume that $k \geq 3$. A vertical T-vertex at level l ($1 \leq l \leq k - 2$) produces 13 non-deletable vertices at most, and produces 11 non-deletable vertices at most at level $k - 1$. At level k there are about half of vertical T-vertices, each of which produces one non-deletable vertex. So the total number of new vertices in T' is less than $\sum_{l=1}^{k-2} 13(2^l n + 4) + 11(2^{k-1} n + 4) + 0.5(2^k n + 4) \approx 12.5n \cdot 2^k + 52k$. Since there are about $\sum_{l=1}^k 6n_l + 3n_l^2 \approx n^2 \cdot 4^k + 18n \cdot 2^k + 24k$ old vertices, we get an estimate for the ratio $R \approx \frac{12.5n \cdot 2^k + 52k}{n^2 \cdot 4^k + 18n \cdot 2^k + 24k}$. It is easy to see that R decreases as k increases, and R tends to zero when k is large enough.

The final example illustrates the worst case – a T-mesh which refines along a diagonal as shown in Fig. 11(d), where the band width of the refinement domain Ω_i is 3. Suppose the starting mesh is an $n \times n$ TP-mesh and $k \geq 3$. Then there are $2^l n - 4$ vertical T-vertices and $11(2^{l-1} n - 2) + 16$ old vertices at level $l, l = 1, 2, \dots, k$. A vertical T-vertex at level l ($1 \leq l \leq k - 2$) produces 11 non-deletable vertices at most, and produces 8 non-deletable vertices at most at level $k - 1$. There are about half of vertical T-vertices at level k , each of which produces one non-deletable vertex. So the new vertices in T' is about $\sum_{l=1}^{k-2} 11(2^l n - 4) + 8(2^{k-1} n - 4) + 0.5(2^k n - 4) = 20n \cdot 2^{k-1} - 22n - 44k + 54$. On the other hand, there are about

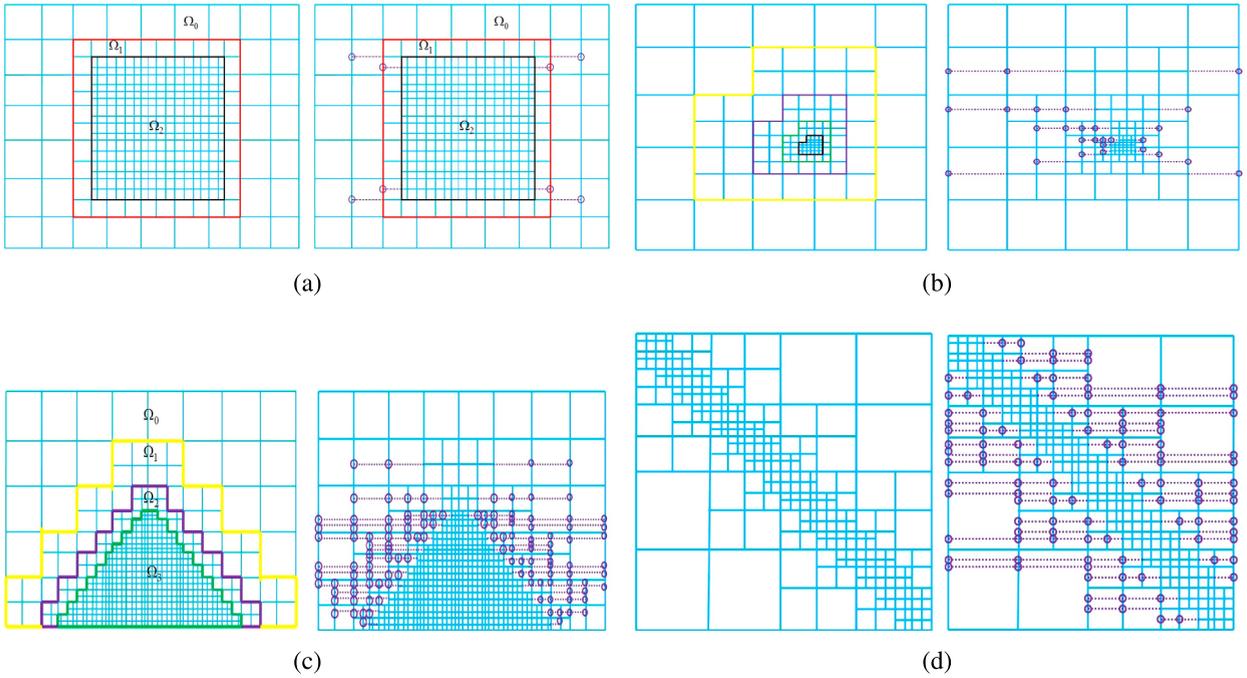


Fig. 11. Mesh T and the extended mesh T' with the ratio R, R' . (a) $R \approx 0.02, R' \approx 0.008$ with $k = 2$. (b) $R \approx 0.17, R' \approx 0.08$ with $k = 4$. (c) $R \approx 0.2, R' \approx 0.1$ with $n = 2$ and $k = 3$. (d) $R \approx 0.33, R' \approx 0.1$ with $n = 4$ and $k = 3$.

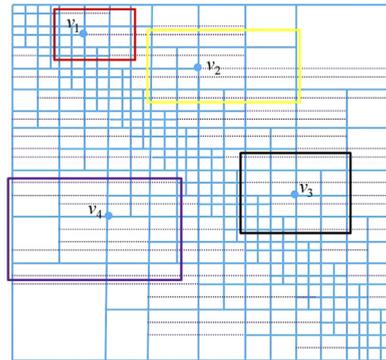


Fig. 12. Supports of Modified T-spline basis functions.

$\sum_{l=1}^k 11(2^{l-1}n - 2) + 16 = 22n \cdot 2^{k-1} - 11n - 6k$ old vertices. So we have $R \approx \frac{20n \cdot 2^{k-1} - 44k}{22n \cdot 2^{k-1} - 6k}$. One can see that R approaches $\frac{10}{11}$ when k is large enough.

The number of additional control points can be quite large and may lead the refined mesh be a tensor product mesh using the local refinement of T-splines (Sederberg et al., 2003). So here we will give another ratio $R' = \frac{N}{TN}$ to give an intuitional expression of how big the refinement region in T' , where N is the total number of new vertices in T' , and TN is the total number of vertices in the corresponding tensor product mesh. For the case in Fig. 11(a), the ratio is $R' = \frac{4k}{(5+2k)^2} \leq \frac{1}{5+k}$. For the case in Fig. 11(b), we have $R' = \frac{7k-1}{(6+3k)^2} \leq \frac{7}{36+9k}$. For the case in Fig. 11(c), there are $(2^{k+1}n + 4k + 5)(2^k n + 2k + 4)$ vertices in the corresponding tensor product mesh, so the ratio $R' \leq \frac{12.5n2^k + 52k}{2^{2k+1}n^2 + 2^k(8k+13)n} \leq \frac{12.5n}{2^{k+1}n^2 + (8k+13)n}$. For the case in Fig. 11(d), there are $(2^k n + 1)^2$ vertices in the corresponding tensor product mesh, so the ratio $R' \leq \frac{20n2^{k-1} - 44k}{(2^k n + 1)^2} \leq \frac{20n}{(2^{k+1}n^2 + 4n)}$.

The above results provide upper bounds for the ratio R and R' . In practice, the ratio is much smaller as shown in Fig. 11.

Next we briefly discuss the supports of Modified T-spline basis functions. Suppose that T is a hierarchical T-mesh and the level differences between adjacent cells are at most one. Denote the corresponding tensor product mesh of T at level l by T_l . Let v be a vertex at level l , then the support of Modified T-spline basis function at v is contained in an $m \times n$ rectangular mesh grid G at T_{l-1} , where $m, n \leq 3$. Fig. 12 shows the supports of the Modified T-spline basis functions at four vertices v_1, v_2, v_3, v_4 , where v_1 is in level 3, v_2 and v_3 are in level 2, and v_4 is in level 1.

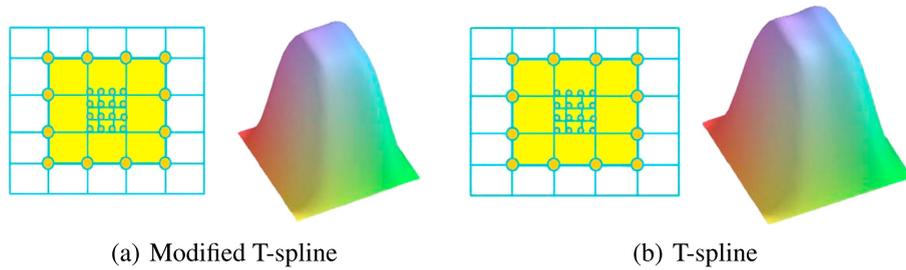


Fig. 13. Modified T-spline and T-spline. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

3.4. Properties of modified T-splines

For a given T-mesh T , we have constructed a set of blending functions $\{B_i(s, t)\}_{i=1}^n$, where n is the number of vertices in T . The constructed blending functions have the following nice properties.

Theorem 3.3. The blending functions $\{B_i(s, t)\}_{i=1}^n$ constructed in Section 3.2 have the following properties:

- C^2 continuity: $B_i(s, t)$ is C^2 continuous over T , $i = 1, 2, \dots, n$.
- Non-negativity: $B_i(s, t) \geq 0$, $i = 1, 2, \dots, n$.
- Partition of unity: $\sum_{i=1}^n B_i(s, t) \equiv 1$.
- Linear independence: $B_i(s, t)$, $i = 1, 2, \dots, n$, are linearly independent.
- Compact support: $B_i(s, t)$ has compact support.

Proof. We only prove that $\{B_i(s, t)\}_{i=1}^n$ are linearly independent and form a partition of unity. We adopt the notations as in Section 3.2.

Assume the extended T-mesh T' has m ($m > n$) vertices. By Lemma 3.2, the T-spline blending functions $\{T_i(s, t)\}_{i=1}^m$ over T' are linearly independent and form a partition of unity.

Without loss of generality, let v_1, v_2, \dots, v_n be the vertices in T and v_{n+1}, \dots, v_m be the new vertices. By the construction process of the blending functions $\{B_i(s, t)\}_{i=1}^n$,

$$\begin{pmatrix} B_1(s, t) \\ B_2(s, t) \\ \vdots \\ B_n(s, t) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & c_{1,n+1} & \cdots & c_{1m} \\ 0 & 1 & 0 & \vdots & c_{2,n+1} & \cdots & c_{2m} \\ 0 & 0 & \ddots & 0 & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & 1 & c_{n,n+1} & \cdots & c_{nm} \end{pmatrix} \begin{pmatrix} T_1(s, t) \\ T_2(s, t) \\ \vdots \\ T_m(s, t) \end{pmatrix} = M(T_1, T_2, \dots, T_m)^T \tag{9}$$

where $\sum_{i=1}^n c_{ij} = 1$, $j = n + 1, \dots, m$.

Obviously, the matrix M in (9) has full rank, and the column sums of M are all one. By Lemma 2.1, $\{B_i(s, t)\}_{i=1}^n$ are linearly independent and form a partition of unity. The lemma is thus proved. \square

Remark 3.2. In the following, we call the blending functions $\{B_i(s, t)\}_{i=1}^n$ basis functions over a T-mesh.

Fig. 13 compares a Modified T-spline function and a T-spline function over a T-mesh. In both figures (a) and (b), the control coefficients corresponding to the vertices marked by orange circles are taken as one while others are taken as zero. So the functions are actually the sum of the blending functions defined over the vertices labeled with orange circles. Figs. 13(a) and (b) depict the shapes of the Modified T-spline function and T-spline function respectively. It can be seen that the Modified T-spline looks more fair than the T-spline.

3.5. Approximation property

Let $\mathcal{S} = span\{B_i(s, t)\}_{i=1}^n$ be the Modified T-spline space defined over T-mesh T , where $B_i(s, t)$ are the basis functions defined over T . For a given continuous function $f(s, t)$, we would like to know the approximation error of f from \mathcal{S} .

In this section, we assume that T is a hierarchical T-mesh such that for any two neighboring cells in T , the level difference is at most one. Ω is the rectangular parameter domain that is occupied by T . Ω_l is the parametric domain that consists of all the cells in level l . h_l is the maximum length of the cells in level l .

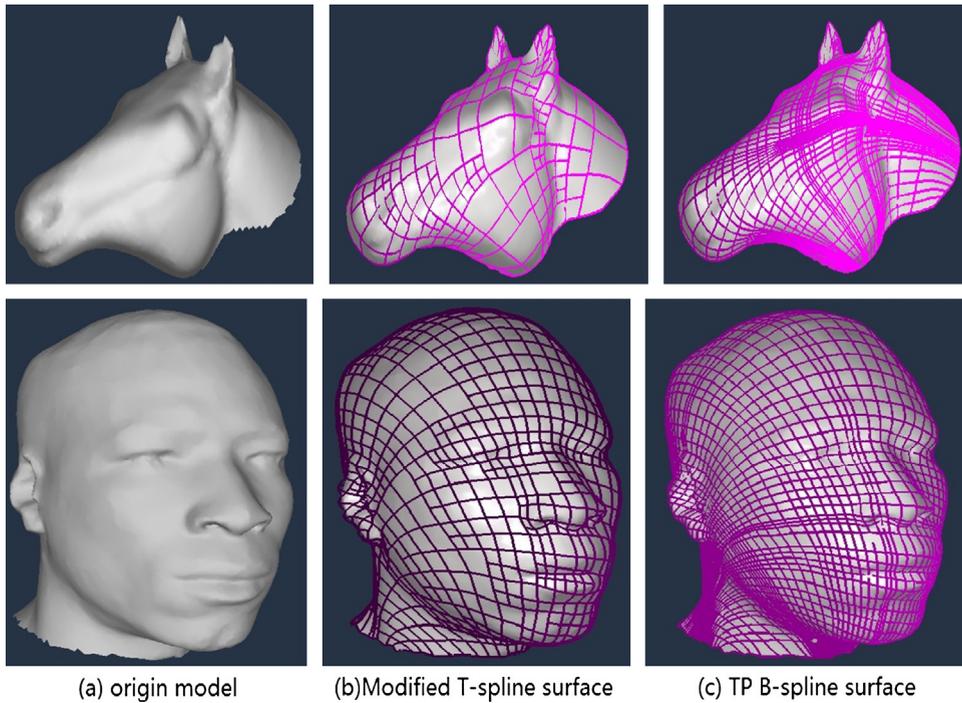


Fig. 14. Conversion of Modified T-spline surfaces to TP-spline surfaces.

Theorem 3.4. For any continuous function $f \in C(\Omega)$, there exists a Modified T-spline function $g(s, t) \in \mathcal{S}$ such that

$$|f(s, t) - g(s, t)| \leq Cw(f, h_l), \quad (s, t) \in \Omega_l. \tag{10}$$

Here $w(f, h)$ is the modulus of continuity of f , that is, $w(f, h) = \max_{\|x-y\|_2 \leq h} |f(x) - f(y)|$. C is a constant which is independent of the mesh T .

Proof. For any function f , define

$$Af(s, t) = \sum_{i=1}^n f(\xi_i, \eta_i)B_i(s, t),$$

where (ξ_i, η_i) are the coordinates of the vertex in T corresponding to the basis function $B_i(s, t)$.

Let $\theta \subset \Omega_l$ be a cell in level l . For any $(\hat{s}, \hat{t}) \in \theta$, we have

$$\begin{aligned} f(\hat{s}, \hat{t}) - Af(\hat{s}, \hat{t}) &= f(\hat{s}, \hat{t}) - \sum_{i \in K} f(\xi_i, \eta_i)B_i(\hat{s}, \hat{t}) \\ &= \sum_{i \in K} [f(\hat{s}, \hat{t}) - f(\xi_i, \eta_i)]B_i(\hat{s}, \hat{t}) \\ &\leq \max_{i \in K} |f(\hat{s}, \hat{t}) - f(\xi_i, \eta_i)| \end{aligned}$$

by the partition of unity of basis functions. Here $K = \{i \mid B_i(\hat{s}, \hat{t}) \neq 0, 1 \leq i \leq n\}$.

Now suppose θ is contained in a cell θ' at level $l-1$. Denote the corresponding tensor product mesh of T at level $l-1$ as T_{l-1} . Then there is a 3×3 rectangular mesh grid G in T_{l-1} whose central cell is θ' , and the B-spline basis functions at the vertices of G do not vanish in θ . If G doesn't contain new vertices of T' (extended T-mesh of T), then $\{(\xi_i, \eta_i) \mid i \in K\} \subset G$, so

$$\max_{i \in K} |f(\hat{s}, \hat{t}) - f(\xi_i, \eta_i)| \leq w(f, 3\sqrt{2}h_{l-1}) \leq (3\sqrt{2} + 1)w(f, h_{l-1}).$$

For a hierarchical T-mesh, we generally have $h_l \geq h_{l-1}/2$. Therefore

$$|f(\hat{s}, \hat{t}) - Af(\hat{s}, \hat{t})| \leq (3\sqrt{2} + 1)w(f, h_{l-1}) \leq 2(3\sqrt{2} + 1)w(f, h_l). \tag{11}$$

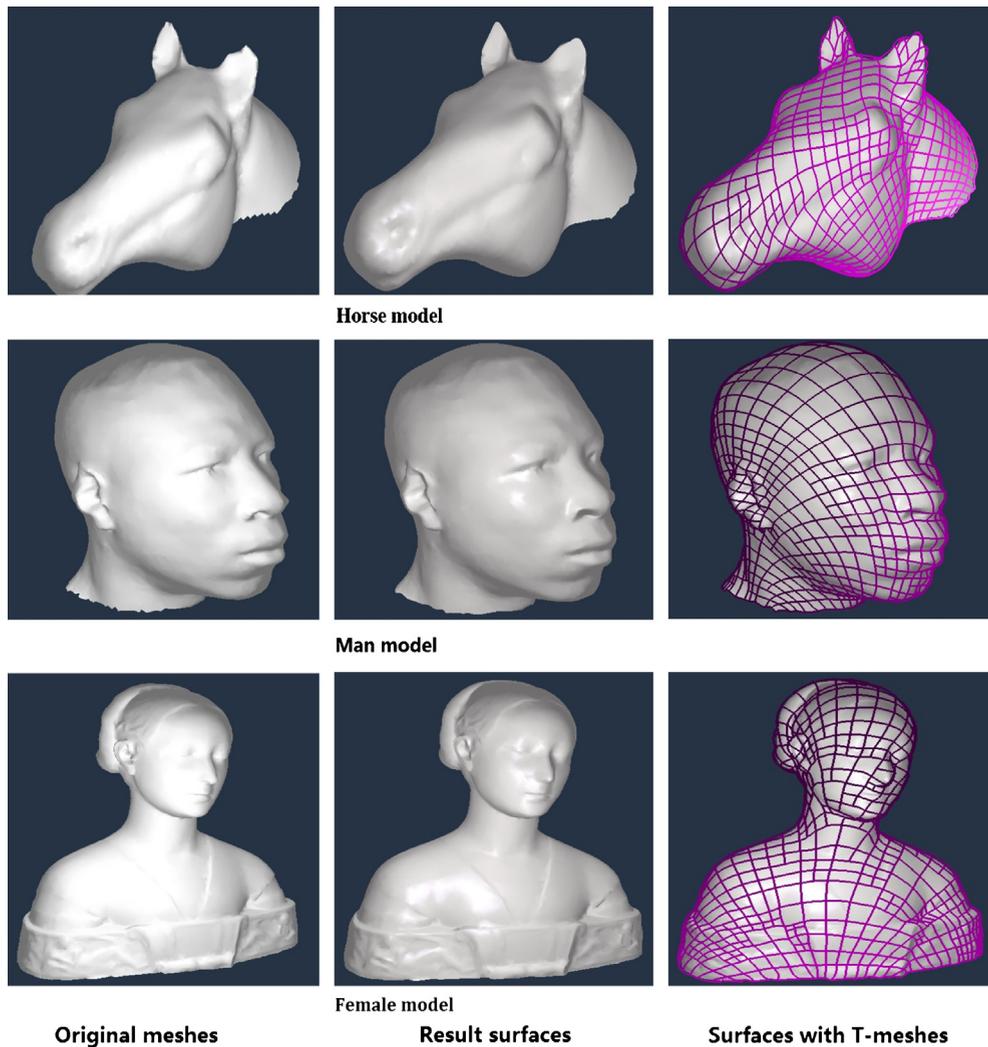


Fig. 15. Fitting open mesh models with Modified T-splines.

Table 1
Statistic data for fitting mesh models.

| Mesh model | #Points | #Faces | #CP | #Levels |
|------------|---------|--------|------|---------|
| Horse | 4750 | 9369 | 898 | 7 |
| Man | 9104 | 18 092 | 1585 | 6 |
| Female | 6301 | 12 487 | 1067 | 8 |

When G contains new vertices of T' , then there exist Modified T-spline basis functions outside of G whose supports contain θ . So we have to enlarge G . Such changes only influence the const C in (10). Fortunately, according to the distribution process, if the level difference between adjacent cells is at most one, G does not need to be enlarged. \square

4. Applications

4.1. Conversion of Modified T-splines to TP-splines

Since Modified T-spline basis functions $\{B_i(s, t)\}_{i=1}^n$ are linear combinations of the T-spline functions $\{T_i(s, t)\}_{i=1}^m$, and by knot insertion, the T-spline functions $\{T_i(s, t)\}_{i=1}^m$ are linear combinations of the TP B-spline basis functions $\{N_i(s, t)\}_{i=1}^l$, so $\{B_i(s, t)\}_{i=1}^n$ are linear combinations of $\{N_i(s, t)\}_{i=1}^l$. In matrix form,

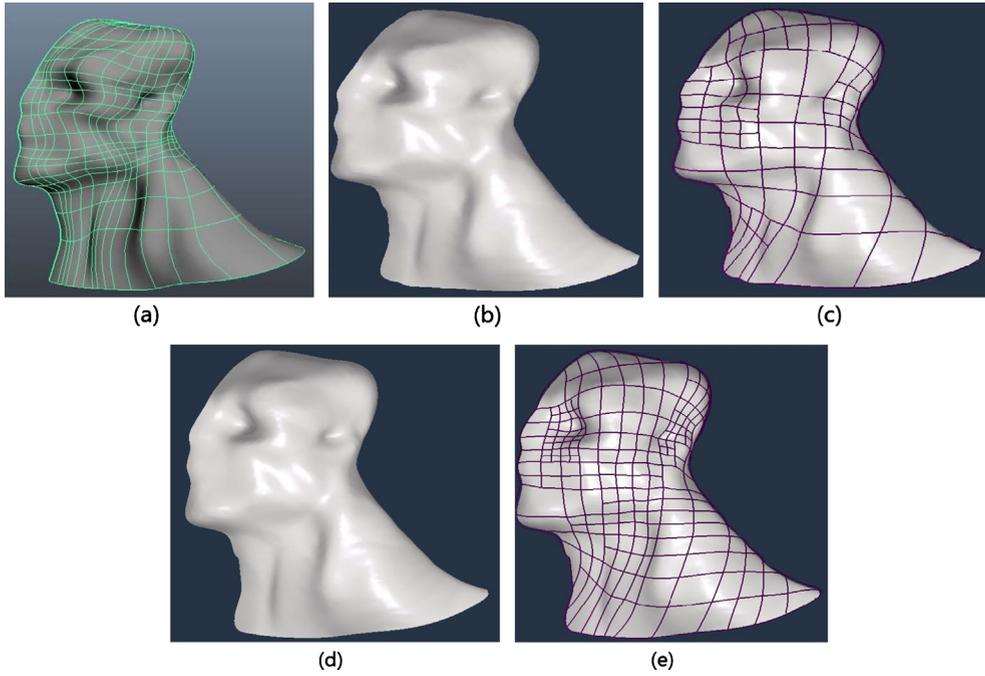


Fig. 16. NURBS approximation. (a) The original NURBS surface with 504 control points. (b) and (c) A Modified T-spline approximation with 184 control points and $\varepsilon = 2.9\%$. (d) and (e) A Modified T-spline approximation with 343 control points and $\varepsilon = 1.8\%$. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

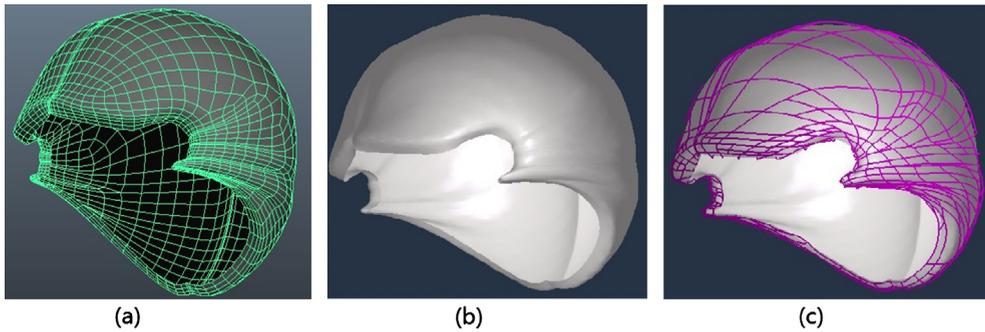


Fig. 17. NURBS approximation. (a) The original NURBS surface with 2096 control points. (b) and (c) A Modified T-spline approximation with 714 control points and $\varepsilon = 1.61\%$. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

$$\begin{aligned} (B_1, \dots, B_n)^T &= M(T_1, \dots, T_m)^T, \\ (T_1, \dots, T_m)^T &= L(N_1, \dots, N_l)^T. \end{aligned}$$

Thus

$$(B_1, \dots, B_n)^T = ML(N_1, \dots, N_l)^T.$$

Suppose we have a Modified T-spline surface

$$S(s, t) = \sum_{i=1}^n \mathbf{P}_i B_i(s, t) = (\mathbf{P}_1, \dots, \mathbf{P}_n)(B_1, \dots, B_n)^T,$$

then it can be converted into a TP B-spline surface:

$$S(s, t) = \sum_{i=1}^l \mathbf{Q}_i N_i(s, t) = (\mathbf{Q}_1, \dots, \mathbf{Q}_l)(N_1, \dots, N_l)^T,$$

where $(\mathbf{Q}_1, \dots, \mathbf{Q}_l) = (\mathbf{P}_1, \dots, \mathbf{P}_n)ML$.

This conversion makes it easy for Modified T-splines to be conveniently imported into the current surface modeling system. Fig. 14 shows the conversion results of the Modified T-spline surface. Here the curves on the surfaces in column (b) are the images of the T-mesh, and the curves on the surface in column (c) are the images of tensor product mesh.

4.2. Fitting open meshes

Suppose we are given an open mesh model with vertices \mathbf{P}_i , $i = 1, 2, \dots, N$, in 3D space, and their corresponding parameter values (s_i, t_i) , $i = 1, 2, \dots, N$, obtained from some parameterization of the mesh (we use the method in Floater, 1997, in the current paper). The parameter domain is assumed to be $[0, 1] \times [0, 1]$.

The surface fitting scheme repeats the following steps 2 and 3 until the fitting error in each cell is less than some tolerance ε .

1. Construct a uniform tensor product mesh T_0 as the initial mesh. Set $k = 0$.
2. Solve a least square fitting problems on the k th level mesh T_k to find a Modified T-spline surface $S_k(s, t)$ to fit the given mesh model.
3. Search for the cells of the T-mesh T_k whose fitting errors are greater than ε , then split these cells into four sub-cells to obtain a new mesh T_{k+1} . The fitting error over a cell s is defined to be $\max_{(s_i, t_i) \in s} \|\mathbf{P}_i - S_k(s_i, t_i)\|$. Set $k := k + 1$.

Fig. 15 illustrates three examples for fitting open meshes with Modified T-splines, and Table 1 shows the statistic data including the number points/faces of the mesh models, the number of control points of the fitting spline surfaces and the fitting level.

4.3. NURBS approximation

The surface fitting scheme provided in the previous subsections can be easily adapted to approximate a NURBS model with a Modified T-spline surface. We illustrate two examples as shown in Figs. 16 and 17. In each example, the approximation error ε and the number of control points in the approximating Modified T-spline are provided. Here the control nets of NURBS surface are displayed in bright green, and the one of the Modified T-splines are in pink. From the examples, it can be seen that there is a considerable reduction of control points of Modified T-splines compared with NURBS representations.

5. Conclusions and future work

This paper proposes a new local refinement splines called Modified T-splines. The basic idea is to construct a set of basis functions over a T-mesh which have good properties such as non-negativity, partition of unity and compact support. Due to the properties of the basis functions, Modified T-splines inherit many good properties of current local refinement splines, and thus should be useful both in geometric modeling and isogeometric analysis.

There are a few problems worthy of further investigation. First, the local refinement algorithm is crucial in adaptive modeling and analysis of splines. We will discuss the local refinement algorithm for Modified T-splines. Second, in the current construction, the Modified T-splines are C^2 continuous globally. We will investigate the possibility to insert multiple knots such that C^0 or C^1 continuity could be achieved at the knot lines, i.e., we will consider Modified T-splines over μ -extended box partitions (Dokken et al., 2013). Finally, we will investigate further applications of Modified T-splines in geometric modeling and isogeometric analysis.

Acknowledgements

The authors thank the reviewers for providing useful comments and suggestion. The work is supported by 973 Program 2011CB302400, the NSF of China No. 11031007.

References

- Bazilevs, Y., Calo, V.M., Cottrell, J.A., Evans, J.A., Hughes, T.J.R., Lipton, S., Scott, M.A., Sederberg, T.W., 2010. Isogeometric analysis using T-splines. *Comput. Methods Appl. Mech. Eng.* 199 (5–8), 229–263.
- Buffa, A., Cho, D., Sangalli, G., 2010. Linear independence of the T-spline blending functions associated with some particular T-meshes. *Comput. Methods Appl. Mech. Eng.* 199, 1437–1445.
- Cottrell, J.A., Hughes, T.J.R., Bazilevs, Y., 2009. *Isogeometric Analysis: Toward Integration of CAD and FEA*. Wiley, Chichester.
- Deng, Jiandong, Chen, Falai, Li, Xin, Hu, Changqi, Tong, Weihua, Yang, Zhouwang, Feng, Yuyu, 2008. Polynomial splines over hierarchical T-meshes. *Graph. Models* 70, 76–86.
- Dokken, T., Lyche, Tom, Pettersen, Kjell Fredrik, 2013. Polynomial splines over locally refined box-partitions. *Comput. Aided Geom. Des.* 30, 331–356.
- Dörfler, M., Jüttler, B., Simeon, B., 2009. Adaptive isogeometric analysis by local h-refinement with T-splines. *Comput. Methods Appl. Mech. Eng.* 199 (5–8), 264–275.
- Floater, Michael S., 1997. Parametrization and smooth approximation of surface triangulations. *Comput. Aided Geom. Des.* 14, 231–250.
- Forsey, D.R., Bartels, R.H., 1988. Hierarchical B-spline refinement. *Comput. Graph.* 22 (4), 205–212.
- Giannelli, Carlotta, Jüttler, Bert, Speleers, Hendrik, 2012. THB-splines: The truncated basis for hierarchical splines. *Comput. Aided Geom. Des.* 29 (7), 485–498.

- Hughes, T.J.R., Cottrell, J.A., Bazilevs, Y., 2005. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry, and mesh refinement. *Comput. Methods Appl. Mech. Eng.* 194, 4135–4195.
- Li, Xin, Zheng, Jianming, Sederberg, T.W., Hughes, T.J.R., Scott, M.A., 2012. On the linear independence of T-spline blending functions. *Comput. Aided Geom. Des.* 29 (1), 63–76.
- Nguyen-Thanh, N., Nguyen-Xuan, H., Bordas, S.P.A., Rabczuk, T., 2011a. Isogeometric analysis using polynomial splines over hierarchical T-meshes for two-dimensional elastic solids. *Comput. Methods Appl. Mech. Eng.* 200 (21–22), 1892–1908.
- Nguyen-Thanh, N., Kiendl, J., Nguyen-Xuan, H., Wüchner, R., Bletzinger, K.U., Bazilevs, Y., Rabczuk, T., 2011b. Rotation free isogeometric thin shell analysis using PHT-splines. *Comput. Methods Appl. Mech. Eng.* 200 (47–48), 3410–3424.
- Scott, M.A., Li, X., Sederberg, T.W., Hughes, T.J.R., 2012. Local refinement of analysis-suitable T-splines. *Comput. Methods Appl. Mech. Eng.* 213–216, 206–222.
- Sederberg, T., Zheng, J., Bakenov, A., Nasri, A., 2003. T-splines and T-NURCCs. *ACM Trans. Graph.* 22 (3), 477–484.
- Sederberg, T., Cardon, D., Finnigan, G., North, N., Zheng, J., Lyche, T., 2004. T-spline simplification and local refinement. *ACM Trans. Graph.* 23 (3), 276–283.