ARTICLE IN PRESS

Computers and Mathematics with Applications ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

Truncated Hierarchical Loop Subdivision Surfaces and application in isogeometric analysis

Hongmei Kang, Xin Li *, Falai Chen, Jiansong Deng

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

ARTICLE INFO

Article history:
Available online xxxx

Keywords:
Loop subdivision
Local refinement
Truncation
Extraordinary vertices
Isogeometric analysis

ABSTRACT

Subdivision Surface provides an efficient way to represent free-form surfaces with arbitrary topology. Loop subdivision is a subdivision scheme for triangular meshes, which is C^2 continuous except at a finite number of extraordinary vertices with G^1 continuous. In this paper we propose the Truncated Hierarchical Loop Subdivision Surface (THLSS), which generalizes truncated hierarchical B-splines to arbitrary topological triangular meshes. THLSS basis functions are linearly independent, form a partition of unity, and are locally refinable. THLSS also preserves the geometry during adaptive h-refinement and thus inherits the surface continuity of Loop subdivision surface. Adaptive isogeometric analysis is performed with the THLSS basis functions on several complex models with extraordinary vertices to show the potential application of THLSS.

1. Introduction

Isogeometric analysis (IGA) was originally introduced by Hughes et al. [1] and described in detail in [2]. With IGA, traditional design-through-analysis procedures such as geometry clean-up, defeaturing, and mesh generation are simplified or eliminated entirely. Additionally, the higher-order smoothness provides substantial gains to analysis in terms of accuracy and robustness of finite element solutions [3–5]. However, a global geometric discretization, based on NURBS, is usually not suitable as a basis for analysis. Many different methods have been developed in these years to define locally refinable splines, such as the (Truncated) Hierarchical B-splines [6–8], the (Analysis-suitable) T-splines [9–12], the PHT-splines [13–15], the LR B-splines [16,17] and the Modified T-splines [18]. Truncated hierarchical Catmull–Clark subdivision (THCCS) [19,20] generalized Truncated Hierarchical B-splines [8] to control grids of arbitrary topology. THCCS provide a method to define locally refinable splines on quadrilateral meshes with extraordinary nodes.

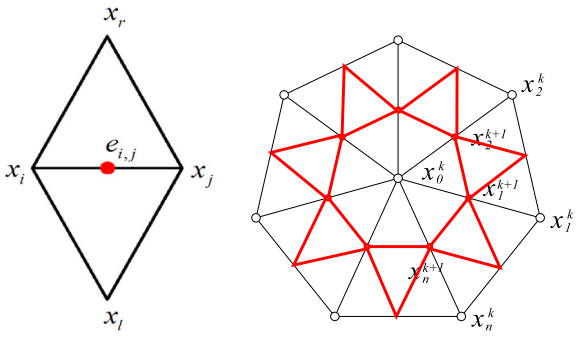
Recently locally refinable splines on triangular partitions also attract researchers's interest because of the flexibility and the popular using in classical finite element analysis of triangular partitions. Hierarchical bivariate splines on regular (type-I and type-II) triangular partitions were introduced in [21] and applied to numerical solving PDEs. Later, Jüttler et al. [22] generalized the truncated hierarchical B-splines [8] to hierarchies of spaces that are spanned by generating systems that potentially possess linear dependencies, a special box splines defined on criss-cross grid called Zwart–Powell (ZP) elements was discussed as an example. Speleers et al. [23,24] proposed hierarchical Powell–Sabin splines for isogeometric analysis applications, where Powell–Sabin splines are C^1 piecewise quadratic polynomials defined on a special refinement of any given triangulation.

Loop subdivision [25] is a subdivision scheme for triangular meshes. The limit surface defined by Loop subdivision is C^2 continuous except at a finite number of extraordinary vertices (an extraordinary vertex has other than six faces adjacent

* Corresponding author.
E-mail address: lixustc@ustc.edu.cn (X. Li).

http://dx.doi.org/10.1016/j.camwa.2016.06.045
0898-1221/

Please cite this article in press as: H. Kang, et al., Truncated Hierarchical Loop Subdivision Surfaces and application in isogeometric analysis, Computers and Mathematics with Applications (2016), http://dx.doi.org/10.1016/j.camwa.2016.06.045



(a) Edge point rule. (b) Vertex point rule.

Fig. 1. Mask for Loop subdivision.

to it) where the surface is G^1 continuous. Explicit Loop basis functions were explored by J. Stam [26] and have several nice properties: linear independence, partition of unity and local support. There recently have been a few works on the application of Loop subdivision in isogeometric analysis. Loop subdivision surfaces were used for describing the geometry of shell and the displacement fields in thin-shell finite element analysis [27]. Extended Loop subdivision surfaces were used in isogeometric analysis in [28], where Poisson equations with the Dirichlet boundary condition were considered and the approximation properties of extended Loop basis functions were established.

In this paper we introduce the truncation hierarchical mechanism [8] into Loop subdivision surfaces, which are called Truncated Hierarchical Loop Subdivision Surfaces (THLSS), to be adapted to triangular meshes with arbitrary topology and support local refinement. THLSS preserve the exact geometry when adaptive h-refinement is performed and inherit the surface continuity of Loop subdivision surfaces. THLSS basis functions are global linearly independent, form a partition of unity and have local support. We applied THLSS basis functions in isogeometric analysis on several complex geometries. The simulation results show potential wide application of the proposed method in integrating design and analysis. Through a benchmark numerical experiment, we demonstrate its efficiency with the comparison to the classical finite element analysis piecewise linear elements.

The paper is organized as follows: Section 2 briefly reviews Loop subdivision scheme including Stam's explicit basis functions. Section 3 presents the detail of THLSS construction. Section 4 shows several numerical experiments with the comparison to the FEA with linear elements and Loop basis functions. Section 5 is the conclusion and future work.

2. Loop subdivision surface

In this section, we briefly review Loop's subdivision scheme and the explicit basis functions introduced by J. Stam [26].

2.1. Loop subdivision scheme

Loop subdivision scheme is an approximating subdivision scheme. Referring to Fig. 1, let x_l and x_r be the two wing neighbor vertices of edge$[x_i x_j]$, then the new edge point added on this edge is defined as

$$e_{i,j} = \frac{3}{8}x_i + \frac{3}{8}x_j + \frac{1}{8}x_l + \frac{1}{8}x_r.$$

And for a vertex x_0^k at level k with neighboring vertices x_i^k, $i = 1, 2, \ldots, n$, where n is the valence of vertex x_0^k. The old vertex is updated to x_0^{k+1} according to

$$x_0^{k+1} = (1 - n\alpha)x_0^k + \alpha(x_1^k + x_2^k + \cdots + x_n^k),$$

where $\alpha = \frac{1}{n}[\frac{5}{8} - (\frac{3}{8} + \frac{1}{4}\cos(\frac{2\pi}{n}))^2]$. This linear relationship can be expressed by a so-called subdivision matrix. The repeated global refinement generates a sequence of meshes $\mathbb{M}^0, \ldots, \mathbb{M}^n$, where $\mathbb{M}^0$ is the initial control grid, and n is the number of subdivisions. As n goes to infinity, $\mathbb{M}^n$ converges to a limit surface. We call this limit surface as *Loop subdivision surface*.

2.2. Loop basis functions

An alternative way to obtain the limit surface takes advantage of the Stam's basis functions [26]. These basis functions are analogous to B-spline basis functions, whereas each mesh $\mathbb{M}^l$ is served as a control grid. Thus we can express the limit surface S_{limit} by a mapping from the parametric domain to the physical domain,

$$S_{limit}(v, w) = \sum_{i=1}^{N^l} B_i^l(v, w)P_i^l, \tag{1}$$

Please cite this article in press as: H. Kang, et al., Truncated Hierarchical Loop Subdivision Surfaces and application in isogeometric analysis, Computers and Mathematics with Applications (2016), http://dx.doi.org/10.1016/j.camwa.2016.06.045

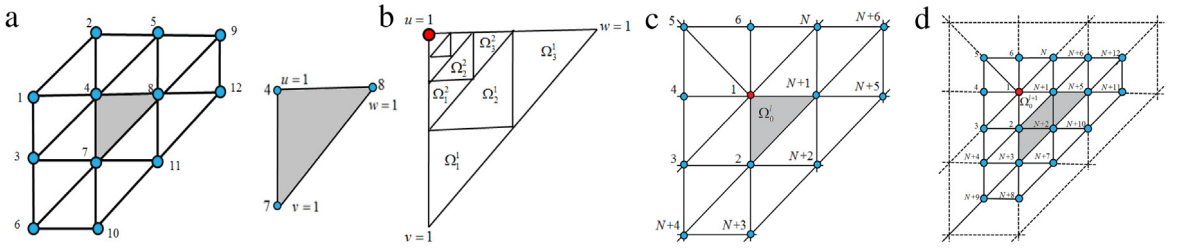


Fig. 2. (a) A regular patch (shaded patch) defined by 12 control vertices. (b) The parameter domain is partitioned into a set of triangular tiles. (c) An irregular triangular patch (shaded patch) defined by $N+6$ control vertices. (d) The order of control vertices after one subdivision step. The shaded patches are regular patches.

where $B_i^l(v, w)$ are Stam's basis functions, P_i^l are the control points for mesh $\mathbb{M}^l$ and (v, w) are the local parametric coordinates. Next, we give the details for evaluation of the basis functions.

For any given triangular control grid, after one level of subdivision, all the extraordinary vertices are separated, i.e., each triangular face contains at most one extraordinary vertex. Here an extraordinary vertex means it has other than six patches adjacent to it. Such mesh is called a *valid control mesh*. Thus, all the meshes $\mathbb{M}^l$, $l > 0$ are valid control meshes.

First, we provide the basis functions $B_i^l(v, w)$ for $l > 0$ since all the control meshes are valid control meshes. A triangular patch is called *regular* if all of its three vertices have a valence of 6, otherwise it is called *irregular*. A regular patch can be exactly described by a quartic C^2 continuous box spline, which is defined as follows:

$$S(v, w) = C^T \mathbf{b}(v, w), \quad (v, w) \in \Omega, \tag{2}$$

where C is the control vertices in $\mathbb{R}^3$ of the patch ordered as in Fig. 2(a), $\mathbf{b}(v, w)$ is a vector containing 12 box spline basis functions which do not vanish on the patch and $\Omega = \{(v, w) \in \mathbb{R}^2 : v \geq 0, w \geq 0, v + w \leq 1\}$ is an unit triangle.

An irregular patch with an extraordinary vertex of valence N is determined by its two-ring neighborhood control vertices, namely $N + 6$ control vertices. In the paper [26], they seek a parametrization $S(v, w)$ for an irregular patch. First, the parameter domain Ω is partitioned n times until the resulting irregular region is small enough, as shown in Fig. 2(b). These sub-domains are defined more precisely as:

$$\begin{aligned}
\Omega_1^n &= \{(v, w) | v \in [2^{-n}, 2^{-n+1}], w \in [0, 2^{-n+1} - v]\},\\
\Omega_2^n &= \{(v, w) | v \in [0, 2^{-n}], w \in [0, v]\},\\
\Omega_3^n &= \{(v, w) | v \in [0, 2^{-n}], w \in [2^{-n}, 2^{-n+1} - v]\}.
\end{aligned}$$

Each Ω_k^n is a regular patch. Denote the $N + 6$ control vertices around the irregular patch as a vector C_0^T. Fig. 2(c) shows the order of $N + 6$ control points of an irregular patch, where the extraordinary vertex is labeled with 1. Then the surface patch $S(v, w)$ is defined by its restriction to each of the sub-domains:

$$S(v, w)|_{\Omega_k^n} = C_0^T (P_k \overline{A} A^{n-1})^T \mathbf{b}(t_{n,k}(v, w)), \quad k = 1, 2, 3, \tag{3}$$

where P_k is a selection matrix to locate which regular patch ($\Omega_k^n, k = 1, 2, 3$) contains parametric value (v, w) after n times of subdivisions, the subdivision matrix A and $\overline{A}$ can be found in the appendix in [26], and the transformation $t_{n,k}$ maps the tile Ω_k^n onto the unit tile Ω:

$$\begin{aligned}
t_{n,1}(v, w) &= (2^n v - 1, 2^n w),\\
t_{n,2}(v, w) &= (1 - 2^n v, 1 - 2^n w),\\
t_{n,3}(v, w) &= (2^n v, 2^n w - 1).
\end{aligned}$$

With the expression of the triangular patch $S(v, w)$ in (2) and (3), we are ready to define Loop basis functions $B_i^l(v, w)$ at level l.

The Loop basis functions restricted on a regular patch are defined as box splines, according to (2),

$$\mathbf{B}^l(v, w) = \mathbf{b}(v, w). \tag{4}$$

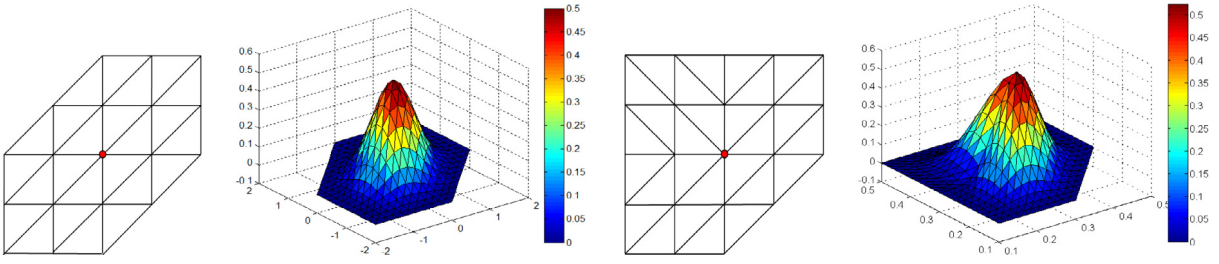
While the Loop basis functions restricted on an irregular patch are defined as follows, according to (3),

$$\mathbf{B}^l(v, w) = (V^{-1})^T \Lambda^{n-l-1} (P_k \overline{A} V)^T \mathbf{b}(v, w), \tag{5}$$

where Λ is the Jordan canonical form of A with $A = V \Lambda V^{-1}$. $\mathbf{B}^l(v, w)$ only depends on the valence number of the vertex.

A Loop basis function restricted on a patch is either a box spline or a linear combination of box splines. Owing to the properties of box splines and subdivision matrix, Loop basis functions are nonnegative, form a partition of unity and have global linear independency [29]. A Loop basis function has support on its two-ring neighborhood patches. Fig. 3 shows two Loop basis functions associated with a regular node and an irregular node, together with their support.

Please cite this article in press as: H. Kang, et al., Truncated Hierarchical Loop Subdivision Surfaces and application in isogeometric analysis, Computers and Mathematics with Applications (2016), http://dx.doi.org/10.1016/j.camwa.2016.06.045



(a) A Loop basis function associated with a regular node.

(b) A Loop basis function associated with an irregular node.

Fig. 3. Two basis functions together with their local support, where red circles are the associated nodes. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

We can define the basis functions $B_i^0(v, w)$ in the similar way. First according to the subdivision rule, there exists a subdivision matrix S such that $P^1 = SP^0$, where P^1, P^0 are the vectors that contain the vertices in the mesh of $\mathbb{M}^1$ and $\mathbb{M}^0$ respectively. Notice that both control grids $\mathbb{M}^1$ and $\mathbb{M}^0$ will define the same limit surface via loop subdivision, thus

$$S_{limit} = \sum_{i=1}^{N^0} B_i^0(\xi, \eta) P_i^0 = \sum_{i=1}^{N^1} B_i^1(\xi, \eta) P_i^1. \tag{6}$$

This relationship can be written into a matrix form

$$[B_1^0, \ldots, B_{N^0}^0]P^0 = [B_1^1, \ldots, B_{N^1}^1]P^1 = [B_1^1, \ldots, B_{N^1}^1]SP^0,$$

thus,

$$[B_1^0, \ldots, B_{N^0}^0] = [B_1^1, \ldots, B_{N^1}^1]S.$$

3. Truncated hierarchical loop subdivision surfaces

In this section we are going to construct Truncated Hierarchical Loop Subdivision Surfaces in a similar manner as for truncated hierarchical B-splines.

3.1. Notations

An initial Loop control mesh is denoted by $\mathbb{M}^0$. After performing l times of Loop subdivision on $\mathbb{M}^0$, we denote the resulting control mesh as $\mathbb{M}^l$. A triangular patch(element) in $\mathbb{M}^l$ is denoted by Ω_i^l, and a Loop basis function associated with a vertex(node) in $\mathbb{M}^l$ is denoted by B_i^l. We also call Ω_i^l a patch at level l and B_i^l a Loop basis function at level l.

Let $\mathcal{F}^l$ be the set of all patches in $\mathbb{M}^l$ and $\mathcal{B}^l$ be the set of all the basis functions at level l,

$$\mathcal{F}^l = \{\Omega_i^l \mid i = 0, 1, \ldots, n_e^l\}, \tag{7}$$

$$\mathcal{B}^l = \{B_i^l \mid i = 0, 1, \ldots, n_b^l\}, \tag{8}$$

where n_e^l is the number of patches in $\mathbb{M}^l$ and n_b^l is the number of basis functions at level l. The domain covers the support of all basis functions in $\mathcal{B}^0$ is denoted by Ω^0, where $\Omega^0 = \text{supp}\ \mathcal{B}^0 = \bigcup_{i=0}^{n_e^0} \Omega_i^0$. The refined domain at level l is denoted by $\Omega^l, l > 0$.

Refinability of Loop basis functions. Denote the Loop basis functions over irregular patches Ω_0^l and Ω_0^{l+1} in Fig. 2(c), (d) as $\mathbf{B}^l$ and $\mathbf{B}^{l+1}$. According to (5), we have

$$\begin{aligned}\mathbf{B}^{l+1} &= (V^{-1})^T \Lambda^{n-l-2} (P_k \bar{A} V)^T b \\ &= (V^T)^{-1} \Lambda^{-1} \Lambda^{n-l-1} (P_k \bar{A} V)^T b \\ &= (\Lambda V^T)^{-1} \Lambda^{n-l-1} (P_k \bar{A} V)^T b.\end{aligned}$$

Recall that $AV = V\Lambda$ and Λ is a diagonal matrix, so $\Lambda = \Lambda^T$ and $\Lambda V^T = V^T A^T$. Then we have

$$\mathbf{B}^{l+1} = (V^T A^T)^{-1} \Lambda^{n-l-1} (P_k \bar{A} V)^T b = (A^T)^{-1} \mathbf{B}^l.$$

That is

$$\mathbf{B}^l = A^T \mathbf{B}^{l+1}. \tag{9}$$

On a regular patch, relationship equation (9) is obviously valid. Eq. (9) indicates the refinability still holds for Loop basis functions via the subdivision matrix A. So we can define the truncation mechanism of Loop basis functions as truncated hierarchical B-splines [8].

Please cite this article in press as: H. Kang, et al., Truncated Hierarchical Loop Subdivision Surfaces and application in isogeometric analysis, Computers and Mathematics with Applications (2016), http://dx.doi.org/10.1016/j.camwa.2016.06.045

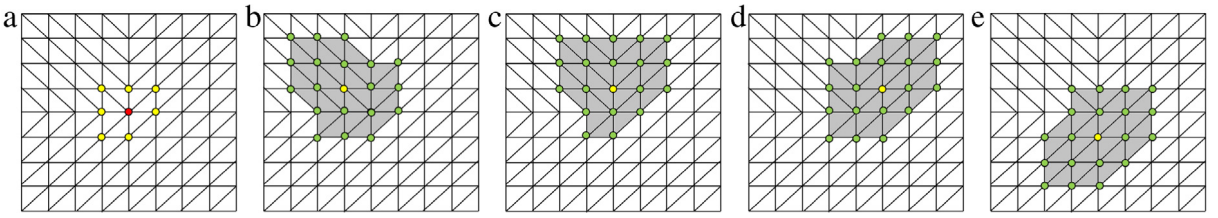


Fig. 4. The marked yellow circles are identified as to be refined. The shaded triangles serve as the fine-level domain Ω^{l+1}. The basis functions to be truncated are marked by green circles. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Definition 1 ([8]). Let $\beta \in \mathcal{B}^l$ and let

$$\beta = \sum_{\tau \in \mathcal{B}^{l+1}} c_\tau^{l+1}(\beta)\tau, \quad c_\tau^{l+1} \in \mathbb{R},$$

be its representation with respect to the finer basis of $\mathcal{B}^{l+1}$. The truncation of β with respect to $\mathcal{B}^{l+1}$ and Ω^{l+1} is defined as

$$\text{trunc}^{l+1}\beta = \sum_{\tau \in \mathcal{B}^{l+1}, \text{supp}\,\tau \not\subseteq \Omega^{l+1}} c_\tau^{l+1}(\beta)\tau, \tag{10}$$

where the coefficients c_τ^{l+1} come from the subdivision matrix A according to (9).

3.2. Construction

Now we can construct truncated hierarchical Loop subdivision surface by Definition 2, which is similar to the definition of truncated hierarchical B-splines [8].

Definition 2. Suppose L is the maximum subdividing step. $\mathcal{B}^l$ and Ω^l are defined as in Section 3.1, then the truncated hierarchical Loop subdivision surface basis $\mathcal{H}$ is recursively constructed as follows:

1. Initialization: $\mathcal{H}^0 = \{\beta \in \mathcal{B}^0 : \text{supp}\,\beta \neq \emptyset\}$.
2. Recursive case: $\mathcal{H}^{l+1} = \mathcal{H}_A^{l+1} \cup \mathcal{H}_B^{l+1}$, for $l = 0, \ldots, L-1$, where

$$\mathcal{H}_A^{l+1} = \{\text{trunc}^{l+1}\beta : \beta \in \mathcal{H}^l,\ \text{supp}\,\beta \not\subseteq \Omega^{l+1}\},$$

and

$$\mathcal{H}_B^{l+1} = \{\beta \in \mathcal{B}^{l+1} : \text{supp}\,\beta \subseteq \Omega^{l+1}\}.$$

3. $\mathcal{H} = \mathcal{H}^L$.

In the following, we discuss the construction in detail.

- **Identification of refinement domains** Ω^{l+1}.
  From level l to $l+1$, suppose a set of basis functions at level l is selected to be refined, denoted as $\mathcal{H}_r^l$, according to the local geometry or simulation error with a given threshold. The fine-level domain Ω^{l+1} is defined by the support of all the to-be-refined basis functions,

$$\Omega^{l+1} = \text{supp}\,\mathcal{H}_r^l. \tag{11}$$

  The set of to-be-refined elements $\mathcal{F}_r^l$ contains all the elements inside Ω^{l+1},

$$\mathcal{F}_r^l = \{\Omega_i^l \mid \Omega_i^l \in \mathcal{F}^l,\ \Omega_i^l \subseteq \Omega^{l+1}\}. \tag{12}$$

  If the marked yellow basis functions in Fig. 4(a) are identified as to-be-refined ($\mathcal{H}_r^l$), then all the shaded elements in Fig. 4(b)–(e) will be identified as to-be-refined elements ($\mathcal{F}_r^l$). In this situation the shaded region will also serve as the fine-level domain Ω^{l+1}.
- **Construction of part** $\mathcal{H}_A^{l+1}$.
  According to (9), for any basis function $\beta \in \mathcal{H}^l$, it can be represented by $\mathcal{B}^{l+1}$,

$$\beta = \sum_{i=1}^{n_b^{l+1}} d_j B_j^{l+1}, \quad d_j \in \mathbb{R}.$$

  When $d_j \neq 0$, the basis B_j^{l+1} is called a child at level $l+1$ of β.

Please cite this article in press as: H. Kang, et al., Truncated Hierarchical Loop Subdivision Surfaces and application in isogeometric analysis, Computers and Mathematics with Applications (2016), http://dx.doi.org/10.1016/j.camwa.2016.06.045

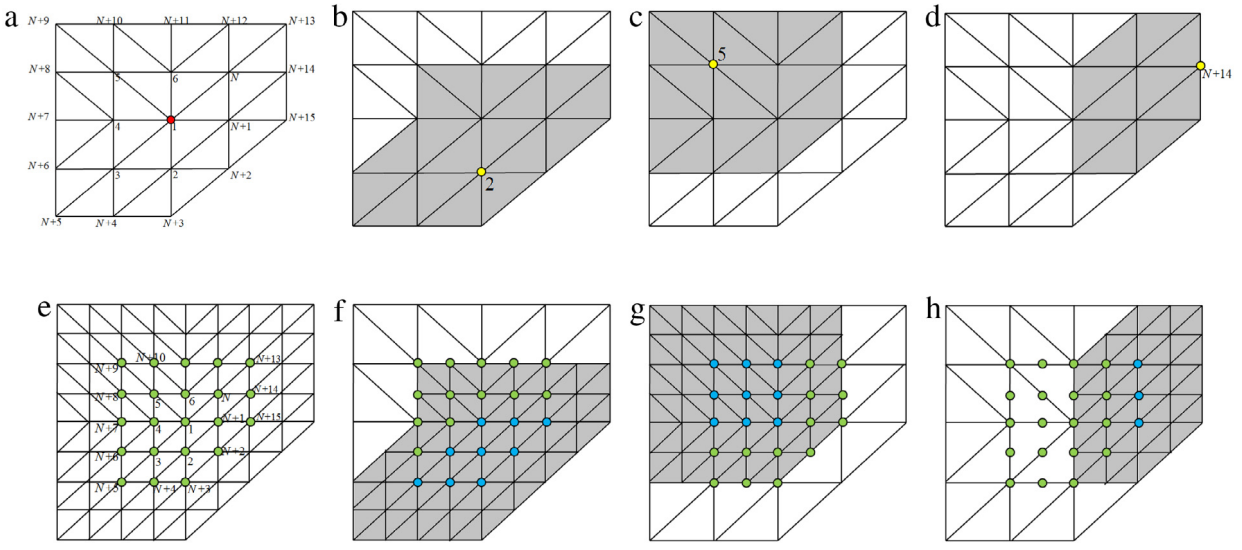


Fig. 5. Construction of the truncated basis function $\text{trunc}^{l+1}B_1^l$. (a) The two-ring neighborhood around a valence-7 extraordinary node at Level l; (b) B_2^l is to be refined (Case 1); (c) B_5^l is to be refined (Case 2); (d) B_{N+14}^l is to be refined (Case 3); and (e–h) are refinement of (a–d), respectively. Blue dots are discarded in constructing $\text{trunc}^{l+1}B_1^l$, while green dots are kept. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

In this step, we first identify to-be-truncated basis functions $\mathcal{H}_t^l$ from the remaining basis functions $\mathcal{H}_a^l = \mathcal{H}^l \setminus \mathcal{H}_r^l$. If a basis function $\beta \in \mathcal{H}_a^l$ has a children at level $l+1$ with support fully contained in Ω^{l+1}, then it will be truncated. We have

$$\mathcal{H}_t^l = \{\beta \mid \beta \in \mathcal{H}_a^l,\ \exists\, \text{chd}\ \beta \text{ s.t. } \text{supp chd}\ \beta \subseteq \Omega^{l+1}\}.$$

For Loop basis functions, all the two-ring neighboring basis functions around $\mathcal{H}_r^l$ should be truncated. In Fig. 4(b)–(e), the to-be-truncated basis functions are marked by green circles. Then we truncate each basis function $\beta \in \mathcal{H}_t^l$ by discarding its children with support fully contained in Ω^{l+1}, and obtain

$$\text{trunc}^{l+1}\beta = \sum_{\text{supp}\, B_j^{l+1} \not\subseteq \Omega^{l+1}} d_j B_j^{l+1}.$$

- **Construction of part $\mathcal{H}_B^{l+1}$.**

  This step aims at refining all the elements in $\mathcal{F}_r^l$ and obtaining all the basis functions whose support is contained in Ω^{l+1}. All the children basis functions of $\mathcal{H}_r^l$ are contained in Ω^{l+1}, thus $\mathcal{H}_B^{l+1} = \text{chd}\ \mathcal{H}_r^l$. The active element set at level l is $\mathcal{F}_a^l = \mathcal{F}^l \setminus \mathcal{F}_r^l$ and the active element set at level $l+1$ is $\mathcal{F}_a^{l+1} = \text{supp}\ \mathcal{H}_B^{l+1}$.

Finally the THLSS basis functions at level $l+1$ are composed of two parts: $\mathcal{H}^{l+1} = \mathcal{H}_A^{l+1} \cup \mathcal{H}_B^{l+1}$. The active elements for $\mathcal{H}^{l+1}$ is defined as $\mathcal{F}_a^l \cup \mathcal{F}_a^{l+1}$, which form the THLSS control meshes.

3.3. Examples

In the following, we take a valence-7 extraordinary node, Node 1 in Fig. 5(a), as an example to construct the truncated basis function associated with it. For the present case, we only need to consider the two-ring neighborhood elements around the extraordinary node.

To simplify the exposition, we consider only one basis function to be refined at one time. Particularly, we study three cases as shown in Fig. 5(b)–(d):

1. Case 1: $\mathcal{H}_r^l = \{B_2^l\}$,
2. Case 2: $\mathcal{H}_r^l = \{B_5^l\}$,
3. Case 3: $\mathcal{H}_r^l = \{B_{N+14}^l\}$.

Corresponding to each case, the fine-level domain Ω^{l+1} is the two-ring neighborhood of $\mathcal{H}_r^l$, which is shaded in Fig. 5(b)–(d). All the elements in Ω^{l+1} are identified as to-be-refined elements $\mathcal{F}_r^l$.

The two-ring neighborhood basis functions around $\mathcal{H}_r^l$ are identified as to-be-truncated because they have children whose support contained in Ω^{l+1}. Without loss of generality, here we study how to construct the truncated basis function

Please cite this article in press as: H. Kang, et al., Truncated Hierarchical Loop Subdivision Surfaces and application in isogeometric analysis, Computers and Mathematics with Applications (2016), http://dx.doi.org/10.1016/j.camwa.2016.06.045

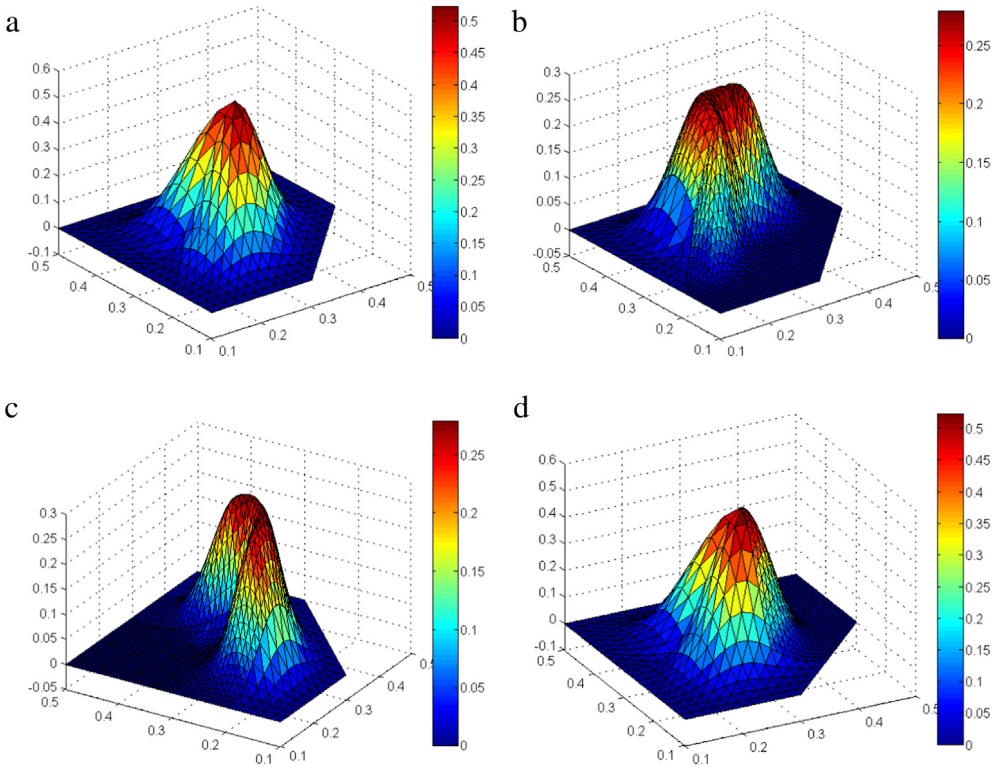


Fig. 6. (a) Original basis functions associated with extraordinary nodes of valence-7; (b–d) Truncated basis functions corresponding to three truncation cases.

for B_1^l in all three cases, see Fig. 5(a), which is a basis function associated with the extraordinary node (red circles). The other truncated basis functions can be constructed in a similar manner. Fig. 5(e) shows the refinement of all the elements in Fig. 5(a), where the $N + 15$ basis functions marked by green circles are the children of B_1^l. B_1^l can be represented by a linear combination of its children,

$$B_1^l = \sum_{j=1}^{N+15} c_{1j} B_j^{l+1}, \tag{13}$$

where the coefficients c_{1j} can be obtained from the subdivision matrix $\bar{A}$, and are equal to

$$c_{1j} = \left\{0.656826, \frac{3}{8}, \frac{3}{8}, \frac{3}{8}, \frac{3}{8}, \frac{3}{8}, \frac{3}{8}, \frac{3}{8}, \frac{1}{8}, \frac{1}{16}, \frac{1}{8}, \frac{1}{16}, \frac{1}{8}, \frac{1}{16}, \frac{1}{8}, \frac{1}{16}, \frac{1}{8}, \frac{1}{16}, \frac{1}{8}, \frac{1}{16}, \frac{1}{8}, \frac{1}{16}\right\}.$$

We take B_2^l (Case 1) as an example to show how to truncate B_1^l, see Fig. 5(f) for a reference. Suppose only B_2^l is refined. First, we check all the children of B_1^l to find out those with support fully contained in Ω^{l+1} (the shaded region), and mark them in blue dots. The remaining children are marked in green. All the children basis functions associated with these blue dots should be discarded in constructing $\text{trunc}^{l+1} B_1^l$. We then define an index set I^{l+1} to include all the blue dots in Fig. 5(f); we have

$$\begin{aligned}
&\textit{Case } 1 : I^{l+1} = \{1, 2, 3, 8, 9, 10, 11, 12, 22\};\\
&\textit{Case } 2 : I^{l+1} = \{1, 4, 5, 6, 14, 15, 16, 17, 18\}\\
&\textit{Case } 3 : I^{l+1} = \{20, 21, 22\}.
\end{aligned}$$

The truncated basis function $\text{trunc}^{l+1} B_1^l$ is then derived in each case by setting $c_{1j} = 0$ in Eq. (13) if $j \in I^{l+1}$.

The truncated basis functions associated with a valence-7 extraordinary node corresponding to Case 1, 2, 3 are plotted in Fig. 6(b)–(d) and compared with non-truncated ones in Fig. 6(a). From Fig. 4(f)–(h), we can observe that we need to discard nine, nine and three basis functions (blue dots) for Cases 1, 2 and 3, respectively. As shown in Fig. 6(b)–(d), the more basis functions discarded in building the truncated basis function, the more truncation. We do not need to truncate basis functions beyond a two-ring neighborhood (shaded domain) of $\mathcal{B}_r^l$.

Please cite this article in press as: H. Kang, et al., Truncated Hierarchical Loop Subdivision Surfaces and application in isogeometric analysis, Computers and Mathematics with Applications (2016), http://dx.doi.org/10.1016/j.camwa.2016.06.045

3.4. Control vertices update

During the construction from level l to $l+1$, we perform truncation on the basis set $\mathcal{H}^l \setminus \mathcal{H}_r^l$, resulting the first part $\mathcal{H}_A^{l+1}$, and refinement on the basis set $\mathcal{H}_r^l$, resulting the second part $\mathcal{H}_B^{l+1}$. Suppose the THLSS defined at level l is $S^l = \sum_{\tau\in\mathcal{H}^l} P_\tau^l \tau$, where $P_\tau \in \mathbb{R}^3$ are the position of vertices in the control mesh. First we consider the control vertices of the first part $\mathcal{H}_A^{l+1}$. Notice that a truncated basis function is actually the same as the original basis function over the active elements $\mathcal{F}_a^l$ at level l. And only the basis functions $\beta \in \mathcal{H}_A^{l+1}$ have support on $\mathcal{F}_a^l$. So in order to preserve geometry we choose the control vertices $P_\beta^{l+1} = P_\tau^l$, for $\beta \in \mathcal{H}_A^{l+1}$, $\beta = \text{trunc}^{l+1}\tau$, $\tau \in \mathcal{H}^l \setminus \mathcal{H}_r^l$.

Next we consider the control vertices of the second part $\mathcal{H}_B^{l+1}$. We take Fig. 2(c) as an example to show how to update control vertices. The $N+6$ control vertices of the shaded patch in Fig. 2(c) are locally labeled and have the matrix form $P_e^l = [P_{e,1}^l, \ldots, P_{e,N+6}^l]$. Fig. 2(d) shows the $N+12$ control vertices $P_e^{l+1} = [P_{e,1}^{l+1}, \ldots, P_{e,N+12}^{l+1}]^T$. We have

$$P_e^{l+1} = \overline{A} P_e^l, \tag{14}$$

where $\overline{A}$ is the subdivision matrix.

Then the THLSS at level $l+1$ is updated as

$$S^{l+1} = \sum_{\tau\in\mathcal{H}^l\setminus\mathcal{H}_r^l} P_\tau^l \text{trunc}^{l+1}\tau + \sum_{\beta\in\mathcal{H}_B^{l+1}} P_\beta^{l+1}\beta.$$

In Proposition 1, we will prove the above method of control vertices update preserves geometry, which means $S^{l+1} = S^l$.

3.5. Properties

Now we discuss some properties of THLSS basis functions constructed in Section 3.2.

Theorem 1. *The THLSS basis functions $\mathcal{H}$ constructed in Definition 2 have the following properties:*

1. *Nonnegativity: $\beta \geq 0, \forall \beta \in \mathcal{H}$.*
2. *Compact support: $\beta \in \mathcal{H}$ has compact support.*
3. *Partition of unity: $\sum_{\beta\in\mathcal{H}} \beta \equiv 1$.*
4. *Linear independency: the functions in $\mathcal{H}$ are linearly independent.*

Proof. The first two properties are obvious. The partition of a unity, linear independency and nested property are proved in Lemmas 1 and 2. □

Lemma 1. *The functions in $\mathcal{H}$ form a partition of unity:*

$$\sum_{\beta\in\mathcal{H}^l} \beta = 1 \quad \text{on } \Omega^0,\ l = 0, 1, \ldots, L.$$

Proof. We recall that Loop basis functions $\mathcal{B}^l$ form a partition of unity

$$\sum_{\beta\in\mathcal{B}^l} \beta = 1 \quad \text{on } \Omega^0,\ l = 0, 1, \ldots, L. \tag{15}$$

The partition of unity can be shown by induction on the hierarchical level l. The base case simply follows from (15) with $l = 0$. The inductive step

$$\sum_{\beta\in\mathcal{H}^l} \beta = 1 \Rightarrow \sum_{\beta\in\mathcal{H}^{l+1}} \beta = 1$$

can be proved by re-arranging the sums as follows.

$$\begin{aligned}
1 &= \sum_{\beta\in\mathcal{H}^l} \beta = \sum_{\beta\in\mathcal{H}^l} \sum_{\tau\in\mathcal{B}^{l+1}} c_\tau^{l+1}(\beta)\tau \\
&= \sum_{\beta\in\mathcal{H}^l} \left(\sum_{\substack{\tau\in\mathcal{B}^{l+1} \\ \text{supp}\,\tau\subseteq\Omega^{k+1}}} c_\tau^{l+1}(\beta)\tau + \sum_{\substack{\tau\in\mathcal{B}^{l+1} \\ \text{supp}\,\tau\not\subseteq\Omega^{k+1}}} c_\tau^{l+1}(\beta)\tau \right) \\
&= \sum_{\beta\in\mathcal{H}^l} \left(\sum_{\substack{\tau\in\mathcal{B}^{l+1} \\ \text{supp}\,\tau\not\subseteq\Omega^{k+1}}} c_\tau^{l+1}(\beta)\tau \right) + \sum_{\substack{\tau\in\mathcal{B}^{l+1} \\ \text{supp}\,\tau\subseteq\Omega^{k+1}}} \left(\sum_{\beta\in\mathcal{H}^l} c_\tau^{l+1}(\beta) \right) \tau.
\end{aligned} \tag{16}$$

Please cite this article in press as: H. Kang, et al., Truncated Hierarchical Loop Subdivision Surfaces and application in isogeometric analysis, Computers and Mathematics with Applications (2016), http://dx.doi.org/10.1016/j.camwa.2016.06.045

The term in brackets in the first sum of the last equation is just $\text{trunc}^{l+1}\tau$. Moreover, by (15) and by swapping the order of sums in the first line of (16), we have

$$\sum_{\beta\in\mathcal{B}^{l+1}}\beta = 1 \quad \text{as well as } 1 = \sum_{\tau\in\mathcal{B}^{l+1}}\left(\sum_{\beta\in\mathcal{H}^l} c_\tau^{l+1}(\beta)\right)\tau.$$

By comparing coefficients and by the linear independence of the Loop basis functions we can conclude that

$$\sum_{\beta\in\mathcal{H}^l} c_\tau^{l+1}(\beta) = 1$$

for all $\tau \in \mathcal{B}^{l+1}$ and in particular for each τ such that $supp\,\tau \subseteq \Omega^{k+1}$. From (16), we then obtain

$$1 = \sum_{\beta\in\mathcal{H}^l}\text{trunc}^{l+1}\beta + \sum_{\substack{\tau\in\mathcal{B}^{l+1}\\ supp\,\tau\subseteq\Omega^{k+1}}}\tau = \sum_{\tau\in\mathcal{H}_A^{l+1}}\tau + \sum_{\tau\in\mathcal{H}_B^{l+1}}\tau = \sum_{\tau\in\mathcal{H}^{l+1}}\tau. \quad \square$$

Lemma 2. *The functions in $\mathcal{H}$ are linearly independent.*

Proof. We use recursion to prove this lemma. It is easy to verify that the basis functions associated with M_0 and the domain Ω^0 are linearly independent, since Loop basis functions are linearly independent. Then we assume that the functions in $\mathcal{H}^i$ hold for levels up to l, which implies

$$\sum_{\beta\in\mathcal{H}^l} c_\beta\,\beta = 0 \Leftrightarrow c_\beta = 0, \quad \text{for all } \beta\in\mathcal{H}^l. \tag{17}$$

For the level $l+1$ case, suppose $\sum_{\beta\in\mathcal{H}^{l+1}} c_\beta\,\beta = 0$. According to Definition 2, $\mathcal{H}^{l+1} = \mathcal{H}_A^{l+1}\cup\mathcal{H}_B^{l+1}$, we have

$$\sum_{\beta\in\mathcal{H}_A^{l+1}} c_\beta\,\beta + \sum_{\beta\in\mathcal{H}_B^{l+1}} c_\beta\,\beta = 0. \tag{18}$$

Notice that the basis functions $\beta \in \mathcal{H}_B^{l+1}$ form the refinement domain Ω^{l+1}, and vanish on the domain $\Omega^0\setminus\Omega^{l+1}$. Hence on $\Omega^0\setminus\Omega^{l+1}$, Eq. (18) is simplified as

$$\sum_{\beta\in\mathcal{H}_A^{l+1}} c_\beta\,\beta = 0. \tag{19}$$

According to (10), $\text{trunc}^{l+1}\tau$ is different from τ only in the refined domain Ω^{l+1} and stays the same in $\Omega^0\setminus\Omega^{l+1}$, that is, $\beta = \text{trunc}^{l+1}\tau = \tau$ on $\Omega^0\setminus\Omega^{l+1}$, for any $\beta\in\mathcal{H}_A^{l+1}$ and $\tau\in\mathcal{H}^l\setminus\mathcal{H}_r^l$. By plugging it into (19), we obtain

$$\sum_{\beta\in\mathcal{H}_A^{l+1},\beta=\text{trunc}^{l+1}\tau,\tau\in\mathcal{H}^l\setminus\mathcal{H}_r} c_\beta\,\tau = 0.$$

In view of the assumption (17), therefore we have $c_\beta = 0$, for $\beta\in\mathcal{H}_A^{l+1}$.

On the domain Ω^{l+1}, Eq. (17) is left with one term $\sum_{\beta\in\mathcal{H}_B^{l+1}} c_\beta\,\beta = 0$ because of the truncation. Now the basis functions in $\mathcal{H}_B^{l+1}$ are standard Loop basis functions, so it is obvious that they are linearly independent. Thus we have $c_\beta = 0$, for $\beta\in\mathcal{H}_B^{l+1}$.

To sum up, we obtain $c_\beta = 0$, for $\beta\in\mathcal{H}^{l+1}$. The lemma is thus proved. $\square$

In the following, we theoretically study the geometry preservation of THLSS during h-refinement. THLSS construction starts with an initial Loop control mesh, denoted as $\mathbb{M}^0$. We define a sequence of Loop control meshes $\mathbb{M}^0,\ldots,\mathbb{M}^N$. $\mathbb{M}^l(0\le l\le N)$ is generated by l subdivisions of $\mathbb{M}^0$, and we denote the surface obtained from $\mathbb{M}^0$ as S_{loop}^0. Recall that evaluation of each $\mathbb{M}^l$ gives the same Loop surface, $S_{loop}^l = S_{loop}^0 (l=0,\ldots,N)$. After recursive construction, THLSS contains hierarchical levels up to N. Denote the surface evaluated with the THLSS basis functions as S_{THL}^N. Geometry preservation means that $S_{THL}^N = S_{loop}^N = S_{loop}^0$.

Proposition 1. *Given an initial valid Loop control mesh $\mathbb{M}^0$ and corresponding Loop surface S_{loop}^0, the THLSS surface obtained at levels up to N (S_{THL}^N) is exactly the same as the Loop surface evaluated after N subdivisions (S_{loop}^N). We have $S_{THL}^N = S_{loop}^N = S_{loop}^0$.*

Proof. We prove the proposition using recursion as in [19]. It is trivial to verify the initial step since no local refinement is performed. The THLSS basis functions and control mesh are exactly the same as the Loop basis functions and control mesh $\mathbb{M}^0$, we have $S_{THL}^0 = S_{loop}^0$.

Please cite this article in press as: H. Kang, et al., Truncated Hierarchical Loop Subdivision Surfaces and application in isogeometric analysis, Computers and Mathematics with Applications (2016), http://dx.doi.org/10.1016/j.camwa.2016.06.045

Now we assume that the proposition holds for Level $l, 0 \le l \le N$. This assumption means that the THLSS surface evaluated with levels up to l is the same as the Loop surface evaluated after l times of subdivision with the control mesh $\mathbb{M}^l$, we have

$$S^l_{THL} = S^l_{loop} = \sum_{i\in\bar{I}^l} P^l_i B^l_i = S^0_{loop}, \tag{20}$$

where $\bar{I}^l$ is an index set denoting all the Loop basis functions in $\mathbb{M}^l$ and B^l_i is the ith Loop basis function at level l. Among them, the THLSS basis functions $\mathcal{H}^l$ at level l are denoted as an index set $I^l \subseteq \bar{I}^l$. The set $\bar{I}^l \setminus I^l$ denotes the basis functions in $\mathbb{M}^l$ that are not the level l THLSS basis functions. Eq. (20) can be rewritten as

$$S^l_{THL} = \sum_{i\in I^l} P^l_i B^l_i + \sum_{i\in\bar{I}^l\setminus I^l} P^l_i B^l_i. \tag{21}$$

During the construction of THLSS from level l to $l+1$, we perform refinement on the to-be-refined basis functions ($\mathcal{H}^l_r$ with the index set R^l) and truncation on the to-be-truncated basis functions ($\mathcal{H}^l_t$ with the index set K^l). We have $R^l \subseteq I^l$ and $K^l \subseteq I^l$. Therefore in (21) only the first term on the right needs to be updated. Then, the THLLS surface is evaluated as

$$\begin{aligned} S^{l+1}_{THL} &= \sum_{j\in I^{l+1}} P^{l+1}_j B^{l+1}_j + \sum_{i\in K^l} P^l_i trun^{l+1} B^l_i + \sum_{i\in I^l\setminus(K^l\cup R^l)} P^l_i B^l_i + \sum_{i\in\bar{I}^l\setminus I^l} P^l_i B^l_i, \\ &= \sum_{j\in I^{l+1}} P^{l+1}_j B^{l+1}_j + \sum_{i\in K^l} P^l_i trun^{l+1} B^l_i + \sum_{i\in\bar{I}^l\setminus(K^l\cup R^l)} P^l_i B^l_i, \end{aligned} \tag{22}$$

where I^{l+1} is the index set of all the THLSS basis functions at level $l+1$ (denoted as $\mathcal{H}^{l+1}_a$).

In the following, we check each term in (22). In the first term on the right, P^{l+1}_j at level $l+1$ can be derived from P^l_i via the subdivision matrix $\bar{A}$ in (14) and we have

$$P^{l+1}_j = \sum_{i\in\bar{I}^l} c_{ij} P^l_i \quad \text{for } j \in I^{l+1}. \tag{23}$$

Note that (23), $c_{ij} = 0$ if $i \in \bar{I}^l \setminus (K^l \cup R^l)$. We now check the children term of (22) on the right. According to (9), we can represent B^l_i using their children,

$$B^l_i = \sum_{j\in\bar{I}^{l+1}\setminus I^{l+1}} c_{ij} B^{l+1}_j \quad \text{for } i \in \bar{I}^l \setminus (K^l \cup R^l). \tag{24}$$

This implies that for the basis function B^l_i with $i \in \bar{I}^l \setminus (K^l \cup R^l)$, its children are not active THLSS basis functions at level $l+1$. The other B^l_i with $i \in K^l \cup R^l$ can be either to-be-refined or truncated basis functions. The truncated basis functions in the second term of (22) can be expressed as

$$\text{trunc}^{l+1} B^l_i = \sum_{j\in\bar{I}^{l+1}\setminus I^{l+1}} c_{ij} B^{l+1}_j \quad \text{for } i \in K^l. \tag{25}$$

Note that in (25) also holds when $i \in R^l$ because all the children $B^{l+1}_j = 0$ when $j \in \bar{I}^{l+1} \setminus I^{l+1}$. Therefore in (25), we replace $i \in K^l$ with $i \in K^l \cup R^l$. Then, we plug Eqs. (23)–(25) into (22) and obtain

$$\begin{aligned} S^{l+1}_{THL} &= \sum_{j\in I^{l+1}} \sum_{i\in\bar{I}^l} c_{ij} P^l_i B^{l+1}_j + \sum_{i\in K^l\cup R^l} \sum_{j\in\bar{I}^{l+1}\setminus I^{l+1}} P^l_i c_{ij} B^{l+1}_j + \sum_{i\in\bar{I}^l\setminus(K^l\cup R^l)} \sum_{j\in\bar{I}^{l+1}\setminus I^{l+1}} c_{ij} B^{l+1}_j P^l_i \\ &= \sum_{j\in I^{l+1}} \sum_{i\in\bar{I}^l} c_{ij} P^l_i B^{l+1}_j + \sum_{i\in\bar{I}^l} \sum_{j\in\bar{I}^{l+1}\setminus I^{l+1}} c_{ij} B^{l+1}_j P^l_i \\ &= \sum_{j\in\bar{I}^{l+1}} \sum_{i\in\bar{I}^l} c_{ij} P^l_i B^{l+1}_j \\ &= \sum_{j\in\bar{I}^{l+1}} P^{l+1}_j B^{l+1}_j = S^{l+1}_{loop} = S^0_{loop}. \end{aligned} \tag{26}$$

Eq. (26) indicates that the proposition also holds for level $l+1$. Therefore, it holds for all the level up to N, and we have $S^N_{THL} = S^N_{loop} = S^0_{loop}$. □

Please cite this article in press as: H. Kang, et al., Truncated Hierarchical Loop Subdivision Surfaces and application in isogeometric analysis, Computers and Mathematics with Applications (2016), http://dx.doi.org/10.1016/j.camwa.2016.06.045

Based on this proposition, we can conclude that THLSS preserves geometry and a THLSS surface inherits the continuity of the Loop subdivision surface, namely C^2-continuity everywhere except G^1 around extraordinary nodes.

4. Numerical experiment

In this section, we perform adaptive isogeometric analysis using THLSS basis functions. First we briefly introduce the model problem, then give several numerical examples to demonstrate the approximation power of THLSS on solving PDEs. We compare the error estimate from FEM with linear elements, Loop basis functions and THLSS in the first three examples.

4.1. PDE model problem

The model problem we considered is a Poisson equation defined on a manifold surface $S \subseteq \mathbb{R}^3$ with the Dirichlet boundary condition, which is defined as follows:

$$\begin{aligned} -\Delta_S u &= f \quad \text{on } S \\ u &= g \quad \text{on } \Gamma \end{aligned} \tag{27}$$

where $\Gamma = \partial S$ and $f \in L^2(\Omega)$.

Suppose the physical domain S is parameterized by THLSS, defined as

$$G(v, w) = (x, y, z)^T = \sum_{i=1}^{m} P_i \beta_i(v, w), \tag{28}$$

where $(v, w) \in \Omega$, $(x, y, z) \in S$, $P_i \in \mathbb{R}^3$ and β_i, $i = 1, 2, \ldots, m$ are THLSS basis functions. The numerical solution u^h is expressed as

$$u^h(x, y, z) = \sum_{i=1}^{m} c_i\, \beta_i \circ G^{-1} = \sum_{i=1}^{m} c_i \hat{G}_i(x, y, z), \tag{29}$$

where $c_i \in \mathbb{R}$, $i = 1, 2, \ldots, n$ $(n < m)$ are fixed coefficients and $c_i \in \mathbb{R}$, $i = n+1, 2, \ldots, m$ are unknown. Then the IGA approximation of problem (27) is to find u^h such that for all $v \in V^h$

$$a(u^h, v) = F(v), \tag{30}$$

where $a(u, v) = \int_S \nabla u \cdot \nabla v \, ds$, $F(v) = \int_S f v \, ds$ and V^h is a finite dimensional subspace defined as

$$V^h = span\{\hat{G}_i(x, y, z) |\, \hat{G}_i(x, y, z) = \beta_i \circ G^{-1}, i = n+1, \ldots, m\}.$$

Problem (30) is equivalent to solving the following linear system

$$LC = R, \tag{31}$$

where L is a $(m-n) \times (m-n)$ matrix with the element $L_{ij} = a(\hat{G}_i, \hat{G}_j)$, R is a $(m-n)$-dimensional column vector with the element $R_j = F(\hat{G}_j) - a(\sum_{i=1}^{n} c_i \hat{G}_i, \hat{G}_j)$, $j = n+1, \ldots, m$, and $C = (c_{n+1}, \ldots, c_m)^T$.

The L_2-norm error between the exact solution u and the numerical solution u_h defined as in (29) is defined as $e_h = (\int_S |u_h - u|^2 ds)^{1/2}$. Q. Pan et al. [28] gave a L_2-norm error estimate of IGA method with Loop subdivision

$$\|u - u_h\|_{L_2(S)} \leq Ch^2, \tag{32}$$

where C is a constant independent of u and h, and h indicates the global mesh size $h = \max\{h_T | T \in \mathbb{M}^l\}$, h_T is the diameter of the element T. The convergence rate is defined by $CR = \frac{e_h^l}{e_h^{l+1}}$, where e_h^l is the L_2-norm error at level l. If uniform refinement is performed, according to (32), the optimal convergence rate of IGA method with Loop subdivision should be four.

From (32), we see that the IGA method with Loop subdivision has the same order of the L_2-norm error estimate as that of the FEA method with linear elements, which is $O(h^2)$. Additionally, FEA with linear elements is the most common used in industry. So in the following numerical examples, we always compare the error estimate and convergence rate at each level from THLSS, Loop subdivision and FEA with linear elements. We start from the same initial mesh. Uniform refinement is performed on Loop subdivision and FEA with linear elements, while adaptive refinement is performed on THLSS. Four Gauss points integral scheme is used for integral over elements.

Please cite this article in press as: H. Kang, et al., Truncated Hierarchical Loop Subdivision Surfaces and application in isogeometric analysis, Computers and Mathematics with Applications (2016), http://dx.doi.org/10.1016/j.camwa.2016.06.045

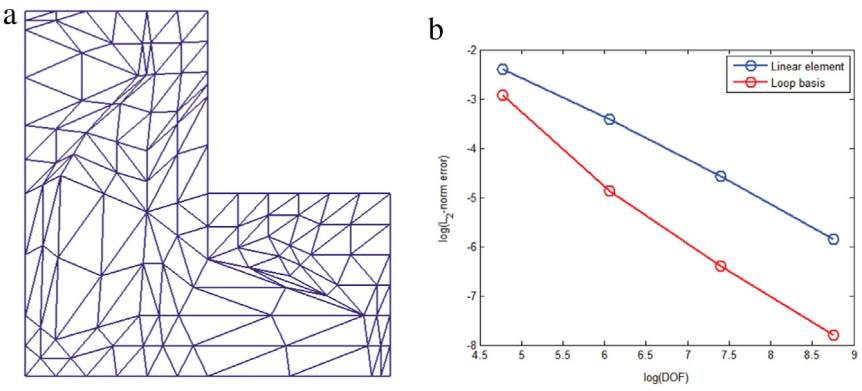


Fig. 7. (a) L-shape domain. (b) Convergence rate plot.

Table 1
Comparison of linear elements and Loop subdivision.

Vertices	e_h (linear)	CR	e_h (loop)	CR
119	0.09127		5.3965e−2	
429	0.03318	2.7510	7.727e−3	6.9840
1625	0.01031	3.2182	1.6699e−3	4.6272
6321	0.002861	3.6027	4.0729e−4	4.1000

4.2. Convergence rate test

In the following, we give an example to compare the convergence rates of linear element space and THLSS (Loop) spline space. We consider the Poisson equation on a planar L-shape domain as shown in Fig. 7(a). The exact solution u is chosen as $u(x, y) = \sin(\pi x)\sin(\pi y)$ and the right term f is derived by the problem (27).

Fig. 7(b) shows the convergence plot of Loop subdivision and FEA with linear elements. As in the planar domain case, the approximation error from Loop subdivision is much smaller than that from FEM with linear elements with the same degree of freedom. The convergence rate of Loop subdivision is faster than FEA with linear elements. Table 1 shows the results of L_2-norm error estimate.

4.3. Adaptive refinement with THLSS

Now, we perform the framework for adaptive refinement with THLSS. We start from an initial valid control mesh M^0. The Poisson equation is defined on the limit Loop surface S of M^0. Adaptive refinement is performed. Since THLSS preserve geometry, the domain S will not change during the refinement. At each refinement step the basis function with the largest error is identified to be refined. Its two-ring neighboring elements are then identified and refined. Here we use the basis-wise error $Err(\beta_i^l)$ instead of element-wise error, which is defined as,

$$Err(\beta^l) = \sum_{\Omega_j^l \subseteq \text{supp}\,\beta^l} Err(\Omega_j^l), \tag{33}$$

where β^l is a *THLSS* basis function at level l and $Err(\Omega_j^l)$ is the error on a element which is obtained with the aid of so-called bubble functions [8].

We consider the Poisson equation on a planar L-shape domain as shown in Fig. 7(a). The exact solution u is chosen as $u(x, y) = (x^3-x)(y^3-y)/((x+0.5)^2+(y+0.5)^2+0.001)$ and the right term f is derived by the problem (27). The considered L-shape domain is a planar domain with boundary. The Loop basis functions defined on an open surface should be carefully treated because of the boundary. We omit the detailed construction and refer the reader to [28] for a reference. We start from an initial valid control mesh shown in Fig. 8(a), which clearly has a lot of sharp triangles which are not welcomed in FEA. The valence of the control vertices varies from 4 to 9. The exact solution for this example is also a smooth function, but it decays quickly away from the peak, thus it has a large gradient near $(-0.5, -0.5)$. We approximate it by solving problem (27) based on FEA with linear elements, Loop subdivision and THLSS. Fig. 8(b) and (d) show the three adaptive refinement levels of THLSS, where the red triangles form the refinement area. It is worth noting that the numerical solutions quickly capture the peak as the refinement proceeds. The number of basis functions and corresponding error at four levels are shown in Table 2. we see that the error range is decreased quickly as the mesh refinement procedure going on.

The L_2-norm error estimate statistics from FEA with linear elements, Loop subdivision and THLSS is shown in Table 2. Fig. 9 shows the plot of L_2-norm error with respect to the degree of freedom. For this example, the error from IGA with

Please cite this article in press as: H. Kang, et al., Truncated Hierarchical Loop Subdivision Surfaces and application in isogeometric analysis, Computers and Mathematics with Applications (2016), http://dx.doi.org/10.1016/j.camwa.2016.06.045

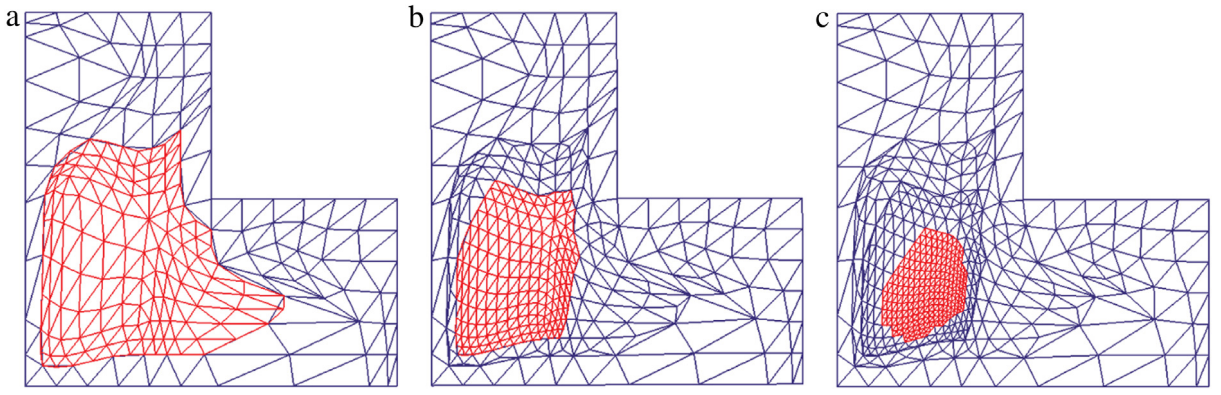


Fig. 8. Adaptively refined control meshes when solving Poisson equation over a planar L-shape domain. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

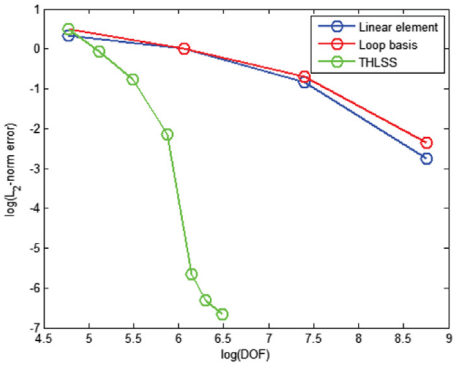


Fig. 9. Convergence rate plot for THISS.

Table 2
Error comparison of three kinds of spaces.

Level	Vertices	e_h (linear)	CR (linear)	e_h (loop)	CR (loop)	DOF (THLSS)	e_h (THLSS)
0	119	1.40375		1.65121		119	1.65121
1	429	1.00925	1.3909	1.00325	1.6459	167	0.89298
2	1625	0.438298	2.3027	0.498471	2.0127	242	0.42071
3	6321	0.0635495	6.8970	0.0958274	5.2018	464	3.50611e−3

Loop subdivision is similar with that from the FEM with linear elements at the same level and the two convergence plot almost overlap. But the error from IGA with THLSS is much smaller than that of both linear elements and Loop subdivision. Furthermore, to achieve the same magnitude of error estimate, THLSS needs much fewer basis functions. From Fig. 9, THLSS have a much faster convergence rate owing to the adaptive refinement. This example indicates that THLSS can improve the numerical solution significantly when the given mesh is not good enough or with extraordinary vertices.

4.4. Complex models

Finally the Poisson problem on complex geometries without boundary is solved. For convenience, we restrict ourself to the case $f = 0$ of problem (27), which is called Laplace equation. We imposed Dirichlet boundary conditions on several patches to obtain a non-zero numerical solution. We consider the Poisson equation with right term $f = 0$ (Laplace equation) on three models: kitten model (genus-1 model) (Fig. 10), hand model (Fig. 11) and bunny model (Fig. 12).

Table 3 summarizes the statistics of each model. In Figs. 10–12(a), the areas where the Dirichlet boundary conditions are applied are indicated by the arrows. Figs. 10–12(b) display the simulation results on the initial valid control meshes, whereas Figs. 10–12(c) show the results after global refinement steps. Figs. 10–12(d) shows the results after local refinement. Invalid elements exist in these three models and the input meshes need to be preprocessed. From Figs. 10–12, we can observe that the proposed THLSS method works robustly for complex geometries, which provides a potential wide application in isogeometric analysis with extraordinary nodes involved.

Please cite this article in press as: H. Kang, et al., Truncated Hierarchical Loop Subdivision Surfaces and application in isogeometric analysis, Computers and Mathematics with Applications (2016), http://dx.doi.org/10.1016/j.camwa.2016.06.045

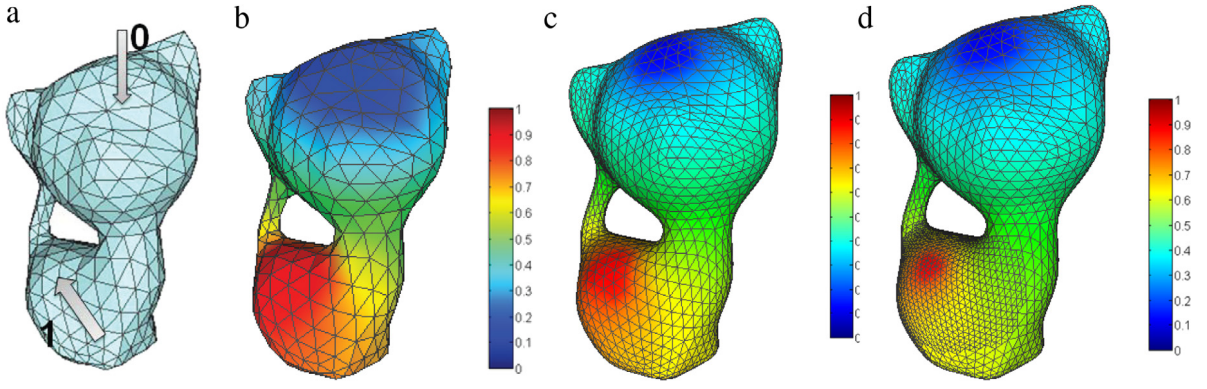


Fig. 10. Solving Laplace equation over a Kitten model. (a) Input quadrilateral mesh with boundary conditions; and (b, c, d) simulations results on the initial valid control mesh and the control mesh after three levels refinement.

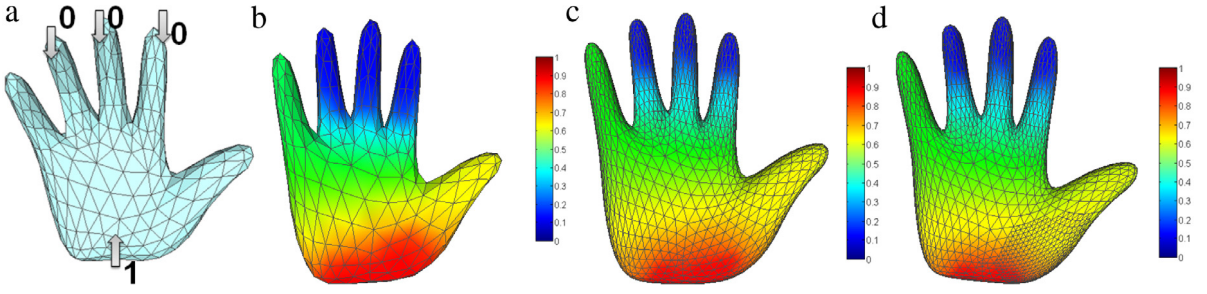


Fig. 11. Solving Laplace equation over a Hand model. (a) Input quadrilateral mesh with boundary conditions; and (b, c, d) simulations results on the initial valid control mesh and the control mesh after three levels refinement.

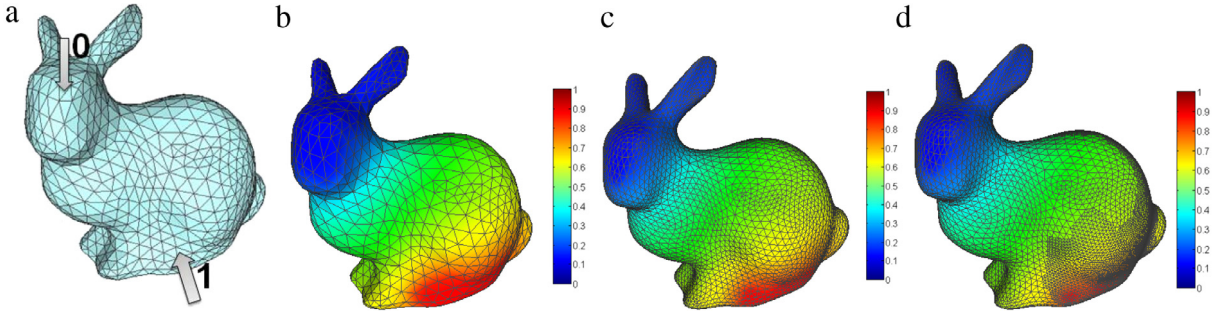


Fig. 12. Solving Laplace equation over a Bunny model. (a) Input quadrilateral mesh with boundary conditions; and (b, c, d) simulations results on the initial valid control mesh and the control mesh after three levels refinement.

Table 3
Statistics of all the tested models.

Models	Nodes	Elements	Extra	Levels
Kitten	400	800	55	3
Hand	402	800	62	3
Bunny	1002	2000	166	3

5. Conclusion and future work

In this paper we propose Truncated Hierarchical Loop Subdivision Surface (THLSS), which generalizes truncated hierarchical B-splines to triangular mesh of arbitrary topology. THLSS basis functions are linearly independent, form a partition of unity and support local refinement. THLSS also preserves geometry during adaptive h-refinement and thus inherits the surface continuity of Loop subdivision surface. Adaptive isogeometric analysis is performed with THLSS basis functions on several complex models with extraordinary vertices. From the numerical results, it shows IGA with THLSS has a faster convergence rate than both FEA using linear elements and Loop basis functions owing to the local refinement.

Please cite this article in press as: H. Kang, et al., Truncated Hierarchical Loop Subdivision Surfaces and application in isogeometric analysis, Computers and Mathematics with Applications (2016), http://dx.doi.org/10.1016/j.camwa.2016.06.045

Furthermore, THLSS can significantly reduce the degree of freedom when obtaining the same magnitude of error estimate. In the future, we will address the numerical integral scheme over irregular patch and extend the method of THLSS construction to tetrahedral elements.

References

[1] T.J.R. Hughes, J.A. Cottrell, Y. Bazilevs, Isogeometric analysis: CAD, finite elements, NURBS, exact geometry, and mesh refinement, Comput. Methods Appl. Mech. Engrg. 194 (2005) 4135–4195.
[2] J.A. Cottrell, T.J.R. Hughes, Y. Bazilevs, Isogeometric Analysis: Toward Integration of CAD and FEA, Wiley, Chichester, 2009.
[3] J.A. Cottrell, A. Reali, Y. Bazilevs, T.J.R. Hughes, Isogeometric analysis of structural vibrations, Comput. Methods Appl. Mech. Engrg. 195 (2006) 5257–5296.
[4] J.A. Evans, Y. Bazilevs, I. Babuska, T.J.R. Hughes, n-widths, sup-infs, and optimality ratios for the k-version of the isogeometric finite element method, Comput. Methods Appl. Mech. Engrg. 198 (21–26) (2009) 1726–1741.
[5] S. Lipton, J.A. Evans, Y. Bazilevs, T. Elguedj, T.J.R. Hughes, Robustness of isogeometric structural discretizations under severe mesh distortion, Comput. Methods Appl. Mech. Engrg. 199 (5–8) (2010) 357–373.
[6] D. Forsey, R. Bartels, Hierarchical B-spline refinement, Comput. Graph. 22 (1988) 205–212.
[7] A.-V. Vuong, C. Giannelli, B. Jüttler, B. Simeon, A hierarchical approach to adaptive local refinement in isogeometric analysis, Comput. Methods Appl. Mech. Engrg. 200 (2011) 3554–3567.
[8] C. Giannelli, B. Jüttler, H. Speleers, THB-splines: The truncated basis for hierarchical splines, Comput. Aided Geom. Design 29 (7) (2012) 485–498.
[9] T.W. Sederberg, J. Zheng, A. Bakenov, A. Nasri, T-splines and T-NURCCSs, ACM Trans. Graph. 22 (3) (2003) 477–484.
[10] T.W. Sederberg, D.L. Cardon, G.T. Finnigan, N.S. North, J. Zheng, T. Lyche, T-spline simplification and local refinement, ACM Trans. Graph. 23 (3) (2004) 276–283.
[11] X. Li, J. Zheng, T.W. Sederberg, T.J.R. Hughes, M.A. Scott, On the linear independence of T-splines blending functions, Comput. Aided Geom. Design 29 (2012) 63–76.
[12] M.A. Scott, X. Li, T.W. Sederberg, T.J.R. Hughes, Local refinement of analysis-suitable T-splines, Comput. Methods Appl. Mech. Engrg. 213–216 (2012) 206–222.
[13] J. Deng, F. Chen, Y. Feng, Dimensions of spline spaces over T-meshes, J. Comput. Appl. Math. 194 (2006) 267–283.
[14] J. Deng, F. Chen, X. Li, C. Hu, W. Tong, Z. Yang, Y. Feng, Polynomial splines over hierarchical T-meshes, Graph. Models 74 (2008) 76–86.
[15] Hongmei Kang, Falai Chen, Jiansong Deng, A new basis for PHT-splines, Graph. Models 82 (2015) 149–159.
[16] T. Dokken, Tom Lyche, Kjell Fredrik Pettersen, Polynomial splines over locally refined box-partitions, Comput. Aided Geom. Design 30 (3) (2013) 331–356.
[17] K.A. Johannessen, T. Kvamsdal, T. Dokken, Isogeometric analysis using {LR} b-splines, Comput. Methods Appl. Mech. Engrg. 269 (2014) 471–514.
[18] Hongmei Kang, Falai Chen, Jiansong Deng, Modified T-splines, Comput. Aided Geom. Design 30 (2013) 827–843.
[19] Xiaodong Wei, Yongjie Zhang, Thomas J.R. Hughes, Michael A. Scott, Truncated hierarchical Catmull-Clark subdivision with local refinement, Comput. Methods Appl. Mech. Engrg. 291 (2015) 1–20.
[20] X. Wei, Y. Zhang, T.J.R. Hughes, M.A. Scott, Extended truncated hierarchical Catmull-Clark subdivision, Comput. Methods Appl. Mech. Engrg. 299 (2016) 316–336.
[21] H. Kang, F. Chen, J. Deng, Hierarchical B-splines on regular triangular partitions, Graph. Models 76 (2014) 289–300.
[22] U. Zorea, B. Juttler, Adaptively refined multilevel spline spaces from generating systems, Comput. Aided Geom. Design 31 (2014) 545–566.
[23] H. Speleers, P. Dierckx, S. Vandewalle, Quasi-hierarchical powell-sabin b-splines, Comput. Aided Geom. Design 26 (2009) 174–191.
[24] H. Speleers, C. Manni, F. Pelosi, M.L. Sampoli, Isogeometric analysis with powell-sabin splines for advection-diffusion- reaction problems, Comput. Methods Appl. Mech. Engrg. 221–222 (2012) 132–148.
[25] C.T. Loop, Smooth subdivision surfaces based on trianglesgg (M.S. Thesis), Department of Mathematics, University of Utah, 1987.
[26] J. Stam, Fast evaluation of Loop triangular subdivision surfaces at arbitrary parameter values, in: SIGGRAPH 98 Proceedings, 1998, pp. 395–404, CD-ROM Supplement.
[27] F. Cirak, M. Ortiz, P. Schroder, Subdivision surfaces: a new paradigm for thin-shell finite-element analysis, Internat. J. Numer. Methods Engrg. 47 (2000) 2039–2072.
[28] Qing Pan, Guoliang Xu, Gang Xu, Yongjie Zhang, Isogeometric analysis based on extended Loop's subdivision, J. Comput. Phys. 299 (2015) 731–746.
[29] J. Peters, X. Wu, On the local linear independence of generalized subdivision functions, SIAM J. Numer. Anal. 44 (6) (2006) 2389–2407.

Please cite this article in press as: H. Kang, et al., Truncated Hierarchical Loop Subdivision Surfaces and application in isogeometric analysis, Computers and Mathematics with Applications (2016), http://dx.doi.org/10.1016/j.camwa.2016.06.045