A Surface Subdivision Scheme Based on Four-Directional S_3^1 Non-Box Splines

Z. Huang[1,2,3]

[1]University of Science and Technology of China, China
[2]Anhui Province Key Laboratory of Software in Computing and Communication, China
[3]USTC-Deqing Alpha Innovation Research Institute, China

Abstract

In this paper, we propose a novel surface subdivision scheme called non-box subdivision, which is generalized from four-directional S_3^1 non-box splines. The resulting subdivision surfaces achieve C^1 continuity with the convex hull property. This scheme can be regarded as either a four-directional subdivision or a special quadrilateral subdivision. When used as a quadrilateral subdivision, the proposed scheme can control the shape of the limit surface more flexibly than traditional schemes due to the natural introduction of auxiliary face control vertices.

CCS Concepts
• *Computing methodologies* → *Shape modeling;*

1. Introduction

Subdivision surfaces are obtained as the limit of a recursive refinement process applied to polyhedra. The power of this technique comes from its ability to model complicated shapes with much simpler polyhedra using a few subdivision rules.

Box splines [dBHR93] provide a natural framework for building subdivision surfaces. Most of the known spline-based subdivision schemes are derived from box splines. Tensor product biquadratic and bicubic B-splines lead to Doo-Sabin [DS78] and Catmull-Clark [CC78] subdivision, respectively. Three-directional quartic box spline gives raise to the Loop subdivision scheme [Loo87]. Midedge scheme [PR97, HW99] is derived from four-directional quadratic box splines, and 4-8 subdivision [VZ01] is generalized from four-directional sextic box splines. Except for box splines, there are relatively few schemes derived from other types of splines. Recently, Barendrecht et al. [BSK19] proposed a novel C^1 honeycomb subdivision scheme based on cubic half-box splines.

The above spline-based schemes are all derived from scalar-valued subdivision, whose underlying spline spaces are spanned by one single refinable spline. On the other hand, vector-valued subdivision schemes [MS98] are deduced from two or more refinable functions.

Conti and Jetter [CJ00] proposed S_1^0 and S_3^1 bivariate splines on the four-directional meshes, starting from two non-box spline generators. Here, S_1^0 and S_3^1 stand for piecewise linear splines and piecewise cubic C^1-splines, respectively. Based on their vector-valued subdivision scheme for four-directional S_3^1 splines, we de-

rive the subdivision rules for regular cases and extend them to polygonal meshes of arbitrary topology. The proposed non-box subdivision scheme has the following characteristics:

- *Extension from vector-valued subdivision*. The four-directional S_3^1 spline has two generating splines. Thus, its subdivision scheme is vector-valued.
- *Generalization of non-box spline subdivision*. Neither of the two generating functions for the four-directional S_3^1 splines is a box spline. We generalize the subdivision scheme for the S_3^1 non-box spline from regular four-directional meshes to arbitrary topology.
- C^1 *continuity*. The generated subdivision surfaces are C^1-continuous.
- *Convex hull property*. All subdivision weights are non-negative.
- *Flexible modeling ability*. It can be applied to arbitrary topological quasi-four-directional meshes and polygonal meshes with face control vertices (FCVs). By setting different FCVs, different limit surfaces can be produced from the same initial mesh.

The rest of this paper is organized as follows. The next section briefly reviews the four-directional S_3^1 non-box splines. Section 3 deduces the subdivision rules for regular four-directional meshes, then extend them to quasi-four-directional meshes and polygonal meshes with FCVs. In Section 4, we discuss the selection of the weights of the subdivision scheme. By analyzing the subdivision matrix and characteristic map, we verify that the scheme is C^1-continuous, and derive the formula of the limit position and tangent vectors. Section 5 briefly discusses the implementation of the algorithm, boundary rules and selection of FCVs. In Section 6, some

examples are given and compared with other subdivision schemes. Finally, we conclude the paper with some suggestions for future work.

2. Four-directional non-box splines

In this section, we briefly review vector-valued subdivision [MS98] and four-directional S_3^1 non-box splines [CJ00].

2.1. Vector-valued subdivision

A compactly supported continuous function vector $\Phi = \begin{pmatrix} \phi_1 \\ \phi_2 \end{pmatrix}$: $\mathbb{R}^2 \to \mathbb{R}^2$ is refinable with respect to the dilation matrix $\begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix}$, if it satisfies the following *matrix refinement equation*:

$$\Phi = \sum_{\alpha \in \mathbb{Z}^2} A_\alpha \Phi(2 \cdot - \alpha) \qquad (1)$$

Here, the *refinement matrix mask* $\mathbf{A} = (A_\alpha)_{\alpha \in \mathbb{Z}^2}$ is a bi-infinite matrix; and $A_\alpha, \alpha \in \mathbb{Z}^2$ are real (2×2)-matrices.

Denote $\ell(\mathbb{Z}^2)$ as the linear space formed by all infinite sequences on $\mathbb{Z}^2$, and $\ell^\infty(\mathbb{Z}^2)$ as the linear space of all bounded infinite sequences on $\mathbb{Z}^2$. The *vector-valued subdivision operator* associated with $\mathbf{A}$ is defined as

$$S_\mathbf{A} : (\ell(\mathbb{Z}^2))^2 \to (\ell(\mathbb{Z}^2))^2$$

$$(S_\mathbf{A}\Lambda)_\alpha := \sum_{\beta \in \mathbb{Z}^2} A_{\alpha - 2\beta}^T \Lambda_\beta, \quad \alpha \in \mathbb{Z}^2, \Lambda \in (\ell(\mathbb{Z}^2))^2 \qquad (2)$$

For a given initial vector sequence $\Lambda \in (\ell(\mathbb{Z}^2))^2$, the *vector-valued subdivision scheme* is defined as

$$\begin{aligned} \Lambda^0 &:= \Lambda, \qquad \Lambda \in (\ell(\mathbb{Z}^2))^2 \\ \Lambda^m &:= S_\mathbf{A}\Lambda^{m-1}, \quad m = 1, 2, \dots \end{aligned}$$

We say that the subdivision scheme *converges* with respect to $\Lambda \in (\ell(\mathbb{Z}^2))^2$, if there exists a continuous function $f_\Lambda : \mathbb{R}^2 \to \mathbb{R}$, such that

$$\lim_{m \to \infty} \left\| f_\Lambda \left(\frac{\cdot}{2^m} \right) \mathbf{e} - \Lambda^m \right\|_\infty = 0$$

Here, $\mathbf{e} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$, and $\| \cdot \|_\infty$ denotes the sup-norm of a vector sequence $\Lambda = \begin{pmatrix} \lambda^1 \\ \lambda^2 \end{pmatrix}$, i.e.

$$\left\| \begin{pmatrix} \lambda^1 \\ \lambda^2 \end{pmatrix} \right\|_\infty := \sum_{i=1}^2 \|\lambda^i\|_\infty := \sum_{i=1}^2 \sup_{\alpha \in \mathbb{Z}^2} \|\lambda_\alpha^i\|$$

2.2. Four-directional S_3^1 non-box splines

Denote C as the unit square $[0,1]^2$ and D as the rectangle on the uniform four-directional mesh with $(-1, 0), (0, -1), (1, 0), (0, 1)$ as vertices. ϕ_1 and ϕ_2 are two pyramidal hat functions with C and D as supports, respectively. That is, they are S_1^0 spline functions on the four-directional grid satisfying $\phi_1(1/2, 1/2) = \phi_2(0, 0) = 1$.

It is easy to see that the $\mathbb{Z}^2$-translates of ϕ_1 and ϕ_2 span the S_1^0 spline space on a regular four-direction grid [CJ00].

Let χ_C be the characteristic function of the square $C := [0,1]^2$, which is a piecewise constant function with C as support:

$$\chi_C(x, y) = \begin{cases} 1, & (x, y) \in C := [0,1]^2; \\ 0, & \text{otherwise} . \end{cases}$$

We convolve ϕ_1 and ϕ_2 with χ_C respectively to obtain two S_3^1 non-box spline functions:

$$\begin{aligned} \overline{\phi}_1 &= \phi_1 * \chi_C, \\ \overline{\phi}_2 &= \phi_2 * \chi_C. \end{aligned}$$

The S_3^1 non-box spline function vector $\overline{\Phi} = \begin{pmatrix} \overline{\phi}_1 \\ \overline{\phi}_2 \end{pmatrix}$ satisfies the matrix refinement equation (1), where the refinement matrix mask $\mathbf{A}$ is as follows [CJ00]:

$\mathbf{A} = (A_\alpha)_{\alpha \in \mathbb{Z}^2}$

$$= \begin{pmatrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \cdots\ 0 & 0 & 0 & 0 & 0 & 0\ \cdots \\ \cdots\ 0 & 0 & \begin{pmatrix} \frac{1}{8} & 0 \\ 0 & \frac{1}{8} \end{pmatrix} & \begin{pmatrix} \frac{1}{8} & 0 \\ \frac{1}{4} & \frac{1}{4} \end{pmatrix} & \begin{pmatrix} 0 & 0 \\ \frac{1}{4} & \frac{1}{8} \end{pmatrix} & 0\ \cdots \\ \cdots\ 0 & \begin{pmatrix} \frac{1}{8} & \frac{1}{8} \\ 0 & 0 \end{pmatrix} & \begin{pmatrix} \frac{1}{2} & \frac{1}{4} \\ 0 & \frac{1}{4} \end{pmatrix} & \begin{pmatrix} \frac{1}{2} & \frac{1}{8} \\ \frac{1}{4} & \frac{1}{2} \end{pmatrix} & \begin{pmatrix} \frac{1}{8} & 0 \\ \frac{1}{4} & \frac{1}{4} \end{pmatrix} & 0\ \cdots \\ \cdots\ 0 & \begin{pmatrix} \frac{1}{8} & \frac{1}{4} \\ 0 & 0 \end{pmatrix} & \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ 0 & \frac{1}{8} \end{pmatrix} & \begin{pmatrix} \frac{1}{2} & \frac{1}{4} \\ 0 & \frac{1}{4} \end{pmatrix} & \begin{pmatrix} \frac{1}{8} & 0 \\ 0 & \frac{1}{8} \end{pmatrix} & 0\ \cdots \\ \cdots\ 0 & \begin{pmatrix} 0 & \frac{1}{8} \\ 0 & 0 \end{pmatrix} & \begin{pmatrix} \frac{1}{8} & \frac{1}{4} \\ 0 & 0 \end{pmatrix} & \begin{pmatrix} \frac{1}{8} & \frac{1}{8} \\ 0 & 0 \end{pmatrix} & 0 & 0\ \cdots \\ \cdots\ 0 & 0 & 0 & 0 & 0 & 0\ \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix}$$

$$(3)$$

Here, $A_{(0,0)} = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ 0 & \frac{1}{8} \end{pmatrix}$, and we have the following convergence theorem [CJ00]:

Theorem 1. *The subdivision scheme associated with the matrix mask $\mathbf{A}$ is convergent for all $\Lambda = \begin{pmatrix} \lambda^1 \\ \lambda^2 \end{pmatrix} \in (\ell(\mathbb{Z}^2))^2$, and the limit function is given by the piecewise cubic four-directional C^1 spline*

$$f_\Lambda = \sum_{\alpha \in \mathbb{Z}^2} \lambda_\alpha^1 \overline{\phi}_1(\cdot - \alpha) + \sum_{\alpha \in \mathbb{Z}^2} \lambda_\alpha^2 \overline{\phi}_2(\cdot - \alpha).$$

3. Non-box subdivision scheme

In this section, we deduce a non-box subdivision scheme for regular four-directional meshes, and then extend this scheme to meshes of arbitrary topology.

3.1. Regular case

There are two types of vertices and two types of edges on a regular four-directional mesh. We call a vertex *type 1* if its valence is 8; and *type 2* if its valence is 4. An edge is called *type 1* if both of its endpoints have valence 8, i.e. both vertices are type 1. Otherwise,

an edge is called *type 2*, if the valences of its two endpoints are 4 and 8 respectively, that is, one is a type 1 vertex and the other is a type 2 vertex.
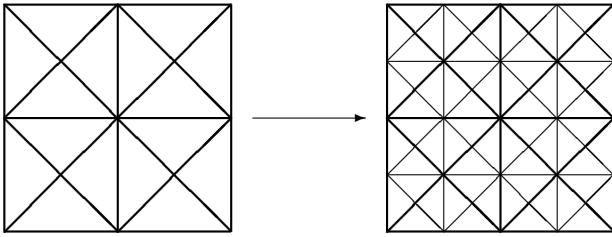

Figure 1: *1-4 split on regular four-directional meshes*

For regular four-directional meshes, *1-4 split* inserts a split vertex on each edge to subdivide each triangle into four sub-triangles as shown in Figure 1:

- On each type 1 edge, a new *type 1 E-vertex* is inserted, which is a type 1 vertex with valence 8 on the refined mesh.
- On each type 2 edge, a new *type 2 E-vertex* is added, which is a type 2 vertex of valence 4 on the refined mesh.
- Each type 1 vertex corresponds to a *type 1 V-vertex* on the refined mesh, which remains a type 1 vertex.
- Each type 2 vertex is associated with a *type 2 V-vertex* on the refined mesh, which is a type 1 vertex with valence 8.

From the matrix mask **A** in Eq. (3), we obtain four different subdivision stencils correspond to type 1 E-vertex, type 2 E-vertex, type 1 V-vertex, and type 2 V-vertex, respectively, as illustrated in Figure 2.

3.2. Arbitrary topology

For the non-box subdivision scheme on regular four-directional meshes, we can consider it as either a four-directional subdivision or a special quadrilateral subdivision. When viewed as a quadrilateral subdivision, type 2 vertices are regarded as auxiliary face control vertices (FCVs). Following these two perspectives, we will generalize the regular non-box subdivision scheme to arbitrary topology.

3.2.1. Topological refinement rules

To extend the regular non-box subdivision to polygonal meshes with arbitrary topology, we can triangulate a polygonal mesh into a *quasi-four-directional mesh* as depicted in Figure 3:

1. **Face vertex insertion**: a face control vertex (FCV) is introduced on each face (see Figure 3(b)).
2. **Triangulation**: an *n*-sided polygon is divided into *n* triangles by connecting its *n* vertices with the FCV (see Figure 3(d)).

For regular quadrilateral meshes, the corresponding quasi-four-directional meshes are obviously regular four-directional meshes.

In the quasi-four-directional mesh, we regard the vertices and edges of the original polygon mesh as type 1 vertices and type 1
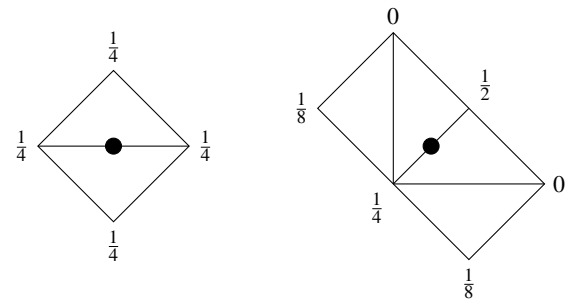
(a) Stencil for type 1 E-vertices   (b) Stencil for type 2 E-vertices
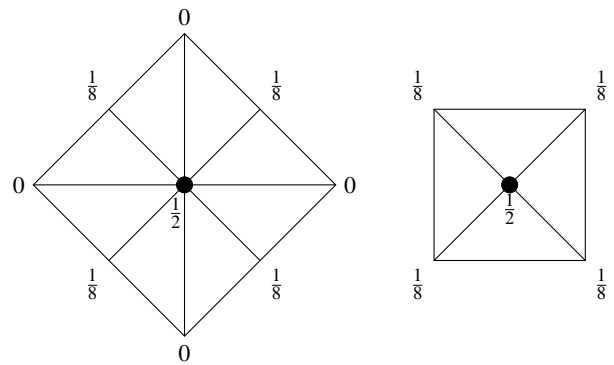
(c) Stencil for type 1 V-vertices   (d) Stencil for type 2 V-vertices

Figure 2: *Subdivision stencils for regular case*

edges of regular four-directional meshes; and the FCVs and new edges as type 2 vertices and type 2 edges. Thus, each triangle has two type 1 vertices and one type 2 vertex, and has one type 1 edge and two type 2 edges.

For quasi-four-direction meshes, the following topological refinement rules are adopted (Figure 3(d)→(e)):

1. On each edge of the coarse mesh in Figure 3(d), a new *E-vertex* is inserted, which is called type 1 E-vertex or type 2 E-vertex, according to the type of the edge.
2. For each triangle of the coarse mesh, the new type 1 E-vertex is connected with the old type 2 vertex and the two new type 2 E-vertices to generate three new edges and four small triangles.
3. In the refined mesh of Figure 3(e), the old type 1 vertices, type 2 vertices and the new type 1 E-vertices become type 1 vertices; and new type 2 E-vertices become type 2 vertices.
4. In the refined mesh, an edge joined by two type 1 vertices is a type 1 edge; and an edge joined by a type 1 vertex and a type 2 vertex is a type 2 edge.

After one step of four-directional subdivision, type-2 vertices in the refined mesh all have valence 4.

If only FCV insertion step is performed on a polygonal mesh, we can get a *polygonal mesh with FCVs* (see Figure 3(b)). In such
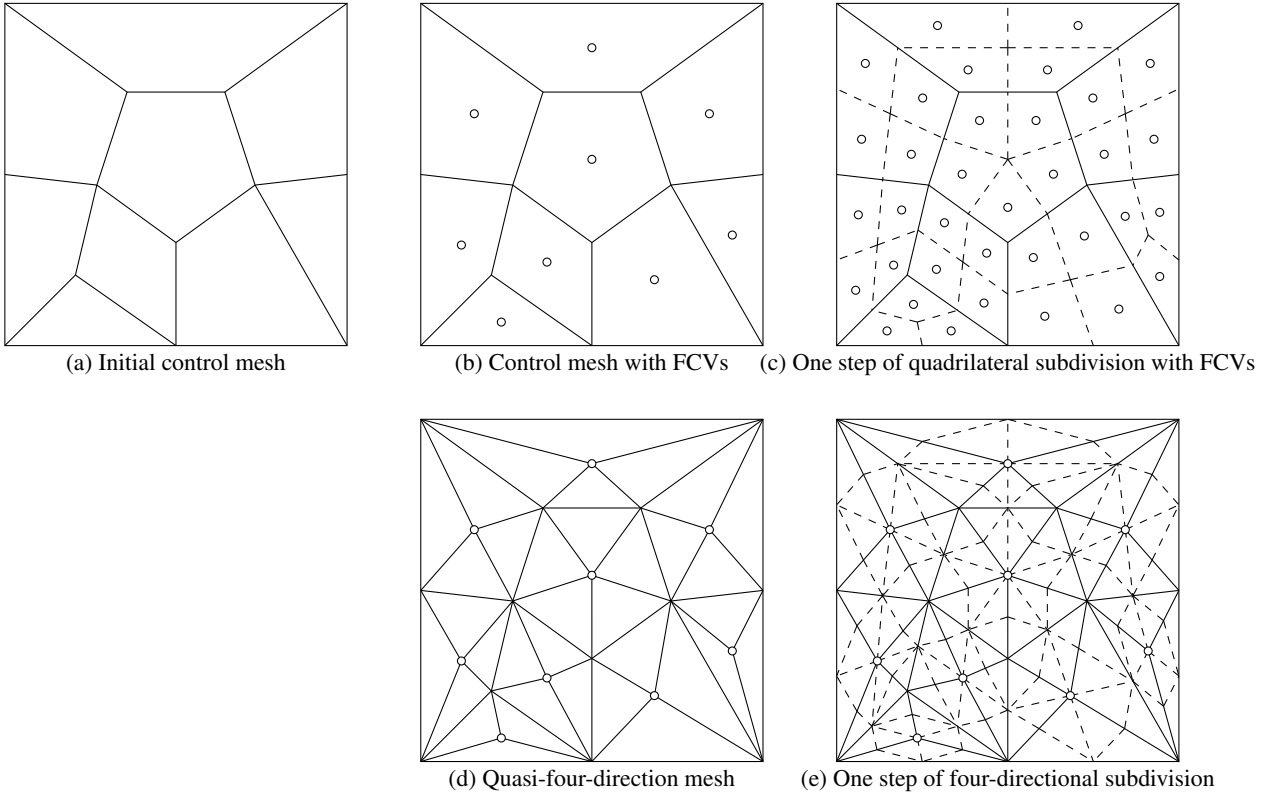
Z. Huang / A Non-Box Spline Subdivision Scheme

(a) Initial control mesh   (b) Control mesh with FCVs   (c) One step of quadrilateral subdivision with FCVs

(d) Quasi-four-direction mesh   (e) One step of four-directional subdivision

Figure 3: *Topological refinement rules.*

mesh, we similarly treat the vertices and edges of the original polygonal mesh as type 1 vertices and type 1 edges respectively, and regard FCVs as type 2 vertices. We adopt the topological refinement rules as follows (Figure 3(b)→(c)):

1. On each edge of the coarse mesh, a new *E-vertex* is inserted.
2. For each face of the coarse mesh, its FCV is linked to the E-vertices associated with its edges.
3. On each face of the refined mesh, a new FCV is inserted.

After one refinement step, a polygonal mesh with FCVs becomes a quadrilateral mesh with FCVs. Except for FCVs, the topology is the same as Catmull-Clark subdivision. Therefore, one can regard this subdivision as a special quadrilateral subdivision. And if we perform the triangulation step on each quadrilateral, we will get the same mesh as the four-directional subdivision (Figure 3(c)→(e)).

Because of this connection, we can utilize similar geometric refinement rules for the above two subdivisions. The difference is that after refinement, triangular meshes and quadrilateral meshes with FCVs are obtained, respectively.

3.2.2. Geometric refinement rules

For four-directional subdivision, there are four subdivision stencils as follows:

1. **Type 1 E-vertex rule** (see Figure 4(a)): the new E-vertex on a type 1 edge is the barycenter of the four vertices of two old triangles with this edge as the common edge.

2. **Type 2 E-vertex rule** (see Figure 4(b)): the new E-vertex on a type 2 edge is computed as

$$\frac{1}{4}V_0^I + \frac{1}{2}V_0^{II} + \frac{1}{8}V_1^{II} + \frac{1}{8}V_{-1}^{II},$$

where V_0^I and V_0^{II} are the type 1 vertex and type 2 vertex of the type 2 edge respectively; and V_1^{II}, V_{-1}^{II} are the two type 2 vertices adjacent to V_0^I and closest to V_0^{II}.

3. **Type 1 V-vertex rule** (see Figure 4(c)): the new position of an existing type 1 vertex V^I is computed as

$$(1-a_k)V^I + (a_k/k)\sum_{i=0}^{k-1}V_i^{II},$$

where V^I is a type 1 vertex with valence k, and $V_i^{II}, i = 0, \ldots, k-1$ are the k vertices of type 2 adjacent to it.

4. **Type 2 V-vertex rule** (see Figure 4(d)): the new position of an existing type 2 vertex V^{II} is computed as

$$(1-a_k)V^{II} + (a_k/k)\sum_{i=0}^{k-1}V_i^I,$$

where V^{II} is a type 2 vertex of valence k, and $V_i^I, i = 0, \ldots, k-1$ are the k vertices of type 1 adjacent to it.

Note that after one level of refinement, the valence of type 2 vertices will remain 4. Then the type 2 V-vertex rule will degenerate to the regular case as shown in Figure 2(d).

In Figure 4, black solid lines represent the type 1 edges; blue dotted lines represent the type 2 edges; black hollow dots represent the type 2 vertices; red solid dots represent the new vertices to be calculated; and the remaining vertices are type 1 vertices.

For quadrilateral subdivision with FCVs, there are four subdivision stencils as well:

1. **E-vertex rule** (see Figure 4(a)): the newly inserted E-vertex for each edge is computed as an average of the two endpoints of the corresponding edge and the two FCVs of the faces sharing the same corresponding edge, i.e. an average of four related vertices.

2. **V-vertex rule** (see Figure 4(c)): the new position of an existing vertex V is computed as

$$(1 - a_k)V + (a_k/k)\sum_{i=0}^{k-1} FCV_i,$$

where V is a vertex with valence k, and $FCV_i, i = 0,\ldots,k-1$ are the FCVs of the k faces incident to the corresponding vertex.

3. **FCV rule** (see Figure 4(d)): the new position of an existing FCV is computed as

$$(1 - a_k)FCV + (a_k/k)\sum_{i=0}^{k-1} V_i,$$

where FCV is the corresponding old FCV of a k-sided face with $V_i, i = 0,\ldots,k-1$ as vertices.
Note that after one refinement step, all faces will be quads. Then the FCV rule will degenerate to the regular case as shown in Figure 2(d).

4. **NFCV rule** (see Figure 4(b)): the new FCV associated with each new face is computed as

$$\frac{1}{4}V_0 + \frac{1}{2}FCV_0 + \frac{1}{8}FCV_1 + \frac{1}{8}FCV_{-1},$$

A new face is a quadrilateral composed of one V-vertex, one FCV and two E-vertices. Here, V_0 is the old vertex corresponding to the V-vertex, FCV_0 is the FCV of the old face F_0 corresponding to the new face, and FCV_1, FCV_{-1} are the FCVs of the two old faces incident to V_0 and adjacent to F_0.

In Figure 4, black solid lines represent old edges; green dotted lines represent new edges; black hollow dots represent FCVs; and red solid dots represent new vertices to be calculated.

4. Smoothness analysis

On regular four-dimensional meshes, the surfaces generated by the proposed subdivision scheme are S_3^1 non-box spline surfaces, so we only need to analyze the smoothness at extraordinary positions.

Following the framework of [Rei95, Zor00] for analyzing convergence and continuity of subdivision schemes, we will perform eigenstructure analysis on a local *subdivision matrix* S_k for an appropriate invariant neighborhood of *extraordinary vertices*, that is, type-1 vertices of valence $2k, k \neq 4$ on a quasi-quadrilateral mesh, or vertices of valence $k, k \neq 4$ on a polygonal mesh with FCVs.

For simplicity, we will analyze the quadrilateral subdivision with FCVs. The invariant neighborhood of an extraordinary vertex with

valence k has $1 + 10k$ control vertices. With the labeling illustrated in Figure 5, we can construct a local subdivision matrix S_k of $(1 + 10k) \times (1 + 10k)$. Let us assume that the subdivision matrix S_k has eigenvalues $\{\lambda_0, \lambda_1, .., \lambda_{1+10k}\}$, with eigenvalues organized in decreasing moduli $|\lambda_i| \geq |\lambda_{i+1}|$.
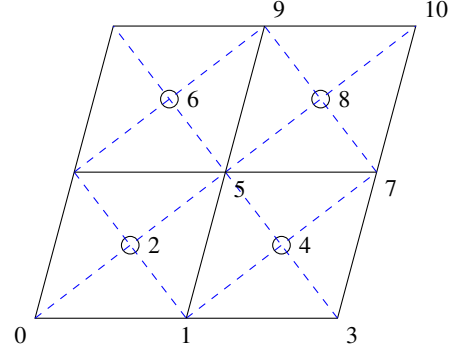

Figure 5: *Labeling of control vertices of an extraordinary sector.*

By applying the standard discrete Fourier transform (DFT) approach [PR08], the local subdivision matrix $S_k, k \geq 3$ can be transformed into a similar block diagonal matrix:

$$diag(B_0, B_1, \ldots, B_{k-1}).$$

Here,

$$B_0 := B(0)$$

$$= \begin{pmatrix}
1-a_k & 0 & a_k & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{1}{4} & \frac{1}{4} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{1}{4} & 0 & \frac{3}{4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{1}{2} & \frac{1}{4} & 0 & \frac{1}{8} & 0 & \frac{1}{8} & 0 & 0 & 0 & 0 \\
0 & \frac{1}{4} & \frac{5}{8} & 0 & \frac{1}{8} & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{1}{8} & \frac{1}{4} & \frac{1}{2} & 0 & 0 & \frac{1}{8} & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{1}{4} & \frac{5}{8} & 0 & 0 & 0 & \frac{1}{8} & 0 & 0 & 0 & 0 \\
0 & \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{1}{2} & 0 & \frac{1}{8} & \frac{1}{4} & \frac{1}{8} & 0 & 0 & 0 & 0 \\
0 & \frac{1}{4} & \frac{1}{4} & 0 & 0 & \frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{1}{8} & 0 & \frac{1}{8} & \frac{1}{2} & \frac{1}{8} & 0 & \frac{1}{8} & 0 & 0
\end{pmatrix}$$

and

$$B_m := B(m)$$

$$= \begin{pmatrix}
\frac{1}{4} & \frac{1}{4}(1+\overline{\omega}) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{1}{2} + \frac{1}{4}c & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{1}{2} & \frac{1}{8}(1+\overline{\omega}) & 0 & \frac{1}{8} & 0 & \frac{1}{8}\overline{\omega} & 0 & 0 & 0 & 0 \\
\frac{1}{4} & \frac{1}{2} + \frac{1}{8}\overline{\omega} & 0 & \frac{1}{8} & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{1}{8}(1+\omega) & \frac{1}{2} & 0 & 0 & \frac{1}{8} & 0 & 0 & 0 & 0 & 0 \\
\frac{1}{4}\omega & \frac{1}{2} + \frac{1}{8}\omega & 0 & 0 & 0 & \frac{1}{8} & 0 & 0 & 0 & 0 \\
\frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{1}{2} & 0 & \frac{1}{8} & \frac{1}{4} & \frac{1}{8} & 0 & 0 & 0 & 0 \\
\frac{1}{4}\omega & \frac{1}{4} & 0 & 0 & \frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 \\
0 & \frac{1}{8} & 0 & \frac{1}{8} & \frac{1}{2} & \frac{1}{8} & 0 & \frac{1}{8} & 0 & 0
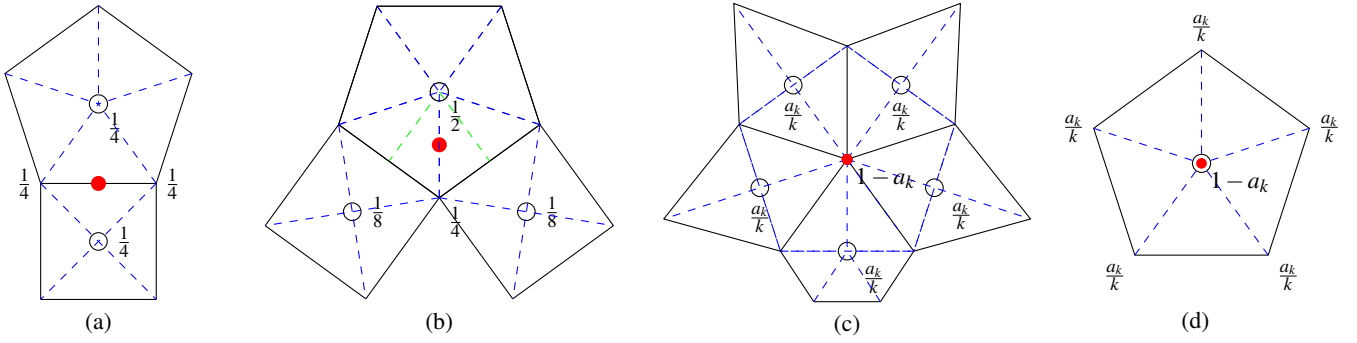\end{pmatrix}$$

Figure 4: *Geometric refinement rules. (a) Stencil for type 1 E-vertex rule of four-directional subdivision, and stencil for E-vertex rule of quadrilateral subdivision with FCVs. (b) Stencil for type 2 E-vertex rule of four-directional subdivision, and stencil for NFCV rule of quadrilateral subdivision with FCVs. (c) Stencil for type 1 V-vertex rule of four-directional subdivision, and stencil for V-vertex rule of quadrilateral subdivision with FCVs. (d) Stencil for type 2 V-vertex rule of four-directional subdivision, and stencil for FCV rule of quadrilateral subdivision with FCVs.*

where $0 \leq a_k \leq 1$, $\omega = \exp(2\pi i m/k)$, $c = \cos(2\pi m/k)$, $i^2 = -1$, $m = 1, \ldots, k-1$, $k \geq 3$.

The eigenvalues of the matrix block $B(0)$ are $1, \frac{3}{4} - a_k, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, 0, 0, 0, 0, 0$, and the eigenvalues of matrix blocks $B(m)$ are $\frac{1}{2} + \frac{c}{4}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, 0, 0, 0, 0, 0$, $m = 1, \ldots, k-1$.

For regular meshes, $a_k = a_4 = \frac{1}{2}$. If $a_k = \frac{1}{2}$ is still adopted for general valence k, the resulting limit surface has a poor appearance near extraordinary vertices. The weights a_k should therefore be a function related to valence k of extraordinary vertices such that $a_4 = \frac{1}{2}$.

Convex hull property. All subdivision stencils should be non-negative, which guarantees the convex hull property. This constraint is equivalent to $0 \leq a_k \leq 1$, $k \geq 3$.

Convergence. A subdivision scheme converges if and only if

$$1 = \lambda_0 > \lambda_1.$$

It is easy to see that $\frac{3}{4} - a_k < 1$ and $\frac{1}{2} + \frac{c}{4} < 1$. According to [Rei95], the real eigenvalues cannot be negative. It follows that

$$\frac{3}{4} - a_k \geq 0.$$

Therefore, if $0 \leq a_k \leq \frac{3}{4}$, $k \geq 3$, then the proposed scheme converges.

G^1 **continuity.** The resulting subdivision surface is tangent plane continuous at each extraordinary vertex if there exists a pair of subdominant eigenvalues λ_1, λ_2 satisfy

$$1 = \lambda_0 > \lambda_1 = \lambda_2 > \lambda_3 \cdots$$

Since $\frac{1}{2} + \frac{1}{4}\cos(2\pi/k) > \frac{1}{4}$, $k \geq 3$, the constraints for tangent plane continuity are therefore

$$\lambda_1 = \lambda_2 = \frac{1}{2} + \frac{1}{4}\cos(2\pi/k),$$
$$\frac{3}{4} - a_k < \frac{1}{2} + \frac{1}{4}\cos(2\pi/k).$$

Thus, if $\frac{1}{4} - \frac{1}{4}\cos(2\pi/k) < a_k \leq \frac{3}{4}$, $k \geq 3$, then the limit surfaces generated by the presented scheme are G^1 continuous.

C^1 **continuity.** The *characteristic map* [Rei95] of an k-valence vertex is defined as the planar limit surface whose initial control mesh is defined by the two right eigenvectors corresponding to the two subdominant eigenvalues λ_1, λ_2, respectively. If the characteristic map of a G^1 subdivision scheme is regular and injective, then the surfaces generated by subdivision are C^1 continuous.

The regularity and injectivity of this map can be approximately judged from the triangulation obtained from the control meshes after several refinement steps [OS03].
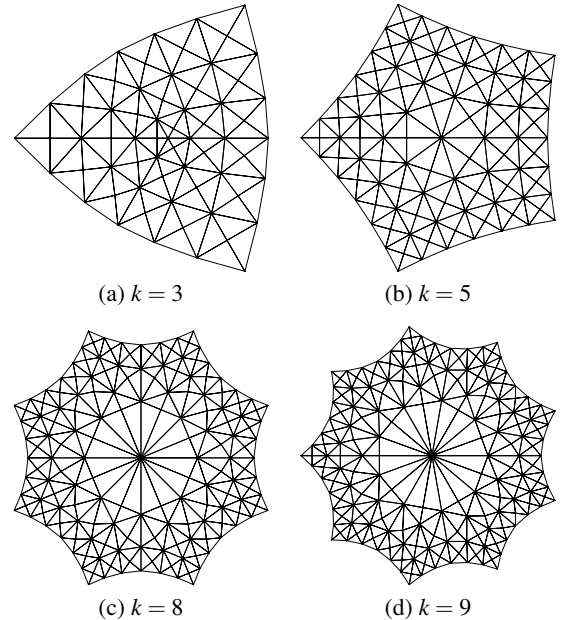

(a) $k = 3$ (b) $k = 5$

(c) $k = 8$ (d) $k = 9$

Figure 6: *Natural configurations for extraordinary vertices with valence $k = 3, 5, 8, 9$.*
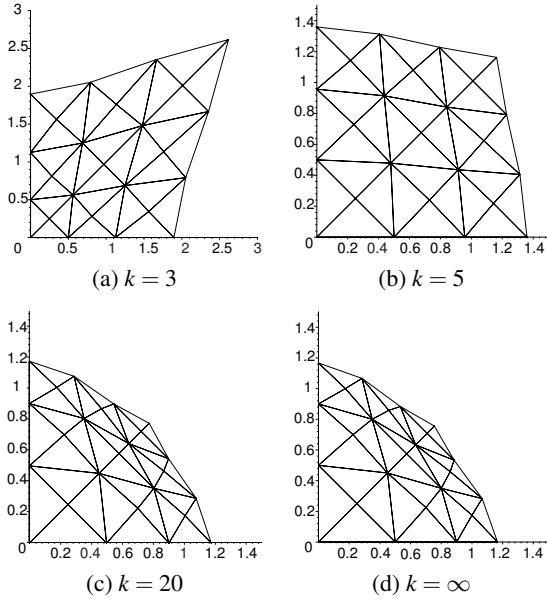
Figure 7: *Normalized characteristic wedges for extraordinary vertices with valence $k = 3, 5, 20, \infty$.*

According to the above G^1 constraints, we set the weights a_k as

$$a_3 = \frac{3}{4}, \quad a_k = \frac{3}{4} - \left(\frac{1}{2} + \frac{1}{4}\cos(2\pi/k)\right)^2, k \geq 4.$$

The natural configurations for extraordinary vertices with valence $k = 3, 5, 8, 9$ are illustrated in Figure 6.

To better visualize the behavior of the characteristic map as $k \to \infty$, one can map this wedge of angle $2\pi/k$ to a *normalized characteristic wedge* of angle $\pi/2$ [OS03]. Figure 7 shows the normalized characteristic wedges for extraordinary vertices with valence $k = 3, 5, 20, \infty$.

According to the visualization of natural configurations in Figure 6 and normalized characteristic wedges in Figure 7 respectively, the characteristic map for arbitrary valence is regular and injective. It follows that

Proposition 1. *If $a_3 = \frac{3}{4}, a_k = \frac{3}{4} - \left(\frac{1}{2} + \frac{1}{4}\cos(2\pi/k)\right)^2, k \geq 4$, then the proposed non-box subdivision scheme achieves C^1 continuity at extraordinary vertices of arbitrary valence.*

Limit position and tangent vectors. The limit position of a control vertex and the tangent vectors at the limit position can be obtained by the eigen analysis of the local subdivision matrix [HKD93]. Denote $\mathbf{V} = (V_0, V_1^0, \ldots, V_1^{k-1}, \ldots, V_{10}^0, \ldots, V_{10}^{k-1})^T$ as the column vector formed by $10k+1$ vertices in the invariant neighborhood of an extraordinary vertex V_0 with valence k; and l_0, l_1, l_2 as the left eigenvectors of the local subdivision matrix S_k with respect to the largest three eigenvalues $\lambda_0, \lambda_1, \lambda_2$, respectively.

The limit position of the control vertex V_0 is defined by

$$V_0^\infty = l_0 \cdot \mathbf{V}.$$

And the tangent vectors at the limit position V_0^∞ are defined by

$$u_1 = l_1 \cdot \mathbf{V}, \ u_2 = l_2 \cdot \mathbf{V}.$$

For the proposed scheme, it follows that

$$
\begin{aligned}
l_0 &= \frac{1}{1+4a_k}(1, 0, \ldots, 0, 4a_k/k, \ldots, 4a_k/k, 0, \ldots, 0)\\
l_1 &= (0, 0, \ldots, 0, c_0, c_1, \ldots, c_{k-1}, 0, \ldots, 0)\\
l_2 &= (0, 0, \ldots, 0, s_0, s_1, \ldots, s_{k-1}, 0, \ldots, 0)
\end{aligned}
$$

where $c_j = \cos(2\pi j/k), s_j = \sin(2\pi j/k), j = 0, \ldots, k-1$.

Thus, the limit position of a control vertex with valence k is

$$V_0^\infty = \frac{kV_0 + 4a_k \sum_{j=0}^{k-1} V_2^j}{k(1+4a_k)}.$$

The tangent vectors at V_0^∞ are

$$
\begin{aligned}
u_1 &= \sum_{j=0}^{k-1} c_j V_2^j,\\
u_2 &= \sum_{j=0}^{k-1} s_j V_2^j.
\end{aligned}
$$

5. Implementation

For arbitrary polygonal meshes, one can transform them into polygonal meshes with FCVs or quasi-four-directional meshes. However, polygonal meshes with FCVs are simpler than quasi-four-directional meshes in data structure and implementation. In the following we only describe algorithms for polygonal meshes with FCVs as examples. To obtain a quasi-four-directional mesh, we only need to triangulate each face.

Algorithm 1: One step of quadrilateral subdivision with FCVs.

/* (1) Compute the vertices on the new refined mesh */
/* (1.1) Calculate V-vertices */
for *each vertex v* **do**
  Compute the new position of *v* according to the V-vertex rule (see Figure 4(c));

/* (1.2) Calculate E-vertices */
for *each edge e* **do**
  Compute the position of new E-vertex corresponding to *e* according to the E-vertex rule (see Figure 4(a));

/* (1.3) Update FCVs */
for *each face f* **do**
  Compute the new position of FCV on *f* according to the FCV rule (see Figure 4(d));

/* (2) Construct new faces in the refined mesh */
for *each face f* **do**
  for *each vertex v in face f* **do**
    Form a new quad face by the FCV of *f*, the V-vertex corresponding to *v*, and the two new E-vertices on the edges of *f* adjacent to *v*;

    /* (3) Calculate FCV of the new face */
    Compute the position of FCV for the new face according to the NFCV rule (see Figure 4(b));

Considering the similarity with general polygonal meshes, the

data structure and implementation can refer to the algorithms for Catmull-Clark subdivision, except that each face needs to store an additional FCV.

Boundary rules. The non-box subdivision scheme presented in previous section applies to closed meshes. For open meshes, since the FCV rule is only related to the vertices of the face where the FCV lies, it is still applicable near the boundary. We need to design boundary E-vertex rule, V-vertex rule and NFCV rule. Following [HDD*94], we use the subdivision rules of cubic B-spline on the boundary. Figure 8 presents the stencils of the boundary geometric refinement rules. The initial mesh and subdivision surface of an open pipe surface is depicted in Figure 9.
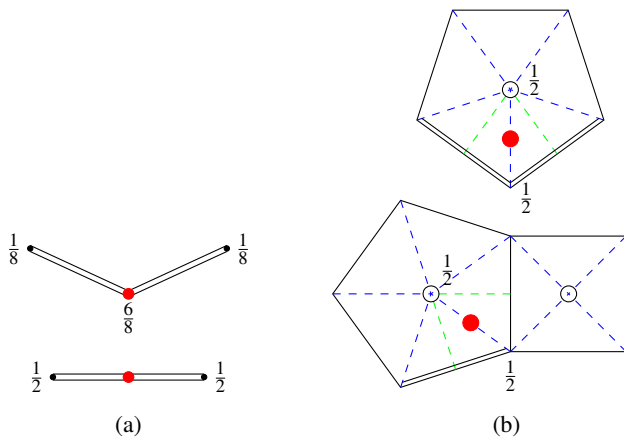
Figure 8: *Boundary geometric refinement rules. (a) Boundary V-vertex stencil (upper) and E-vertex stencil (lower). (b) Boundary NFCV stencil.*
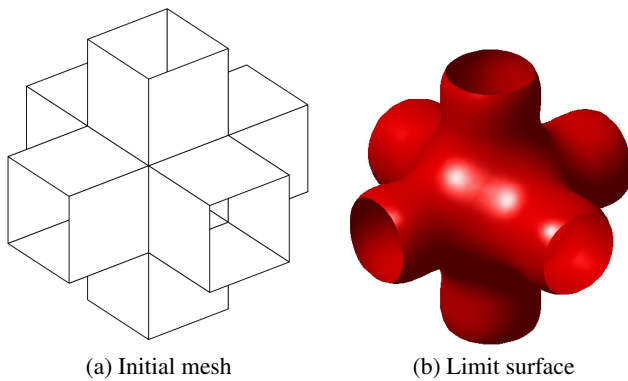

(a) Initial mesh   (b) Limit surface

Figure 9: *Subdivision surface of an open mesh.*

Choice of initial FCVs. To apply the proposed subdivision schemes to general polygonal meshes, FCVs should be introduced to convert the general meshes into quasi-four-directional meshes or polygonal meshes with FCVs.

The selection of initial FCVs has a great influence on the shape of the limit surfaces. In practice, an initial FCV can be taken as the barycenter of each face, and then its position can be modified interactively by the designer according to the modeling requirements.

Given that the initial control mesh is a cube centered at the origin, we take FCV as 0.5, 1, and 1.5 times the barycenter of each face, respectively, to obtain the different results in Figure 10.

6. Examples and comparison

In this section, we compare the presented non-box subdivision with other subdivision schemes, and give some surface modeling examples. Unless otherwise specified, initial FCVs in the example are taken as barycenters of the faces.

Figure 11 shows the subdivision surfaces generated from a cube or a tetrahedron by different subdivision schemes. We can see that although non-box subdivision is also derived from C^1 spline functions, the generated surfaces are smoother than those generated by Doo-Sabin scheme. The visual appearance of the surfaces produced by our scheme is close to Loop and 4-8 subdivision schemes.

As a special quadrilateral subdivision, compared with the traditional schemes, the proposed scheme introduces FCVs which can be used to control the limit shapes. In interactive modeling, the shape of the limit surface can be controlled more flexibly to get the desired surface. The control mesh in Figure 12(a) generates a doughnut shape by general schemes. While with the proposed scheme, different limit surfaces can be produced by adjusting the positions of the FCVs, as illustrated in Figure 12.

Some examples of complex surfaces are given in Figure 13 and Figure 14. The initial mesh of the model in Figure 13 is a quadrilateral mesh with vertices of valences from 3 to 6, and the initial mesh of the Manequin head model in Figure 14 is a triangular mesh.

7. Conclusion

In this paper, we present a surface subdivision scheme based on four-directional S_3^1 non-box splines. The resulting limit surfaces are C^1 continuous with the convex hull property. The scheme can be viewed as either a four-directional subdivision or a special quadrilateral subdivision. As a quadrilateral subdivision, due to the introduction of auxiliary face control vertices (FCVs) in the control meshes, we can obtain limit surfaces with different characteristics by setting different initial FCVs. Compared with traditional schemes, our scheme is more flexible and simple in shape control.

The proposed boundary refinement rules are heuristic and lack smoothness analysis. In future work we will investigate C^1 boundary rules and tune the presented non-box subdivision scheme to achieve better limit surface quality. It's possible to generate a limit surface with a crease or self-intersects if moving FCVs to some positions. How to get a good shape by FCVs will be an interesting topic. Additionally, we will explore interpolatory surface schemes based on interpolatory vector subdivision schemes [CZ04].
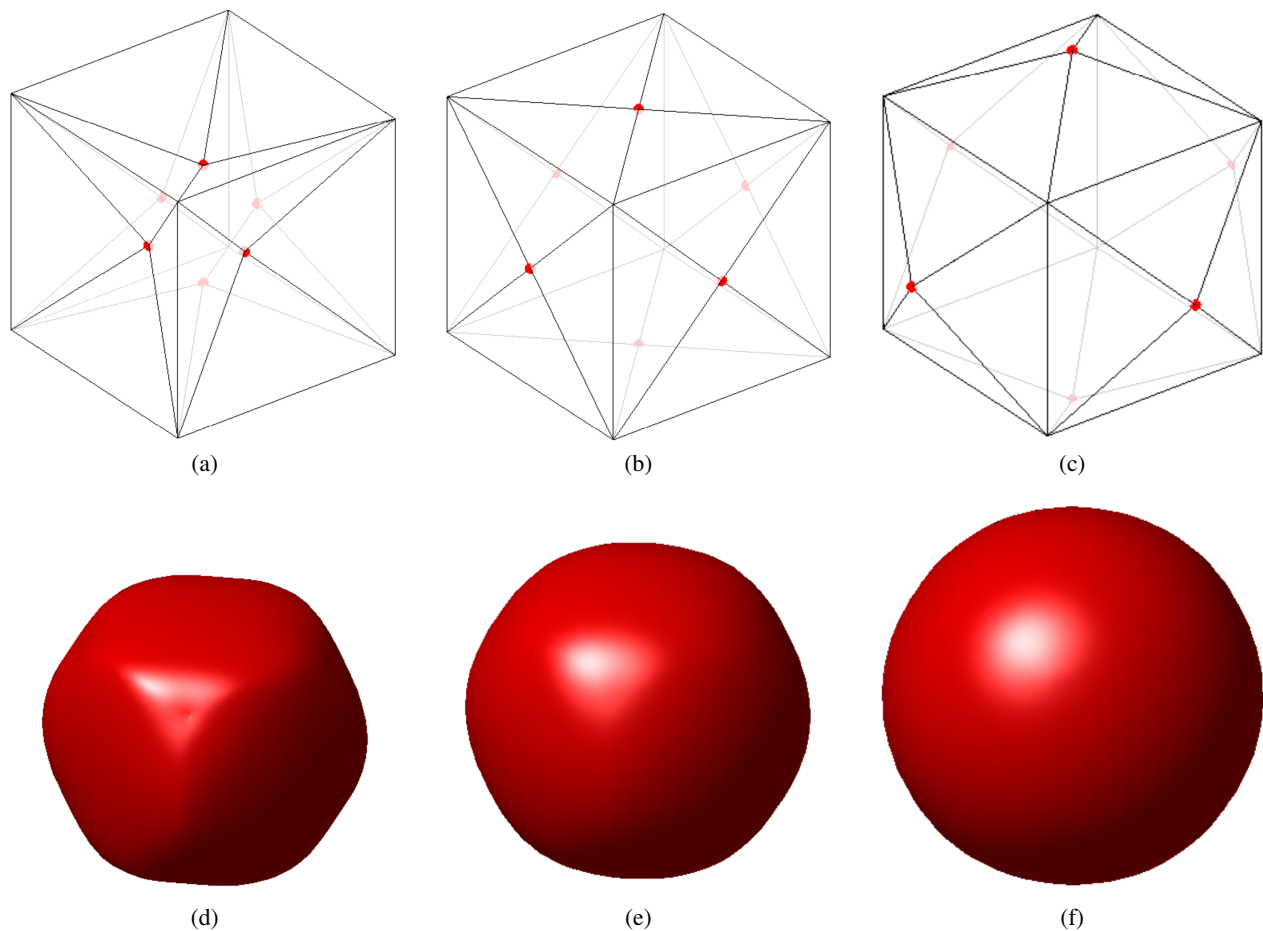
Figure 10: *Selection of initial FCVs. (a) Polygonal control mesh with 0.5 times barycenters as FCVs. (b) Polygonal control mesh with barycenters as FCVs. (c) Polygonal control mesh with 1.5 times barycenters as FCVs. (d) Limit surface corresponding to (a). (e) Limit surface corresponding to (b). (f) Limit surface corresponding to (c). FCVs are marked as red dots.*

References

[BSK19] BARENDRECHT P., SABIN M., KOSINKA J.: A bivariate C^1 subdivision scheme based on cubic half-box splines. *Computer Aided Geometric Design 71* (2019), 77–89. URL: https://www.sciencedirect.com/science/article/pii/S0167839619300184, doi:https://doi.org/10.1016/j.cagd.2019.04.004. 1

[CC78] CATMULL E., CLARK J.: Recursively generated B-spline surfaces on arbitrary topological meshes. *Computer-Aided Design 10*, 6 (1978), 350–355. URL: https://www.sciencedirect.com/science/article/pii/0010448578901100, doi:https://doi.org/10.1016/0010-4485(78)90110-0. 1

[CJ00] CONTI C., JETTER K.: A new subdivision method for bivariate splines on the four-directional mesh. *J. Comput. Appl. Math. 119*, 1-2 (jul 2000), 81–96. URL: https://doi.org/10.1016/S0377-0427(00)00372-1, doi:10.1016/S0377-0427(00)00372-1. 1, 2

[CZ04] CONTI C., ZIMMERMANN G.: Interpolatory rank-1 vector subdivision schemes. *Computer Aided Geometric Design 21*, 4 (2004), 341–351. URL: https://www.sciencedirect.com/science/article/pii/S0167839603001596, doi:https://doi.org/10.1016/j.cagd.2003.11.003. 8

[dBHR93] DE BOOR C., HÖLLIG K., RIEMENSCHNEIDER S.: *Box Splines*. Springer-Verlag, Berlin, Heidelberg, 1993. 1

[DS78] DOO D., SABIN M.: Behaviour of recursive division surfaces near extraordinary points. *Computer-Aided Design 10*, 6 (1978), 356–360. URL: https://www.sciencedirect.com/science/article/pii/0010448578901112, doi:https://doi.org/10.1016/0010-4485(78)90111-2. 1

[HDD*94] HOPPE H., DEROSE T., DUCHAMP T., HALSTEAD M., JIN H., MCDONALD J., SCHWEITZER J., STUETZLE W.: Piecewise smooth surface reconstruction. In *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1994), SIGGRAPH '94, Association for Computing Machinery, p. 295–302. URL: https://doi.org/10.1145/192161.192233, doi:10.1145/192161.192233. 8

[HKD93] HALSTEAD M., KASS M., DEROSE T.: Efficient, fair interpolation using Catmull-Clark surfaces. In *Proceedings of the 20th Annual Conference on Computer Graphics and Interactive Techniques* (New York, NY, USA, 1993), SIGGRAPH '93, Association for Com-
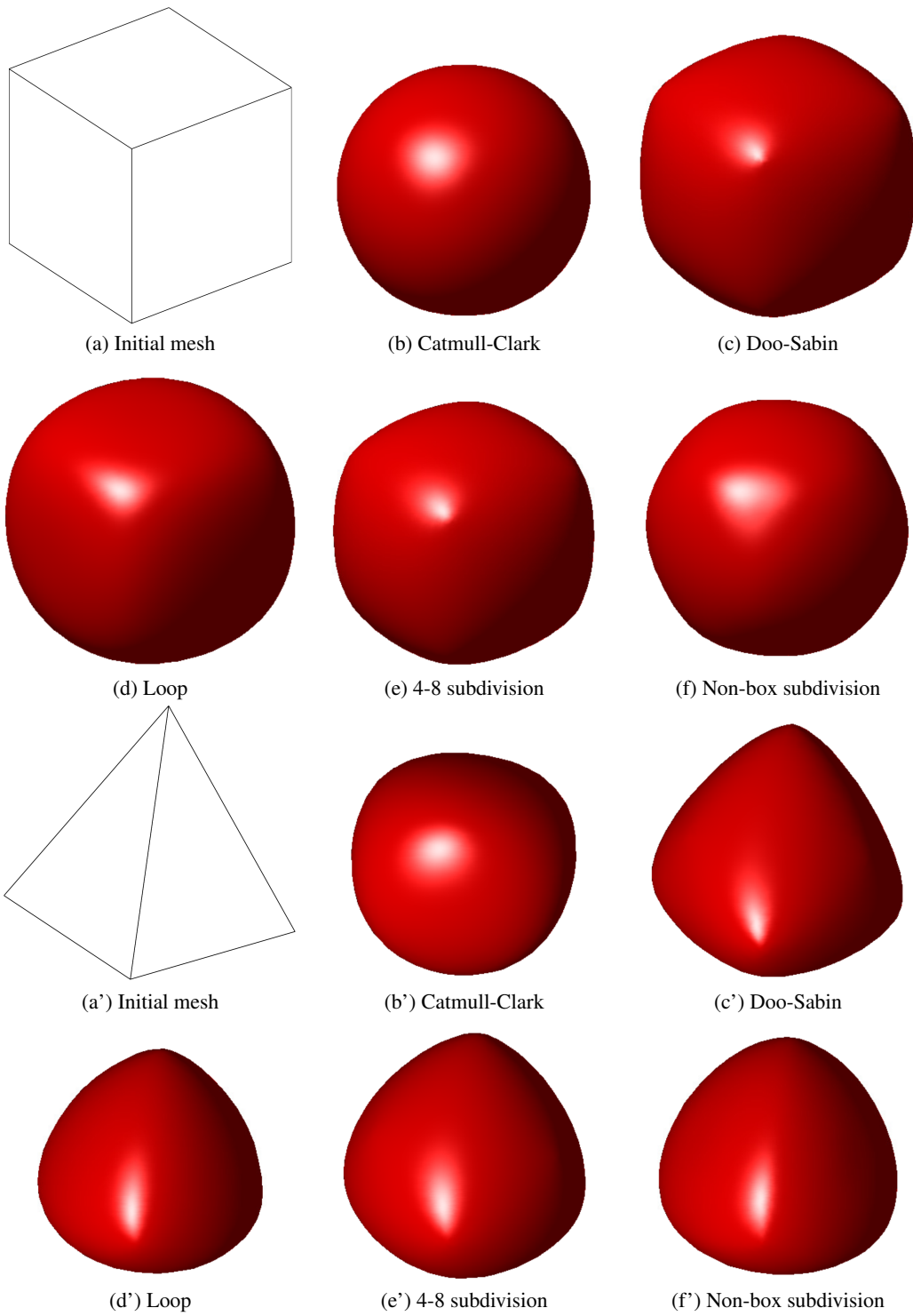
Z. Huang / A Non-Box Spline Subdivision Scheme

(a) Initial mesh   (b) Catmull-Clark   (c) Doo-Sabin

(d) Loop   (e) 4-8 subdivision   (f) Non-box subdivision

(a') Initial mesh   (b') Catmull-Clark   (c') Doo-Sabin

(d') Loop   (e') 4-8 subdivision   (f') Non-box subdivision

Figure 11: *Comparison between surfaces generated by different subdivision schemes.*
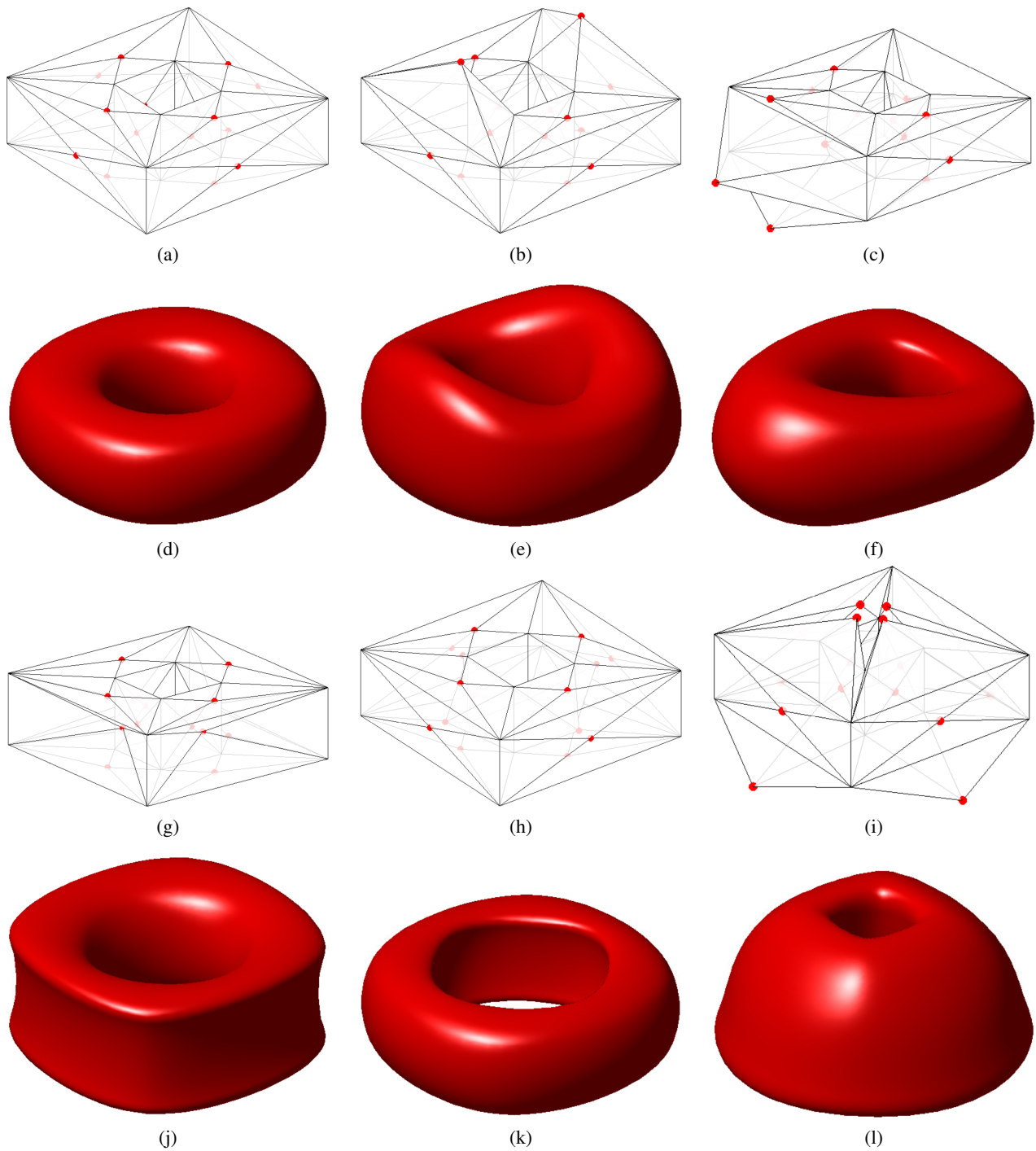
Figure 12: *Comparison of surfaces generated from different FCVs. (d), (e), (f), (j), (k) and (l) are the limit surfaces generated from the control meshes in (a), (b), (c), (g), (h) and (i), respectively. FCVs are marked as red dots.*
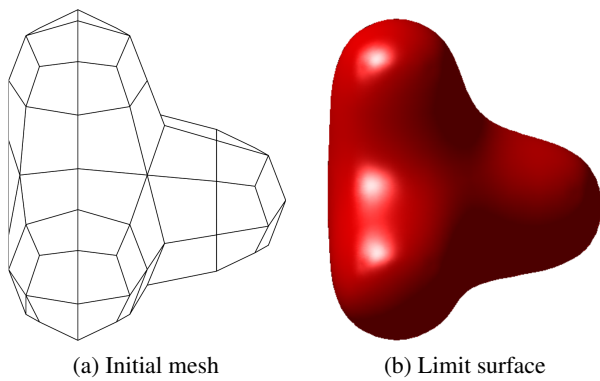
(a) Initial mesh   (b) Limit surface

Figure 13: *A model generated from a quadrilateral mesh with vertices of valences from 3 to 6.*

(a) Initial mesh   (b) Limit surface
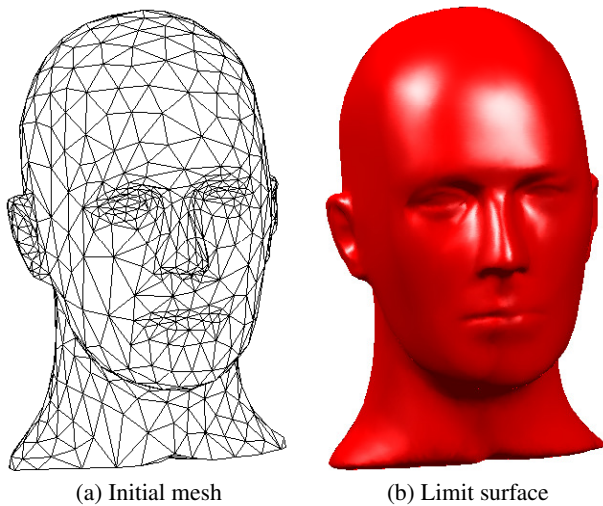
Figure 14: *Manequin head model.*

puting Machinery, p. 35–44. URL: https://doi.org/10.1145/166117.166121, doi:10.1145/166117.166121. 7

[HW99] HABIB A., WARREN J.: Edge and vertex insertion for a class of C^1 subdivision surfaces. *Computer Aided Geometric Design 16*, 4 (1999), 223–247. URL: https://www.sciencedirect.com/science/article/pii/S0167839698000454, doi:https://doi.org/10.1016/S0167-8396(98)00045-4. 1

[Loo87] LOOP C.: *Smooth Subdivision Surfaces Based on Triangles.* Master's thesis, University of Utah, August 1987. 1

[MS98] MICCHELLI C. A., SAUER T.: On vector subdivision. *Mathematische Zeitschrift 229* (1998), 621–674. doi:https://doi.org/10.1007/PL00004676. 1, 2

[OS03] OSWALD P., SCHRÖDER P.: Composite primal/dual $\sqrt{3}$-subdivision schemes. *Computer Aided Geometric Design 20*, 3 (2003), 135–164. URL: https://www.sciencedirect.com/science/article/pii/S0167839603000268, doi:https://doi.org/10.1016/S0167-8396(03)00026-8. 6, 7

[PR97] PETERS J., REIF U.: The simplest subdivision scheme for smoothing polyhedra. *ACM Trans. Graph. 16*, 4 (oct 1997), 420–

431. URL: https://doi.org/10.1145/263834.263851, doi:10.1145/263834.263851. 1

[PR08] PETERS J., REIF U.: *Subdivision Surfaces*, 1 ed. Geometry and Computing. Springer Berlin, Heidelberg, 2008. 5

[Rei95] REIF U.: A unified approach to subdivision algorithms near extraordinary vertices. *Computer Aided Geometric Design 12*, 2 (1995), 153–174. URL: https://www.sciencedirect.com/science/article/pii/016783969400007F, doi:https://doi.org/10.1016/0167-8396(94)00007-F. 5, 6

[VZ01] VELHO L., ZORIN D.: 4-8 subdivision. *Comput. Aided Geom. Des. 18*, 5 (jun 2001), 397–427. URL: https://doi.org/10.1016/S0167-8396(01)00039-5, doi:10.1016/S0167-8396(01)00039-5. 1

[Zor00] ZORIN D.: A method for analysis of C^1-continuity of subdivision surfaces. *SIAM Journal on Numerical Analysis 37*, 5 (2000), 1677–1708. URL: https://doi.org/10.1137/S003614299834263X, doi:10.1137/S003614299834263X. 5