Chapter 4

Subdivision Zoo

Denis Zorin, New York University

4.1   Overview of Subdivision Schemes

In this section we describe most known stationary subdivision schemes generating C^1-continuous surfaces on arbitrary meshes. Without doubt, our discussion is not exhaustive even as far as stationary schemes are concerned. There are even wholly different classes of subdivision schemes, most importantly variational schemes, that we do not discuss here (see Chapter 9).

At first glance, the variety of existing schemes might appear chaotic. However, there is a straightforward way to classify most of the schemes based on four criteria:

- the type of refinement rule (face split or vertex split);
- the type of generated mesh (triangular or quadrilateral);
- whether the scheme is approximating or interpolating;
- smoothness of the limit surfaces for regular meshes (C^1, C^2 etc.)

The following table shows this classification:

Face split		
	Triangular meshes	*Quad. meshes*
Approximating	Loop (C^2)	Catmull-Clark (C^2)
Interpolating	Mod. Butterfly (C^1)	Kobbelt (C^1)

Vertex split
Doo-Sabin, Midedge (C^1)
Biquartic (C^2)

Out of recently proposed schemes, $\sqrt{3}$ subdivision [12], and subdivision on $4 - k$ meshes [31, 32] do not fit into this classification. In this survey, we focus on the better-known and established schemes, and this classification is sufficient for most purposes. It can be extended to include the new schemes, as discussed in Section 4.9.

The table shows that there is little replication in functionality: most schemes produce substantially different types of surfaces. Now we consider our classification criteria in greater detail.

First, we note that each subdivision scheme defined on meshes of arbitrary topology is based on a *regular subdivision scheme*, for example, one based on splines. Our classification is primarily a classification of regular subdivision schemes—once such a scheme is fixed, additional rules have to be specified only for extraordinary vertices or faces that cannot be part of a regular mesh.

Mesh Type.   Regular subdivision schemes act on regular control meshes, that is, vertices of the mesh correspond to regularly spaced points in the plane. However, the faces of the mesh can be formed in different ways. For a regular mesh, it is natural to use faces that are identical. If, in addition, we assume that the faces are regular polygons, it turns out that there are only three ways to choose the face polygons: we can use squares, equilateral triangles and regular hexagons. Meshes consisting of hexagons are not very common, and the first two types of tiling are the most convenient for practical purposes. These lead to two types of regular subdivision schemes: those defined for quadrilateral tilings, and those defined for triangular tilings.

Face Split and Vertex Split.   Once the tiling of the plane is fixed, we have to define how a refined tiling is related to the original tiling. There are two main approaches that are used to generate a refined tiling: one is *face split* and the other is *vertex split* (see Figure 4.1). The schemes using the first method are often called *primal*, and the schemes using the second method are called *dual*. In the first case, each face of a triangular or a quadrilateral mesh is split into four. Old vertices are retained, new vertices are inserted on the edges, and for quadrilaterals, an additional vertex is inserted for each face. In the second case, for each old vertex, several new vertices are created, one for each face adjacent to the vertex. A new face is created for each edge and old faces are retained; in addition, a new face is created for each vertex. For quadrilateral tilings, this results in tilings in which each vertex has valence 4. In the case of triangles vertex split (dual) schemes results in non-nesting hexagonal tilings. In this sense quadrilateral tilings are special: they support both primal and dual subdivision schemes easily (see also Chapter 5).
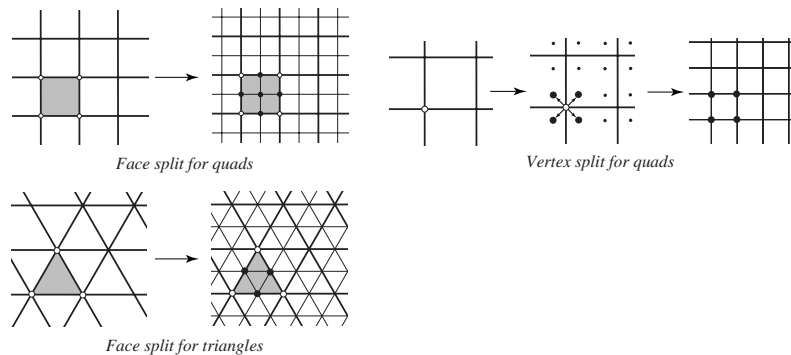
Face split for quads

Vertex split for quads

Face split for triangles

Figure 4.1: *Different refinement rules.*

Approximation vs. Interpolation. Face-split schemes can be interpolating or approximating. Vertices of the coarser tiling are also vertices of the refined tiling. For each vertex a sequence of control points, corresponding to different subdivision levels, is defined. If all points in the sequence are the same, we say that the scheme is interpolating. Otherwise, we call it approximating. Interpolation is an attractive feature in more than one way. First, the original control points defining the surface are also points of the limit surface, which allows one to control it in a more intuitive manner. Second, many algorithms can be considerably simplified, and many calculations can be performed "in place." Unfortunately, the quality of these surfaces is not as high as the quality of surfaces produced by approximating schemes, and the schemes do not converge as fast to the limit surface as the approximating schemes.

4.1.1 Notation and Terminology

Here we summarize the notation that we use in subsequent sections. Some of it was already introduced earlier.

Regular and extraordinary vertices. We have already seen that subdivision schemes defined on triangular meshes create new vertices of valence 6 in the interior. On the boundary, the newly created vertices have valence 4. Similarly, on quadrilateral meshes both face-split and vertex-split schemes create only vertices of valence 4 in the interior, and 3 on the boundary. Hence, after several subdivision steps, most vertices in a mesh will have one of these valences (6 in the interior, 4 on the boundary for triangular meshes, 4 in the interior, 3 on the boundary for quadrilateral). The vertices with these valences are called *regular* and vertices of other valences *extraordinary*.

Notation for vertices near a fixed vertex. In Figure 4.2 we show the notation that we use for control points of quadrilateral and triangular subdivision schemes near a fixed vertex. Typically, we need it for extraordinary vertices. We also use it for regular vertices when describing calculations of limit positions and tangent vectors.

Odd and even vertices. For face-split (primal) schemes, the vertices of the coarser mesh are also vertices of the refined mesh. For any subdivision level, we call all new vertices that are created at that level, *odd vertices*. This term comes from the one-dimensional case, when vertices of the control polygons can be enumerated sequentially and on any level the newly inserted vertices are assigned odd numbers. The vertices inherited from the previous level are called *even*. (See also Chapter 2).

Face and edge vertices. For triangular schemes (Loop and Modified Butterfly), there is only one type of odd vertex. For quadrilateral schemes, some vertices are inserted when edges of the coarser mesh are split, other vertices are inserted for a face. These two types of odd vertices are called *edge* and *face* vertices respectively.

Boundaries and creases. Typically, special rules have to be specified on the boundary of a mesh. These rules are commonly chosen in such a way that the boundary curve of the limit surface does not depend on any interior control vertices, and is smooth or piecewise smooth (C^1 or C^2-continuous). The same rules can be used to introduce sharp features into C^1-surfaces: some interior edges can be *tagged* as crease edges, and boundary rules are applied for all vertices that are inserted on such edges.
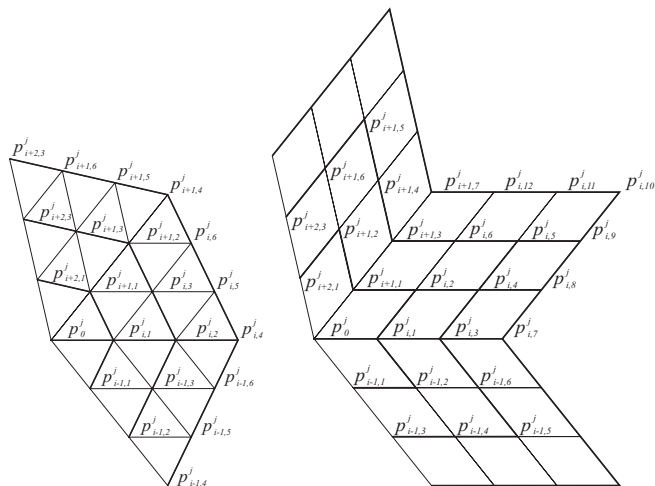
Figure 4.2: *Enumeration of vertices of a mesh near an extraordinary vertex; for a boundary vertex, the*
$0 - th$ *sector is adjacent to the boundary.*

Masks. We often specify a subdivision rule by providing its *mask*. The mask is a picture showing the control points used to compute a new control point, which we usually denote with a black dot. The numbers next to the vertices are the coefficients of the subdivision rule.

4.2 Loop Scheme

The Loop scheme is a simple approximating face-split scheme for triangular meshes proposed by Charles Loop [16]. C^1-continuity of this scheme for valences up to 100, including the boundary case, was proved by Schweitzer [28]. The proof for all valences can be found in [35].

The scheme is based on the *three-directional box spline*, which produces C^2-continuous surfaces over regular meshes. The Loop scheme produces surfaces that are C^2-continuous everywhere except at extraordinary vertices, where they are C^1-continuous. Hoppe, DeRose, Duchamp et al. [10] proposed a piecewise C^1-continuous extension of the Loop scheme, with special rules defined for edges; in [2, 3],

the boundary rules are further improved, and new rules for concave corners and normal modification are proposed.

The scheme can be applied to arbitrary polygonal meshes, after the mesh is converted to a triangular mesh, for example, by triangulating each polygonal face.

Subdivision Rules. The masks for the Loop scheme are shown in Figure 4.3. For boundaries and edges tagged as *crease* edges, special rules are used. These rules produce a cubic spline curve along the boundary/crease. The curve only depends on control points on the boundary/crease.
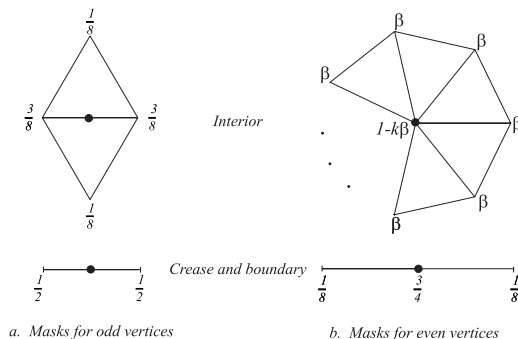

a. Masks for odd vertices   b. Masks for even vertices

Figure 4.3: *Loop subdivision: in the picture above,* β *can be chosen to be either* $\frac{1}{n}(5/8 - (\frac{3}{8} + \frac{1}{4}\cos\frac{2\pi}{n})^2)$ *(original choice of Loop [16]), or, for* $n > 3$, $\beta = \frac{3}{8n}$ *as proposed by Warren [33]. For* $n = 3$, $\beta = 3/16$ *can be used.*

In [10], the rules for extraordinary crease vertices and their neighbors on the crease were modified to produce tangent plane continuous surfaces on either side of the crease (or on one side of the boundary). In practice, this modification does not lead to a significant difference in the appearance of the surface. At the same time, as a result of this modification, the crease curve becomes dependent on the valences of vertices on the curve. This is a disadvantage in situations when two surfaces have to be joined together along a boundary. It appears that for display purposes it is safe to use the rules shown in Figure 4.3. Although the surface will not be formally C^1-continuous near vertices of valence greater than 7, the result will be visually indistinguishable from a C^1-surface obtained with modified rules, with the additional advantage of independence of the boundary from the interior.

If it is necessary to ensure C^1-continuity, a different modification can be used. Rather than modifying the rules for a crease, and making them dependent on the valence of vertices, we modify rules for interior odd vertices adjacent to an extraordinary vertex. For $n < 7$, no modification is necessary. For $n > 7$, it is sufficient to use the mask shown in Figure 4.4. Then the limit surface can be shown to be C^1-continuous at the boundary. A better, although slightly more complex modification can be found in [3, 2]: instead of $\frac{1}{2}$ and $\frac{1}{4}$ we can use $\frac{1}{4} + \frac{1}{4}\cos\frac{2\pi}{k-1}$ and $\frac{1}{2} - \frac{1}{4}\cos\frac{2\pi}{k-1}$ respectively, where k is the valence of the boundary/crease vertex.
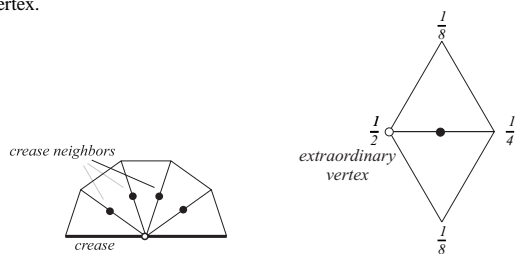

Figure 4.4: *Modified rule for odd vertices adjacent to a boundary/crease extraordinary vertex (Loop scheme).*

Tangent Vectors. The rules for computing tangent vectors for the Loop scheme are especially simple. To compute a pair of tangent vectors at an interior vertex, use

$$t_1 = \sum_{i=0}^{k-1} \cos\frac{2\pi i}{k} p_{i,1}$$
$$t_2 = \sum_{i=0}^{k-1} \sin\frac{2\pi i}{k} p_{i,1}. \tag{4.1}$$

These formulas can be applied to the control points at any subdivision level.

Quite often, the tangent vectors are used to compute a normal. The normal obtained as the cross product $t_1 \times t_2$ can be interpreted geometrically. This cross product can be written as a weighted sum of normals to all possible triangles formed by p_0, $p_{i,1}$, $p_{l,1}$, $i,l = 0\ldots k-1$, $i \neq l$. The standard way of obtaining vertex normals for a mesh by averaging the normals of triangles adjacent to a vertex, can be regarded as a first approximation to the normals given by the formulas above. At the same time, it is worth observing that computing normals as $t_1 \times t_2$ is less expensive than averaging the normals of

triangles. The geometric nature of the normals obtained in this way suggests that they can be used to compute approximate normals for other schemes, even if the precise normals require more complicated expressions.

At a boundary vertex, the tangent along the curve is computed using $t_{along} = p_{0,1} - p_{k-1,1}$. The tangent across the boundary/crease is computed as follows [10]:

$$t_{across} = p_{0,1} + p_{1,1} - 2p_0 \quad \text{for } k = 2$$
$$t_{across} = p_{2,1} - p_0 \quad \text{for } k = 3 \tag{4.2}$$
$$t_{across} = \sin\theta\,(p_{0,1} + p_{k-1,1}) + (2\cos\theta - 2)\sum_{i=1}^{k-2} \sin i\theta\, p_{i,1} \quad \text{for } k \geq 4$$

where $\theta = \pi/(k-1)$. These formulas apply whenever the scheme is tangent plane continuous at the boundary; it does not matter which method was used to ensure tangent plane continuity.

Limit Positions. Another set of simple formulas allows one to compute limit positions of control points for a fixed vertex, that is, the limit $\lim_{j\to\infty} p^j$ for a fixed vertex. For interior vertices, the mask for computing the limit value at an interior vertex is the same as the mask for computing the value on the next level, with β replaced by $\chi = \frac{1}{3/8\beta+n}$.

For boundary and crease vertices, the formula is always

$$p_0^\infty = \frac{1}{5}p_{0,1} + \frac{3}{5}p_0 + \frac{1}{5}p_{1,k-1}$$

This expression is similar to the rule for even boundary vertices, but with different coefficients. However, different formulas have to be used if the rules on the boundary are modified as in [10].

4.3   Modified Butterfly Scheme

The Butterfly scheme was first proposed by Dyn, Gregory and Levin in [7]. The original Butterfly scheme is defined on arbitrary triangular meshes. However, the limit surface is not C^1-continuous at extraordinary points of valence $k = 3$ and $k > 7$ [35], while it is C^1 on regular meshes.

Unlike approximating schemes based on splines, this scheme does not produce piecewise polynomial surfaces in the limit. In [39] a modification of the Butterfly scheme was proposed, which guarantees that the scheme produces C^1-continuous surfaces for arbitrary meshes (for a proof see [35]). The scheme is known to be C^1 but not C^2 on regular meshes. The masks are shown in Figure 4.5.
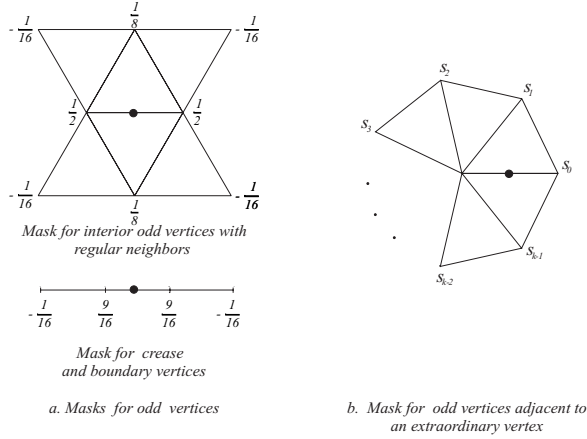
Mask for interior odd vertices with
regular neighbors

Mask for crease
and boundary vertices

a. Masks for odd vertices

b. Mask for odd vertices adjacent to
an extraordinary vertex

Figure 4.5: *Modified Butterfly subdivision. The coefficients s_i are $\frac{1}{k}\left(\frac{1}{4}+\cos\frac{2i\pi}{k}+\frac{1}{2}\cos\frac{4i\pi}{k}\right)$ for $k > 5$. For $k = 3$, $s_0 = \frac{5}{12}$, $s_{1,2} = -\frac{1}{12}$; for $k = 4$, $s_0 = \frac{3}{8}$, $s_2 = -\frac{1}{8}$, $s_{1,3} = 0$.*

The tangent vectors at extraordinary interior vertices can be computed using the same rules as for the Loop scheme. For regular vertices, the formulas are more complex: in this case, we have to use control points in a 2-neighborhood of a vertex. If the control points are arranged into a vector $p = [p_0, p_{0,1}, p_{1,1}, \ldots, p_{5,1}, p_{0,2}, p_{1,2}, p_{2,2}, \ldots p_{5,3}]$ of length 19, then the tangents are given by scalar products $(l_1 \cdot p)$ and $(l_2 \cdot p)$, where the vectors l_1 and l_2 are

$$l_1 = \left[0, 16, 8, -8, -16, -8, 8, -4, 0, 4, 4, 0, -4, 1, \frac{1}{2}, -\frac{1}{2}, -1, -\frac{1}{2}, \frac{1}{2}\right]$$

$$l_2 = \sqrt{3}\left[0, 0, 8, 8, 0, -8, -8, -\frac{4}{3}, -\frac{8}{3}, -\frac{4}{3}, \frac{4}{3}, \frac{8}{3}, \frac{4}{3}, 0, \frac{1}{2}, \frac{1}{2}, 0, -\frac{1}{2}, -\frac{1}{2}\right]$$

(4.3)

Because the scheme is interpolating, no formulas are needed to compute the limit positions: all control points are on the surface. On boundaries and creases the four-point subdivision scheme, also shown in Figure 4.5, is used [6]. To achieve C^1-continuity on the boundary, special coefficients have to be used for crease neighbors, similar to the case of the Loop scheme. One can also adopt a simpler solution: obtain missing vertices by reflection whenever the butterfly stencil is incomplete, and always use the standard Butterfly rule, when there is no adjacent interior extraordinary vertex. This approach however results in

73

visible singularities. For completeness, we describe a set of rules that ensure C^1-continuity, as these rules were not previously published.

Boundary Rules. The rules extending the Butterfly scheme to meshes with boundary are somewhat more complex, because the stencil of the Butterfly scheme is larger. A number of different cases have to be considered separately: first, there is a number of ways in which one can chop off triangles from the butterfly stencil; in addition, the neighbors of the vertex that we are trying to compute can be either regular or extraordinary.

A complete set of rules for a mesh with boundary (up to head-tail permutations), includes 7 types of rules: regular interior, extraordinary interior, regular interior-crease, regular crease-crease 1, regular crease-crease 2, crease, and extraordinary crease neighbor; see Figures 4.5, 4.6, and 4.7. To put it all into a system, the main cases can be classified by the types of head and tail vertices of the edge on which we add a new vertex.

Recall that an interior vertex is a regular if its valence is 6, and a crease vertex is regular if its valence is 4. The following table shows how the type of rule to be applied to compute a *non-crease* vertex is determined from the valence of the adjacent vertices and whether they are on a crease or not. As we have already mentioned, the 4-point rule is used to compute new crease vertices. The only case when additional information is necessary, is when both neighbors are regular crease vertices. In this case the decision is based on the number of crease edges of the adjacent triangles (Figure 4.6).

Head	Tail	Rule
regular interior	regular interior	standard rule
regular interior	regular crease	regular interior-crease
regular crease	regular crease	regular crease-crease 1 or 2
extraordinary interior	extraordinary interior	average two extraordinary rules
extraordinary interior	extraordinary crease	same
extraordinary crease	extraordinary crease	same
regular interior	extraordinary interior	interior extraordinary
regular interior	extraordinary crease	crease extraordinary
extraordinary interior	regular crease	interior extraordinary
regular crease	extraordinary crease	crease extraordinary

74

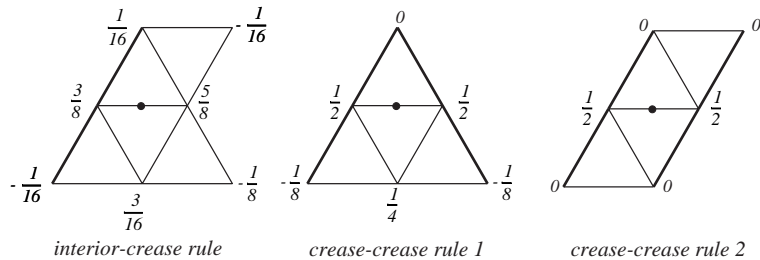interior-crease rule   crease-crease rule 1   crease-crease rule 2

Figure 4.6: *Regular Modified Butterfly boundary/crease rules.*

The extraordinary crease rule (Figure 4.7) uses coefficients c_{ij}, $j = 0 \ldots k$, to compute the vertex number i in the ring, when counted from the boundary. Let $\theta_k = \pi/(k-1)$. The following formulas define c_{ij} :

$$c_0 = 1 - \big(1/(k-1)\big)\sin\theta_k \sin i\theta_k /(1 - \cos\theta_k)$$
$$c_{i0} = c_{ik} = 1/4\cos i\theta_k - \big(1/4(k-1)\big)\sin 2\theta_k \sin 2\theta_k i/\big(\cos\theta_k - \cos 2\theta_k\big)$$
$$c_{ij} = (1/k)\big(\sin i\theta_k \sin j\theta_k + (1/2)\sin 2i\theta_k \sin 2j\theta_k\big)$$

Figure 4.7: *Modified Butterfly rules for neighbors of a crease/boundary extraordinary vertex.*

4.4 Catmull-Clark Scheme

The Catmull-Clark scheme was described in [4]. It is based on the tensor product bicubic spline. The masks are shown in Figure 4.8. The scheme produces surfaces that are C^2 everywhere except at extraordinary vertices, where they are C^1. The tangent plane continuity of the scheme was analyzed by Ball and Storry [1], and C^1-continuity by Peters and Reif [18]. The values of α and β can be chosen from a wide range (see Figure 4.10). On the boundary, using the coefficients for the cubic spline produces acceptable

75

results, however, the resulting surface formally is not C^1-continuous. A modification similar to the one performed in the case of Loop subdivision makes the scheme C^1-continuous (Figure 4.9). Again, a better, although a bit more complicated choice of coefficients is $\frac{3}{8} + \frac{1}{4}\cos\frac{2\pi}{k-1}$ instead of $\frac{5}{8}$ and $\frac{3}{8} - \frac{1}{4}\cos\frac{2\pi}{k-1}$ instead of $\frac{1}{8}$. See [38] for further details about the behavior on the boundary.
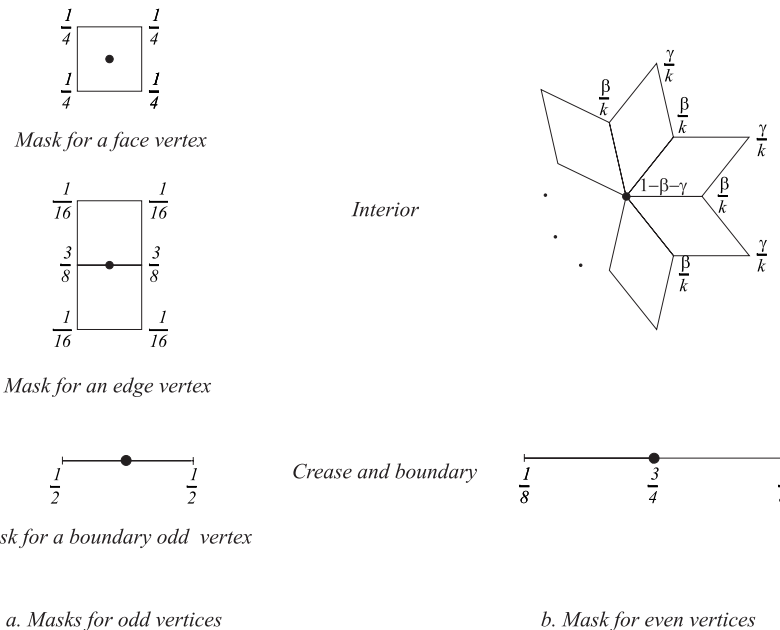

Mask for a face vertex

Interior

Mask for an edge vertex

Crease and boundary

Mask for a boundary odd vertex

a. Masks for odd vertices   b. Mask for even vertices

Figure 4.8: *Catmull-Clark subdivision. Catmull and Clark [4] suggest the following coefficients for rules at extraordinary vertices:* $\beta = \frac{3}{2k}$ *and* $\gamma = \frac{1}{4k}$

The rules of Catmull-Clark scheme are defined for meshes with quadrilateral faces. Arbitrary polygonal meshes can be reduced to a quadrilateral mesh using a more general form of Catmull-Clark rules [4]:

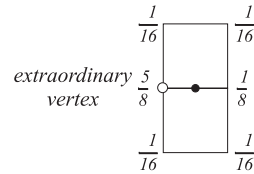- a face control point for an n-gon is computed as the average of the corners of the polygon;

75

75

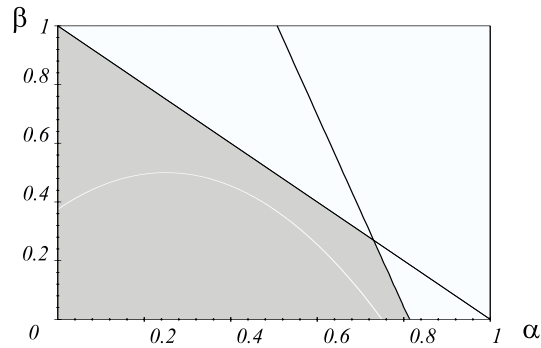Figure 4.9: *Modified rule for odd vertices adjacent to a boundary extraordinary vertex (Catmull-Clark scheme).*

Figure 4.10: *Ranges for coefficients* α *and* β *of the Catmull-Clark scheme;* $\alpha = 1 - \gamma - \beta$ *is the coefficient of the central vertex.*

- an edge control point as the average of the endpoints of the edge and newly computed face control points of adjacent faces;

- the formula for even control points can be chosen in different ways; the original formula is

$$p_0^{j+1} = \frac{k-2}{k}p_0^j + \frac{1}{k^2}\sum_{i=0}^{k-1} p_{i,1}^j + \frac{1}{k^2}\sum_{i=0}^{k-1} p_{i,2}^{j+1}$$

using the notation of Figure 4.2. Note that face control points on level $j+1$ are used.

4.5   Kobbelt Scheme

This interpolating scheme was described by Kobbelt in [11]. For regular meshes, it reduces to the tensor product of the four point scheme. C^1-continuity of this scheme for interior vertices for all valences is proven in [36].

Mask for a face vertex

Mask for edge, crease and boundary vertices

a. Regular masks

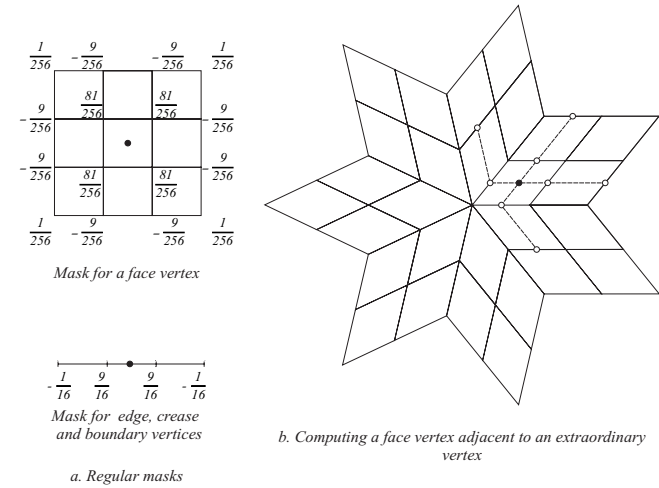*b. Computing a face vertex adjacent to an extraordinary vertex*

Figure 4.11:   *Kobbelt subdivision.*

Crucial for the construction of this scheme is the observation (valid for any tensor-product scheme) that the face control points can be computed in two steps: first, all edge control points are computed. Next, face vertices are computed using the *edge rule* applied to a sequence of edge control points on the same level. As shown in Figure 4.11, there are two ways to compute a face vertex in this way. In the regular case, the result is the same. Assuming this method of computing all face control points, only one rule of the regular scheme is modified: the edge odd control points adjacent to an extraordinary vertex

are computed differently. Specifically,

$$p_{i,1}^{j+1} = (\frac{1}{2} - w)p_0^j + (\frac{1}{2} - w)p_{i,1}^j + wp_i^j + wp_{i,3}^j$$

$$v_i^j = \frac{4}{k}\sum_{i=0}^{k-1} p_{i,1}^j - (p_{i-1,1}^j + p_{i,1}^j + p_{i+1,1}^j) - \frac{w}{1/2 - w}(p_{i-2,2}^j + p_{i-1,2}^j + p_{i,2}^j + p_{i+1,2}^j) + \frac{4w}{(1/2 - w)k}\sum_{i=0}^{k-1} p_{i,2}^j$$

(4.4)

where $w = -1/16$ (also, see Figure 4.2 for notation). On the boundaries and creases, the four point subdivision rule is used.

Unlike other schemes, eigenvectors of the subdivision matrix cannot be computed explicitly; hence, there are no precise expressions for tangents. In any case, the effective support of this scheme is too large for such formulas to be of practical use: typically, it is sufficient to subdivide several times and then use, for example, the formulas for the Loop scheme (see discussion in the section on the Loop scheme).

For more details on this scheme, see the part of the notes written by Leif Kobbelt.

4.6   Doo-Sabin and Midedge Schemes

Doo-Sabin subdivision is quite simple conceptually: there is no distinction between odd and even vertices, and a single mask is sufficient to define the scheme. A special rule is required only for the boundaries, where the limit curve is a quadratic spline. It was observed by Doo that this can also be achieved by replicating the boundary edge, i.e., creating a quadrilateral with two coinciding pairs of vertices. Nasri [17] describes other ways of defining rules for boundaries. The rules for the Doo-Sabin scheme are shown in Figure 4.12. C^1-continuity for schemes similar to the Doo-Sabin schemes was analyzed by Peters and Reif [18].

An even simpler scheme was proposed by Habib and Warren [9] and by Peters and Reif [19]: this scheme uses even smaller stencils than the Doo-Sabin scheme; for regular vertices, only three control points are used (Figure 4.13).

A remarkable property of both Midedge and Doo-Sabin subdivision is that the interior rules, at least in the regular case, can be decomposed into a sequence of averaging steps, as shown in Figures 4.14 and Figures 4.15

In both cases the averaging procedure generalizes to arbitrary meshes. However, the edge averaging procedure, as it was established in [19], does not result in well-behaved surfaces, when applied to arbitrary meshes. In contrast, centroid averaging, when applied to arbitrary meshes, results precisely in the
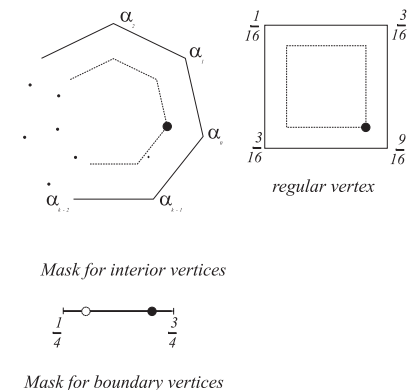
Mask for interior vertices

Mask for boundary vertices

Figure 4.12:   *Doo-Sabin subdivision. The coefficients are defined by the formulas* $\alpha_0 = 1/4 + 5/4k$ *and* $\alpha_i = (3 + 2\cos(2i\pi/k))/4k$, *for* $i = 1 \ldots k - 1$. *Another choice of coefficients was proposed by Catmull and Clark:* $\alpha_0 = 1/2 + 1/4k$, $\alpha_1 = \alpha_{k-1} = 1/8 + 1/4k$, *and* $\alpha_i = 1/4k$ *for* $i = 2 \ldots k - 2$.

Catmull-Clark variant of the Doo-Sabin scheme. Another important observation is that centroid averaging can be applied more than once. This idea provides us with a different view of a class of quadrilateral subdivision schemes, which we now discuss in detail.

4.7   Uniform Approach to Quadrilateral Subdivision

As we have observed in the previous section, the Doo-Sabin scheme can be represented as midpoint subdivision followed by a centroid averaging step. What if we apply the centroid averaging step one more time? The result is a primal subdivision scheme, in the regular case coinciding with Catmull-Clark. In the irregular case the stencil of the resulting scheme is the same as the stencil of Catmull-Clark, but the coefficients α and β used in the vertex rule are different. However, the new coefficients also result in a well-behaved scheme producing surfaces only slightly different from Catmull-Clark.

Clearly, we can apply the centroid averaging to midpoint-subdivided mesh any number of times, obtaining in the regular case splines of higher and higher degree. Similar observations were made independently by a number of people: [34, 29, 30].

For arbitrary meshes we will get subdivision schemes which have higher smoothness away from iso-
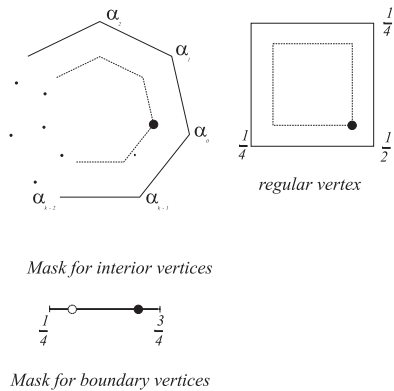
Mask for interior vertices

Mask for boundary vertices

Figure 4.13: *Midedge subdivision. The coefficients are defined by the formulas* $\alpha_i = 2\sum_{j=0}^{\bar{n}} 2^{-ji} \cos\frac{2\pi i j}{k}$, $\bar{n} = \lfloor\frac{n-1}{2}\rfloor$ *for* $i = 0\ldots k-1$
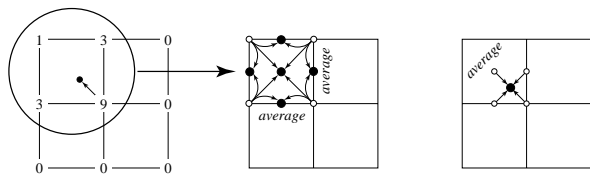

Figure 4.14: *The subdivision stencil for Doo-Sabin subdivision in the regular case (left). It can be understood as midpoint subdivision followed by averaging. At the averaging step the centroid of each face is computed; then the barycenters are connected to obtain a new mesh. This procedure generalizes without changes to arbitrary meshes.*

lated points on the surface. Unfortunately, smoothness at the extraordinary vertices (for primal schemes) and at the centroids of faces (for dual schemes) remains, in general, C^1.

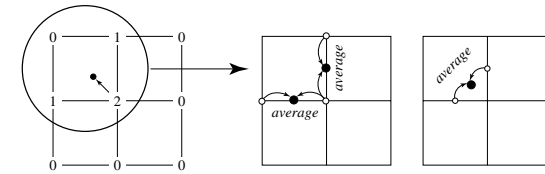Our observations are summarized in the following table:

Figure 4.15: *The subdivision stencil for Midedge subdivision in the regular case (left). It can be understood as a sequence of averaging steps; at each step, two vertices are averaged.*

centroid averaging steps	scheme	smoothness in regular case
0	midpoint	C^0
1	Doo-Sabin	C^1
2	Catmull-Clark	C^2
3	Bi-Quartic	C^3
4	…	…

Biquartic subdivision scheme is a new dual scheme that is obtained by applying three centroid averaging steps after midpoint subdivision, as illustrated in Figure 4.16. As this scheme was not discussed before, we discuss it in greater detail here.

Generalized Biquartic Subdivision. The centroid averaging steps provide a nice theoretical way of deriving a new scheme, however, in practice we may want to use the complete masks directly (in particular, if we have to implement adaptive subdivision). Figure 4.16 shows the support of the stencil for Biquartic b-spline subdivision in the regular case (leftmost stencil).

Note that Biquartic subdivision can be implemented with very little additional work, compared to Doo-Sabin or Midedge. In an implementation of dual subdivision, vertices are organized as quadtrees. It is then natural to compute all four children of a given vertex at the same time. Considering the stencils for Doo-Sabin or the Midedge scheme we see that this implies access to all vertices of the faces incident to a given vertex. If these vertices have to be accessed we may as well use non-zero coefficients for all of them for each child to be computed. Qu [23] was the first to consider a generalization of the Biquartic B-splines to the arbitrary topology setting. He derived some conditions on the stencils but did not give a concrete set of coefficients. Repeated centroid averaging provides a simple way to derive the coefficients. It is possible to show that the resulting scheme is C^1 at extraordinary vertices. Assuming that only one of the incident faces for a vertex is extraordinary, we can write the subdivision masks for
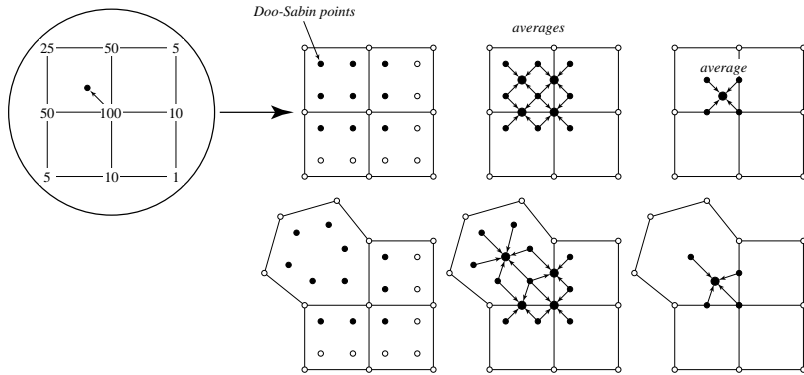
Figure 4.16: *The subdivision stencil for bi-quartic b-splines (top row for the regular setting) can be written as a sequence of averaging steps. In a first step Doo-Sabin points are computed. These are subsequently averaged twice to arrive at the final point. This effects a factorization of the original mask (left) into a sequence of pure averaging steps. The same procedure is repeated using as an example a setting in which one incident face has valence $\neq 4$ (bottom row).*

vertices near extraordinary faces in a more explicit form. There are three different masks for the four children (Figure 4.17). This is in contrast to the Doo-Sabin and Midedge schemes which have only one mask type for all children (modulo rotation). Vertices incident to the extraordinary faces contribute

Figure 4.17: *Generalized Biquartic compound masks for the north-west (nw), north-east (ne), and south-east (se) children of the center vertex. The south-west mask is the reflected (along the diagonal) version of the ne mask. All weights must be normalized by $1/256$ and the weights for the extraordinary vertices must be added. They are given in equation 4.5.*

additional weights as

$$
\begin{aligned}
nw_i &= \frac{64}{k} + 48w_i + 16w_{i-1} + 16w_{i+1} \\
ne_i &= 32w_i + 16w_{i-1} \\
se_i &= 16w_i,
\end{aligned}
\tag{4.5}
$$

where w_i are the Doo-Sabin weights, $i = 0, \dots, k-1$ and indices are taken modulo k.

4.8 Comparison of Schemes

In this section we compare different schemes by applying to a variety of meshes. First, we consider Loop, Catmull-Clark, Modified Butterfly and Doo-Sabin subdivision.

Figure 4.18 shows the surfaces obtained by subdividing a cube. Not surprisingly, Loop and Catmull-Clark subdivision produce more pleasing surfaces, as these schemes reduce to C^2 splines on a regular mesh. As all faces of the cube are quads, Catmull-Clark yields the nicest surface; the surface generated by the Loop scheme is more asymmetric, because the cube had to be triangulated before the scheme could be applied. At the same time, Doo-Sabin and Modified Butterfly reproduce the shape of the cube more closely. The surface quality is worst for the Modified Butterfly scheme, which interpolates the original mesh. We observe that there is a tradeoff between interpolation and surface quality: the closer the surface is to interpolating, the lower the surface quality.

Figure 4.19 shows the results of subdividing a tetrahedron. Similar observations hold in this case. In addition, we observe extreme shrinking for the Loop and Catmull-Clark subdivision schemes. This is a characteristic feature of approximating schemes: for small meshes, the resulting surface is likely to occupy much smaller volume than the original control mesh.

Finally, Figure 4.20 demonstrates that for sufficiently "smooth" meshes, with uniform triangle size and sufficiently small angles between adjacent faces, different schemes may produce virtually indistinguishable results. This fact might be misleading however, especially when interpolating schemes are used; interpolating schemes are very sensitive to the presence of sharp features and may produce low quality surfaces for many input meshes unless an initial mesh smoothing step is performed.

Overall, Loop and Catmull-Clark appear to be the best choices for most applications, which do not require exact interpolation of the initial mesh. The Catmull-Clark scheme is most appropriate for meshes with a significant fraction of quadrilateral faces. It might not perform well on certain types of meshes, most notably triangular meshes obtained by triangulation of a quadrilateral mesh (see Figure 4.21). The

Loop scheme performs reasonably well on any triangular mesh, thus, when triangulation is not objectionable, this scheme might be preferable. There are two main reasons why a quadrilateral scheme may be preferable: natural texture mapping for quads, and a natural number of symmetries (2). Indeed, many objects and characters have two easily identifiable special directions ("along the axis of the object" and "perpendicular to the axis"). The mesh representing the object can be aligned with these directions. Objects with three natural directions, that can be used to align a triangular mesh with the object, are much less common.

Loop   *Butterfly*

Catmull-Clark   *Doo-Sabin*

Figure 4.18: *Results of applying various subdivision schemes to the cube. For triangular schemes (Loop and Butterfly) the cube was triangulated first.*
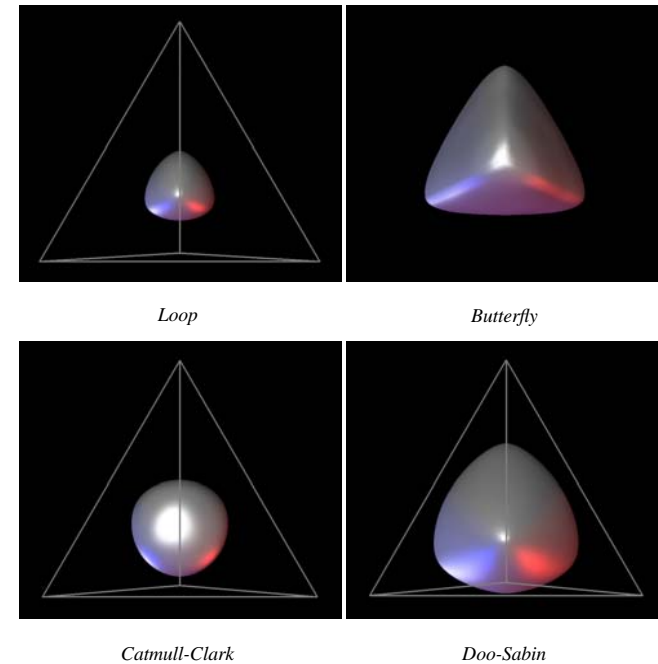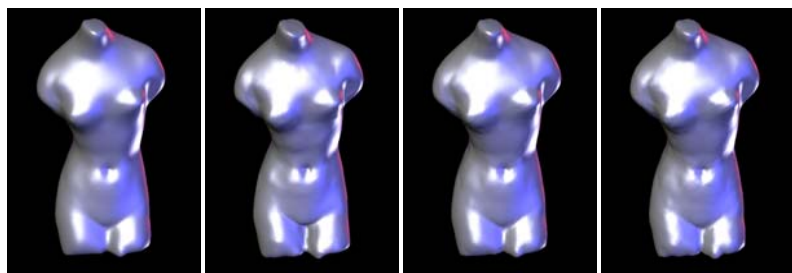
Loop   *Butterfly*

Catmull-Clark   *Doo-Sabin*

Figure 4.19: *Results of applying various subdivision schemes to a tetrahedron.*

4.8.1 Comparison of Dual Quadrilateral Schemes

Dual quadrilateral schemes are the only class of schemes with several members: Doo-Sabin, Midedge, Biquartic. In this section we give some numerical examples comparing the behavior of different dual quadrilateral subdivision schemes.

Much about a subdivision scheme is revealed by looking at the associated basis functions, i.e., the result of subdividing an initial control mesh which is planar except for a single vertex which is pulled out of the plane. Figure 4.22 shows such basis functions for Midedge, Doo-Sabin, and the Biquartic scheme in the vicinity of a k-gon for $k = 4$ and $k = 9$. Note how the smoothness increases with higher order. The

Loop	*Butterfly*	*Catmull-Clark*	*Doo-Sabin*

Figure 4.20: *Different subdivision schemes produce similar results for smooth meshes.*

Initial mesh	*Loop*	*Catmull-Clark*	*Catmull-Clark,after triangulation*

Figure 4.21: *Applying Loop and Catmull-Clark subdivision schemes to a model of a chess rook. The initial mesh is shown on the left. Before the Loop scheme was applied, the mesh was triangulated. Catmull-Clark was applied to the original quadrilateral model and to the triangulated model; note the substantial difference in surface quality.*

distinction is already apparent in the case $k = 4$, but becomes very noticeable for $k = 9$.

Figure 4.23 provides a similar comparison showing the effect of different dual quadrilateral subdivision schemes when the control polyhedron is a simple cube (compare to 4.18). Notice the increasing
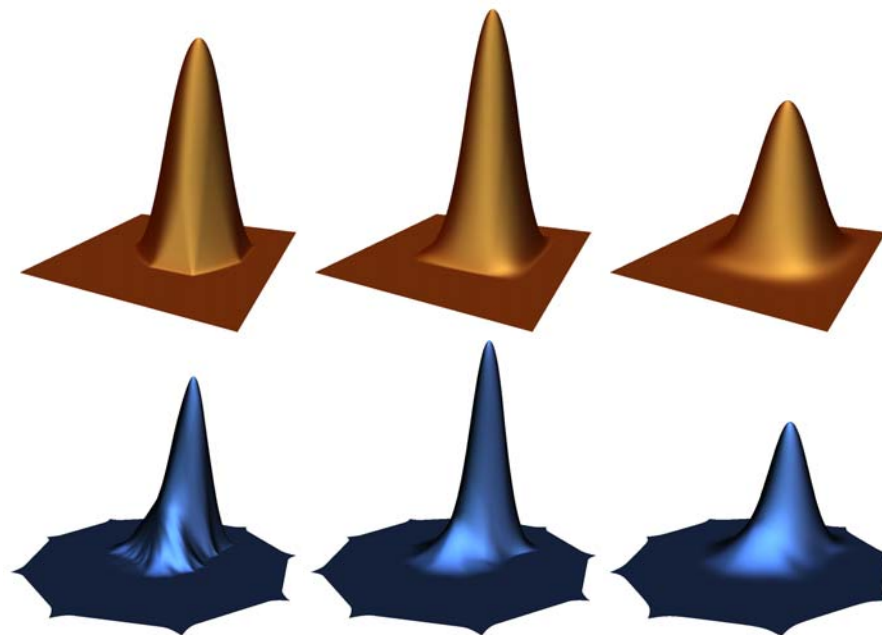
Figure 4.22: *Comparison of dual basis functions for a 4-gon (the regular case) on top and a 9-gon on the bottom. On the left the Midedge scheme (Warren/Habib variant), followed by the Doo-Sabin scheme and finally by the Biquartic generalization. The increasing smoothness is particularly noticeable in the 9-gon case.*

shrinkage with increasing smoothness. Since averages are convex combinations, the more averages are cascaded the more shrinkage can be expected.

Figure 4.24 shows a pipe shape with boundaries showing the effect of boundaries in the case of Midedge, Doo-Sabin and the Biquartic scheme.

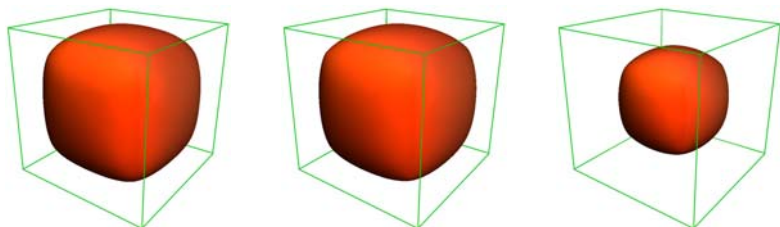Finally, Figure 4.25 shows the control mesh, limit surface and an adaptive tesselation blowup for a head shape.

Figure 4.23: *Comparison of dual subdivision schemes (Midedge, Doo-Sabin, Biquartic) for the case of a cube. The control polyhedron is shown in outline. Notice how Doo-Sabin and even more so the Biquartic scheme exhibit considerable shrinkage in this case, while the difference between Midedge and Doo-Sabin is only slight in this example.*
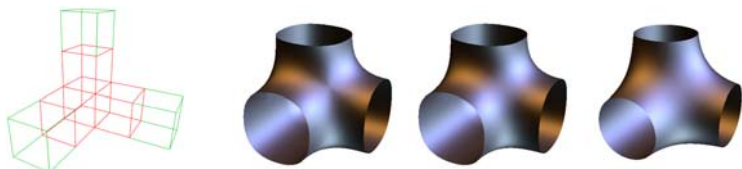

Figure 4.24: *Control mesh for a three legged pipe (left). The red parts denote the control mesh for Midedge and Doo-Sabin, while the additional green section is necessary to have a complete set of boundary conditions for the bi-quartic scheme. The resulting surfaces in order: Midedge, Doo-Sabin, and Biquartic. Note the pinch point visible for Midedge and the increasing smoothness and roundness for Doo-Sabin and Biquartic.*

4.9 Tilings

The classification that we have described in the beginning of the chapter, captures most known schemes. However, new schemes keep appearing, and some of the recent schemes do not fit well into this classification. It can be easily extended to handle a greater variety of schemes, if we include other refinement rules, in addition to vertex and face splits.

The starting point for refinement rules are the *isohedral tilings* and their dual tilings. A tiling is called isohedral, or Laves, if all tiles are identical, and for any vertex the angles between successive edges meeting at the vertex are equal.

In general, there are 11 tilings of the plane, shown in Figure 4.26; their dual tilings, obtained by con-
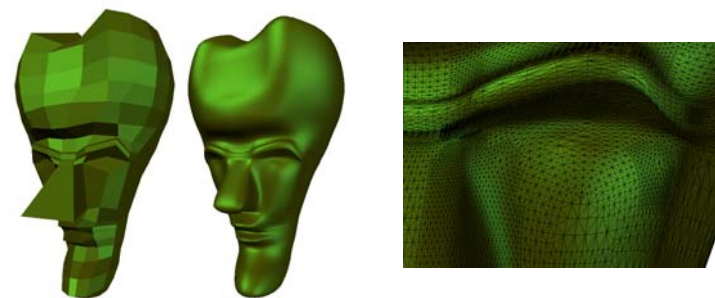
Figure 4.25: *An example of adaptive subdivision. On the left the control mesh, in the middle the smooth shaded limit surface and on the right a closeup of the adaptively triangulated limit surface.*

necting the centers of the tiles are called Archimedean tilings, and are shown in Figure 4.27. Archimedean tilings consist of regular polygons. We will refer to Laves and Archimedean tilings as regular tilings. Generalizing the idea of refinement rules to arbitrary regular tilings, we say that a refinement rule is an algorithm to obtain a finer regular tiling of the same type from a given regular tiling. This definition is quite general, and it is not known what all possible refinement rules are. The finer tiling is a scaled version of the initial tiling; the scaling factor can be arbitrary. For vertex and face splits, it is 2.

In practice, we are primarily interested in refinement rules that generalize well to arbitrary meshes. Face and vertex splits are examples of such rules. Three more exotic refinement rules have been considered: honeycomb refinement, $\sqrt{3}$ refinement and bisection.

Honeycomb refinement [8] shown in Figure 4.28, can be regarded as dual to the face split applied to the triangular mesh. While it is possible to design stationary schemes for honeycomb refinement, the scheme described in [8] is not stationary.

The $\sqrt{3}$ refinement [12], when applied to the regular triangulation of the plane (3^6 tiling), produces a tiling scaled by the factor $\sqrt{3}$ (Figure 4.29). The subdivision scheme described in [12] is stationary and produces C^2 subdivision surfaces on regular meshes.

Bisection, a well-known refinement technique often used for finite-element mesh refinement, can be used to refine $4 - k$ meshes [32, 31]. The refinement process for the regular 4.8^2 tiling is illustrated in Figure 4.30. Note that a single refinement step results in a new tiling scaled by $\sqrt{2}$. As shown in [30], Catmull-Clark and Doo-Sabin subdivision schemes, as well as some higher order schemes based on face
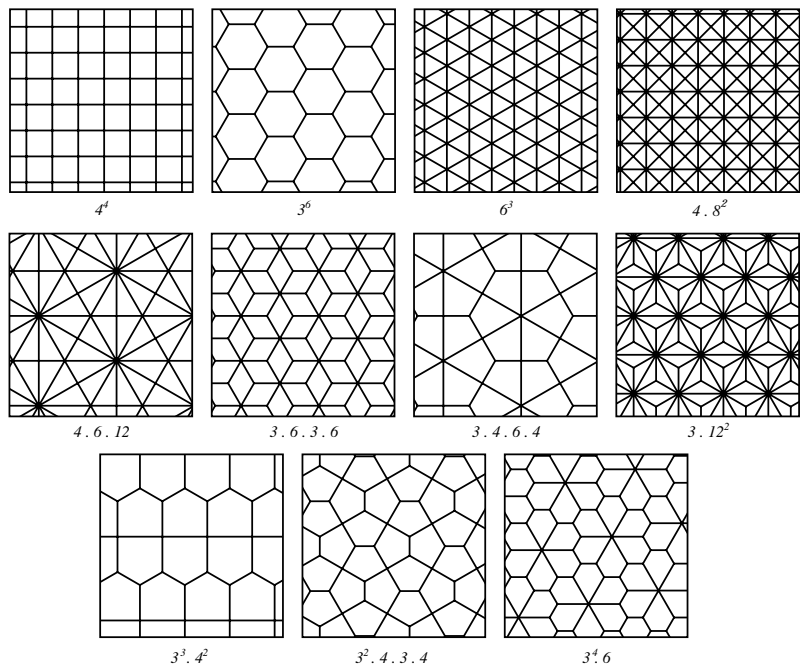
Figure 4.26: *11 Laves (isohedral) tilings.*

or vertex splits, can be decomposed into sequences of bisection refinement steps. Both $\sqrt{3}$ and $4-k$ subdivision have the advantage of approaching the limit surface more gradually. At each subdivision step, the number of triangles triples and doubles respectively, rather then quadruple, as is the case for face split refinement. This allows finer control of the approximation. In addition, adaptive subdivision can be easier to implement, if edge-based data structures are used to represent meshes (see also Chapter 5).
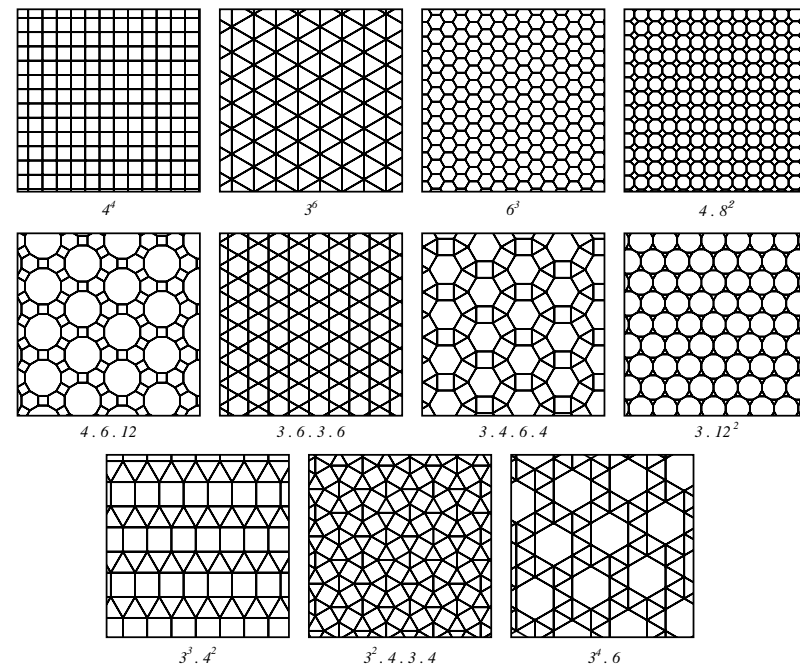
Figure 4.27: *11 Archimedean tilings, dual to Laves tilings.*

4.10   Limitations of Stationary Subdivision

Stationary subdivision, while overcoming certain problems inherent in spline representations, still has a number of limitations. Most problems are much more apparent for interpolating schemes than for approximating schemes. In this section we briefly discuss a number of these problems.
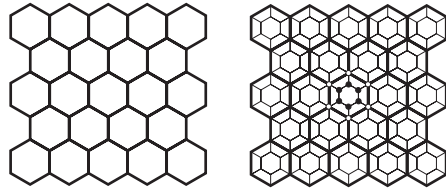
Figure 4.28: *Honeycomb refinement. Old vertices are preserved, and 6 new vertices are inserted for each face.*
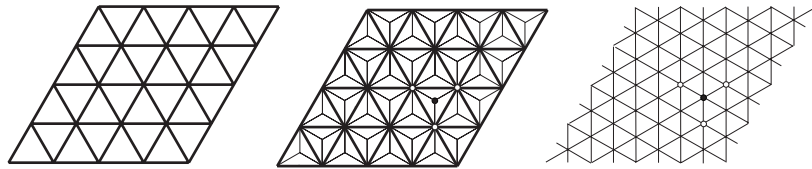

Figure 4.29: $\sqrt{3}$ *refinement. The barycenter is inserted into each triangle; this results in a* 3.12^2 *tiling. Then the edges are are flipped, to produce a new* 3^6 *tiling, which is scaled by* $\sqrt{3}$ *and rotated by 30 degrees with respect to the original.*
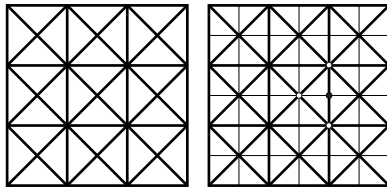

Figure 4.30: *Bisection on a 4-8 tiling: the hypotenuse of each triangle is split. The resulting tiling is a new 4-8 mesh, shrunk by* $\sqrt{2}$ *and rotated by 45 degrees.*

Problems with Curvature Continuity. While it is possible to obtain subdivision schemes which are C^2-continuous, there are indications that such schemes either have very large support [24, 21], or necessarily have zero curvature at extraordinary vertices. A compromise solution was recently proposed by Umlauf [22]. Nevertheless, this limitation is quite fundamental: degeneracy or discontinuity of curvature

typically leads to visible defects of the surface.

Decrease of Smoothness with Valence. For some schemes, as the valence increases, the magnitude of the third largest eigenvalue approaches the magnitude of the subdominant eigenvalues. As an example we consider surfaces generated by the Loop scheme near vertices of high valence. In Figure 4.31 (right
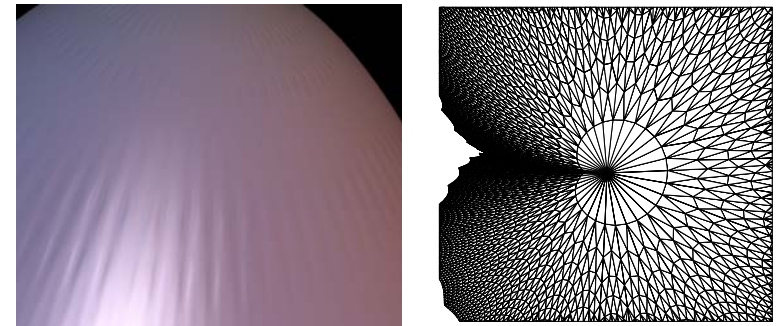

Figure 4.31: *Left: ripples on a surface generated by the Loop scheme near a vertex of large valence; Right: mesh structure for the Loop scheme near an extraordinary vertex with a significant "high-frequency" component; a crease starting at the extraordinary vertex appears.*

side), one can see a typical problem that occurs because of "eigenvalue clustering:" a crease might appear, abruptly terminating at the vertex. In some cases this behavior may be desirable, but our goal is to make it controllable rather than let the artifacts appear by chance.

Ripples. Another problem, presence of ripples in the surface close to an extraordinary point, is also shown in Figure 4.31. It is not clear whether this artifact can be eliminated. It is closely related to the curvature problem.

Uneven Structure of the Mesh. On regular meshes, subdivision matrices of C^1-continuous schemes always have subdominant eigenvalue $1/2$. When the eigenvalues of subdivision matrices near extraordinary vertices significantly differ from $1/2$, the structure of the mesh becomes uneven: the ratio of the size of triangles on finer and coarser levels adjacent to a given vertex is roughly proportional to the magnitude of the subdominant eigenvalue. This effect can be seen clearly in Figure 4.33.

Optimization of Subdivision Rules. It is possible to eliminate eigenvalue clustering, as well as the difference in eigenvalues of the regular and extraordinary case by prescribing the eigenvalues of the subdivision matrix and deriving suitable subdivision coefficients. This approach was used to derive coefficients of the Butterfly scheme.

As expected, the meshes generated by the modified scheme have better structure near extraordinary points (Figure 4.32). However, the ripples become larger, so one kind of artifact is traded for another. It is, however, possible to seek an optimal solution or one close to optimal; alternatively, one may resort to a family of schemes that would provide for a controlled tradeoff between the two artifacts.
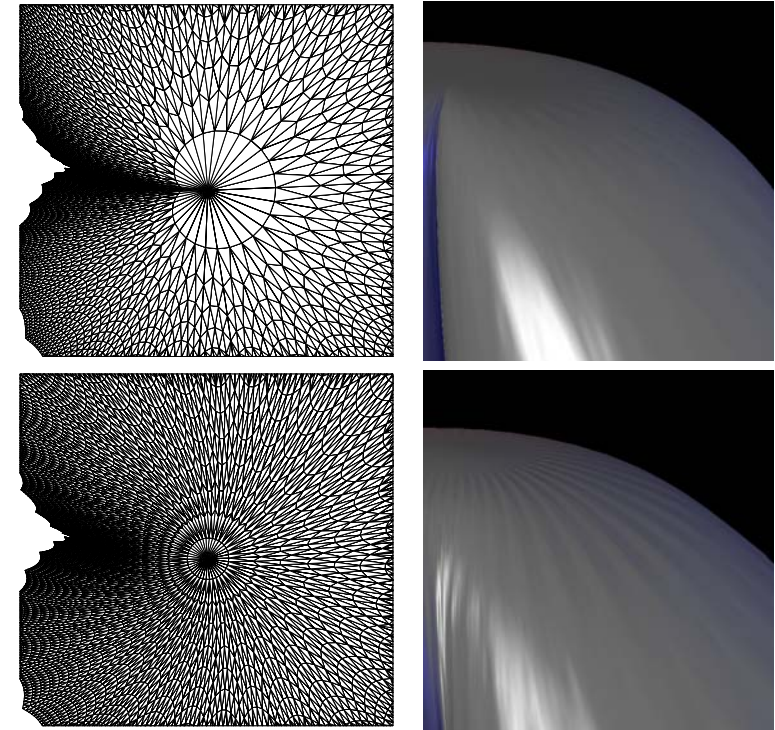

Figure 4.32: *Left: mesh structure for the Loop scheme and the modified Loop scheme near an extraordinary vertex; a crease does not appear for the modified Loop. Right: shaded images of the surfaces for Loop and modified Loop; ripples are more apparent for modified Loop.*
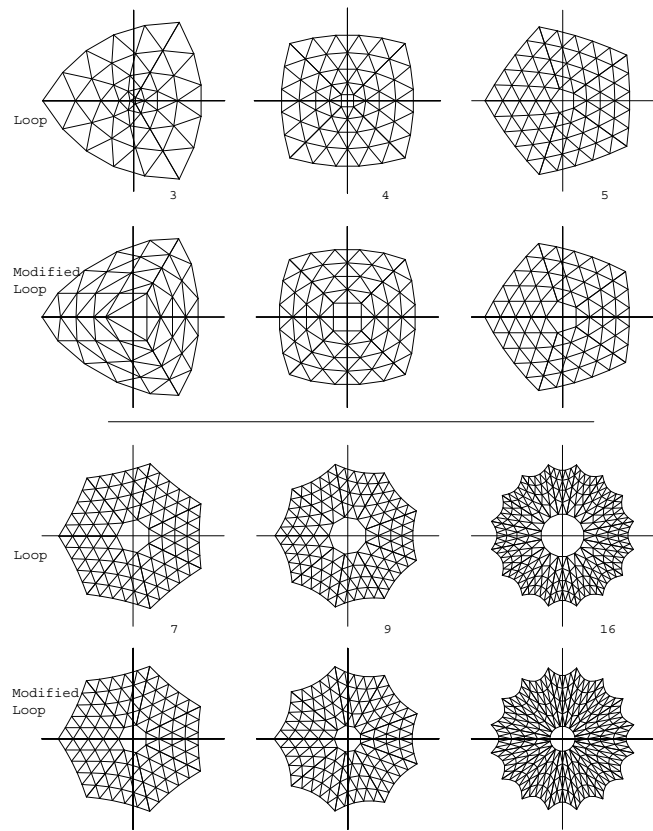
Figure 4.33: *Comparison of control nets for the Loop and modified Loop scheme. Note that for the Loop scheme the size of the hole in the ring (1-neighborhood removed) is very small relative to the surrounding triangles for valence 3 and becomes larger as k grows. For the modified Loop scheme this size remains constant.*

Bibliography

[1] BALL, A. A., AND STORRY, D. J. T. Conditions for Tangent Plane Continuity over Recursively Generated B-Spline Surfaces. *ACM Trans. Gr. 7*, 2 (1988), 83–102.

[2] BIERMANN, H., LEVIN, A., AND ZORIN, D. Piecewise smooth subdivision surfaces with normal control. Tech. Rep. TR1999-781, NYU, 1999.

[3] BIERMANN, H., LEVIN, A., AND ZORIN, D. Piecewise smooth subdivision surfaces with normal control. In *SIGGRAPH 2000 Conference Proceedings*, Annual Conference Series, July 2000.

[4] CATMULL, E., AND CLARK, J. Recursively Generated B-Spline Surfaces on Arbitrary Topological Meshes. *Computer Aided Design 10*, 6 (1978), 350–355.

[5] DOO, D., AND SABIN, M. Analysis of the Behaviour of Recursive Division Surfaces near Extraordinary Points. *Computer Aided Design 10*, 6 (1978), 356–360.

[6] DYN, N., GREGORY, J. A., AND LEVIN, D. A Four-Point Interpolatory Subdivision Scheme for Curve Design. *Comput. Aided Geom. Des. 4* (1987), 257–268.

[7] DYN, N., LEVIN, D., AND GREGORY, J. A. A Butterfly Subdivision Scheme for Surface Interpolation with Tension Control. *ACM Trans. Gr. 9*, 2 (April 1990), 160–169.

[8] DYN, N., LEVIN, D., AND LIU, D. Interpolatory convexity-preserving subdivision for curves and surfaces. *Computer-Aided Design 24*, 4 (1992), 211–216.

[9] HABIB, A., AND WARREN, J. Edge and Vertex Insertion for a Class of C^1 Subdivision Surfaces. presented at 4th SIAM COnference on Geometric Design, November 1995.

[10] HOPPE, H., DEROSE, T., DUCHAMP, T., HALSTEAD, M., JIN, H., MCDONALD, J., SCHWEITZER, J., AND STUETZLE, W. Piecewise Smooth Surface Reconsruction. In *Computer Graphics Proceedings*, Annual Conference Series, 295–302, 1994.

[11] KOBBELT, L. Interpolatory Subdivision on Open Quadrilateral Nets with Arbitrary Topology. In *Proceedings of Eurographics 96*, Computer Graphics Forum, 409–420, 1996.

[12] KOBBELT, L. $\sqrt{3}$ Subdivision. *Computer Graphics* Proceedings, Annual Conference Series, 2000.

[13] LEVIN, A. Boundary algorithms for subdivision surfaces. In *Israel-Korea Bi-National Conference on New Themes in Computerized Geometrical Modeling*, 117–121, 1998.

[14] LEVIN, A. Combined subdivision schemes for the design of surfaces satisfying boundary conditions. To appear in CAGD, 1999.

[15] LEVIN, A. Interpolating nets of curves by smooth subdivision surfaces. to appear in SIGGRAPH'99 proceedings, 1999.

[16] LOOP, C. Smooth Subdivision Surfaces Based on Triangles. Master's thesis, University of Utah, Department of Mathematics, 1987.

[17] NASRI, A. H. Polyhedral Subdivision Methods for Free-Form Surfaces. *ACM Trans. Gr. 6*, 1 (January 1987), 29–73.

[18] PETERS, J., AND REIF, U. Analysis of generalized B-spline subdivision algorithms. *SIAM Jornal of Numerical Analysis* (1997).

[19] PETERS, J., AND REIF, U. The simplest subdivision scheme for smoothing polyhedra. *ACM Trans. Gr. 16(4)* (October 1997).

[20] PRAUTZSCH, H. Analysis of C^k-subdivision surfaces at extraordianry points. Preprint. Presented at Oberwolfach, June, 1995, 1995.

[21] PRAUTZSCH, H., AND REIF, U. Necessary Conditions for Subdivision Surfaces. 1996.

[22] PRAUTZSCH, H., AND UMLAUF, G. A G^2-Subdivision Algorithm. In *Geometric Modeling*, G. Farin, H. Bieri, G. Brunnet, and T. DeRose, Eds., vol. Computing Suppl. 13. Springer-Verlag, 1998, pp. 217–224.

[23] QU, R. *Recursive Subdivision Algorithms for Curve and Surface Design*. PhD thesis, Brunel University, 1990.

[24] REIF, U. A Degree Estimate for Polynomial Subdivision Surface of Higher Regularity. Tech. rep., Universität Stuttgart, Mathematisches Institut A, 1995. preprint.

[25] REIF, U. Some New Results on Subdivision Algorithms for Meshes of Arbitrary Topology. In *Approximation Theory VIII*, C. K. Chui and L. Schumaker, Eds., vol. 2. World Scientific, Singapore, 1995, pp. 367–374.

[26] REIF, U. A Unified Approach to Subdivision Algorithms Near Extraordinary Points. *Comput. Aided Geom. Des. 12* (1995), 153–174.

[27] SAMET, H. *The Design and Analysis of Spatial Data Structures*. Addison-Wesley, 1990.

[28] SCHWEITZER, J. E. *Analysis and Application of Subdivision Surfaces*. PhD thesis, University of Washington, Seattle, 1996.

[29] STAM, J. On Subdivision Schemes Generalizing Uniform B-Spline Surfaces of Arbitrary Degree. Submitted for Publication, 2000.

[30] VELHO, L., AND GOMES, J. Decomposing Quadrilateral Subdivision Rules into Binary 4–8 Refinement Steps. `http://www.impa.br/~lvelho/h4k/`, 1999.

[31] VELHO, L., AND GOMES, J. Quasi 4-8 Subdivision Surfaces. In *XII Brazilian Symposium on Computer Graphics and Image Processing*, 1999.

[32] VELHO, L., AND GOMES, J. Semi-Regular 4-8 Refinement and Box Spline Surfaces. Unpublished., 2000.

[33] WARREN, J. Subdivision Methods for Geometric Design. Unpublished manuscript, November 1995.

[34] WARREN, J., AND WEIMER, H. Subdivision for Geometric Design. 2000.

[35] ZORIN, D. *Subdivision and Multiresolution Surface Representations*. PhD thesis, Caltech, Pasadena, 1997.

[36] ZORIN, D. A method for analysis of C^1-continuity of subdivision surfaces. *SIAM Journal of Numerical Analysis 37*, 4 (2000).

[37] ZORIN, D. Smoothness of subdivision on irregular meshes. *Constructive Approximation 16*, 3 (2000).

[38] ZORIN, D. Smoothness of subdivision surfaces on the boundary. preprint, Computer Science Department, New York University, 2000.

[39] ZORIN, D., SCHRÖDER, P., AND SWELDENS, W. Interpolating Subdivision for Meshes with Arbitrary Topology. *Computer Graphics Proceedings (SIGGRAPH 96)* (1996), 189–192.

[40] ZORIN, D., SCHRÖDER, P., AND SWELDENS, W. Interactive Multiresolution Mesh Editing. *Computer Graphics* Proceedings, Annual Conference Series, 1997.