Quasi 4–8 Subdivision

Luiz Velho

Visgraf Laboratory
IMPA – Instituto de Matemática Pura e Aplicada
Estrada Dona Castorina 110, Rio de Janeiro, RJ, Brazil, 22460-320.
`lvelho@impa.br`

Abstract

This paper presents a new scheme for subdivision surfaces based on four-directional meshes. It combines geometry-sensitive refinement with convolution smoothing. The scheme has a simple, efficient implementation and generates smooth well-shaped meshes.

Keywords: subdivision schemes, four-directional meshes, quincunx lattice

1 Introduction

In recent years subdivision surfaces have become one of the most important mathematical tools for shape modeling in Computer Graphics and Computer-Aided Geometric Design (CAGD). The main reason for their importance is due to the fact that they overcome some limitations of classical tools, such as NURBS (non-uniform rational B-splines).

Subdivision surfaces provide a natural generalization of spline surfaces with several advantages: they can handle control meshes of arbitrary topology; they guarantee global surface smoothness while making possible control of local features; they effectively bridge the gap between continuous models and discrete representations; and, lastly, they are associated with efficient algorithms that are simple to implement.

The power of a subdivision surface model lies in the quality of the underlying limit surface, as well as, in the applicability of the associated computational scheme. The various types of subdivision surfaces offer different compromises between these two aspects. In this paper, we present a new type of subdivision surface that achieves a good balance of such features.

1.1 Previous Work

Subdivision surfaces started as a generalization of uniform splines [7]. The very idea of a subdivision scheme draws upon *knot insertion* techniques [19], and have its roots on the "De Boor" algorithm [3]. Therefore, tensor product B-splines and Box splines can be viewed as a special case of subdivision surfaces on regular meshes [6,5].

Nonetheless, the beginning of the field is identified with the development of the first subdivision surfaces for irregular meshes. Catmull and Clark [4], and Doo and Sabin [9], extended bicubic and biquadratic B-splines, respectively, to arbitrary meshes that generalize quadrilateral meshes. Later, Loop [21], created a generalization of quartic 3-direction Box splines to arbitrary triangular meshes. Peters and Reif [24], and Habib and Warren [14], independently introduced schemes that generalize quadratic 4-direction Box splines on irregular rectangular meshes.

All the schemes mentioned so far result in surfaces that approximate a mesh. Dyn, Gregory and Levin [11], designed the "Butterfly" scheme for interpolating C^1 surfaces based on triangular meshes. This scheme was subsequently improved by Zorin, Schroeder and Sweldens [33]. Kobbelt [17] described a C^1 interpolating scheme for quadrilateral meshes with arbitrary topology.

Approximate subdivision schemes can be used to construct subdivision surfaces that interpolate the vertices of a control polyhedron, The process replaces the original control mesh by a new one that satisfies these interpolation constraints [23,15]. It is also possible to modify the subdivision scheme to interpolate a set of control curves [20,29,22].

Further work investigates control of surface features, such as creases, corners and normals. Hoppe, De Rose, et al. [16], proposed an extension of the Loop scheme for piecewise C^1 surfaces with feature control. Zorin, Biermann and Levin [2], introduced modified rules for Catmull-Clark and Loop subdivision that allows normal control. Prautzsch and Umlauf [26] improved the Butterfly and Catmull-Clark to generate G^1 and G^2 surfaces respectively.

A rigorous study of the convergence of subdivision schemes based on the characteristic map was proposed by Reif [27]. Another important theoretical contribution was given by Zorin [32] . Earlier work on this problem can be found in Doo and Sabin [10] and Ball and Storry [1] and Prautzsch [25] .

The stationary subdivision methods cited above still have a number of shortcomings: they are too rigid, and may exhibit undesirable behavior near extraordinary points. Variational subdivision schemes attempt to overcome such limitations. This issue has been addressed by Kobbelt [18] and by Warren [31].

1.2 Contribution

Convolution schemes that generalize Box splines employ carefully designed smoothing rules which depend on the local neighborhood structure of a vertex. Variational schemes on irregular meshes employ a smoothing energy functional which depends on the global neighborhood structure and its geometry.

It is interesting to observe that, in all existing schemes, most of the effort was concentrated on the design of the smoothing operator. In fact, almost all of them adopt the same standard refinement operators. One characteristic of these refinement operators is that they do not depend on the geometry of the mesh.

In this paper we describe a new subdivision scheme that uses a geometry-sensitive refinement operator in conjunction with a convolution smoothing operator. We call this scheme *Quasi 4–8 Subdivision*, because it induces a quasi-stationary subdivision process. Our scheme has a simple implementation and generates smooth surfaces that approximate the initial control mesh.

The structure of the paper is as follows: Section 2 provides a background on 4–8 meshes and uniform refinement of four directional grids. Section 3 describes the quasi-stationary 4–8 subdivision scheme proposed in this paper. Section 4 gives some examples of subdivision surfaces generated using quasi 4–8 subdivision. Section 5 concludes with a summary of our results and a discussion of future work.

2 Background

Before going into the details of our subdivision scheme and its operators, we provide some background on the 4–8 mesh structure,

2.1 4–8 Meshes

A *4–8 mesh* is a triangular mesh which has only vertices of valence 4 and 8. More formally, a 4–8 mesh is a 2D simplicial complex $K = (V, E, F)$, where V, E and F are respectively the sets of vertices, edges and faces of K. Moreover, V is divided into two classes $V = V_4 \cup V_8$, where $V_4 = \{v; \deg(v) = 4\}$, and $V_8 = \{v; \deg(v) = 8\}$.

A *regular 4–8 mesh* is a triangular mesh in which the 1-neighborhood of every internal (i.e. non-boundary) vertex of valence 4 has only neighbors of valence 8, and the 1-neighborhood of every internal vertex of valence 8 consists of a ring of vertices with alternating valences 4 and 8. (See Figure 1).

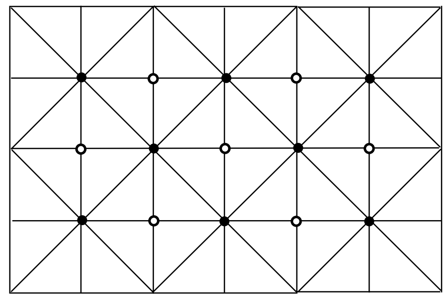


Fig. 1. Regular 4–8 mesh: ∘ - valence 4; • - valence 8.

Note that in a regular 4–8 mesh, every internal face is formed by linking two vertices of valence 8 with one of vertex valence 4, thus a 4–8 mesh corresponds to a $[4.8^2]$ Laves tiling [13].

2.2 *Refinement of 4–8 Meshes*

A *refinement operator* for a regular 4–8 mesh $K = (V, E, F)$ is defined by the following procedure:

(1) Split all edges $e = (v, w) \in E$ by inserting a *split vertex* $s_{vw} \in V'$, and connecting it to the endpoints $v, w \in V$ of e. That is, $e \mapsto \{e_v, e_w\}$, where $e_v = (v, s_{vw})$, $e_w = (s_{vw}, w)$, and $e_v, e_w \in E'$.

(2) Subdivide all faces $f \in F$ into four new faces by linking the vertex of degree 4, $u \in V_4$, to the split point s_{vw} of the opposite edge, and linking s_{vw} to the split points s_{wu} and s_{uv} of the remaining edges. That is, $f \mapsto \{f_w, f_{uw}, f_{uv}, f_v\}$, where $f = (u, v, w)$, $f_w = (w, s_{wu}, s_{vw})$, $f_{uw} = (u, s_{vw}, s_{wu})$, $f_{uv} = (u, s_{uv}, s_{vw})$, $f_v = (v, s_{vw}, s_{uv})$, and $f_w, f_{uw}, f_{uv}, f_v \in F'$. We remark that, on a face $(u, v, w) \in F$, by the regularity property, if $u \in V_4$ then $v, w \in V_8$.

(3) Update the complex: $K' \mapsto K$, where $K' = (V \cup V', E', F')$.

The subdivision template corresponding to this procedure is illustrated in Figure 2.

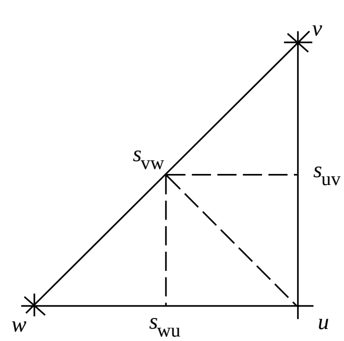


Fig. 2. Subdivision template for 4–8 mesh.

An important property of this refinement operator is that it can be decomposed into two interleaved refinement sequences. This will be useful for the development of our quasi 4–8 refinement algorithm. The operator decomposition is as follows: The quaternary subdivision performed in one refinement step is replaced by two nested binary subdivisions performed in subsequent steps.

The interleaved refinement procedure is very similar to the normal one. The difference is that step 1 splits only edges connecting valence 8 vertices, and step 2 subdivides faces in two, accordingly. The regularity of the mesh guarantees that just one edge bisects in each face.

Figure 3 compares the normal and interleaved refinement procedures.

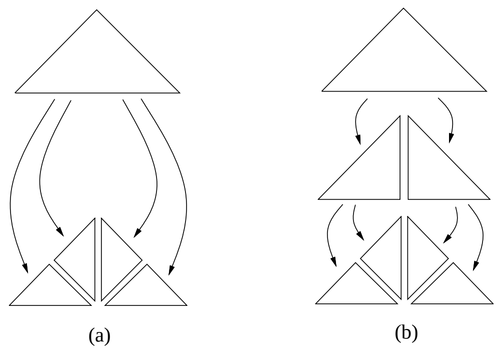


Fig. 3. Normal (a) and interleaved (b) 4–8 refinement.

A *uniform* 4–8 mesh is a planar embedding of a 4–8 mesh. Uniform 4–8 meshes are also known as four-directional grids. These meshes are closely related with the 4 direction Box splines [34], that are generated from the set of four direction vectors $\{e_1, e_2, e_1 + e_2, e_1 - e_2\}$, where $e_1 = (1, 0)$ and $e_2 = (0, 1)$.

Since a uniform 4–8 mesh is regular, it can be refined using the procedure described above. On the other hand, the fact that a uniform mesh is embedded in $\mathbb{R}^2$, makes it possible to exploit a geometric component in the design of the refinement operator. Note that the topological criteria for edge bisection can be replaced by a geometric criteria. A uniform 4–8 mesh has edges of length 1 (horizontal and vertical), and $\sqrt{2}$ (diagonals). Thus, in the case of uniform 4–8 meshes, it is easy to verify that an interleaved refinement procedure which splits the *longest* edges would produce the same results as the one using the topological criteria (i.e. edges with vertices of valence 8). This observation will be crucial in the design of our quasi 4–8 refinement operator.

3 The Quasi 4–8 Subdivision Scheme

In this section we describe the quasi 4–8 subdivision scheme, including the associated refinement and smoothing operators.

As we have seen 4–8 meshes posses very nice properties but, unfortunately they cannot represent surfaces of arbitrary topology. This motivates us to look for a generalization of 4–8 meshes that can be extended to arbitrary surfaces.

A quasi 4–8 mesh is a triangular mesh which has mostly vertices of valence 4 and 8, except for isolated vertices with some other valence. These are *extraordinary vertices*. Obviously, such characterization is only applicable to dense meshes. In fact, the real interest is to find a method to generate dense meshes with these properties, starting from coarse meshes.

A *quasi 4–8 refinement operator* is a transformation on simplicial complexes embedded in $\mathbb{R}^n$, such that its iterated application to an initial arbitrary (coarse) mesh will produce a quasi 4–8 mesh.

We now present an algorithm that implements this operator.

Algorithm 1 : quasi_4–8_refinement (K)

```
sort_edges (E) in list L
while L ≠ ∅ do
  get next e from L
  if e not marked then
    split (e)
    mark_cluster (e)
for all f ∈ F do
  subdiv (f)
```

The routine sort_edges, sorts edges by decreasing length and radially around each vertex. The routine mark_cluster of edge e, marks e and the edges sharing a face with e.The routine subdiv, performs a binary decomposition of a face by linking the split point of its longest edge to the opposite vertex.

Note that, the longest side bisection gives the best aspect ratio of the triangles in a binary subdivision [28]. Also, edge cluster marking ensures that, at most, one edge in each face splits.

The above remarks indicate that algorithm 1 produces a mesh with compatible geometry. The refinement procedure is also quite stable in relation to small geometric perturbations, as we demonstrate in subsection 4.1.

3.2 Smoothing

After the application of the refinement operator to the complex K, the set of vertices V of K, can be naturally divided into two classes: newly inserted vertices, $v' \in V'$, which we will call *new vertices*, and vertices inherited from the previous mesh, $v \in V$, which we will call *old vertices*. The smoothing operator is a convolution filter. It uses a different smoothing rule for each class of vertices.

The stencil for new vertices is depicted in Figure 4(a). Recall that the 1-neighborhood of internal new vertices, by construction, consists of exactly 4 vertices. The filter function is an average of the coordinate values of these 1-neighbors.

The stencil for old vertices is depicted in Figure 4(b). Observe that the filter kernel extends beyond the first neighbors of the vertex. It correspond to a smoothing filter based on 1-neighbors before the introduction of new vertices (dashed lines in Figures 4 and 4 are newly created edges). Note also that this smoothing filter produces less shrinking than a gaussian filter, because more weight is put on the central vertex. The design of this filter was inspired by the non-shrinking Laplacian smoothing of Taubin [30].

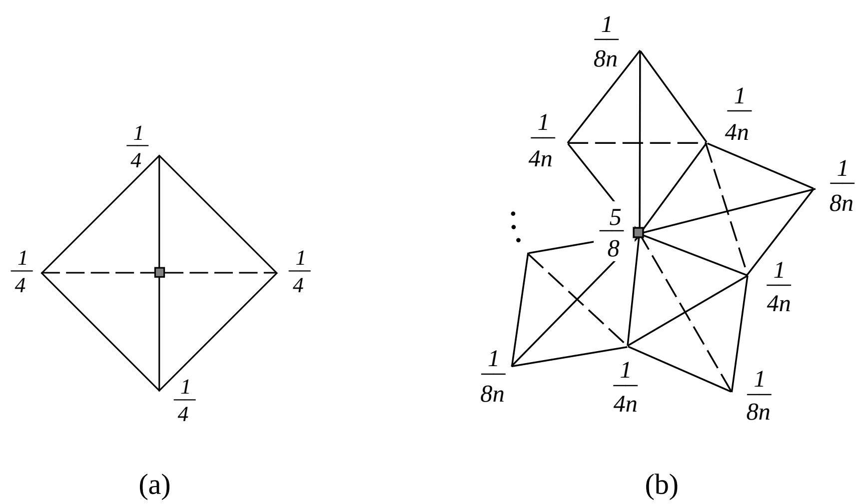


Fig. 4. Filter masks for new vertex (a) and old vertex (b).

The smoothing operator can be implemented very efficiently using repeated convolution by a sequence of averaging operations. In a first pass, new vertices are calculated as the average of their four neighbors. In a second pass, old vertices are updated as the average of their current value and the average of the values of new vertices in their 1-neighborhood. It is easy to verify that this procedure corresponds to the stencils in Figure 4 (a) and (b). Note that, this cascade convolution makes it possible to execute all the computations "in-place".

This smoothing algorithm is shown below

Algorithm 2 : quasi_4-8_smoothing (K)

for $v'_i \in V'$ **do**

$$p^{l+1}(v'_i) = \frac{1}{4} \sum_{v_j \in N_1(v'_i)} p^l(v_j)$$

for $v_i \in V$ **do**

$$p^{l+1}(v_i) = \frac{1}{2} p^l(v_i) + \frac{1}{2k} \sum_{v'_j \in N_1(v_i) \cap V'} p^{l+1}(v'_j)$$

where k is the number of new vertices in $N_1(v_i)$.

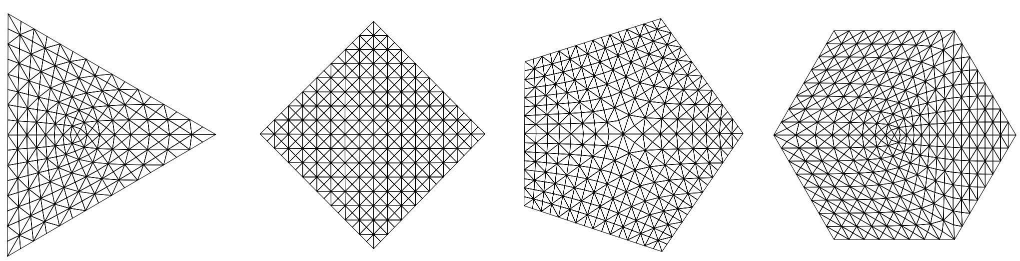
4 Examples

In this section, we give some examples of surfaces generated with our subdivision scheme.

4.1 N-Regular Neighborhoods

The following examples illustrate the behavior of the scheme for planar 2D neighborhoods.

Figure 5 shows the subdivision of regular planar polygons with 3 to 6 sides. Note that, for $n = 3, 4$ and 5, the central vertex after subdivision has valence $2n$. Note also that, except for boundary vertices, all other vertices are ordinary and structured according to a regular 4–8 pattern. For the regular hexagon in Figure 5(d) the valence of the central vertex remained unchanged. Observe that the algorithm constructed the same subdivision as in the triangle of Figure 5(a).



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

Fig. 5. Subdivision of n-regular planar polygons.

Figure 6 reveals how the algorithm behaves in the case of a non-uniform initial triangulation. The input mesh, shown in Figure 6(a) is the 6-regular triangulation of Figure 5(d), but warped such that the horizontal internal edges have a 1:2 length ratio. Note that, because the subdivision is based on geometry, the final triangulation adapts nicely to the polygonal domain. Since one of the horizontal edges is longer than the rest, it has split in the first subdivision step. The end result is that the final mesh in Figure 6(b), gradually transitions from a 4-regular structure (on the left hand side) to a 5-regular structure (on the right hand side).

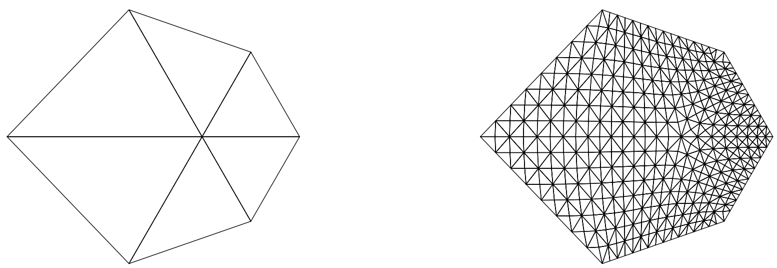
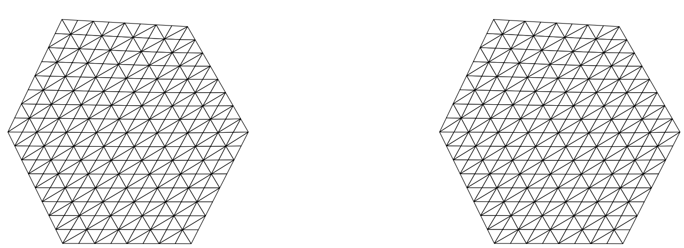


Fig. 6. Warped hexagon.

A consequence of the adaptivity capability of the quasi 4-8 subdivision scheme is that the connectivity of the subdivided mesh depends on geometry. This property makes it more difficult to predict the exact valence of a particular vertex as the mesh is refined. Nonetheless, the subdivision process is quite stable and robust to perturbations of the local geometry. In order to test the sensitivity of the scheme we added a radial random displacement to the initial vertices of a planar regular hexagon. Figure 7 displays the result of this test. The magnitude of the perturbations are uniformly distributed in the intervals $[-0.1, 0.1]$ for Figure 7(a), and $[-0.3, 0.3]$ for Figure 7(b). The radius of the hexagon is 1.0. Note that the topology of the mesh is the same in both figures (the central vertex has valence 8), but it is different from the topology of the mesh in Figure 5(d) (where the central vertex has valence 6). The regular hexagon is a limit case, since the initial mesh is composed of equilateral triangles.



(a) (b)

Fig. 7. Perturbations to edge lengths.

Figure 8 exemplifies the subdivision of n-regular planar polygons for which $n > 6$. It contrasts the behavior of the algorithm when the valence n is even and odd. Figure 8(a) shows the mesh for a 12-sided polygon after the first subdivision step. Note the radial decomposition structure around the central vertex. The final mesh, shown in Figure 8(b), has a quasi 4–8 symmetric structure. Figure 8(c) shows the mesh for a 9-sided polygon after the first subdivision. The decomposition cannot be completely radial, because n is odd. The final mesh, shown in Figure 8(d), has a quasi 4–8 structure, but exhibits a slight asymmetry. Observe that in both cases, the triangulations of Figures 8(b) and (d) are somewhat distorted near the boundaries. This is, in part, due to the fact that boundary vertices are not allowed to move.

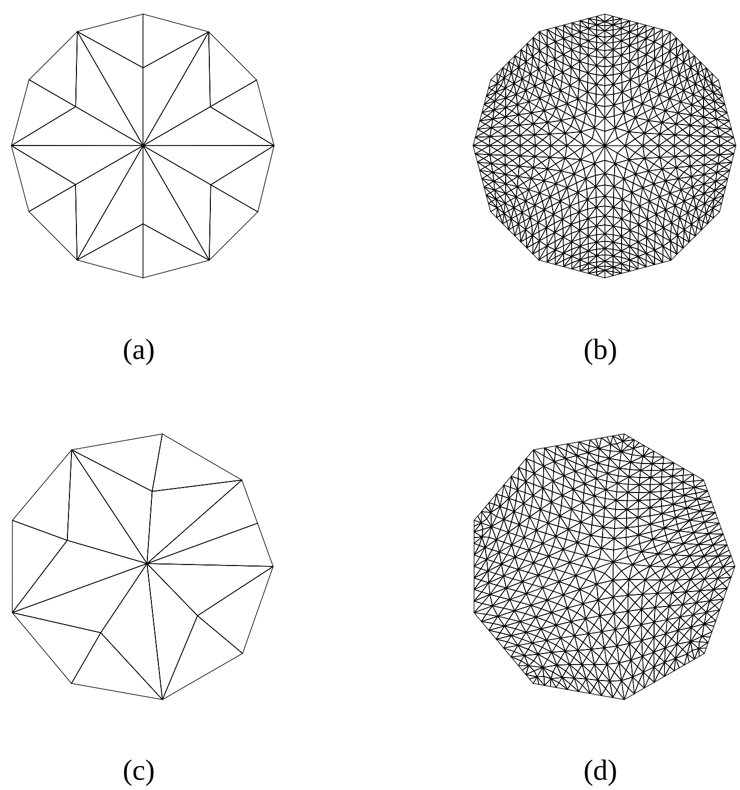


Fig. 8. Subdivision of regular polygons with even (a,b) and odd (c,d) number of sides.

4.2 3D Objects

The following examples illustrate the application of the scheme to control meshes in three dimensions.

The first example is a torus. The control mesh is shown in Figure 9(a). Polygonal meshes after two and four subdivision steps are shown in Figures 9(b) and (c). Note that these meshes are 4–8 regular (i.e. there are no extraordinary vertices). Note also that the resulting surface gives a good piecewise linear approximation of the torus.

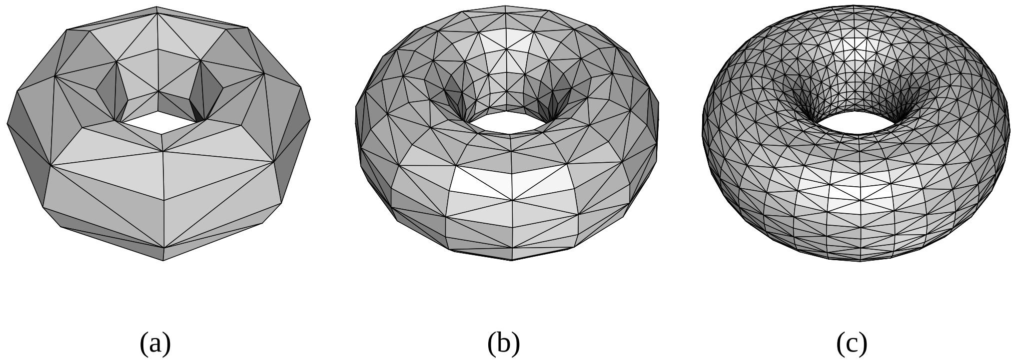


Fig. 9. Polygonal surface generated from a coarse approximation of a torus.

The second example is the mannequin head. The control mesh is shown in Figure 10(a). Figure 10(b) shows the resulting subdivision surface after two subdivision steps. Figure 10(c) shows a flat shaded rendering of the mesh after nine subdivision steps. Note that the shape is very smooth and, yet the main features are preserved.

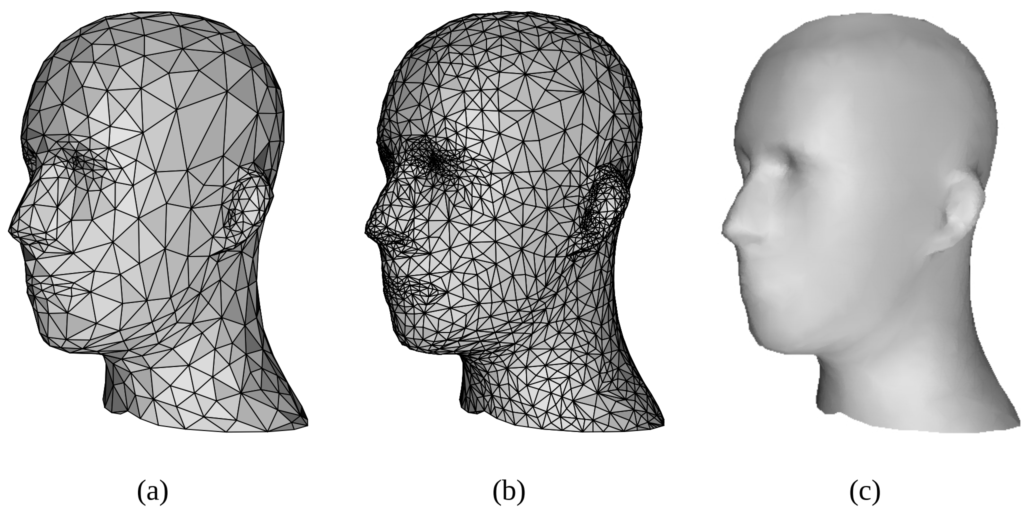


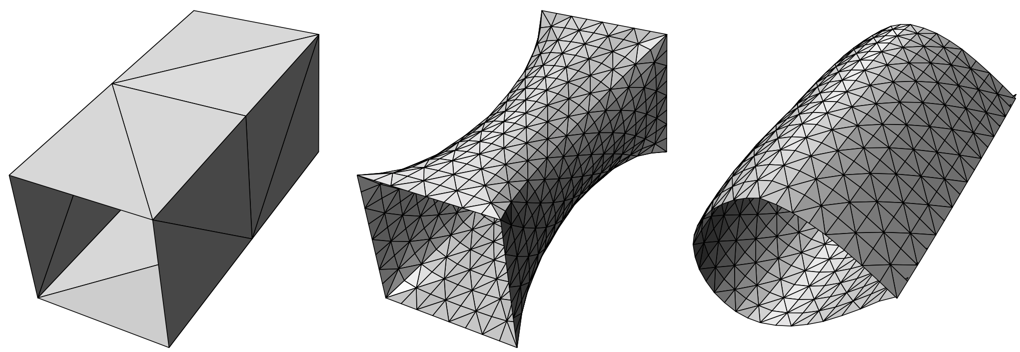
Fig. 10. Mannequin head. Control mesh (a), two levels of subdivision (b), Flat shading after nine levels of subdivision (c).

4.3 Feature Control

The final example demonstrates the result of incorporating control of shape features in the subdivision scheme. It also shows the treatment of surfaces with boundary.

Boundary vertices are constrained to move along the surface normal direction. We employ a tagged mesh for controlling features, as in other schemes [8,12]. Currently, we have implemented only the simplest kind of control mechanism, in which a tagged vertex or edge is not affected by smoothing. We plan to further investigate this issue, and experiment with different types of context sensitive smoothing filters.

The example in Figure 11 is a surface with boundary. The control mesh, shown in Figure 11(a), is a triangulated box shape with two open sides. The surface shown in Figure 11(b) was constructed by tagging all boundary edges and vertices. Note the smooth blend between the two open extremities. To produce the surface shown in Figure 11(c), boundary edges were untagged and one internal longitudinal edge was tagged. Note that, now, the boundary curves are smooth, except at the vertex shared with the tagged edge.



(a) (b) (c)

Fig. 11. Control of shape features and boundary treatment: (a) initial mesh; (b) tagged boundary edges; (c) tagged internal edge.

5 Conclusions

In this paper we presented a new scheme for subdivision surfaces. It is based on quasi 4–8 regular meshes, and combines a geometry sensitive refinement operator with a convolution smoothing operator.

5.1 Overview

Overall, the scheme integrates effectively the two main operations of refinement and smoothing in order to exploit the adaptivity of the mesh structure. The implementation is simple and efficient.

We have demonstrated the capabilities of the scheme through examples of different surfaces: with and without boundary; of arbitrary topological type; as well as, from shapes with variable level of complexity.

This scheme is an addition to the repertoire of modeling tools for subdivision surfaces. In the range of existing techniques, it lies halfway between stationary and variational subdivision methods. Our scheme is somewhat more flexible than stationary subdivision, but it is slightly less efficient, and it does not have a closed form description, as those methods. Compared with variational methods, our scheme is more efficient but less powerful, particularly in relation to user-defined constraints.

5.2 Future Work

An important theoretical aspect that was not addressed in this paper is the convergence analysis of the proposed subdivision scheme. In this respect, we have conducted numerical experiments to investigate the eigenstructure of the associated subdivision matrices. Based on these experiments, we conjecture that our scheme produces C^1 surfaces. Nonetheless, a complete formal analysis is required to demonstrate this conjecture in a conclusive way. Unfortunately, the analysis of irregular subdivision is still an open problem. We are currently looking for a characterization of the quasi 4–8 subdivision scheme with the intent to show that it is *quasi-stationary*, i.e. that it is asymptotically equivalent to a convergent stationary subdivision scheme.

References

[1] A. A. Ball and D. J. T. Storry. Conditions for tangent plane continuity over recursively generated B-spline surfaces. *ACM Transactions on Graphics*, 7(2):83–102, 1988.

[2] Henning Biermann, Adi Levin, and Denis Zorin. Piecewise smooth subdivision surfaces with normal control. *Proceedings of SIGGRAPH 2000*, pages 113–120, July 2000. ISBN 1-58113-208-5.

[3] W. Boehm. *Surfaces in Computer Aided Geometric Design*, chapter The de Boor algorithm for triangular splines, pages 109–120. North-Holland, 1982.

[4] E. Catmull and J. Clark. Recursively generated B-spline surfaces on arbitrary topological meshes. *Comput. Aided Design*, 10:350–365, 1978.

[5] E. Cohen, T. Lyche, and R. Riesenfeld. Discrete box splines and refinement algorithms. *Computer Aided Geometric Design*, 1(2):131–148, 1984.

[6] W. Dahmen and C. A. Micchelli. Subdivision algorithms for the generation of box-spline surfaces. *Computer-Aided Geometric Design*, 1:115–129, 1984.

[7] C. de Boor. *A Practical Guide to Splines*. Springer, Berlin, 1978.

[8] T. DeRose, M. Kass, and T. Truong. Subdivision surfaces in character animation. In *SIGGRAPH 98 Conference Proceedings*, pages 85–94, 1998.

[9] D. Doo and M. Sabin. Behaviour of recursive division surfaces near extraordinary points. *Comput. Aided Design*, 10:356–360, 1978.

[10] D. Doo and M. Sabin. Behaviour of recursive division surfaces near extraordinary points. *Computer-Aided Design*, 10:356—360, September 1978.

[11] N. Dyn, D. Levin, and J. A. Gregory. A butterfly subdivision scheme for surface interpolation with tension control. *ACM Transactions on Graphics*, 9(2):160–169, April 1990.

[12] Nira Dyn, David Levin, and John A. Gregory. A butterfly subdivision scheme for surface interpolation with tension control. *ACM Transactions on Graphics*, 9(2):160–169, April 1990. ISSN 0730-0301.

[13] B. Grünbaum and G. Shephard. *Tilings and Patterns*. W. H. Freeman, 1987.

[14] A. Habbib and J. Warren. Edge and vertex insertion for a class of C^1 subdivision surfaces. preprint, 1997.

[15] Mark Halstead, Michael Kass, and Tony DeRose. Efficient, fair interpolation using Catmull-Clark surfaces. In *Computer Graphics (SIGGRAPH '93 Proceedings)*, volume 27, pages 35–44, 1993.

[16] H. Hoppe, T. DeRose, T. Duchamp, M. Halstead, H. Jin, J. McDonald, J. Schweitzer, and W. Stuetzle. Piecewise smooth surface reconstruction. In *Proceedings of SIGGRAPH '94*, pages 295–302, 1994.

[17] L. Kobbelt. Interpolatory subdivision on open quadrilateral nets with arbitrary topology. *Computer Graphics Forum*, 15(3):409–420, 1996.

[18] L. Kobbelt. A variational approach to subdivision. *Computer Aided Geometric Design*, 13(8):743–761, 1996.

[19] J. Lane and R. Riesenfeld. A theoretical development for the computer generation and display of piecewise polynomial surfaces. *IEEE Trans. Pattern Analysis Machine Intell.*, 2(1):35–46, 1980.

[20] A. Levin. Interpolating nets of curves by smooth subdivision surfaces. In *Proceedings of SIGGRAPH 99, Computer Graphics Proceedings, Annual Conference Series*, pages 57–64, 1999.

[21] C. Loop. Smooth subdivision for surfaces based on triangles. Master's thesis, University of Utah, 1987.

[22] Ahmad H. Nasri. Recursive subdivision of polygonal complexes and its applications in computer-aided geometric design. *Computer Aided Geometric Design*, 17(7):595–619, August 2000. ISSN 0167-8396.

[23] Ahmad H. Nasri. Polyhedral subdivision methods for free-form surfaces. *ACM Transactions on Graphics*, 6(1):29–73, January 1987.

[24] J. Peters and U. Reif. The simplest subdivision scheme for smoothing polyhedra. *ACM Transactions on Graphics*, 16(4):420–431, October 1997.

[25] H. Prautzsch. Generalized subdivision and convergence. *Computer Aided Geometric Design*, 2(1-3):69–76, 1985.

[26] H. Prautzsch and G. Umlauf. Improved triangular subdivision schemes. *Computer Graphics International 1998*, June 1998. Held in Hannover, Germany.

[27] U. Reif. A unified approach to subdivision algorithms near extraordinary vertices. *Computer Aided Geometric Design*, 12(2):153–174, 1995.

[28] M. C. Rivara. Mesh refinement processes based on the generalized bisiection of simplices. *SIAM J. Numer. Anal.*, 21(3):604–613, 1984.

[29] Jean Schweitzer. *Analysis and Application of Subdivision Surfaces*. PhD thesis, University of Washington, 1996.

[30] G. Taubin. A signal processing approach to fair surface design. In *SIGGRAPH 95 Conference Proceedings*, pages 351–358, 1995.

[31] J. Warren. Subdivision schemes for variational splines. preprint, 1997.

[32] D. Zorin. *Stationary Subdivision and Multiresolution Surface Representations*. PhD thesis, Caltech, 1997.

[33] D. Zorin, P. Schroeder, and W. Sweldens. Interpolating subdivision for meshes with arbitrary topology. In *SIGGRAPH 96 Conference Proceedings*, pages 189–192, 1996.

[34] P. B. Zwart. Multivariate splines with nondegenerate partitions. *SIAM J. Numer. Anal.*, 10(4):665–673, 1973.