Reverse Catmull-Clark Subdivision

Sandrine Lanquetin
LE2I, UMR CNRS 5158
Université de Bourgogne
BP 47870
21078, DIJON, France
sandrine.lanquetin@u-bourgogne.fr

Marc Neveu
LE2I, UMR CNRS 5158
Université de Bourgogne
BP 47870
21078, DIJON, France
marc.neveu@u-bourgogne.fr

ABSTRACT

Reverse subdivision consists in constructing a coarse mesh of a model from a finer mesh of this same model. In this paper, we give formulas for reverse Catmull-Clark subdivision. These formulas allow the constructing of a coarse mesh for almost all meshes. The condition for being able to apply these formulas is that the mesh to be reversed must be generated by the subdivision of a coarse mesh. Except for this condition, the mesh can be arbitrary. Vertices can be regular or extraordinary and the mesh itself can be arbitrary (triangular, quadrilateral…).

Keywords

Reverse Subdivision, multiresolution, Catmull-Clark scheme.

1. INTRODUCTION

Subdivision surfaces were introduced in 1978 by Catmull-Clark [Cat78] and Doo-Sabin [Doo78] as an extension of the Chaikin algorithm [Cha74]. These surfaces are widely used in character animation (such as Geri's Game © or Finding Nemo ©) to smooth out models. Indeed, from a coarse mesh, successive refinements give finer meshes. A sequence of subdivided meshes converges towards a smooth surface called limit surface. Since the earliest subdivision surfaces in 1978, many subdivision schemes were proposed. Some are approximating and others are interpolating (i.e. control vertices of successive meshes belong to the limit surface).

The main advantage of the Catmull-Clark scheme over the triangle based subdivision such as the Loop scheme [Loo87] is that the control mesh faces can have an arbitrary number of edges. This is an important feature in modeling because most of the time designers build their model by symmetry as shown in Figure 1. Moreover subdivision surfaces are more and more used in CAGD and in this field; most meshes are quadrilateral, in coherence with parametric surfaces (Bezier, B-splines, NURBS…).

Subdivision methods produce an increasingly fine sequence of meshes. On the contrary, it can be interesting to pass quickly from a mesh to a coarser one according to the point of view for example. Using a local formula for decreasing the resolution of a mesh is a crucial element for the implementation of multiresolution surfaces. It reverses the subdivision process. While formulas for subdividing meshes are local, the existence of local formulas for the respective reverse subdivision is less evident.

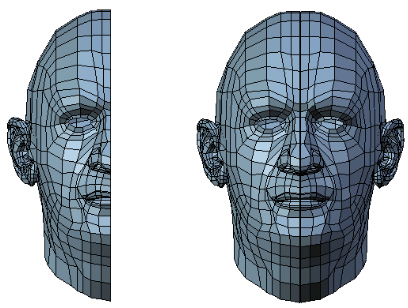

Figure 1. A character head build by symmetry.

There are several global methods such as multiresolution methods [Lou97], as Loop reverse subdivision [Mon04]. The interest for the local methods however is new. Samavati and Bartels

Conference proceedings ISBN 80-86943-03-8
WSCG'2006, January 30-February 3, 2006
Plzen, Czech Republic.

determined the local reverse subdivision masks for the Butterfly and Loop scheme restricted to regular vertices (valence 6) in [Sam02]. Samavati et al. focused on the Doo-Sabin scheme for arbitrary meshes in [Sam02]. Samavati, Pakdel, Smith and Prusinkiewicz [Sam03] propose a local method for the Loop scheme which consists in locally reversing the formula for a given set of vertices.

In this paper, we focus on Catmull-Clark and develop a local method to reverse this subdivision. Section 2 overviews the Catmull-Clark scheme. Then, we present our method for reversing Catmull-Clark scheme in Section 3. Thus obtained results are shown in Section 4, illustrating different cases that can occur.

2. BACKGROUND

The Catmull-Clark subdivision scheme

The Catmull-Clark subdivision scheme generalizes the generation algorithm of cubic B-splines to surfaces; it is based on the tensor product bicubic spline [Dyn90]. The rules of Catmull-Clark were first defined for meshes with quadrilateral faces but they can easily be generalized to arbitrary polygonal meshes. The *valence* of a vertex is the number of vertices connected to this vertex by an edge. In the case of Catmull-Clark, if the vertex valence is not four the vertex is denoted as an *extraordinary vertex*. Even if the initial mesh is not quadrilateral, meshes generated at each subdivision level are quadrilateral.

Let M^k be the mesh at the subdivision level k. Each vertex of M^{k+1} can be associated to a face, an edge or a vertex of M^k. These vertices are respectively called face point, edge point or vertex point.

- A face point denoted f_j^{k+1} is computed as the average of the vertices of this face:

$$f_j^{k+1} = \frac{1}{\left|F_j\right|} \sum_{f_i^k \in F_j} f_i^k \qquad \textbf{(1)}$$

- An edge point e_j^{k+1} is computed as the average of the endpoints of this edge and the face points of the two incident faces of this edge :

$$e_j^{k+1} = \frac{v^k + e_j^k + f_j^{k+1} + f_{j-1}^{k+1}}{4} \qquad \textbf{(2)}$$

where v^k and e_j^k are the endpoints of the edge on the k^{th} subdivision level and f_{j-1}^{k+1} and f_j^{k+1} are the newly computed face points of the faces incident to this edge.

- A vertex point is a weighted average of its incident vertices of the same level and of the face points of the incident faces

$$v^{k+1} = \frac{n-2}{n} v^k + \frac{1}{n^2} \sum_{j=0}^{n-1} e_j^k + \frac{1}{n^2} \sum_{j=0}^{n-1} f_j^{k+1} \qquad \textbf{(3)}$$

where f_j^{k+1} are the newly computed face points, e_j^k are the neighbours of v^k on the same subdivision level and n is the valence of the vertex v^k.

Figure 2 illustrates these notations for a vertex v^k with valence n.

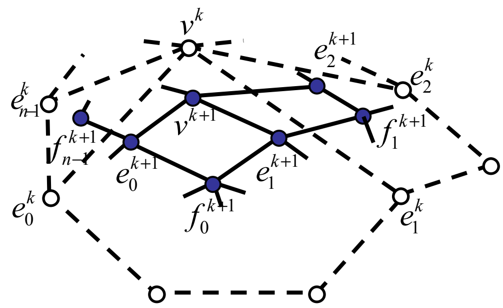

Figure 2. Catmull-Clark Subdivision for a vertex v^k with valence of n.

The valence of a vertex point remains the same after subdivision i.e. $\#v^k = \#v^{k+1}$ and $\#v_j^k = \#v_j^{k+1}$ where $\#$ denotes the valence of a point. The valence of edge points are always four and the valence of face points corresponds to the number of edges of the face. Figure 3 shows the successive meshes obtained with the Catmull-Clark subdivision.

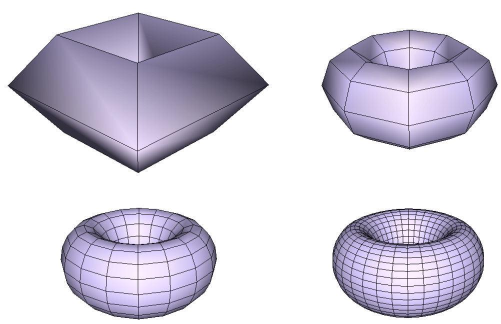

Figure 3. Catmull-Clark subdivision applied on a torus mesh.

Boundaries

The formulas used for vertices on a mesh boundary are different from those for interior vertices. The notations used are the same than previously but this time v^k, e_0^k and e_1^k form a boundary of the mesh (Figure 4).

The formula used for computed boundary vertices with Catmull-Clark scheme are those of cubic B-spline curves:

$$v^{k+1} = \frac{1}{8}e_0^k + \frac{3}{4}v^k + \frac{1}{8}e_1^k$$

$$e_0^{k+1} = \frac{1}{2}v^k + \frac{1}{2}e_0^k$$

$$e_1^{k+1} = \frac{1}{2}v^k + \frac{1}{2}e_1^k$$

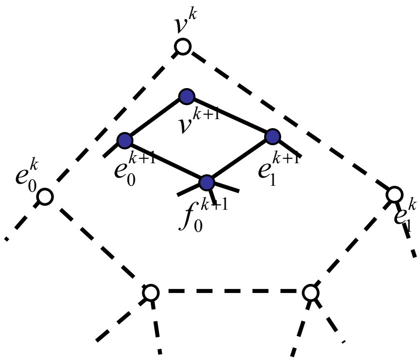

Figure 4. v^k, e_0^k and e_1^k form a boundary of the mesh.

3. Reverse formula for Catmull-Clark subdivision

Method

For the reverse subdivision process, it is necessary to expand a formula in which v^k is only determined from v^{k+1} and its neighbourhood e_j^{k+1}, f_j^{k+1} $j \in [\![0, n-1]\!]$ with n the valence of v^{k+1}. Let v^k be a vertex with valence n and M^k be an arbitrary mesh as shown in Figure 5.

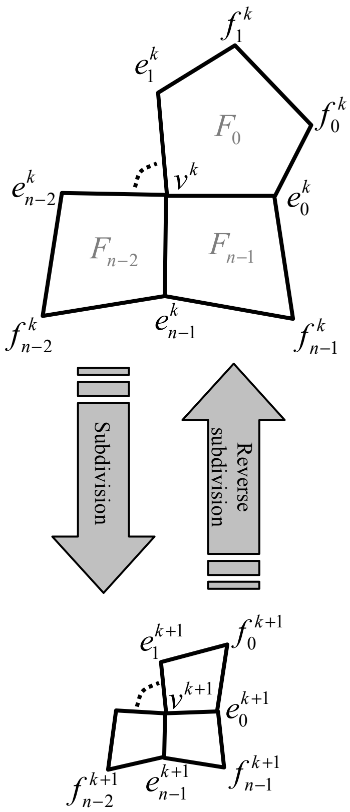

Figure 5. General neighbourhood for an extraordinary vertex.

Ordinary and extraordinary vertices

In the reverse problem, we know vertices of the level $k+1$ and we want to find the vertex v^k using a formula which respects the following conditions:

- The formula is an affine operation
- The neighbours e_j^{k+1} of v^{k+1} in the formula must have the same weight as in Equation **(3)** and the centroid f_j^{k+1} of faces F_i incident to v^{k+1} in the formula must also have the same weight
- The application of the reverse formula must exactly reconstruct v^k i.e. the subdivision of the vertex v^k generates the vertex v^{k+1} and the reverse subdivision of the vertex v^{k+1} gives v^k.

The second condition yields to the diagram shown in Figure 6. This diagram is called mask and shows the coefficient to apply on vertices.

 ISBN 80-86943-03-8

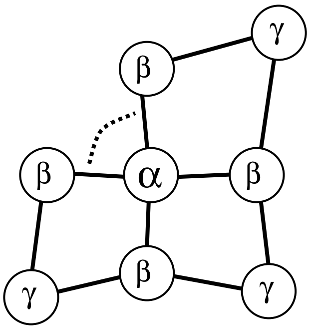

Figure 6. Reverse Mask corresponding to the general neighbourhood of Figure 5.

Let α, β, γ be respectively the coefficients for the vertex point, edge points and face points. For the third condition presented above, the following equation must hold:

$$\alpha v^{k+1} + \beta \sum_{j=0}^{n-1} e_j^{k+1} + \gamma \sum_{j=0}^{n-1} f_j^{k+1} = v^k .$$

From Equation **(1)** to **(3)**, we obtain:

$$\left(\frac{(n-2)\alpha}{n} + \frac{n\beta}{4}\right) v^k + \left(\frac{\alpha}{n^2} + \frac{\beta}{4}\right)\sum_{j=0}^{n-1} e_j^k + \left(\frac{\alpha}{n^2} + \frac{\beta}{2} + \gamma\right)\sum_{j=0}^{n-1} f_j^{k+1} = v^k$$

.

This equation yields to the following system:

$$\begin{cases} \frac{n-2}{n}\alpha + \frac{n}{4}\beta = 1 \\ \frac{1}{n^2}\alpha + \frac{1}{4}\beta = 0 \\ \frac{1}{n^2}\alpha + \frac{1}{2}\beta + \gamma = 0 \end{cases}$$

This can be written with matrix:

$$A \times \Theta = B \quad \text{where} \quad A = \begin{bmatrix} \frac{n-2}{n} & \frac{n}{4} & 0 \\ \frac{1}{n^2} & \frac{1}{4} & 0 \\ \frac{1}{n^2} & \frac{1}{2} & 1 \end{bmatrix}, \quad \Theta = \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix}$$

and $B = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$.

The determinant of the matrix A is $\det(A) = \frac{n-3}{4n}$.

So the system can be solved only when $n \neq 3$.

For $n \neq 3$, we find:

$$(\alpha, \beta, \gamma) = \left(\frac{n}{n-3}, \frac{-4}{n(n-3)}, \frac{1}{n(n-3)}\right)$$

Finally, for an ordinary or extraordinary vertex with valence $n \neq 3$:

$$\boxed{v^k = \frac{n}{n-3} v^{k+1} + \frac{-4}{n(n-3)} \sum_{j=0}^{n-1} e_j^{k+1} + \frac{1}{n(n-3)} \sum_{j=0}^{n-1} f_j^{k+1}} \quad \textbf{(4)}$$

This allows to compute v^k in most of the cases.

Vertices with valence $n = 3$

If $n = 3$, the system of the previous section cannot be reversed because the determinant is equal to zero. So, we have to find another way to construct v^k. The solution we chose is to perform the rebuilding of the mesh at the level k in two stages:

- First, vertices of valence $n \neq 3$ are constructed
- Then, vertices with valence $n = 3$ are treated

Let v^k be a vertex with valence $n = 3$. Figure 7 illustrates notations used in this section. The mesh at subdivision level k is drawn with dotted lines and the subdivided mesh is drawn with solid lines.

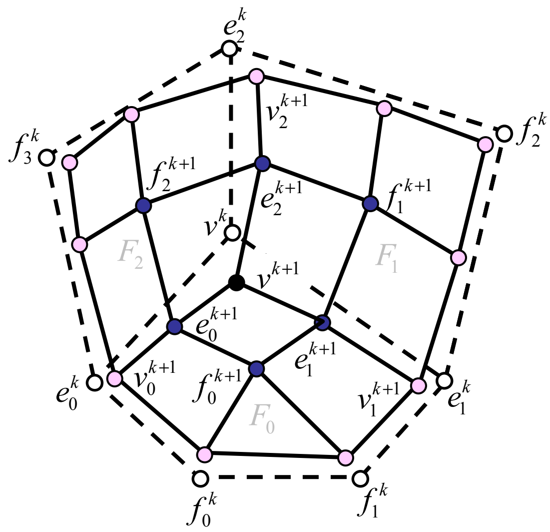

Figure 7. Notations used in the case $n = 3$.

 ISBN 80-86943-03-8

We now will explain the method used when $n = 3$. The vertex v_j^{k+1} associated to e_j^k, $j \in [\![0,2]\!]$ at the level $k+1$ necessarily has the same valence. Thus even if e_j^k is not already reconstructed, we know its valence. Moreover, from the previous section, we know that only vertices with valence $n = 3$ are not reconstructed. This property is used to classify the vertices in two categories:

- Either there is at least one vertex among the incident vertices e_j^k of the vertex v^k, $j \in [\![0,2]\!]$ which is already constructed (with valence $n = 3$ or not)
- Or there is no vertex among the incident vertices e_j^k of the vertex v^k, $j \in [\![0,2]\!]$ which is already constructed (with valence $n = 3$ or not).

In the first case, i.e. when one of the e_j^k is known, the coordinates of v^k can easily be computed from the Equation **(2)**. The formula can be written as:

$$\boxed{v^k = 4e_j^{k+1} - e_j^k - f_j^{k+1} - f_{j-1}^{k+1}}$$

where the coordinates of e_j^{k+1}, e_j^k, f_j^{k+1} and f_{j-1}^{k+1} are known.

In the second case, i.e. when no vertices e_j^k are constructed, we can construct v^k in the cases that we describe below:

If there is one face F_j, $j \in [\![0,2]\!]$ such that all vertices $f_i^k \in F_j$ are already constructed, then we can generate a new system from Equation **(1)** and **(2)** written for this face F_j and the two edge points of this face incident to v^k :

$$\begin{cases} 4f_j^{k+1} = v^k + e_j^k + \sum_{f_i^k \in F_j} f_i^k + e_{(j+1)[n]}^k \\ 4e_j^{k+1} = v^k + e_j^k + f_j^{k+1} + f_{(j-1)[n]}^{k+1} \\ 4e_{(j+1)[n]}^{k+1} = v^k + e_{(j+1)[n]}^k + f_j^{k+1} + f_{(j+1)[n]}^{k+1} \end{cases}$$

From this, the formula to construct v^k verifies:

$$\boxed{\begin{aligned} v^k = 4e_j^{k+1} - f_j^{k+1} - f_{(j-1)[n]}^{k+1} - 5f_j^{k+1} - f_{(j+1)[n]}^{k+1} \\ - f_{(j+1)[n]}^{k+1} + 4e_{(j+1)[n]}^{k+1} + \sum_{f_i^k \in F_j} f_i^k \end{aligned}}$$

The only case where we cannot find a formula for v^k is when there is not adjacent face at the vertex v^k such as its vertices except v^k are already constructed. So we stop here our investigations because these results are sufficient for almost all cases. Indeed, the Catmull-Clark scheme is generated to be applied on quadrilateral meshes with a regular valence $n = 4$. Of course, it is extended to polygonal faces and extraordinary valences $n \neq 4$ in order to be able to subdivide a greater number of meshes with this subdivision scheme. Indeed, the initial condition (a quadrilateral mesh with valence $n = 4$) is a too restrictive condition.

However, applying this scheme on a triangular mesh is not the goal because there are schemes which are better adapted to these meshes.

The cases for which these formulas are not enough are thus those where the valence is 3 for all mesh vertices as in the case of the box for example (the 8 vertices have 3 for valence). Indeed, even in these kinds of cases, among the new inserted vertices, edge points have a valence equal to 4. Thus, at the next level of subdivision, the mesh corresponds to a case described above.

Boundary vertices

With the same notation as in the part of section 2 on boundaries, one formula given by Bartels and Samavati in [Bar00] is the following:

$$v^k = -\frac{1}{2}e_0^{k+1} + 2v^{k+1} - \frac{1}{2}e_1^{k+1}$$

This formula can exactly be used as the Equation **(4)** we give for the interior vertices with valence $n \neq 3$.

4. RESULTS

In this section, the method described in the previous section is applied to on meshes with different features.

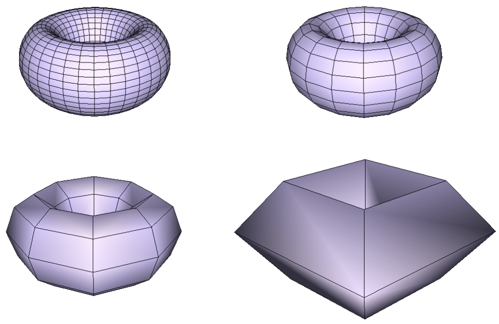

Figure 8. Reverse Catmull-Clark subdivision applied to on the torus mesh subdivided 3 times.

The easier example consists in rebuilding the torus of Figure 3 from one of its subdivision: the mesh at the third level of subdivision. As the initial mesh consists of quadrilateral faces with vertices of valence 4, all the new vertices are also of valence 4. The successive meshes can so be constructed with the general method described in the previous section (Figure 8).

This second example illustrates a case with extraordinary valences. There are valences equal to 3, 4 and 6. Vertices with valence 4 or 6 are constructed from the general method and vertices with valence 3 from the first formula introduced for valence equal to 3 because there is always one of the incident vertices with a valence not equal to 3. Successive meshes are shown in Figure 9.

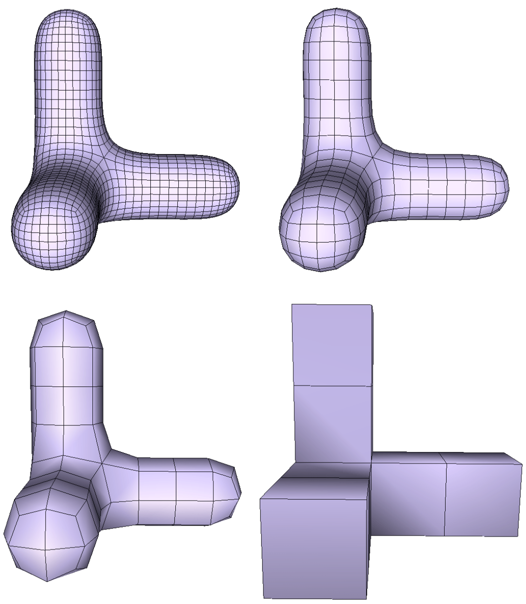

Figure 9. Reverse Catmull-Clark subdivision applied on a mesh with valences equal to 3, 4 or 6.

Let us consider the cat mesh. This mesh is triangular with arbitrary valences except the 3 value. Thus the general method can be applied to reconstruct the initial level from the first subdivision as shown in Figure 10. Indeed, face points are only vertices with valence 3 at the first subdivision level; they are marked with triangles in Figure 11. By construction, face points are computed as the centroid of a face, so there are no points which correspond to face points at the previous level but a face.

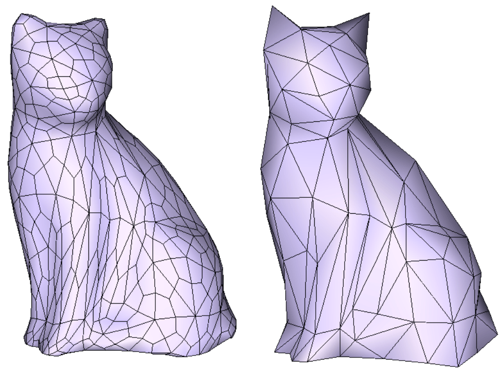

Figure 10. Reverse subdivision of the cat mesh at the first subdivision level.

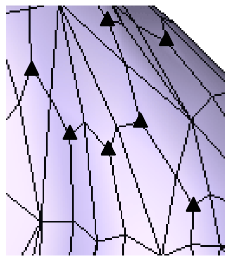

Figure 11. Zoom on a part of the mesh cat. Face points are represented by triangles.

If the mesh is subdivided once more (2 subdivisions), we have to construct vertices from vertices with valence 3 as illustrated in Figure 12.

Figure 13 focus on a part of the mesh to show what happens. Only the vertices to reconstruct are marked: those with valence 3 are represented by circles and the other by squares. In this mesh, there is always a vertex with a valence not equal to 3 between two vertices of valence 3. So the vertices of the previous mesh can be constructed from the general method and the first formula introduced for valence equal to 3.

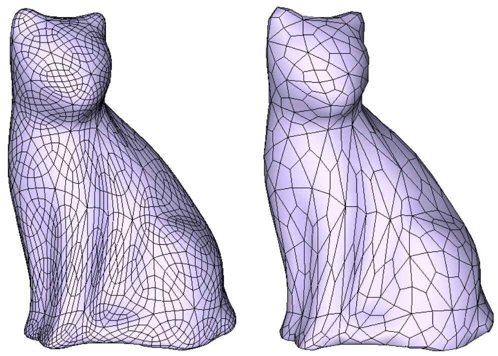

Figure 12. Reverse subdivision of the cat mesh at the second subdivision level.

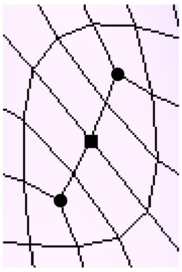

Figure 13. Zoom on a part of the mesh cat. Circles represent vertices with valence 3 and squares represent vertices with valence 4.

Let us consider a cube. At the initial level, all faces are quadrilateral and valence of vertices is always 3 (Figure 14 left). From the first level of subdivision, formulas explained here do not allow to determine vertices of the initial level from the first subdivision level because the vertices to construct all have 3 for valence (Figure 14 right). The same problem occurs with a tetrahedron.

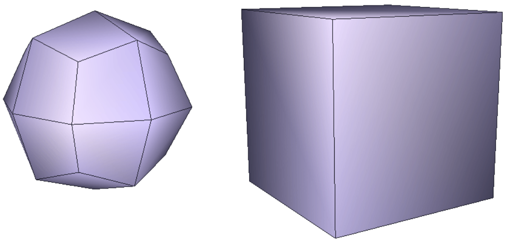

Figure 14. One particular example where the method fails: the cube starting from the first level of subdivision.

However, from the other successive subdivided meshes (except the first subdivision level), meshes of the previous level can be constructed from the general method and the first formula introduced for valence equal to 3. Indeed, vertices with valence 3 are isolated. Thus, Figure 15 shows on the left the cube mesh subdivided twice and its reconstructed mesh at the previous level.

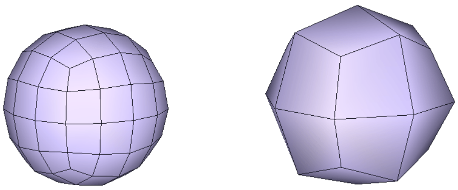

Figure 15. Reverse subdivision of the cube mesh at the second subdivision level.

5. CONCLUSION

With this method, coarser meshes can be quickly computed in almost all cases, with a local reverse subdivision mask. As for most reverse subdivision processes [Mon04], the locality allows very quick reverse subdivision. However, the coarser mesh can be found only if the current mesh was generated by subdivision. Indeed, if the mesh is not generated by refinement, the points (from vertex v^{k+1}, edge e^{k+1} or face f^{k+1}) are not distinguishable from each other so the application of the reverse process can be found only in particular cases. One reason is that the number of vertices is determined by the applied subdivision. This drawback can be found in any reverse subdivision process whatever the mesh (triangular, quadrilateral…) and whatever the subdivision rule (Loop, Doo-Sabin, Catmull-Clark…).

6. REFERENCES

[Bar00] Bartels, R., Samavati, F., Reversing Subdivision Rules: Local Linear Conditions and Observations on Inner Products in the *Journal of Computational and Applied Mathematics*, **119**(1-2), pp. 29-67, July, 2000.

[Cat78] Catmull E. and Clark J., *Recursively generated B-spline surfaces on arbitrary topological meshes.* Computer Aided Design, **9**(6), pp. 350-355, 1978.

[Cha74] Chaikin G.. "An algorithm for High Speed Curve Generation." CGIP 3, p. 346 – 349, 1974.

[Doo78] Doo D. and Sabin M., *Behaviour of recursive subdivision surfaces near extraordinary points.* Computer Aided Design, **9**(6), p. 356-360, 1978.

[Dyn90] Dyn, N., Levin, D. and Micchelli, CA, *Using parameters to increase smoothness of curves and surfaces generated by subdivision.* Computer Aided Design, **7**(2), pp. 129-140, 1990.

[Loo87] Loop C., *Smooth Subdivision Surfaces Based on Triangles*. Department of Mathematics: Master's thesis, University of Utah, 1987.

[Lou97] Lounsbery, M., DeRose, T.D.and Warren, J., Multiresolution analysis for surfaces of arbitrary topological type. ACM Transactions On Graphics, **16**(1), pp. 34-73, 1997.

[Mon04] Mongkolnam, P., Razdan, A., and Farin G., *Reverse Engineering Using Loop Subdivision* Computer-Aided Design & Applications, **1**(1-4), pp 619-626, 2004.

[Sam98] Samavati F. and Bartels R.. Multiresolution curve and surface representation: Reversing subdivision rules by least-squares data fitting. Computer Graphics Forum, **18**(2), pp. 97-119, 1998.

[Sam02] Samavati, F. and Bartels, H.. Reversing subdivision using local linear conditions : Generating multiresolutions on regular triangular meshes. Technical report 2002-711-14, December 2, 2002.

[Sam02] Samavati, F., Mahdavi-Amiri, N. and Bartels R.. Multiresolution surfaces having arbitrary topologies by a reverse Doo subdivision method. Computer Graphics Forum, **21**(2), pp. 121-136, 2002.

[Sam03] Samavati, F.,. Pakdel, H. R, Smith, C. and Prusinkiewicz, P.. *Reverse Loop Subdivision*. Technical report 2003-730-33, November 4, 2003.