Subdivision surfaces

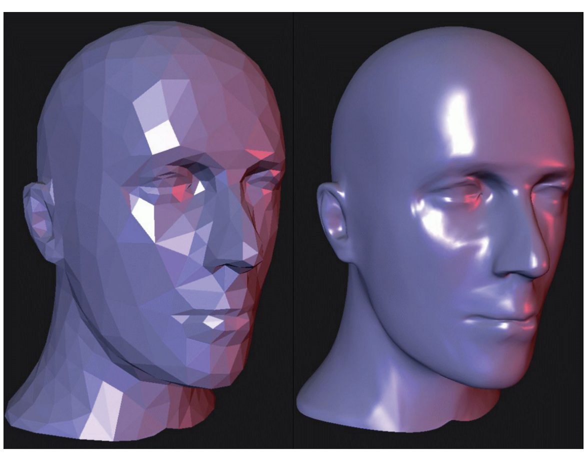
Reading

Recommended:

- Stollnitz, DeRose, and Salesin. *Wavelets for Computer Graphics: Theory and Applications,* 1996, section 10.2.
- DeRose, Kass, and Truong. Subdivision surfaces in character animation, SIGGRAPH '98, pp. 85-94.

Building complex models

We can extend the idea of subdivision from curves to surfaces…



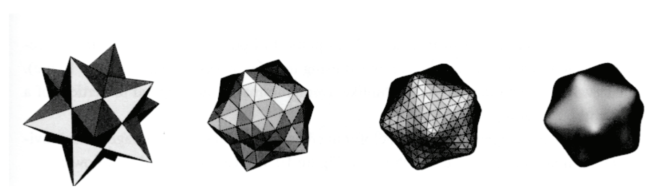
Subdivision surfaces

Chaikin's use of subdivision for curves inspired similar techniques for subdivision surfaces.

Iteratively refine a **control polyhedron** (or **control mesh**) to produce the limit surface

$$\sigma = \lim_{j \to \infty} M^j$$

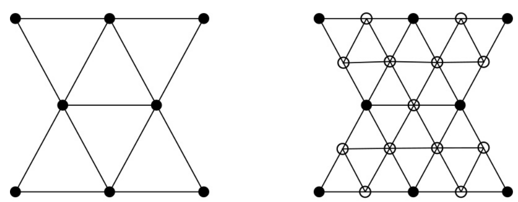
using splitting and averaging steps.



Triangular subdivision

There are a variety of ways to subdivide a poylgon mesh.

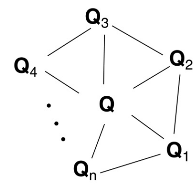
A common choice for triangle meshes is 4:1 subdivision – each triangular face is split into four subfaces:



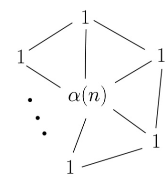
Original After splitting

Loop averaging step

Once again we can use **masks** for the averaging step:



Vertex neighorhood



Averaging mask
(before affine normalization)

$$\mathbf{Q} \leftarrow \frac{\alpha(n)\mathbf{Q} + \mathbf{Q}_1 + \cdots + \mathbf{Q}_n}{\alpha(n) + n}$$

where

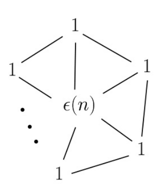
$$\alpha(n) = \frac{n(1-\beta(n))}{\beta(n)} \qquad \beta(n) = \frac{5}{4} - \frac{(3+2\cos(2\pi/n))^2}{32}$$

These values, due to Charles Loop, are carefully chosen to ensure smoothness – namely, tangent plane or normal continuity.

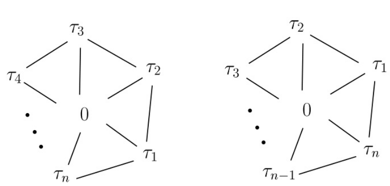
Note: tangent plane continuity is also know as G^1 continuity for surfaces.

Loop evaluation and tangent masks

As with subdivision curves, we can split and average a number of times and then push the points to their limit positions.



Evaluation mask
(before affine normalization)



Tangent masks

$$\mathbf{Q}^{\infty} = \frac{\varepsilon(n)\mathbf{Q} + \mathbf{Q}_1 + \cdots + \mathbf{Q}_n}{\varepsilon(n) + n}$$

$$\mathbf{T}_1^{\infty} = \tau_1(n)\mathbf{Q}_1 + \tau_2(n)\mathbf{Q}_2 + \cdots + \tau_n(n)\mathbf{Q}_n$$

$$\mathbf{T}_2^{\infty} = \tau_n(n)\mathbf{Q}_1 + \tau_1(n)\mathbf{Q}_2 + \cdots + \tau_{n-1}(n)\mathbf{Q}_n$$

where

$$\varepsilon(n) = \frac{3n}{\beta(n)} \qquad \tau_i(n) = \cos(2\pi i/n)$$

Note that the eigenvalues of the related subdivision matrix have the form: $\lambda_1 = 1 > \lambda_2 = \lambda_3 > \cdots \geq \lambda_n \geq 0$.

How do we compute the normal?

Recipe for subdivision surfaces

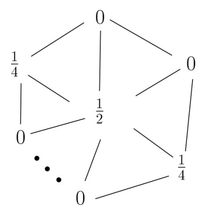
As with subdivision curves, we can now describe a recipe for creating and rendering subdivision surfaces:

- Subdivide (split+average) the control polyhedron a few times. Use the averaging mask.
- Compute two tangent vectors using the tangent masks.
- Compute the normal from the tangent vectors.
- Push the resulting points to the limit positions. Use the evaluation mask.
- Render!

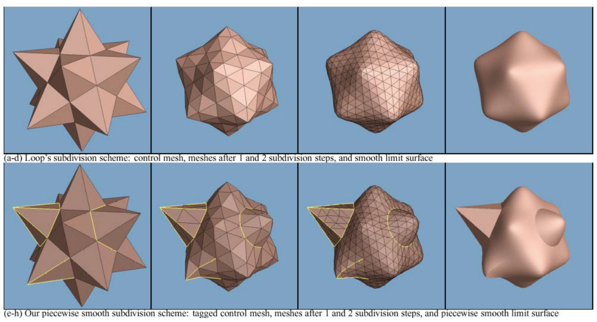
Adding creases without trim curves

In some cases, we want a particular feature such as a crease to be preserved. With NURBS surfaces, this required the use of trim curves.

For subdivision surfaces, we can just modify the subdivision mask:



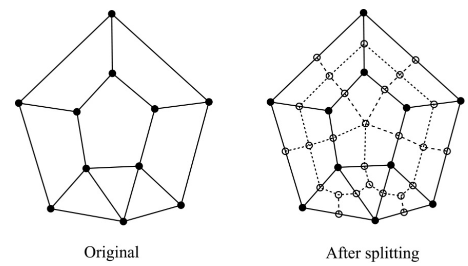
This gives rise to G^0 continuous surfaces (i.e., having positional but not tangent plane continuity)


(a-d) Loop's subdivision scheme: control mesh, meshes after 1 and 2 subdivision steps, and smooth limit surface
(e-h) Our piecewise smooth subdivision scheme: tagged control mesh, meshes after 1 and 2 subdivision steps, and piecewise smooth limit surface

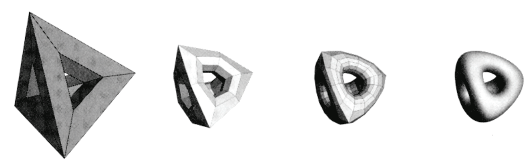
Face schemes

4:1 subdivision of triangles is sometimes called a **face scheme** for subdivision, as each face begets more faces.

An alternative face scheme starts with arbitrary polygon meshes and inserts vertices along edges and at face centroids:



Catmull-Clark subdivision:



Note: after the first subdivision, all polygons are quadilaterals in this scheme.

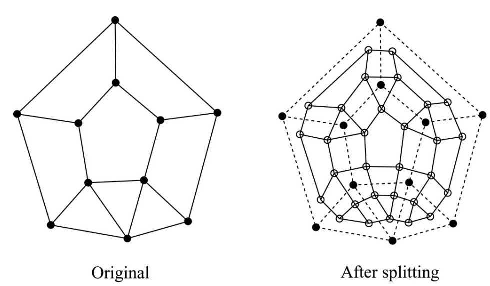
Creases without trim curves, cont.

Here's an example using Catmull-Clark surfaces (based on subdividing quadrilateral meshes):

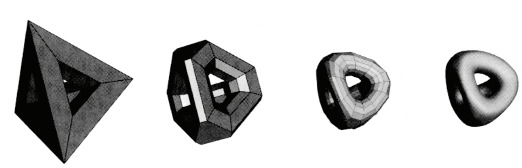
This particular example uses the hybrid technique of DeRose, et al., which applies sharp subdivision rules at some creases for a finite number of steps, and then switches to smooth subdivision, giving more gentle creases. This technique was used in Geri's Game.

Vertex schemes

In a **vertex scheme**, each vertex begets more vertices. In particular, a vertex surrounded by *n* faces is split into *n* sub-vertices, one for each face:



Doo-Sabin subdivision:



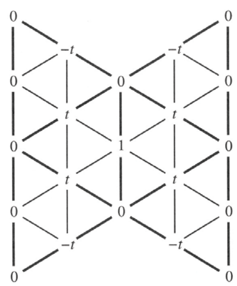
The number edges (faces) incident to a vertex is called its **valence**. Edges with only once incident face are on the **boundary**. After splitting in this subdivision scheme, all non-boundary vertices are of valence 4.

Interpolating subdivision surfaces

Interpolating schemes are defined by

- splitting
- averaging only new vertices

The following averaging mask is used in **butterfly subdivision**:



Setting t=0 gives the original polyhedron, and increasing small values of t makes the surface smoother, until t=1/8 when the surface is provably G^1.