Subdivision curves and surfaces

Brian Curless
CSE 457
Spring 2010

Reading
Recommended:

Note: there is an error in Stollnitz, et al., section A.5. Equation A.3 should read:

MN - VA

This is already fixed in the handout.

Subdivision curves

Idea:
- repeatedly refine the control polygon
  \[ P^3 \rightarrow P^2 \rightarrow P^0 \rightarrow \ldots \]
- curve is the limit of an infinite process
  \[ Q = \lim_{n \to \infty} P^n \]

Chaikin’s algorithm

Chaikin introduced the following “corner-cutting” scheme in 1974:
- Start with a piecewise linear curve
- Insert new vertices at the midpoints (the splitting step)
- Average each vertex with the “next” (clockwise) neighbor (the averaging step)
- Go to the splitting step

Old vertex                  New vertex

1. Split

2. Average

3. Split

4. Average
Averaging masks

The limit curve is a quadratic B-spline!

Instead of averaging with the nearest neighbor, we can generalize by applying an averaging mask during the averaging step:

\[ r = (\ldots, r_0, r_1, r_2, \ldots) \]

In the case of Chaikin’s algorithm:

\[ r = (0 \frac{1}{2} \frac{1}{2}) \]

Subdivide ad nauseum?

After each split-average step, we are closer to the limit curve.

How many steps until we reach the final (limit) position?

\[ \infty \]

Lane-Riesenfeld algorithm (1980)

Use averaging masks from Pascal’s triangle:

\[
\begin{align*}
\text{linear} & : (1) \\
\text{quadratic} & : (0 \frac{1}{2} \frac{1}{2}) \\
\text{cubic} & : (0 \frac{4}{9} \frac{4}{9} \frac{1}{9})
\end{align*}
\]

Recipe for subdivision curves

Can we push a vertex to its limit position without infinite subdivision? Yes!

After subdividing and averaging a few times, we can push each vertex to its limit position by applying an evaluation mask.

Each subdivision scheme has its own evaluation mask, mathematically determined by analyzing the subdivision and averaging rules.

For Lane-Riesenfeld cubic B-spline subdivision, we get

\[ \frac{1}{6} \begin{pmatrix} 1 & 4 & 1 \end{pmatrix} \]

Now we can cook up a simple procedure for creating subdivision curves:

- Subdivide (split-average) the control polygon a few times. Use the averaging mask.
- Push the resulting points to the limit positions. Use the evaluation mask.
DLG interpolating scheme (1987)

Slight modification to subdivision algorithm:
- splitting step introduces midpoints
- averaging step only changes midpoints

For DLG (Dyn-Levin-Gregory), use:

\[ a_{10} = 0 \quad a_{10} = \frac{1}{10} (-1, 5, 10, 5, -1) \]

Since we are only changing the midpoints, the points after the averaging step do not move.

Building complex models

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

Subdivision surfaces

Chaitin’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

\[ S = \lim_{n \to \infty} M^n \]

Subdivision surfaces

using splitting and averaging steps.

Triangular subdivision

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

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

\[ \text{valence} = \text{# of vertices in the 1-ring} \]
**Loop averaging step**

Once again we can use masks for the averaging step:

\[
Q_n = \frac{\alpha_n Q_0 + Q_1 + \cdots + Q_n}{\alpha_n + n}
\]

where

\[
\alpha_n = \frac{n(1 - \beta(n))}{\beta(n)} \quad \beta(n) = \frac{5}{4} \left(1 + 2 \cos \left(\frac{\pi n}{2}\right)\right)^2
\]

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

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

**Recipe for subdivision surfaces**

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

- Subdivide (split and average) the control polygon 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**

For NURBS surfaces, adding sharp features like creases required the use of trim curves.

For subdivision surfaces we can just modify the subdivision masks. E.g., we can mark some edges and vertices as "creases" and modify the subdivision mask for them (and their children):

This gives rise to $C^0$ continuous surfaces (i.e., having positional but not tangent plane continuity).
Catmull-Clark subdivision

4th subdivision of triangles is known as a face scheme for subdivision. Each face is replaced by 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 quadrilaterals 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 Gear's Game.

Interpolating subdivision surfaces

Interpolating schemes are defined by:

- splitting
- averaging only new vertices

The following averaging mask is used in butterfly subdivision:

Interpolating subdivision surfaces

Summary

What to take home:

- The meanings of all the boldfaced terms
- How to perform the splitting and averaging steps on subdivision curves
- How to perform mesh splitting steps for subdivision surfaces, especially Loop
- How to construct and render subdivision surfaces from their averaging masks, evaluation masks, and tangent masks.

Summary

Setting $h=0$ gives the original polygon mesh, and increasing small values of $h$ makes the surface smoother, until $h=1/3$ when the surface is probably $C^1$. 

Summary