MA 323 Geometric Modelling
Course Notes: Day 33
Subdivision Triangle Surfaces

David L. Finn

Today, we want to solve the Hermite style interpolation problem introduced for surface at the end of the previous day's notes. We will solve this by a subdivision scheme. This means that we will not attempt to give a formal expression for describing the surface, but rather provide an iterative scheme for generating the surface. We will not formally show that the iteration scheme converges to a limiting surface as that requires substantial analysis, rather we will outline the process for convergence. The advantage of the solution that we provide is that it in general G^1 smooth, at every point on the surface there is a tangent plane. Furthermore, the method generalizes to any triangular grid structure to topologically define a surface. This enables us to first specify the general shape of the desired surface, and then use the subdivision structure to refine and smooth the topological structure. We note there are exceptional circumstances where a tangent plane will not be well-defined on the surface.

31.1 Triangle Grid Structure

The method that we are discussing today is to solve the following interpolation problem.

> Given three points noncollinear points defining a plane (that is given a triangle) $p_{1,0,0}$, $p_{0,1,0}$, $p_{0,0,1}$ and vectors $n_{1,0,0}$, $n_{0,1,0}$, $n_{0,0,1}$ at each point $p_{i,j,k}$. The vector $n_{i,j,k}$ has base point $p_{i,j,k}$ and terminal point $p_{i,j,k}+n_{i,j,k}$. Find a surface $\mathcal{S}$ that contains the points p_0, p_1, p_2 and has the prescribed normal vectors n_0, n_1, n_2 at respectively each point p_0, p_1, p_2

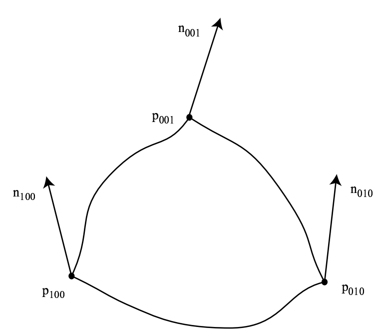


Figure 1: Data for a Single Triangle Subdivision Patch

The general outline of the method is to generate a point and normal vector that corresponds to the midpoint of each edge of the original triangle, see diagram below, and generate from a triangular grid of size 1 a triangular grid of size 2. The iteration scheme then takes over and we repeat the process on each of the four smaller triangles, so that after a second iteration we have a triangular grid of size 4 and sixteen smaller triangles, see diagrams below.

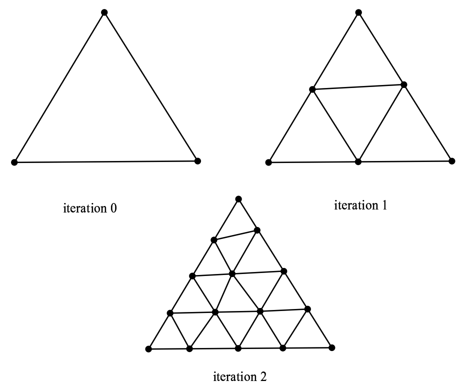


Figure 2: Iteration Scheme for Triangular Subdivision

The method for generating the new point and normal vector is done to depend only on the points and normal vectors for the corresponding edge. This implies that the for surfaces made out of multiple triangles, the surface will be globally G^1 independent of the arrangement of the remaining points and normal vectors. This is a great advantage over working with triangular Bezier patches or triangular Coons patches, where extra compatibility conditions need to be placed on the control points to achieve smooth joining across edges.

31.2 Computing New Edge Points

We compute the new edge point as a variation of the subdivision method for computing a point on a cubic Bezier curve, generated from Hermite data - a point and a tangent vector at that point - similar to the method for cubicly blending between two edge curves. To explain the method, we denote by q_A and q_B the corner points of the triangle and n_A and n_B the corresponding normal vectors. Our first goal is to generate the tangent vectors at the corner points. This is done by forming the vectors $v_{AB} = p_B - p_A$ as the vector from p_A to p_B and $v_{BA} = p_A - p_B$ as the vector from p_B to p_A. We then form the unit tangent vector t_A at p_A by projecting v_{AB} onto the plane with normal vector n_A and then normalizing, that is

$$u_A = v_{AB} - (v_{AB} \cdot n_A)\, n_A$$
$$t_A = u_A / \|u_A\|$$

and the tangent vector t_B at p_B by projecting v_{BA} onto the plane with normal vector n_B and then normalizing

$$u_B = v_{BA} - (v_{BA} \cdot n_B)\, n_B$$
$$t_B = u_B / \|u_B\|,$$

see diagram below.

We now construct the edge point as the point on the Bezier curve with control points $p_0 = p_A$, $p_1 = p_A + \lambda \| p_B - p_A\|\, t_A$, $p_2 = p_B + \lambda\, \|p_B - p_A\|\, t_B$ and $p_3 = p_B$ corresponding to $t = 1/2$. Working out this point, we find the point on the Bezier curve is

$$p_{AB} = \frac{1}{2}(p_A + p_B) + \frac{3}{8}\lambda\, \|p_B - p_A\|\, (t_A + t_B)$$

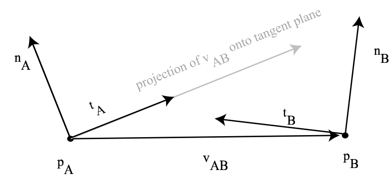


Figure 3: Construction of Tangent Vector for Edge Point

where λ is scaling parameter to shape the patch. The applets for the course set $\lambda = 1/3$. We obtain the normal vector for the edge point by finding the tangent vector to the Bezier curve, and then constructing a vector orthogonal to this vector from n_A and n_B as follows. First, the unit tangent vector to the Bezier curve at p_{AB} is

$$
\begin{aligned}
u_{AB} &= \frac{1}{2}\,(p_B - p_A) + \frac{1}{4}\,\lambda\,|p_B - p_A|\,(t_B - t_A) \\
t_{AB} &= u_{AB}/\|u_{AB}\|.
\end{aligned}
$$

Then, we take the average of the two normals $n = (n_A + n_B)/2$ and find the unit vector for orthogonal complement of n to t_{AB} as the surface normal vector for the edge point, that is defin

$$
\begin{aligned}
n &= \frac{1}{2}\,(n_A + n_B) \\
n' &= n - (n \cdot t_{AB})\,t_{AB} \\
n_{AB} &= n'/\|n'\|
\end{aligned}
$$

The reason for doing this is that the iteration scheme if it works should produce a curve that is close to a Bezier curve along the edge by the subdivision method for producing Bezier curves. The normal vector to the surface should be orthogonal to the tangent vector of the edge curve. We need to use the tangent vector to the curve produced to determine the normal vector, or else we will not get smoothness and the interpolate the desired normal vectors.

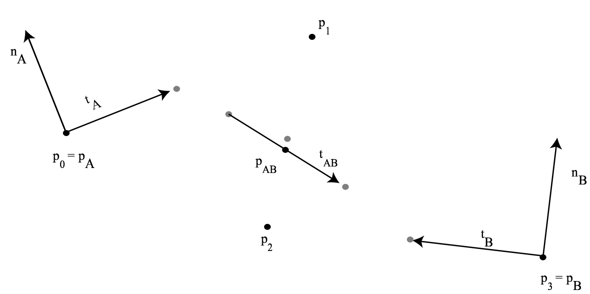


Figure 4: Construction of Edge Point

Below are a couple of gray-scale iterations in parallel view

31.3 Some Advantages and Problems with This Method

This method for constructing a surface has a couple of problems, as the iterative structure can develop singularities, points where the surface will not have a well-defined normal vector.

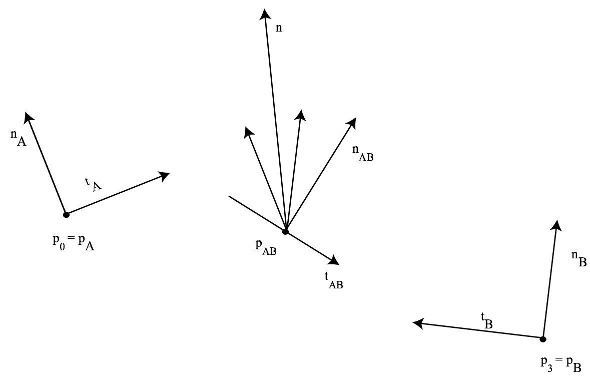
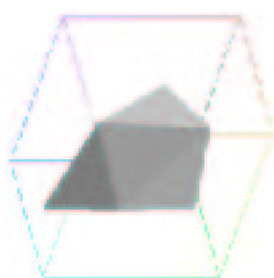
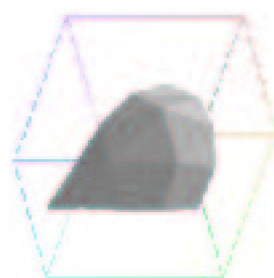


Figure 5: Construction of Edge Normal

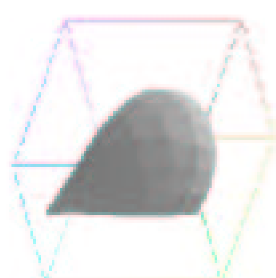
This will happen if the normal vectors at two points on the side of triangle A and B satisfy $n = (n_A + n_B)/2 = 0$, as then the orthogonal complement $n' = 0$, and the unit normal n_{AB} can not be defined. In addition, if $n = (n_A + n_B)/2$ is parallel to t_{AB} then $n' = 0$ and the unit normal n_{AB} can not be defined. These are exceptional circumstances for generating the original points in the first iteration. However, determining general conditions to show that this will remain true for all points generated in the triangle is not easy, or obvious to understand. However, it easy to construct general situations where a problem will arise. For instance, if any of the original normals point towards opposite sides of the base triangle, then there will be a singularity somewhere on the surface.



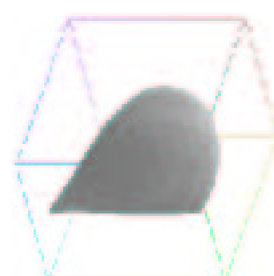
iteration 1



iteration 2



iteration 3



iteration 4

Figure 6: Iteration of Subdivision Method