Subdivision Surfaces

In this set of notes we cover the first five of the following aspects of subdivision surfaces:-

- Notation
- An example
- More examples
- Structures common to all bivariate subdivision
- Generating of all possible bivariate subdivisions
- Analysis of a subdivision scheme
- Interrogations of subdivision surfaces
- Research questions.

Notation

In the univariate context we denoted the operation of refinement by

$$P' = S.P$$

where P and P' were the sequences of vertices along the old and new polygons, and S was the **subdivision matrix**. The columns of S were all copies of the **mask** and the rows of S were the **stencils**.

The mask indicates the effects of a single old vertex on a set of new ones: the stencil indicates the effects on a single new vertex of a set of old ones.

The natural sequence of vertices along a polygon induced a corresponding sequence in the mask and the stencils.

In the bivariate case we still have the new vertices determined as linear combinations of the old, and so the matrix notation is still applicable. However, there is no longer a simple sequence of the vertices. We want to have a bivariate structure for P and P' and this implies that the mask and stencils will also have bivariate structure.

We can do this by considering in the first instance a tensor product structure. Imagine a regular grid of old vertices, indexed by subscripts k and l. We can apply a refinement operator to each line with a constant l subscript, giving a new grid, refined in the direction of increasing k. If we apply the same operator to this new grid in the other direction we can produce a final grid refined both ways.

This can be denoted by

$$P' = S.P.S^T$$

but this is not particularly helpful (although it does make it clear that repetition of the scheme will always lead to the same result, whether one direction is refined fully first and then the other. or the other first, or whether the two refinements are interleaved.)

What is useful is to make the summation , implicit in matrix notation, explicit, by using tensor notation. We then write the combined refinement as

$$P'_{ij} = S_i^k P_{kl} S_j^l.$$

Now that the exact summations involved are clear, we no longer need to keep the factors in their original sequence, and so we can first rewrite the equation as

$$P'_{ij} = S_i^k S_j^l P_{kl}$$

and then combine the S matrices into an S tensor

$$P'_{ij} = S_{ij}^{kl} P_{kl}.$$

Fixing kl means that S_{ij} is a mask. We can think of it as a two-dimensional column of the tensor. Because $S^{kl}_{(i-ka)(j-la)}$ is independent of k and l in a regular grid, we can then drop the superscripts.

Similarly, fixing ij means that S^{kl} is a stencil. There are more than one of these because a is greater than unity. Again, we can think of them as being two-dimensional rows.

For tensor product schemes, which are generated by using univariate schemes over a regular grid, the mask is just the outer product of the two masks, the stencils just the outer products of the stencils from the two schemes.

This means that we can still think in terms of the original

$$P' = S.P$$

equation. More importantly, it gives precise meaning if we denote the mask by a bivariate array of numbers. Such arrays are very evocative, as we can lay them out in a way which emphasises the symmetry of the configuration. If the edges of the original grid are superimposed on the array, the arity is also evident in this notation.

An example

The example we take is the simple tensor product of two quadratic B-spline refinements (Chaikin). The outer product of the two masks gives

$$
\begin{array}{ccc}
 & \begin{bmatrix} 1 & 3 & 3 & 1 \end{bmatrix} & /4 \\
\begin{bmatrix} 1 \\ 3 \\ 3 \\ 1 \end{bmatrix} /4 & \begin{bmatrix} 1 & 3 & 3 & 1 \\ 3 & 9 & 9 & 3 \\ 3 & 9 & 9 & 3 \\ 1 & 3 & 3 & 1 \end{bmatrix} & /16
\end{array}
$$

The centre of symmetry of the scheme is in the middle of the 4x4 matrix, and the arity in each direction is 2, and so this scheme maps four new vertices into each old face of the polygon.

The basis function is also the outer product of two univariate quadratic spline basis functions, so the basis function consists of nine pieces of biquadratic, meeting C1 across all edges.

If we look at the stencils instead of at the mask, we find that each new vertex depends only on the vertices of the old face in which the new vertex appears. We can therefore think of the scheme as putting:-

- a new face into each old face
- a new face across each old edge, and
- a new face across each old vertex

thus completing the manifold connectivity.

Arbitrary Topology

This view is one which we can apply to polyhedra which are not regular rectangular grids. If we generalise the rules a little, so that it is well defined how a new vertex is computed from the old vertices of its face, however many of those old vertices there may be, we can apply the procedure to very general polyhedra. In particular, because the procedure operates locally, we are not limited to polyhedra which are tesselations of the plane. The initial polyhedron can be of arbitrary topological type.

If we start with a primal scheme, the logic is the splitting of each face into regular faces by joining the edge midpoints to the face centre, and again the scheme can be applied to polyhedra of arbitrary topology.

Extraordinary Points

The initial polyhedron can have faces of other than 4 sides, and also vertices with other than 4 edges meeting.

After one iteration, there are still non-4 faces, one in each of the old non-4 faces, and also new non-4 faces, across the old non-4 vertices. All new vertices are 4-valent. The total number of non-4-nesses is constant, and this number remains invariant with refinement.

We call non-4-nesses of both kinds **extraordinary points**, a term which we also use to denote the positions of their eventual convergence.

Because each extraordinary point remains more or less in its original location during refinement, they eventually become separated by a 'sea' of regularity.

In the primal case, the faces are all regular after the first step, and so the extraordinary points are all vertices. In schemes which are *both*, the extraordinary points retain their original natures, and in schemes which are *neither*, they alternate, switching between vertices and faces at alternate steps in the refinement.

Piecewise structure

In the regular regions we know that the surface consists of a regular grid of biquadratic pieces. These pieces leave an n-sided hole around an extraordinary point.

After one iteration, the hole is smaller, because regularity emerges in additional groups of vertices, and thus new pieces of biquadratic are identified.

The overall structure is therefore of biquadratic patches forming a regular array in regular regions, and rings of biquadratics patches forming an infinite regress around the extraordinary points.

The surface is C1 everywhere except, possibly, at the extraordinary points.

More examples

This general picture applies to a wide variety of schemes.

- tensor products
- box-splines
- others

Tensor Products

Any univariate schemes can be used to form a tensor product. The one limitation which we impose to make the resulting scheme extensible to general topology is to insist that the same scheme be used in both directions.

We have tensor products of box-splines, and also tensor products of interpolating schemes such as the 4-pt scheme.

Box-splines

We can also generate box-spline tensor products from generating functions of two variables.

For example, the bilinear scheme can be generated from

$$(1+x)^2(1+y)^2$$

the biquadratic from

$$(1+x)^3(1+y)^3$$

and the bicubic from

$$(1+x)^4(1+y)^4.$$

The array of terms in the mask are naturally laid out according to the powers of x and y in them.

Box-splines are much more general than this, because we can also use factors which involve both of the variables. Thus

$$(1+x)(1+y)(1+xy)$$

expands into

$$\begin{array}{ccccccc} & & & xy^2 & + & x^2y^2 \\ + & y & + & 2xy & + & x^2y \\ + & 1 & + & x & & \end{array} \quad \text{or} \quad \begin{array}{ccc} & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & \end{array}$$

which can be regarded as the basis function for the refinement of a set of triangular facets over a regular triangular grid.

$$(1+x)(1+y)(1+xy)(x+y)$$

expands into the mask

$$\begin{array}{cccc} & 1 & 1 & \\ 1 & 2 & 2 & 1 \\ 1 & 2 & 2 & 1 \\ & 1 & 1 & \end{array} /4$$

In general, boxspline schemes have piecewise polynomial limit surfaces. The degree of the polynomial pieces is just two less than the total number of linear factors in the factorised form of the generating function.

The continuity is the degree, less the maximum number of factors pointing in the same direction.

Box-spline schemes are positive, so that they give limit surfaces inside the convex hull (and thus inside other hulls) of groups of original vertices.

Generating function notation makes the generation of the coefficients of large numbers of different schemes totally routine.

Others

The previous scheme,

$$\begin{array}{cccc} & 1 & 1 & \\ 1 & 2 & 2 & 1 \\ 1 & 2 & 2 & 1 \\ & 1 & 1 & \end{array} /4$$

can be achieved by applying a simpler scheme twice.

The scheme

$$\begin{array}{ccc} & 1 & \\ 1 & & 1 \\ & 1 & \end{array}$$

has its centre of symmetry in the middle, and so it calculates new vertices at the centres of the original edges. These give a grid of new points lying on a square grid at 45 degrees to the original. Repeating this process gives the previous scheme. We may thus regard the edge-midpoint scheme (called '*the simplest*' by Peters and Reif), as its square root.

Because the simplest scheme when applied twice gives an arity=2 scheme, it is described as a $\sqrt{2}$ scheme.

It is not the only one. This is a dual scheme, but a primal $\sqrt{2}$ scheme can be set up which creates a new vertex logically at the centre of every face, and keeps all the old vertices. If we make an interpolating scheme, the mask is

$$\begin{array}{ccccccc} & & -1 & & -1 & & \\ & 0 & & 0 & & 0 & \\ -1 & & 10 & & 10 & & -1 \\ & 0 & & 32 & & 0 & \\ -1 & & 10 & & 10 & & -1 \\ & 0 & & 0 & & 0 & \\ & & -1 & & -1 & & \end{array} /32$$

A simple B-spline-like scheme can be set up with mask

$$\begin{array}{ccccc} & & 1 & & \\ & 2 & & 2 & \\ 1 & & 4 & & 1 \\ & 2 & & 2 & \\ & & 1 & & \end{array} /8$$

There are also primal square root schemes on triangular lattices, where a new vertex is created (logically) at the centre of each triangular face and then new faces are formed by joining each new vertex to the old vertices of its own triangle and the new vertices of immediately adjacent triangles.

Hexagonal lattices also support refinement schemes. A simple one of arity=2 creates a new face in the interior of each old face, and then creates new hexagonal faces by joining each new vertex to the corresponding vertex in its old face. More complex is a pattern of arity $\sqrt{3}$ in which the new face in each old face is rotated through 30 degrees, each new vertex being joined to the nearest new vertex in an adjacent hexagon.

Butterfly

An important scheme in the literature is the Butterfly, an interpolating scheme defined over a triangulation. It also inserts new points logically at the midpoints of the edges, at exact positions which give quadratic precision, but it also retains the original vertices, thus dividing every triangle into four smaller ones.

Structures common to all bivariate subdivision

All the refinement schemes start from an underlying regular situation, in which a mesh matching a regular lattice is refined to another mesh of the same lattice. This may be because we have deliberately excluded really wild schemes. For example, a mesh in which each triangle is refined into three subtriangles by addition of an internal vertex, rapidly leads to vertices of extremely high valency. The maximum edge length never decreases, and so the scope for converging towards something smooth is tiny. This is not very interesting, and so we exclude it.

All of them support generalisation to irregular polyhedra by the handling of extraordinary points. *Primal* and *both* schemes need to handle extraordinary vertices, while *dual* and *both* schemes need to handle extraordinary faces. Schemes which are *neither* can be handled by considering their squares.

In general the actual values of the mask need to be changed at an extraordinary point and the values in a stencil for an extraordinary face.

Generation of all possible bivariate subdivisions

The examples listed in an earlier section seem to be plucked from the air. Historically this is an accurate picture, but recently it has been seen to be possible to generate all of them systematically in just a few families. These families contain many many more than were listed above.

What grid ?

The quadrilateral, triangular and hexagonal meshes are the only regular subdivisions of the plane. We need seek no more. Interestingly, the number of regular subdivisions of space of three dimensions is more limited, only the brick lattice existing which is fully regular.

What type ?

The four types described under curves, of *primal*, *dual*, *both* and *neither* are all that exist in two dimensions also. The reason why is that we expect high levels of symmetry, which can only be achieved if a centre of symmetry of the original mesh (a vertex or a face centre) maps into a centre of symmetry of the new mesh. Note that because triangular and hexagonal meshes have different symmetries for the two, they cannot support some of the types. A triangular mesh has a 6-symmetry for a vertex which would be broken if the scheme were *dual* or *neither*, and so triangular meshes support only *primal* and *both* schemes. Similarly, a face has a 6-symmetry in the hexagon mesh, which cannot therefore be mapped into a vertex without loss of symmetry. Hexagon meshes support only *dual* and *both* schemes.

Taking the grid and the type together, we have a total of eight categories to consider.

What arity ?

Here a big surprise arises. In the bivariate context arity is generally a complex number ! This may be seen if we look at how the old and new meshes of a given type have to be related. Consider primal schemes. Take one vertex of the old mesh (which is also a vertex of the new mesh) as origin, and lay out the new lattice. We may place an adjacent vertex in the old lattice at any of the vertices of the new one apart from the origin itself and its immediate neighbours. All of the vertices of the old lattice will then find themselves on new lattice vertices.

Clearly, the potential patterns form a countable infinity, which can be explored in sequence of the length of the side of the old lattice compared with that of the new.

Apart from the cases where the old lattice is aligned with the new, and the $\sqrt{2}$ and $\sqrt{3}$ schemes (and arities multiples of these) iteration of the refinement never returns to the original orientation. This is somewhat awkward if we have boundaries to our polygon.

However, if we apply alternately a scheme and its complex conjugate, the original orientation is reached at every other step, and boundaries are no longer quite so hard to handle. This is what any practical implementation is likely to do.

What width ?

Here we have a larger combinatorial explosion. The equivalent of width is choosing which new vertices in the mask shall have a non-zero value, and the number of possible combinations increases rapidly with the total number of non-zeros, even if we insist on maximal symmetry in the mask. There are two ways of controlling this. One is to observe that schemes with a very large number of non-zeros are going to be hard to implement anyway, and so any systematic exploration needs to start with the small ones, and probably does not need to go much further than a hundred or so. The other is related to the support analysis, elaborated below, which suggests that there is practical advantage in schemes which fill successive possible convex outlines fairly evenly, rather than having outliers.

What values for the coefficients ?

This is a design question. For each possible type, arity and width, the choice of coefficients is made to satisfy criteria which are described more fully under analysis, below.

There are two main classes of scheme: the ones with positive mask (B-spline-like) and the interpolating schemes. However, there is also an important class of the quasi-interpolants, which have the property of interpolating exactly if the original control points happen to lie on a polynomial of limited degree, but do not have to interpolate otherwise. These have hardly been considered in the subdivision context.

Yet other schemes may turn out to be important even if they do not lie in any of these three, if they turn out to have good properties under analysis.