Lecture 9:
Subdivision surfaces

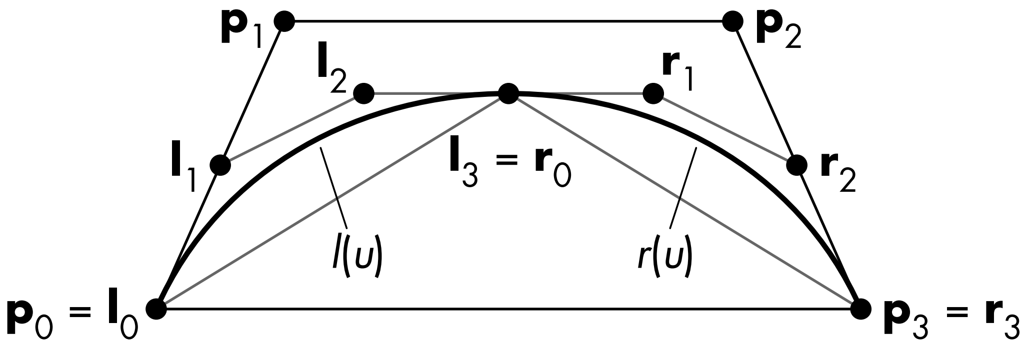
Topics:

1. Subdivision of Bezier curves
2. Chaikin's scheme
3. General subdivision schemes
4. Bi-quadratic and bi-cubic subdivision
5. Subdivision surfaces: Catmull-Clark and Loop

Michael S. Floater, Oslo, Oct. 2003

Subdivision of Bezier Curves

We saw in the last chapter how the de Casteljau algorithm both evaluates the curve *and* divides it into two.



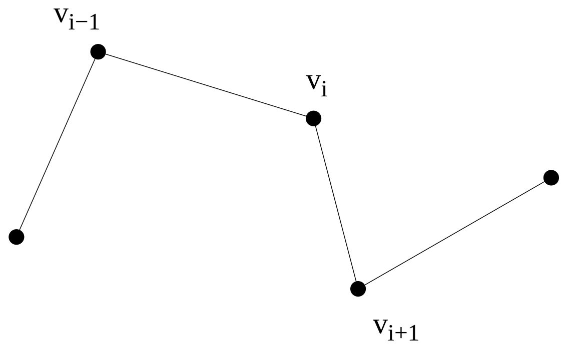
If we divide a cubic curve at its (parametric) midpoint, the initial control points p_0, p_1, p_2, p_3 are replaced by the new control points

$$\begin{aligned} \ell_0 &= p_0 \\ \ell_1 &= (p_0 + p_1)/2, \\ \ell_2 &= (p_0 + 2p_1 + p_2)/4, \\ \ell_3 = r_0 &= (p_0 + 3p_1 + 3p_2 + p_3)/4, \\ r_1 &= (p_1 + 2p_2 + p_3)/4, \\ r_2 &= (p_2 + p_3)/2, \\ r_3 &= p_3. \end{aligned}$$

Under repeated division, called **subdivision**, the control polygon converges to the curve. After only a few iterations the polygon is so close to the curve that we can simply render the polygon rather than the curve.

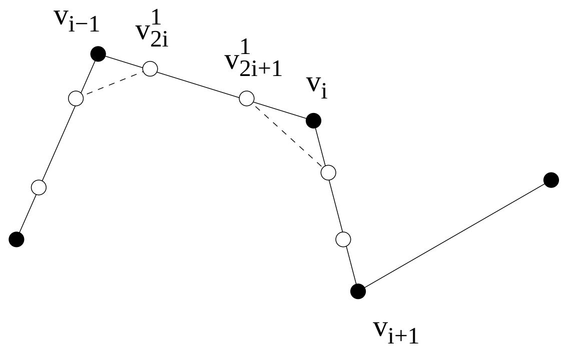
Subdivision curves

A **subdivision curve** is a curve generated by iterative refinement of a given polygon, called the **control polygon**. The limit curve can be rendered by simply rendering the polygon resulting from sufficiently many refinements. Both Bezier curves and spline curves are subdivision curves. For example, Chaikin's scheme generates a C^1 quadratic spline curve with uniform knots.



From a control polygon $\ldots, v_{i-1}, v_i, v_{i+1}, \ldots$, we generate a refined polygon by the rule

$$v^1_{2i} = \frac{3}{4} v_{i-1} + \frac{1}{4} v_i,$$
$$v^1_{2i+1} = \frac{1}{4} v_{i-1} + \frac{3}{4} v_i.$$

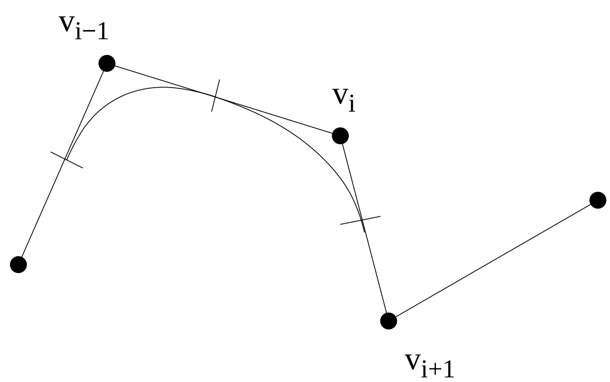


The full subdivision scheme is as follows.

1. Set $v_i^0 = v_i$, for all $i \in \mathbf{Z}$.
2. For $n = 1, 2, \ldots$, set

$$v_{2i}^n = \frac{3}{4} v_{i-1}^{n-1} + \frac{1}{4} v_i^{n-1},$$
$$v_{2i+1}^n = \frac{1}{4} v_{i-1}^{n-1} + \frac{3}{4} v_i^{n-1}.$$

The number of points doubles at each iteration. Here is the limiting curve:



The general (linear) subdivision scheme is

$$v_i^n = \sum_{k\in\mathbf{Z}} a_{i-2k} v_k^{n-1},$$

where $a_0, a_1, \ldots, a_m$ is the (finite) **subdivision mask** (all other a_i are zero). The mask for Chaikin's scheme is

$$(\, a_0 \quad a_1 \quad a_2 \quad a_3 \,) = (\, \tfrac{1}{4} \quad \tfrac{3}{4} \quad \tfrac{3}{4} \quad \tfrac{1}{4} \,)\,.$$

The mask can be split into two masks, for even and odd indexes separately:

$$\begin{aligned} v_{2i} &= \sum_{k\in\mathbf{Z}} a_{2k} v_{i-k}^{n-1}, \\ v_{2i+1} &= \sum_{k\in\mathbf{Z}} a_{2k+1} v_{i-k}^{n-1}, \end{aligned}$$

In Chaikin's scheme, these equations become

$$\begin{aligned} v_{2i}^n &= a_0 v_i^{n-1} + a_2 v_{i-1}^{n-1} = \frac{1}{4} v_i^{n-1} + \frac{3}{4} v_{i-1}^{n-1}, \\ v_{2i+1}^n &= a_1 v_i^{n-1} + a_3 v_{i-1}^{n-1} = \frac{3}{4} v_i^{n-1} + \frac{1}{4} v_{i-1}^{n-1}, \end{aligned}$$

and the two masks are

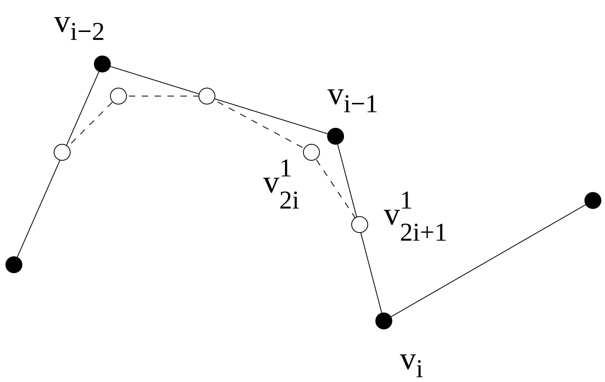
$$(\, a_0 \quad a_2 \,) = (\, \tfrac{1}{4} \quad \tfrac{3}{4} \,) \qquad \text{and} \qquad (\, a_1 \quad a_3 \,) = (\, \tfrac{3}{4} \quad \tfrac{1}{4} \,)\,.$$

Another example is a C^2 cubic spline curve (again with uniform knots). The mask is

$$(a_0 \quad a_1 \quad a_2 \quad a_3 \quad a_4) = \frac{1}{8} (1 \quad 4 \quad 6 \quad 4 \quad 1) .$$

If we split into the two masks (a_0, a_2, a_4) and (a_1, a_3), we get the scheme

$$\begin{aligned} v^n_{2i} &= \frac{1}{8}(v^{n-1}_i + 6v^{n-1}_{i-1} + v^{n-1}_{i-2}), \\ v^n_{2i+1} &= \frac{1}{2}(v^{n-1}_i + v^{n-1}_{i-1}). \end{aligned}$$

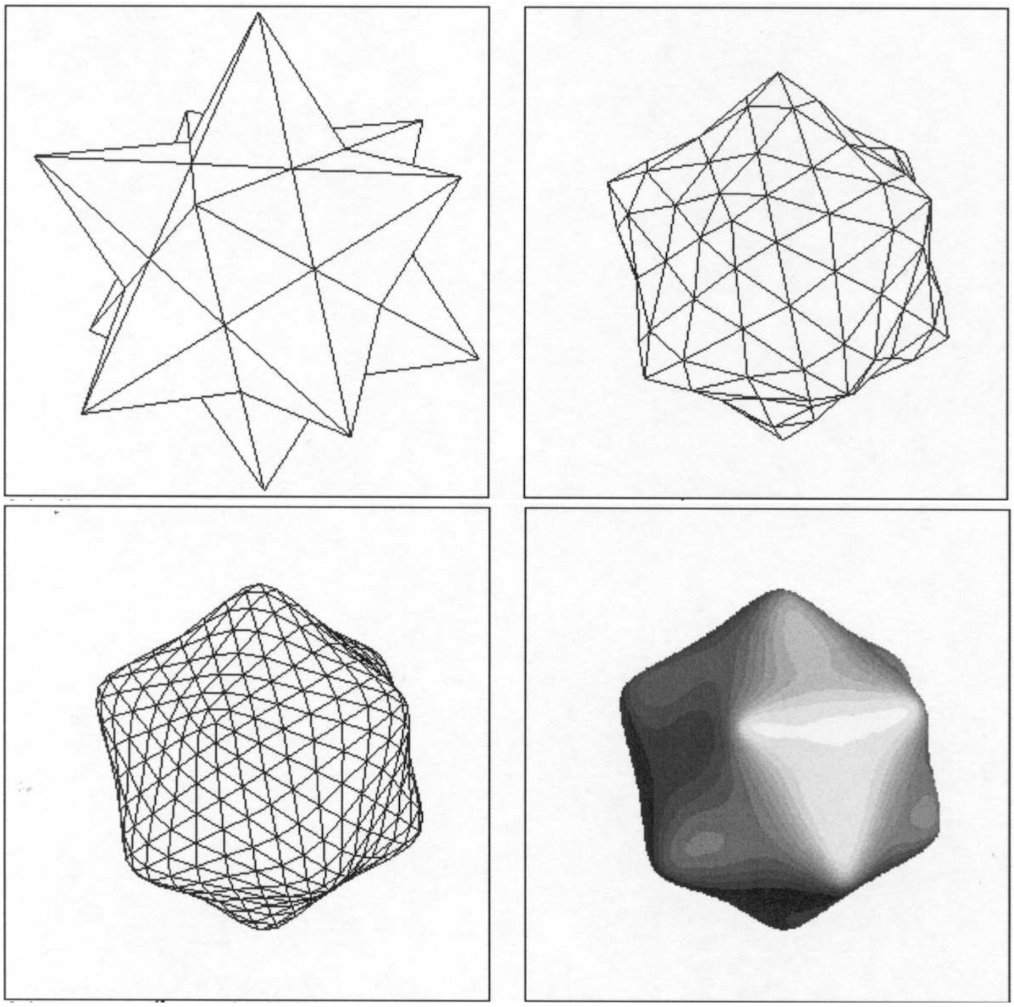


A uniform C^{d-1} spline curve of degree d can be generated by the mask

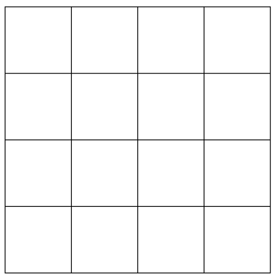
$$(a_0 \quad a_1 \quad \ldots \quad a_{d+1}) = \frac{1}{2^d} (\tbinom{d+1}{0} \quad \tbinom{d+1}{1} \quad \ldots \quad \tbinom{d+1}{d+1}) .$$

Subdivision surfaces

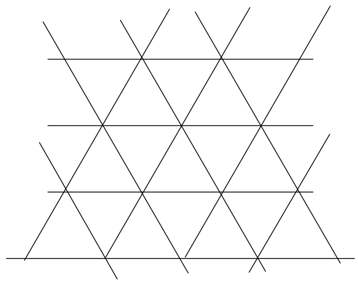
These are generated by iterative refinement of a polygonal mesh, usually with four-sided faces (quadrilateral meshes) or three-sided faces (triangle meshes).



For **uniform** (‘structured’) meshes, the limit surface is a spline surface. We get a tensor-product spline surface from a rectangular mesh, and a ‘box-spline’ surface from a triangular mesh.

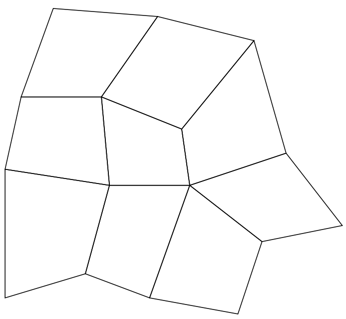


Rectangular grid

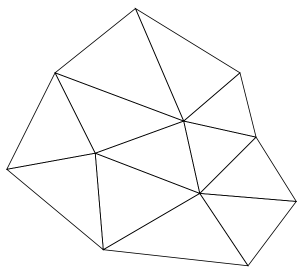


Triangular grid

For **non-uniform** (‘unstructured’) meshes, the limit surface has no closed form. However, the surface is locally a spline surface, except at so-called **extraordinary points**.



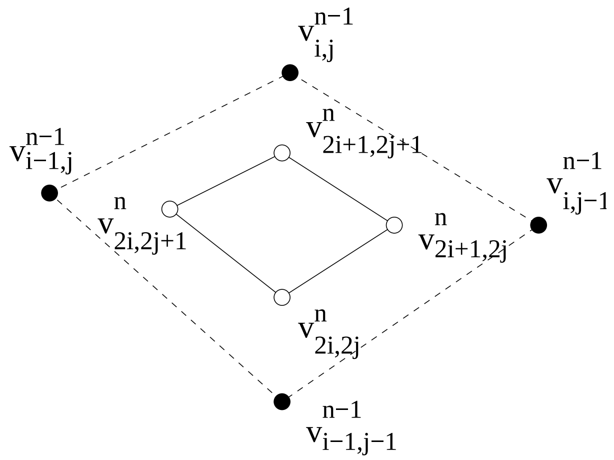
Quadrilateral mesh



Triangular mesh

Tensor-product subdivision on rectangular grids.

Example 1. Chaikin (C^1 biquadratic).



$$\begin{aligned} v^n_{2i,2j} &= \frac{1}{16}(9v^{n-1}_{i-1,j-1} + 3v^{n-1}_{i,j-1} + 3v^{n-1}_{i-1,j} + v^{n-1}_{i,j}) \\ v^n_{2i+1,2j} &= \frac{1}{16}(3v^{n-1}_{i-1,j-1} + 9v^{n-1}_{i,j-1} + v^{n-1}_{i-1,j} + 3v^{n-1}_{i,j}) \\ v^n_{2i,2j+1} &= \frac{1}{16}(3v^{n-1}_{i-1,j-1} + v^{n-1}_{i,j-1} + 9v^{n-1}_{i-1,j} + 3v^{n-1}_{i,j}) \\ v^n_{2i+1,2j+1} &= \frac{1}{16}(v^{n-1}_{i-1,j-1} + 3v^{n-1}_{i,j-1} + 3v^{n-1}_{i-1,j} + 9v^{n-1}_{i,j}). \end{aligned}$$

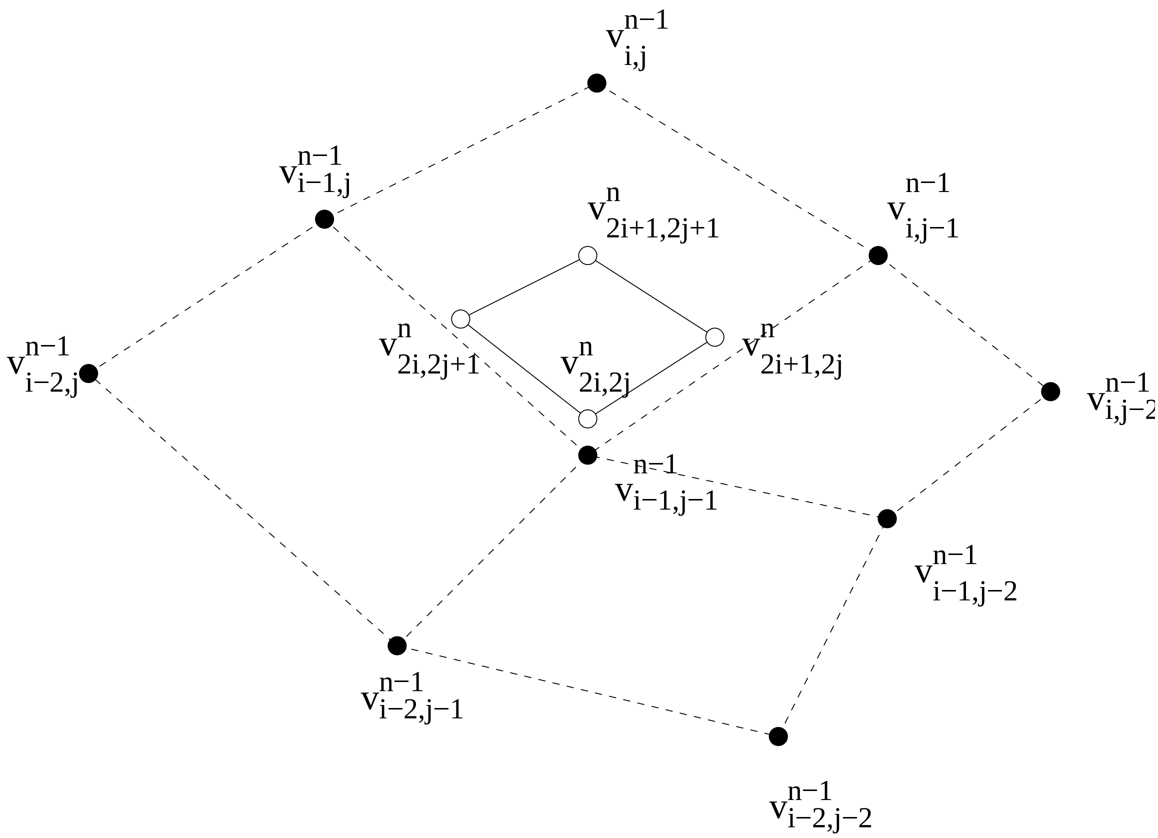
There are four submasks

$$\frac{1}{16}\begin{pmatrix} 3 & 1 \\ 9 & 3 \end{pmatrix}, \quad \frac{1}{16}\begin{pmatrix} 1 & 3 \\ 3 & 9 \end{pmatrix}, \quad \frac{1}{16}\begin{pmatrix} 9 & 3 \\ 3 & 1 \end{pmatrix}, \quad \frac{1}{16}\begin{pmatrix} 3 & 9 \\ 1 & 3 \end{pmatrix}.$$

They are tensor-products of the quadratic curve masks. For example

$$\frac{1}{16}\begin{pmatrix} 3 & 1 \\ 9 & 3 \end{pmatrix} = \frac{1}{4}\begin{pmatrix} 1 \\ 3 \end{pmatrix} \frac{1}{4}\begin{pmatrix} 3 & 1 \end{pmatrix}.$$

Example 2. C^2 bicubic.



The mask for cubic curves is

$$(\, a_0 \quad a_1 \quad a_2 \quad a_3 \quad a_4 \,) = \frac{1}{8} (\, 1 \quad 4 \quad 6 \quad 4 \quad 1 \,)\,.$$

and the two submasks are

$$\frac{1}{8} (\, 1 \quad 6 \quad 1 \,) \qquad \text{and} \qquad \frac{1}{2} (\, 1 \quad 1 \,)\,.$$

If we take tensor-products of these two submasks we get the four bicubic masks

$$\underbrace{\frac{1}{64}\begin{pmatrix}1 & 6 & 1\\ 6 & 36 & 6\\ 1 & 6 & 1\end{pmatrix}}_{\text{Mask A}}, \quad \underbrace{\frac{1}{16}\begin{pmatrix}1 & 1\\ 6 & 6\\ 1 & 1\end{pmatrix}, \quad \frac{1}{16}\begin{pmatrix}1 & 6 & 1\\ 1 & 6 & 1\end{pmatrix}}_{\text{Masks B}}, \quad \underbrace{\frac{1}{4}\begin{pmatrix}1 & 1\\ 1 & 1\end{pmatrix}}_{\text{Mask C}}.$$

These are used to compute the four new points

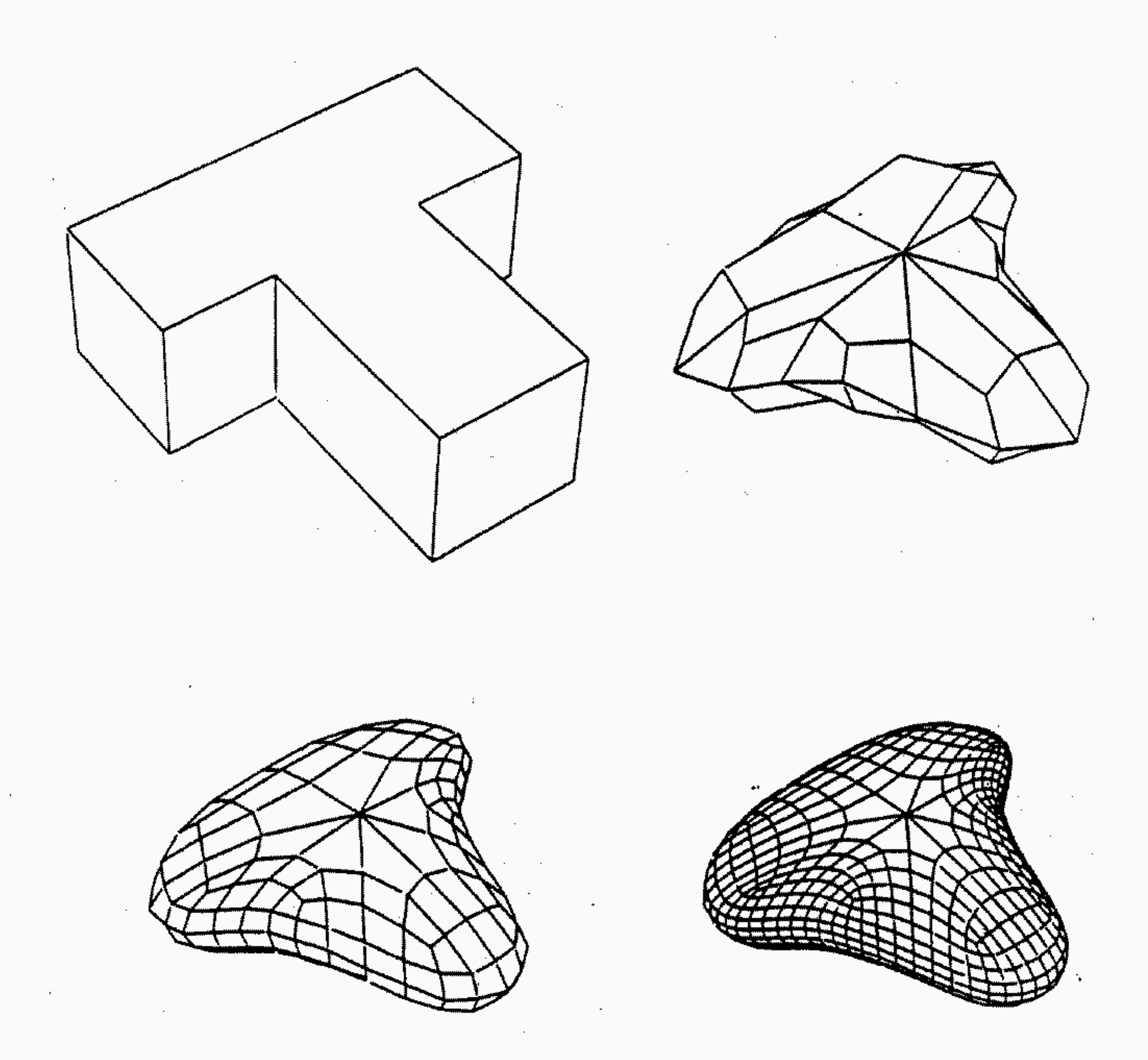
$$\begin{pmatrix} v^n_{2i+1,2j} & v^n_{2i+1,2j+1}\\ v^n_{2i,2j} & v^n_{2i,2j+1}\end{pmatrix}$$

from the old points

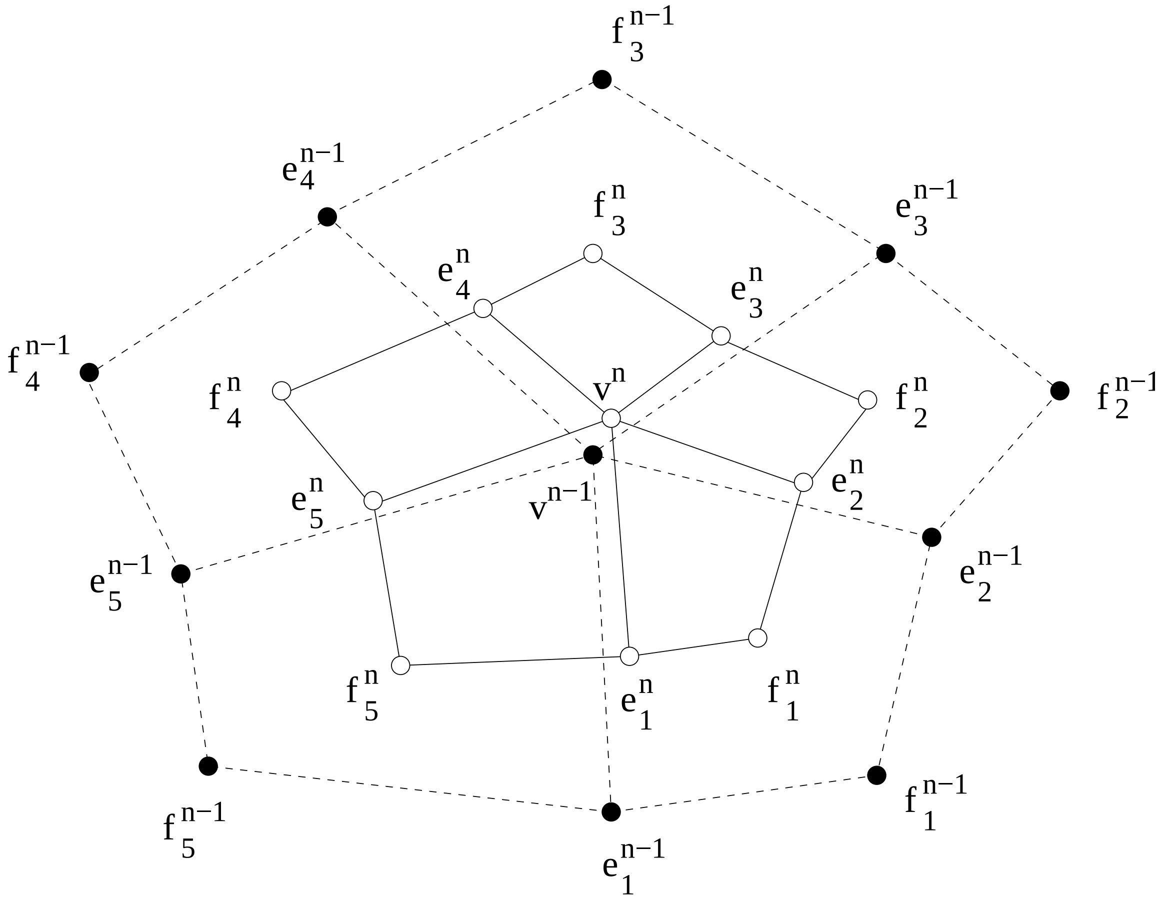
$$\begin{pmatrix} v^{n-1}_{i-2,j} & v^{n-1}_{i-1,j} & v^{n-1}_{i,j}\\ v^{n-1}_{i-2,j-1} & v^{n-1}_{i-1,j-1} & v^{n-1}_{i,j-1}\\ v^{n-1}_{i-2,j-2} & v^{n-1}_{i-1,j-2} & v^{n-1}_{i,j-2}\end{pmatrix}.$$

Catmull-Clark subdivision surfaces

This is a generalization of the C^2 **bicubic** scheme to an arbitrary quadrilateral mesh. The limit surface is C^2 except at extraordinary points.



It is enough to define the masks associated with the following figure. In the figure, 5 faces meet at the vertex v. In general there will be N faces. In the 'canonical' case we have $N = 4$.



As for the $N = 4$ bicubic case, there are three types of points: vertex points v, edge points e, and face points f, and there are three associated masks.

The algorithm goes in three steps.

Step 1. Compute the new face points. We use Mask C as before:

$$f_i^n = \frac{1}{4}(v^{n-1} + e_i^{n-1} + e_{i+1}^{n-1} + f_i^{n-1}).$$

Step 2. Compute the new edge points. We use Mask B as before:

$$e_i^n = \frac{1}{16}(e_{i-1}^{n-1} + f_{i-1}^{n-1} + 6v^{n-1} + 6e_i^{n-1} + e_{i+1}^{n-1} + f_i^{n-1}).$$

Using the new face points f_i^n computed in the first step, this computation reduces to

$$e_i^n = \frac{1}{4}(v^{n-1} + e_i^{n-1} + f_{i-1}^n + f_i^n).$$

Step 3. Compute the new vertex point. For $N = 4$ the rule for Mask A is

$$v^n = \frac{1}{64}\Big(36v^{n-1} + 6\sum_{i=1}^{4} e_i^{n-1} + \sum_{i=1}^{4} f_i^{n-1}\Big),$$

which can be expressed as

$$v^n = \frac{1}{4}\Big(2v^{n-1} + \frac{1}{4}\sum_{i=1}^{4} e_i^{n-1} + \frac{1}{4}\sum_{i=1}^{4} f_i^n\Big).$$

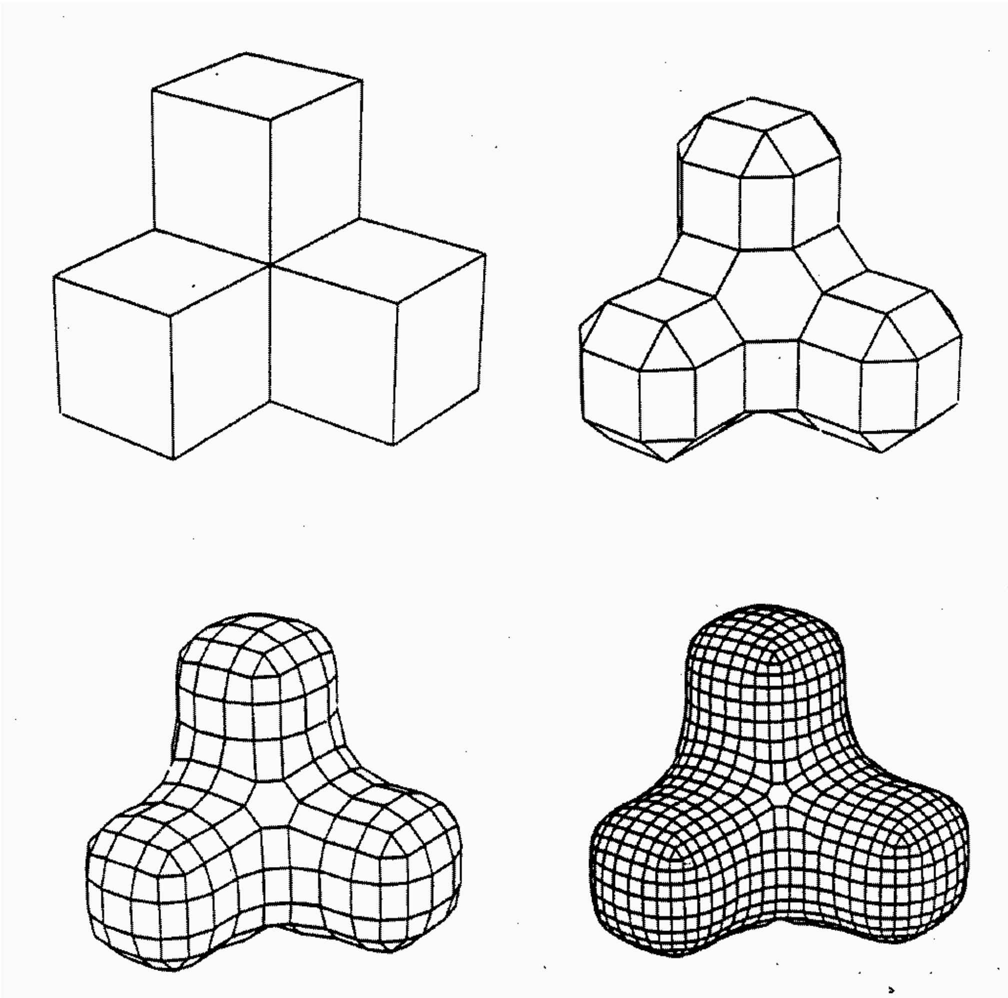
Catmull and Clark proposed the generalization

$$v^n = \frac{1}{N}\Big((N-2)v^{n-1} + \frac{1}{N}\sum_{i=1}^{N} e_i^{n-1} + \frac{1}{N}\sum_{i=1}^{N} f_i^n\Big).$$

This formula ensures C^1 continuity at the extraordinary points. It can be shown that C^2 continuity at extraordinary points is impossible without using larger masks.

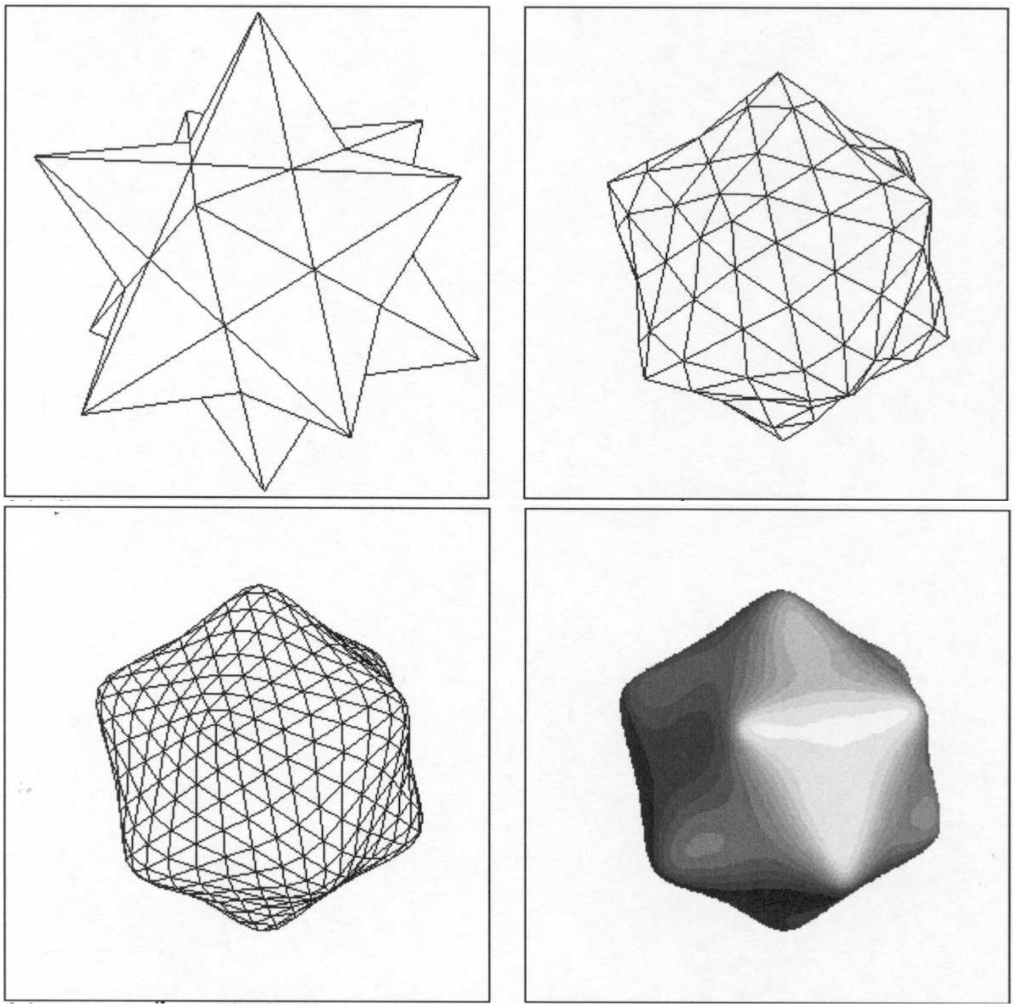
Doo-Sabin subdivision

As Catmull-clark subdivision surfaces generalize C^2 bicubic spline surfaces, Doo-Sabin subdivision surfaces generalize C^1 biquadratic spline surfaces. Tangent plane (C^1) continuity is again achieved at the extrordinary points. We will not give the details, just illustrate with the following example.

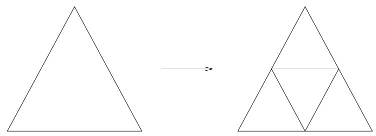


Loop subdivision

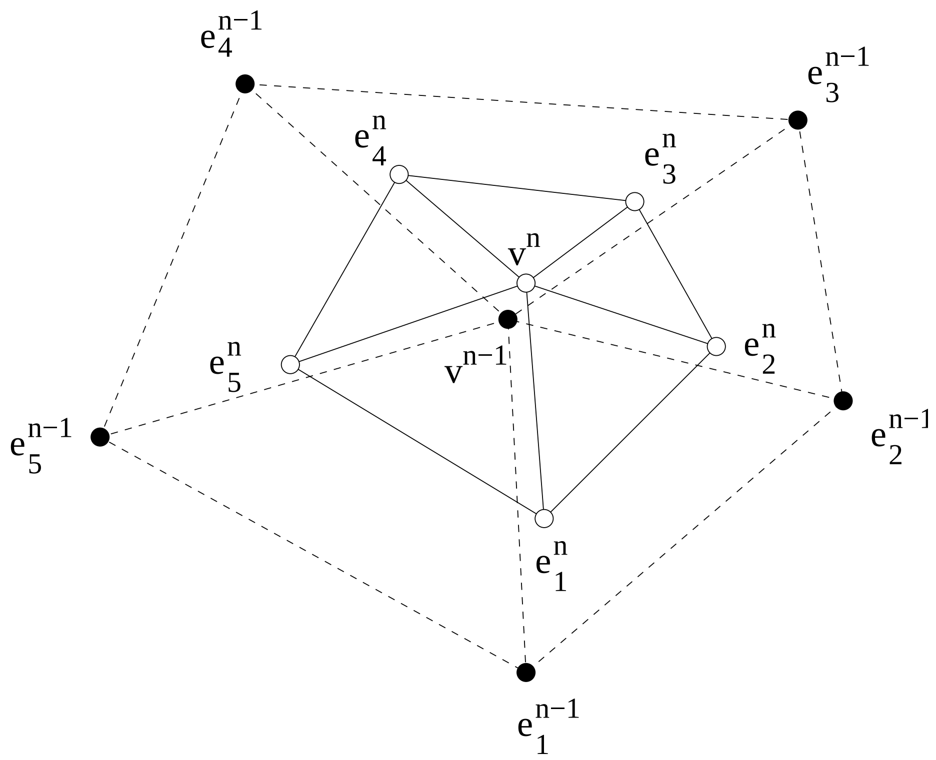
This is a subdivision scheme for arbitrary triangle meshes, based on so-called 'box-splines' (which is beyond the scope of this course), specifically C^2 quartic 'box-splines.



In this scheme we only compute vertex points and edge points, so there are only two masks. After one subdivision step each former triangle is replaced by four, a so-called **1-4 split**.



Suppose we have the situation of the figure below.



Here the number N of neighbouring triangles is 5. The 'canonical' case is $N = 6$ in which case the scheme reduces to 'box-spline' subdivision, yielding a C^2 surface. The algorithm has just two steps.

Step 1. Compute the new edge points by the rule

$$e_i^n = \frac{1}{8}(3v^{n-1} + 3e_i^{n-1} + e_{i-1}^{n-1} + e_{i+1}^{n-1}).$$

Step 2. Compute the new vertex points. The rule for 'box-splines' in the case $N = 6$ is

$$v^n = \frac{5}{8}v^{n-1} + \frac{3}{8}\Big(\frac{1}{6}\sum_{i=1}^{6} e_i^{n-1}\Big).$$

Loop proposed the generalization

$$v^n = \alpha_N v^{n-1} + (1 - \alpha_N)\Big(\frac{1}{N}\sum_{i=1}^{N} e_i^{n-1}\Big),$$

and showed that with the weighting

$$\alpha_N = \Big(\frac{3}{8} + \frac{1}{4}\cos(2\pi/N)\Big)^2 + \frac{3}{8},$$

the limit surface is C^1 at the extraordinary points. The surface is a generalization of a box-spline surface because $\alpha_6 = \frac{5}{8}$.