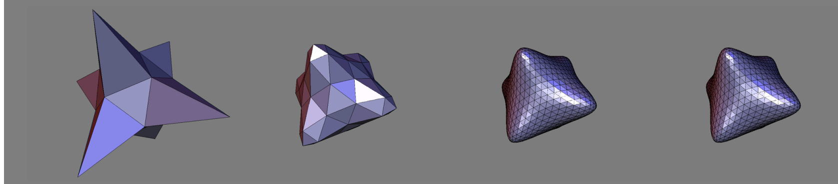
EECS 487: Interactive Computer Graphics

Lecture 39:
- (B-spline) Subdivision and surfaces

Subdivision Surfaces

How do you render a smooth surface?
- start with a polygonal control mesh
- cut corners to smooth: recursively subdivide into larger and larger number of polygons by adding new vertices/faces
- the limit surface is smooth
- mesh representation must enable efficient implementation of subdivision rules



Durand,James

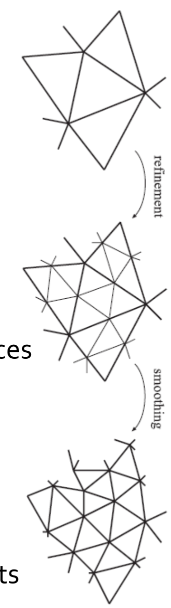
Subdivision Surfaces

Generate smooth surfaces from a given polygonal mesh (polyhedron) with guaranteed continuity
- can handle meshes of arbitrary topology
- implementation and application is straightforward and intuitive
- analysis of continuity is mathematically involved

Originally extensions of B-spline surfaces
- Doo-Sabin scheme produces quadratic B-spline surfaces
- Catmull-Clark scheme produces cubic B-spline surfaces
- Loop subdivision generalizes quartic box-spline

[TP3, Bischoff&Kobbelt]

Subdivision Concepts



Start with initial, discrete representation
- control points, line segments (for curves), polygons (for surfaces)

Repeated application of subdivision rules to make smoother surface
- topological splitting/refinement: how to add vertices
- smoothing/averaging: where to place vertices
- special treatment of extraordinary vertices and surface boundaries

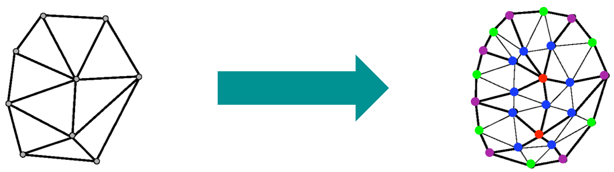
Limit surface (or curve)
- the "mathematical" result after infinite refinements

Gleicher, Funkhouser,

Smoothing

A set of scalars m_i, $1 \le i \le n$, applied to a set of n vertices $\mathbf{v}_i$ to generate a new vertex $\mathbf{w}$:

$$\mathbf{w} = \frac{\sum_{i=1}^{n} m_i \mathbf{v_i}}{\sum_{i=1}^{n} m_i}$$

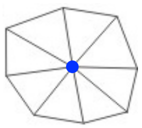


TP3, Gleicher, Funkhouser,

Interior and Boundary Vertices

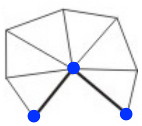
Interior vertices:

- for a closed polyhedron all vertices are interior vertices
- an epsilon neighborhood is homeomorphic to a closed disk



Boundary vertices:

- vertices that make up the "skirt" of a polyhedron
- an edge linking 2 boundary vertices is always shared by one face of the polyhedron

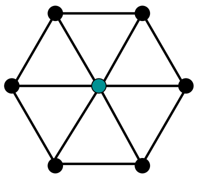


Hart/Carr

Ordinary and Extraordinary Vertices

Valence/degree of a vertex: number of edges incident to vertex

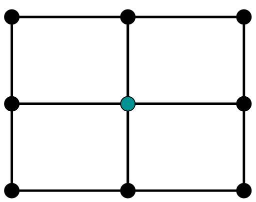
- most schemes have an "ideal" valence for which the limit surface converges to a spline surface, except at extraordinary vertices



triangular mesh
valence 6

Extraordinary vertices have different valence than ordinary vertices

- subdividing a mesh does not add nor remove extraordinary vertices
- make up rules for extraordinary vertices to keep the surface "smooth", though at lower degree of continuity



quad mesh
valence 4

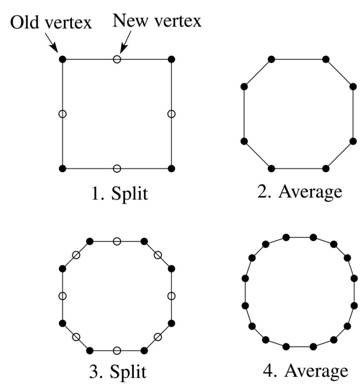
Hart/Carr

Subdivision Curves

Start with a piecewise linear curve

Chaikin's algorithm (1974):

- refinement: insert new edge vertex midpoint on each edge
- smoothing: average each vertex with the clockwise neighbor
- repeat



Curless

Chaikin's Curve Subdivision

Let initial vertices on control polygon be $\mathbf{v}_i$

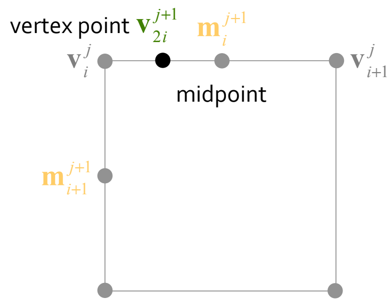
superscript, not power

Vertices at refinement level j are $\mathbf{v}_i^j$, thus $\mathbf{v}_i = \mathbf{v}_i^0$

1. refinement: insert new midpoint ($\mathbf{m}_i^{j+1}$) vertex on each edge
2. smoothing: average each vertex with the clockwise neighbor to create new vertex point ($\mathbf{v}_{2i}^{j+1}$):

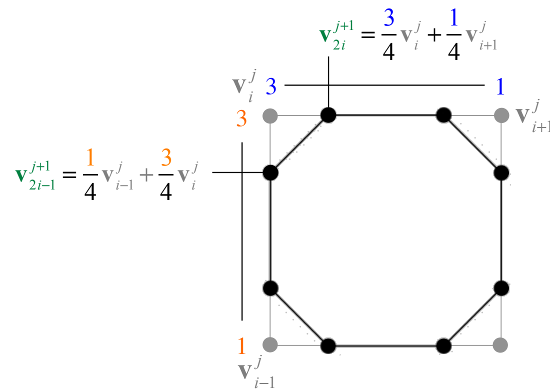
$$\begin{aligned}\mathbf{v}_{2i}^{j+1} &= \tfrac{1}{2}\mathbf{v}_i^j + \tfrac{1}{2}\mathbf{m}_i^{j+1} \\ &= \tfrac{1}{2}\mathbf{v}_i^j + \tfrac{1}{2}\left(\tfrac{1}{2}\mathbf{v}_i^j + \tfrac{1}{2}\mathbf{v}_{i+1}^j\right) \\ &= \tfrac{3}{4}\mathbf{v}_i^j + \tfrac{1}{4}\mathbf{v}_{i+1}^j\end{aligned}$$

3. averaging mask: (3 1)
   (also written as (0 ½ ½)
   for $0\mathbf{m}_{i-1}^{j+1} + \tfrac{1}{2}\mathbf{v}_i^j + \tfrac{1}{2}\mathbf{m}_i^{j+1}$)
4. continue on next slide . . .



Chaikin's Curve Subdivision

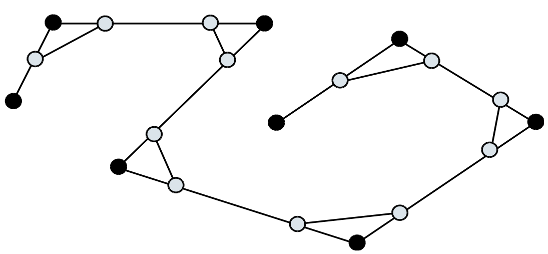
4. apply averaging masks: replace each vertex $\mathbf{v}_i^j$ with 2 vertices using the averaging masks (3 1) and (1 3):



5. connect all new vertex points to form refined curve

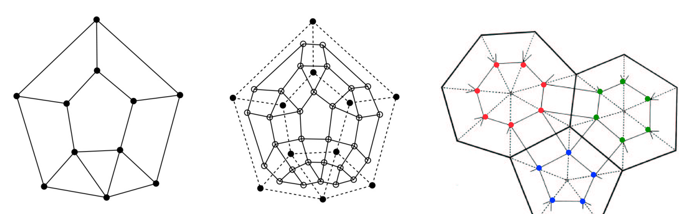
Chaikin's Curve Subdivision

Resulting curve is a uniform quadratic B-spline



Doo-Sabin Subdivision

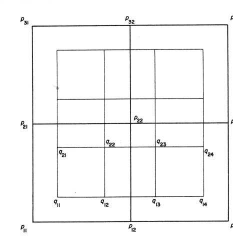
Introduces a new vertex for each face at the midpoint between an old vertex and face centroid



Original

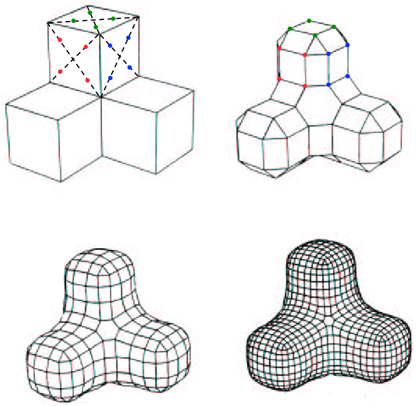
After splitting

Subdivision rules create a dual of the control net: a new face replaces each face, edge, and vertex of the control net



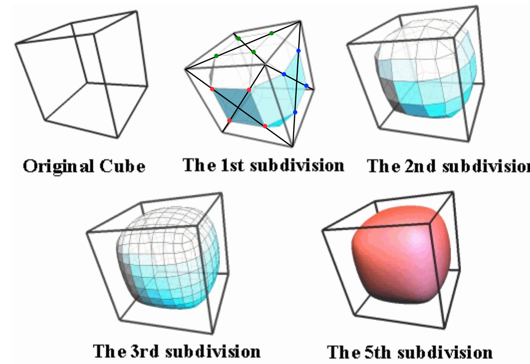
Doo-Sabin Subdivision

A generalization of quadratic curve subdivision (Chaikin's algorithm) to surfaces with arbitrary topology



Doo-Sabin Subdivision

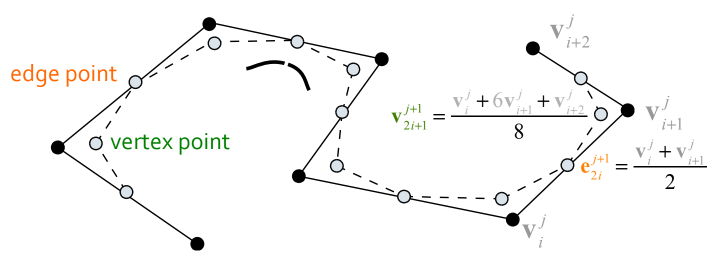
For regular quad meshes, resulting surface is a biquadratic B-spline surface



http://www.ke.ics.saitama-u.ac.jp/xuz/pic/doo-sabin.gif

Cubic B-spline Curve Subdivision

1. For each edge compute a new edge point using the averaging mask $(1\ 1)$: $\mathbf{e}_{2i}^{j+1} = \frac{1}{2}\mathbf{v}_i^j + \frac{1}{2}\mathbf{v}_{i+1}^j$
2. Compute new vertex points using the mask $(1\ 6\ 1)$: $\mathbf{v}_{2i+1}^{j+1} = \frac{1}{8}\mathbf{v}_i^j + \frac{6}{8}\mathbf{v}_{i+1}^j + \frac{1}{8}\mathbf{v}_{i+2}^j$
3. Connect the new edge- and vertex points; resulting curve is a uniform cubic B-spline



Cubic B-spline Curve Subdivision

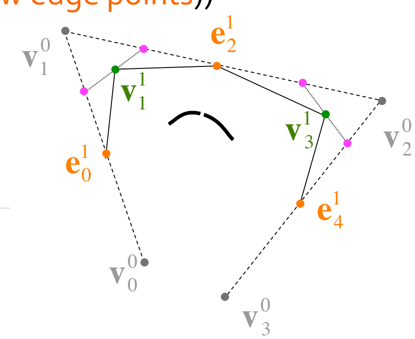
Vertex points are midpoint between
(midpoints between
(old vertices and new edge points))

$$\mathbf{v}_{2i+1}^{j+1} = \frac{1}{2}\left(\frac{1}{2}\mathbf{e}_{2i}^{j+1} + \frac{1}{2}\mathbf{v}_{i+1}^j\right) + \frac{1}{2}\left(\frac{1}{2}\mathbf{v}_{i+1}^j + \frac{1}{2}\mathbf{e}_{2i+1}^{j+1}\right)$$

$$= \frac{1}{4}\mathbf{e}_{2i}^{j+1} + \frac{1}{2}\mathbf{v}_{i+1}^j + \frac{1}{4}\mathbf{e}_{2i+1}^{j+1}$$

$$= \frac{1}{4}\left(\frac{1}{2}\mathbf{v}_i^j + \frac{1}{2}\mathbf{v}_{i+1}^j\right) + \frac{1}{2}\mathbf{v}_{i+1}^j + \frac{1}{4}\left(\frac{1}{2}\mathbf{v}_{i+1}^j + \frac{1}{2}\mathbf{v}_{i+2}^j\right)$$

$$= \frac{1}{8}\mathbf{v}_i^j + \frac{6}{8}\mathbf{v}_{i+1}^j + \frac{1}{8}\mathbf{v}_{i+2}^j$$

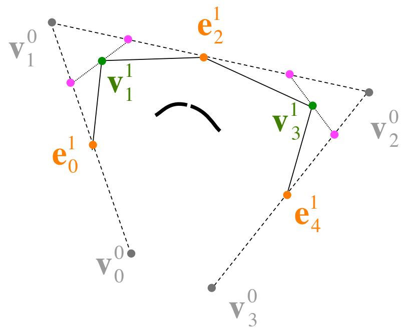


averaging mask $(1\ 6\ 1)$ of only old vertices, or also given as $(\frac{1}{4}\ \frac{1}{2}\ \frac{1}{4})$

[Hart]

Subdivision → New Control Points

Let: $\mathbf{P} = \begin{bmatrix} \mathbf{v}_0^0 \\ \mathbf{v}_1^0 \\ \mathbf{v}_2^0 \\ \mathbf{v}_3^0 \end{bmatrix}, \mathbf{P}^1 = \begin{bmatrix} \mathbf{e}_0^1 \\ \mathbf{v}_1^1 \\ \mathbf{e}_2^1 \\ \mathbf{v}_3^1 \\ \mathbf{e}_4^1 \end{bmatrix}$



$$\mathbf{H}_S = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{8} & \frac{3}{4} & \frac{1}{8} & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{8} & \frac{3}{4} & \frac{1}{8} \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

splitting matrix

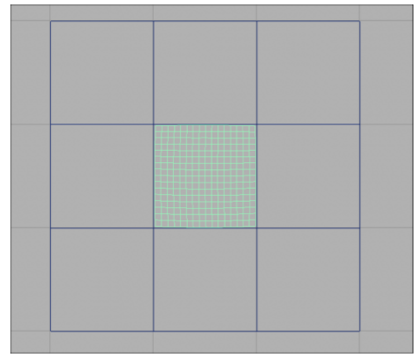
New control points: $\mathbf{P}^1 = \mathbf{H}_S \mathbf{P}$

[Hart]

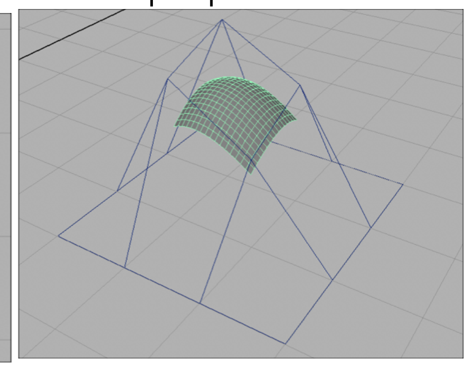
Uniform B-spline Patch Subdivision

$$\mathbf{s}(u,v) = \begin{bmatrix} 1 & u & u^2 & u^3 \end{bmatrix} \mathbf{B}_S \mathbf{P} \mathbf{B}_S^T \begin{bmatrix} 1 & v & v^2 & v^3 \end{bmatrix}^T$$

orthographic top-down view



3D perspective view



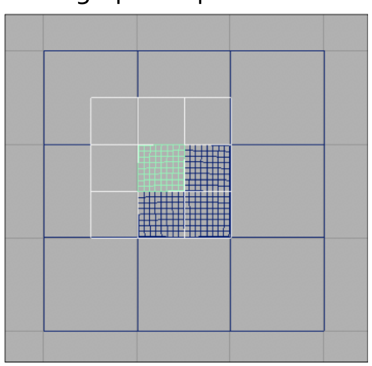
[O'Brien]

Subdivision Reparameterized

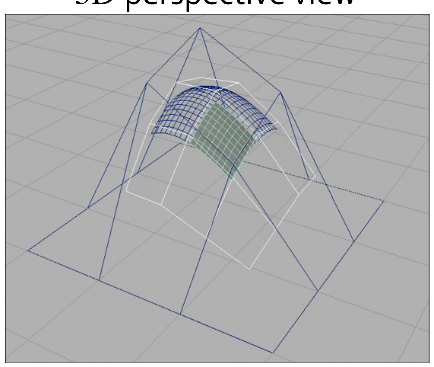
$$\mathbf{s}_{11}(u,v) = \begin{bmatrix} 1 & u & u^2 & u^3 \end{bmatrix} \mathbf{B}_S \mathbf{P}_{11} \mathbf{B}_S^T \begin{bmatrix} 1 & v & v^2 & v^3 \end{bmatrix}^T$$

$\mathbf{P}_{11} = \mathbf{H}_{S1} \mathbf{P} \mathbf{H}_{S1}^T$ new control points!

orthographic top-down view



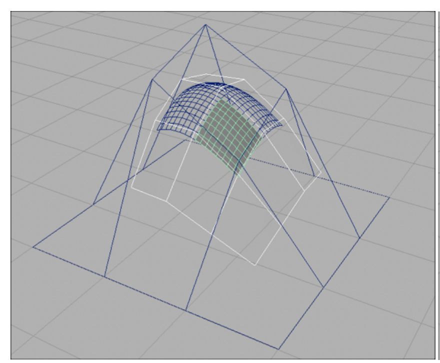
3D perspective view



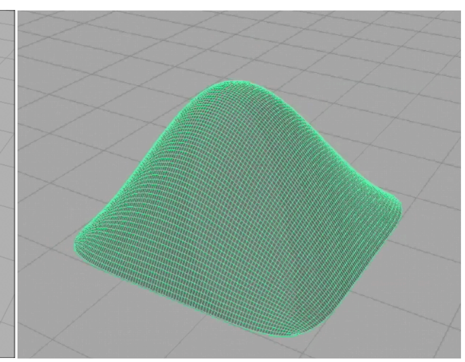
[O'Brien]

Limit of Subdivision

Control mesh approaches surface

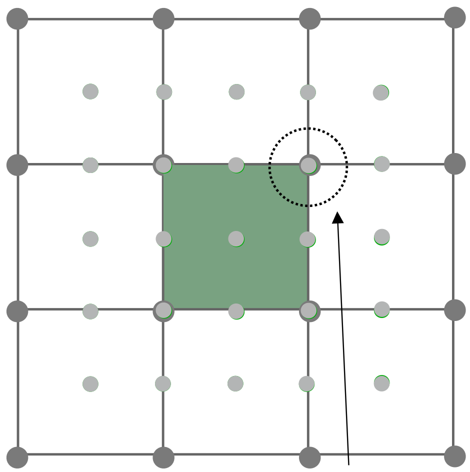


Limit of subdivision is the surface



[O'Brien]

New Control Points



in this parametric view these knot points are collocated; the 3D control points are not

$$\mathbf{P}_{11} = \mathbf{H}_{S1}\mathbf{P}\mathbf{H}_{S1}^T$$
$$\mathbf{P}_{12} = \mathbf{H}_{S1}\mathbf{P}\mathbf{H}_{S2}^T$$
$$\mathbf{P}_{22} = \mathbf{H}_{S2}\mathbf{P}\mathbf{H}_{S2}^T$$
$$\mathbf{P}_{21} = \mathbf{H}_{S2}\mathbf{P}\mathbf{H}_{S1}^T$$

$$\mathbf{H}_{S1} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ \frac{1}{8} & \frac{3}{4} & \frac{1}{8} & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{8} & \frac{3}{4} & \frac{1}{8} \end{bmatrix}$$

$$\mathbf{H}_{S2} = \begin{bmatrix} \frac{1}{8} & \frac{3}{4} & \frac{1}{8} & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{8} & \frac{3}{4} & \frac{1}{8} \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

[O'Brien]

Subdivision → New Control Points

Subdivision as a matrix $\mathbf{H}_S$ of weights w:

- $\mathbf{H}_S$ is very sparse
- limit surface: $\mathbf{P}^\infty = \lim\limits_{j\to\infty} (\mathbf{H}_S)^j\, \mathbf{P}$ ← power, not superscript
- allows for analysis: curvature, limit surface
- not for implementation!

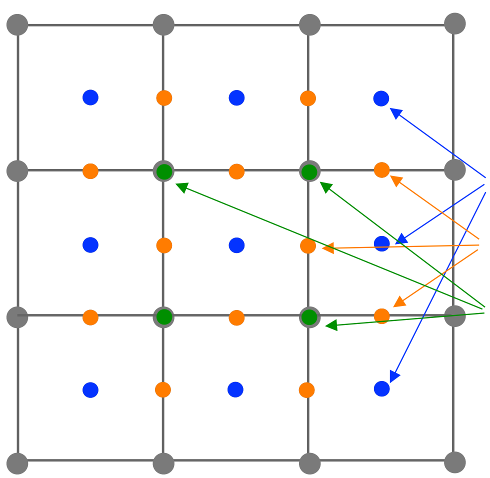
$$\mathbf{P}^{j+1} = \mathbf{H}_S \mathbf{P}^j$$

$$\begin{bmatrix} \mathbf{p}_0^{j+1} \\ \mathbf{p}_1^{j+1} \\ \mathbf{p}_2^{j+1} \\ \vdots \\ \mathbf{p}_n^{j+1} \end{bmatrix} = \begin{bmatrix} w_{00} & w_{01} & \cdots & 0 \\ w_{10} & w_{11} & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & \cdots & w_{nk} \end{bmatrix} \begin{bmatrix} \mathbf{p}_0^j \\ \mathbf{p}_1^j \\ \vdots \\ \mathbf{p}_k^j \end{bmatrix}$$

$\mathbf{H}_S$: 25×16 subdivision matrix
$\mathbf{P}^j$: vector of coarse control points, length 16
$\mathbf{P}^{j+1}$: vector of coarse control points, length 25

[O'Brien, Hart/Carr]

New Control Points



Instead, compute the new control points iteratively:

face points
edge points
moved vertex points

[O'Brien]

New Control Points

Face point:

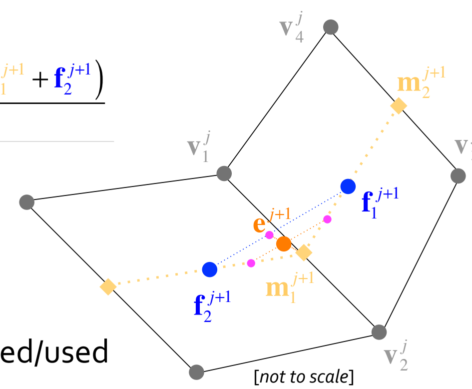
$$\mathbf{f}^{j+1} = \frac{\mathbf{m}_1^{j+1} + \mathbf{m}_2^{j+1}}{2} = \frac{\mathbf{v}_1^j + \mathbf{v}_2^j + \mathbf{v}_3^j + \mathbf{v}_4^j}{4}$$

Midpoint:

$$\mathbf{m}_1^{j+1} = \frac{\mathbf{v}_1^j + \mathbf{v}_2^j}{2}$$

Edge point:

$$\mathbf{e}^{j+1} = \frac{\frac{1}{2}\left(\mathbf{f}_1^{j+1} + \mathbf{m}_1^{j+1}\right) + \frac{1}{2}\left(\mathbf{m}_1^{j+1} + \mathbf{f}_2^{j+1}\right)}{2}$$
$$= \frac{\frac{1}{2}\left(\mathbf{f}_1^{j+1} + \mathbf{f}_2^{j+1}\right) + \mathbf{m}_1^{j+1}}{2}$$
$$= \frac{\mathbf{v}_1^j + \mathbf{v}_2^j + \mathbf{f}_1^{j+1} + \mathbf{f}_2^{j+1}}{4}$$



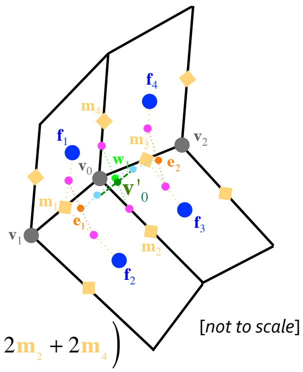
Note $\mathbf{f}^0$, $\mathbf{m}^0$, $\mathbf{e}^0$ not defined/used

[O'Brien, Hart]

Vertex Points

$$\mathbf{w}_0 = \tfrac{1}{2}\left(\tfrac{\mathbf{m}_2+\mathbf{v}_0}{2} + \tfrac{\mathbf{v}_0+\mathbf{m}_4}{2}\right) = \tfrac{1}{4}\mathbf{m}_2 + \tfrac{1}{2}\mathbf{v}_0 + \tfrac{1}{4}\mathbf{m}_4$$

$$\begin{aligned}
\mathbf{v}'_0 &= \tfrac{1}{2}\left(\tfrac{1}{2}(\mathbf{e}_1 + \mathbf{w}_0) + \tfrac{1}{2}(\mathbf{w}_0 + \mathbf{e}_2)\right) \\
&= \tfrac{1}{4}\left(\tfrac{\mathbf{f}_1+\mathbf{f}_2+\mathbf{v}_0+\mathbf{v}_1}{4} + \tfrac{\mathbf{f}_3+\mathbf{f}_4+\mathbf{v}_0+\mathbf{v}_2}{4} + 2\mathbf{w}_1\right) \\
&= \tfrac{1}{4}\left(\tfrac{\mathbf{f}_1+\mathbf{f}_2+\mathbf{f}_3+\mathbf{f}_4}{4} + \tfrac{2\mathbf{v}_0+\mathbf{v}_1+\mathbf{v}_2}{4} + \tfrac{2}{4}\left(2\mathbf{v}_0+\mathbf{m}_2+\mathbf{m}_4\right)\right) \\
&= \tfrac{1}{16}\left(\mathbf{f}_1 + \mathbf{f}_2 + \mathbf{f}_3 + \mathbf{f}_4 + 2\left(\tfrac{\mathbf{v}_0+\mathbf{v}_1}{2} + \tfrac{\mathbf{v}_0+\mathbf{v}_2}{2}\right) + 4\mathbf{v}_0 + 2\mathbf{m}_2 + 2\mathbf{m}_4\right) \\
&= \tfrac{1}{16}\left(\mathbf{f}_1 + \mathbf{f}_2 + \mathbf{f}_3 + \mathbf{f}_4 + 2\mathbf{m}_1 + 2\mathbf{m}_3 + 2\mathbf{m}_2 + 2\mathbf{m}_4 + 4\mathbf{v}_0\right)
\end{aligned}$$



[not to scale]

$$\begin{aligned}
\mathbf{v}^{j+1} &= \frac{\mathbf{v}^j + \frac{1}{n}(\mathbf{f}_1^{j+1} + \mathbf{f}_2^{j+1} + \mathbf{f}_3^{j+1} + \mathbf{f}_4^{j+1}) + \frac{2}{n}(\mathbf{m}_1^{j+1} + \mathbf{m}_2^{j+1} + \mathbf{m}_3^{j+1} + \mathbf{m}_4^{j+1})}{n} \\
&= \frac{2\mathbf{v}^j + \frac{1}{n}(\mathbf{f}_1^{j+1} + \mathbf{f}_2^{j+1} + \mathbf{f}_3^{j+1} + \mathbf{f}_4^{j+1}) + \frac{1}{n}(\mathbf{v}_1^j + \mathbf{v}_2^j + \mathbf{v}_3^j + \mathbf{v}_4^j)}{n}
\end{aligned}$$

[O'Brien, Hart]

Catmull-Clark Subdivision Rules

Subdivision level $j+1$:

- face point: for each face, add a new vertex at its centroid that is the average of the surrounding m vertices:

$$\mathbf{f}^{j+1} = \frac{1}{m}\sum_{i=1}^{m} \mathbf{v}_i^j$$

- edge point: for each edge, add a new edge point which is the average of the 2 vertices and the 2 face points adjacent to the edge:

$$\mathbf{e}^{j+1} = \frac{\mathbf{v}_1^j + \mathbf{v}_2^j + \mathbf{f}_1^{j+1} + \mathbf{f}_2^{j+1}}{4}$$

- moved vertex point: vertex moved to the weighted average between the original position, the n midpoints (not edge) points and the n face points surrounding the vertex (n: vertex valence, $=4$):

$$\mathbf{v}^{j+1} = \frac{n-2}{n}\mathbf{v}^j + \frac{1}{n^2}\sum_{i=1}^{n} \mathbf{v}_i^j + \frac{1}{n^2}\sum_{i=1}^{n} \mathbf{f}_i^{j+1}$$

Catmull-Clark Subdivision (1978)

Subdivision level $j+1$:

- face point, m # of face vertices

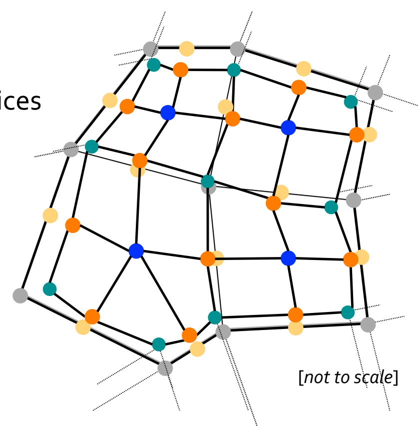
$$\mathbf{f}^{j+1} = \frac{1}{m}\sum_{i=1}^{m} \mathbf{v}_i^j$$

- edge point

$$\mathbf{e}^{j+1} = \frac{\mathbf{v}_1^j + \mathbf{v}_2^j + \mathbf{f}_1^{j+1} + \mathbf{f}_2^{j+1}}{4}$$

- moved vertex point

$$\begin{aligned}
\mathbf{v}^{j+1} &= \frac{(n-3)}{n}\mathbf{v}^j + \frac{2}{n}\overline{\mathbf{m}}^{j+1} + \frac{1}{n}\overline{\mathbf{f}}^{j+1} \\
&= \frac{(n-2)}{n}\mathbf{v}^j + \frac{1}{n}\overline{\mathbf{v}}^j + \frac{1}{n}\overline{\mathbf{f}}^{j+1},\ n = 4
\end{aligned}$$



[not to scale]

$\overline{\mathbf{m}}$: average of adjacent vertices

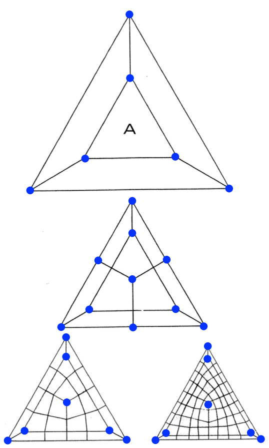
$\overline{\mathbf{f}}$: average of adjacent face points

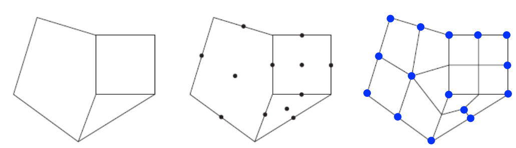
n: vertex valence

Catmull-Clark Subdivision

Works with arbitrary polygonal mesh: after 1^{st} round of subdivision,

- all faces are quads
- the number of extraordinary points remains constant
  - distances between them remain constant: as faces become smaller, there are more faces between them



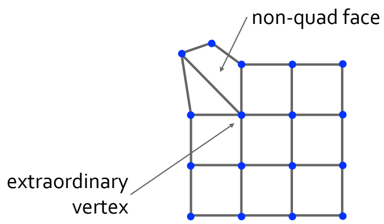


[CatmullClark]

Catmull-Clark Subdivision

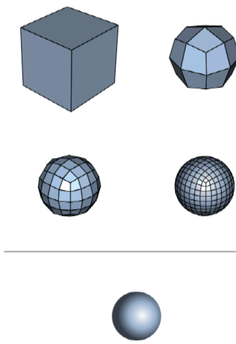
Subdivision rules are chosen to improve continuity

Smoothness of limit surface:
- C^2 almost everywhere
- C^1 at extraordinary vertices
- strictly generalize uniform tensor-product bicubic B-splines: works with existing tools for tensor-product B-splines
- generalization of cubic B-splines subdivision to irregular patch:



[CatmullClark]

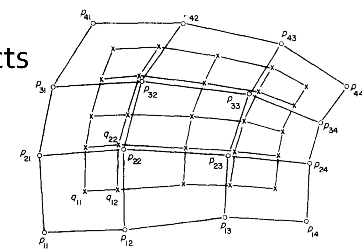
Catmull-Clark Subdivision



Relationship to control mesh:
- does not interpolate control mesh
- within convex hull

Subdivision rules creates a primal (not dual) of the control net

Quads are often better than triangles to represent real objects that are often symmetric, e.g., tube-like surfaces: arms, legs, fingers

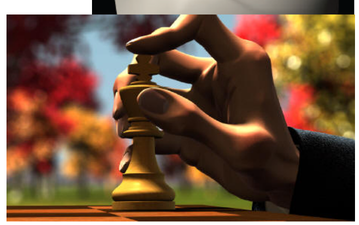


[CatmullClark,DeRoseKassTruong]

Catmull-Clark Subdivision

Any mesh can be subdivided
- cut holes, create unusual topology, stitch pieces together
- no matter how complicated the mesh, it will lead to a smooth surface!

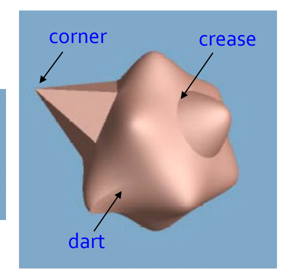
Extensions: localized subdivision rules
- creases: NURBS requires use of trim curves; for subdivision, just modify the subdivision mask
- edge preservation: hard edges
- adaptive subdivision

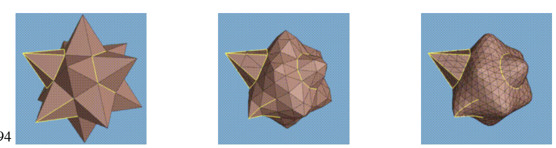


Curless

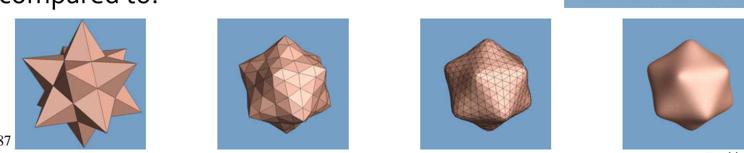
Edge Preservation

To get sharpness and creases, define new subdivision rules for "creased" edges and vertices
- crease: a smooth curve with continuity G^0 on the surface (2 sharp edges)
- corner: a vertex where ≥ 3 sharp edges meet
- dart: a vertex where a crease ends and smoothly blends into the surface (1 sharp edge)



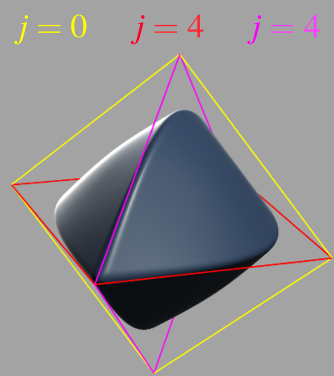
Hoppe94 

compared to:

Loop87 

Funkhouser, Zhang

Sharp Edges



Idea: edges with a sharpness j are subdivided using sharp rules for the first j iterations, and then smoothly, as usual, to the limit surface

- tag edges as sharp or not sharp: newly created edges are assigned a sharpness of j–1
- edges with $j = 0$ are not sharp
- edges with $j > 0$ are sharp

During subdivision, if an edge is not-sharp use normal smooth subdivision rules; if an edge is sharp, use sharp subdivision rules

Approximating subdivision algorithm can be made interpolating

Hart/Carr

Sharp Rules

Compare non-sharp rules:

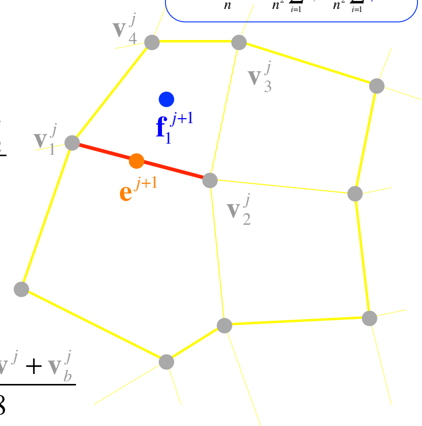
$$\mathbf{f}^{j+1} = \frac{1}{m}\sum_{i=1}^{m}\mathbf{v}_i^j,\ \mathbf{e}^{j+1} = \frac{\mathbf{v}_1^j + \mathbf{v}_2^j + \mathbf{f}_1^{j+1} + \mathbf{f}_2^{j+1}}{4},$$

$$\mathbf{v}^{j+1} = \frac{n-2}{n}\mathbf{v}^j + \frac{1}{n^2}\sum_{i=1}^{n}\mathbf{v}_i^j + \frac{1}{n^2}\sum_{i=1}^{n}\mathbf{f}_i^{j+1}$$

Subdivision level $j+1$:

- face point unchanged
- edge point $\mathbf{e}^{j+1} = \dfrac{\mathbf{v}_1^j + \mathbf{v}_2^j}{2}$
- moved vertex

	# of adjacent sharp edges	
corner:	> 2	$\mathbf{v}^{j+1} = \mathbf{v}^j$
crease: ($(\mathbf{v}, \mathbf{v}_a)$ and $(\mathbf{v}, \mathbf{v}_b)$)	2	$\mathbf{v}^{j+1} = \dfrac{\mathbf{v}_a^j + 6\mathbf{v}^j + \mathbf{v}_b^j}{8}$
dart:	1	unchanged

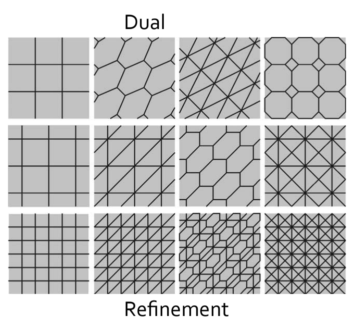


Hart

Subdivision Surfaces

Scheme classification by:

- interpolating or approximating
- mesh type: quads, triangles, hex, …, combination
- subdivision by face split (primal) or vertex split (dual)
- B-spline order of limit surface
- smoothness



Algorithms:

Doo-Sabin	’78	approximate	C^1	quad	dual
Catmull-Clark	’78	approximate	C^2	quad	primal
Loop	’87	approximate	C^2	triad	primal
DLG midpoint	’87	approximate	C^2	quad	dual
Butterfly (mod)	’90, ’96	interpolate	C^1	triad	primal
Kobbelt	’96	interpolate	C^1	quad	primal
$\sqrt{3}$	’00	approximate	C^2	triad	dual

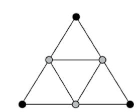
[Narasimhan, Zorin&Schroeder, Bischoff&Kobbelt]

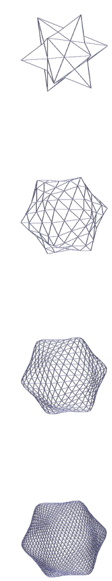
Loop Subdivision

Named after Charles Loop

Start with a triangular mesh

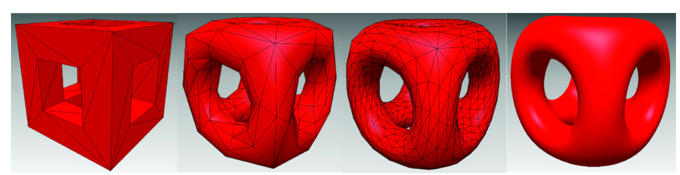
Resulting surface is a generalization of three-direction quartic box-spline





Subdivision rules:

- refinement: break edges at midpoint, for both faces
- smoothing: different averaging masks for new (“odd”) and old (“even”) vertices

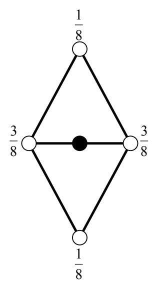


Loop Subdivision Masks

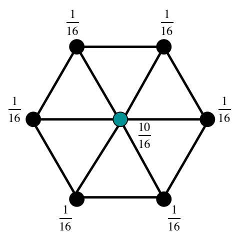
New (“odd”) vertices are placed based on weighted average of old vertices on both faces

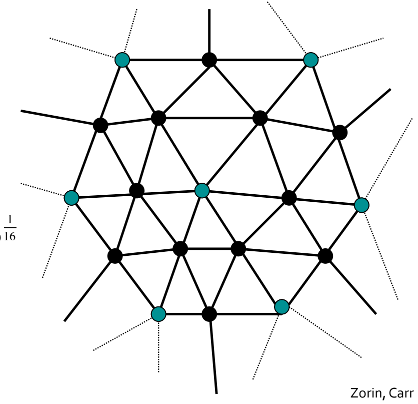
Old (“even”) vertices are moved based on surrounding neighbors

Odd mask:



Even mask:



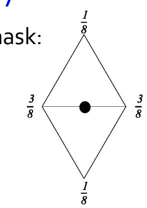


Zorin, Carr

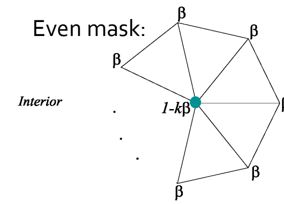
Loop Subdivision Masks

For ordinary vertices inside mesh:

Odd mask:

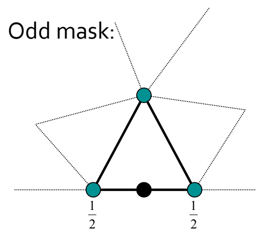


Even mask:

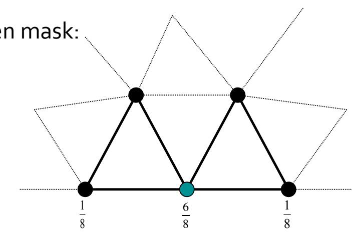


For extraordinary vertices and boundaries:

Odd mask:



Even mask:



Loop Subdivision

How to choose β ?

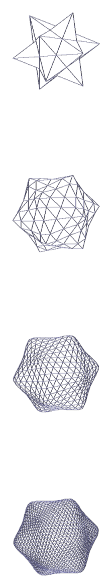
- must ensure tangent plane or normal continuity (G^1) of limit surface
- involves calculating eigenvalues of matrices

Original Loop:

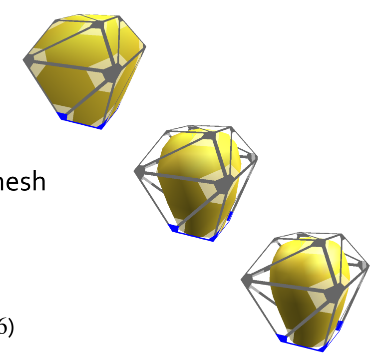
$$\beta = \frac{1}{8n}\left(40 - \left(3 + 2\cos\left(\frac{2\pi}{n}\right)\right)^2\right)$$

Warren:

$$\beta = \begin{cases} \dfrac{3}{8n} & n > 3 \\ \dfrac{3}{16} & n = 3 \end{cases}$$

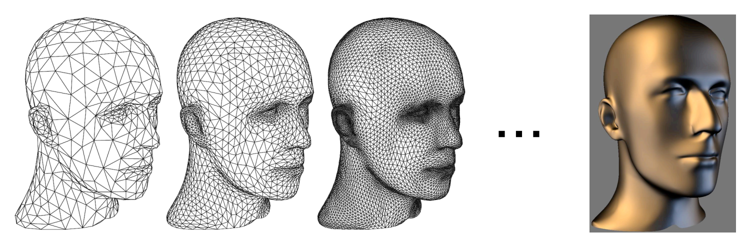


Loop Subdivision



Approximating subdivision

- does not interpolate the control mesh
  - within convex hull
- in the limit a smooth surface
  - C^2 almost everywhere
  - C^1 at extraordinary vertices (valence $\neq 6$)

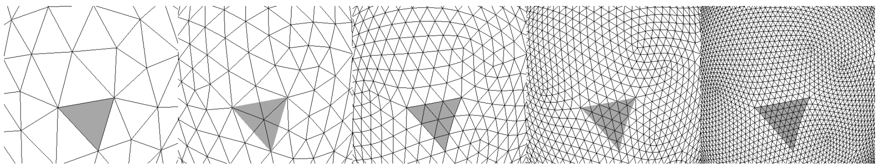


√3 Subdivision Scheme

Starts with a triangle mesh

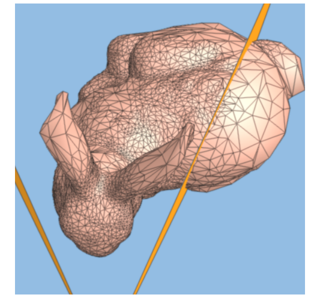
Number of faces triples per iteration
- slower growth rate

Gives finer control over polygon count
- better for adaptive subdivision



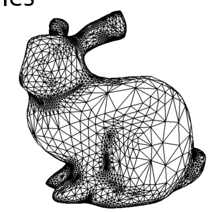
Funkhouser

Adaptive Subdivision

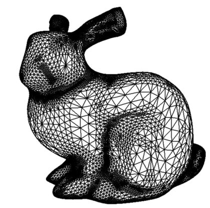


Not all regions of a model must be subdivided to the same resolution
- may be due to limited triangle budget

Stop subdivision at different levels across the surface, depending on:
- local surface curvature
- projected screen size of triangles
- view dependence
  - distance from viewer
  - silhouettes
  - in view frustum
- careful to avoid "cracks"!



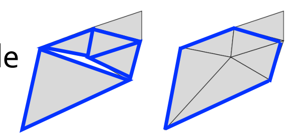
10,072 triangles



228,654 triangles

Funkhouser, Carr, Hoppe

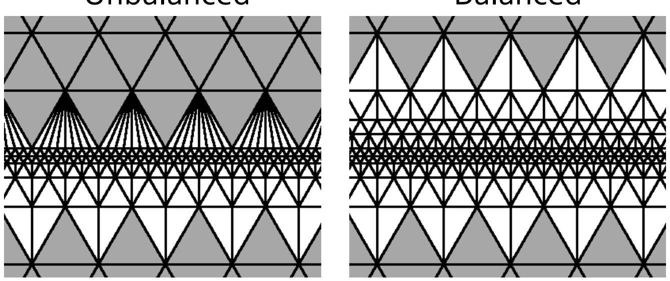
Balanced Subdivision



Crack avoidance: replace incompatible coarse triangles with triangle fan

Balanced subdivision: neighboring subdivision levels must not differ by more than one

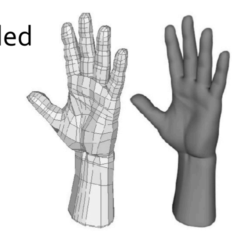
Unbalanced Balanced



Funkhouser, Kobbelt

Subdivision Surfaces

Characteristics and advantages:
- one surface, not a patchwork (collection of patches)
  - no seams, can deform/animate geometry without cracks
  - guaranteed continuity (smooth at boundaries)
- arbitrary control mesh, not limited to quads
  - can make surfaces with arbitrary topology or connectivity
- simple, only need subdivision rule
- adaptive subdivision: areas of surface with higher curvature can be more finely subdivided
  - multiresolution: LoD, scalable
- local support: only look at nearby vertices
  - numerical stability, well-behaved meshes
- affine invariance
- efficient rendering



Funkhouser, Durand, Schulze

Subdivision Surfaces

Disadvantages:

- non-intuitive specification: it's a procedural definition
- non-parametric, not implicit: hard to parameterize
  - no global (u, v) parameters
- hard to compute intersections
- tricky at special vertices (those with more or less than 6 neighbors in a triangular mesh)

Funkhouser, Durand

Parametric vs. Subdivision Surfaces

Parametric B-splines

- smooth
- must be tessellated
  - sampling issues
  - triangle size issue
  - cracking concern
- have uniform resolution
  - detail must be global
- require regular grid
- complex topology hard
  - no corners, holes
  - trimming hard
  - stitching hard
  - creases and sharp edges hard
- (u, v) parameterization
  - but not controllable

Subdivision

- limit surfaces are smooth
- gives meshes
  - subdivide as needed
  - always connected
  - get as many poly as you need
- put details where needed
  - detail is multiresolution
- works with arbitrary mesh
- any topology can be handled
  - easy to make corners, holes
  - trimming easy
  - stitching easy
  - creases and sharp edges easy
- (u, v) parameterization
  - by subdivision of points
  - controllable

Gleicher