Subdivision Surfaces

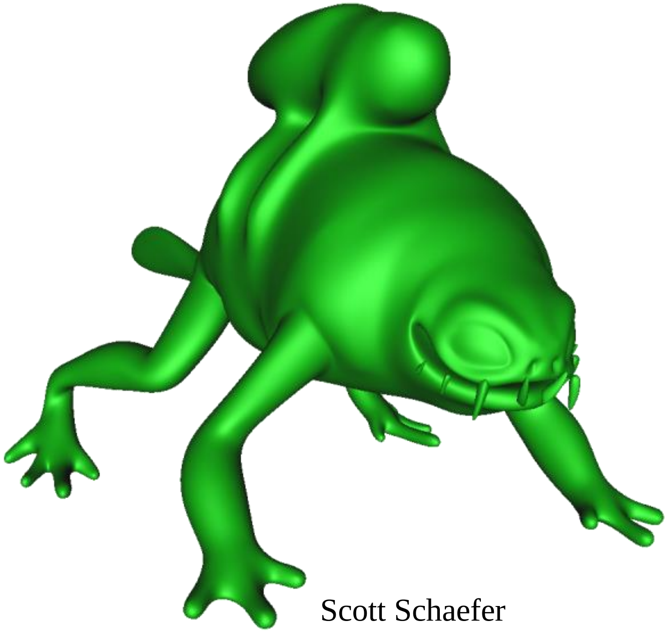
COS 426, Spring 2014
Princeton University

3D Object Representations

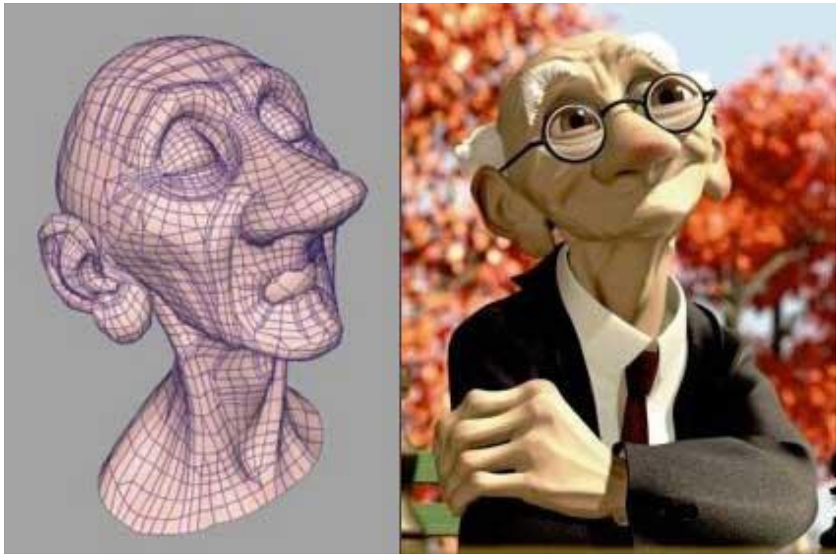
- Raw data
  - Range image
  - Point cloud
- Surfaces
  - Polygonal mesh
  - ➢ Subdivision
  - Parametric
  - Implicit
- Solids
  - Voxels
  - BSP tree
  - CSG
  - Sweep
- High-level structures
  - Scene graph
  - Application specific

Subdivision Surfaces

- Used in movie and game industries
- Supported by most 3D modeling software



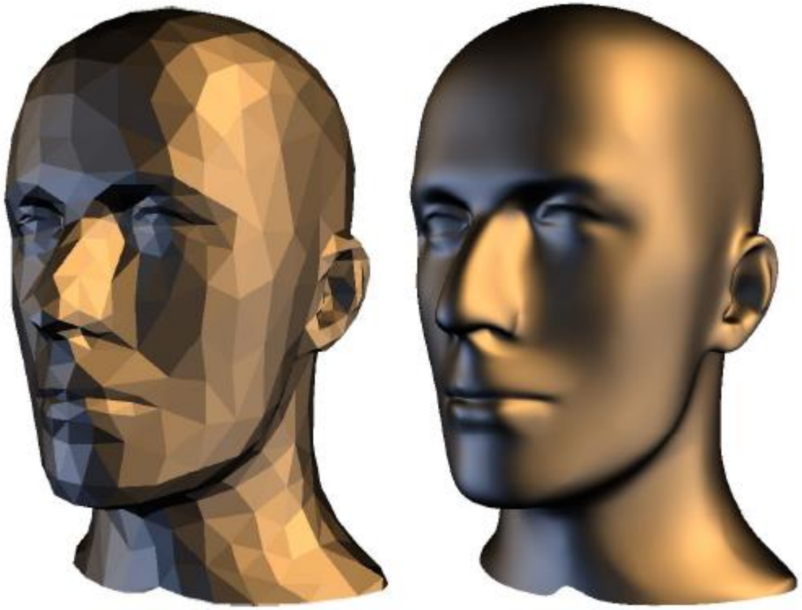
Scott Schaefer



Geri's Game © Pixar Animation Studios

Subdivision Surfaces

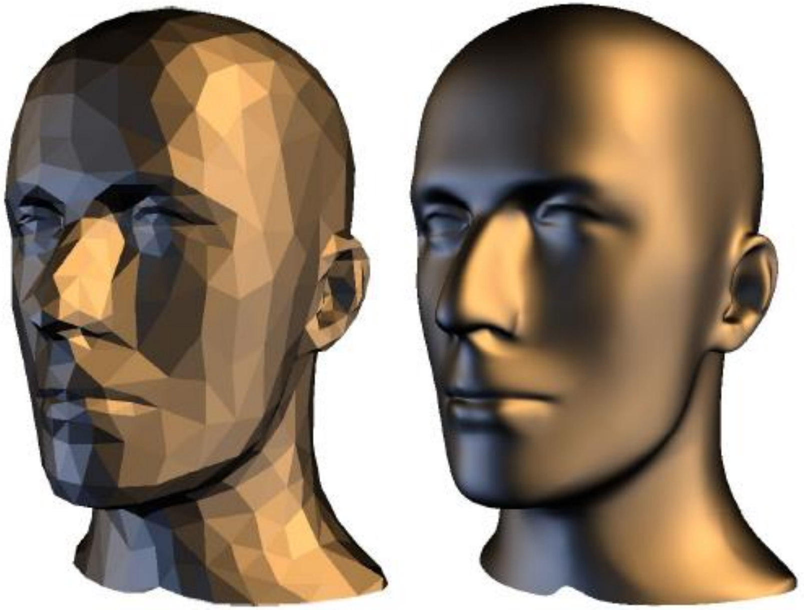
- What makes a good surface representation?
  - Accurate
  - Concise
  - Intuitive specification
  - Local support
  - Affine invariant
  - Arbitrary topology
  - Guaranteed continuity
  - Natural parameterization
  - Efficient display
  - Efficient intersections



Reif & Schroeder 2000

Subdivision Surfaces

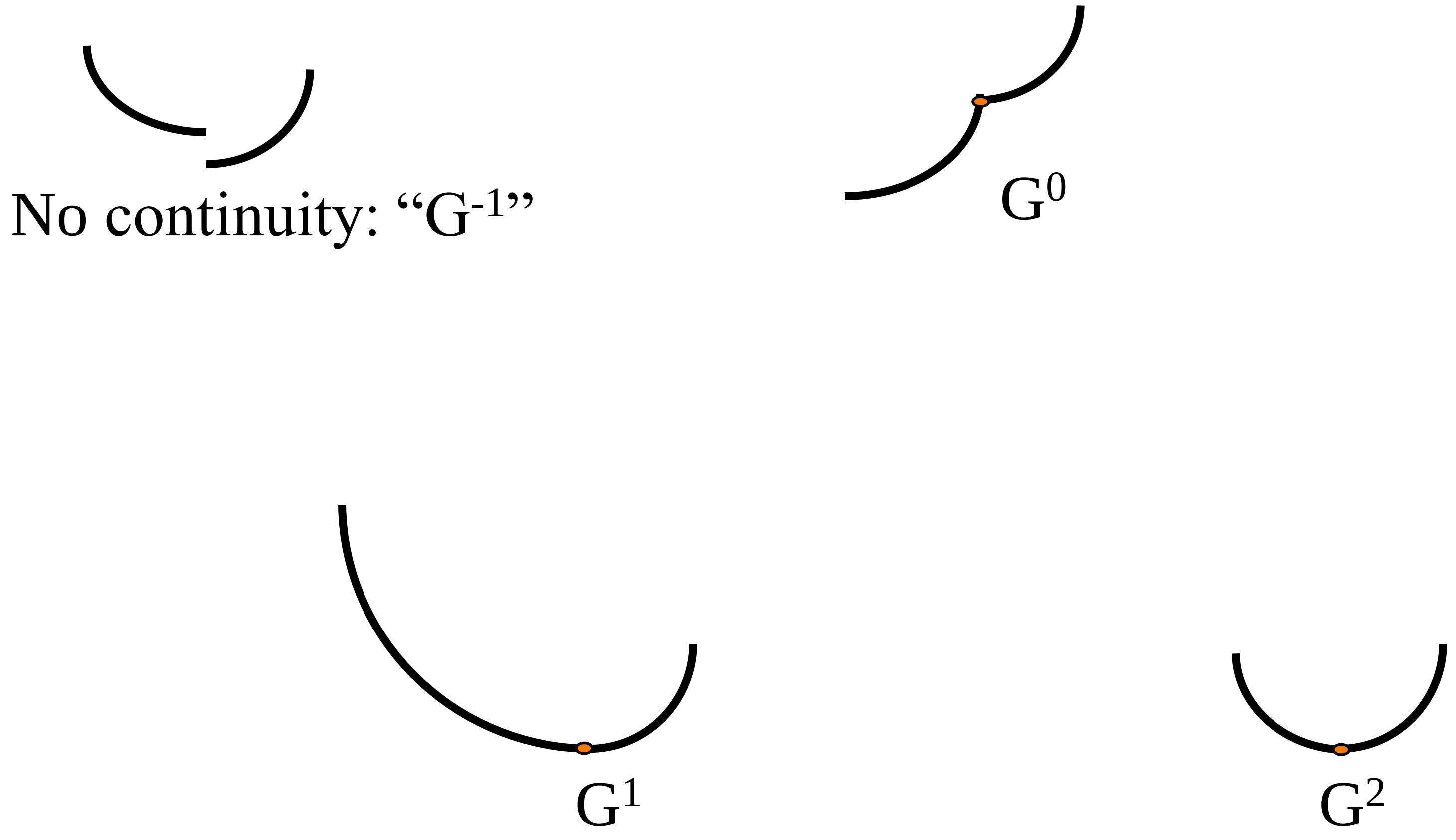
- What makes a good surface representation?
  - Accurate
  - Concise
  - Intuitive specification
  - Local support
  - Affine invariant
  - Arbitrary topology
  - ➢ Guaranteed continuity
  - Natural parameterization
  - Efficient display
  - Efficient intersections



Reif & Schroeder 2000

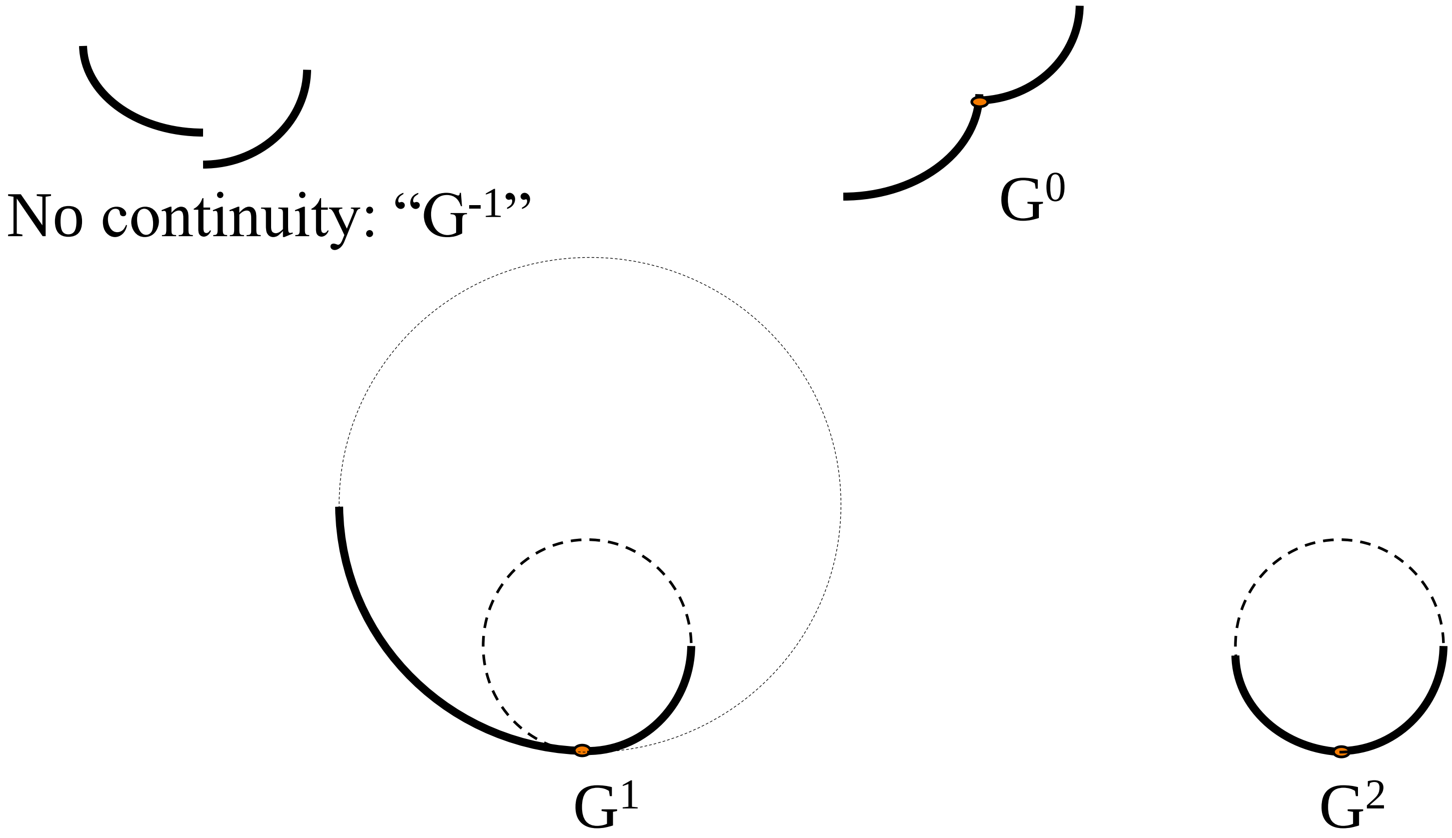
Continuity

- A curve / surface with G^k continuity has a continuous k-th derivative



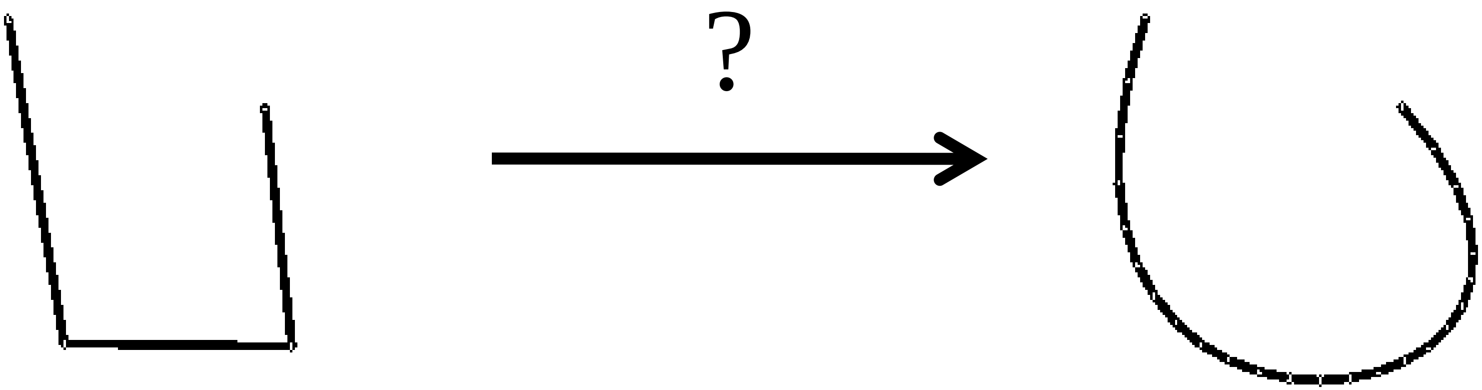
Continuity

- A curve / surface with G^k continuity has a continuous k-th derivative



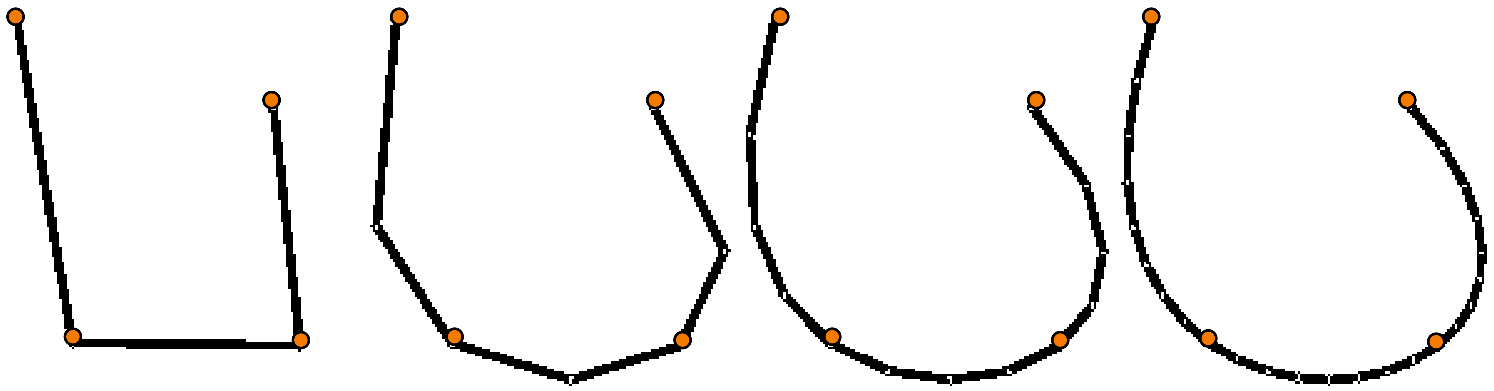
Subdivision

- How do you make a curve with guaranteed continuity?



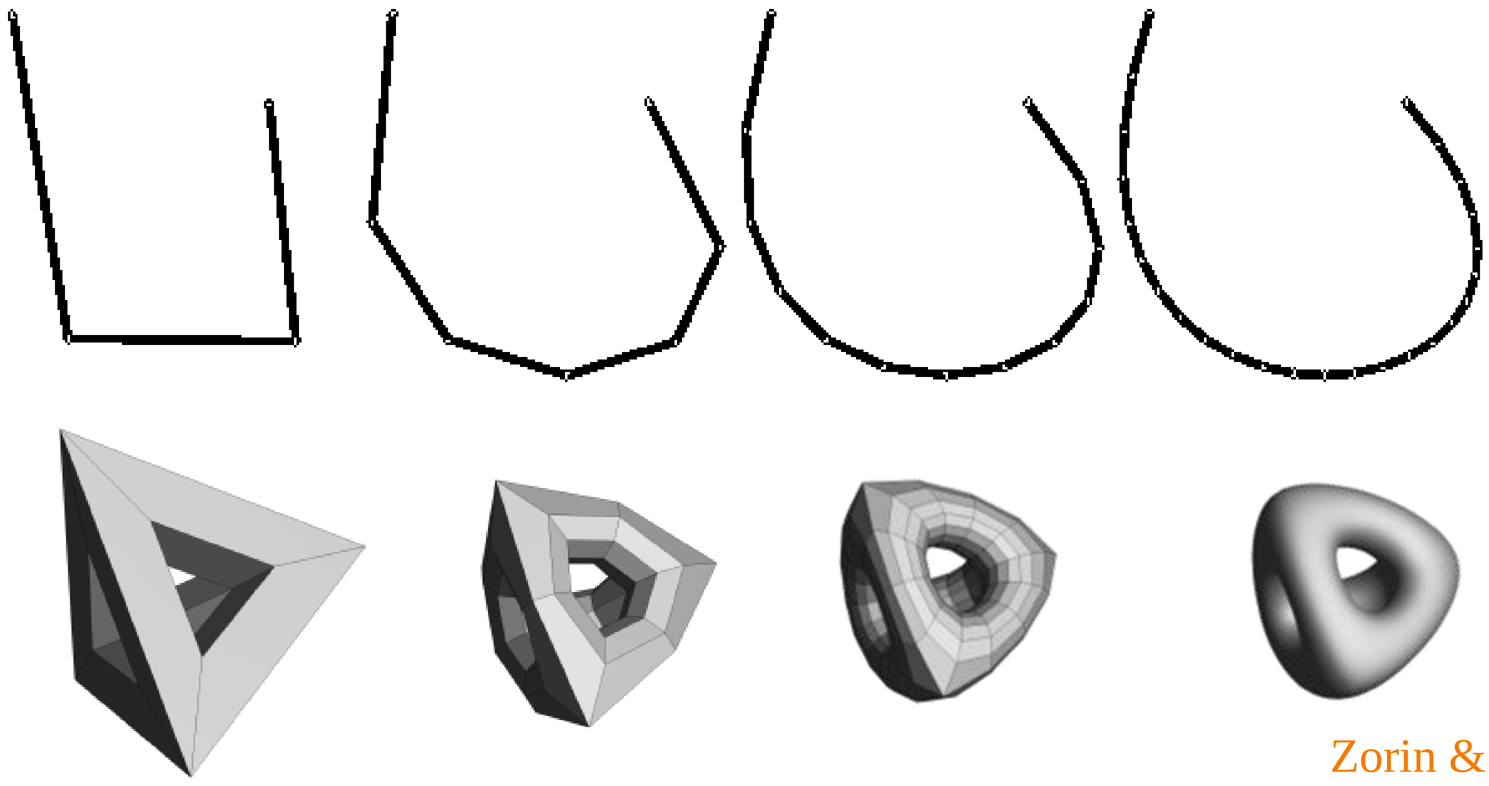
Subdivision

- How do you make a curve with guaranteed continuity? …



Subdivision

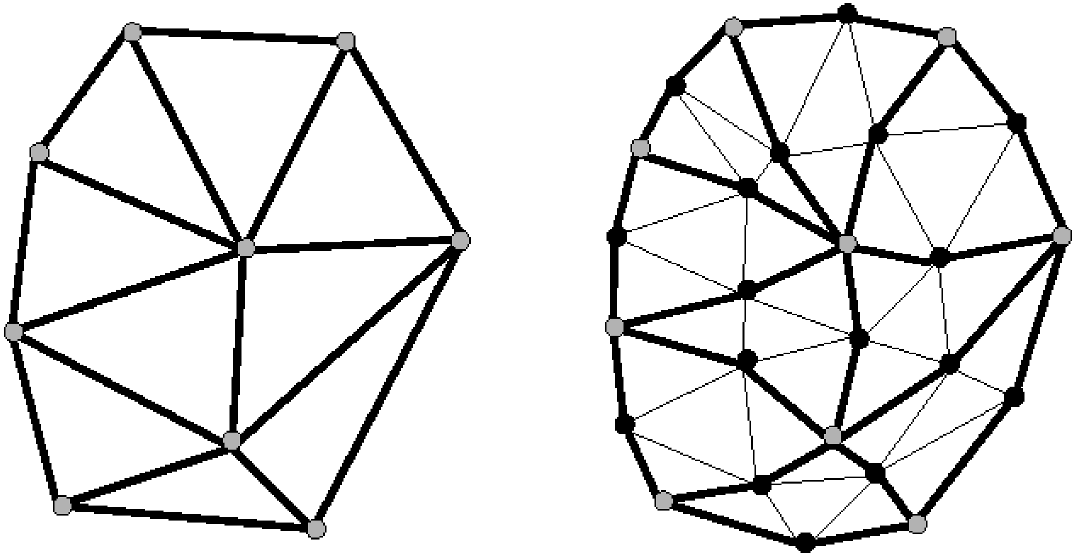
- How do you make a surface with guaranteed continuity?



Zorin & Schroeder
SIGGRAPH 99
Course Notes

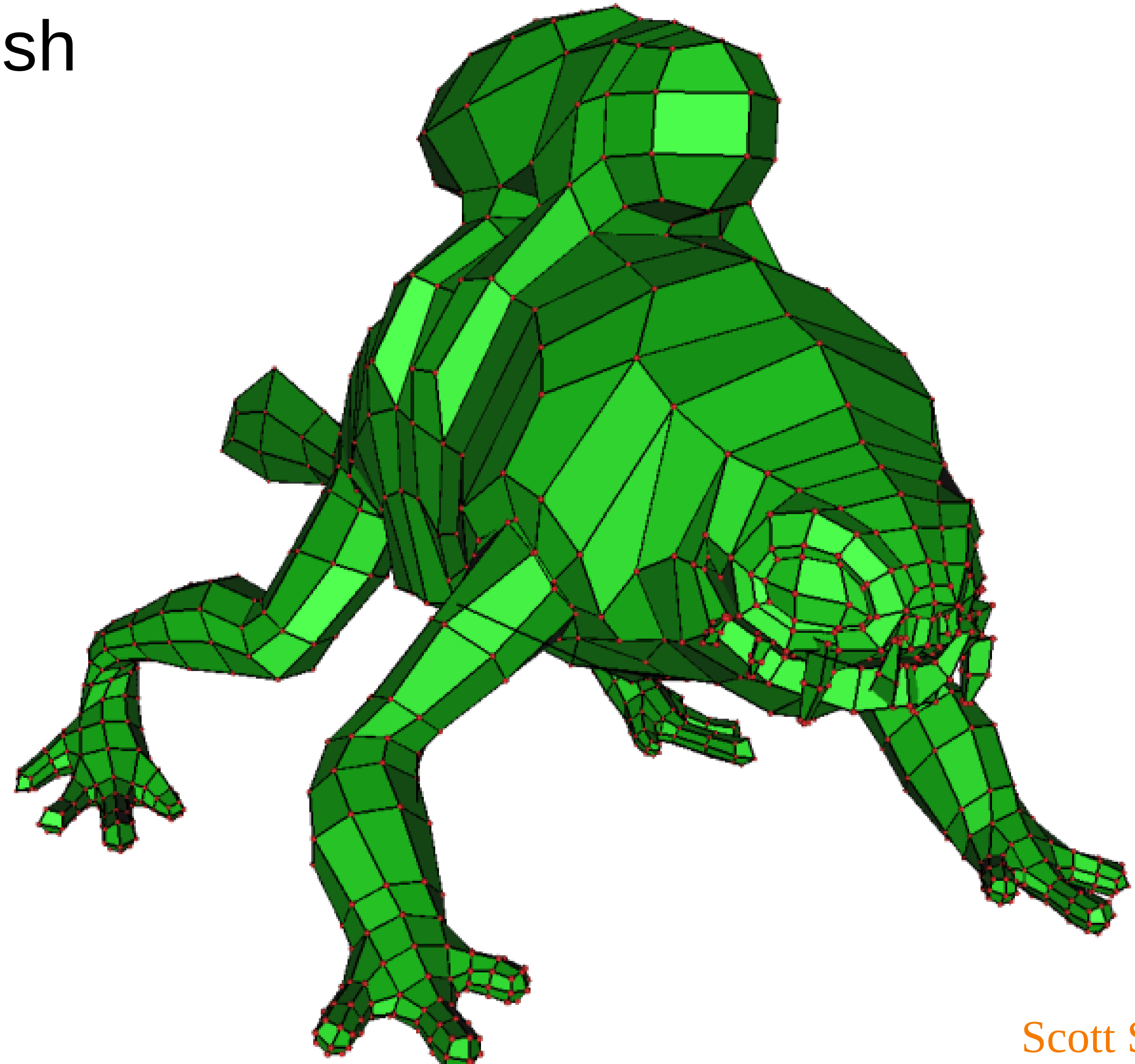
Subdivision Surfaces

- Repeated application of
  - Topology refinement (splitting faces)
  - Geometry refinement (weighted averaging)



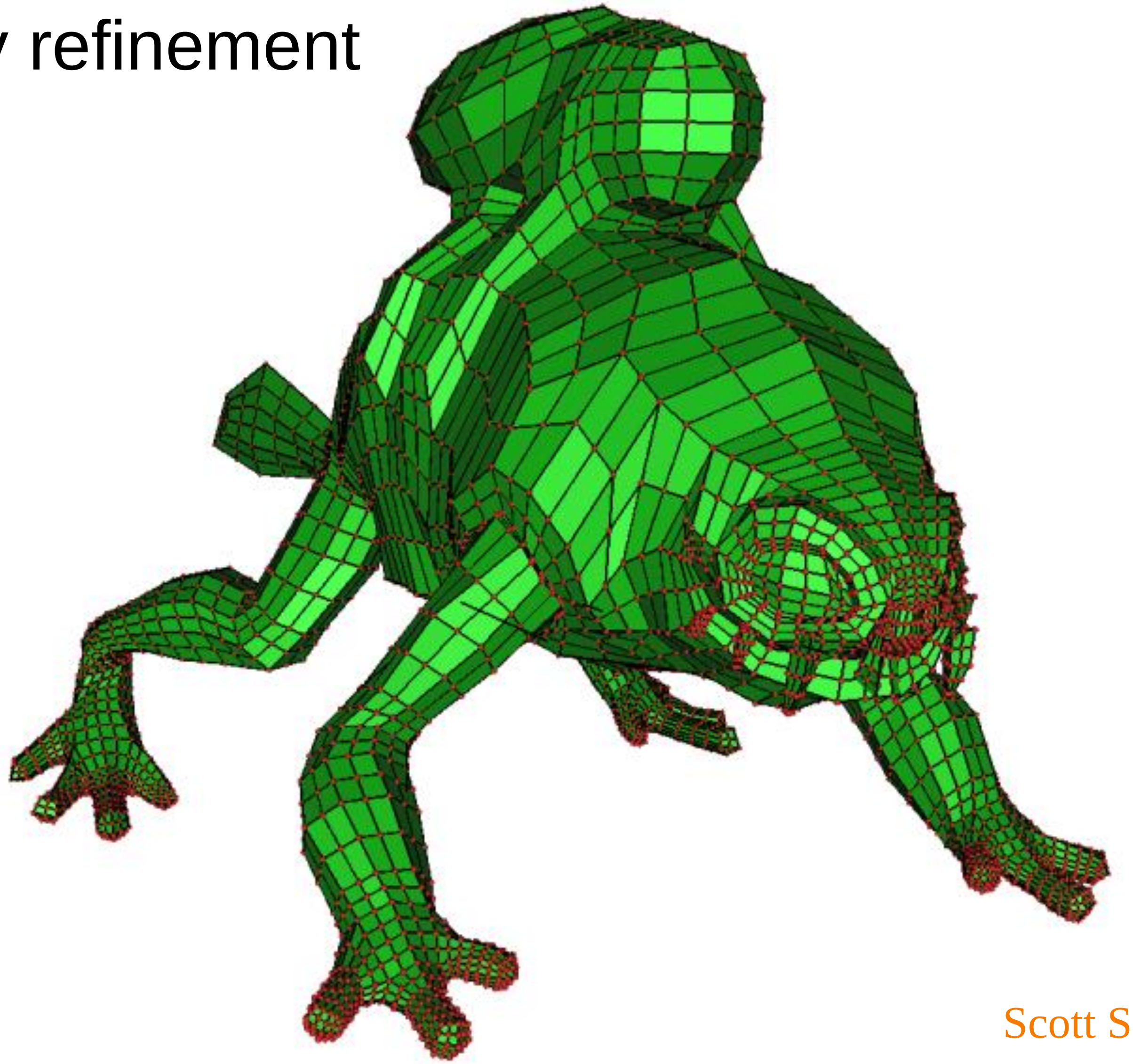
Subdivision Surfaces – Examples

- Base mesh



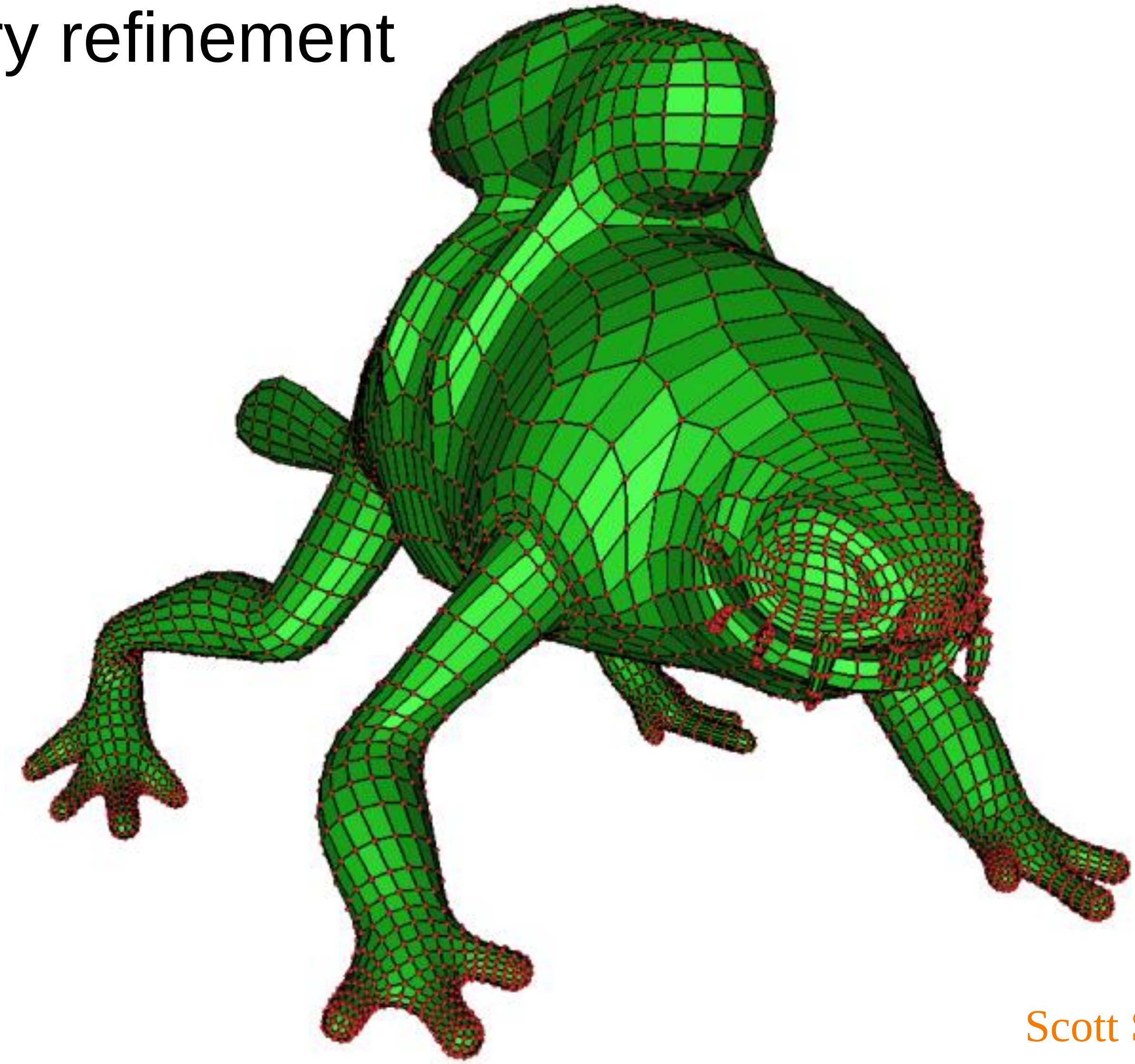
Subdivision Surfaces – Examples

- Topology refinement



Subdivision Surfaces – Examples

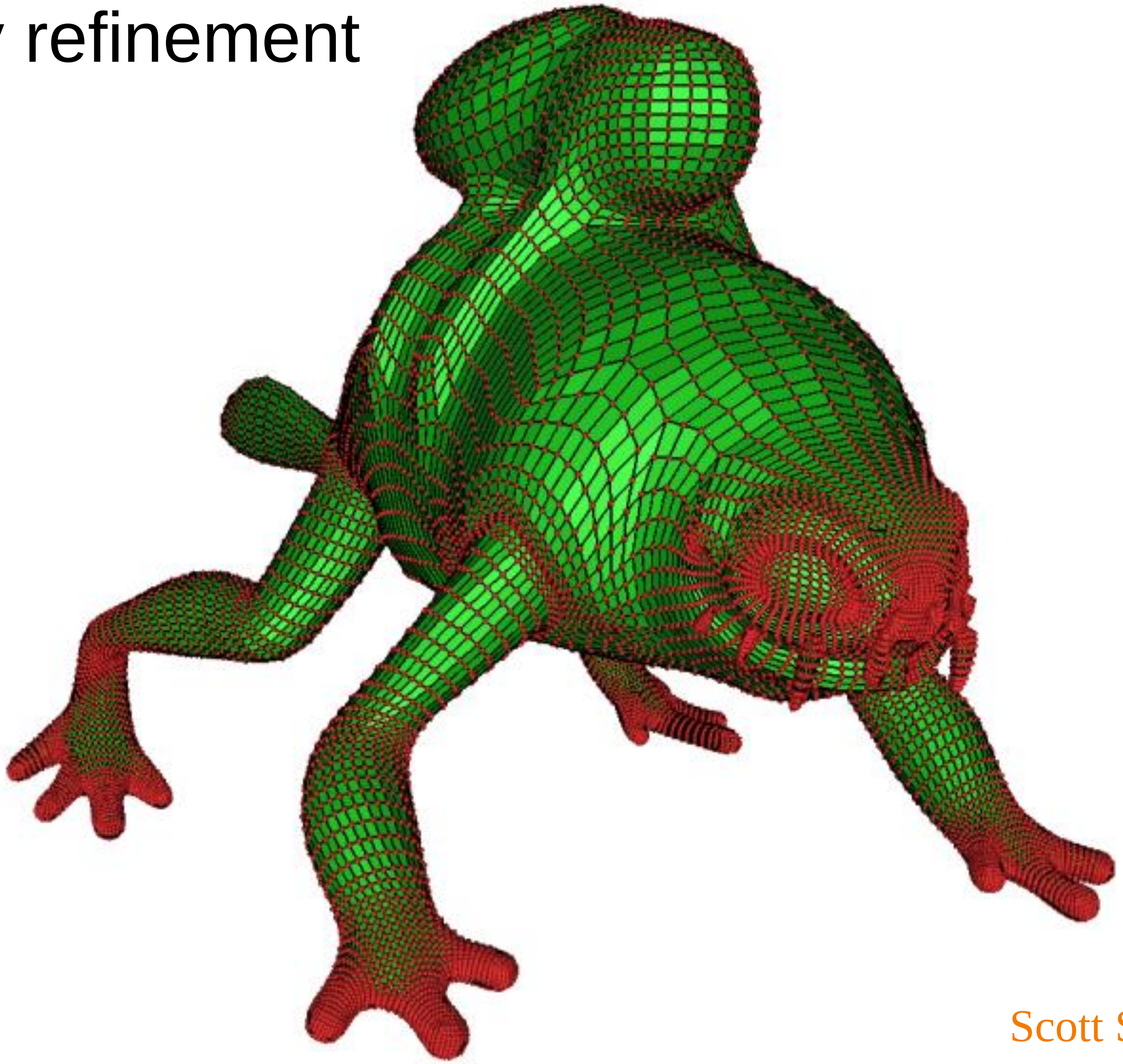
- Geometry refinement



Scott Schaefer

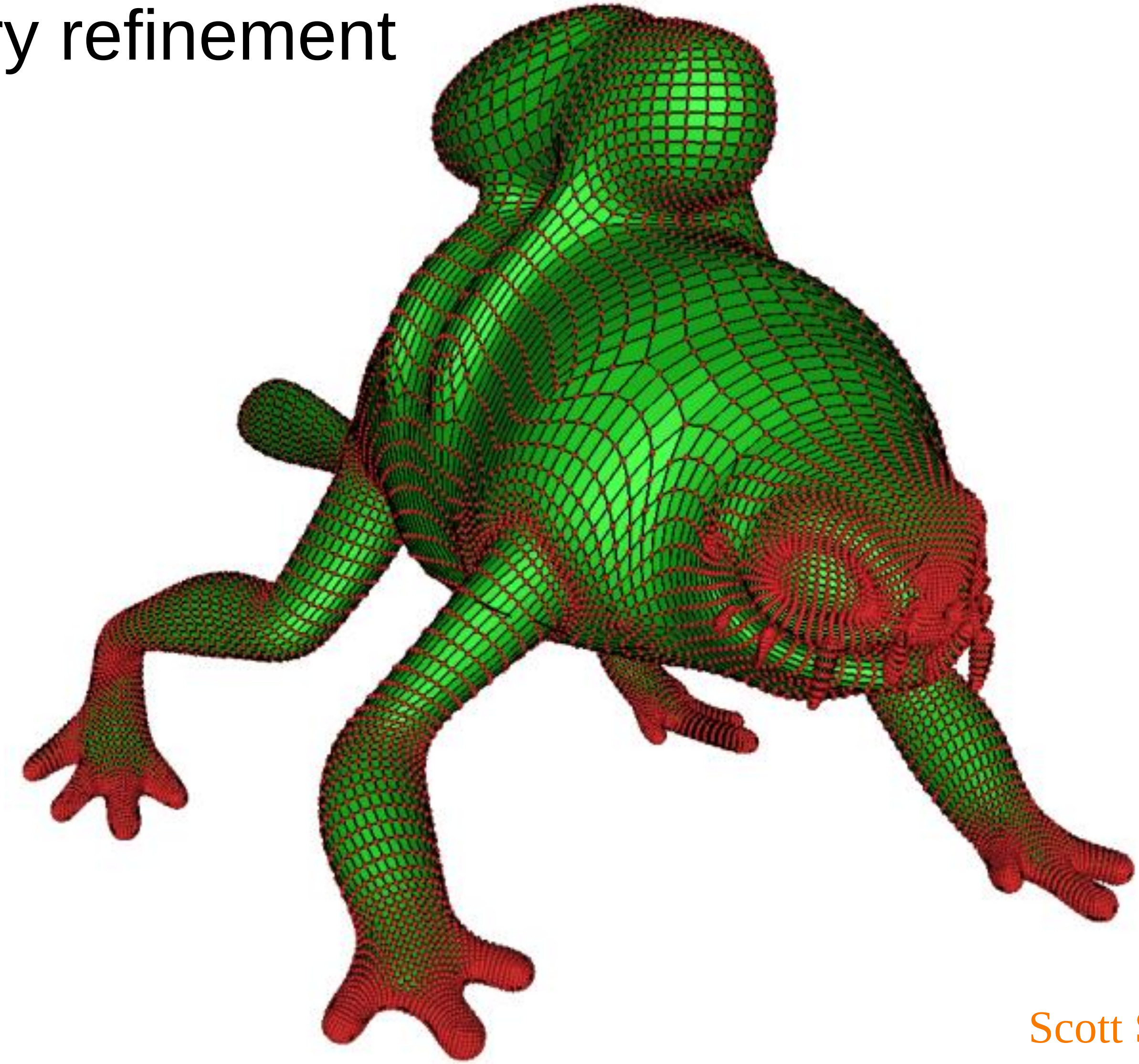
Subdivision Surfaces – Examples

- Topology refinement



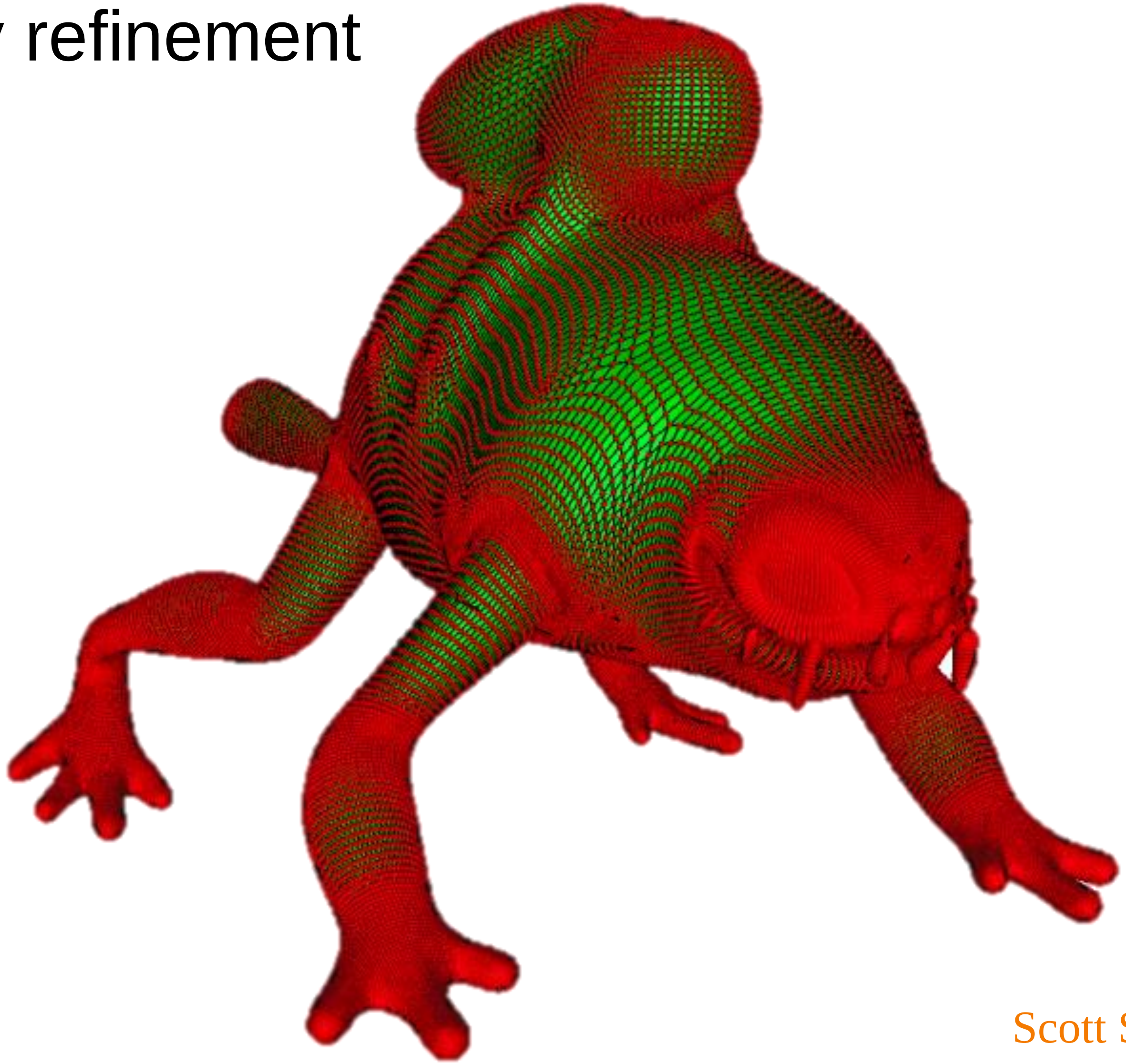
Subdivision Surfaces – Examples

- Geometry refinement



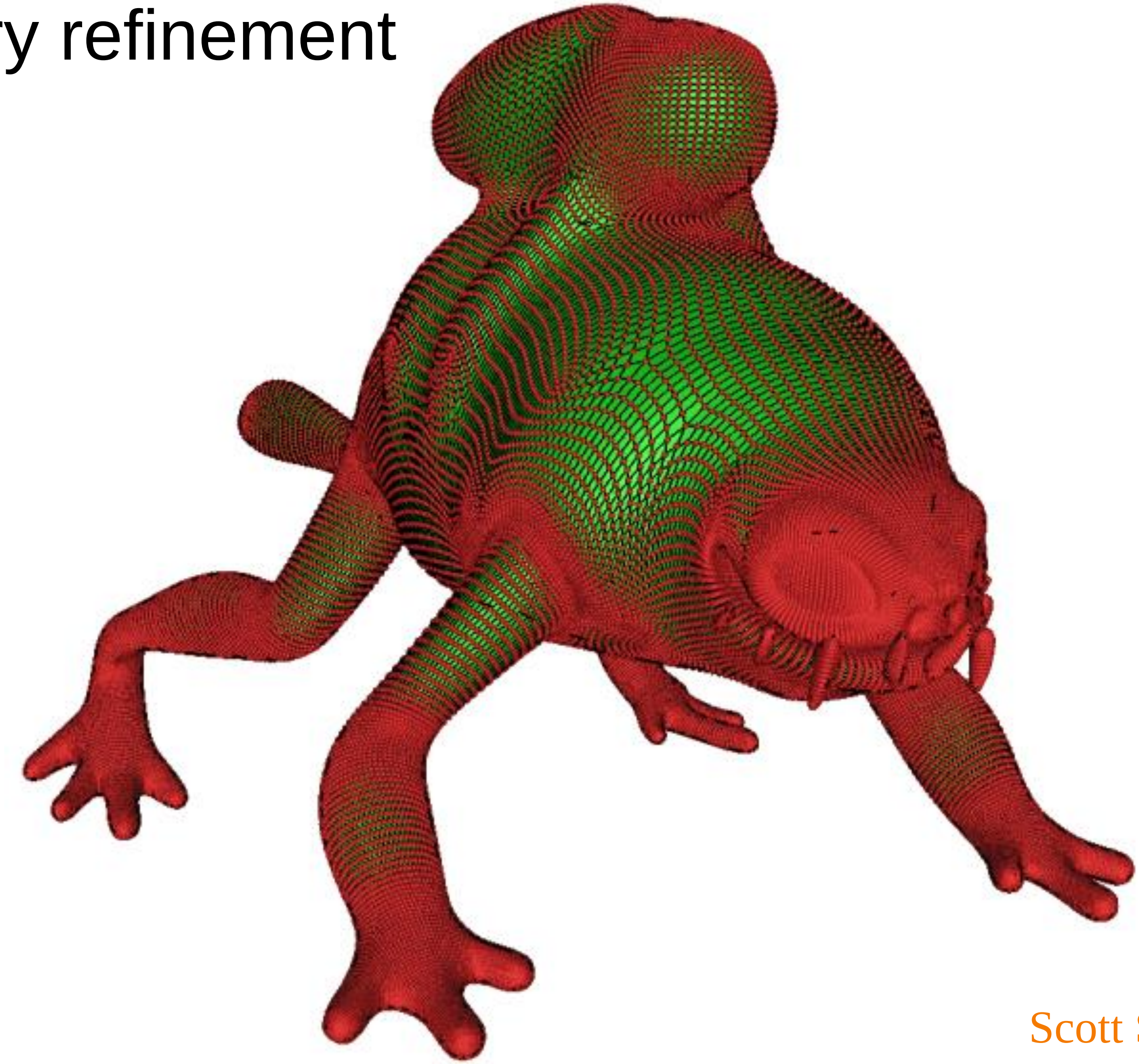
Subdivision Surfaces – Examples

- Topology refinement



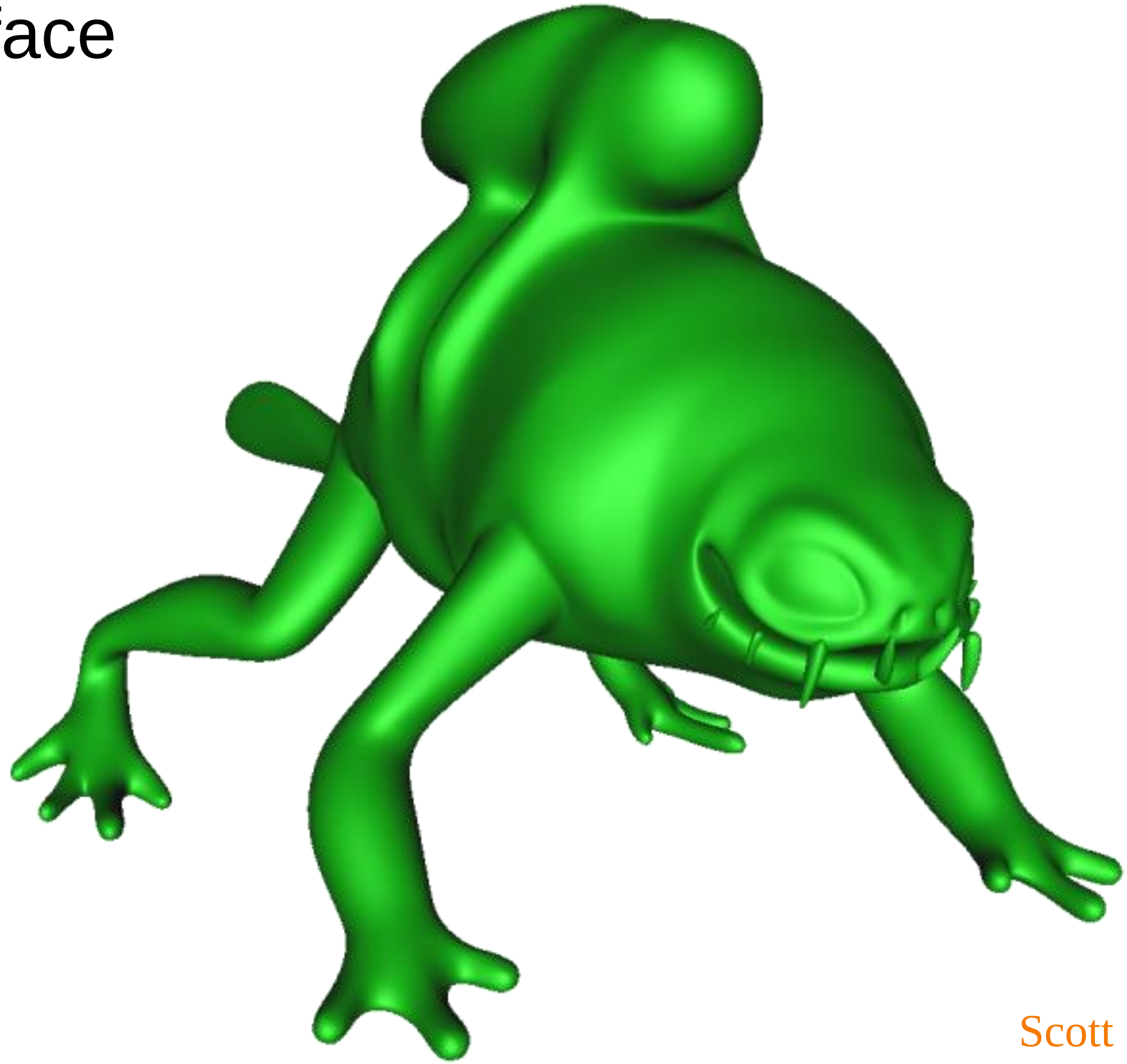
Subdivision Surfaces – Examples

- Geometry refinement



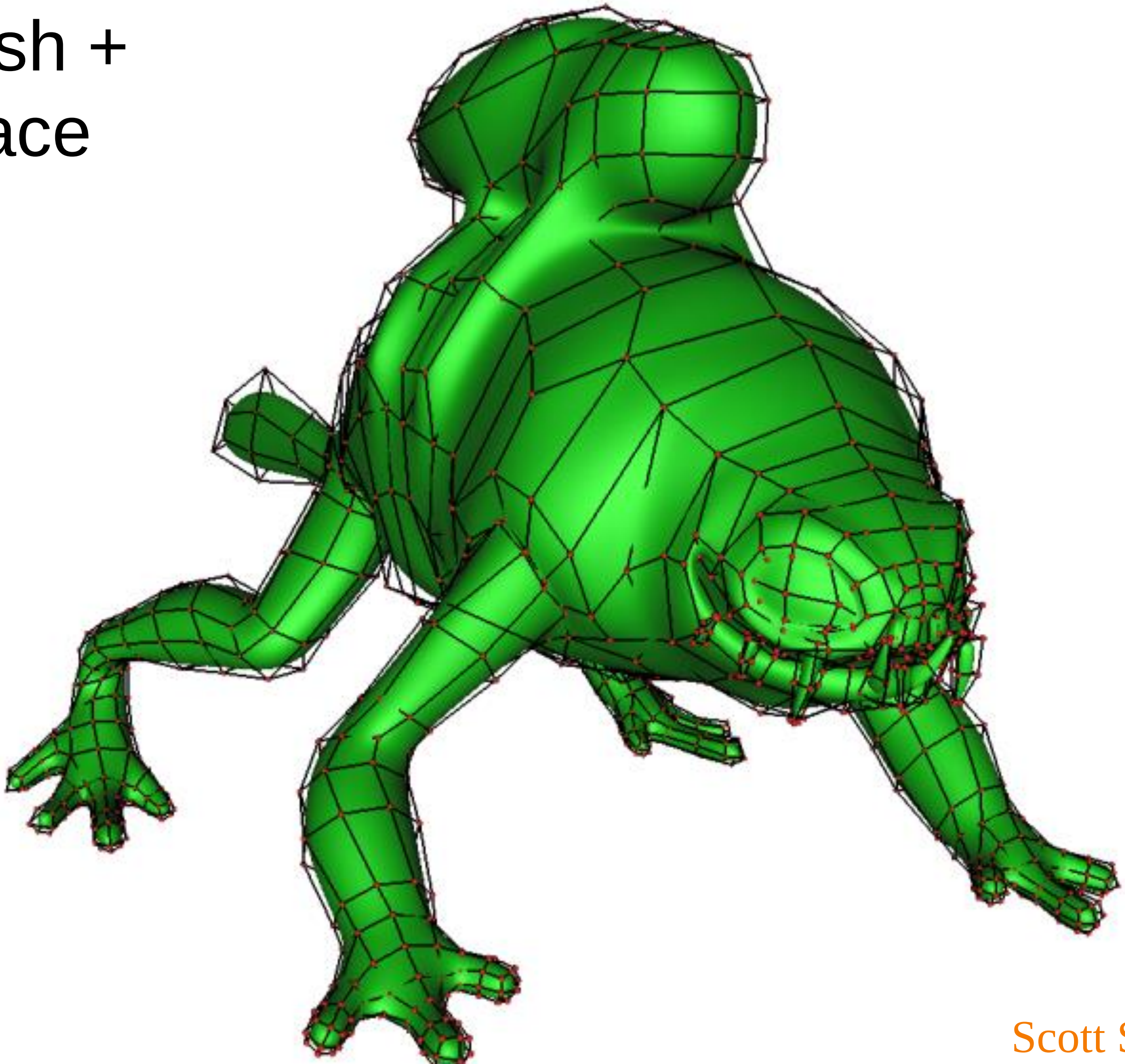
Subdivision Surfaces – Examples

- Limit surface



Subdivision Surfaces – Examples

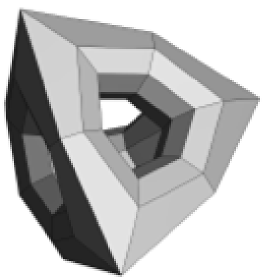
- Base mesh + limit surface

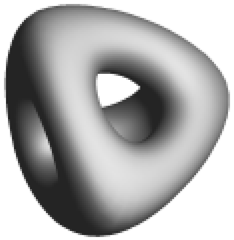


Scott Schaefer

Design of Subdivision Rules

- What types of input?
  - Quad meshes, triangle meshes, etc.
- How to refine topology?
  - Simple implementations
- How to refine geometry?
  - Smoothness guarantees in limit surface
    - » Continuity (C^0, C^1, C^2, …?)
  - Provable relationships between limit surface and original control mesh
    - » Interpolation of vertices?



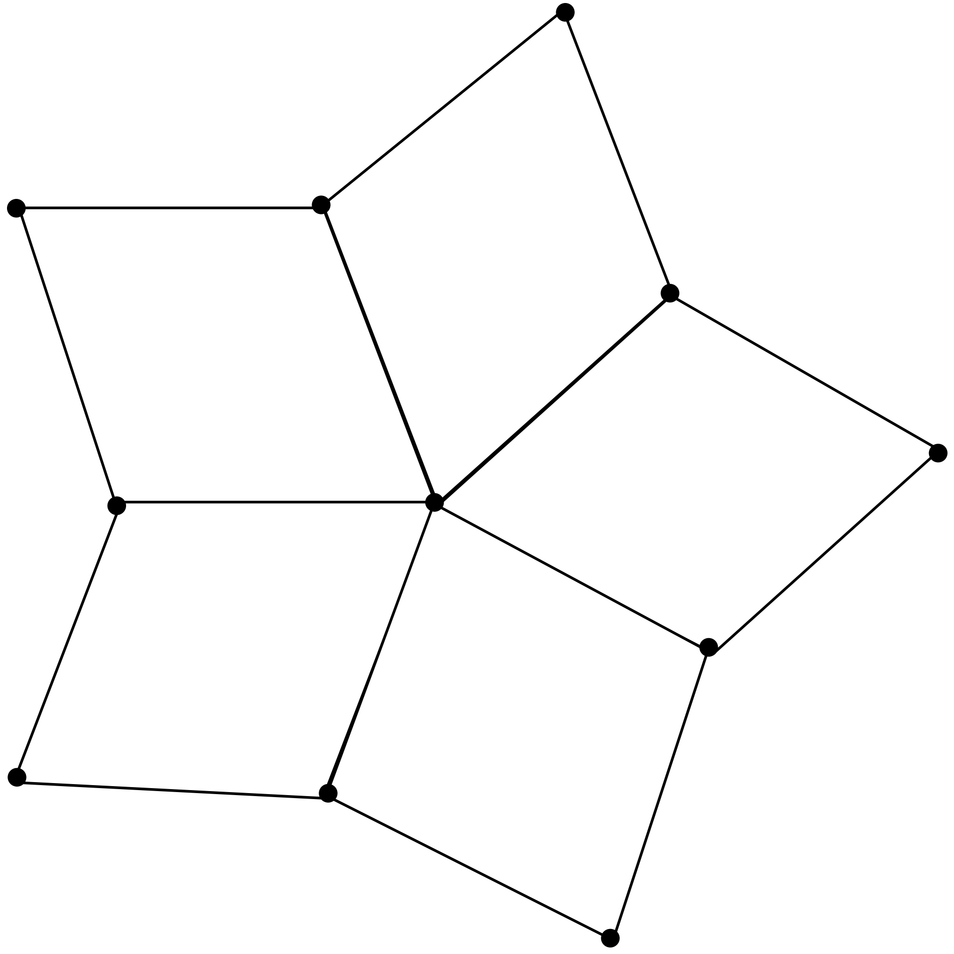


Linear Subdivision

- Type of input
  - Quad mesh -- four-sided polygons (*quads*)
  - Any number of quads may touch each vertex
- Topology refinement rule
  - Split every quad into four at midpoints
- Geometry refinement rule
  - Average vertex positions

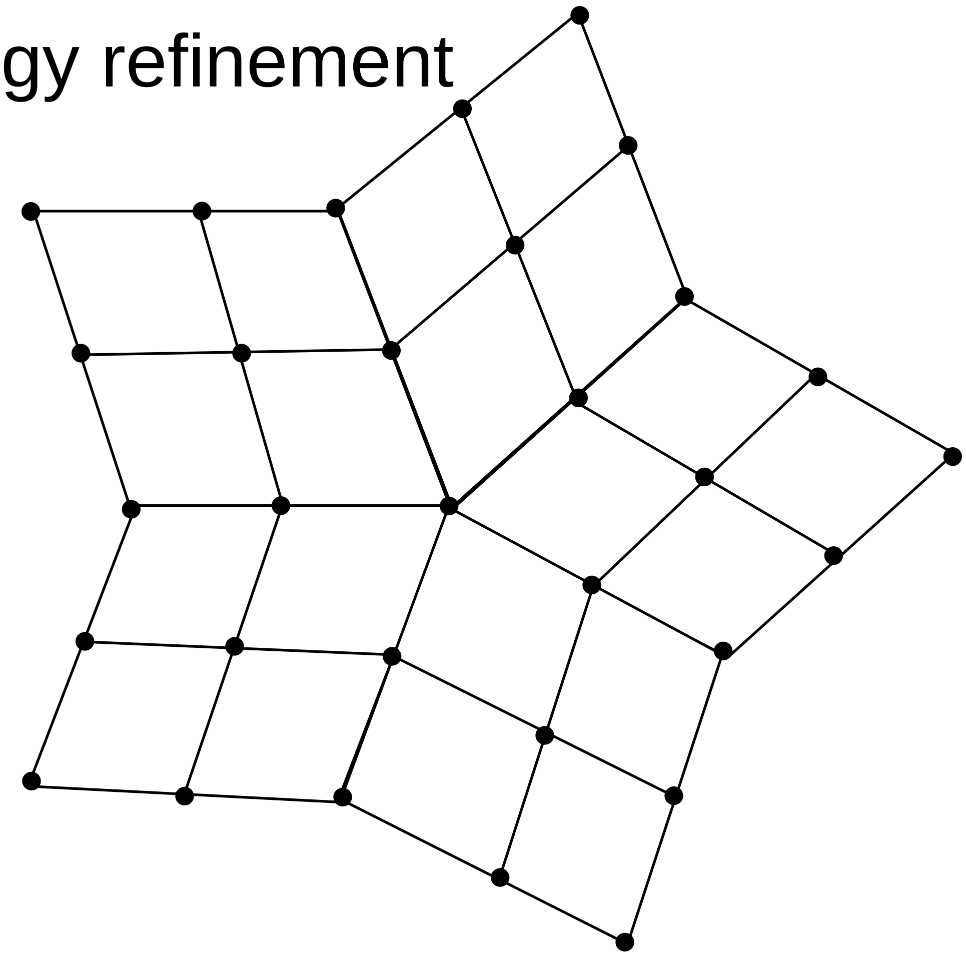
This is a simple example to demonstrate how subdivision schemes work

Linear Subdivision



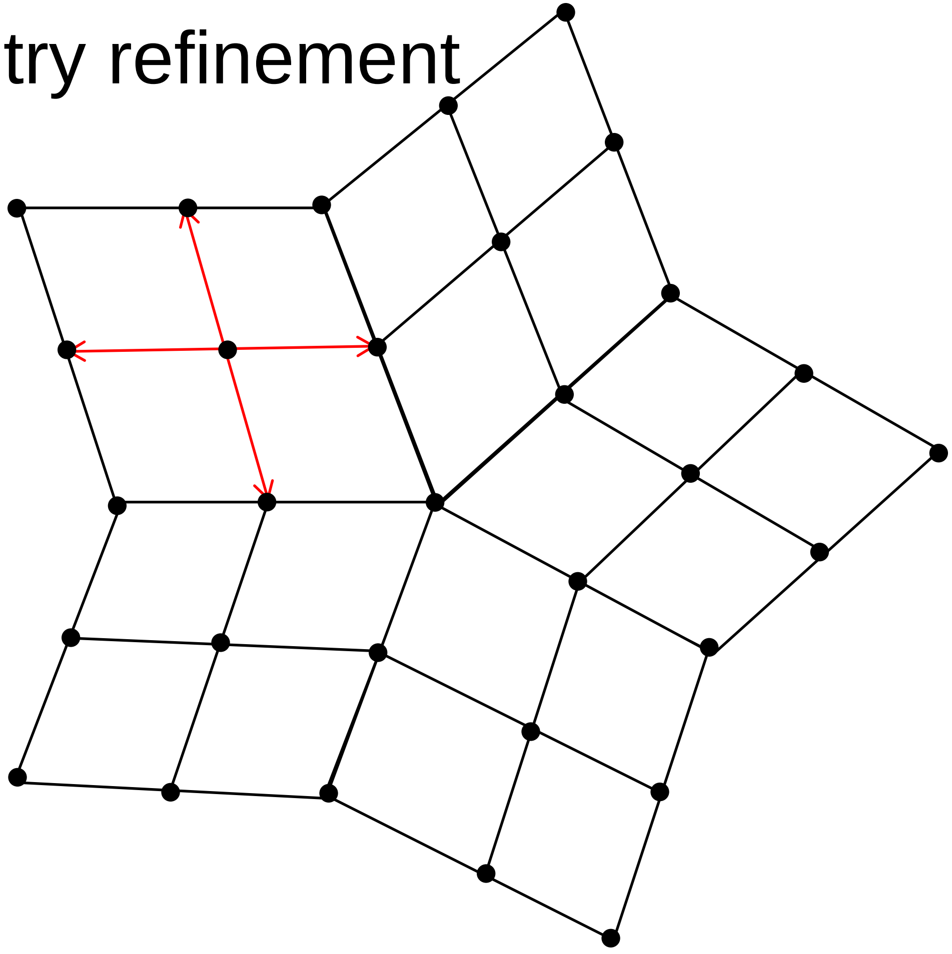
Linear Subdivision

- Topology refinement



Linear Subdivision

- Geometry refinement



Linear Subdivision

LinearSubivision (F_0, V_0, k)

for i = 1 …k levels

(F_i, V_i) = RefineTopology(F_{i-1}, V_{i-1})

RefineGeometry(F_i, V_i)

return (F_k, V_k)

Linear Subdivision

RefineTopology (F, V)

$newV = V$

$newF = \{\}$

for each face F_i

Insert new vertex c at centroid of F_i into $newV$

for j = 1 to 4

Insert in $newV$ new vertex e_j at

centroid of each edge ($F_{i,j}$, $F_{i,j+1}$)

for j = 1 to 4

Insert new face ($F_{i,j}$, e_j, c, $e_{j\text{-}1}$) into $newF$

return ($newF$, $newV$)

Linear Subdivision

RefineGeometry(F, V)

    $newV = V$

    $newF = F$

    for each vertex V_i in $newV$

        $weight$ = 0;

        $newV[i]$ = (0,0,0)

        for each face F_j connected to V_i

                $newV[i]$ += centroid of F_j

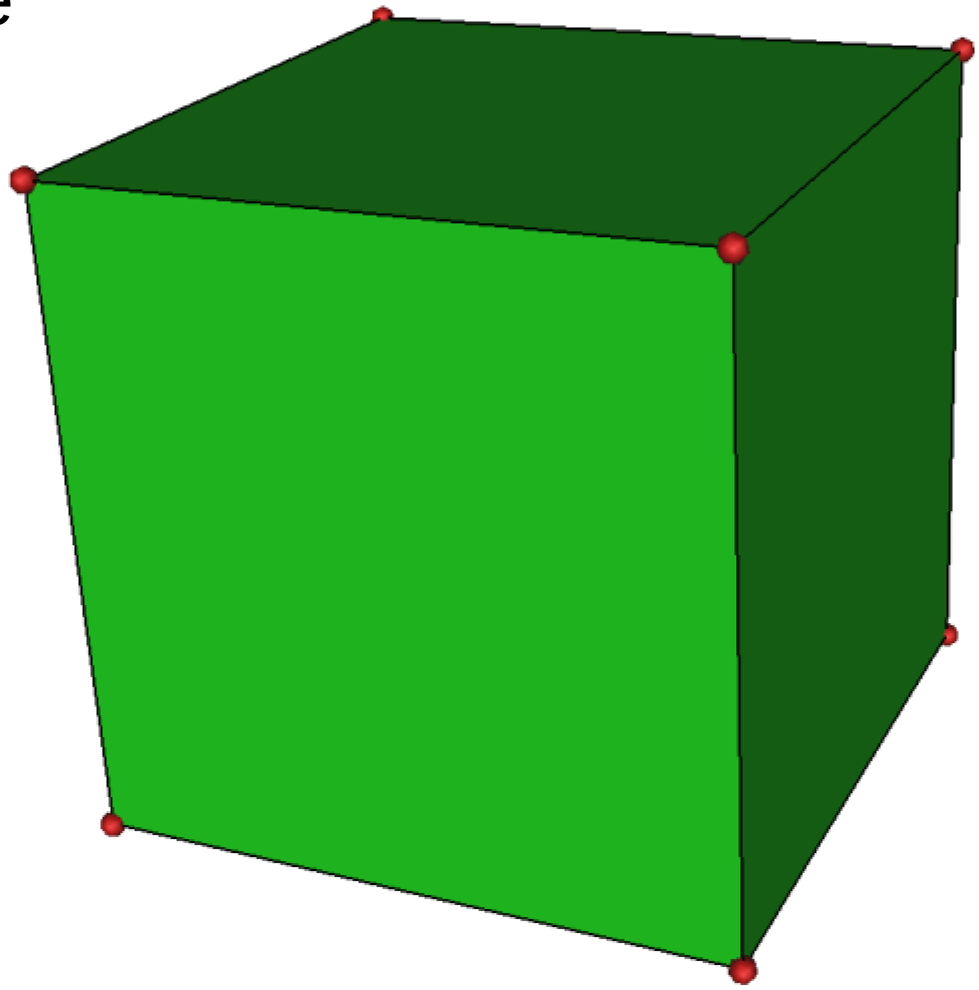
                $weight$ += 1.0;

        $newV[i]$ /= $weight$

    return ($newF$, $newV$)

Linear Subdivision

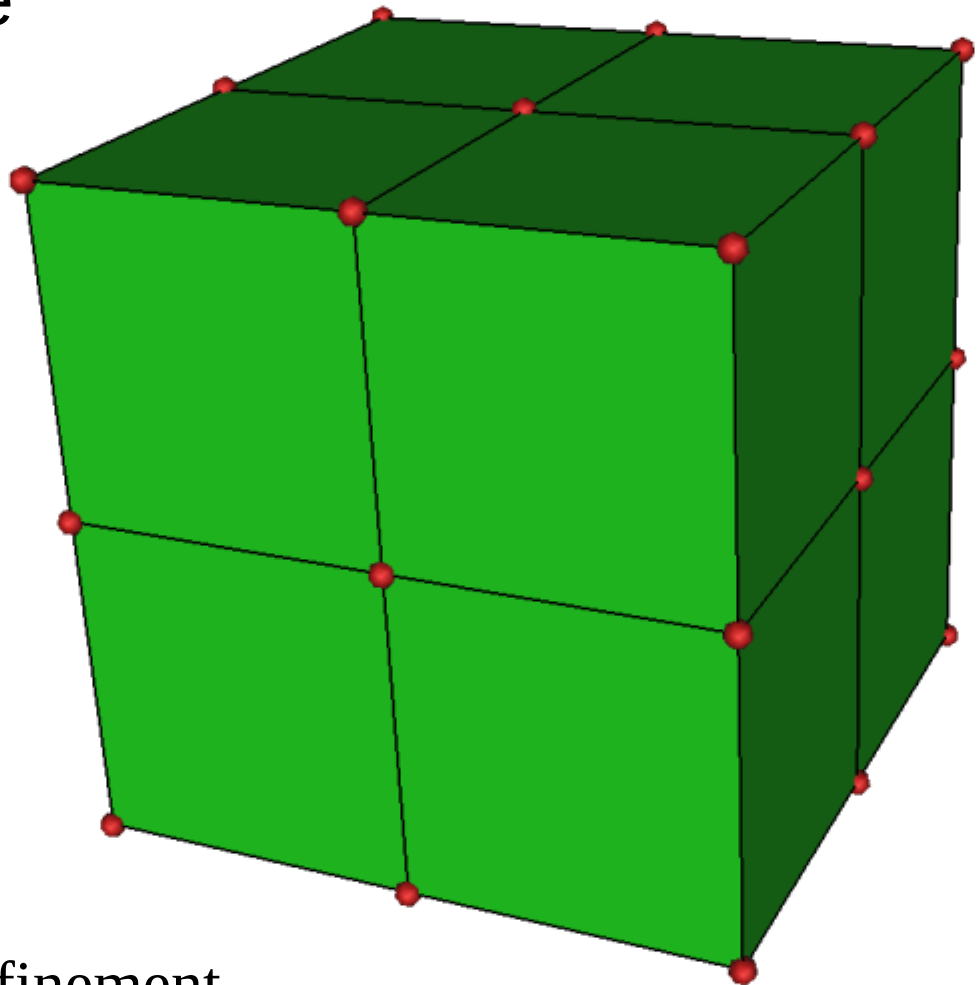
- Example



Input mesh

Linear Subdivision

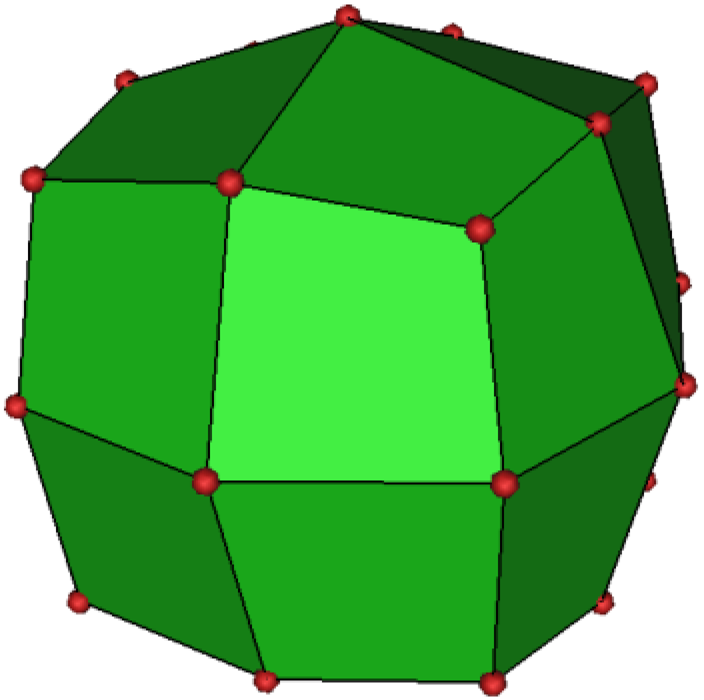
- Example



Topology refinement

Linear Subdivision

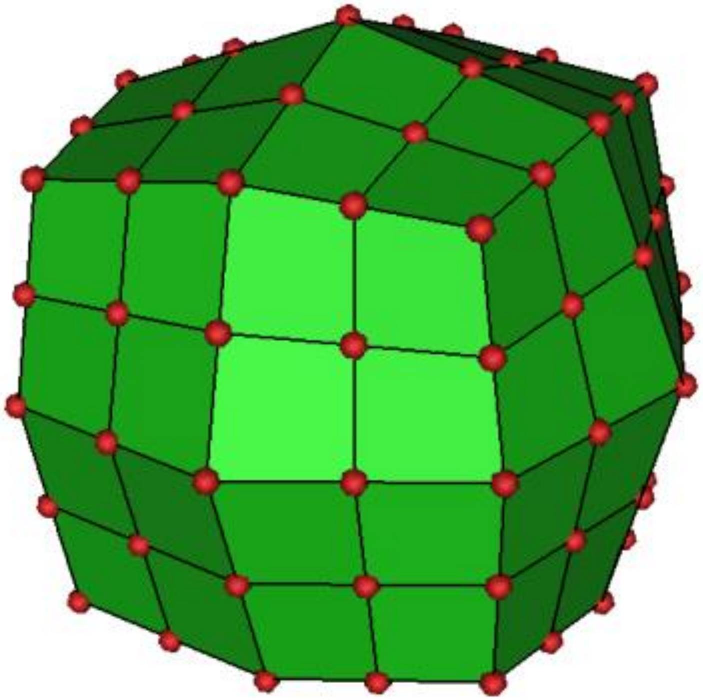
- Example



Geometry refinement

Linear Subdivision

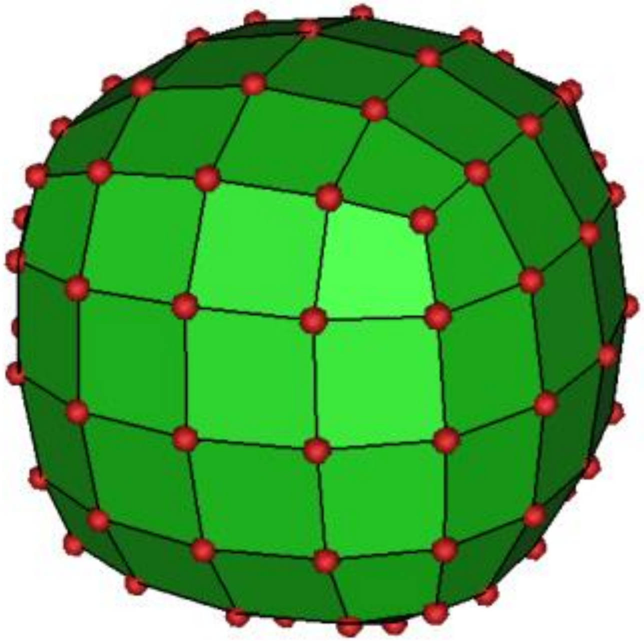
- Example



Topology refinement

Linear Subdivision

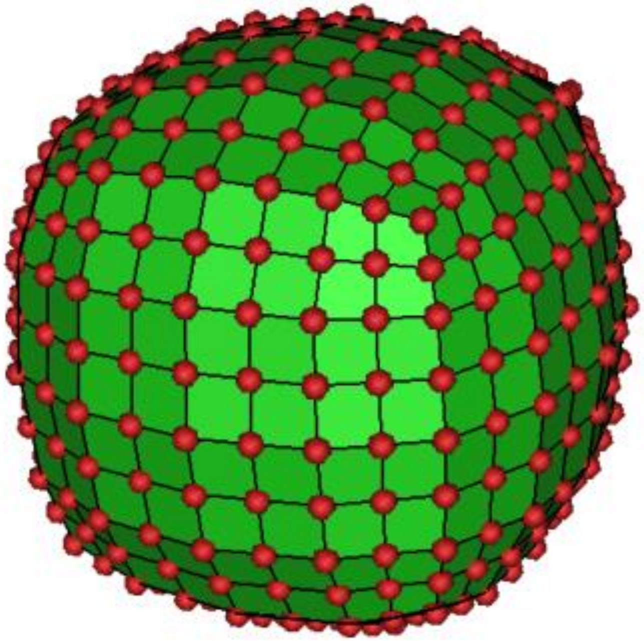
- Example



Geometry refinement

Linear Subdivision

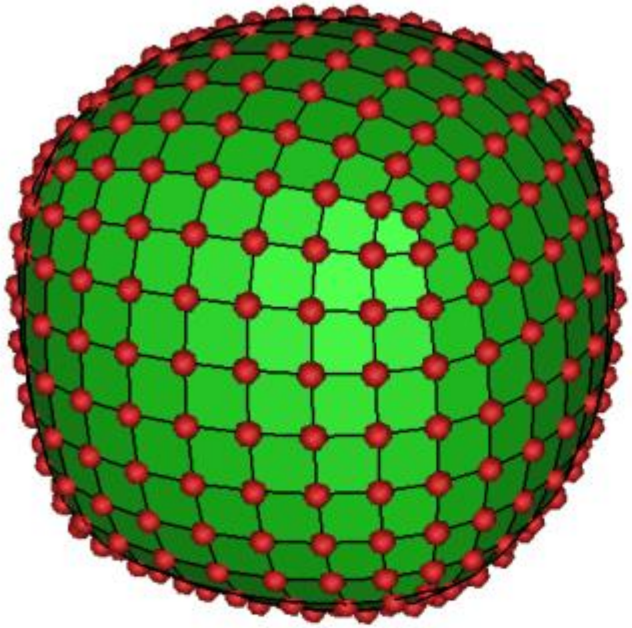
- Example



Topology refinement

Linear Subdivision

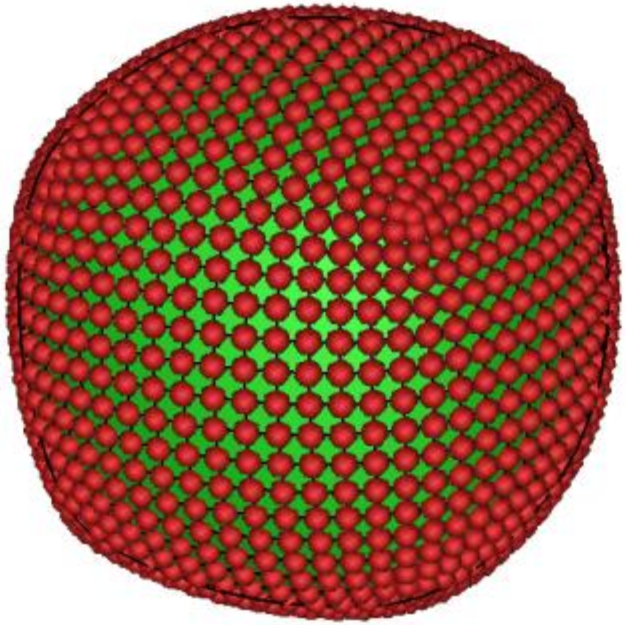
- Example



Geometry refinement

Linear Subdivision

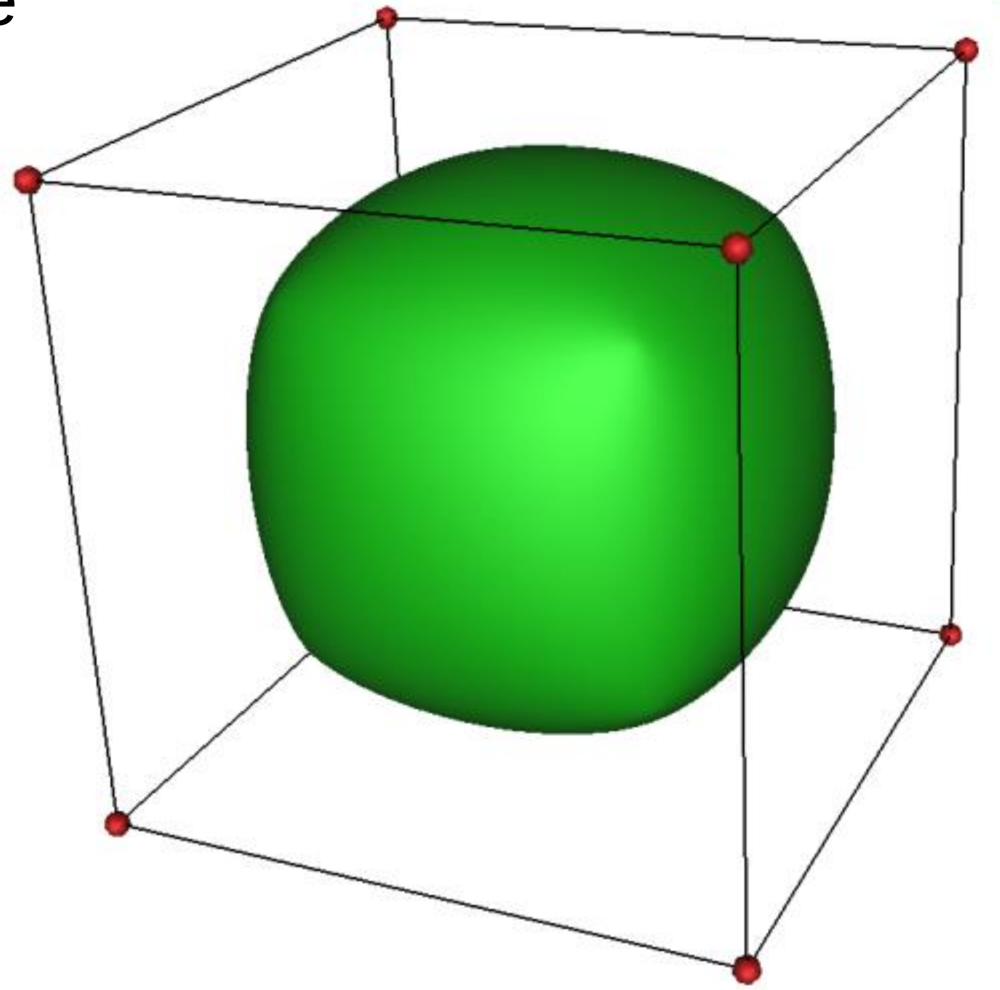
- Example



Topology refinement

Linear Subdivision

- Example



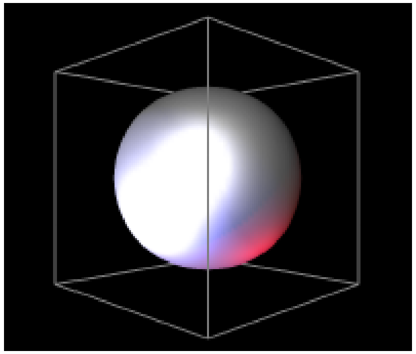
Final result

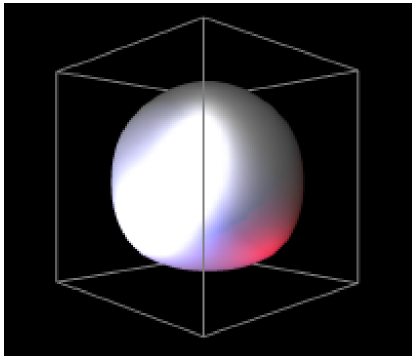
Subdivision Schemes

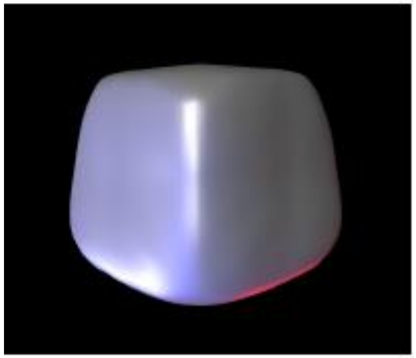
- Common subdivision schemes
  - Catmull-Clark
  - Loop
  - Many others
- Differ in ...
  - Input topology
  - How refine topology
  - How refine geometry

… which makes differences in …

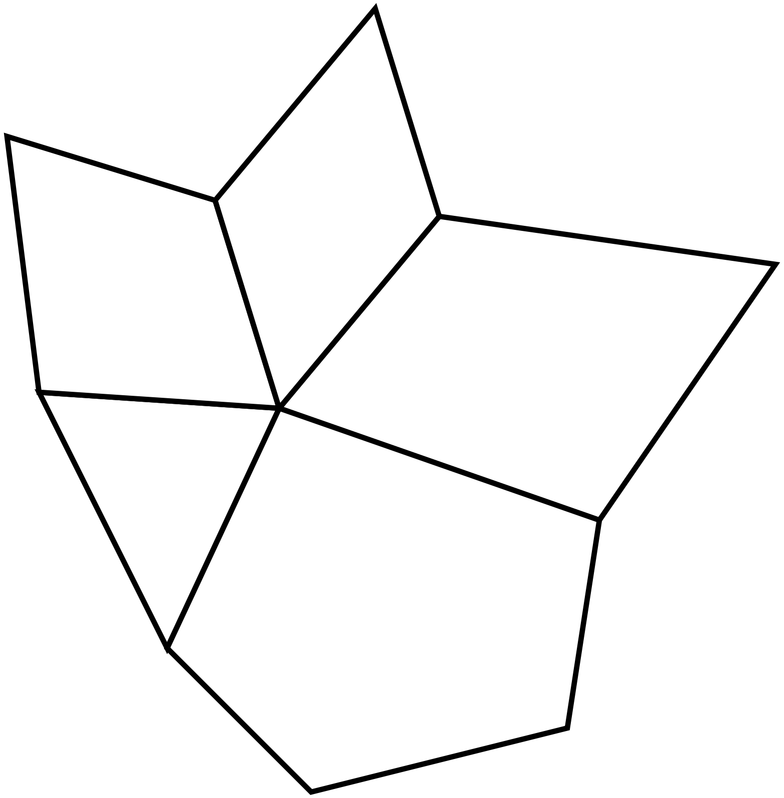
- Provable properties



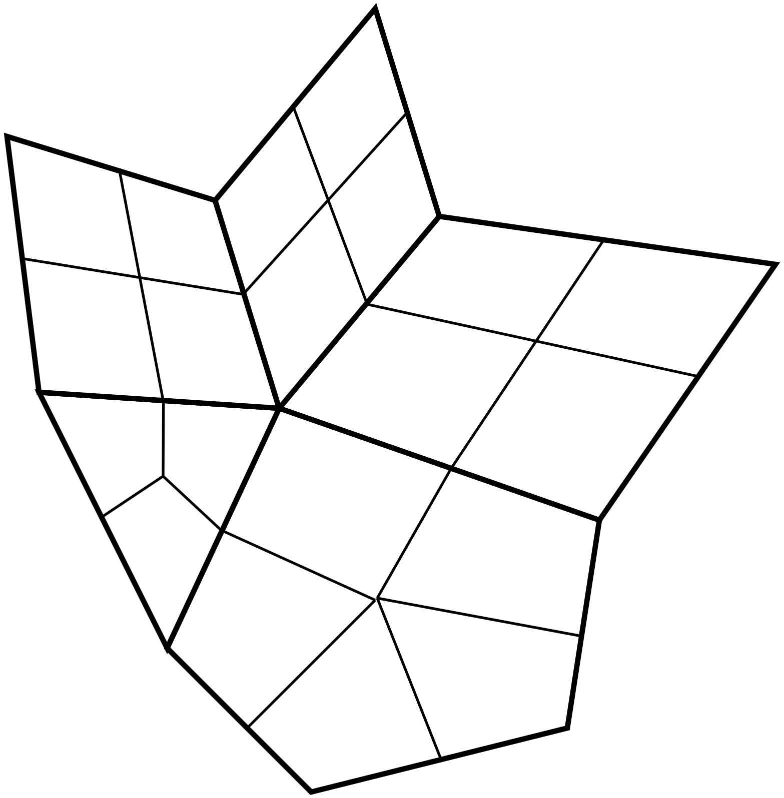




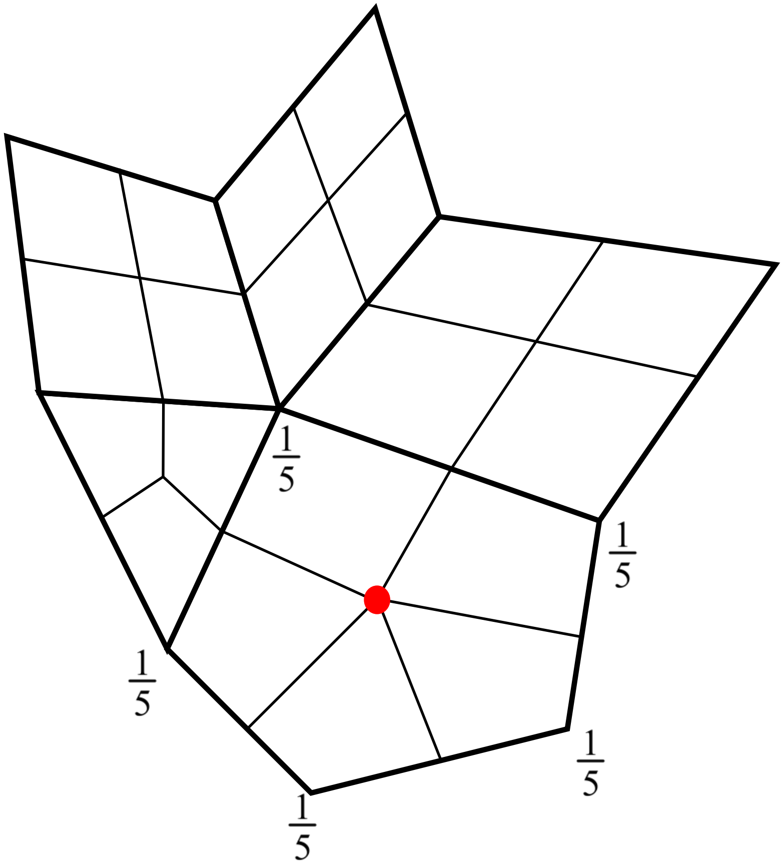
Catmull-Clark Subdivision



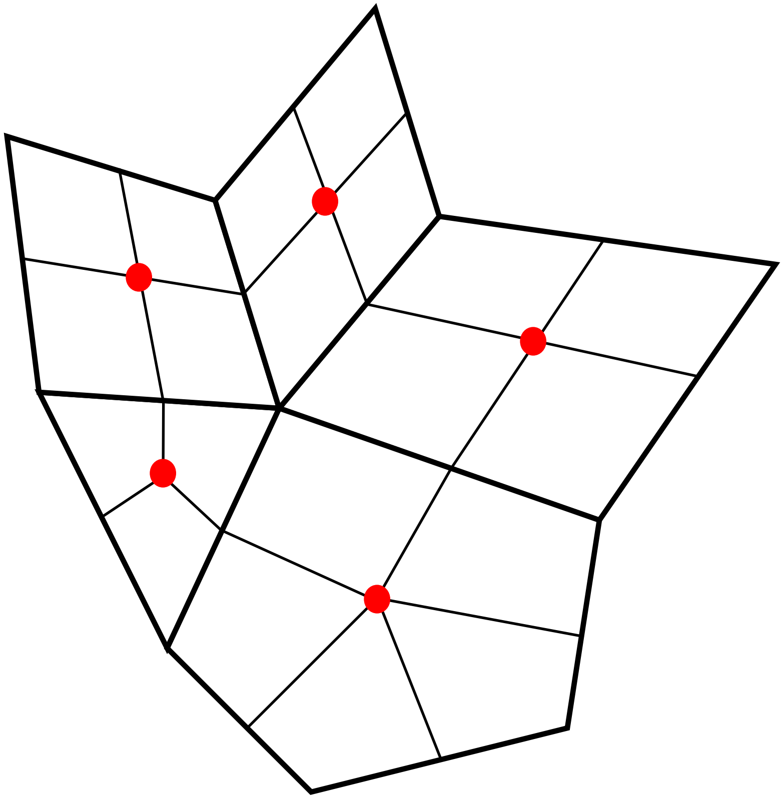
Catmull-Clark Subdivision



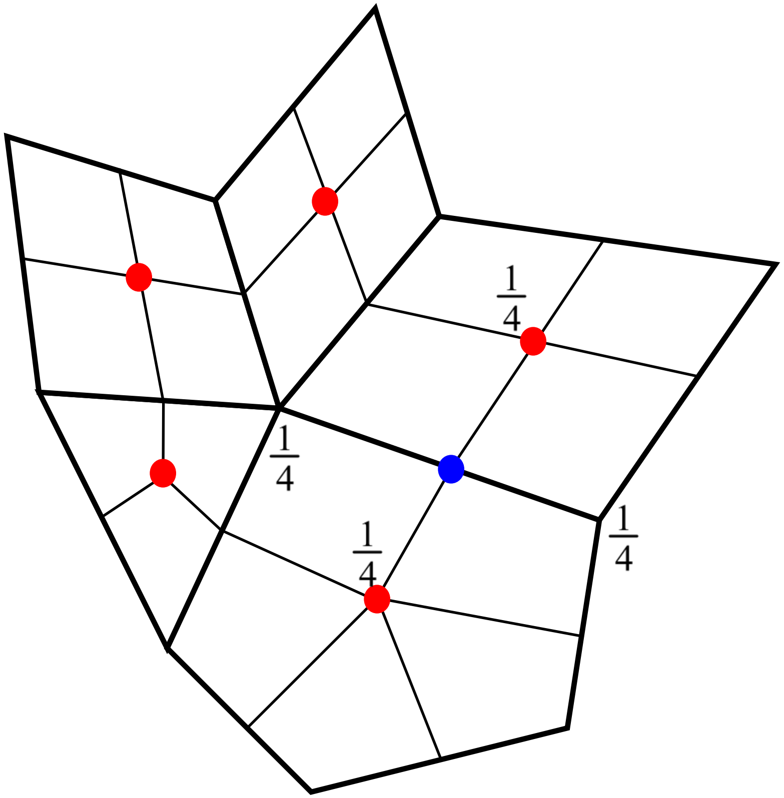
Catmull-Clark Subdivision



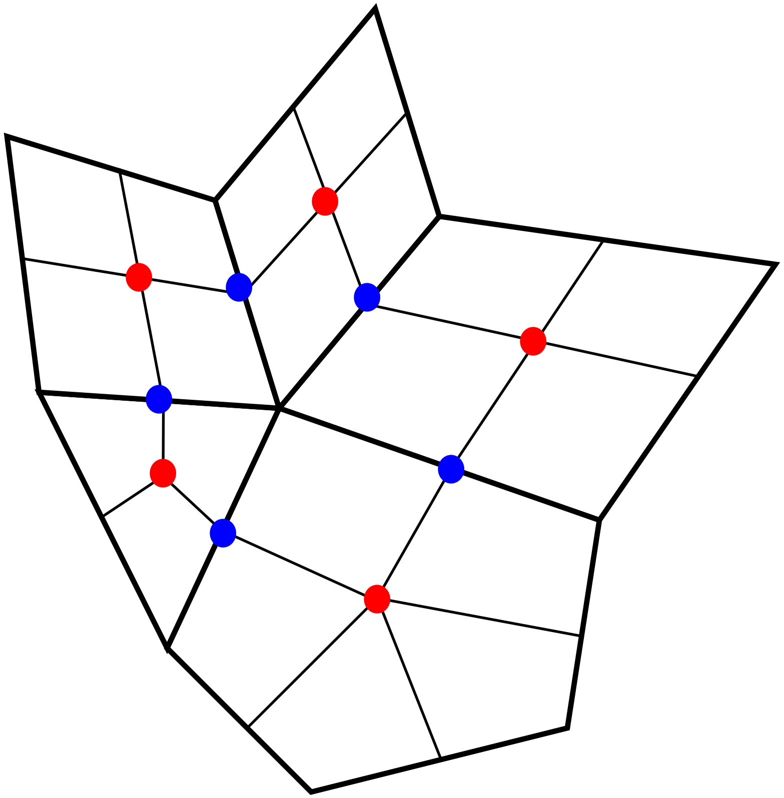
Catmull-Clark Subdivision



Catmull-Clark Subdivision



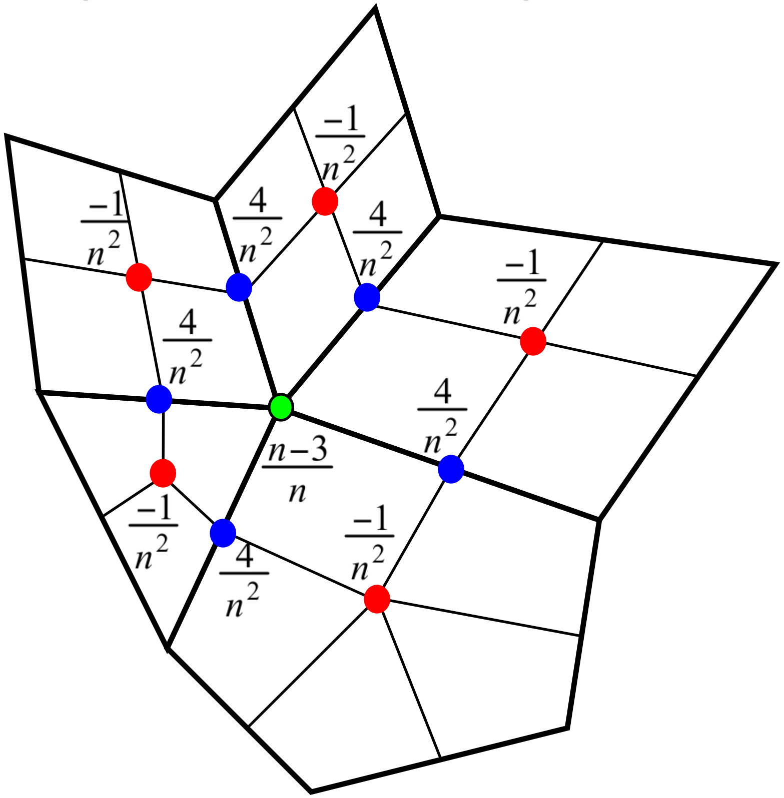
Catmull-Clark Subdivision



Scott Schaefer

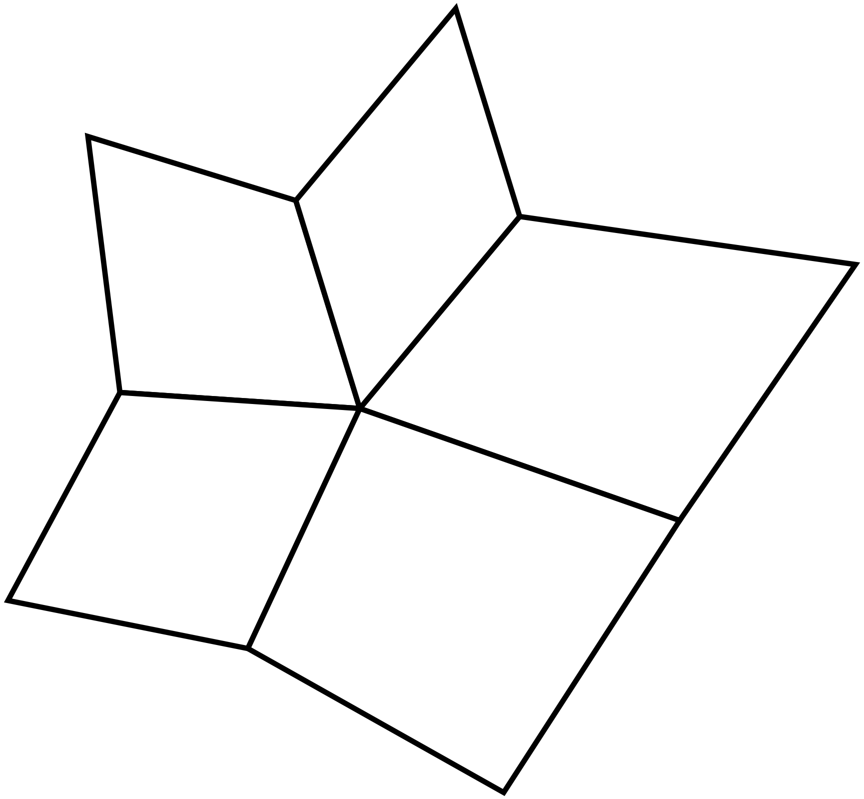
Catmull-Clark Subdivision

New (green dot) = (4 * avg of (blue dots) –1 * avg of (red dots) + (n-3) * (green dot)) / n

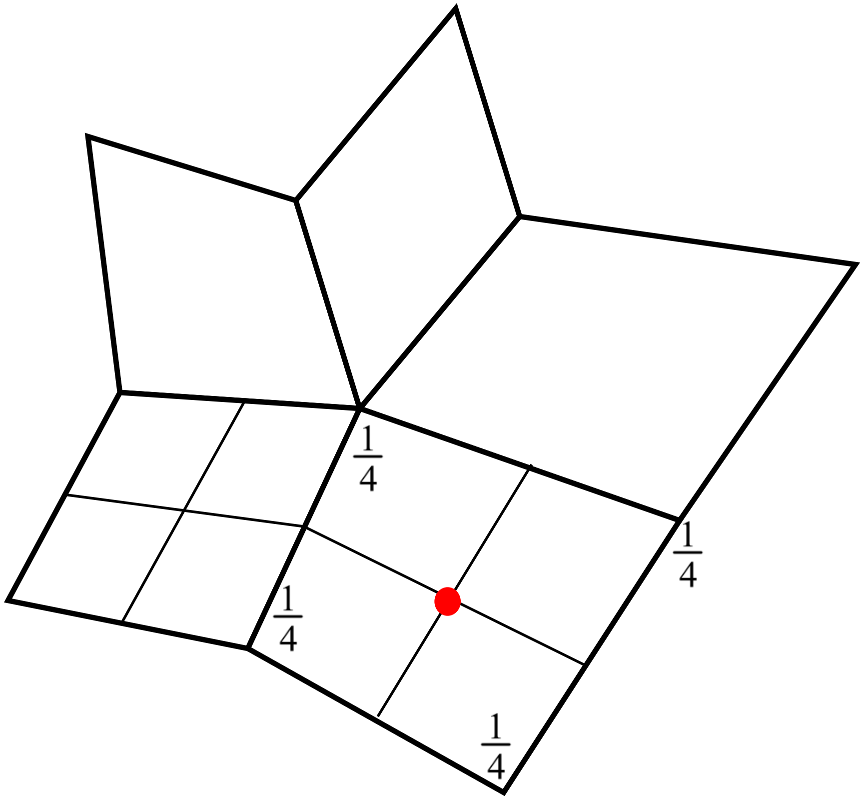


Scott Schaefer

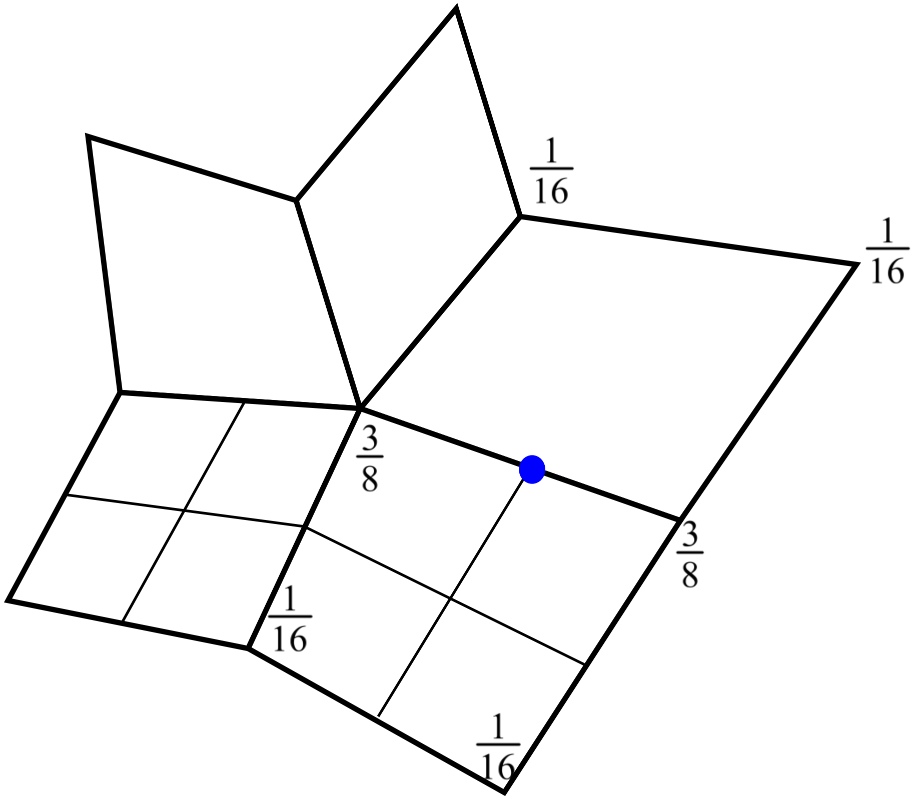
Catmull-Clark Subdivision



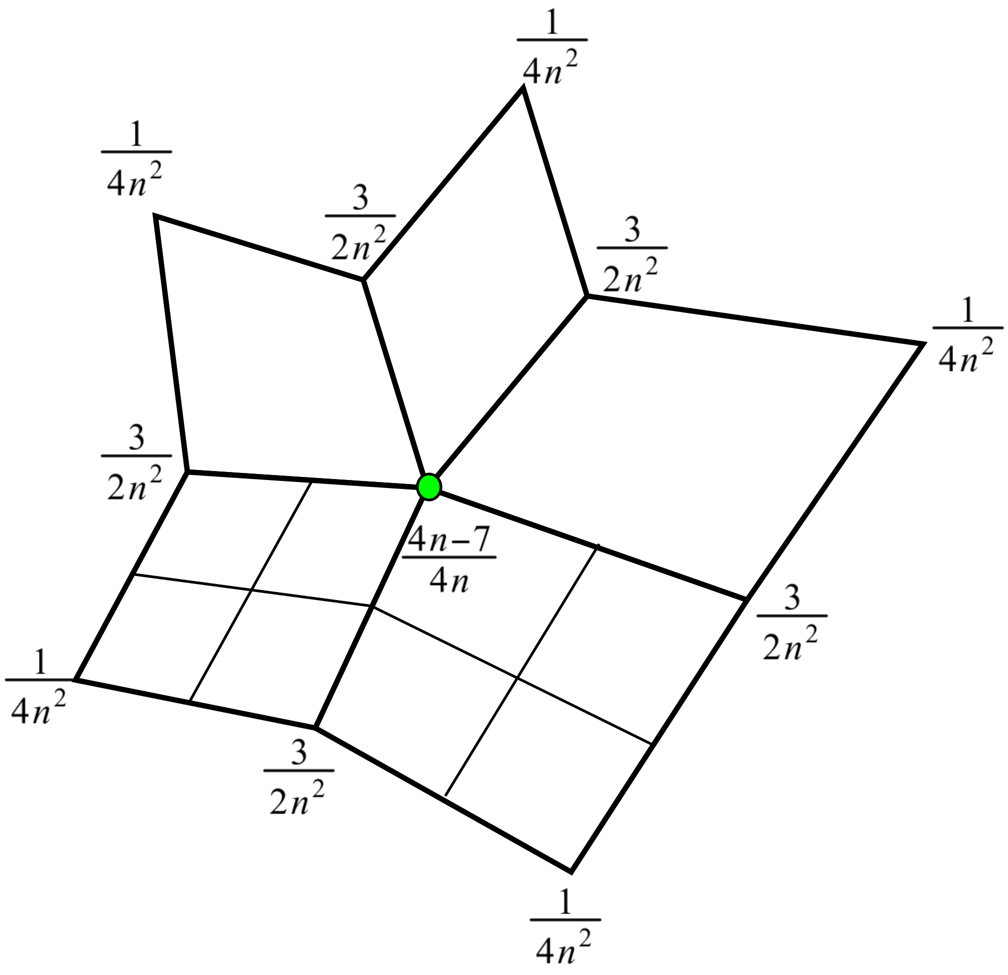
Catmull-Clark Subdivision



Catmull-Clark Subdivision



Catmull-Clark Subdivision



Scott Schaefer

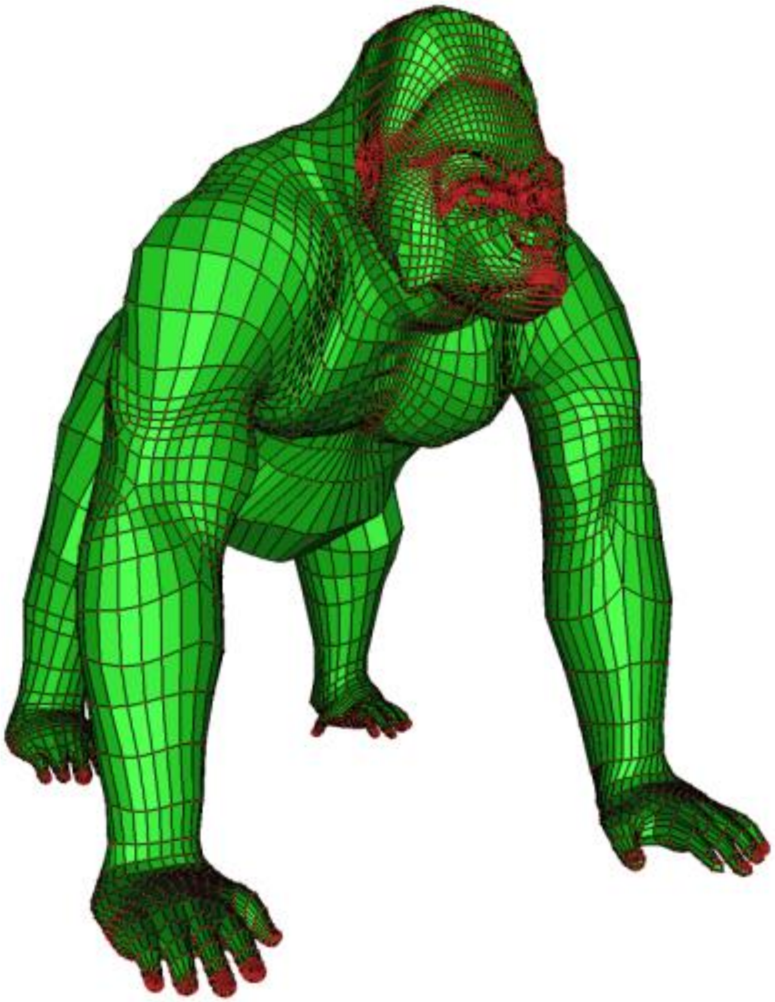
Catmull-Clark Subdivision

Linear Subdivision

Catmull-Clark Subdivision

Scott Schaefer

Catmull-Clark Subdivision

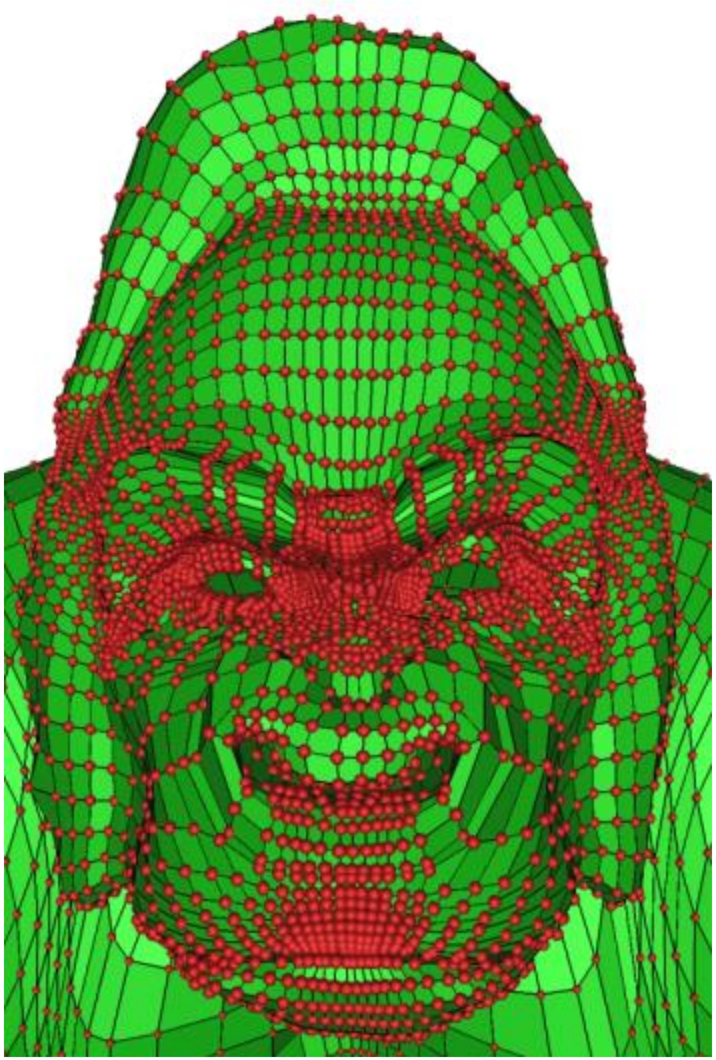


Scott Schaefer

Catmull-Clark Subdivision

Scott Schaefer

Catmull-Clark Subdivision



Catmull-Clark Subdivision

Scott Schaefer

Catmull-Clark Subdivision

- One round of subdivision produces all quads
- Smoothness of limit surface
  - C^2 almost everywhere
  - C^1 at vertices with valence $\neq$ 4
- Relationship to control mesh
  - Does not interpolate input vertices
  - Within convex hull
- Most commonly used subdivision scheme in the movies…

Pixar

Catmull-Clark Subdivision

Geri's Game

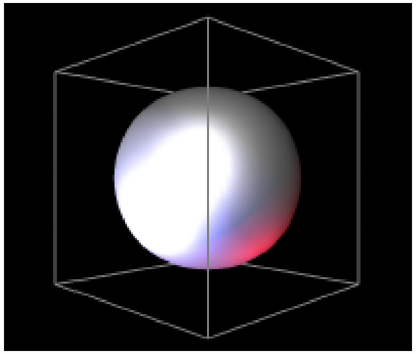
Pixar

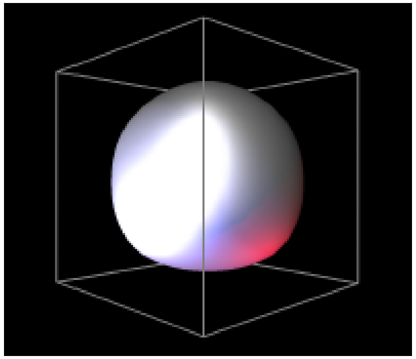
Subdivision Schemes

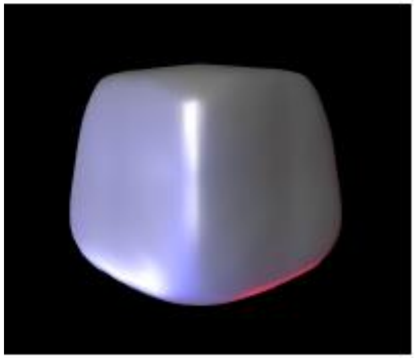
- Common subdivision schemes
  - Catmull-Clark
  - ➢ Loop
  - Many others
- Differ in ...
  - Input topology
  - How refine topology
  - How refine geometry

… which makes differences in …

  - Provable properties

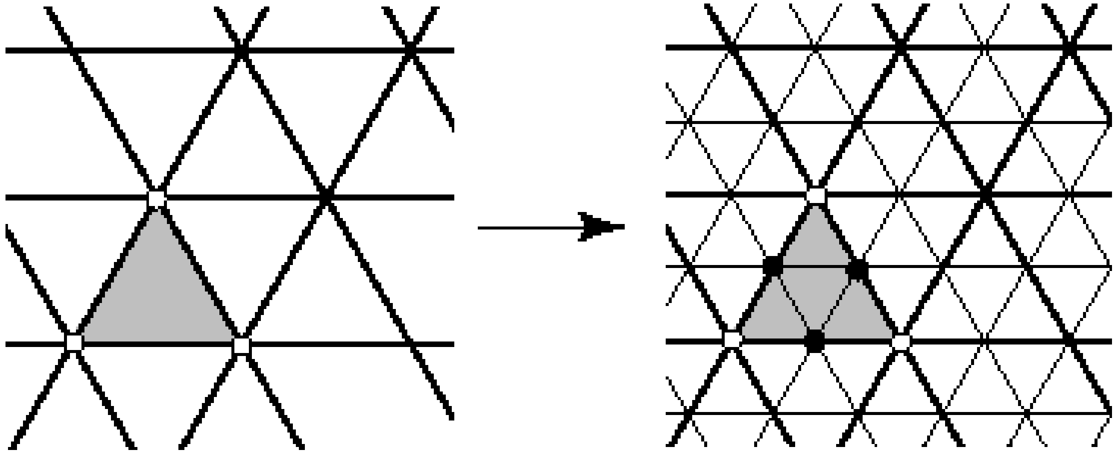






Loop Subdivision

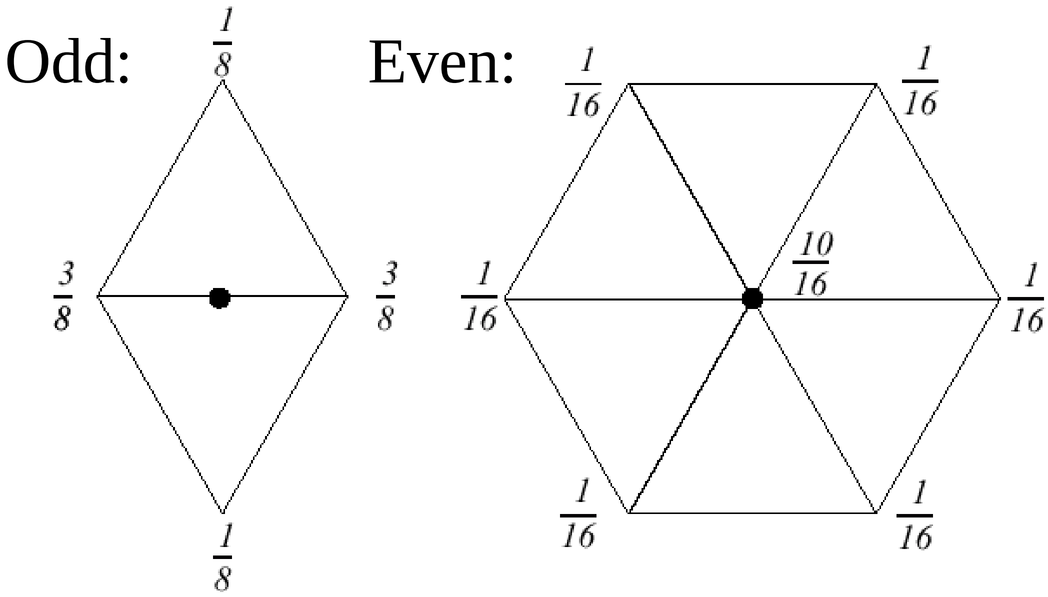
- Operates on pure triangle meshes
- Subdivision rules
  - Linear subdivision
  - Averaging rules for “even / odd” (white / black) vertices



Zorin & Schroeder
SIGGRAPH 99
Course Notes

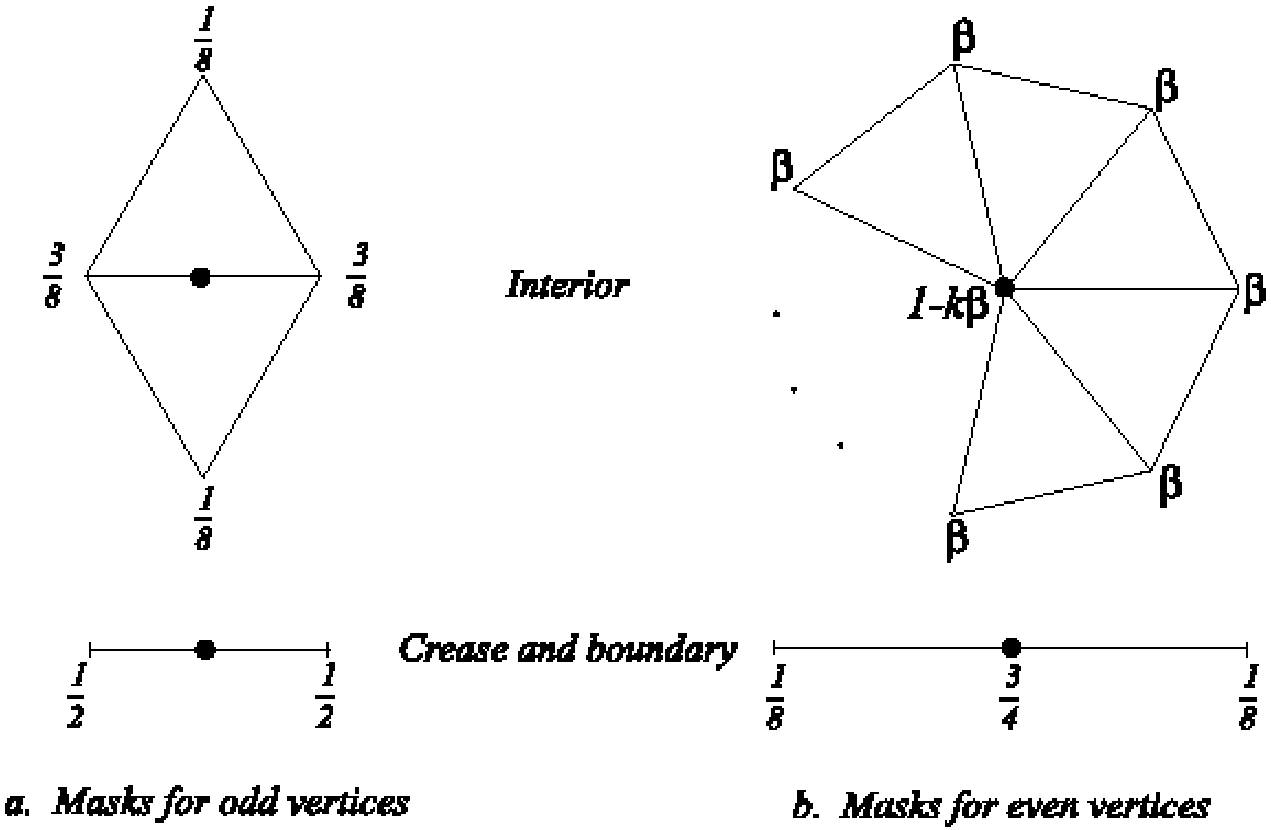
Loop Subdivision

- Averaging rules
  - Weights for “odd” and “even” vertices



Loop Subdivision

- Rules for *extraordinary vertices* and *boundaries*:



Zorin & Schroeder
SIGGRAPH 99
Course Notes

Loop Subdivision

- How to choose β?
  - Analyze properties of limit surface
  - Interested in continuity of surface and smoothness
  - Involves calculating eigenvalues of matrices

» Original Loop

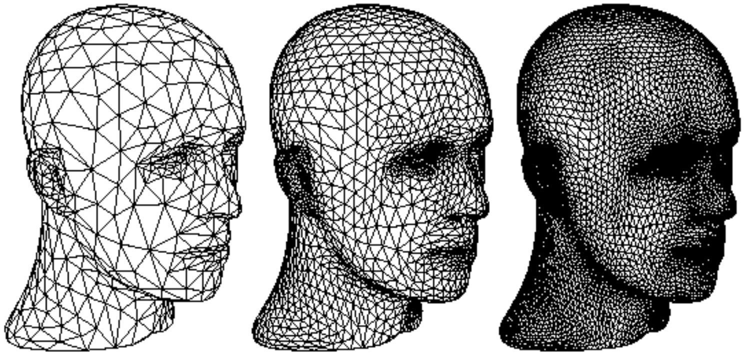
$$\beta = \frac{1}{n}\left(\frac{5}{8} - \left(\frac{3}{8} + \frac{1}{4}\cos\frac{2\pi}{n}\right)^2\right)$$

» Warren

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

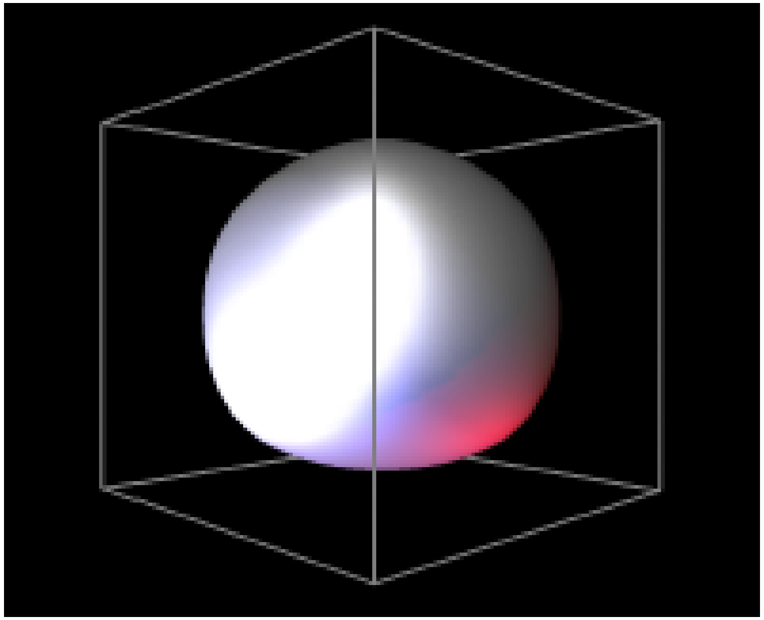
Loop Subdivision

- Operates only on triangle meshes
- Smoothness of limit surface
  - C^2 almost everywhere
  - C^1 at vertices with valence $\neq 6$
- Relationship to control mesh
  - Does not interpolate input vertices
  - Within convex hull

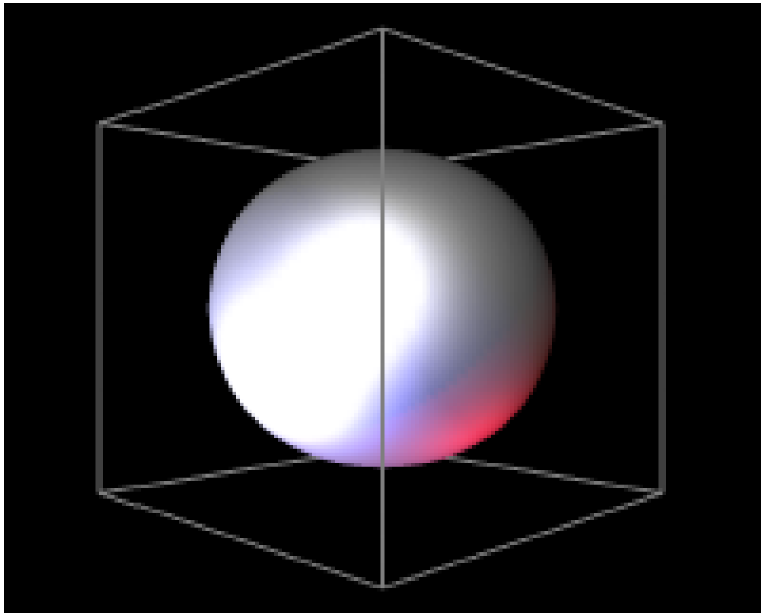


Zorin & Schroeder
SIGGRAPH 99
Course Notes

Subdivision Schemes

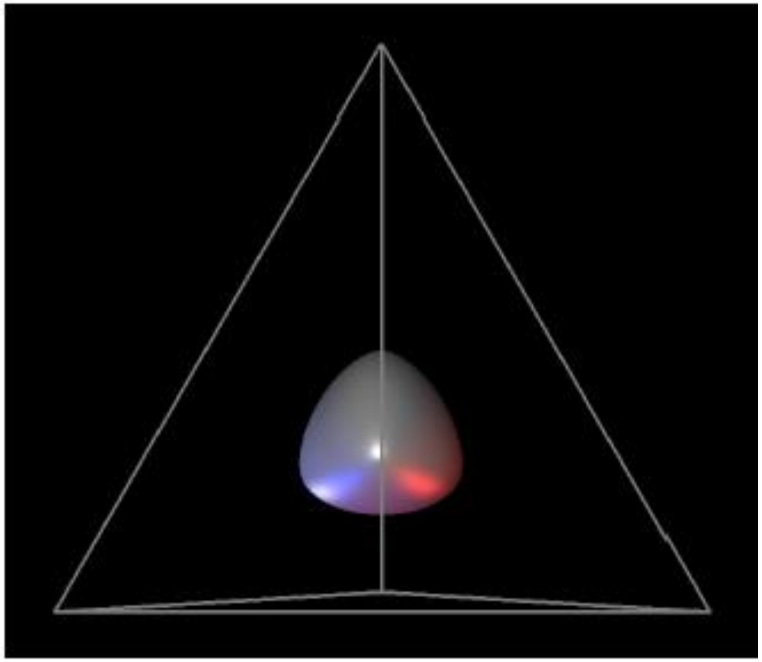


Loop

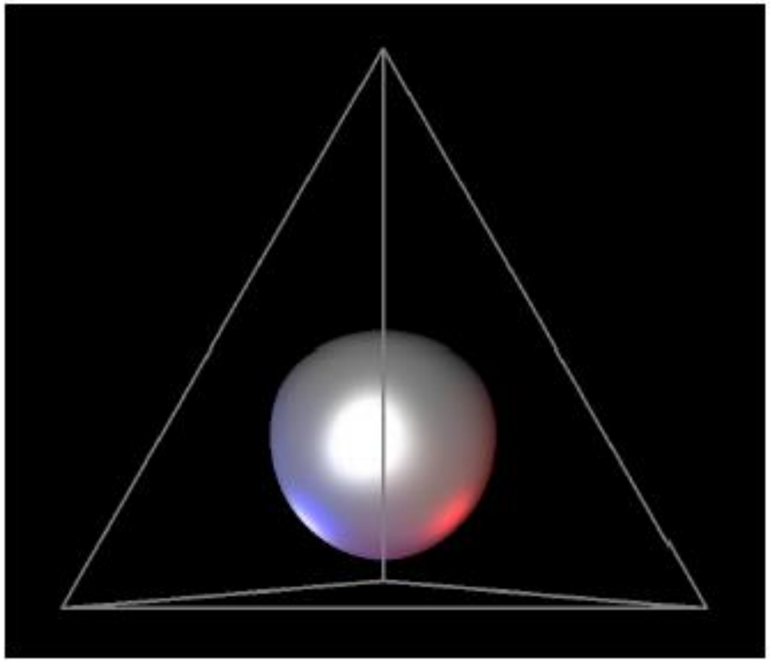


Catmull-Clark

Subdivision Schemes



Loop

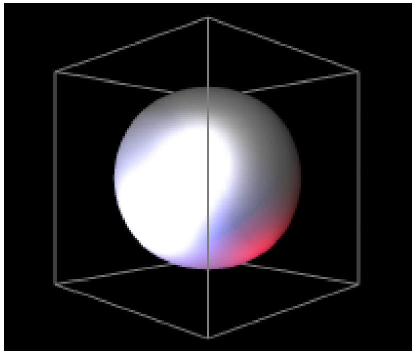
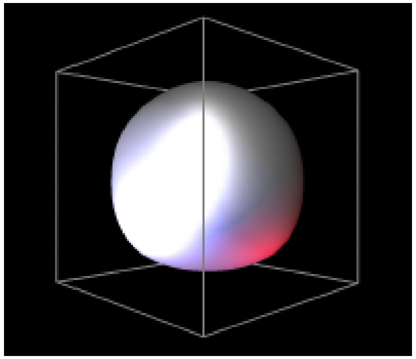
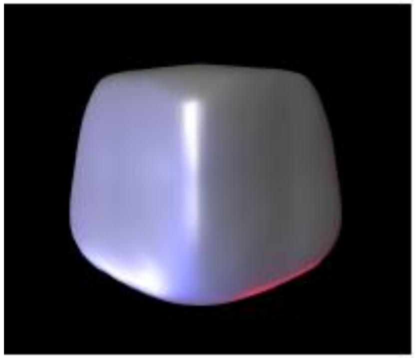


Catmull-Clark

Subdivision Schemes

- Common subdivision schemes
  - Catmull-Clark
  - Loop
  - Many others
- Differ in ...
  - Input topology
  - How refine topology
  - How refine geometry

… which makes differences in …
  - Provable properties

Subdivision Schemes

- Other subdivision schemes

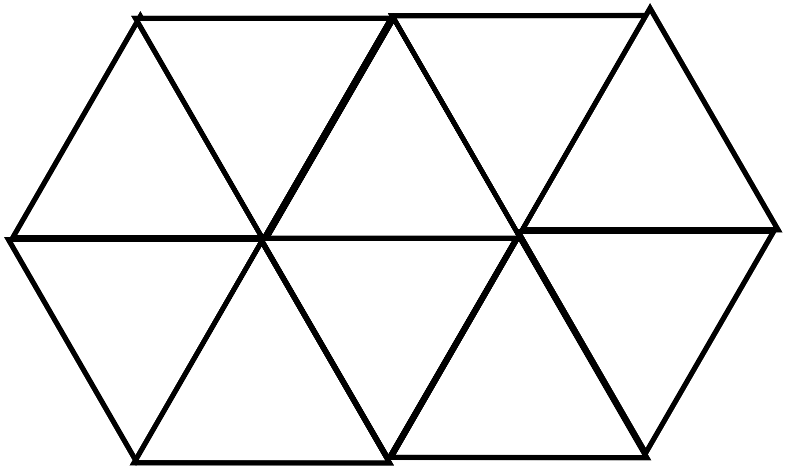
Face split		
	Triangular meshes	*Quad. meshes*
Approximating	Loop (C^2)	Catmull-Clark (C^2)
Interpolating	Mod. Butterfly (C^1)	Kobbelt (C^1)

Vertex split
Doo-Sabin, Midedge (C^1)
Biquartic (C^2)

Zorin & Schroeder, SIGGRAPH 99 , Course Notes

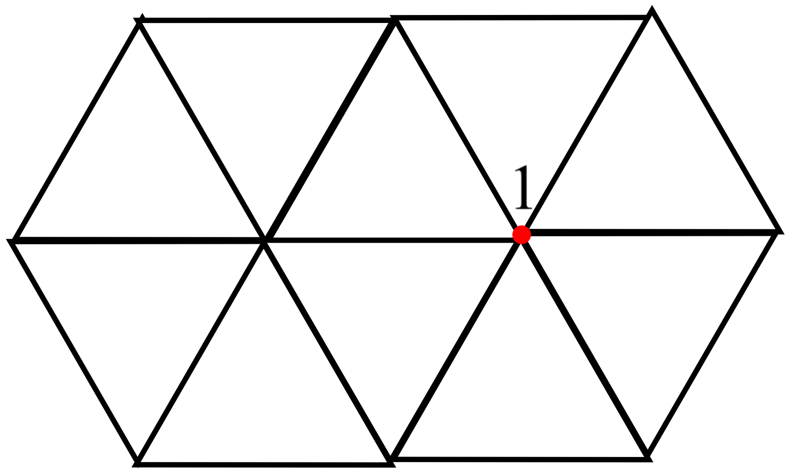
Other Subdivision Schemes

- Butterfly subdivision



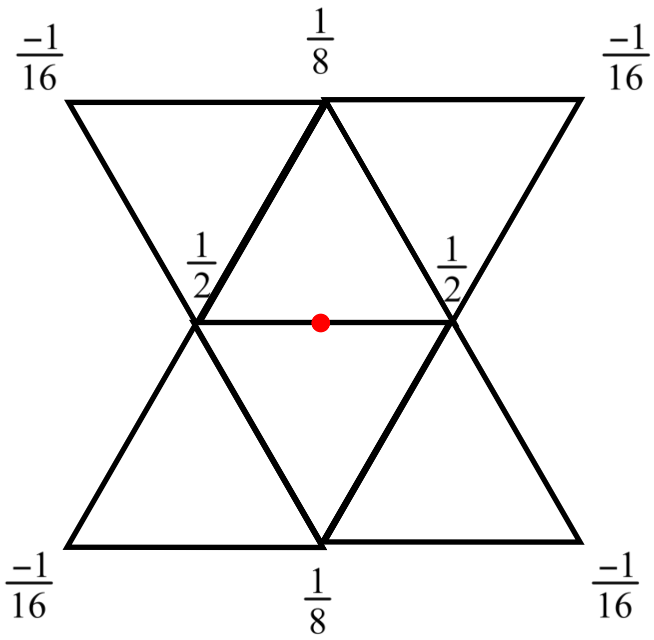
Other Subdivision Schemes

- Butterfly subdivision

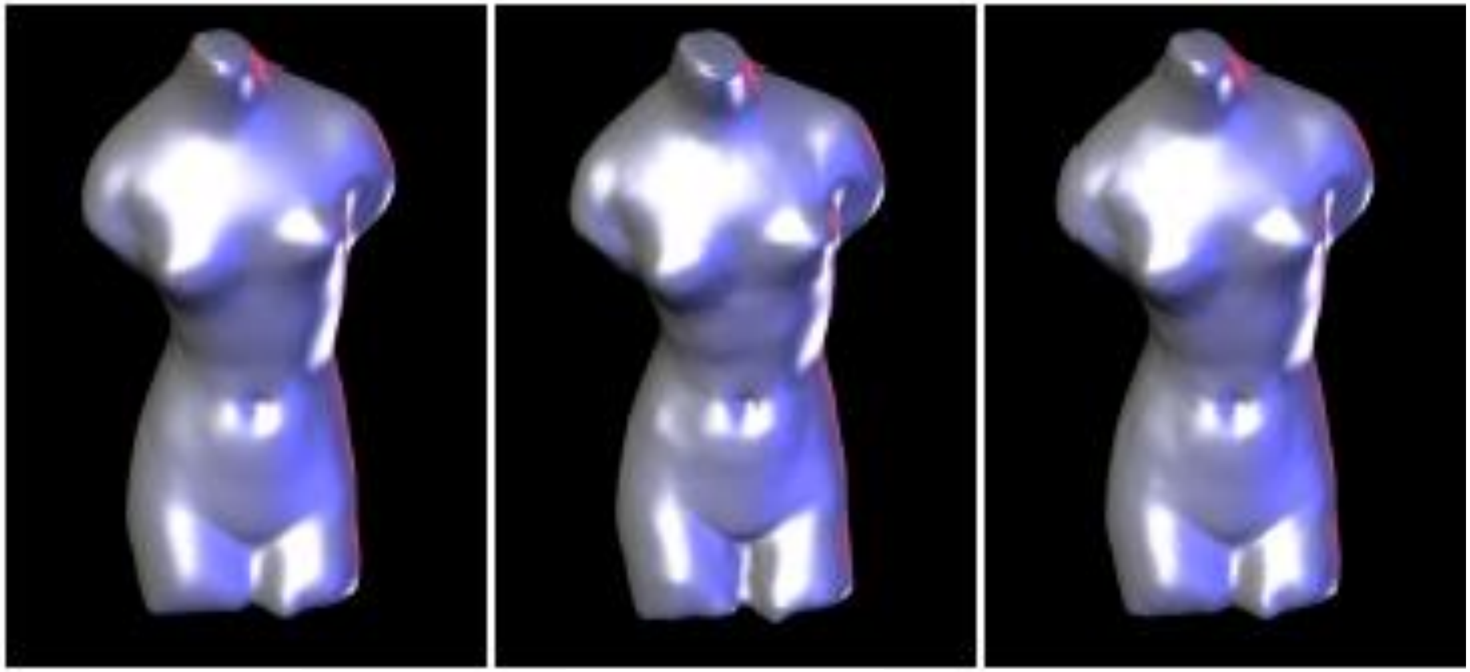


Other Subdivision Schemes

- Butterfly subdivision



Other Subdivision Schemes

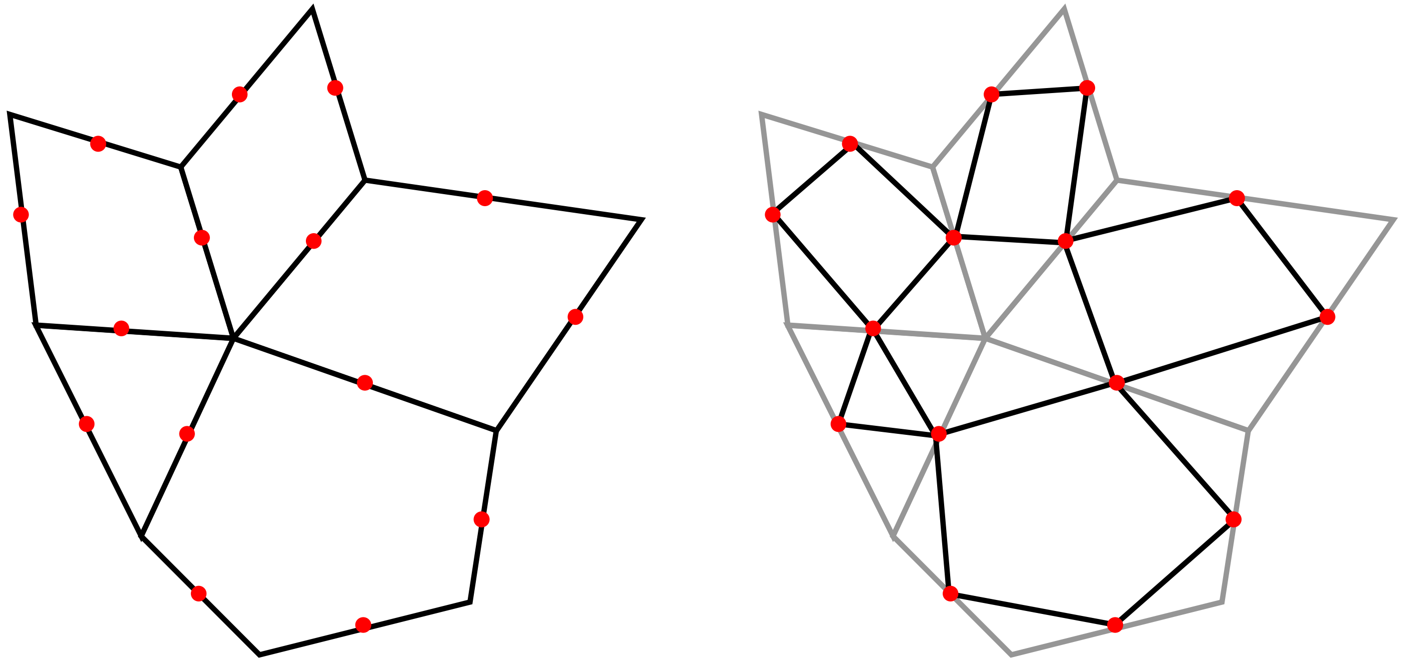


Loop Butterfly Catmull-Clark

Zorin & Schroeder
SIGGRAPH 99
Course Notes

Other Subdivision Schemes

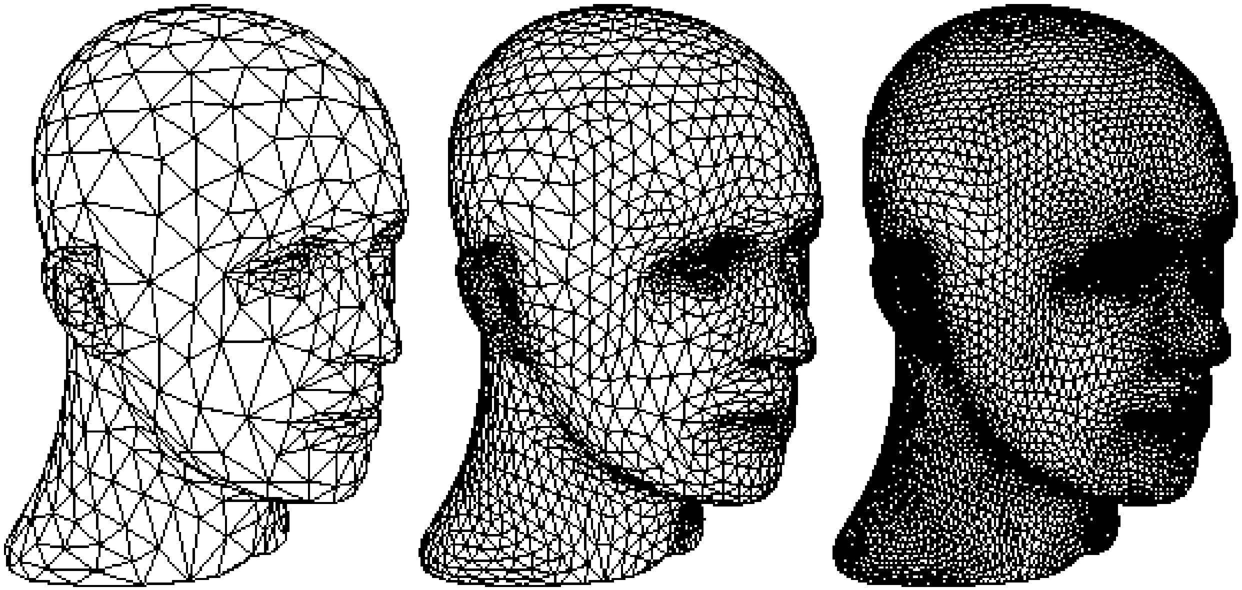
- Vertex-split subdivision
  (Doo-Sabin, Midedge, Biquartic)



One step of Midegde subdivision

Drawing Subdivision Surfaces

- Goal:
  - Draw best approximation of smooth limit surface
  - With limited triangle budget

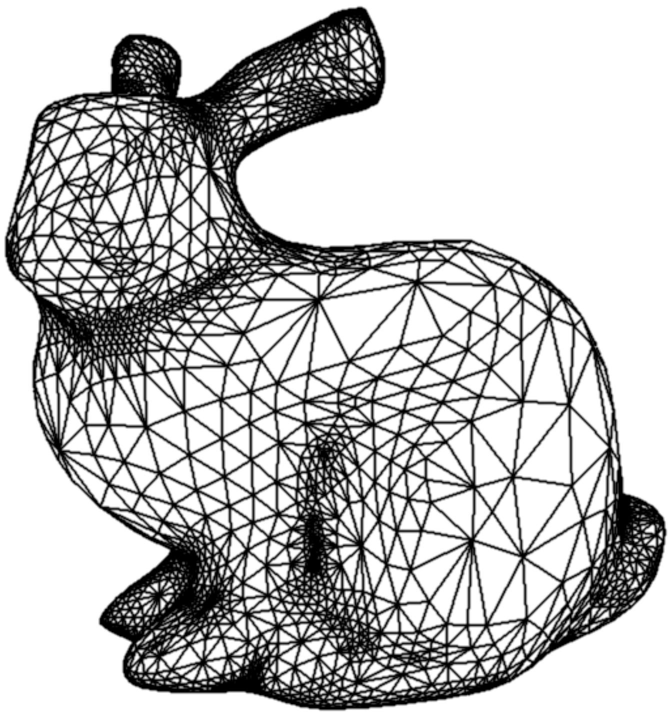


Zorin & Schroeder
SIGGRAPH 99
Course Notes

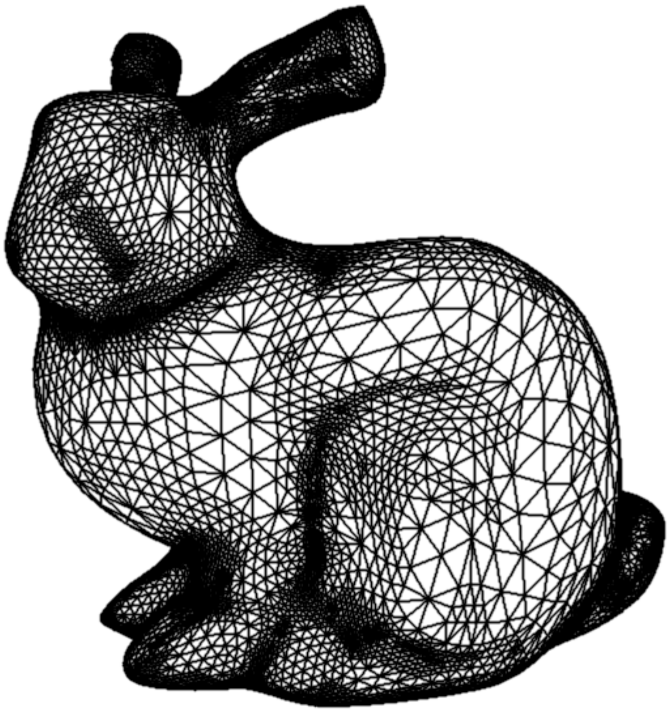
Drawing Subdivision Surfaces

- Goal:
  - Draw best approximation of smooth limit surface
  - With limited triangle budget
- Solution:
  - Stop subdivision at different levels across the surface
  - Stop-criterion depending on quality measure
- Quality of approximation can be defined by
  - Projected (screen) area of final triangles
  - Local surface curvature

Adaptive Subdivision



10072 Triangles

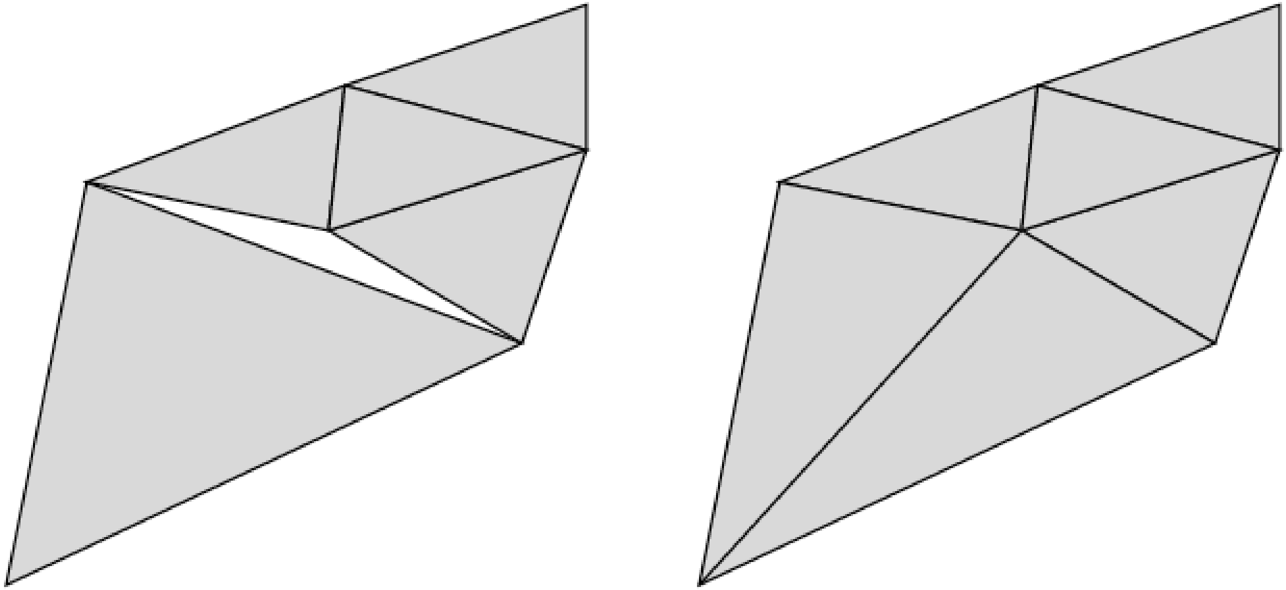


228654 Triangles

[Kobbelt 2000]

Adaptive Subdivision

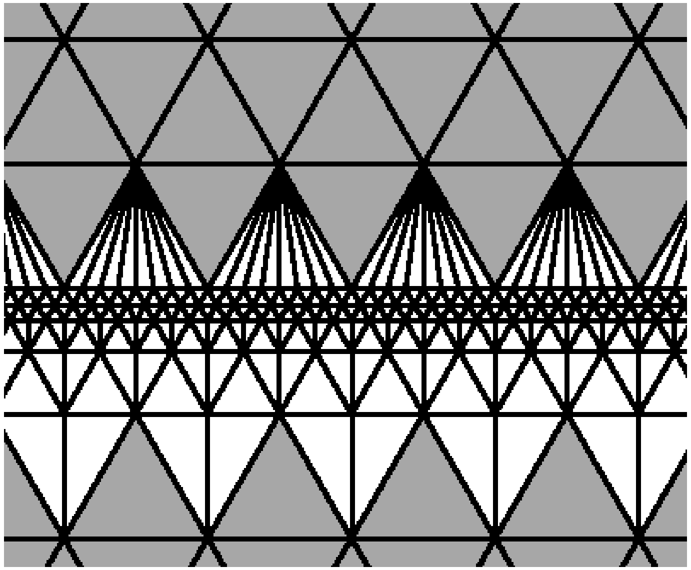
- Problem:
  - Different levels of subdivision may lead to gaps in the surface



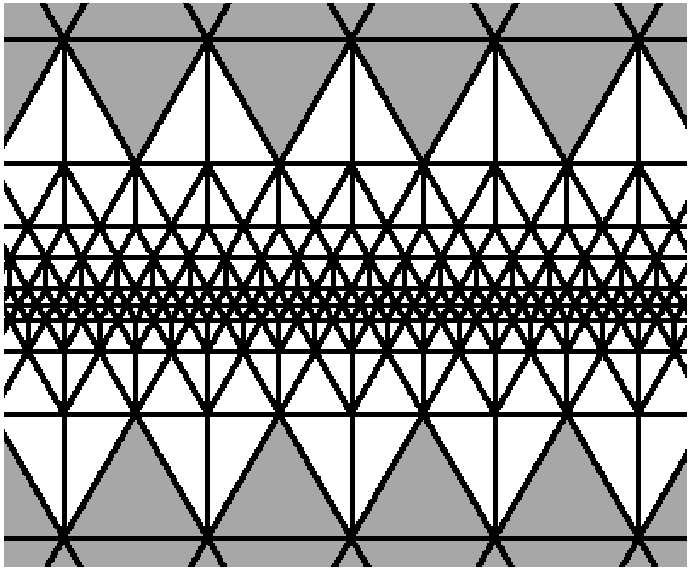
[Kobbelt 2000]

Adaptive Subdivision

- Solution:
  - Replacing incompatible coarse triangles by *triangle fan*
  - Balanced subdivision: neighboring subdivision levels must not differ by more than one



Unbalanced

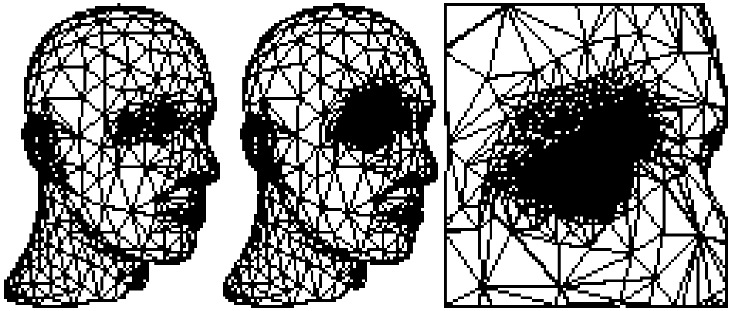


Balanced

[Kobbelt 2000]

Subdivision Surface Summary

- Advantages:
  - Simple method for describing complex surfaces
  - Relatively easy to implement
  - Arbitrary topology
  - Intuitive specification
  - Local support
  - Guaranteed continuity
  - Multiresolution

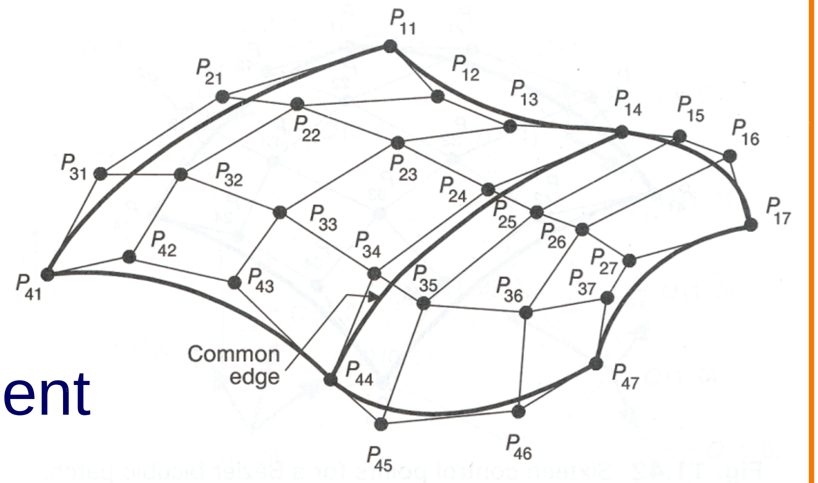


- Difficulties:
  - Parameterization
  - Intersections

Comparison

Parametric surfaces

- Provide parameterization
- More restriction on topology of control mesh
- Some require careful placement of control mesh vertices to guarantee continuity (e.g., Bezier)



Subdivision surfaces

- No parameterization
- Subdivision rules can be defined for arbitrary topologies
- Provable continuity for all placements of control mesh vertices

Comparison

Feature	Polygonal Mesh	Parametric Surface	Subdivision Surface
Accurate	No	Yes	Yes
Concise	No	Yes	Yes
Intuitive specification	No	Yes	Yes
Local support	Yes	Yes	Yes
Affine invariant	Yes	Yes	Yes
Arbitrary topology	Yes	No	Yes
Guaranteed continuity	No	Yes	Yes
Natural parameterization	No	Yes	No
Efficient display	Yes	Yes	Yes
Efficient intersections	No	No	No