CSE-170 Computer Graphics

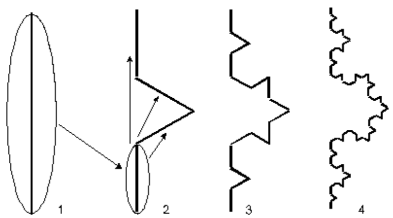
Lecture 19

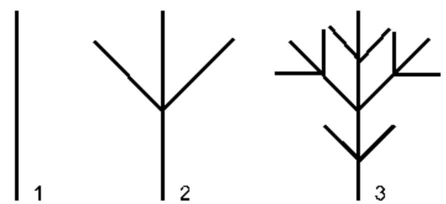
Subdivision Surfaces

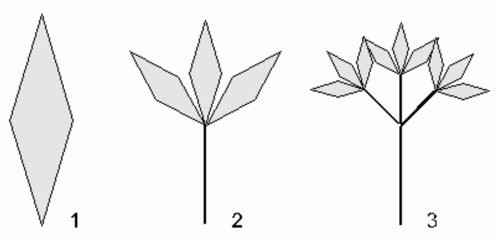
Dr. Renato Farias
rfarias2@ucmerced.edu

Curves

- Main approaches to define curves:
  - Procedural curves: fractals, subdivision rules







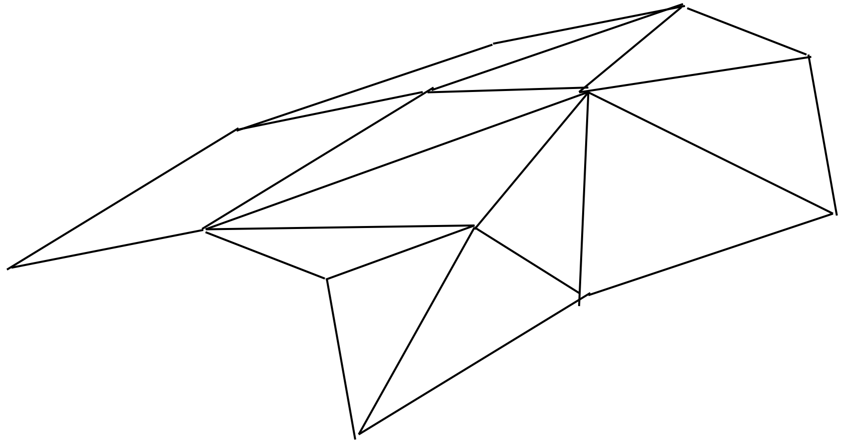
  - Parametric curves: are mappings
    - Ex.: Continuous map from 1D space to n-D space
      $f(t)=(x,y)$, ex: $f(t)=(\cos t, \sin t)$
  - Implicit curves: defined by an equation
    - Described by all points satisfying an equation
      $f(x,y)=0$ ex: $x^2+y^2-1=0$

Surfaces

- Can also be parameterized
  - With 2 parameters instead of only one
    - Recall the torus equation!
- Can be defined implicitly
  - For ex. with an implicit equation
  - We will later see: marching cubes algorithm for surface extraction
- Can also be defined procedurally
  - Several subdivision procedures possible
    - Subdivision surfaces

Subdivision Surfaces

- Surfaces based on equations are very popular for defining surfaces
  - However, parametrization requires two parameters to completely define the surface
    - Recall the torus parameterization
    - We will later also see “control grids/patches”
- What can we do for generic control grids?
  - Or control meshes?



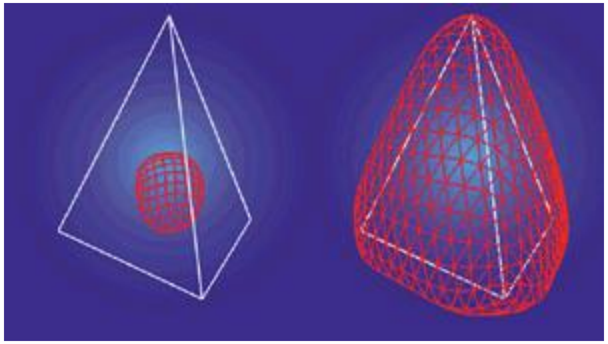
Subdivision Surfaces

- Main idea
  - Create smooth surfaces out of procedural subdivision rules applied to arbitrary meshes

- Main goal
  - Generalize surface modeling to any type of initial “control mesh”

Subdivision Surfaces

- What is a "subdivision surface"?
  – A way to define smooth surfaces as the limit of a sequence of successive refinements



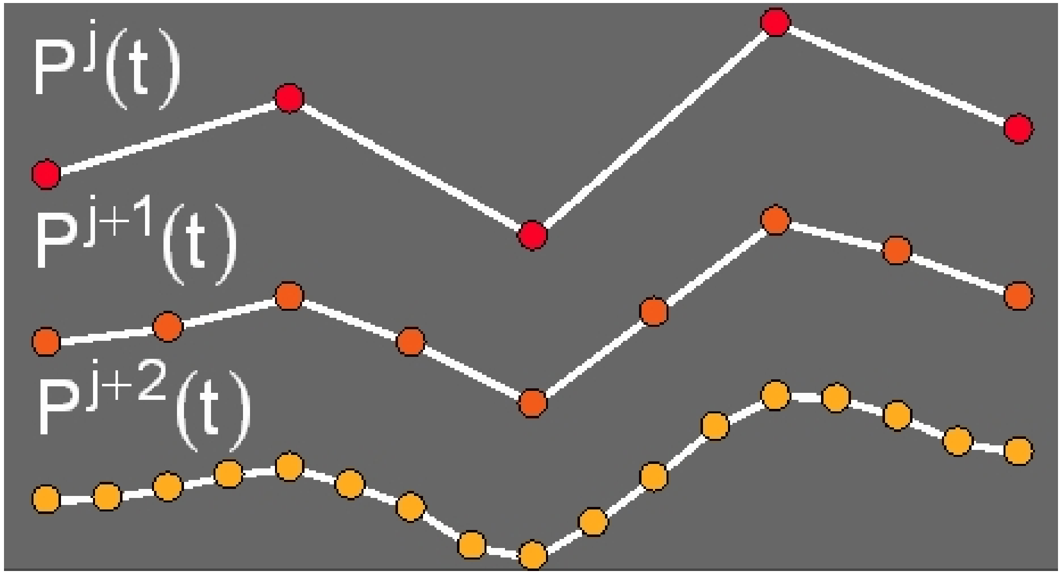
- What is a refinement?
  – A refinement has two steps:

    1st step: topological subdivision (or split)

    2nd step: geometrical rearrangement (or tweak)

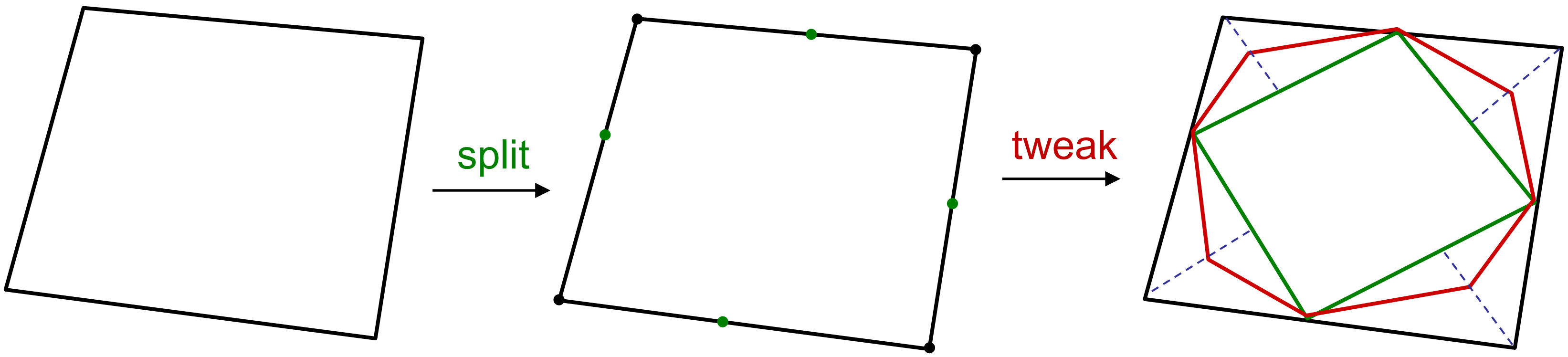
Subdivision Surfaces

- The subdivision is applied to the control points
- Think of drawing successive control polygons instead of the curve itself



B-Spline Tweak

- Example
  - It is possible to define a “quadratic B-Spline” by subdivision
    - We will see B-Splines later
  - B-Spline tweak: move the old vertices <u>halfway</u> towards the <u>average of their new neighbors</u>
    - And then repeat the process



Subdivision Surfaces

- Advantages
  - Can handle arbitrary topology (connectivity) in control meshes
    - Removing restriction to grids
  - Natural multiresolution, good for LOD

    (LOD = levels of detail)
  - Uniform representation for patches or meshes
  - Numerical stability
  - Code simplicity
- Difficulties
  - No equation defining the surface
  - No direct parameterization

Subdivision Surfaces

- Main methods (1978)
  - Doo-Sabin, for generalizing bi-quadratic patches
  - Catmull-Clark, for generalizing bi-cubic patches

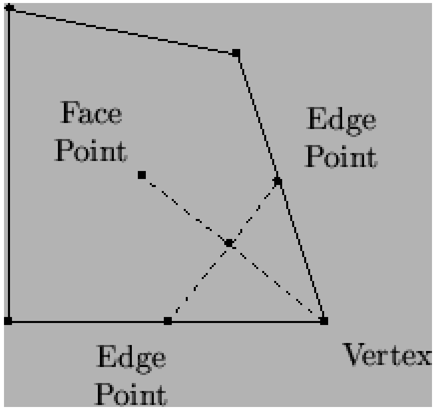
- Several other methods exist
  - Loop, Butterfly, Kobbelt, etc

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

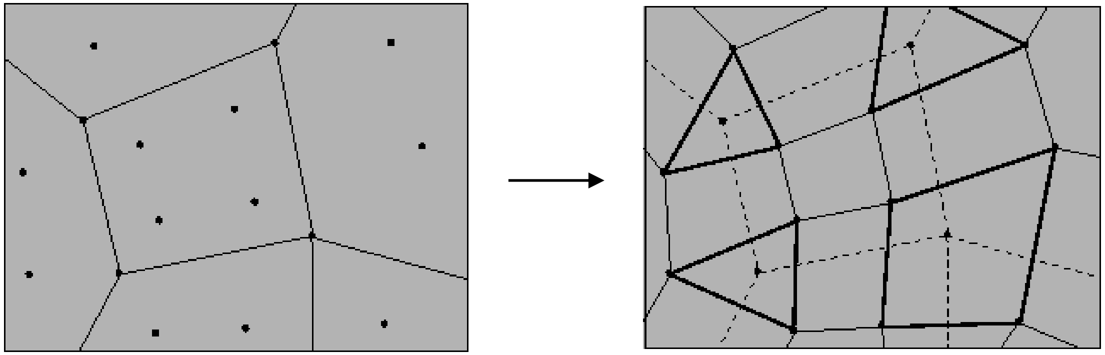
Vertex split
Doo-Sabin, Midedge (C^1)
Biquartic (C^2)

Subdivision Surfaces

- Doo-Sabin
  - For every face F, take each vertex of F, and create a new vertex as the average of:
    - the vertex
    - the face centroid
    - and the 2 centroids (midpoints) of the adjacent edges

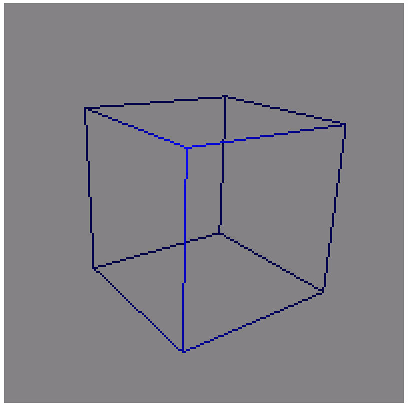


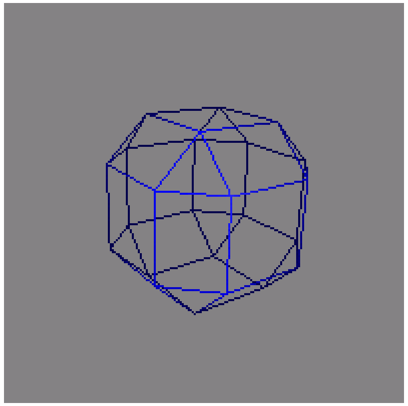
  - Connect all new vertices

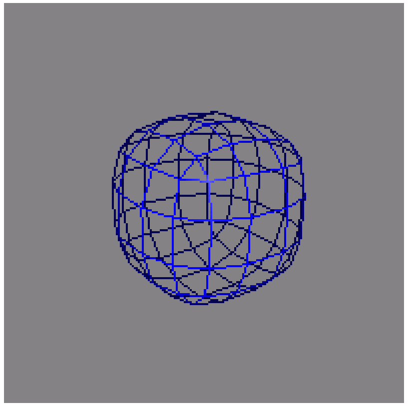


Subdivision Surfaces

- Doo-Sabin
  - Example:

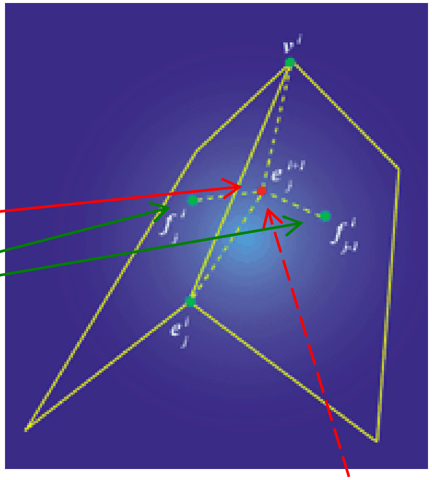




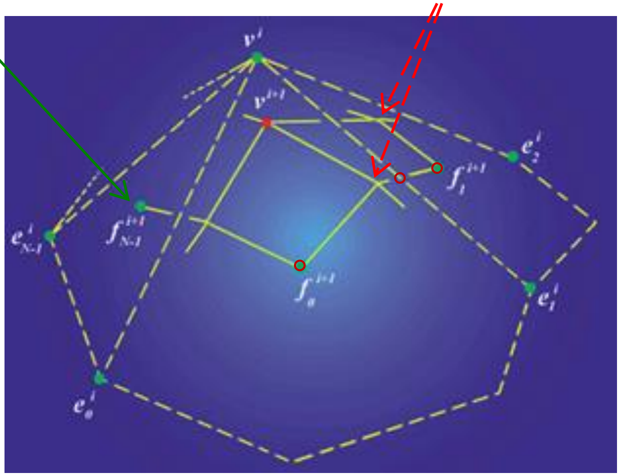


Subdivision Surfaces

- Catmull-Clark
  - Tweak edges and faces:
    - New "edge points" e: average of
      1) <u>midpoint of the original edge</u>, and
      2) the <u>centers of the two adjacent faces</u>
    - New "face points" f: face's original center (average of the vertices in the face)



edge points

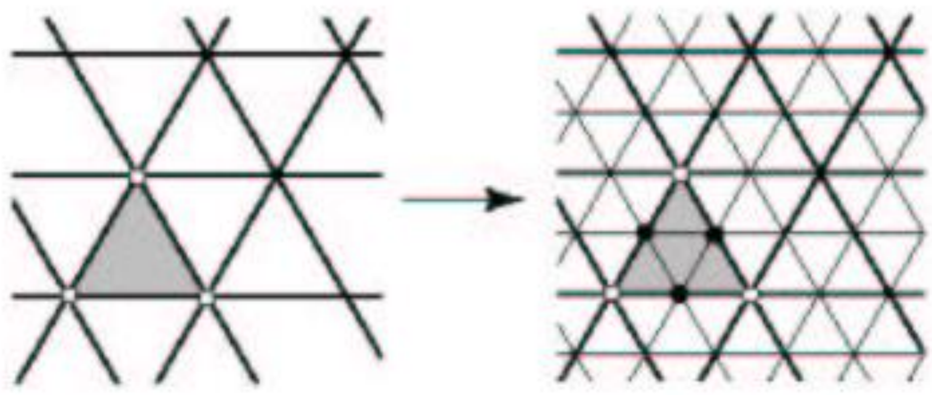


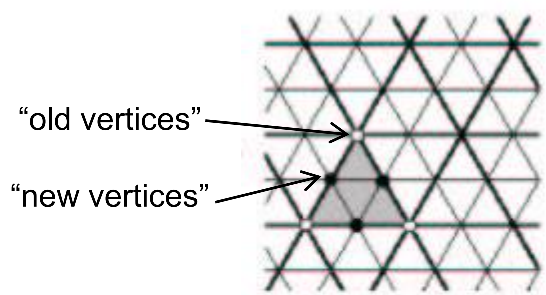
  - Tweak each old vertex to: $\frac{\mathbf{Q}}{n}+\frac{2\mathbf{R}}{n}+\frac{\mathbf{S}(n-3)}{n}$
    - S = the (old) vertex position
    - Q = average of the new face points around old vertex
    - R = average of the midpoints of the edges adjacent to S
    - n = number of edges adjacent to S

Solid line: subdivided mesh

Subdivision Surfaces

- Loop scheme
  - Split: Each edge is split in two



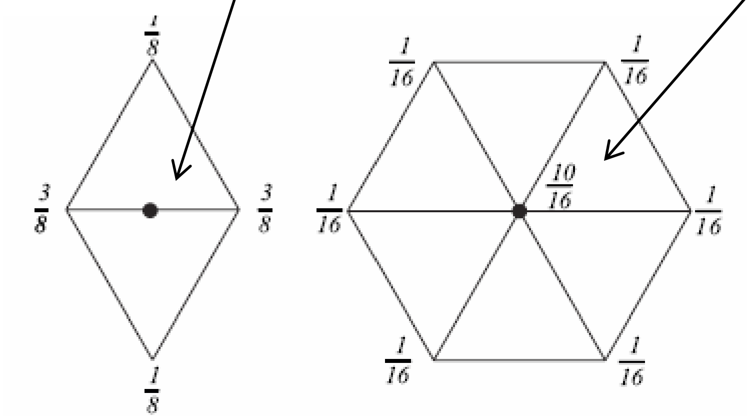


Subdivision Surfaces

- Loop scheme
  - Split: Each edge is split in two
  - Tweak: new vertices are then positioned as a weighted average of adjacent vertices

Each new vertex from an edge split will have position based on a weighted average of its neighbors

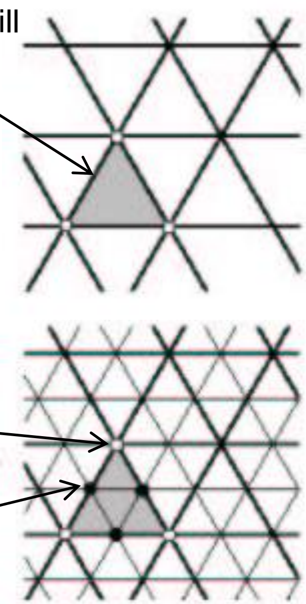
"Old vertices" are also re-positioned based on a weighted average of its neighbors



a new vertex will appear here

"old vertices"

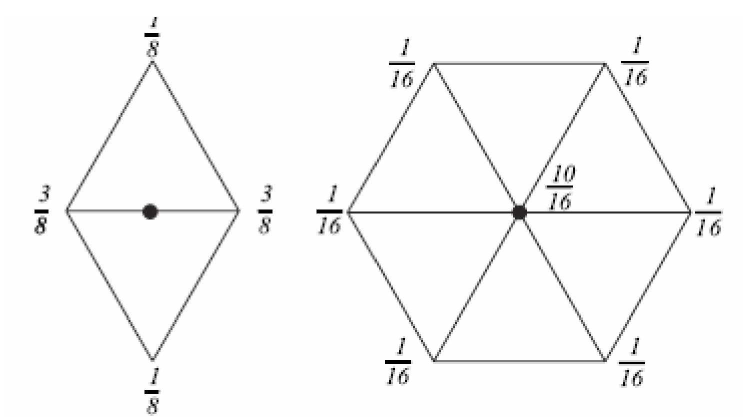
"new vertices"

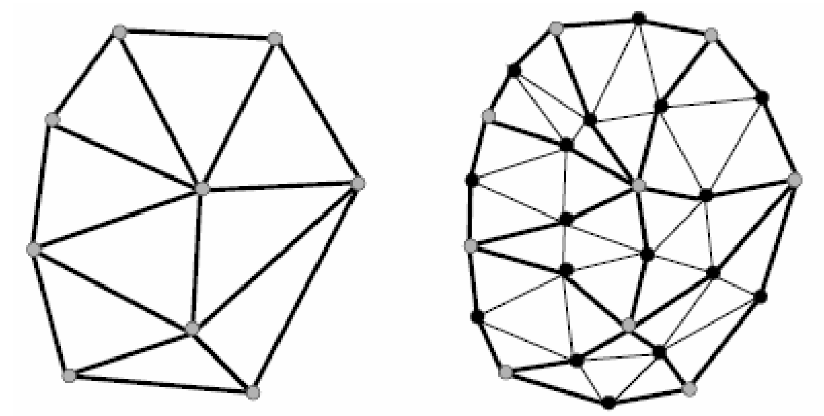


Zorin & Shroeder
SIGGRAPH'99 course notes

Subdivision Surfaces

- Loop scheme
  - Split: Each edge is split in two
  - Tweak: new vertices are then positioned as a weighted average of adjacent vertices
  - Special schemes are needed for “extraordinary” vertices (the ones with k!=6), like in borders, or for “crease edges” (k=number of edges around vertex)

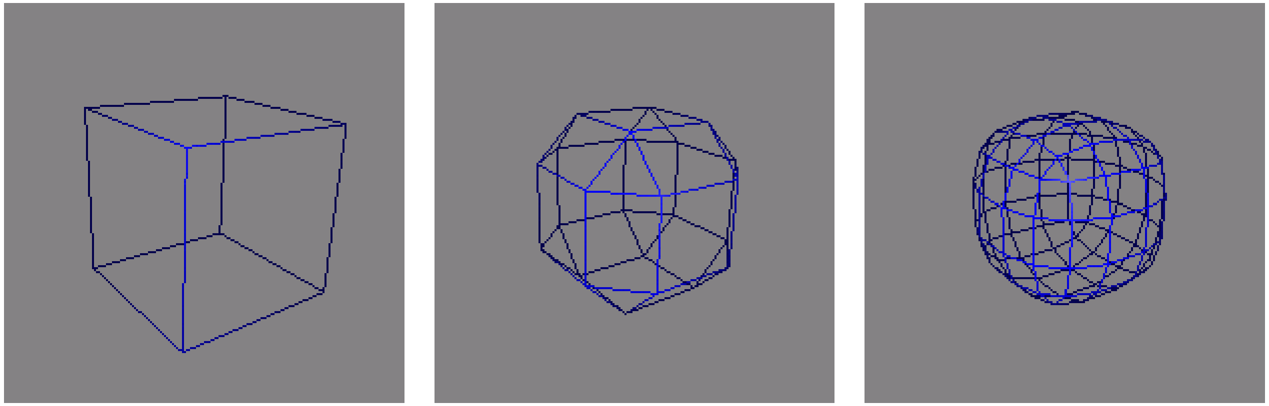




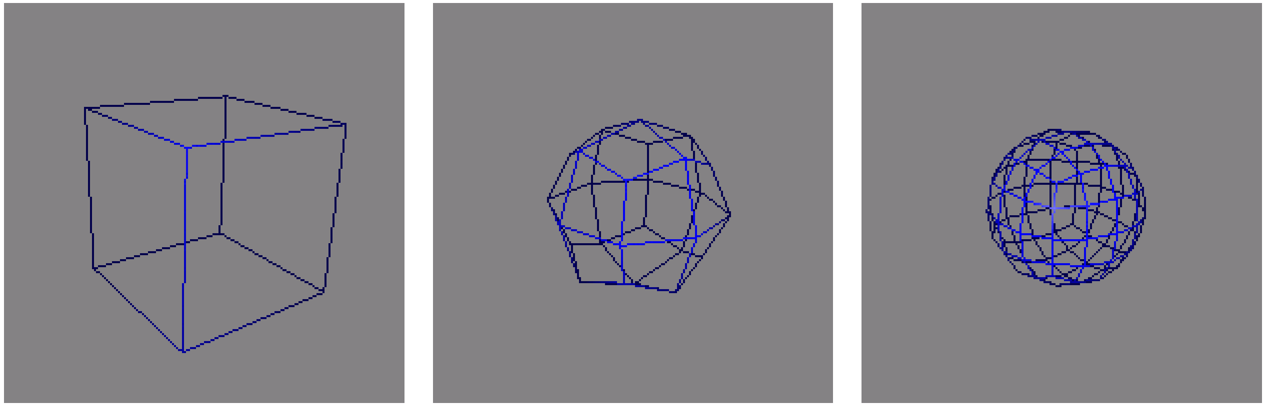
Zorin & Shroeder
SIGGRAPH’99 course notes

Subdivision Surfaces: comparison

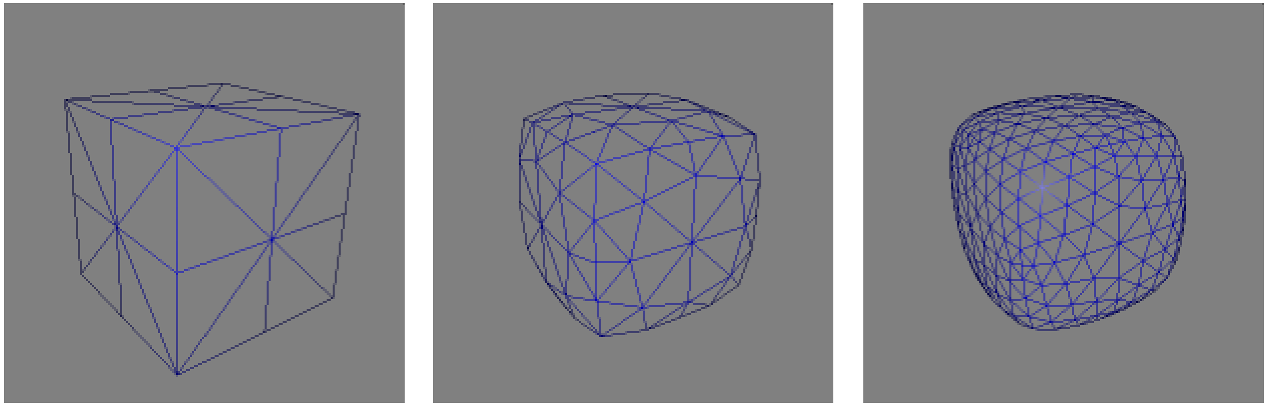
Doo-Sabin



Catmull-Clark

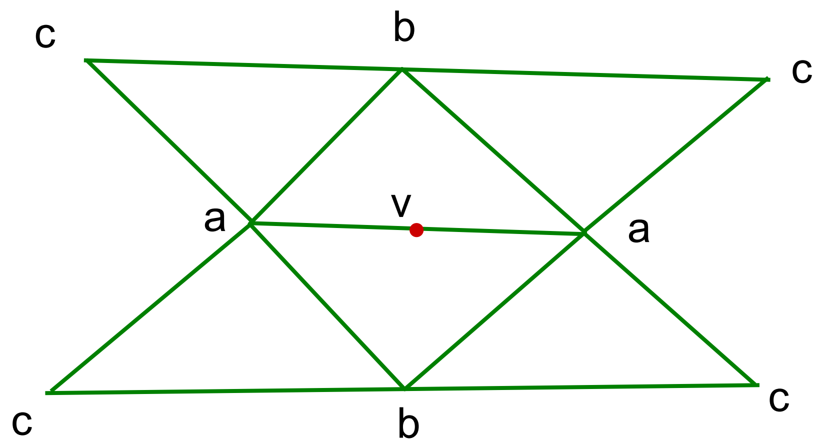


Loop



Subdivision Surfaces

- Butterfly
  - Every edge midpoint is adjacent to 2 triangles, and each of these will be adjacent to 2 others:



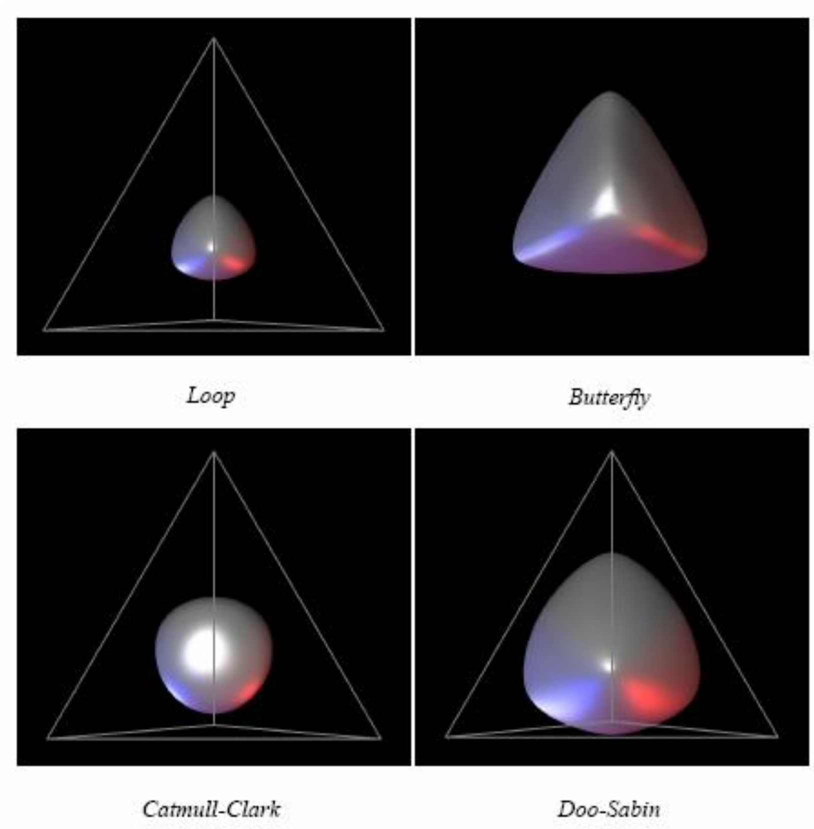
New vertex v will be a combination of the existing vertices, with weights such as:

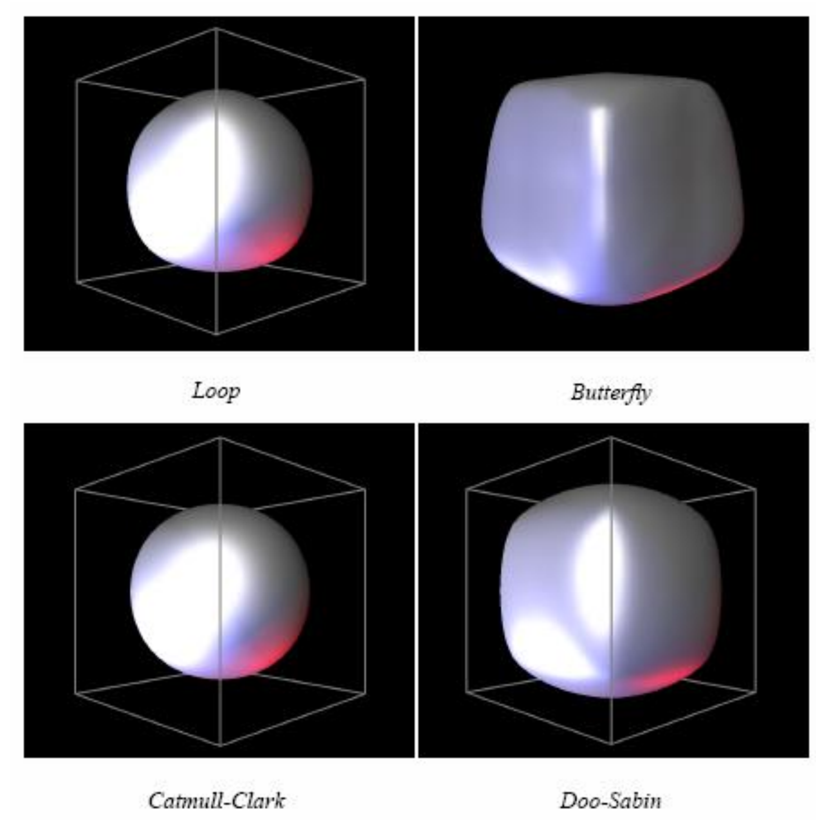
$$a:\frac{1}{2},\;\; b:\frac{1}{8}+2w,\;\; c:\frac{1}{16}-w$$

w is a tension parameter
=>there are several variants of a butterfly subdivision

Subdivision Surfaces

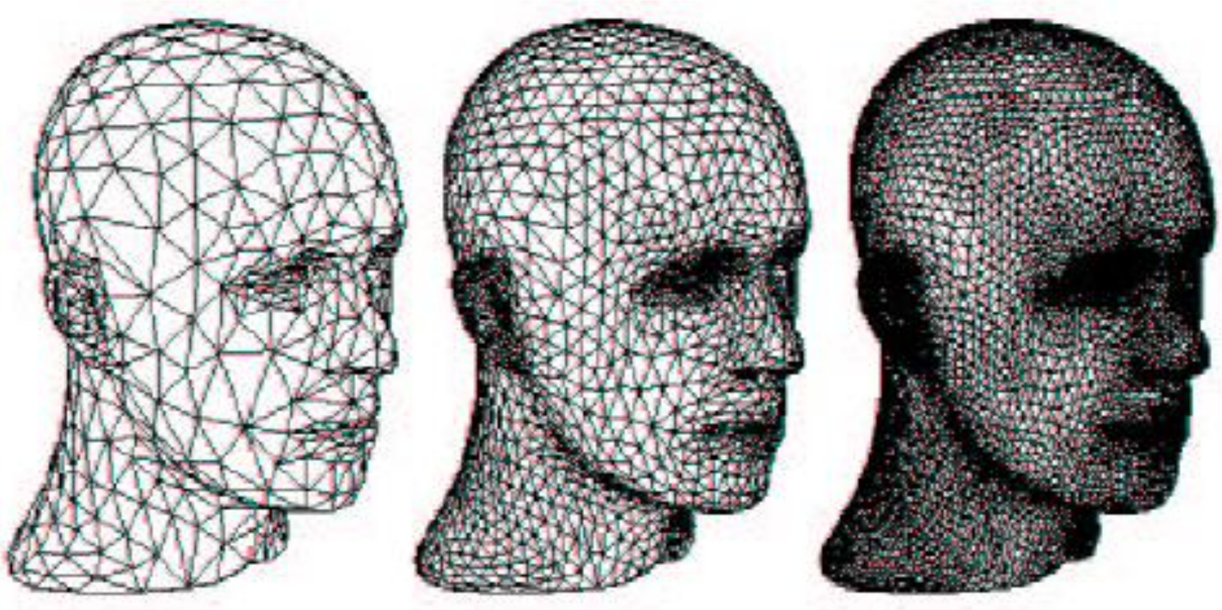
- Cube refinement example





Subdivision Surfaces

- More Examples
  - Loop



Subdivision Surfaces

- More Examples