Lecture 26A
Free-Form Surfaces
Subdivision

Shape

- Curves vs. Surfaces vs. Solids

- Surface "primitives" (spheres, cylinders, cones, ...)
- Surface "general primitives" (generalized cylinders, cones, sweeps, lofts)

- Free form surfaces

Free Form Surfaces: Approaches

Same as curves

- Parametric: $(x, y, z) = \mathbf{f}(u, v)$
- Implicit: $f(x, y, z) = 0$
- Procedural
- Subdivision

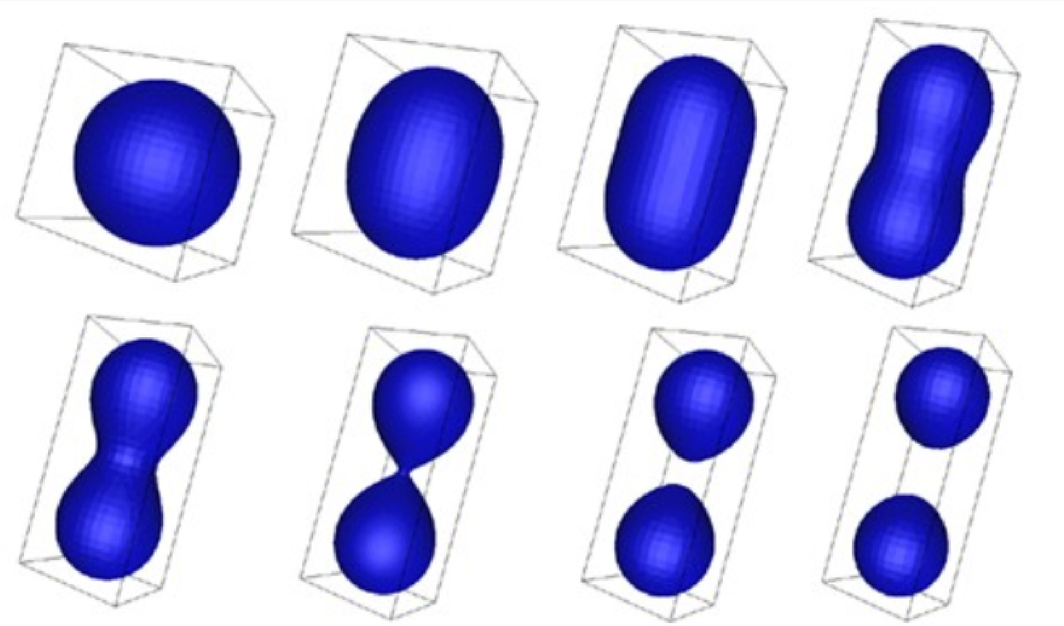
With curves, parametric is the common choice.

With surfaces, parametric is problematic so the others can be worthwhile.

Implicit Surfaces

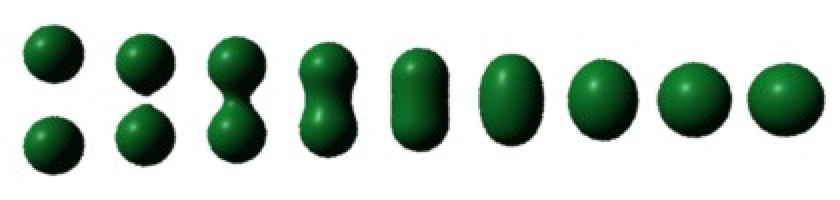
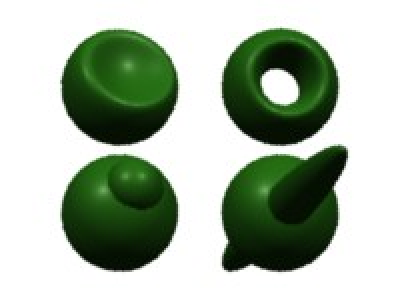
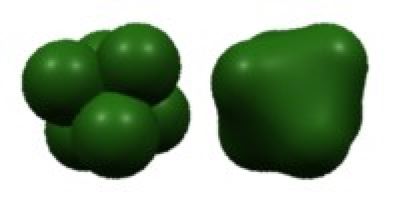
$f(x, y, z) = 0$

- inside of sphere
- inside of set of spheres
- distance to a set of points
- density (blobs)
    - (falls of to zero quickly)
- model by summing blobs



Why do we like this?

Easy to combine simple units

How to draw an implicit surface?

Need to find points on $f(x, y, z) = 0$

Free form surfaces

Is there an analog to polynomial curves?

$$f(u) \to \mathcal{R}^3$$

Cubic Polynomials

curve: $f(u) = a_0 + a_1 u^1 + a_2 u^2 + a_3 u^3$

surface: $f(u, v) = ???$

Polynomial in u and v! (tensor product)

$$
\begin{aligned}
f(u,v) = & a_{00} u^0 v^0 + a_{01} u^1 v^0 + a_{02} u^2 v^0 + a_{03} u^3 v^0 + \\
& a_{10} u^0 v^1 + a_{11} u^1 v^1 + a_{12} u^2 v^1 + a_{13} u^3 v^1 + \\
& a_{20} u^0 v^2 + a_{21} u^1 v^2 + a_{22} u^2 v^2 + a_{33} u^3 v^2 + \\
& a_{30} u^0 v^3 + a_{31} u^1 v^3 + a_{22} u^2 v^3 + a_{33} u^3 v^3
\end{aligned}
$$

Tensor Product Surface Patches

16 coefficients (control points)!

$$
\begin{aligned}
f(u,v) = & a_{00}u^0v^0 + a_{01}u^1v^0 + a_{02}u^2v^0 + a_{03}u^3v^0 + \\
& a_{10}u^0v^1 + a_{11}u^1v^1 + a_{12}u^2v^1 + a_{13}u^3v^1 + \\
& a_{20}u^0v^2 + a_{21}u^1v^2 + a_{22}u^2v^2 + a_{33}u^3v^2 + \\
& a_{30}u^0v^3 + a_{31}u^1v^3 + a_{22}u^2v^3 + a_{33}u^3v^3
\end{aligned}
$$

There are analogs to curve formulations

- Bezier, B-Spline, Interpolating, ...

Thinking about tensor product patches

f(u,v) -> x,y,z (u,v in 0,1)

Tensor Product Surfaces are Hard!

How to connect two patches?

- Continuity
- Stitching together

How to cut a patch?

- Make a Hole?
- Make an edge? (attachment)

How about non-square domains?

- inconvenient stretching?
- different topology?

What do we do instead?

Subdivision: Motivation

Polynomial Surfaces Are Challenging

f(u,v) -> x,y,z

- What if the patches aren't square?
- How do we connect them? (for smoothness)
- How do we cut holes in them?
- How do we stitch them together?

Subdivision: Intuitions from 2D

- Start with a set of (points) line segments

- Add new points / move old points
- Divide segments into more segments

- Repeat
  - until good enough
  - infinitely many times

Design so it converges to a smooth curve

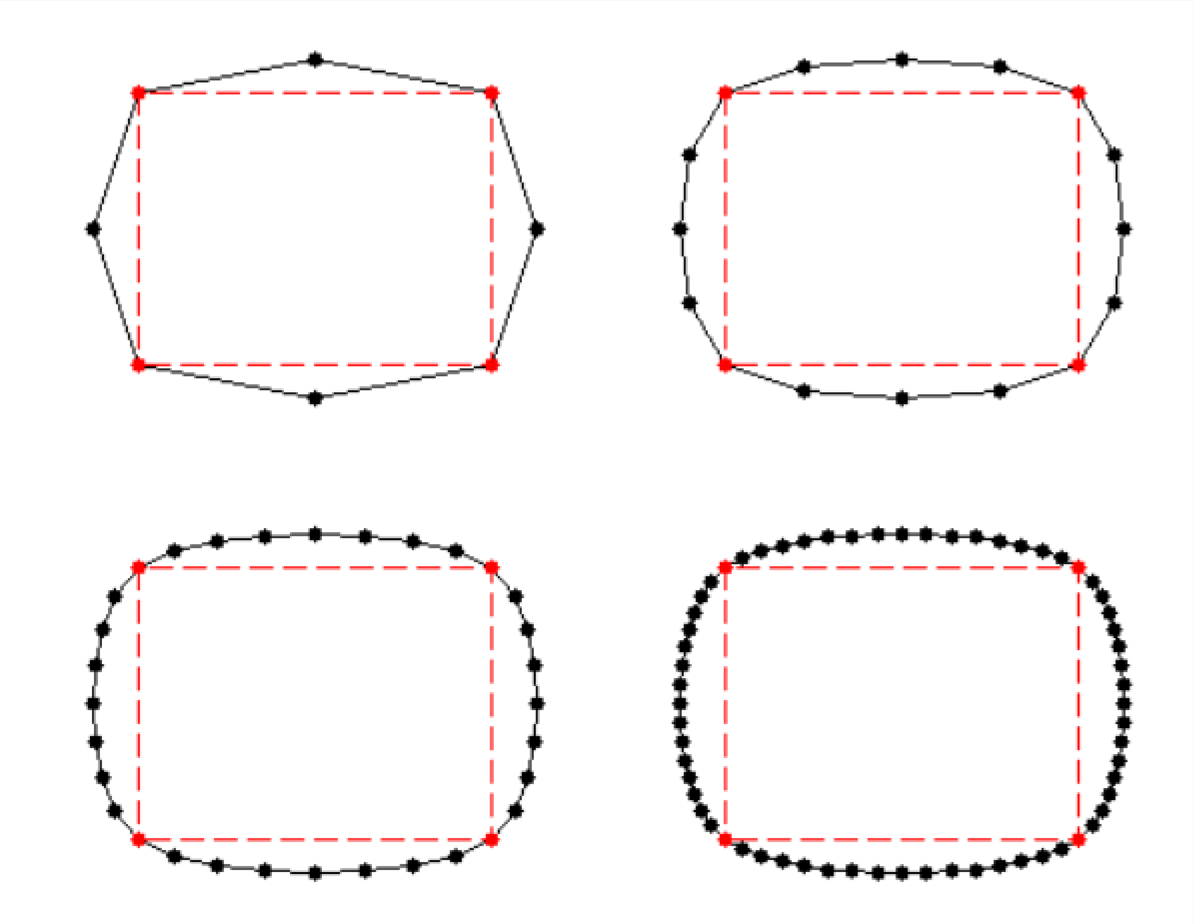
Example 1: Dyn/Levin/Gregory

4 point scheme - each new point looks at 4 neighbors

[- $\frac{1}{16}$, $\frac{1}{2} + \frac{1}{16}$, $\frac{1}{2} + \frac{1}{16}$, - $\frac{1}{16}$]

more generally [$-w$, $\frac{1}{2} + w$, $\frac{1}{2} + w$, - $\frac{1}{16} - w$]

Each time it gets smoother…



Infinitely many times?

Converges to a cubic spline!

(you can read the proof)

Note: Interpolation

Original points continue - forever

Example 2: Not interpolating

Chakin Corner Cutting (from lecture 11)

- each corner -> 2 points (1/4 from edge)
- each segment cut at (1/4, 3/4)

Converges to quadratic B-Spline

In 3D

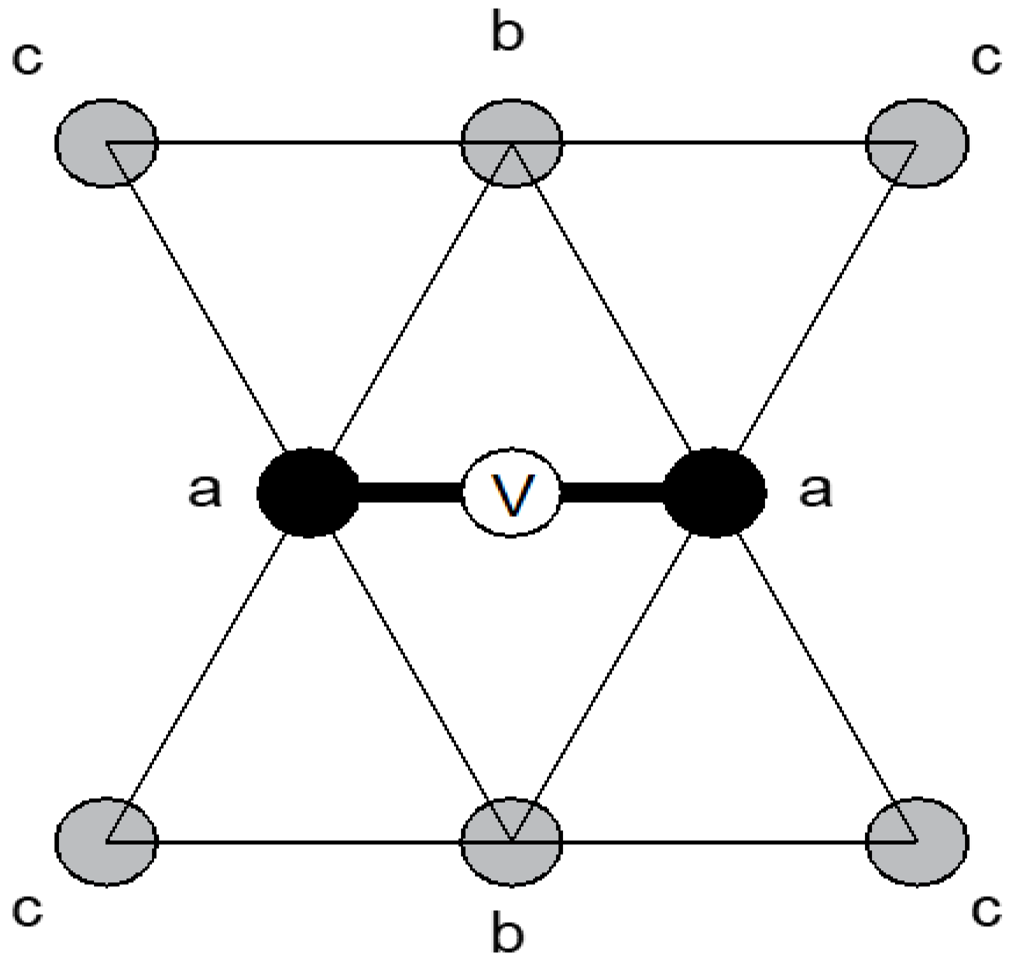
- Cut each triangle into new triangles
  - place the new vertices
  - move the old vertices (non-interpolating)

Dividing triangles

Standard (4-way) scheme

3-way scheme

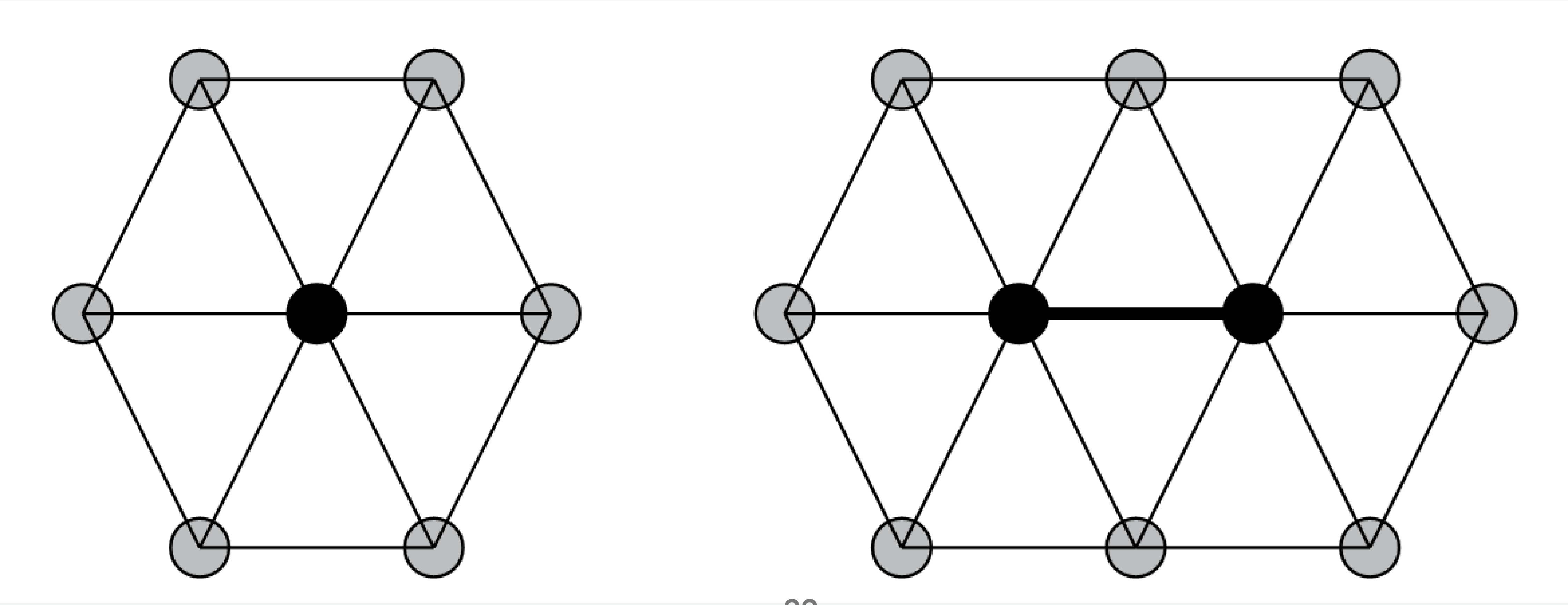
Butterfly



$$v = 1/2\,a + 1/8\,b - 1/16\,c$$

Uniform Meshes

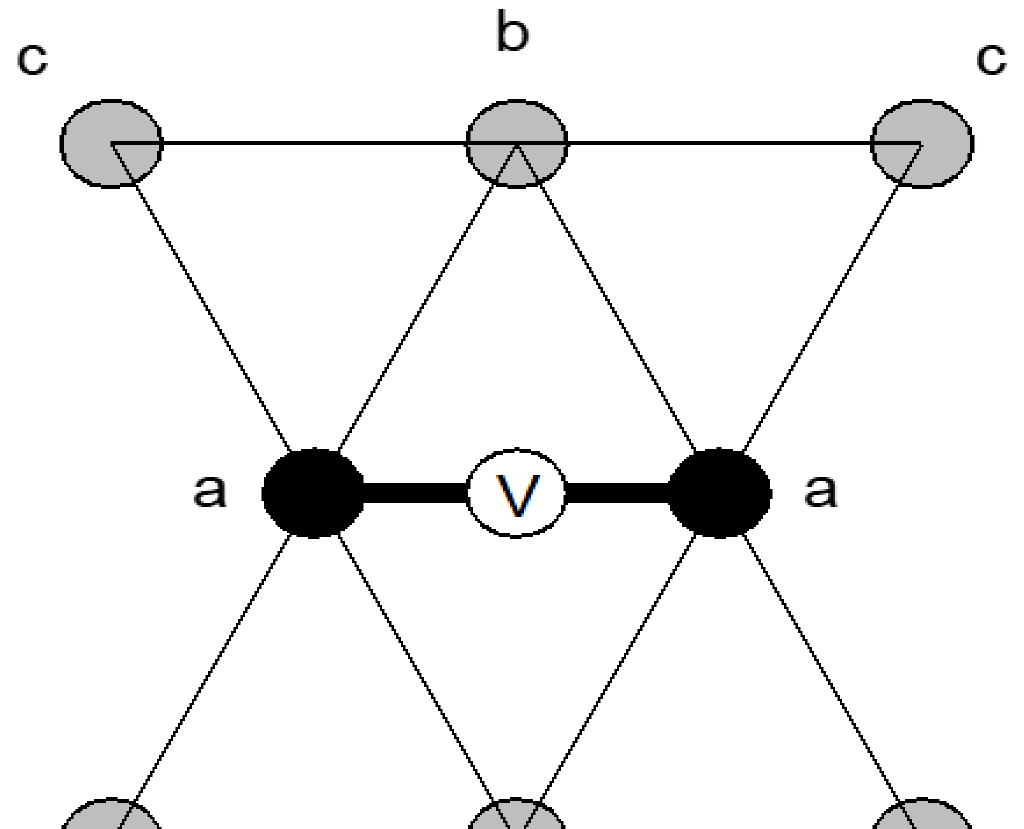
Ordinary and Extra-Ordinary Points



Butterfly

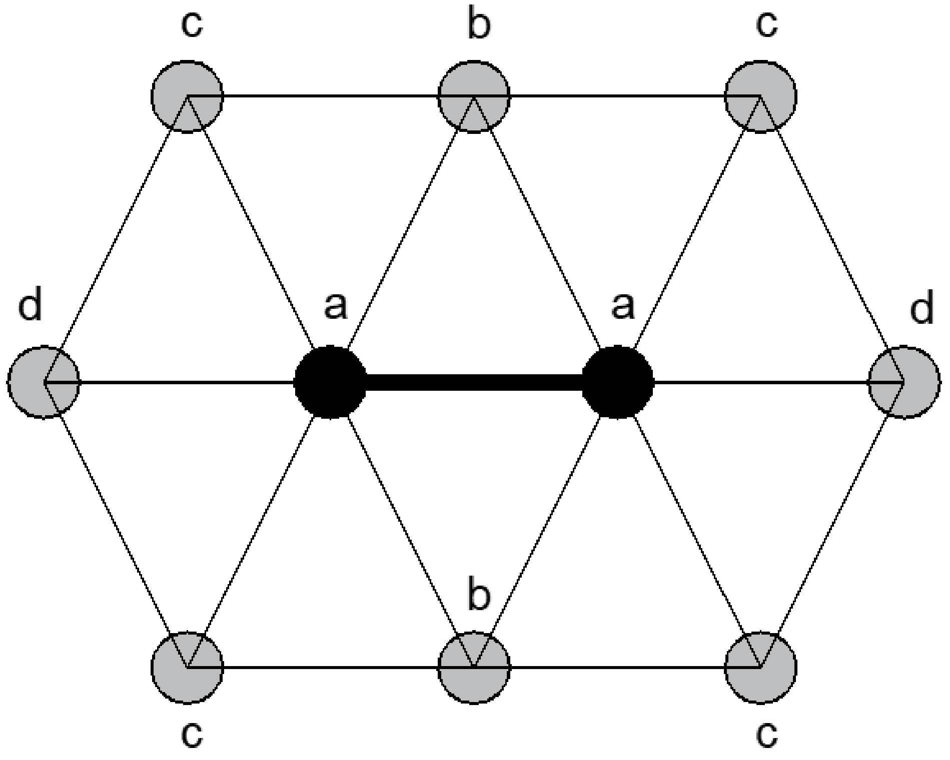
C(1) almost everywhere

Special rules for extra-ordinary points



$$v = 1/2\,a + 1/8\,b - 1/16\,c$$

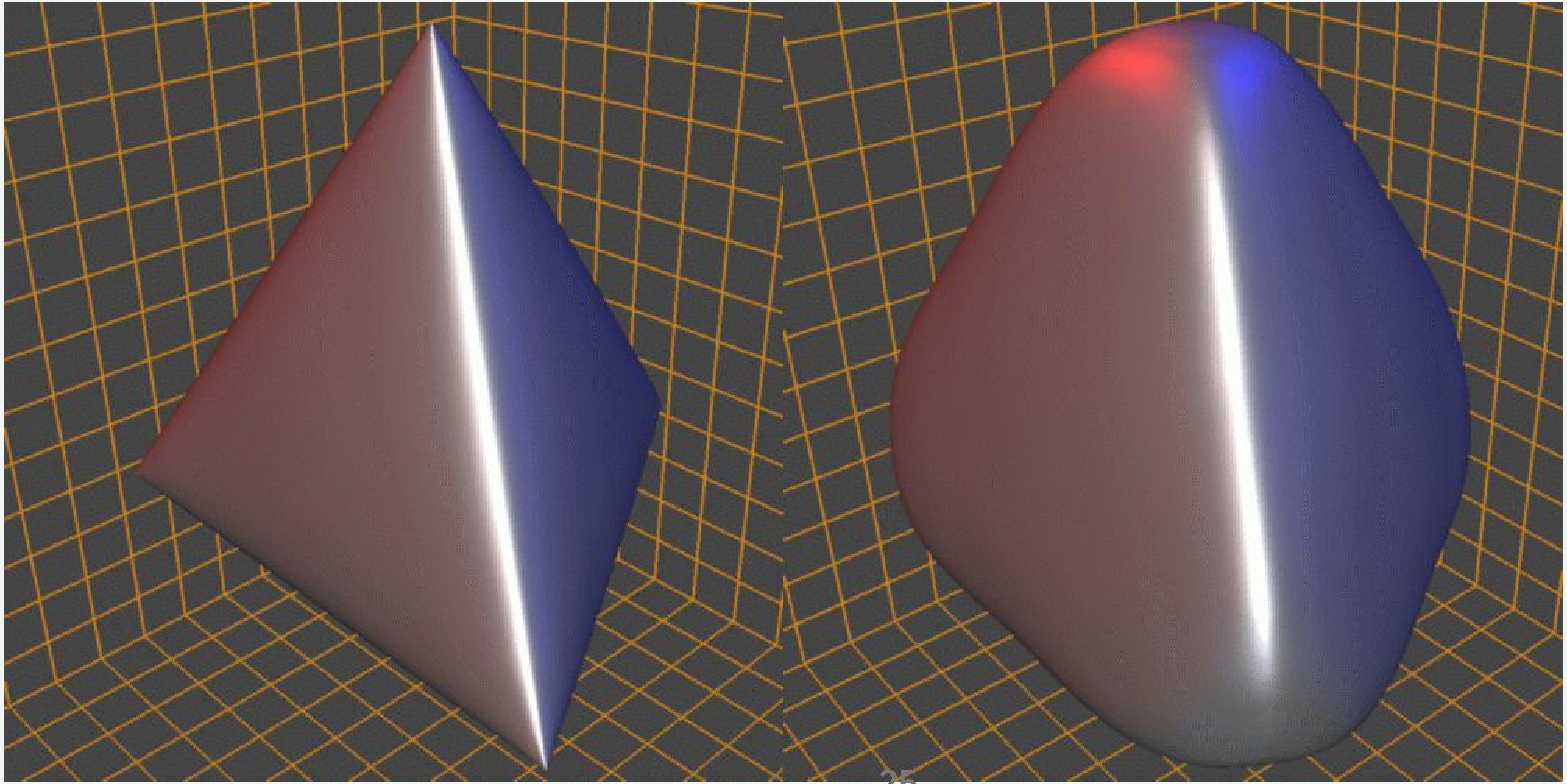
Modified Butterfly



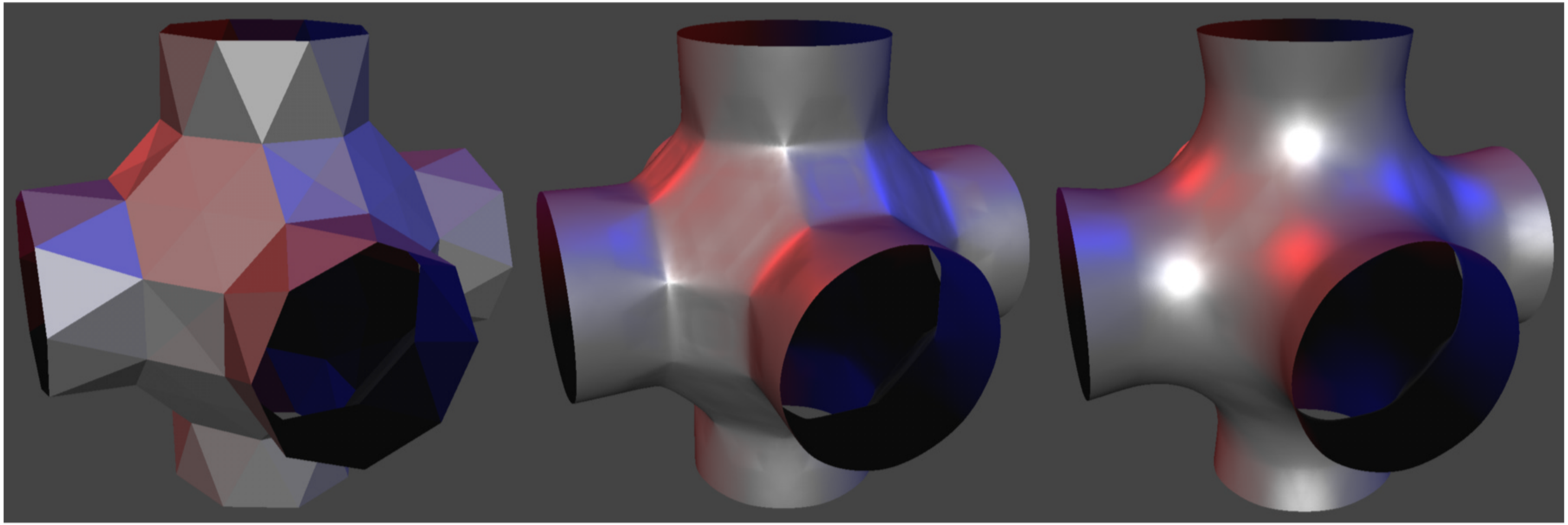
$$\mathbf{v} = (1/2-w)\,\mathbf{a} + (1/8+2w)\,\mathbf{b} - (1/16-w)\,\mathbf{c} + w\,\mathbf{d}$$

tension parameter w
sum over all 10 neighbors

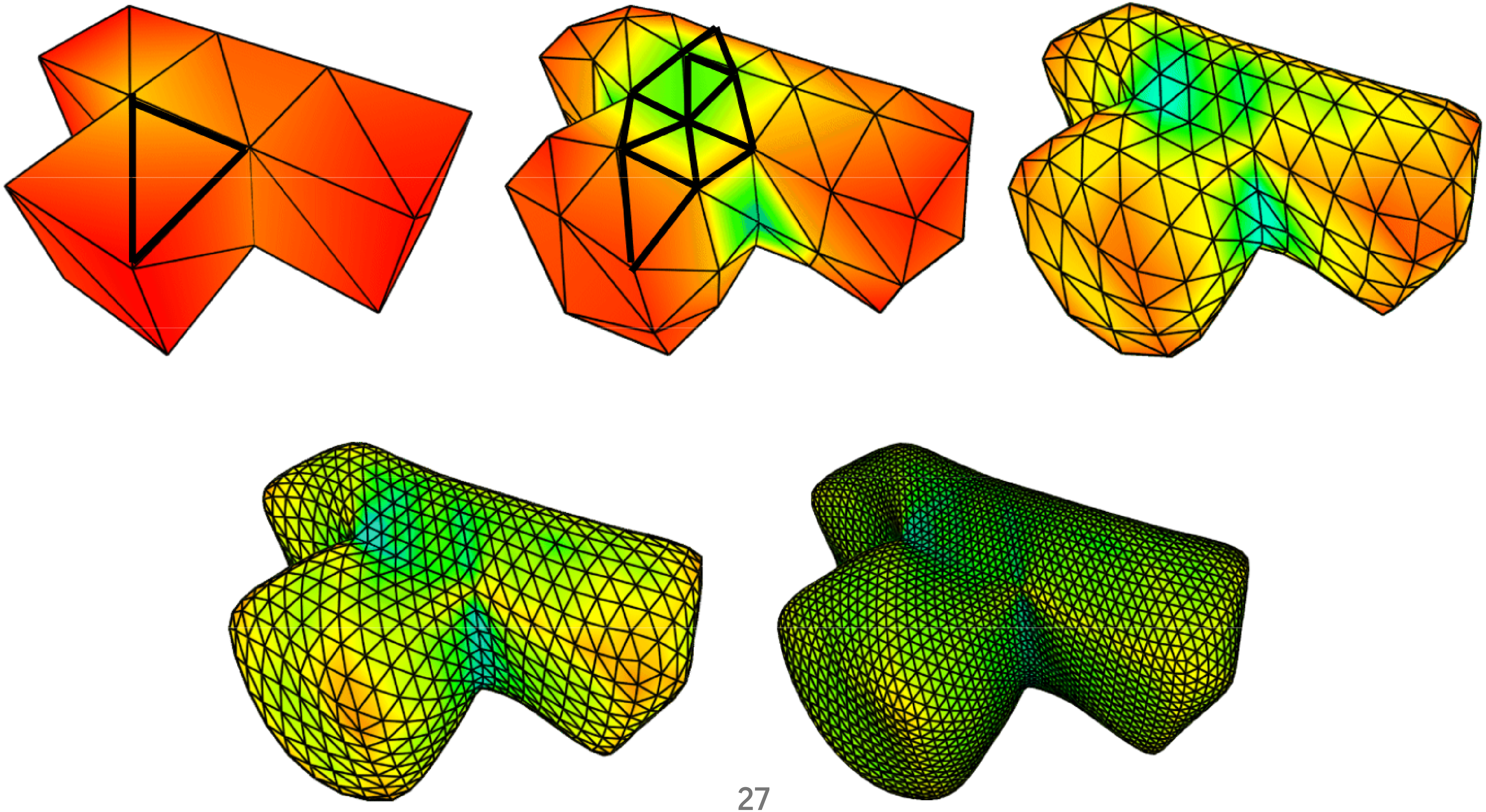
Tension



Butterfly vs. Modified



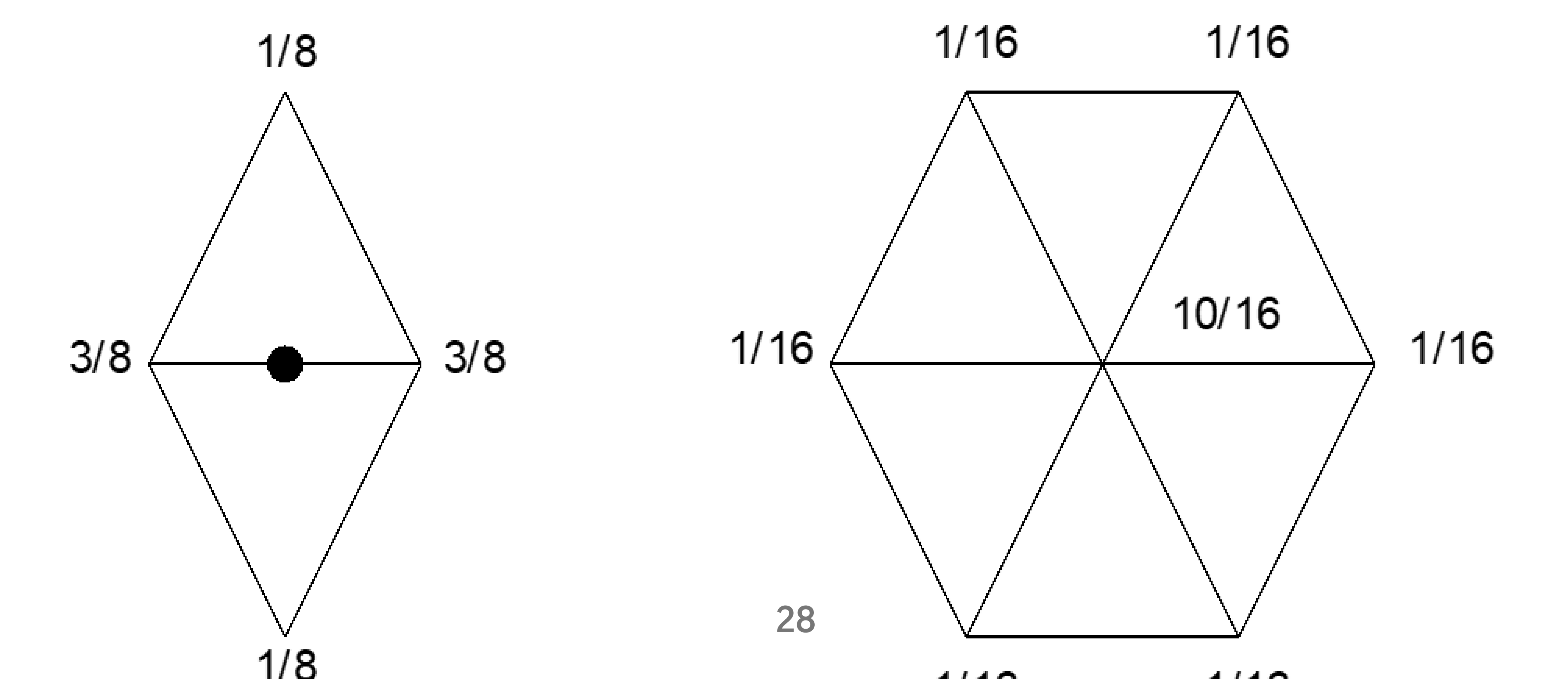
Initial mesh Butterfly scheme interpolation Modified Butterfly interpolation



http://graphics.stanford.edu/courses/cs468-10-

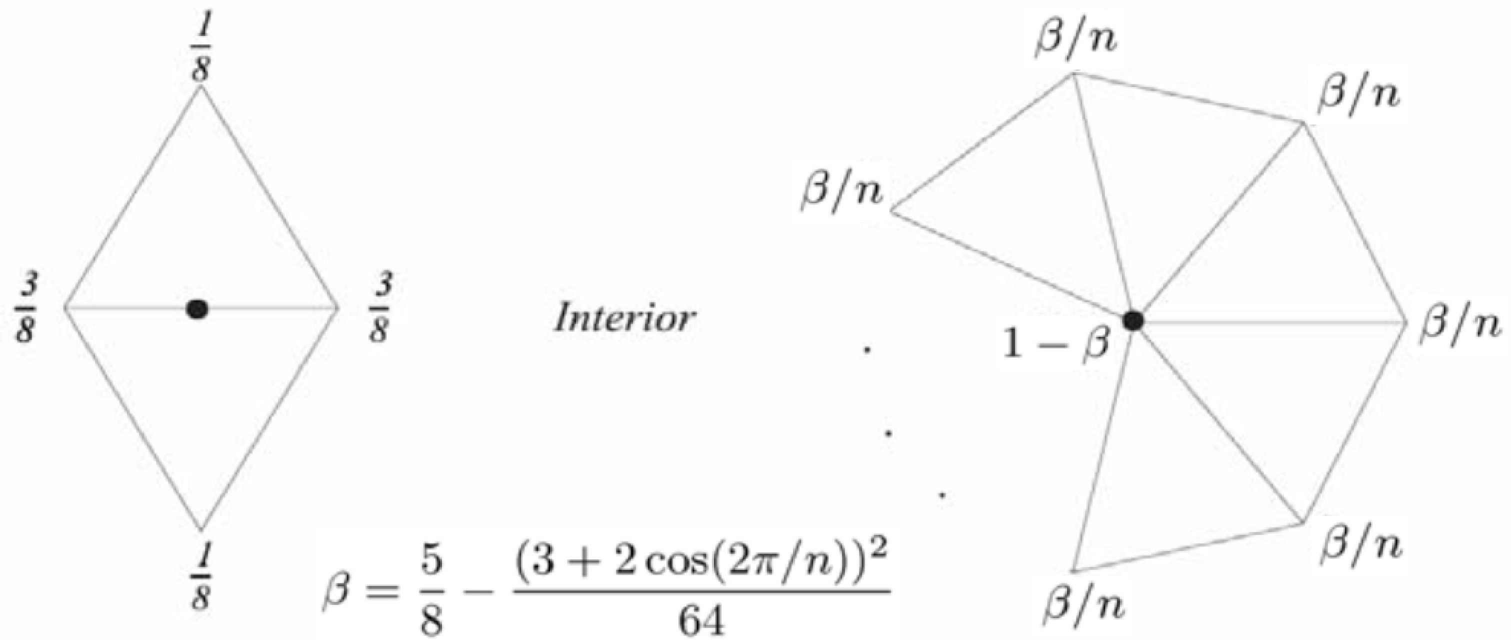
Loop Scheme

- New points split edges
- Old points moved to smooth



Loop Rules - General (irregular)

Full Loop rules (triangle mesh)

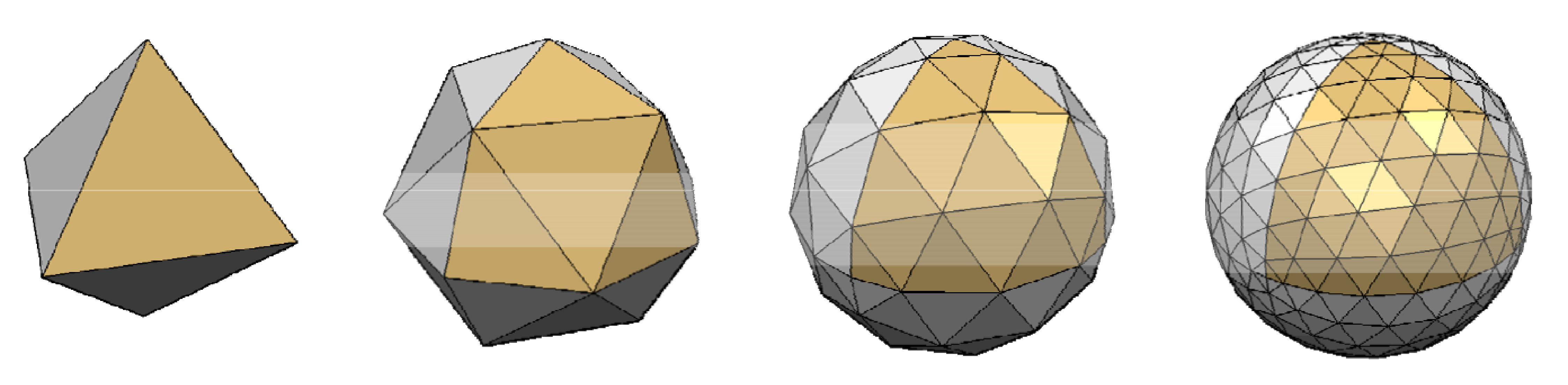


$$\beta = \frac{5}{8} - \frac{(3 + 2\cos(2\pi/n))^2}{64}$$

Loop Rules - Boundaries

- new points half way
- old points 1/8 3/4 1/8
- edges only depend on edges

Loop Example



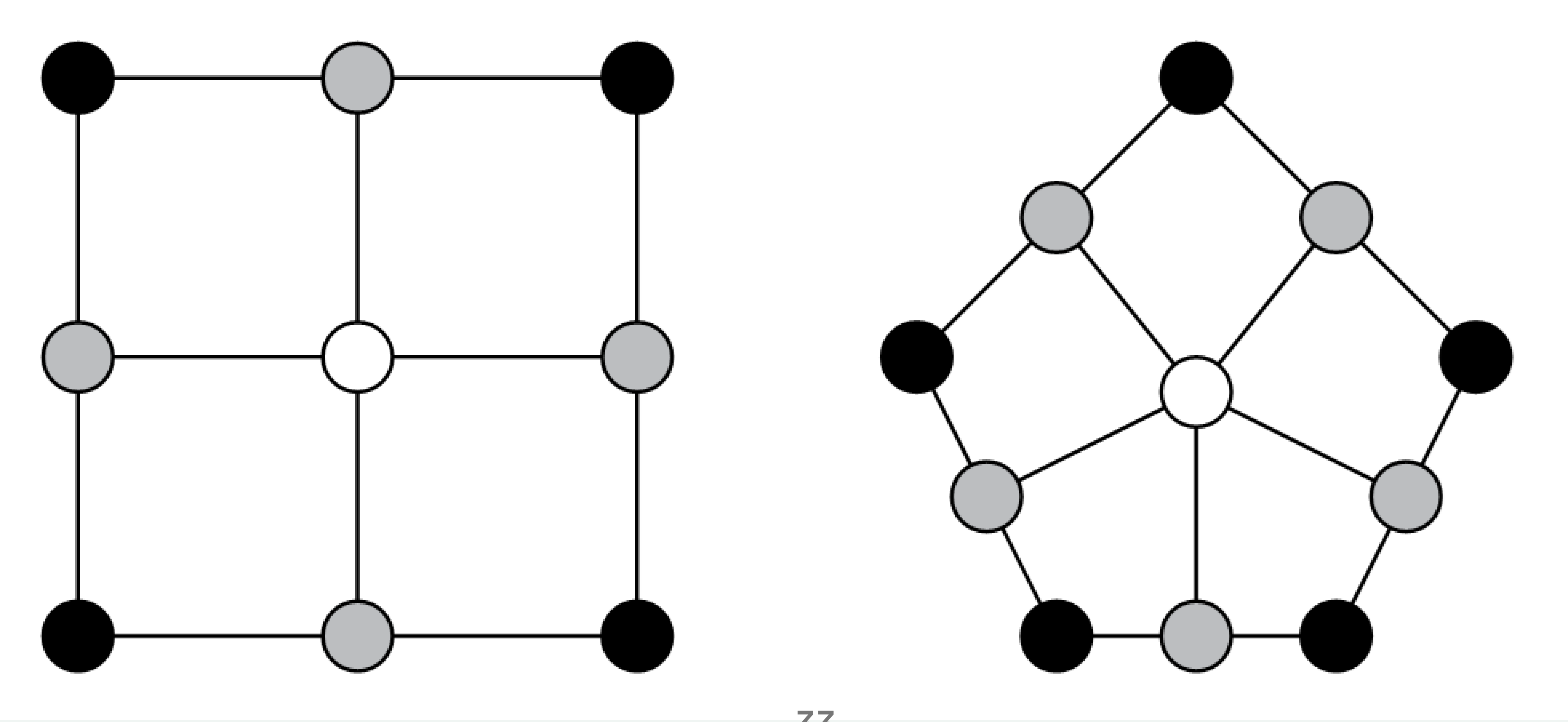
http://graphics.stanford.edu/courses/cs468-10-fall/LectureSlides/10_Subdivision.pdf

In the limit?

- Each iteration it gets smoother
- In the limit its a spline patch
- Can compute where each point will go

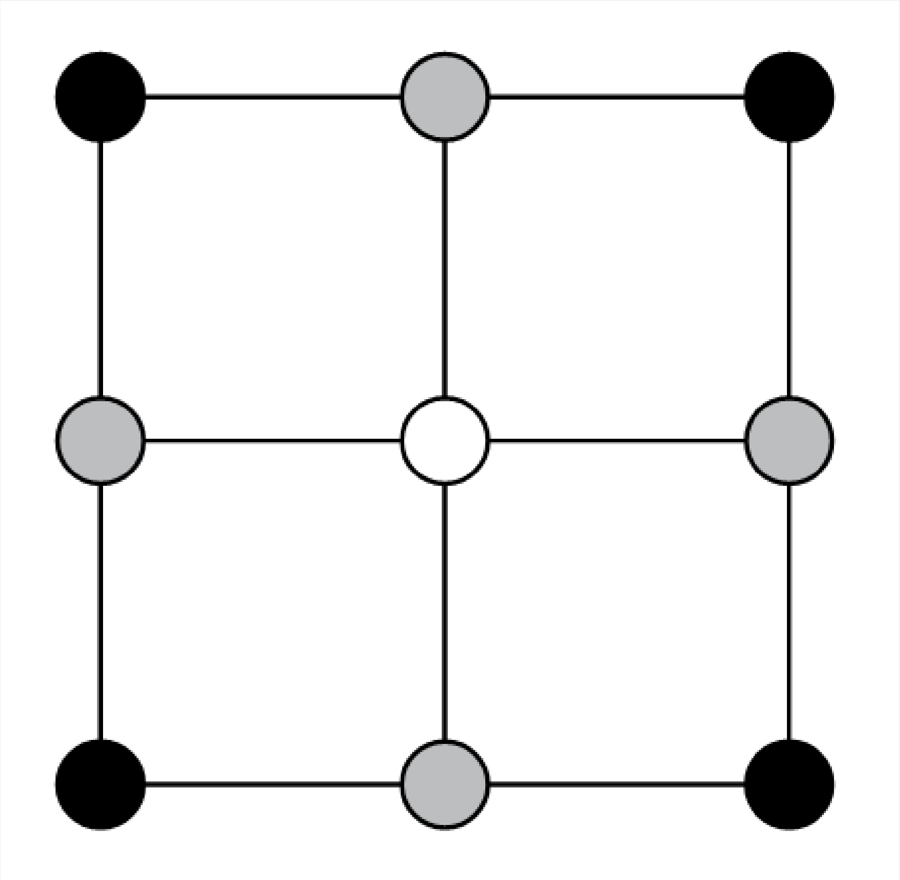
Catmull-Clark

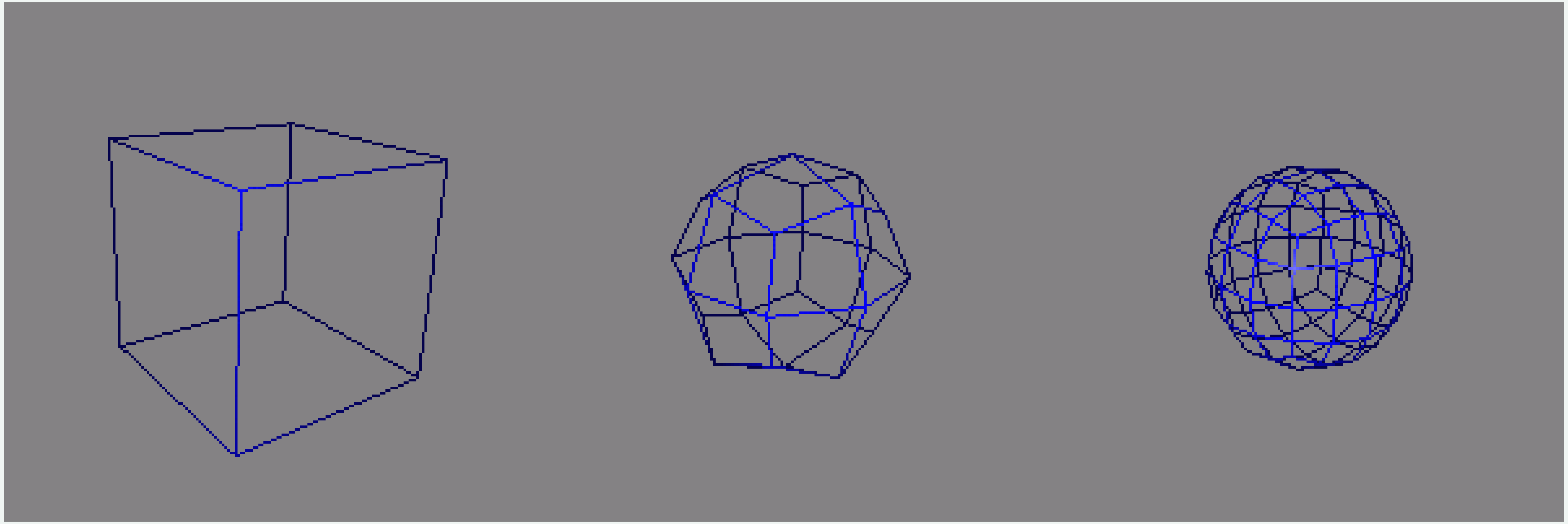
- Quads (everything is a quad after 1 iteration)



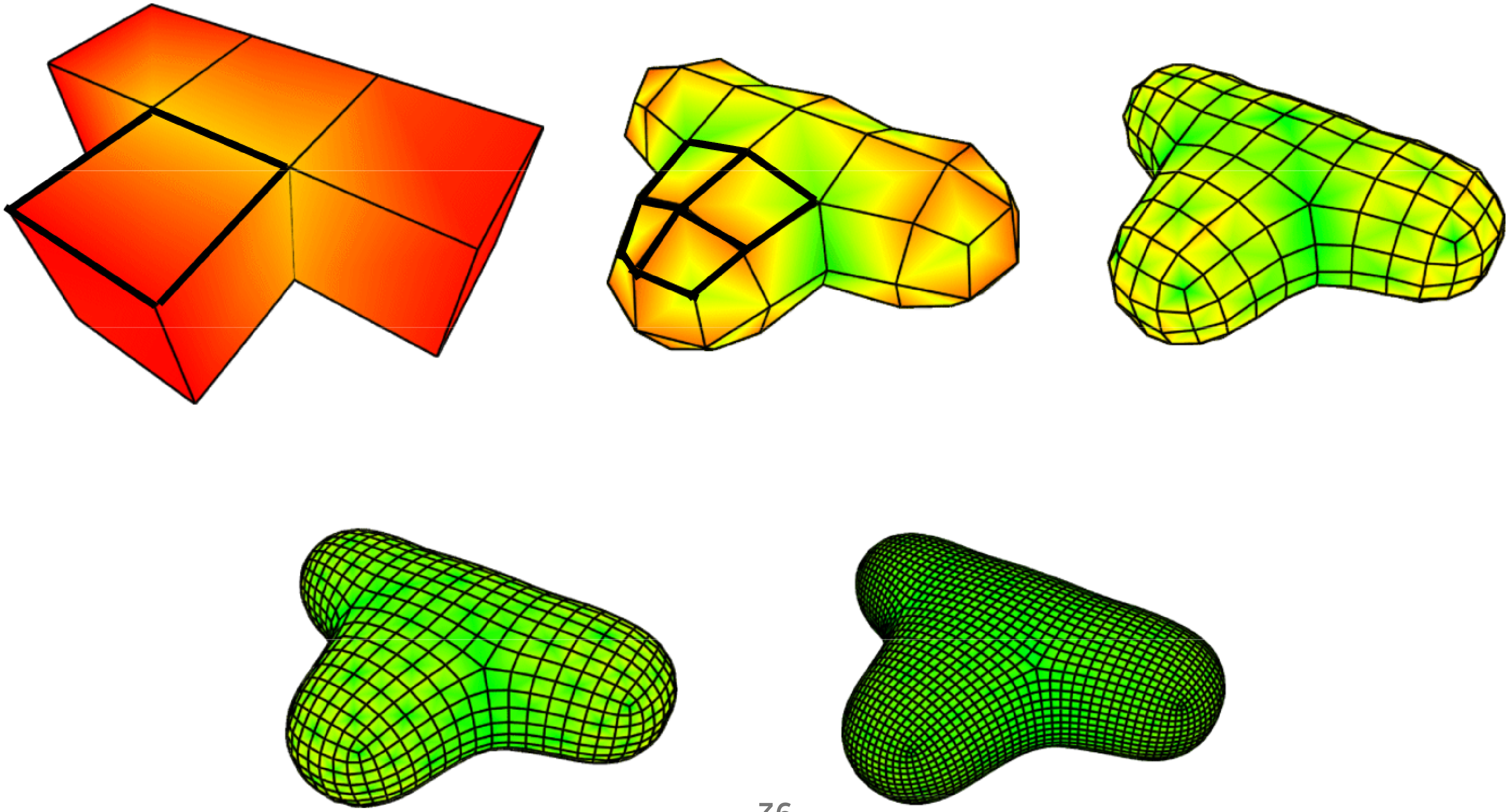
Catmull-Clark Rules

- Face Point = center of polygon
- Edge Point = average 4 neighbors (2 edge, 2 faces)
- Old Points (w/ N edges/faces)
  - $(n-2)/n$ times itself
  - $1/n^2$ average of N edges
  - $1/n^2$ average of N faces



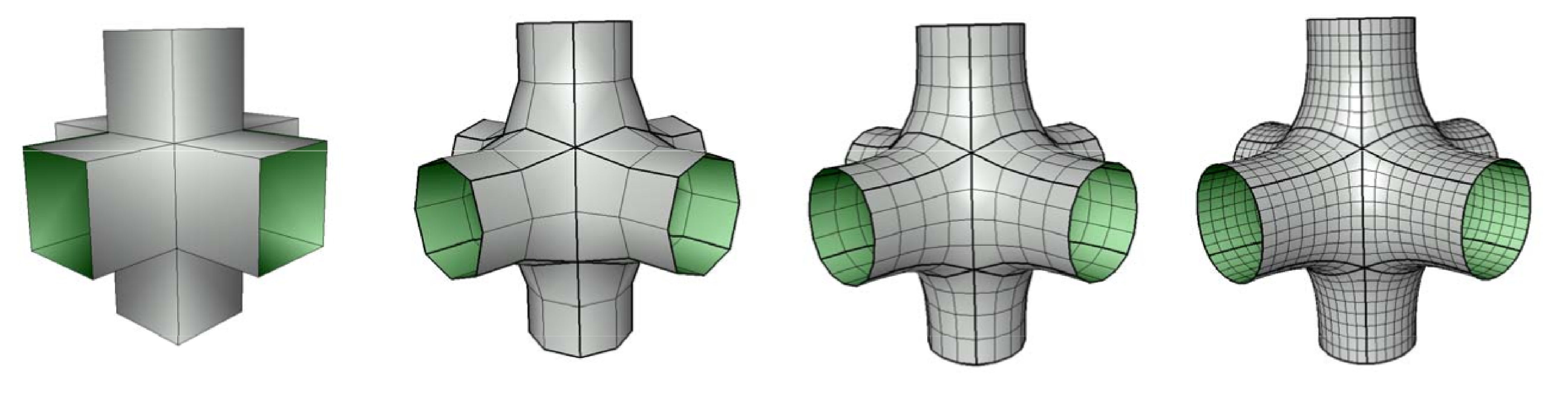


http://www.holmes3d.net/graphics/subdivision/



http://graphics.stanford.edu/courses/cs468-10-

Quads Example



http://graphics.stanford.edu/courses/cs468-10-fall/LectureSlides/10_Subdivision.pdf

What About Edges?

Edges depend only on edges:

- causes them to be "regular curves"

Good Tricks (1) ...

Creases - don't move points for some iterations

Good Tricks (2) … Cutting and Sewing

Put a curve inside of a surface (hole or edge)

Curves stay curves - on any surface!

Why do we like Catmull-Clark so Much?

- Generalizes Cubic B-Splines
- Allows for stopping at any time
- Can compute exact normals (since B-Splines)
- Much easier than Non-Subdivision
- Not that hard to implement
    - requires mesh data structures for splitting and neighbor finding
- Made Popular by Pixar

(Smooth) Surfaces Review

- Surface vs. Solid Vs. Curve

- Not Free-Form
  - primitive shapes
  - generalized primitives (sweeps, lofts, …)

- Free Form
  - Implicit
  - Parametric (and why not)
  - Subdivision (why and how)