Surface Generation

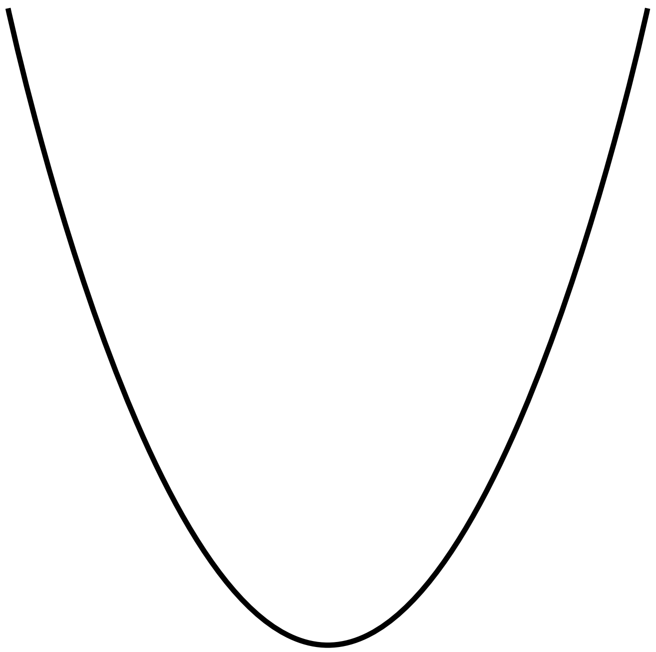
Thanks to Prof. Chuck Hansen for figures and slides

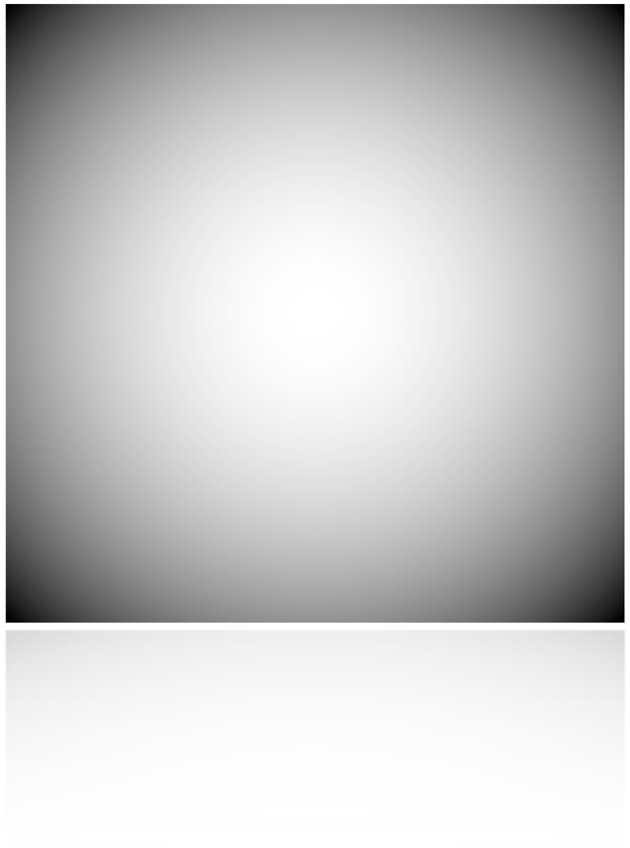
Recap

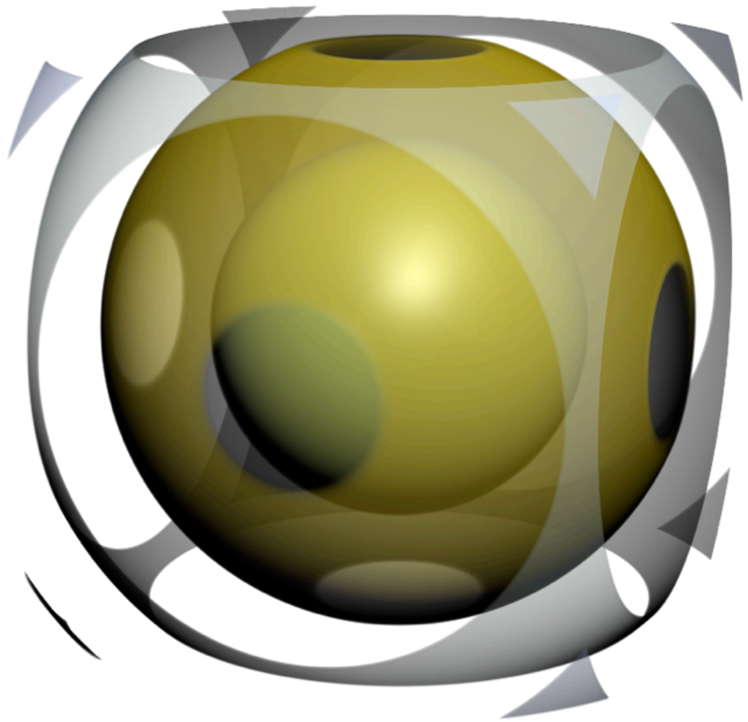
- Implicit surfaces
- $f(x,y,z) = f(s); S = \{s: f(s) = 0\}$

- The representation **works in any dimension**

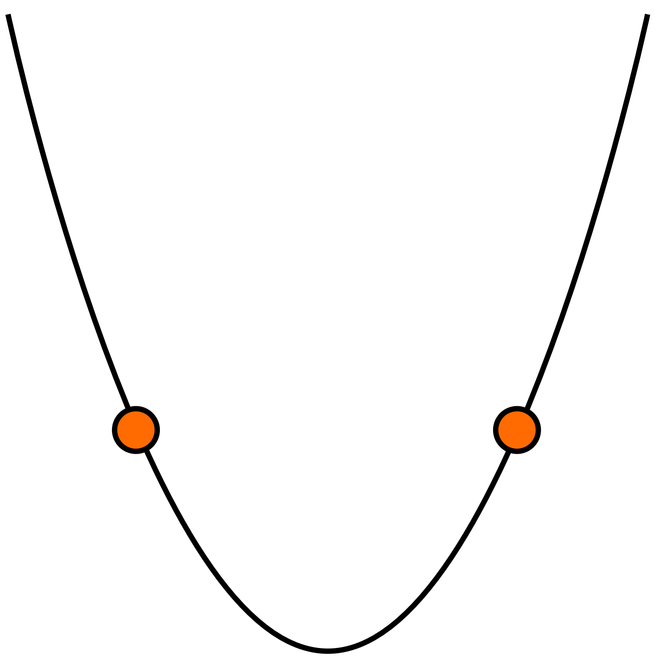
Implicit Surfaces in nD

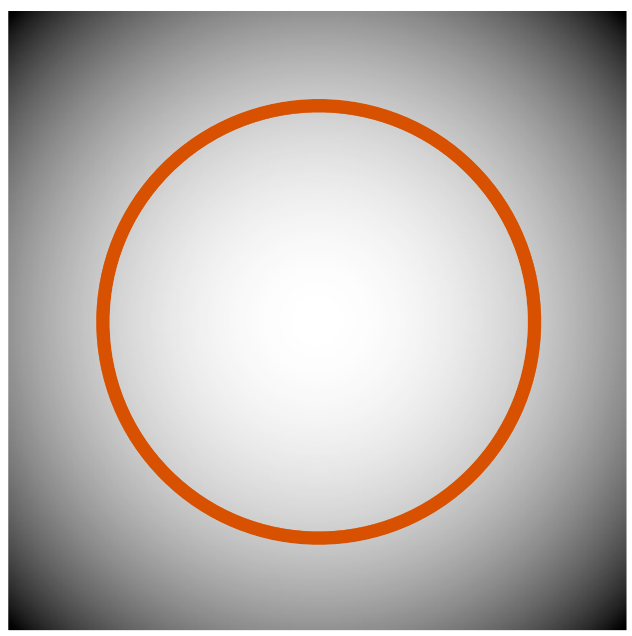


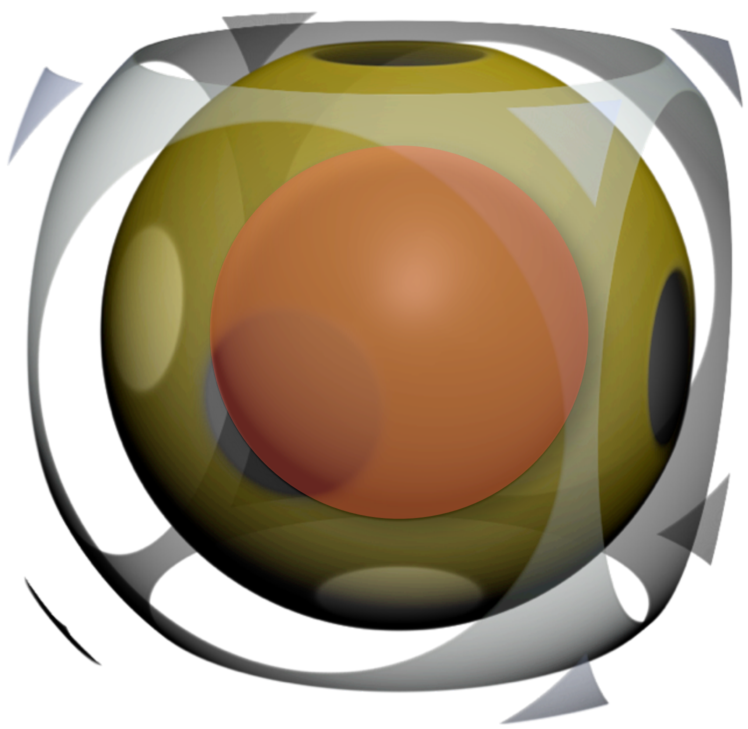




Implicit Surfaces in nD

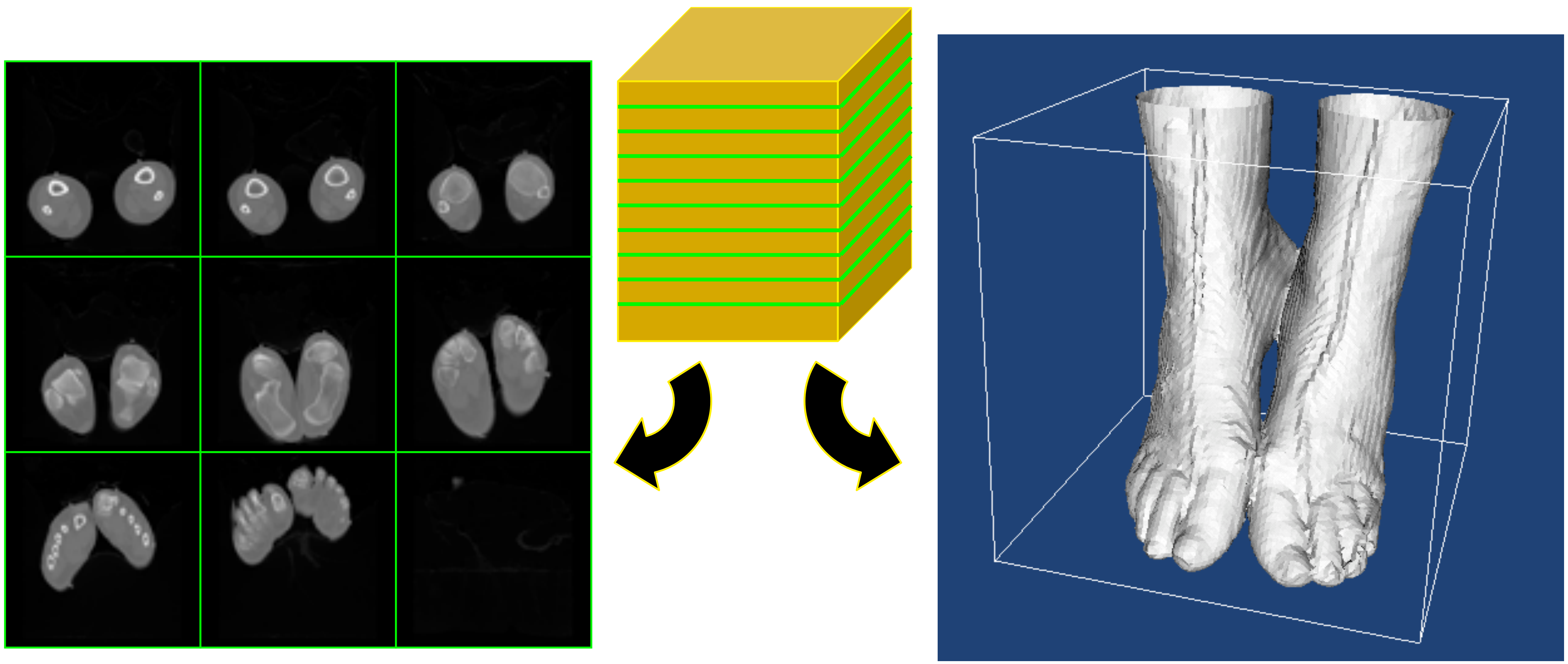






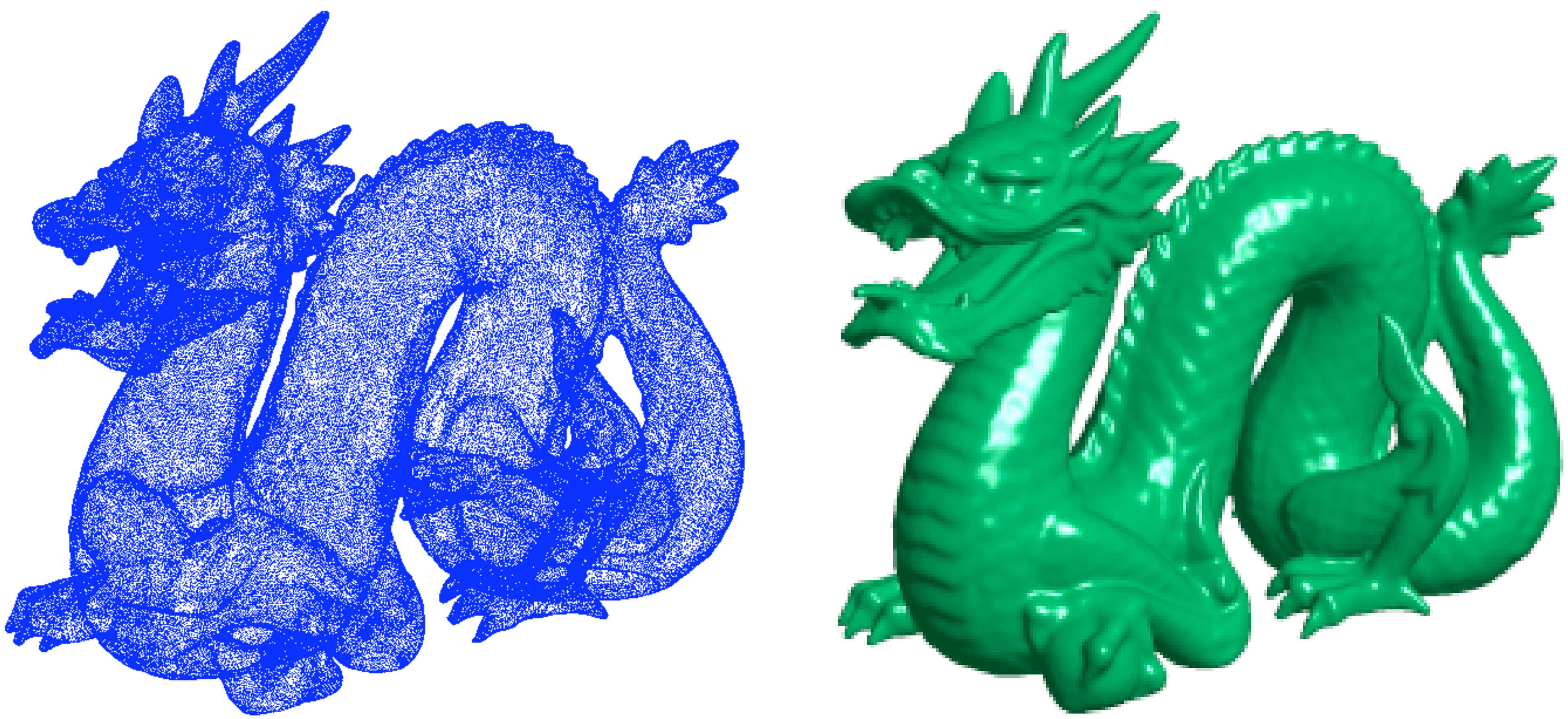
Isosurface Extraction

- Given an implicit surface, get a triangle mesh



Not only SciVis!

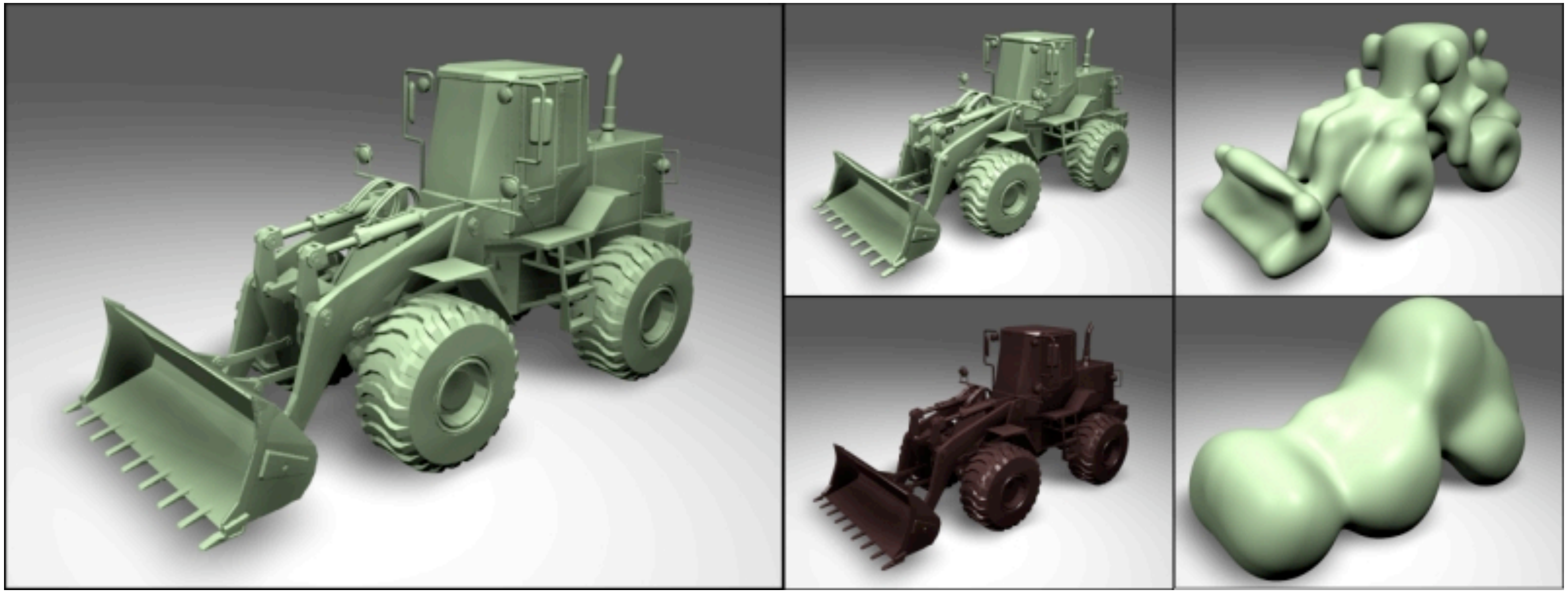
- Implicit surfaces are great for modeling



Carr, Beatson, Cherrie, Mitchell, Fright, McCallum, Evans
Reconstruction and Representation of 3D Objects with Radial Basis Functions

Not only SciVis!

- Implicit surfaces are great for modeling

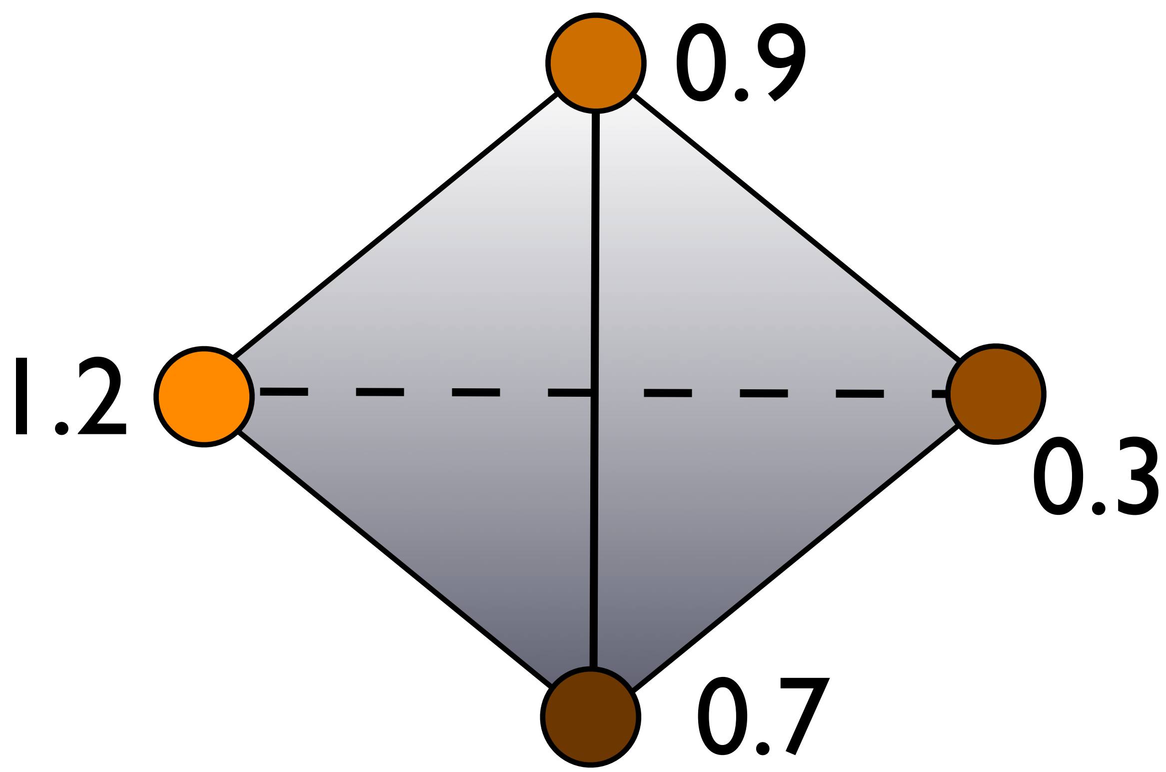


Shen, Brien, Shewchuk.
Interpolating and Approximating Implicit Surfaces from Polygon Soup

Marching Tetrahedra

- Assume volume is represented by tetrahedral mesh, with scalars attached to vertices
- Scalar field built by barycentric interpolation of scalar values

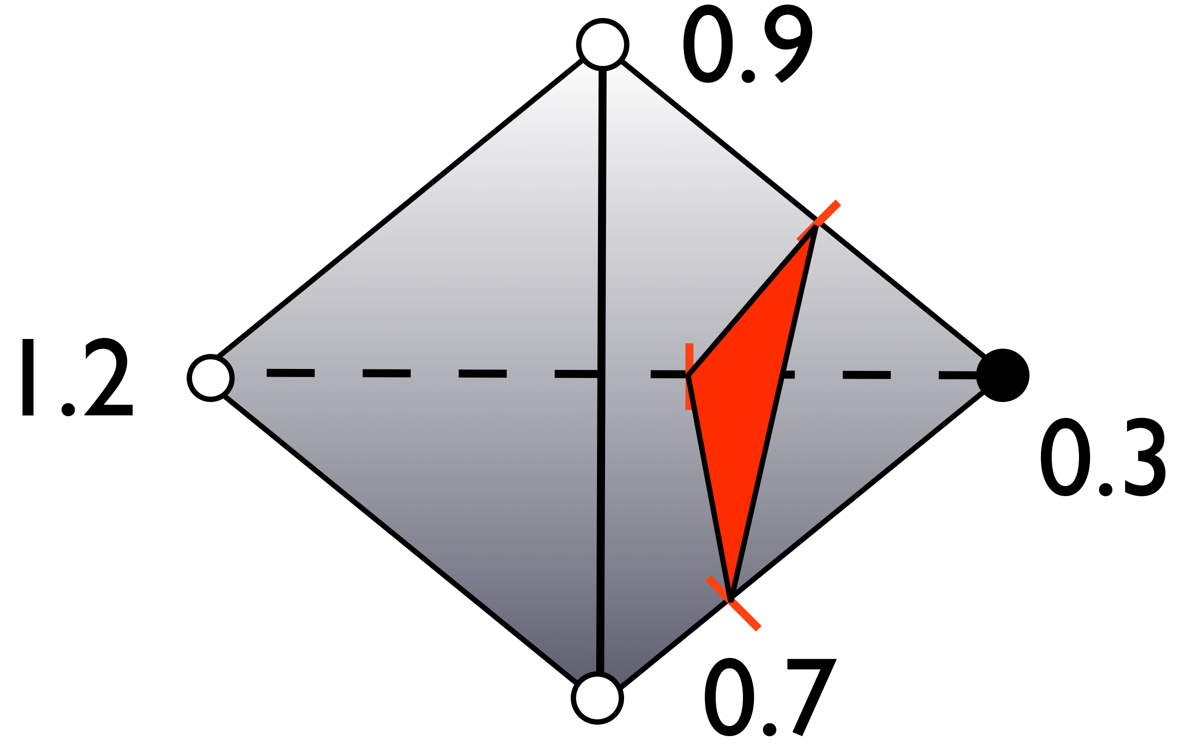
Function Reconstruction in Tet meshes



- Scalar field built by barycentric interpolation of scalar values
- Scalar function reconstruction is **convex**
- Isosurfaces are **piecewise linear**

MT: simple example

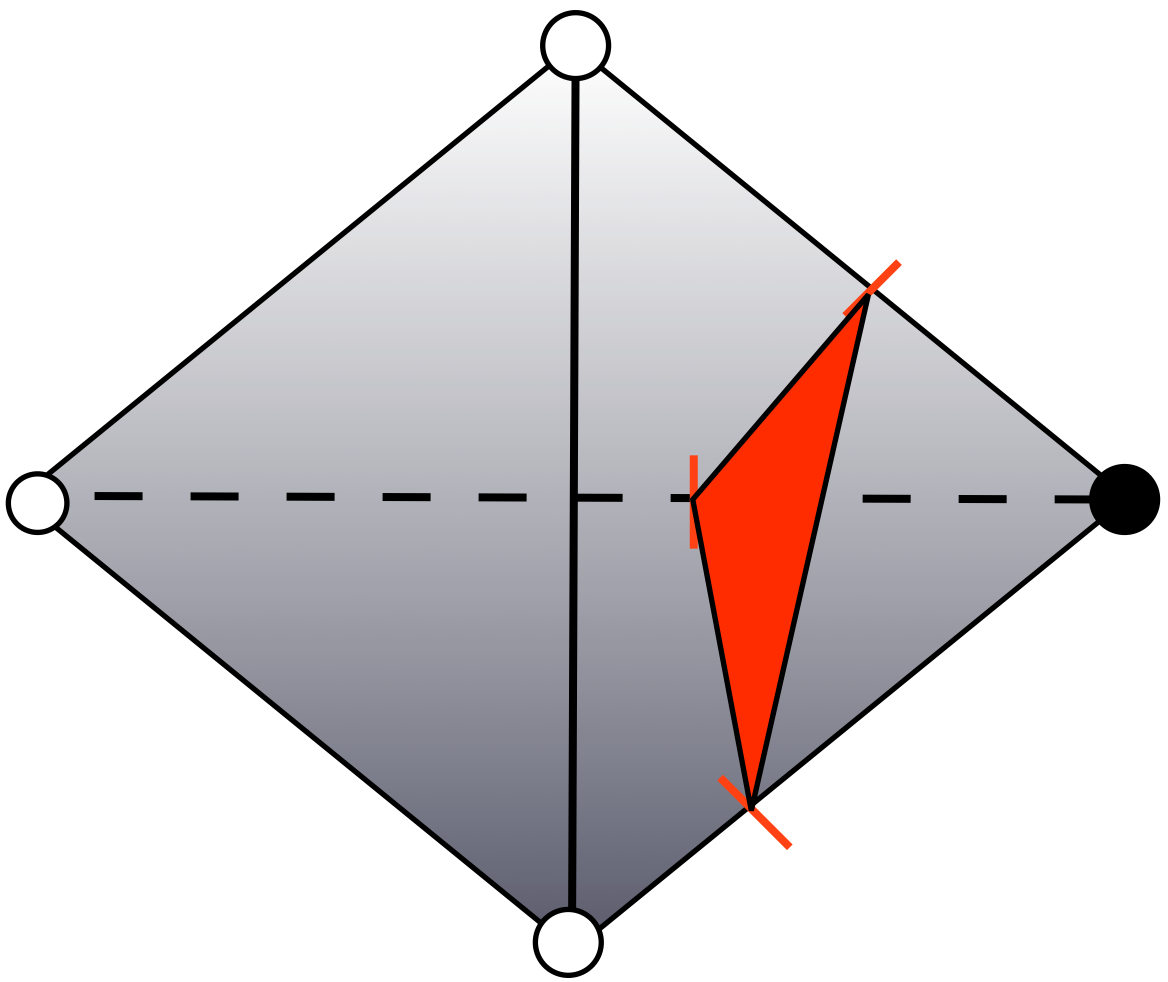
- f(x) = 0.6



- Only look at intersections with edges
- How many cases? Much fewer than 16

MT: only two cases

- Only look at edges where vertices have opposing signs



On to Cubes

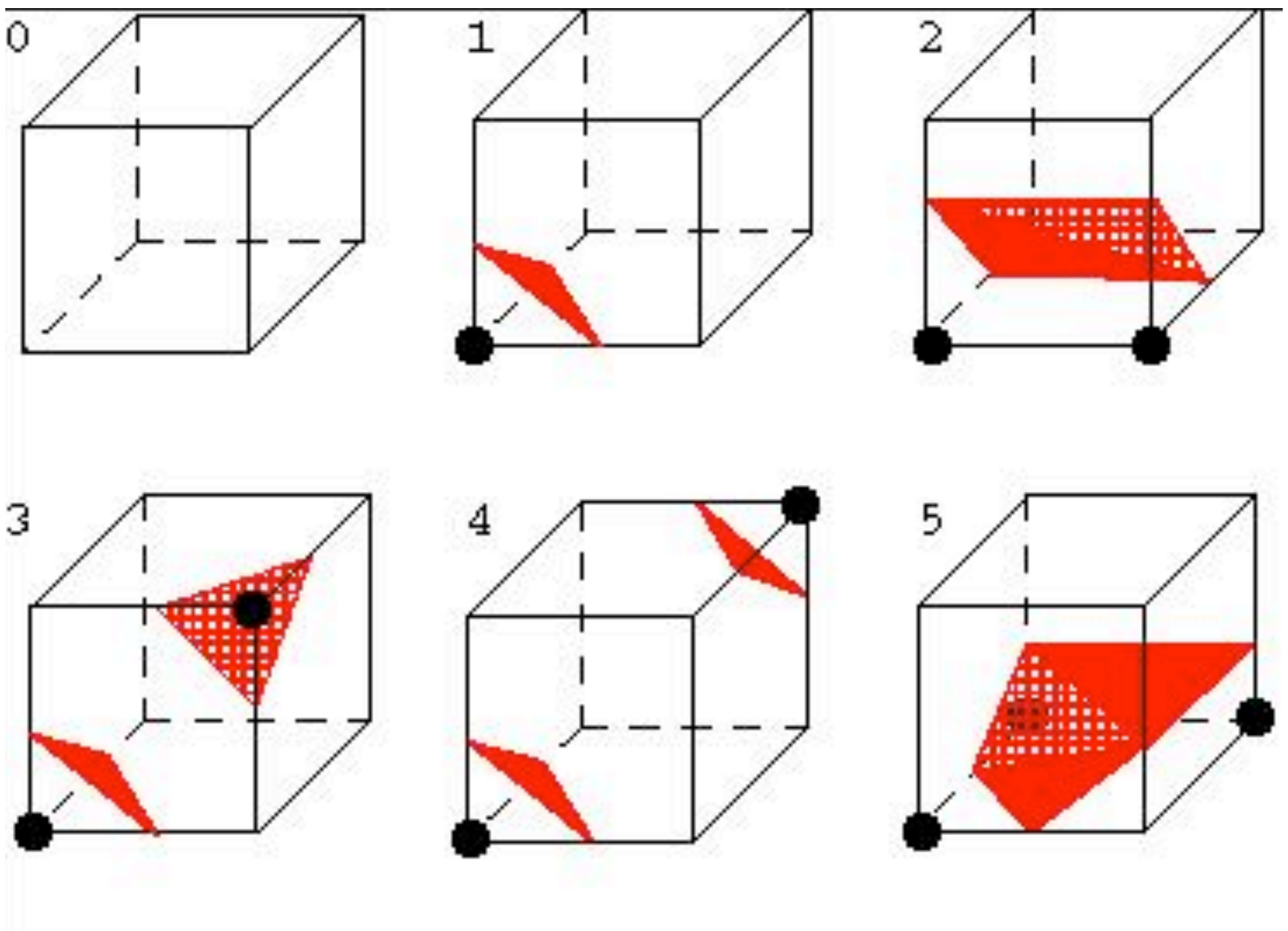
- First idea: do no work! Split cube in tets
- Many ways to split a cube into tets
  - How do you minimize the size of the output?
  - Are there any problems? Look at the props

Splitting into tets is wasteful

Scheme	Triangles	Ratio
Marching Cubes	1,029,936	1.0
Minimal (5), No Parity	2,452,378	2.381
Minimal (5), Even Parity	2,453,046	2.382
Minimal (5), Odd Parity	2,452,370	2.381
Freudenthal (6), Axis 000 - 111	3,011,206	2.924
Freudenthal (6), Axis 001 - 110	3,003,346	2,916

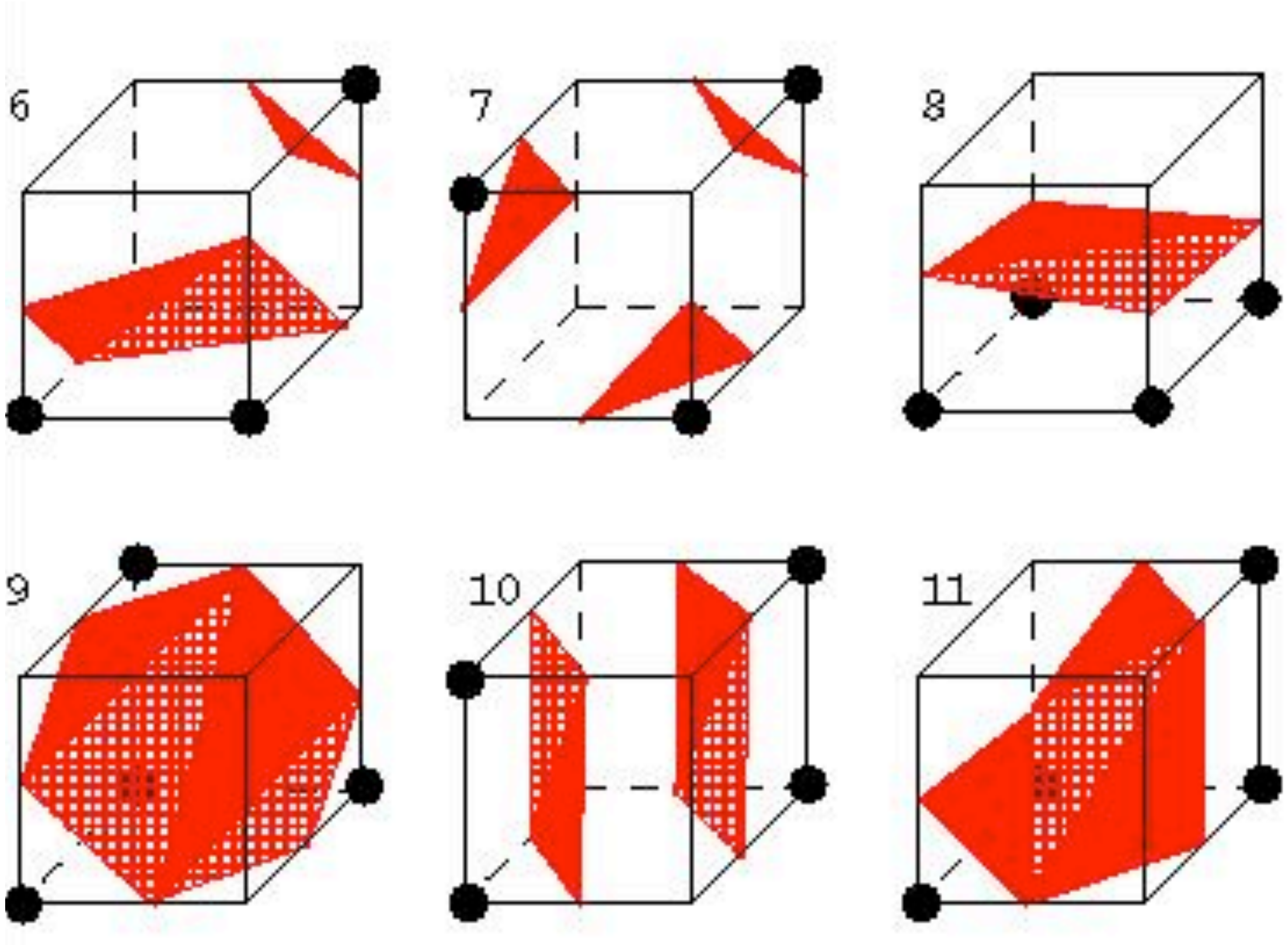
Carr, Möller, Snoeyink
Artifacts Caused By Simplicial Subdivision

Marching Cubes



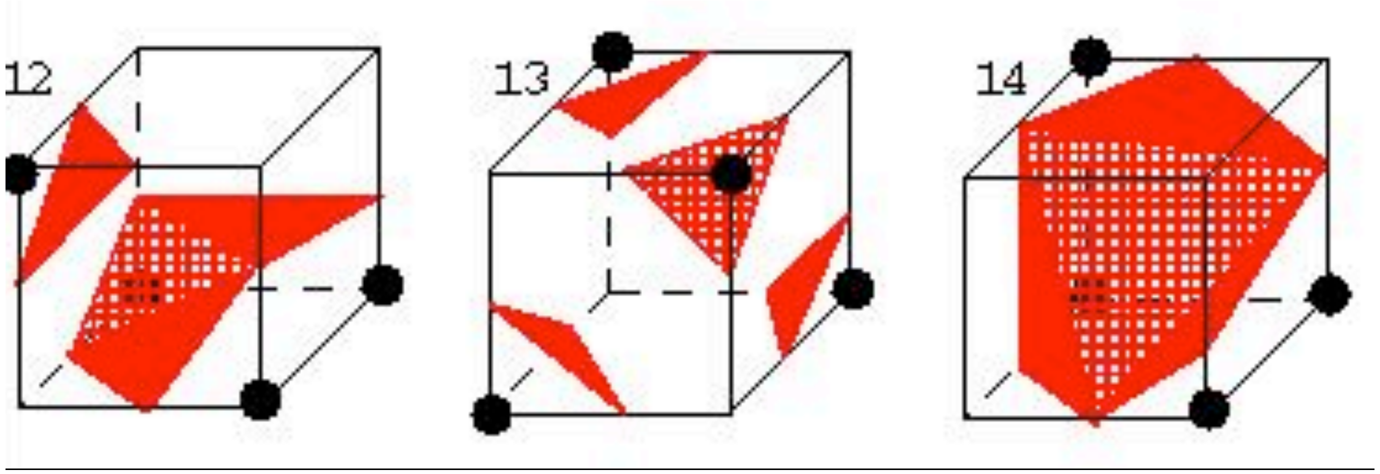
Direct inspection:
15 cases

Marching Cubes



Direct inspection:
15 cases

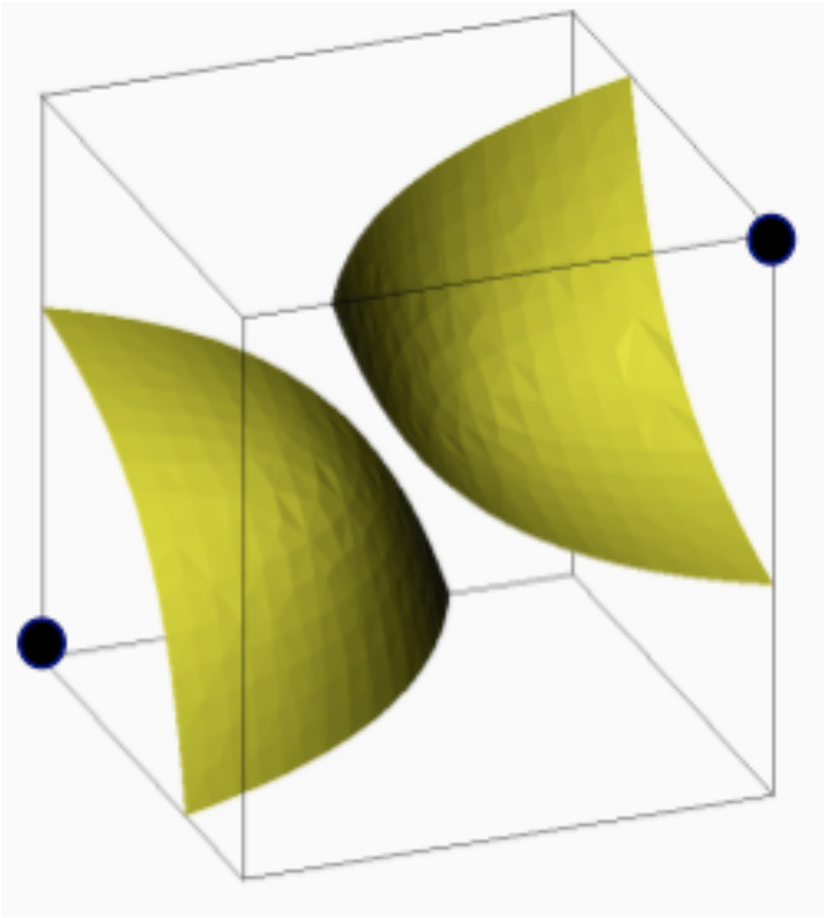
Marching Cubes

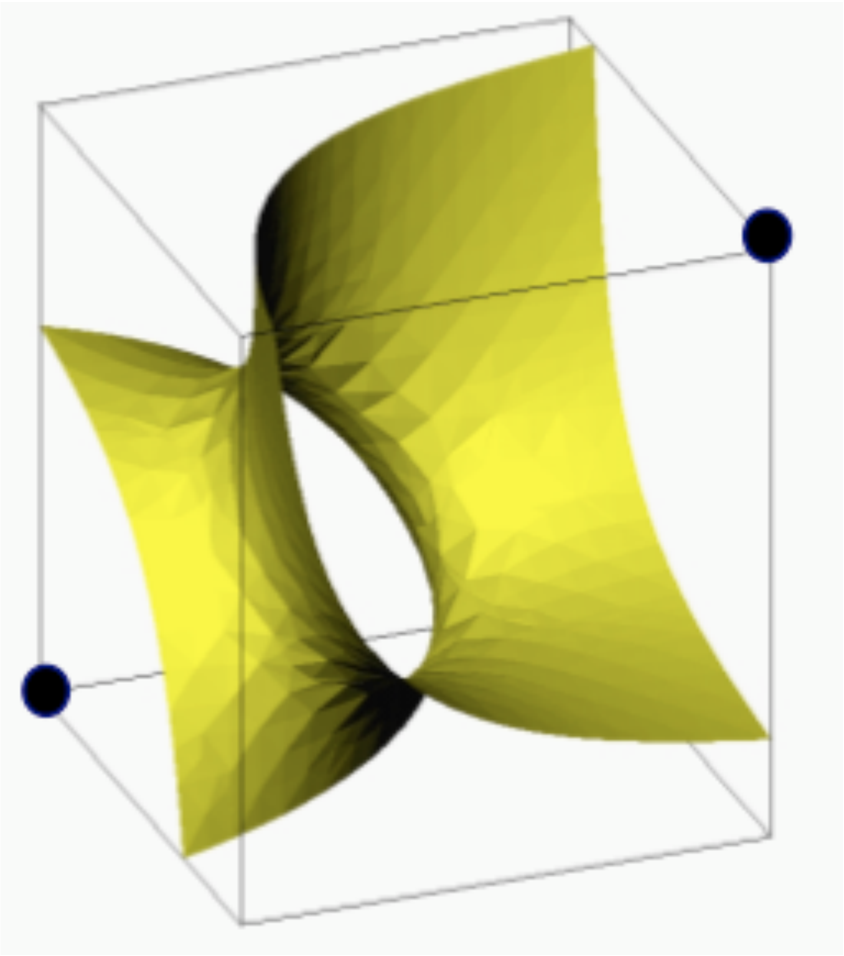


Direct inspection:
15 cases

Ambiguity in MC

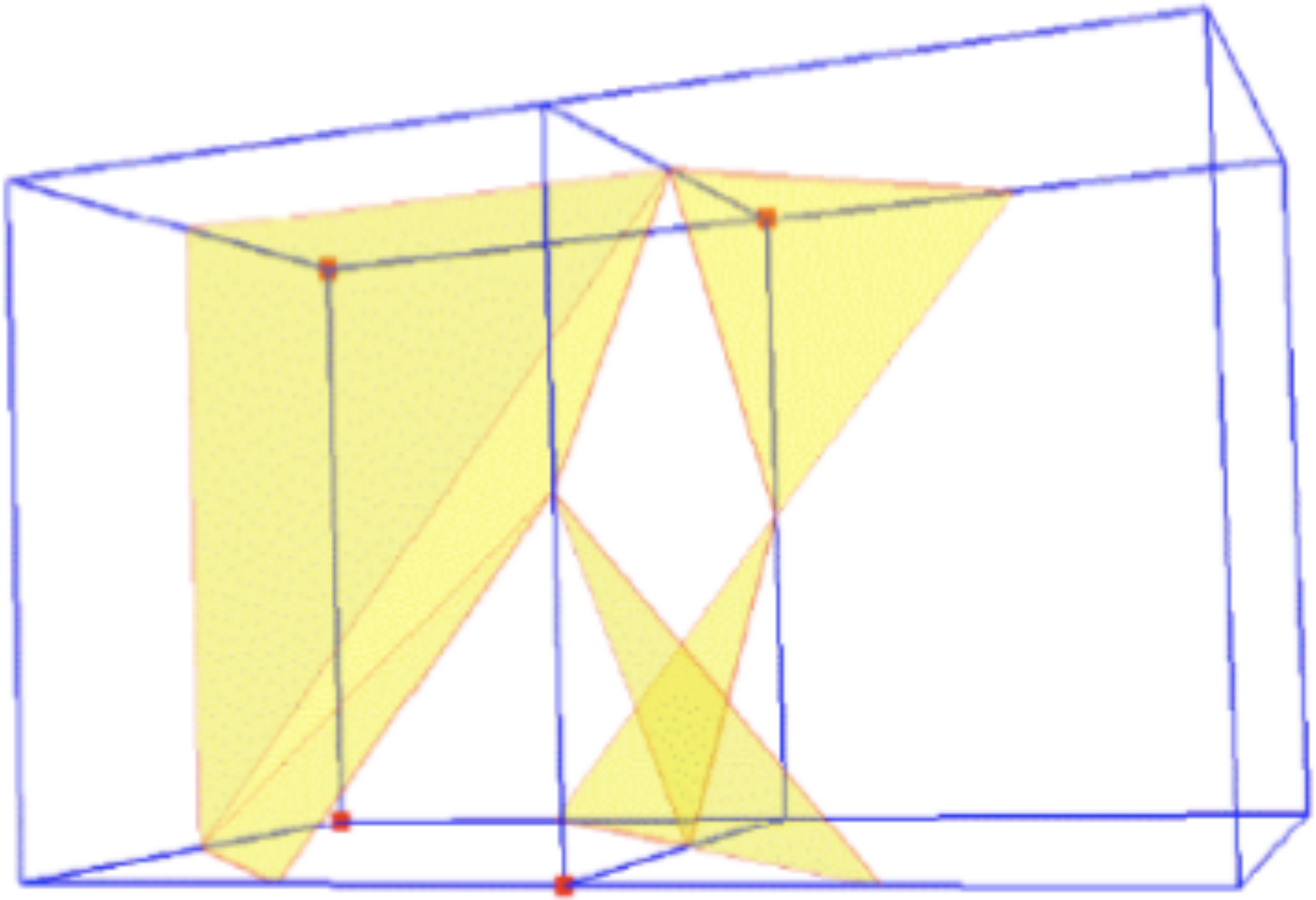
- Trilinear interpolation is tricky



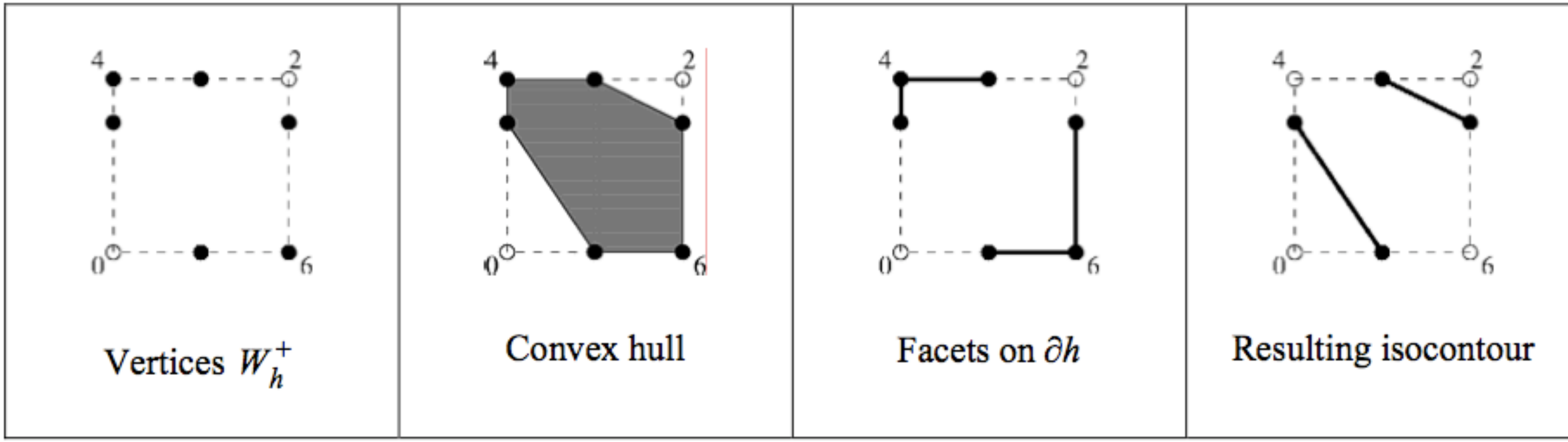


Ambiguity in MC

- Naive table leaves holes!



Generalizing MC



Isosurface Construction in Any Dimension Using Convex Hulls
Bhaniramka, Wenger, Crawfis

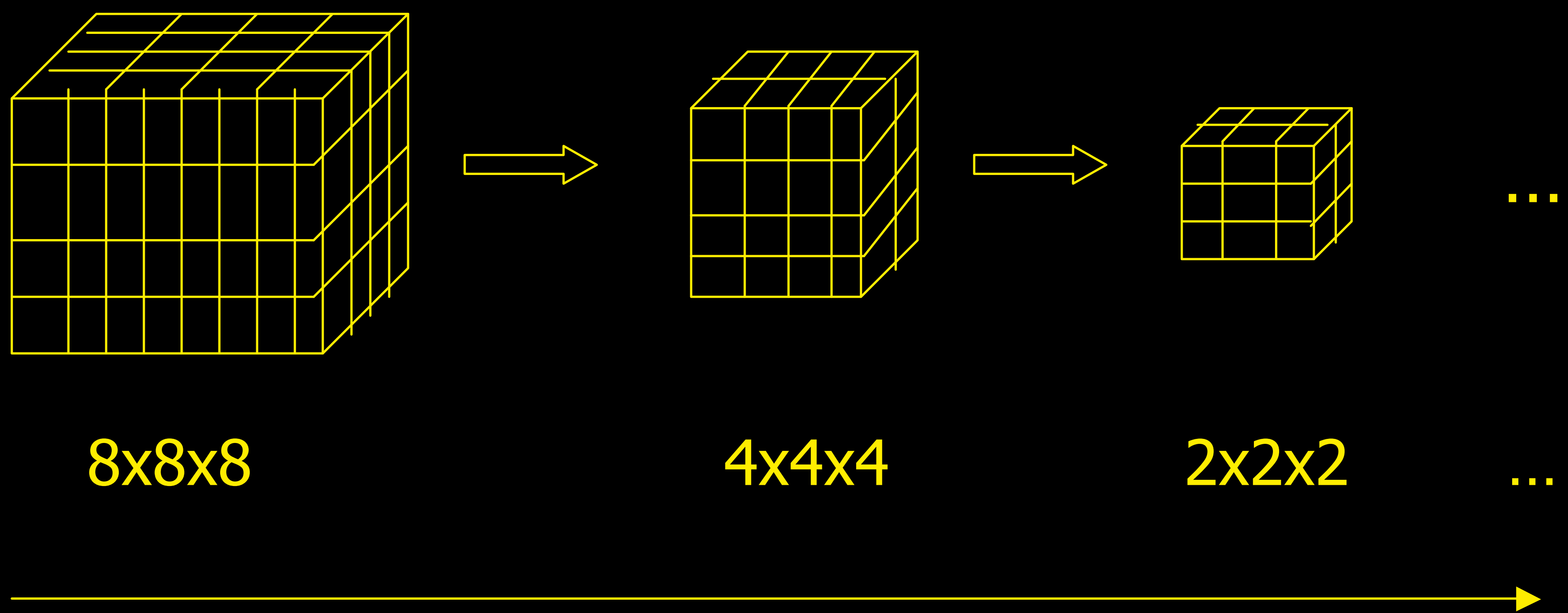
- Higher dimensions, different cells, etc.
- **Automatically constructed!**

Efficiency

- How long does it take to run Marching Cubes?
- How can you make it faster?

Octrees

- One way to handle the big data problem is to use hierarchical data structures (hierarchical volumes)



Less amount of data are required

Other Methods

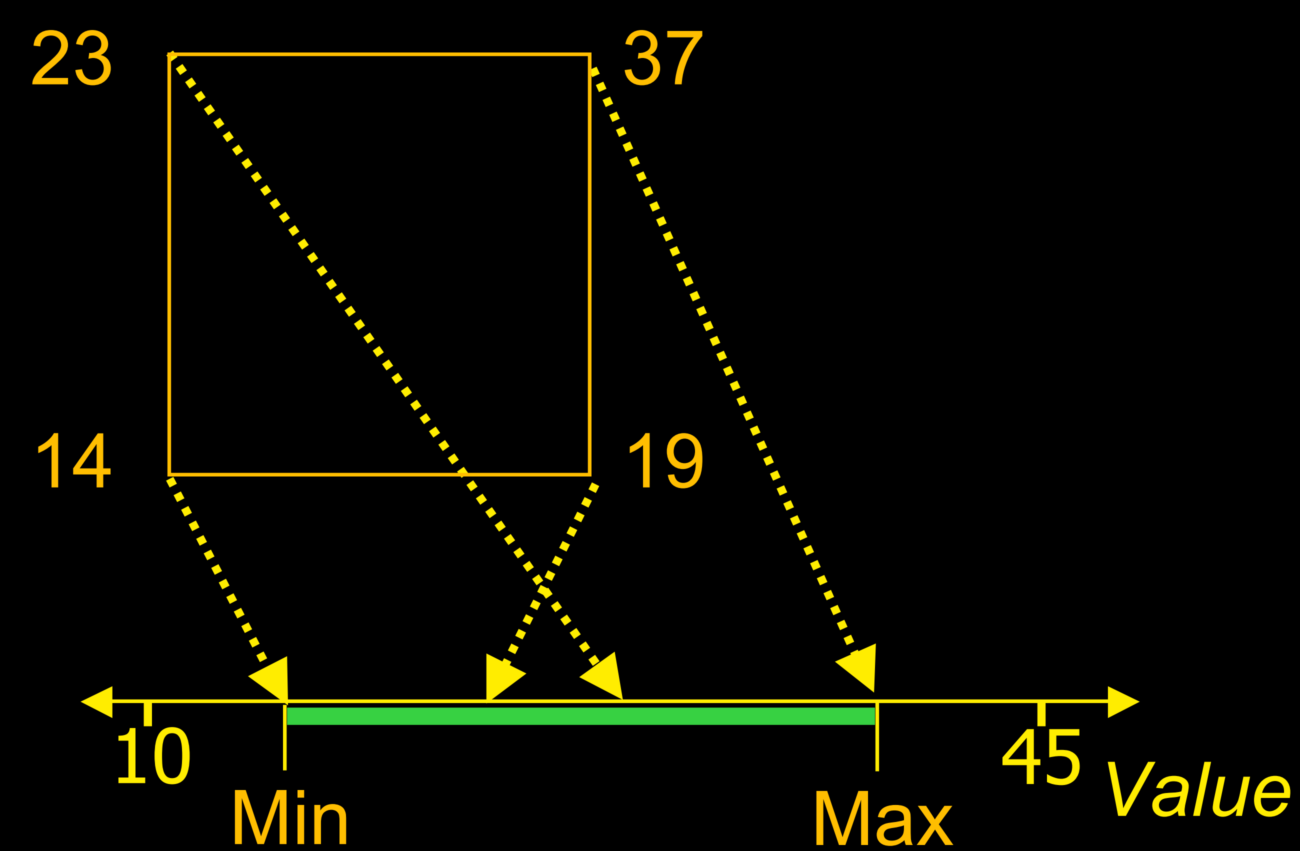
MinMax Octree

Wilhelms and Van Gelder 90/92

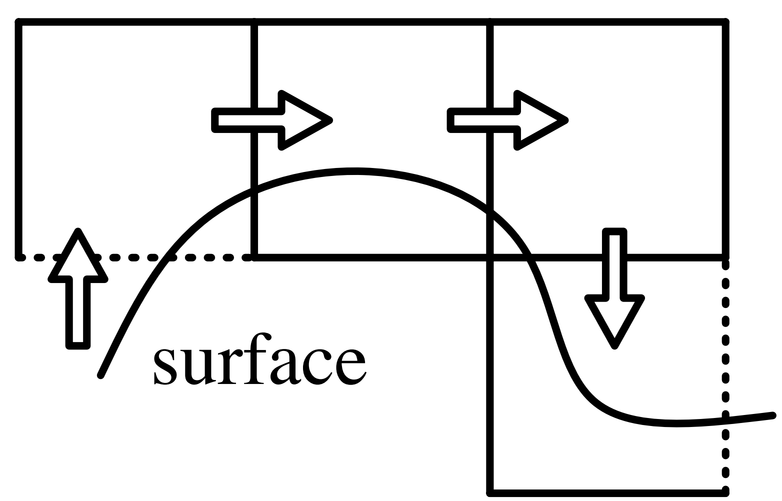
Search Complexity:

$O(k \log(n/k) + k)$

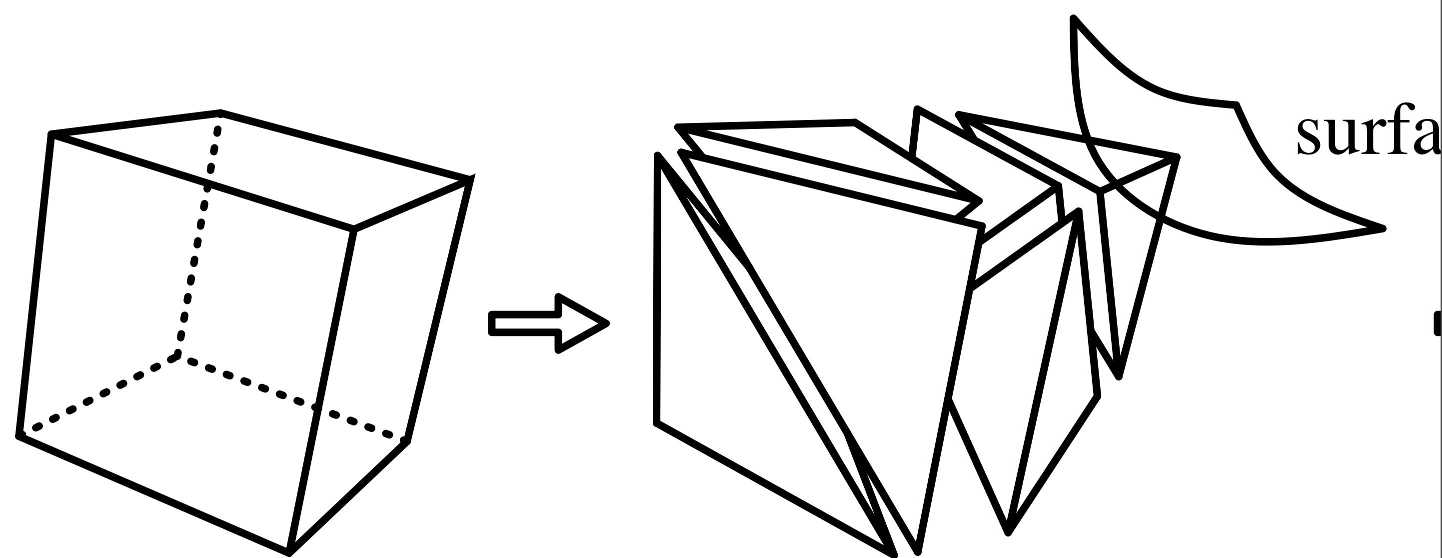
Livnat, Shen and Johnson 96



Interlude: Surface Continuation



continuation (side view)



decomposition

Jules Bloomenthal
An Implicit Surface Polygonizer

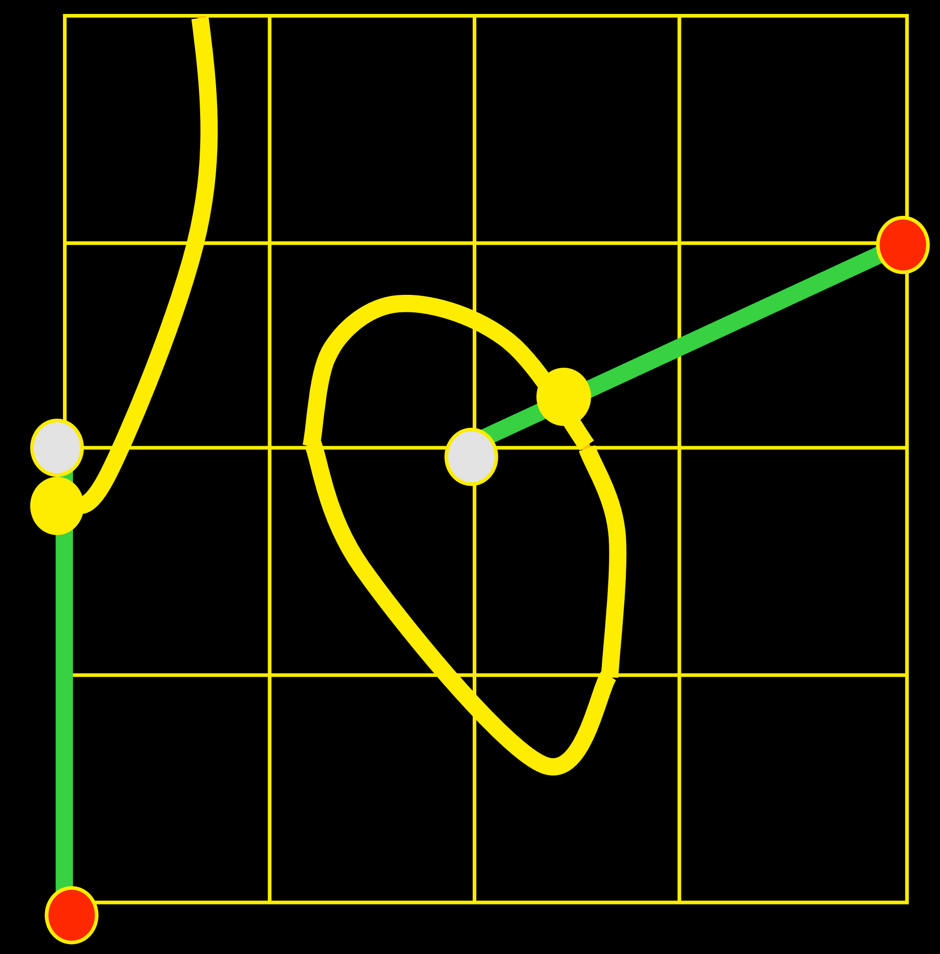
Other Methods

Extrema Graphs

Itoh and Koyamada 94

Volume Thinning

Itoh, Yamaguchi and Kotamada 96



Search Complexity

- Avg O(n exp(2/3))
- Worst case O(n)

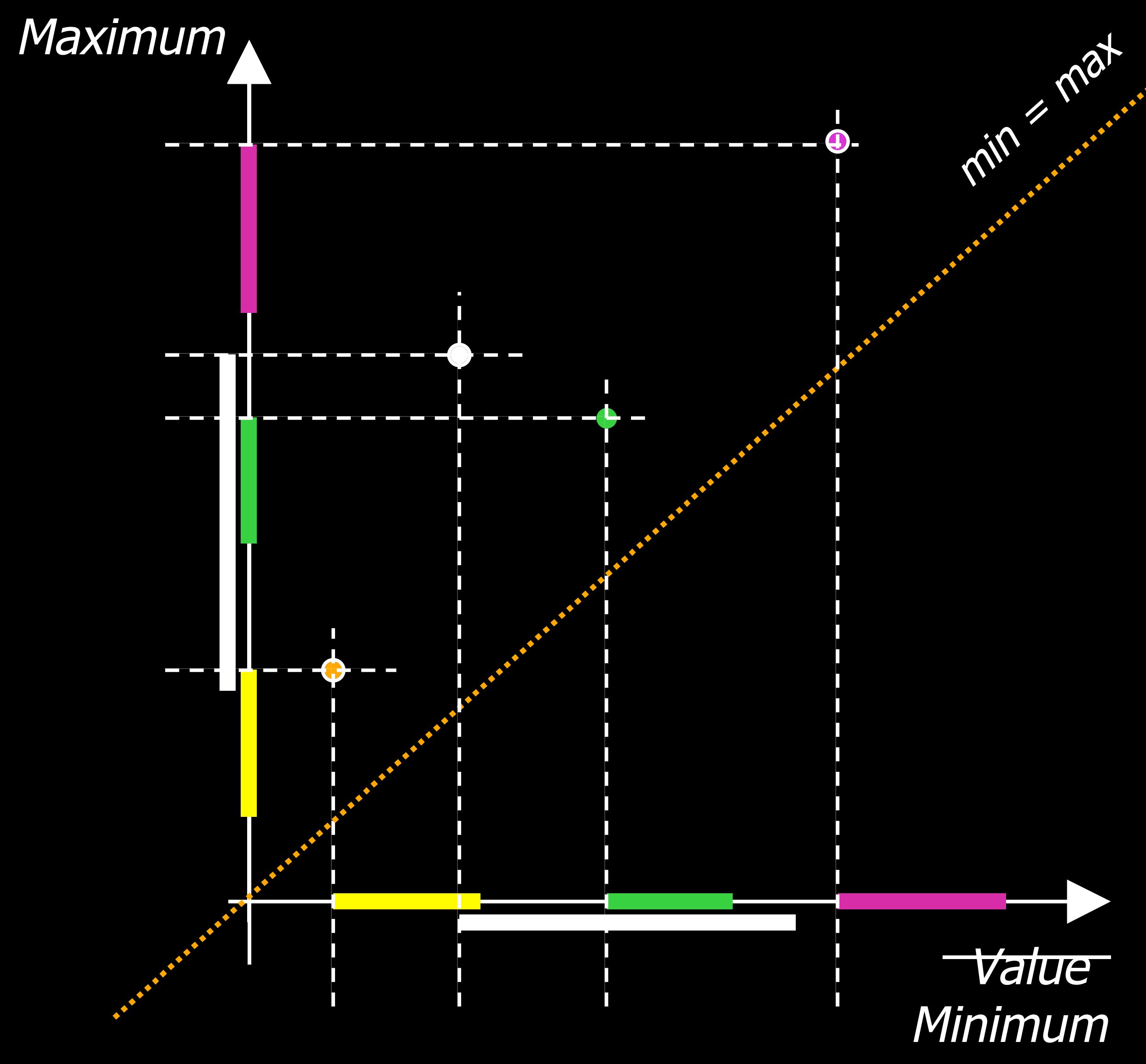
Livnat, Shen and Johnson 96

The Span Space

Livnat, Shen, Johnson 96

- **Given:**
  - Data cells in 8D
- **Past (active list):**
  - Intervals in a 1D Value space
- **New:**
  - Points in the 2D Span Space
- **Benefit:**

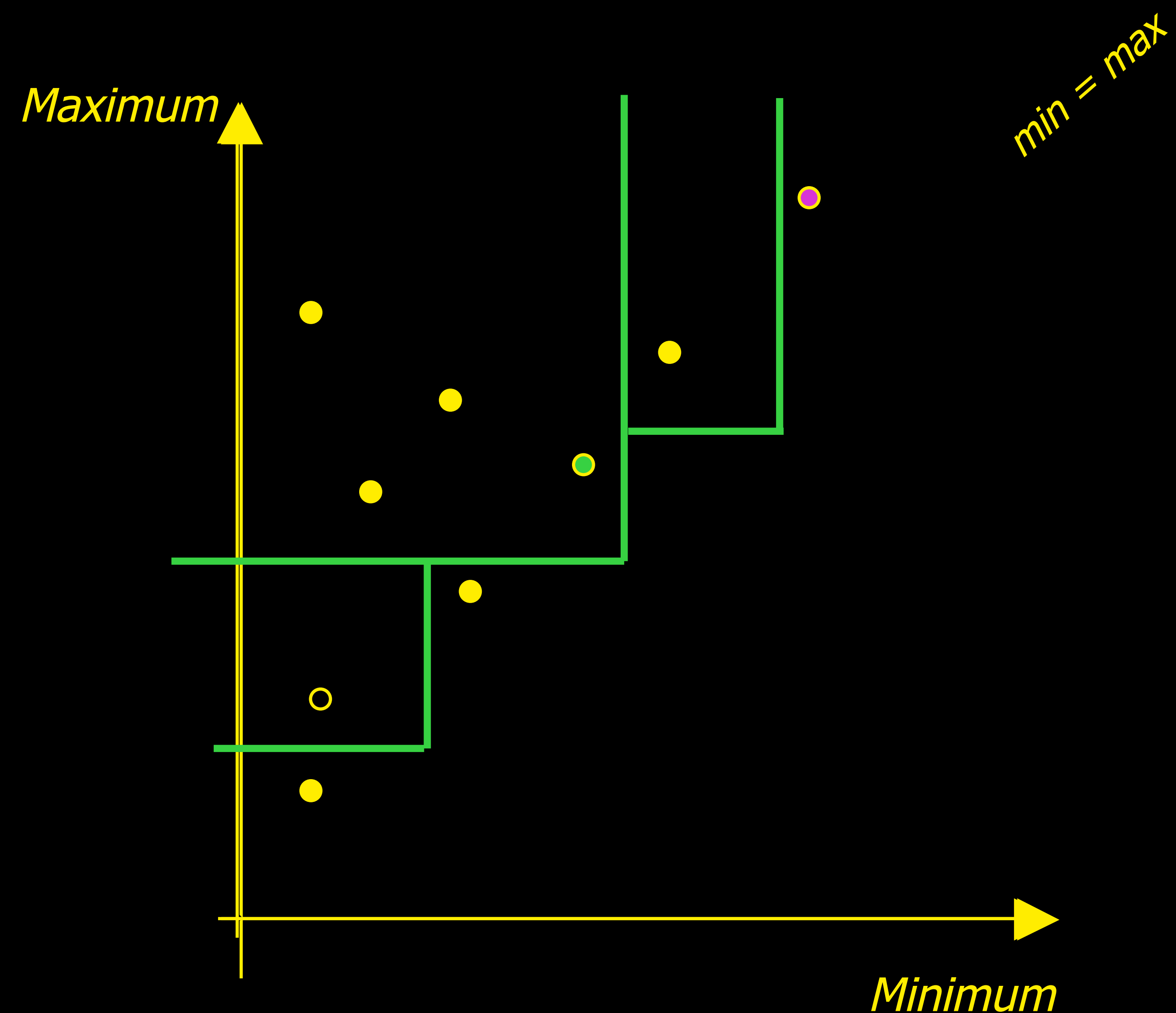
  Points do not exhibit any spatial relationships



Span Space: other works

The Span Space

- NOISE: $O(\sqrt{n}+k)$
  Livnat, Shen, Johnson 96
- Optimal: $O(\log(n)+k)$
  Cignoni et al. 96
  Better search algorithm

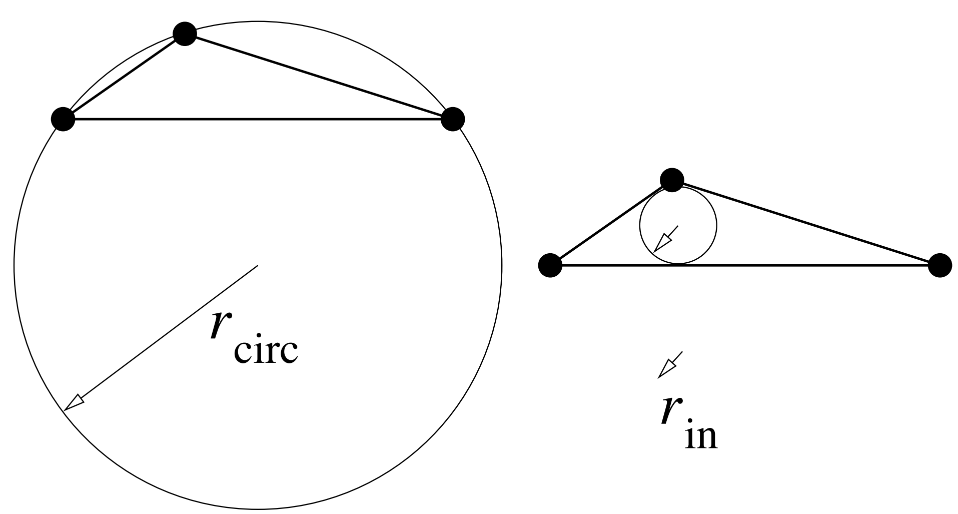


Triangle Quality

- We now know how to make MC faster, so we can use it as a step in the middle of other algorithms
  - Mechanical simulations, etc

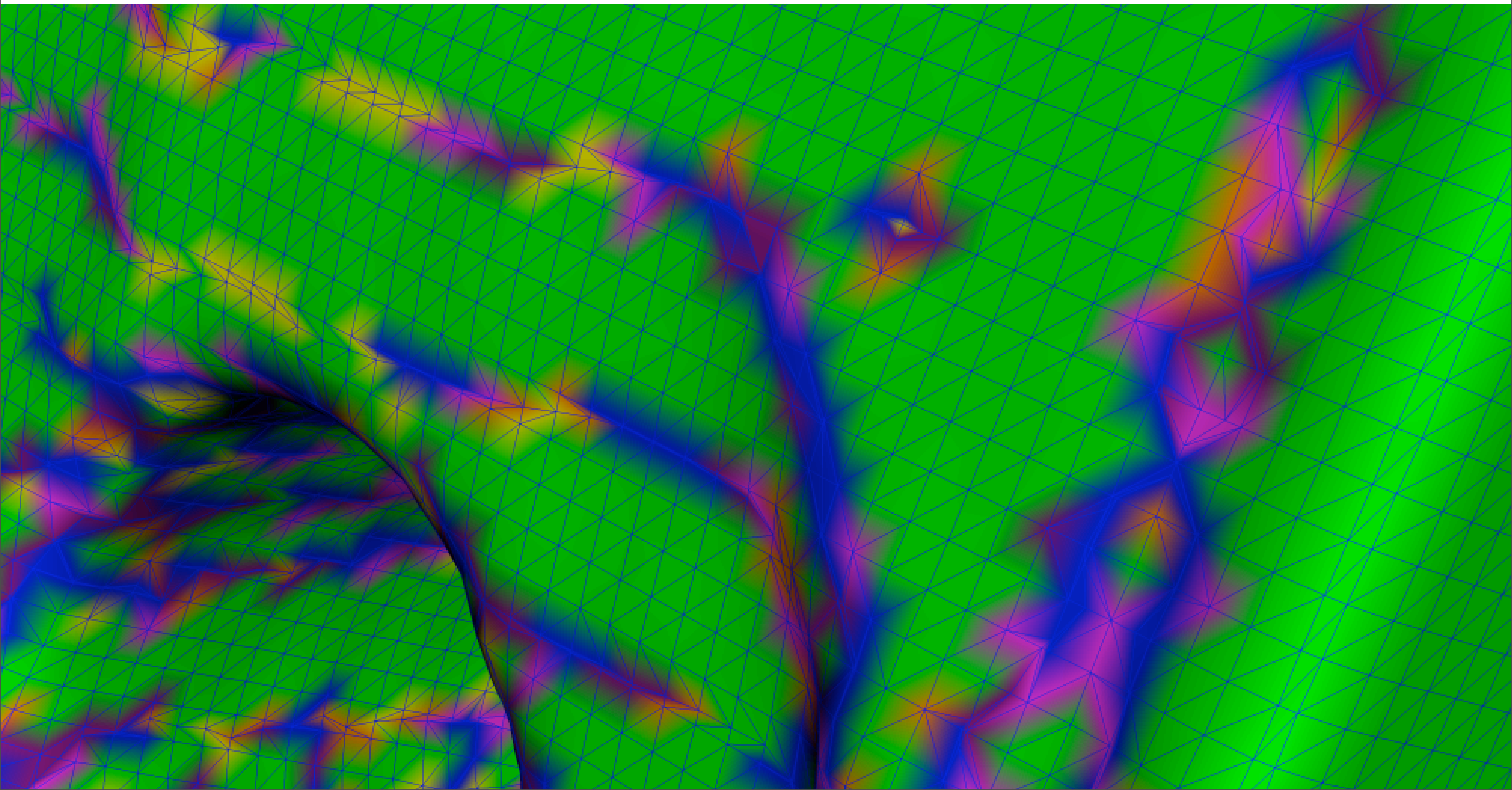
Triangle Quality

- Minimum angle determines condition number of stiffness matrix in some FE simulations
- Maximum angle determines interpolation error (in particular of gradients)
- Radii-ratio goes to zero for **either** of the above

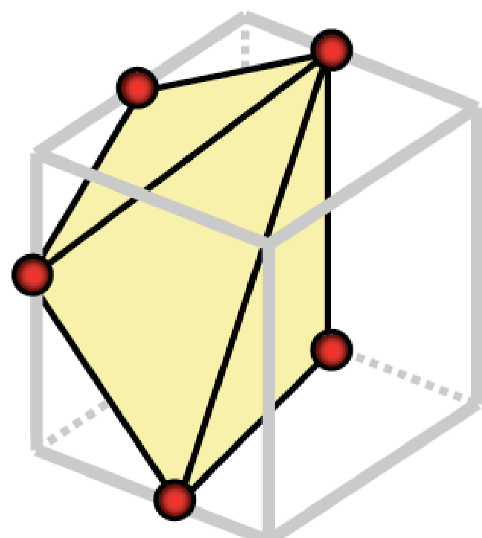


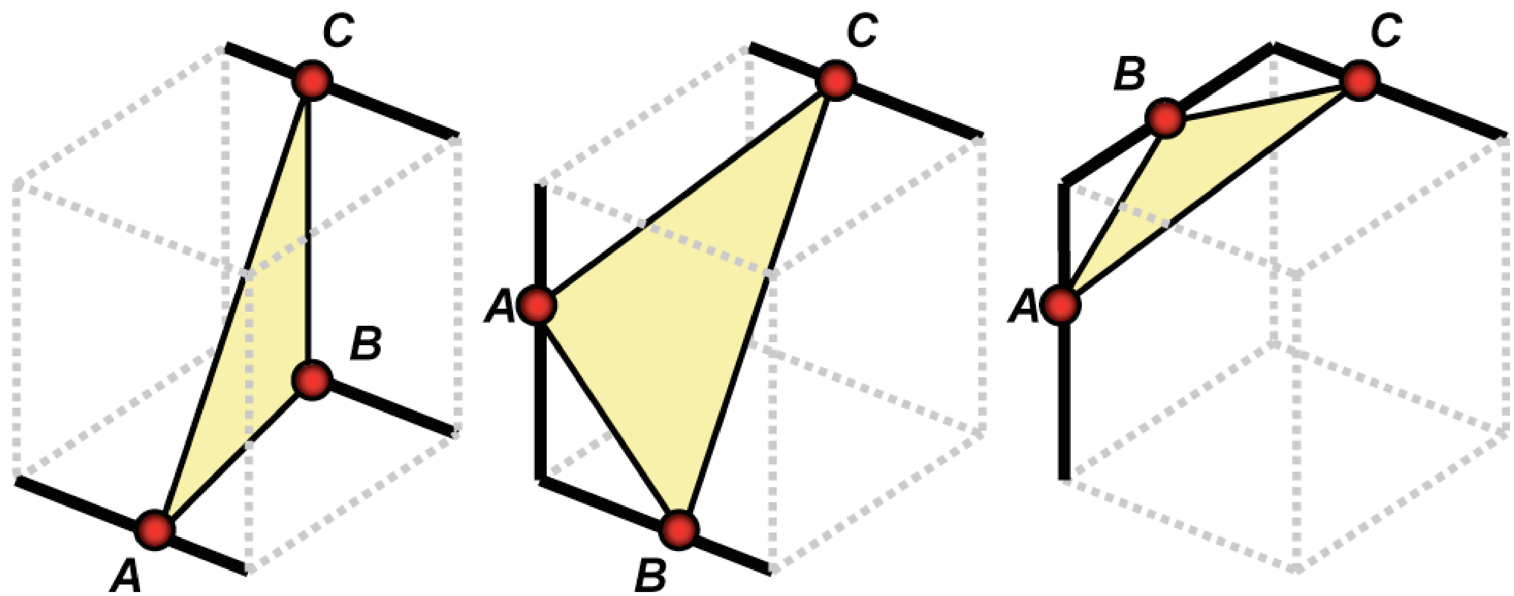
circumcircle incircle

Triangle Quality



Edge Groups

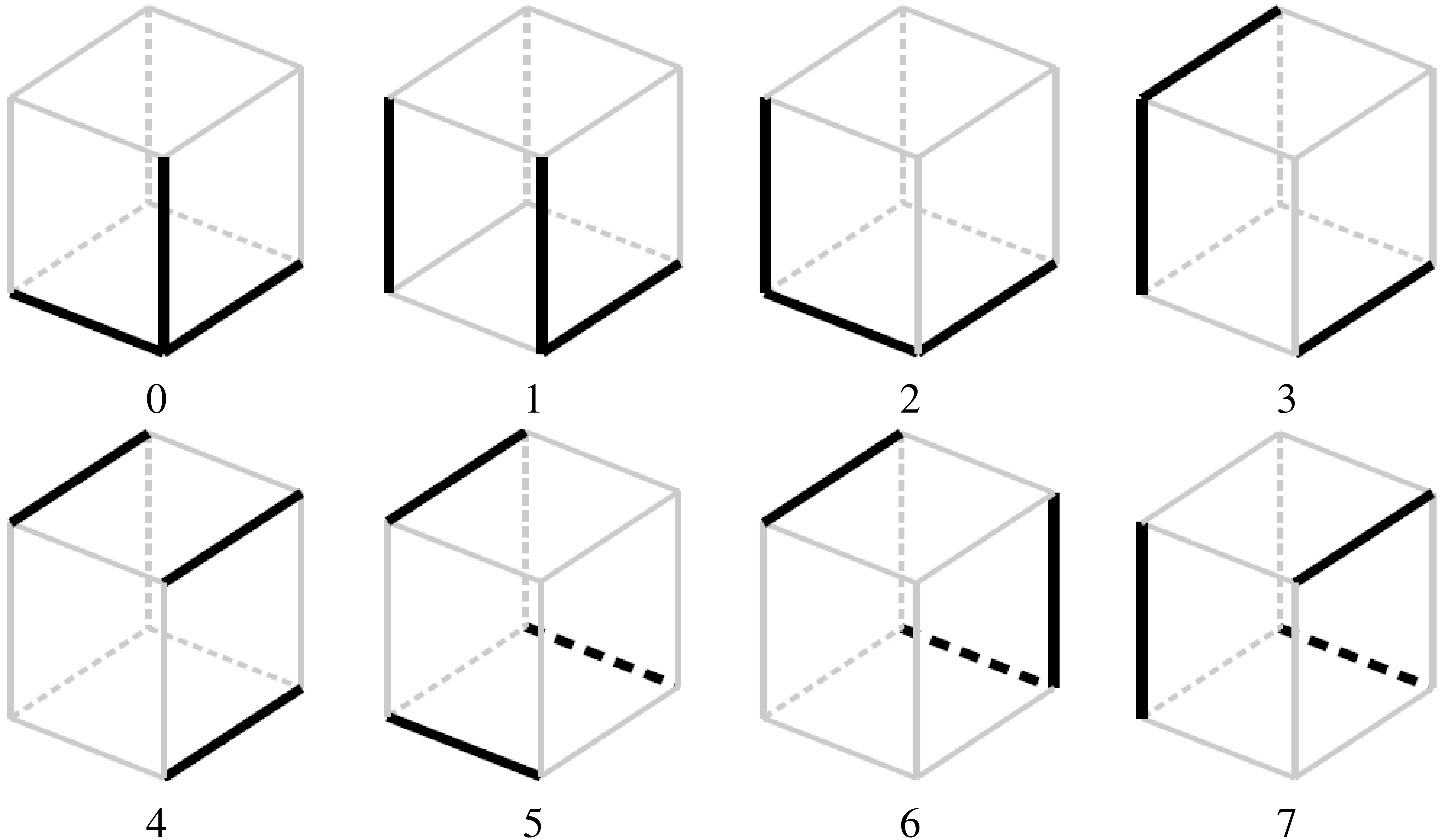




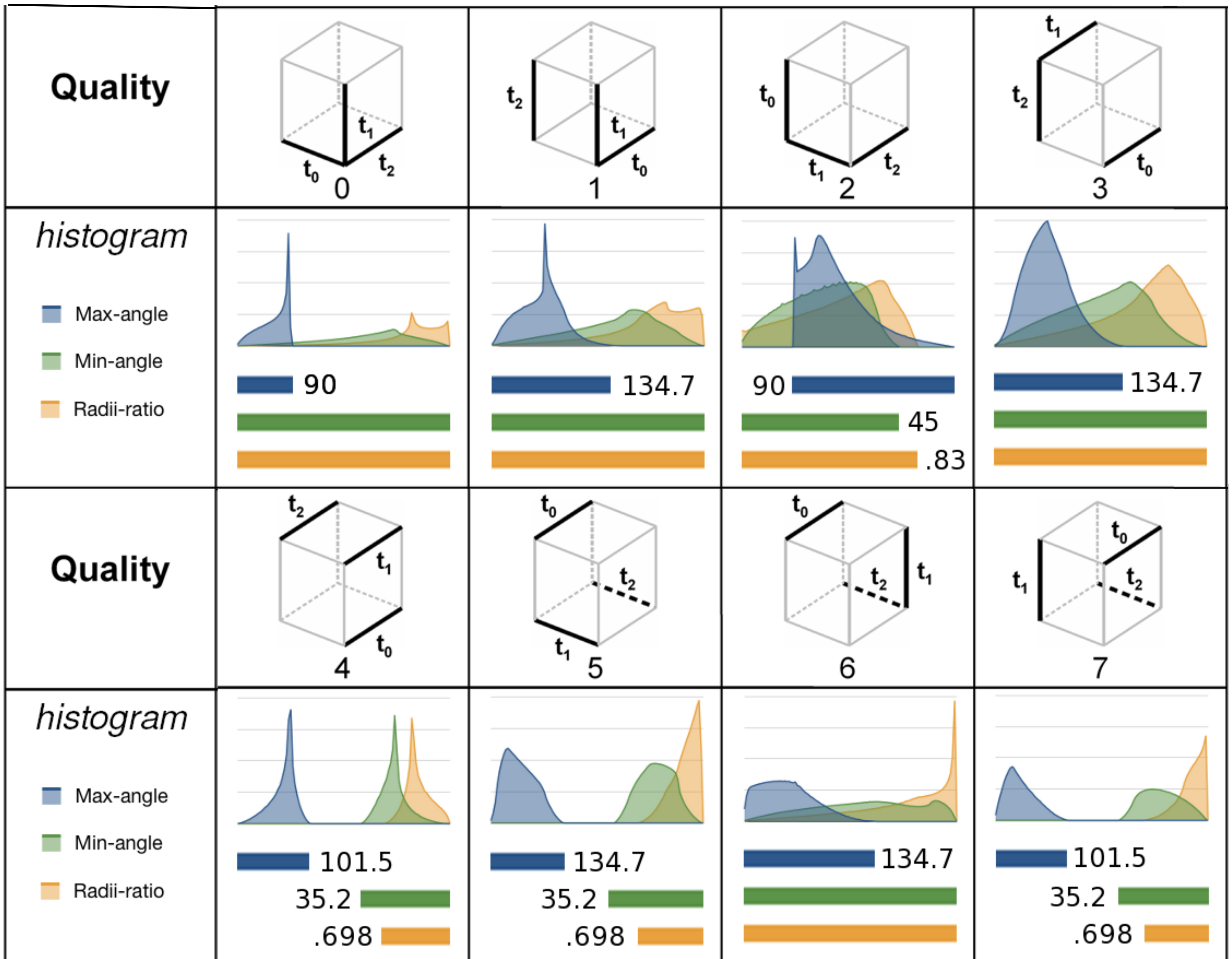
Dietrich, Scheidegger, Comba, Nedel, Silva
Edge Groups: An Approach to Understanding the Mesh Quality of Marching Methods

“Give us a MC table, and we’ll give you a diagnostic”

Edge Groups

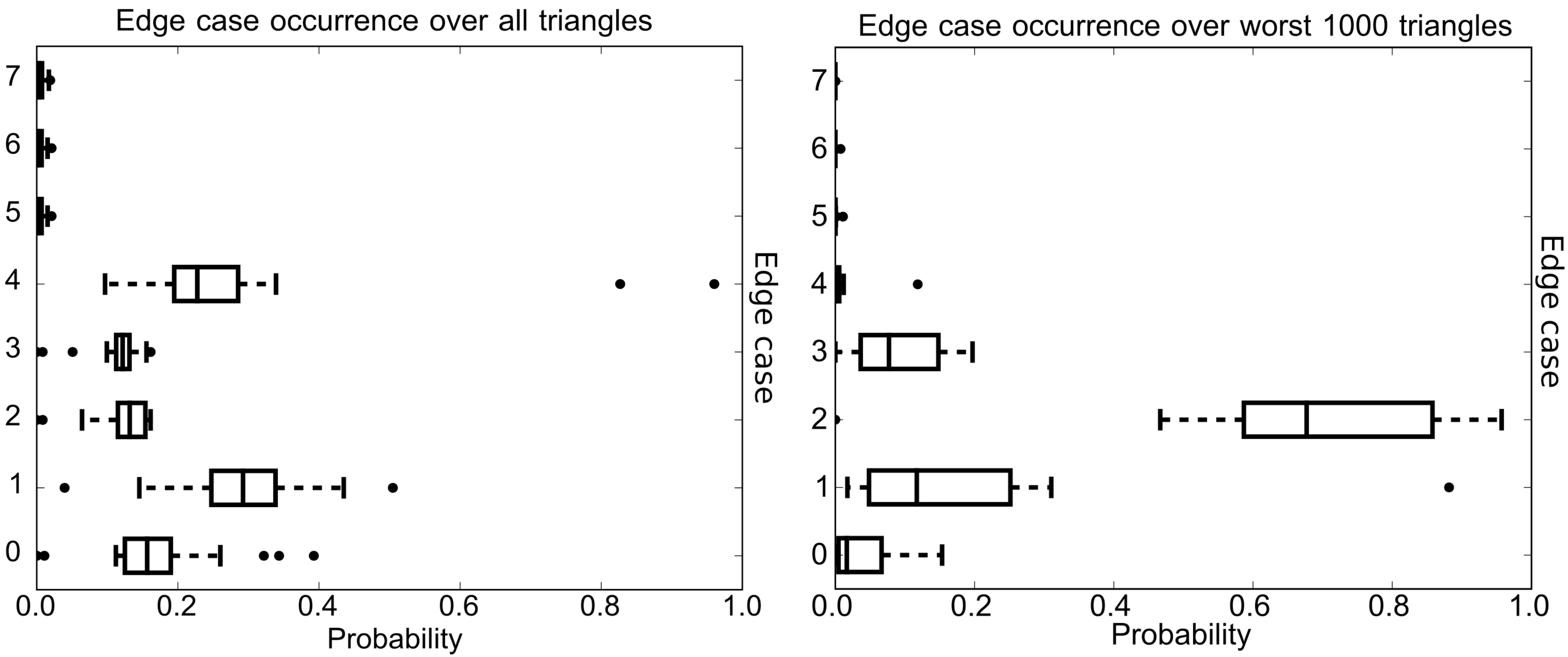


Edge Groups



Quality
histogram
Max-angle
Min-angle
Radii-ratio
0
1
2
3
90
134.7
90
45
.83
134.7
Quality
histogram
Max-angle
Min-angle
Radii-ratio
4
5
6
7
101.5
35.2
.698
134.7
35.2
.698
134.7
101.5
35.2
.698

Edge Groups

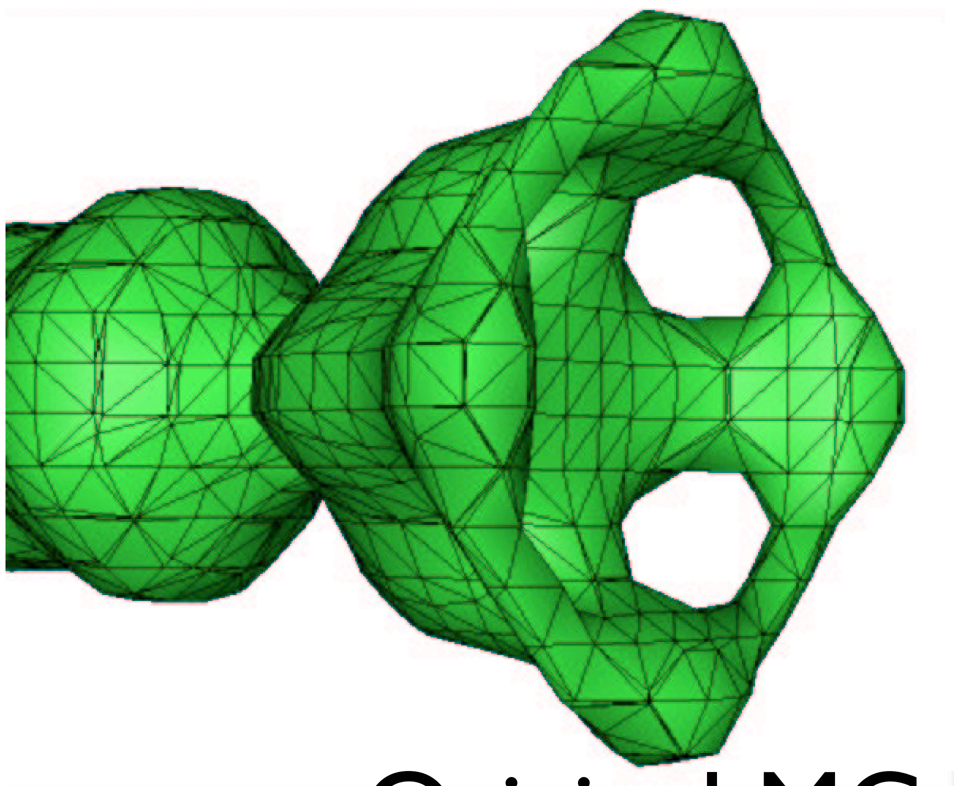


Edge Group 2 is responsible for most bad triangles!

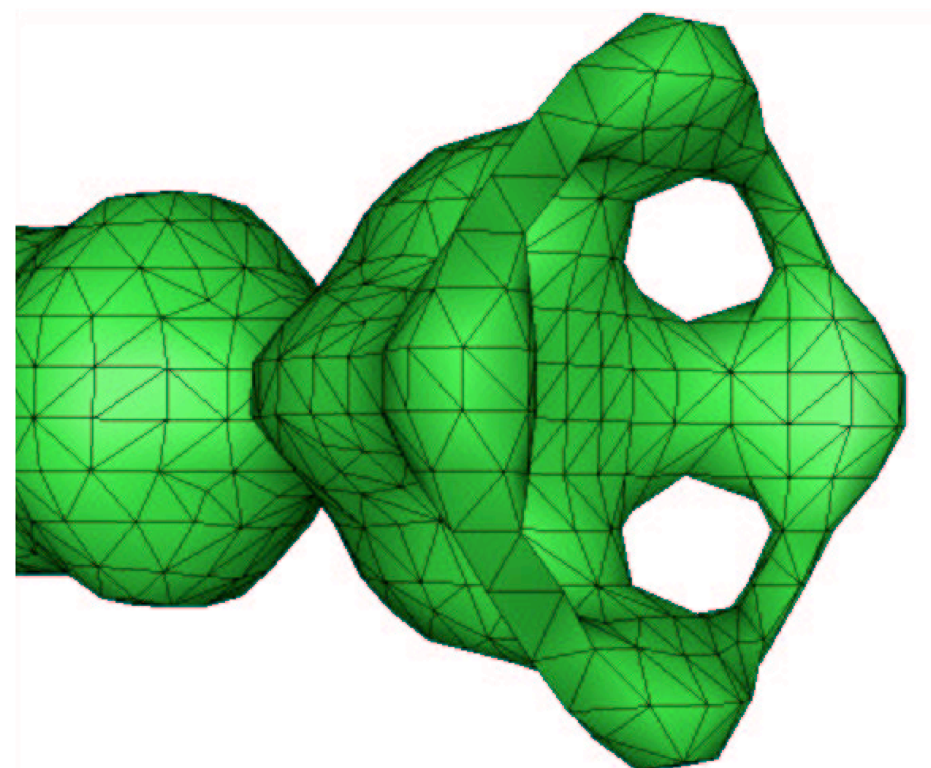
Edge Groups

Name	MC with old table			Macet with new table		
	θ_0	θ_∞	ρ	θ_0	θ_∞	ρ
Chest CT	0.08	179.0	0.0	17.9	118.6	0.46
Bonsai	0.38	178.7	0.0	17.6	119	0.45
Shockwave	1.26	175.7	0.0	20.7	110.7	0.52
Silicium	0.66	177.4	0.0	18.7	117.3	0.47

SnapMC: Extended MC Table



Original MC



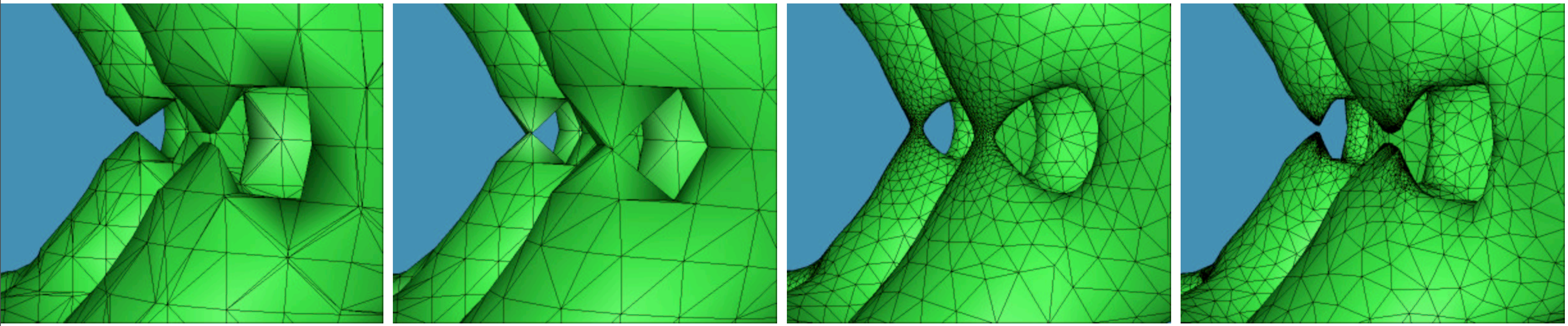
SnapMC

- vertices can be "+", "-", **or "="**
  - **snap scalars close to the isovalue**
- How big is this table? 3^8 entries! 6561

SnapMC: Extended MC Table

- Automatically generated, by the method we described before: easy!
- One extra parameter: how aggressive are we with snapping?
- **Provable** quality bounds
  - But triangle mesh non-manifold..

SnapMC Results



a) Marching Cubes b) SnapMC c) Afront d) DelIso

SnapMC

	isovalue	min edge length	min area	min angle	max angle	min radius ratio	directed Hausdorff
aneurism	100	0.425	0.078	13.09	135.20	0.29	0.86
bonsai	30	0.427	0.083	13.35	135.67	0.28	0.86
engine	100	0.428	0.080	13.96	134.71	0.26	0.66
fuel	80	0.428	0.104	14.30	135.51	0.39	0.30
lobster	20	0.428	0.087	13.55	135.13	0.25	0.86
Marschner-Lobb	100	0.442	0.231	14.58	122.63	0.35	0.71

Name	MC with old table			Macet with new table		
	θ_0	θ_∞	ρ	θ_0	θ_∞	ρ
Chest CT	0.08	179.0	0.0	17.9	118.6	0.46
Bonsai	0.38	178.7	0.0	17.6	119	0.45
Shockwave	1.26	175.7	0.0	20.7	110.7	0.52
Silicium	0.66	177.4	0.0	18.7	117.3	0.47

Edge groups

SnapMC and Edge Groups

- Edge Groups can probably be used to illustrate SnapMC's bounds

- Macet seems to have better experimental results, but has no provable bounds