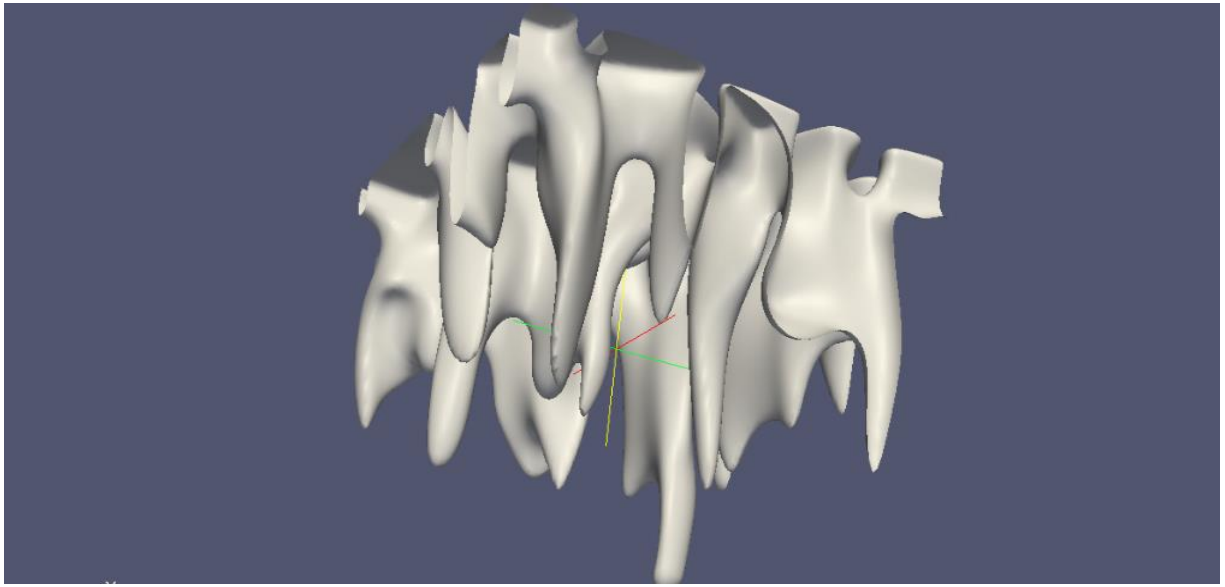
Generalized Coons Surfaces: Iso-Surfaces

Alyn Rockwood
Boulder Graphics LLC

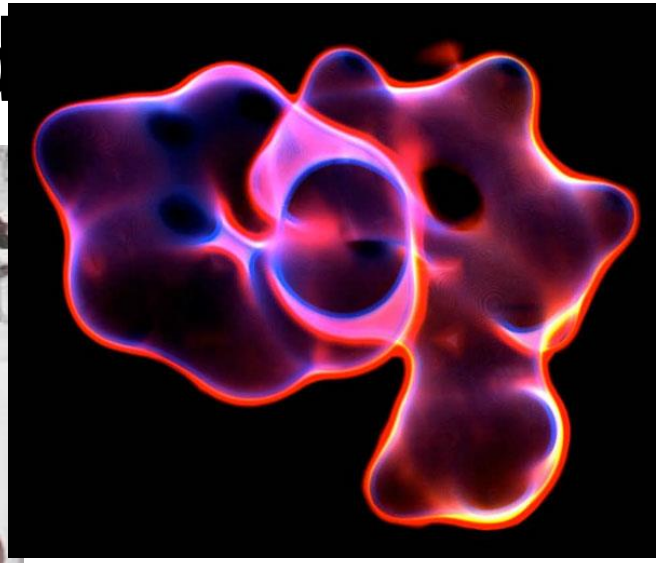


Work in progress

Isosurfaces

Fundamental viewing of scientific d

Biomed -MRI, CT, PET



Chemistry
Pharmaceutical

Geoscience

The most cited SIGGRAPH paper?

(The Rendering Equation, Ray-Tracing, Volume Rendering…?)

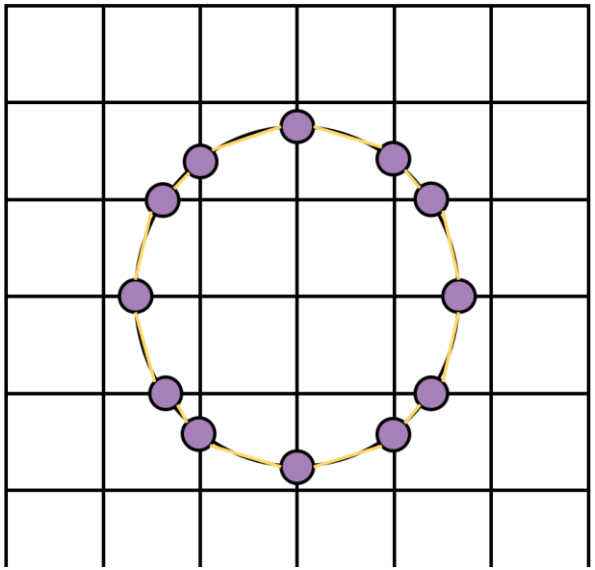
The most cited SIGGRAPH paper?

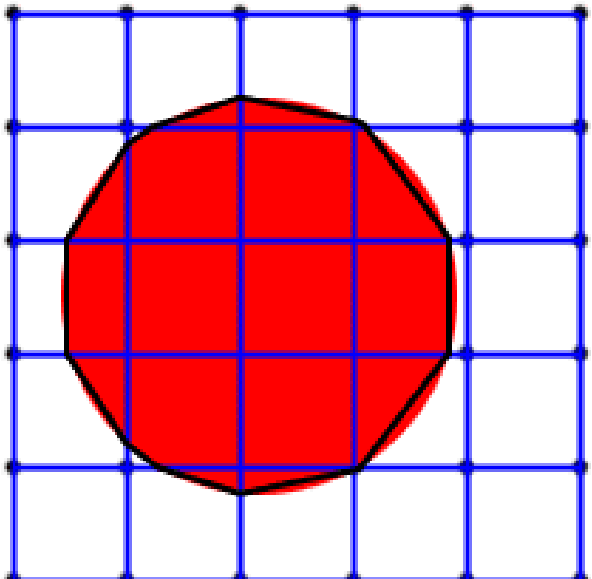
Marching Cubes!

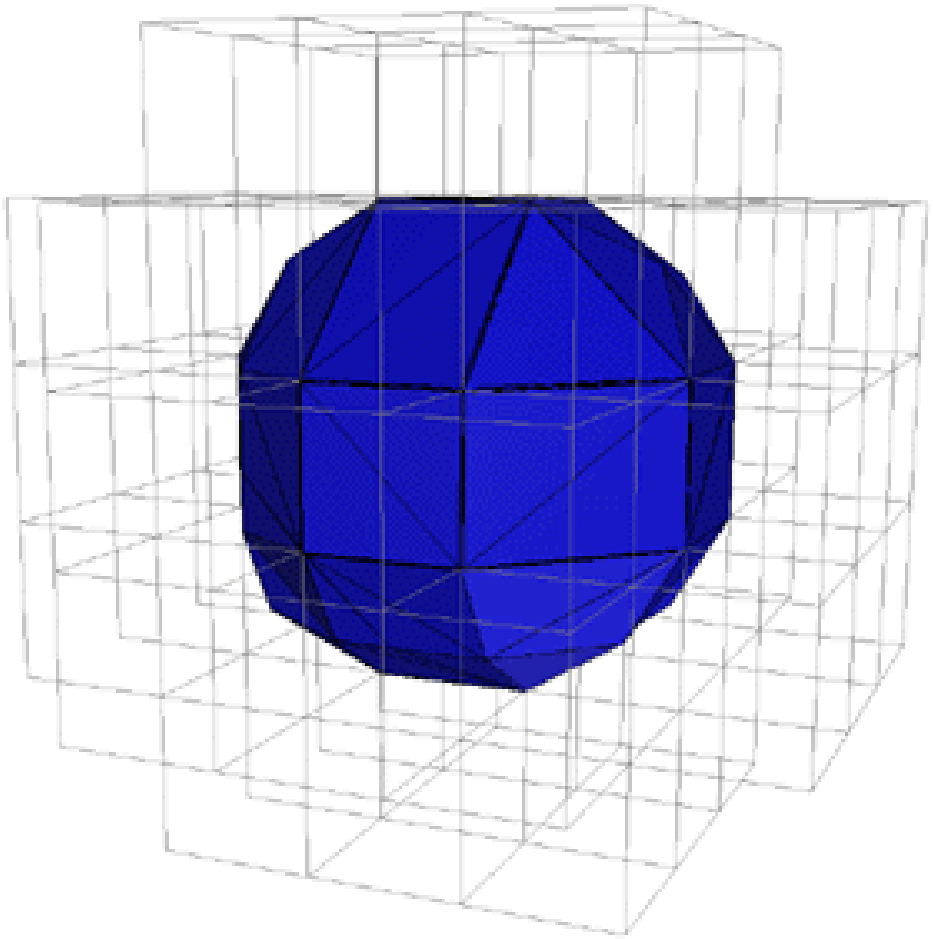
Marching cubes: A high resolution 3D surface construction **...**
dl.acm.org/**citation**.cfm?id=37422

by WE Lorensen - 1987 - Cited by 10444 - Related articles

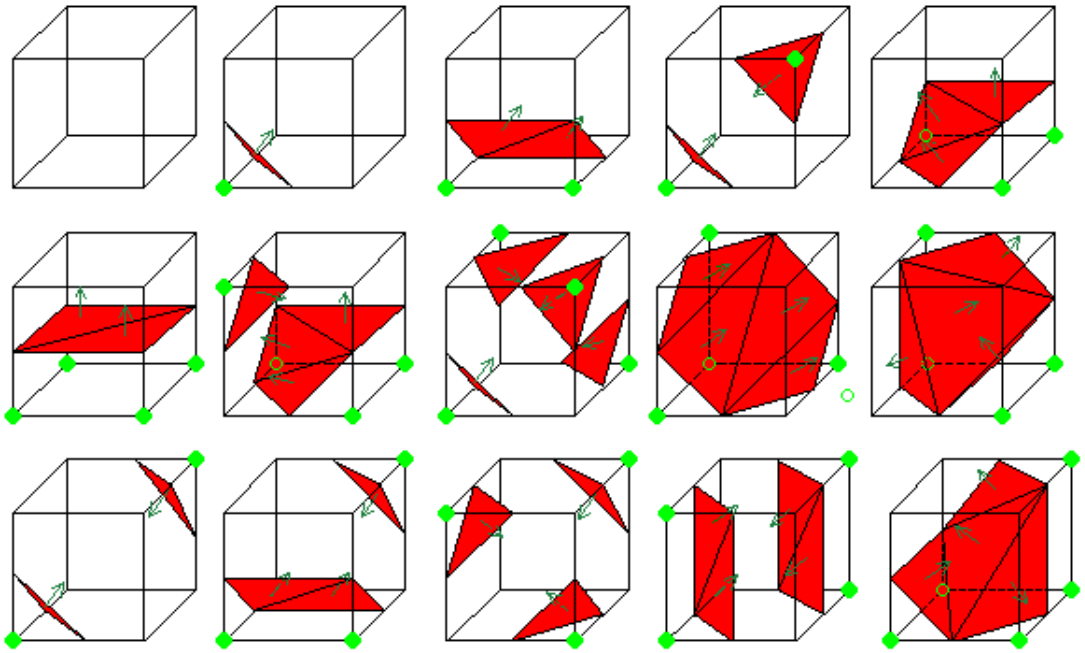
Review Marching Cubes:

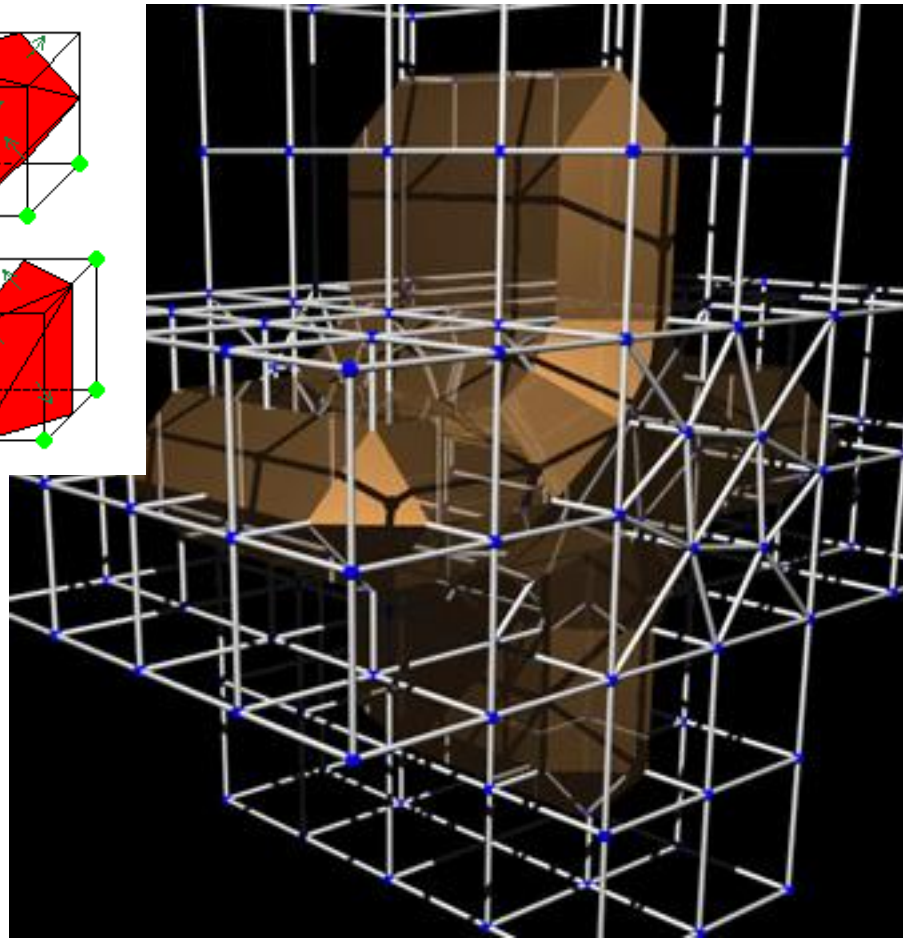




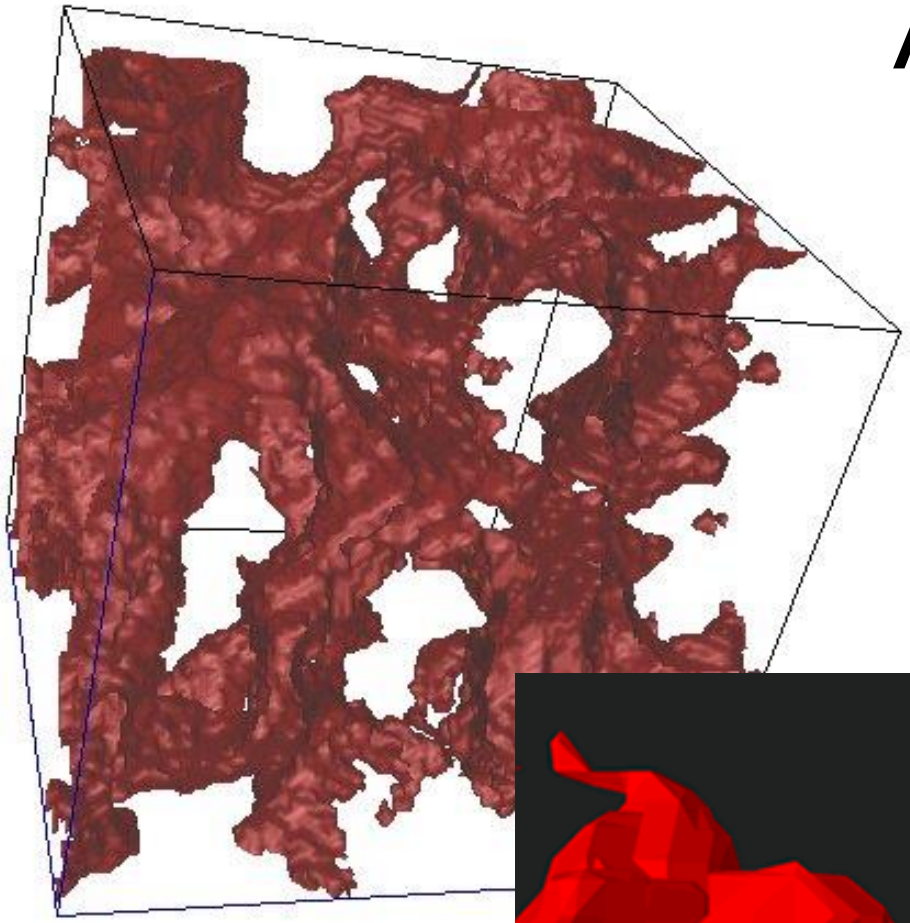


Review Marching Cubes:

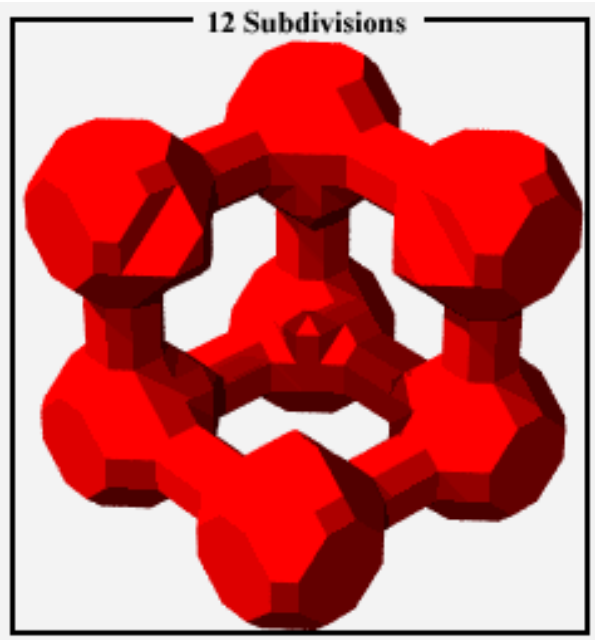




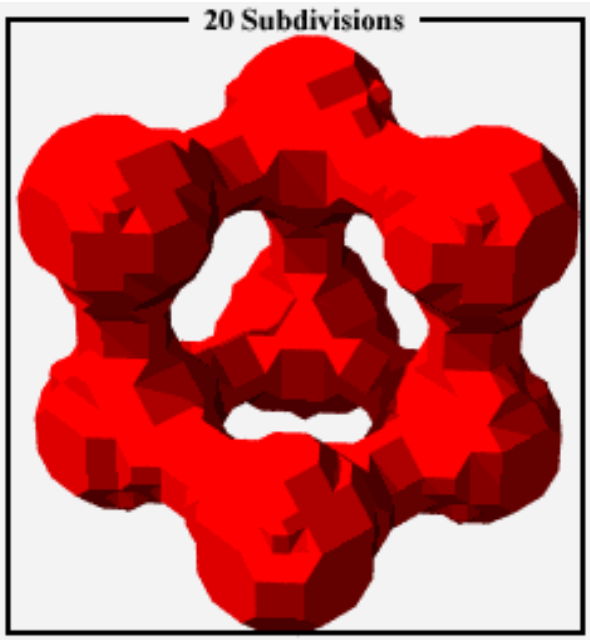
Marching Cubes: Angularities and Serrated edges



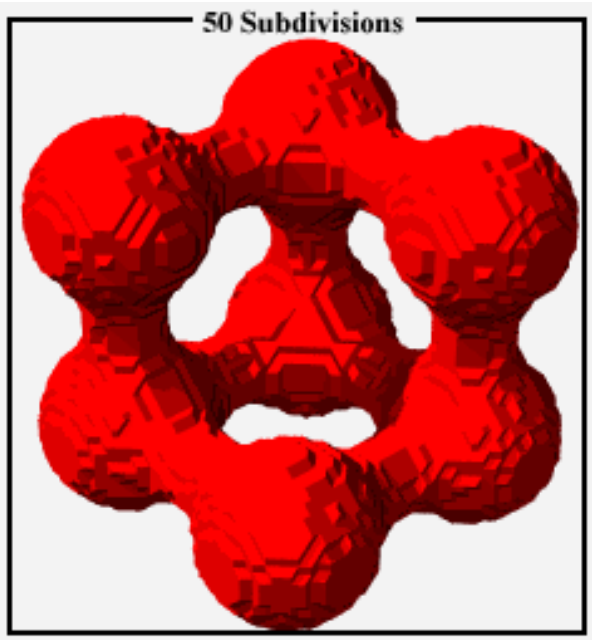
Issues with Marching Cubes: Large Database



1700 cubes
550 triangles

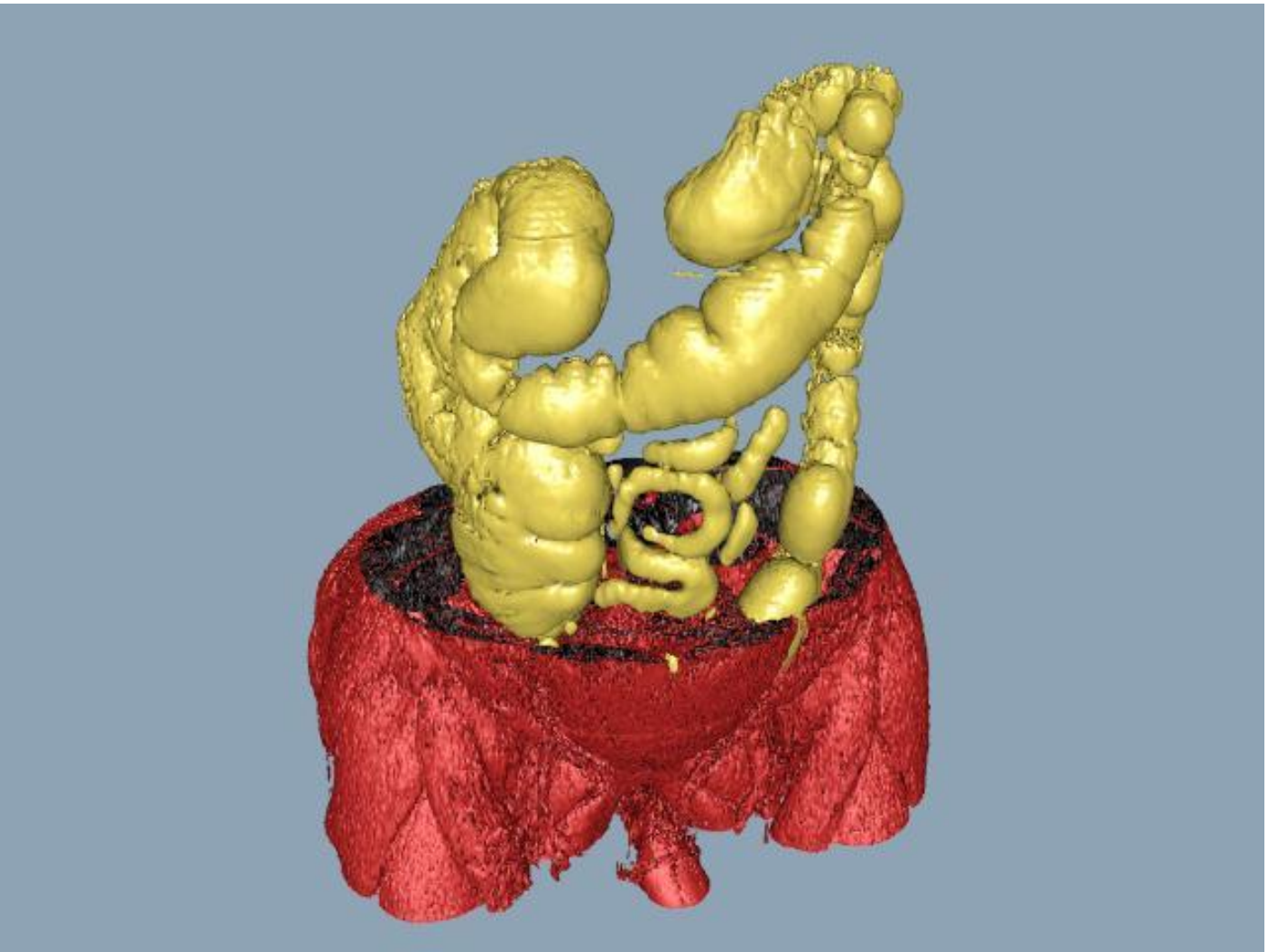


8000 cubes
1300 triangles



125000 cubes
75000 triangles

Issues with Marching Cubes: Large Database



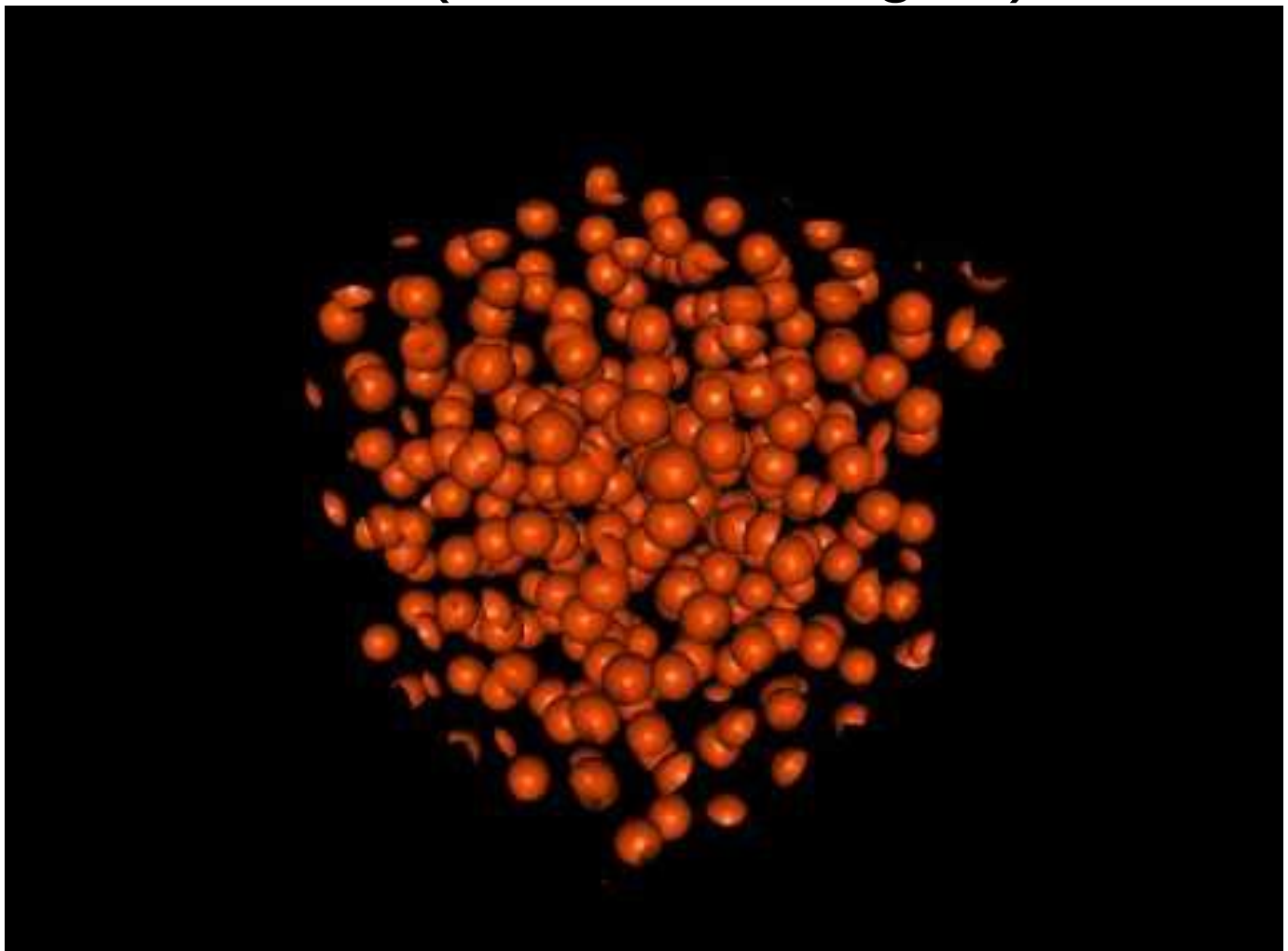
64,000,000 cubes

540,000 triangles

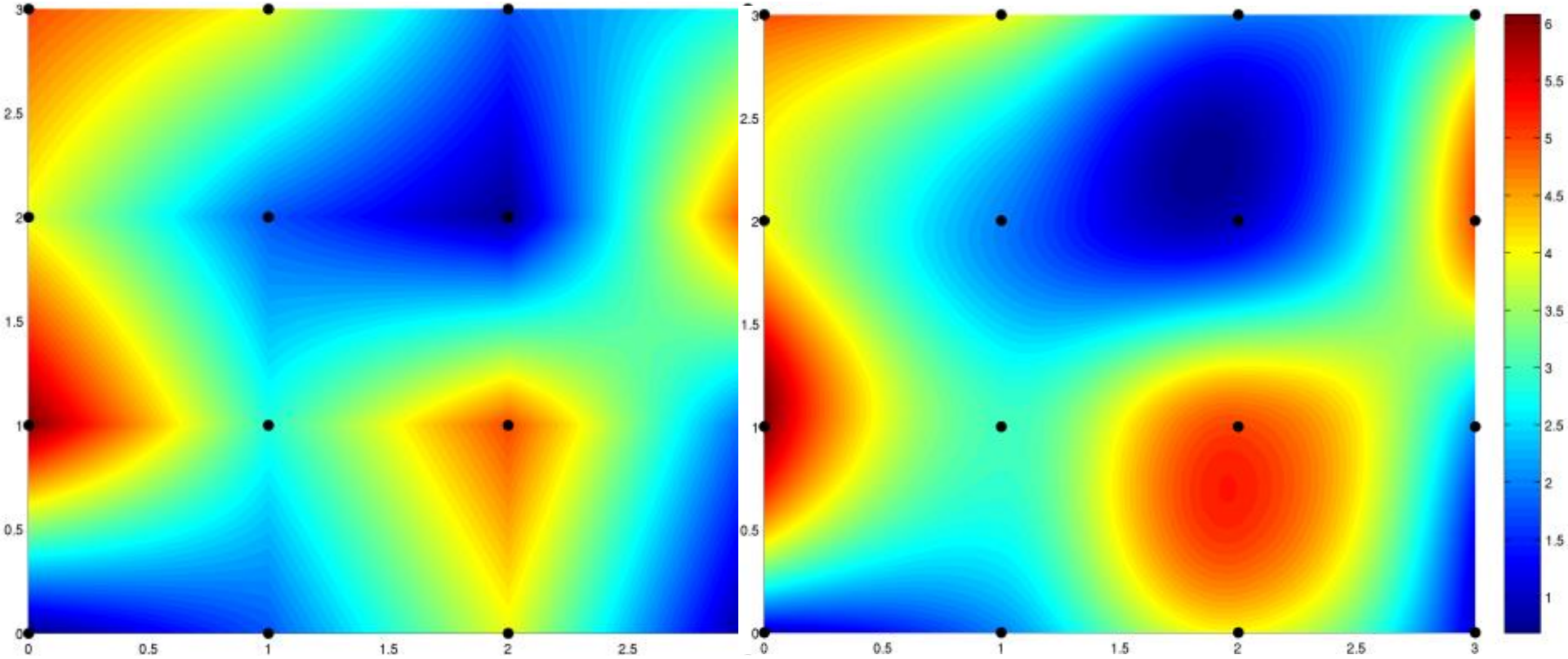
Issues with Marching Cubes:

N^2 < Large Database < N^3

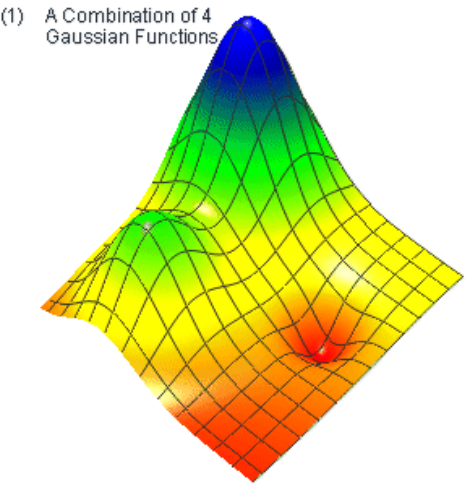
(900,000 triangles)

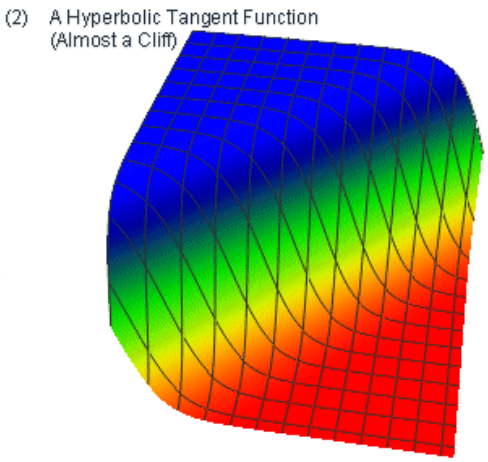


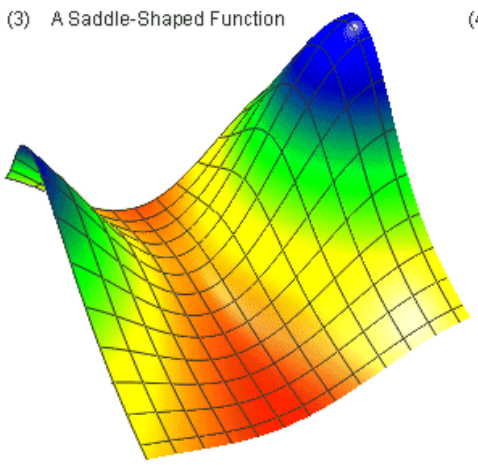
Linear vs higher order interpolation

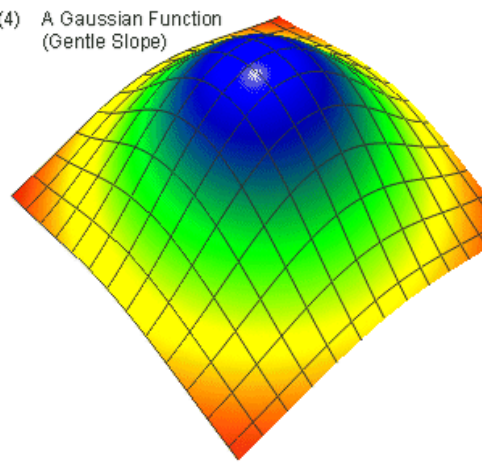


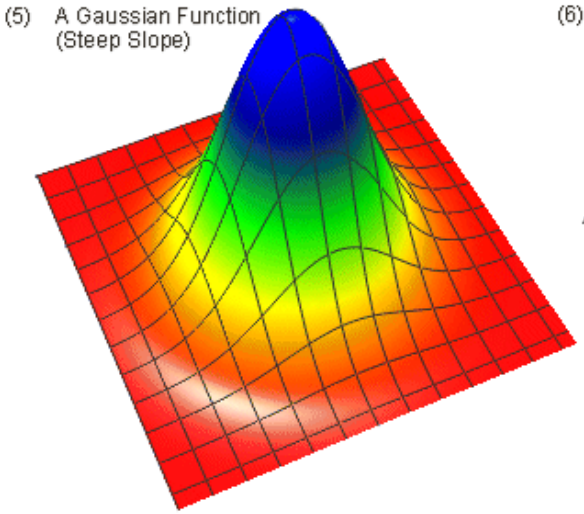
Franke Test set

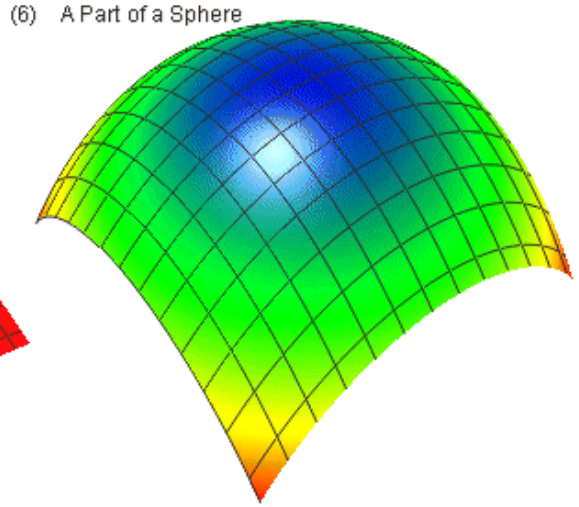












Interpolation error variance

Interpolation Algorithm	Function 1	Function 2	Function 3	Function 4	Function 5	Function 6	Overall Ranking
Level Plane	13	13	13	13	13	13	13
Linear NW-SE	12	9	11	11	11	12	11
Linear NE-SW	11	12	12	12	12	11	12
Double Linear	10	11	10	10	10	10	10
Bilinear	9	10	9	9	9	9	9
Biquadratic (8)	7	7	6	7	6	7	7
Biquadratic (9)	6	5	5	6	5	6	6
Jancaitis	8	8	8	8	8	8	8
Cubic	3	3	3	2	2	2	2
Bicubic (12)	5	6	4	4	4	3	4
Bicubic (16)	4	4	2	3	3	1	3
Bicubic (Akima)	1	1	7	5	7	5	4
Biquintic	2	2	1	1	1	4	1

Interpolation error variance

Interpolation Algorithm	Function 1	Function 2	Function 3	Function 4	Function 5	Function 6	Overall Ranking
Level Plane	13	13	13	13	13	13	13
Linear NW-SE	12	9	11	11	11	12	11
Linear NE-SW	11	12	12	12	12	11	12
Double Linear	10	11	10	10	10	10	10
Bilinear	9	10	9	9	9	9	9
Biquadratic (8)	7	7	6	7	6	7	7
Biquadratic (9)	6	5	5	6	5	6	6
Jancaitis	8	8	8	8	8	8	8
Cubic	3	3	3	2	2	2	2
Bicubic (12)	5	6	4	4	4	3	4
Bicubic (16)	4	4	2	3	3	1	3
Bicubic (Akima)	1	1	7	5	7	5	4
Biquintic	2	2	1	1	1	4	1

Issues with Marching Cubes: $N^2 < \text{Large Database} < N^3$

iPads
smart phones
cloud computing

collaborative work

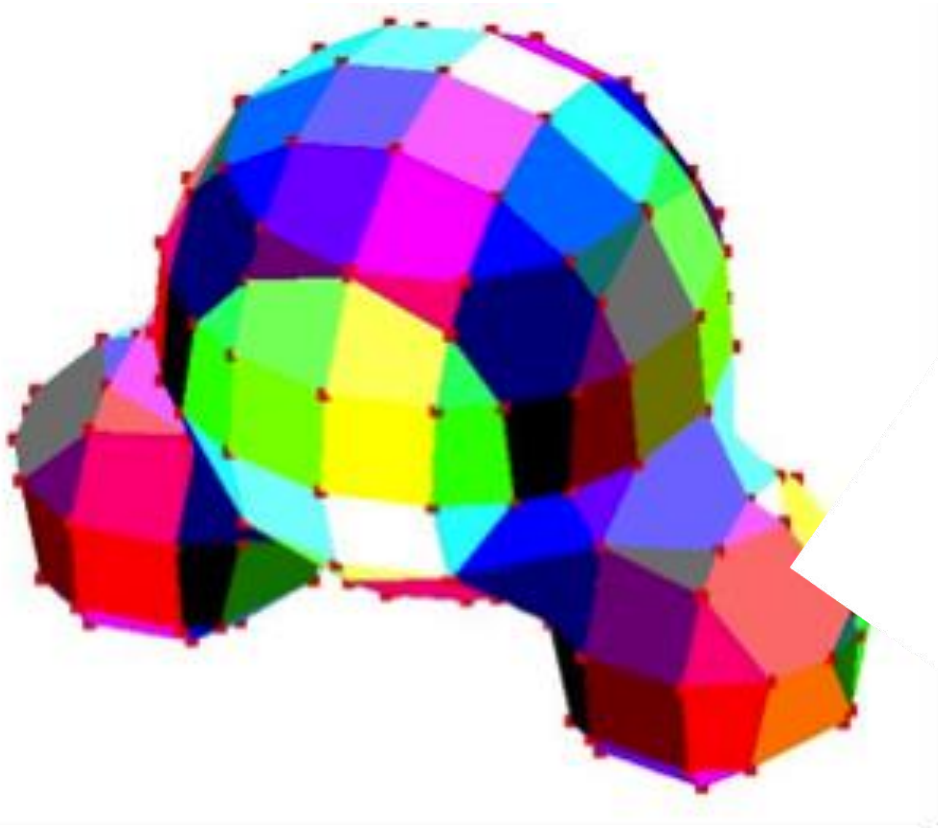
Issues with Marching Cubes: N^2 < Large Database < N^3

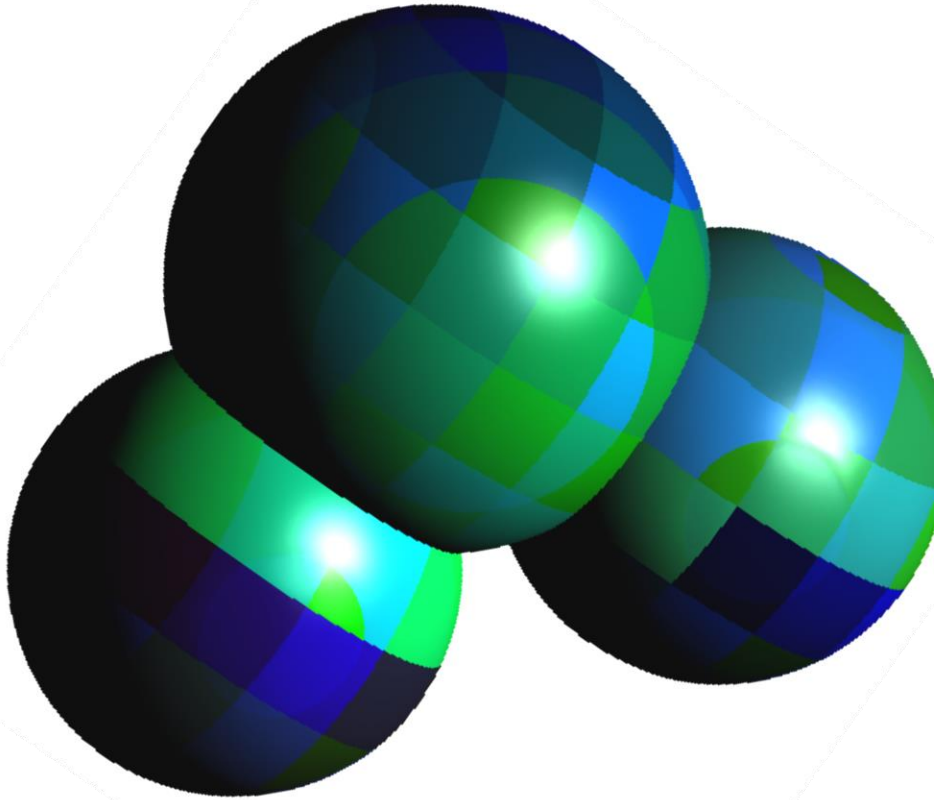
iPads
smart phones
cloud computing

collaborative work

Compression!

N-sided Surface Patches





Previously

Bi-cubic interpolation
Tri-linear Interpolation
Quadratic Bernstein-Bezier

Previously

Bi-cubic interpolation
Tri-linear Interpolation
Quadratic Bernstein-Bezier
NOT (that) compact!

Previously

Bi-cubic interpolation
Tri-linear Interpolation
Quadratic Bernstein-Bezier
NOT (that) compact!
…and no more smoother (as a rule)

Previously

Smooth, G^1 isosurfaces
uses reparametrization

Exact Isosurfaces for Marching Cubes
Holger Theisel

Computer Graphics Forum
Volume 21, **Issue 1**, **pages 19–32**, **March 2002**

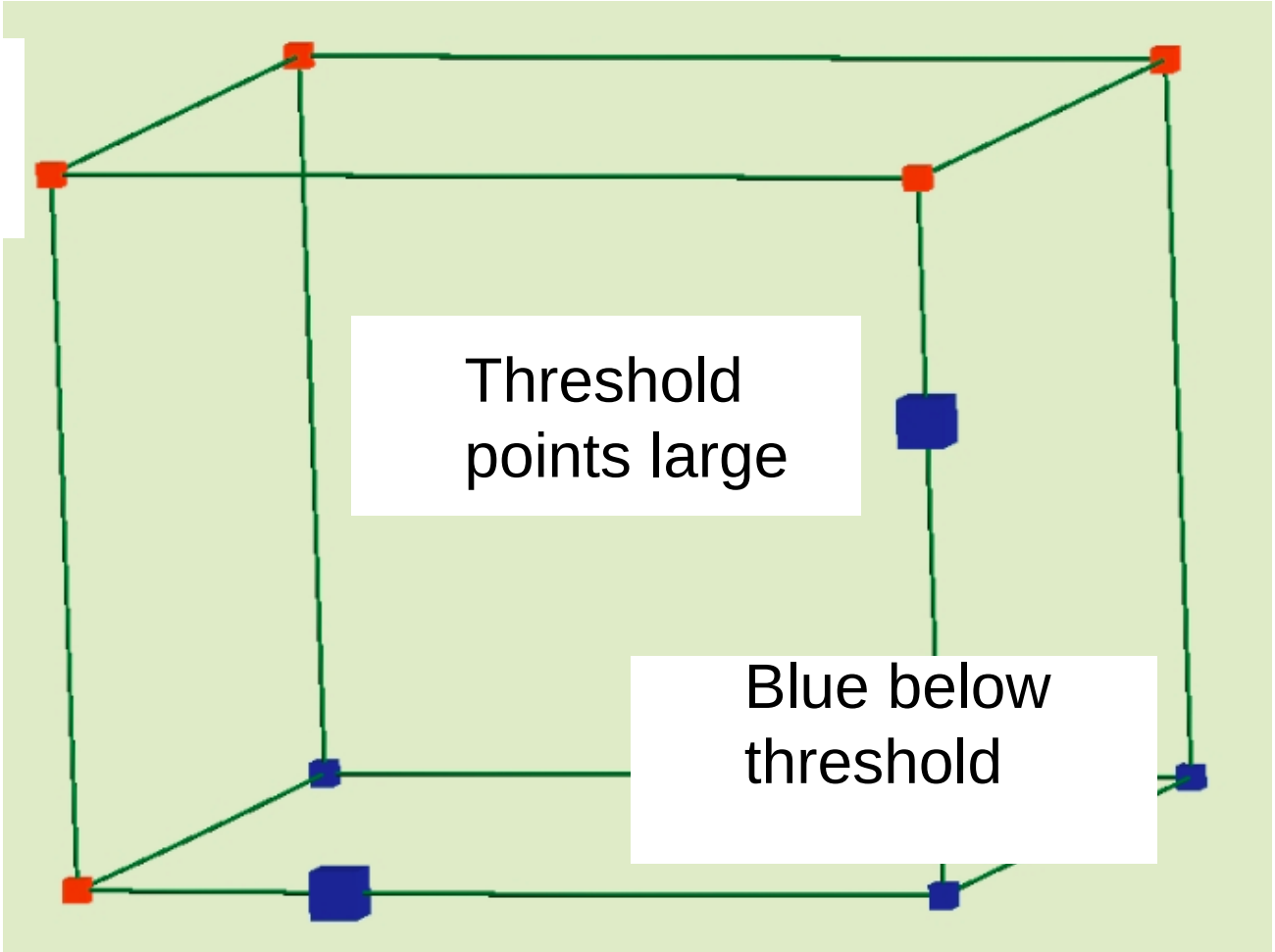
N-sided Surface Patches

- Create **threshold points** by weight-based corner point interpolation.
- Create **normals** at corners from implicit function (gradient), or from central difference approximation. Interpolate to threshold points.
- Create **tangents** from normals and axial planes.
- Create **boundary curves** with points and tangents (cubic interpolation on face).
- Create **cross-tangents** function for adjacent patch smoothness (cubic interpolation on 4-sided face)
- Create **face loops** of curves (disambiguate)
- Create approximate and fair **surface patches** for faces (center point interpolate)
- Adaptively **refine**, if needed

N-sided Surface Patches

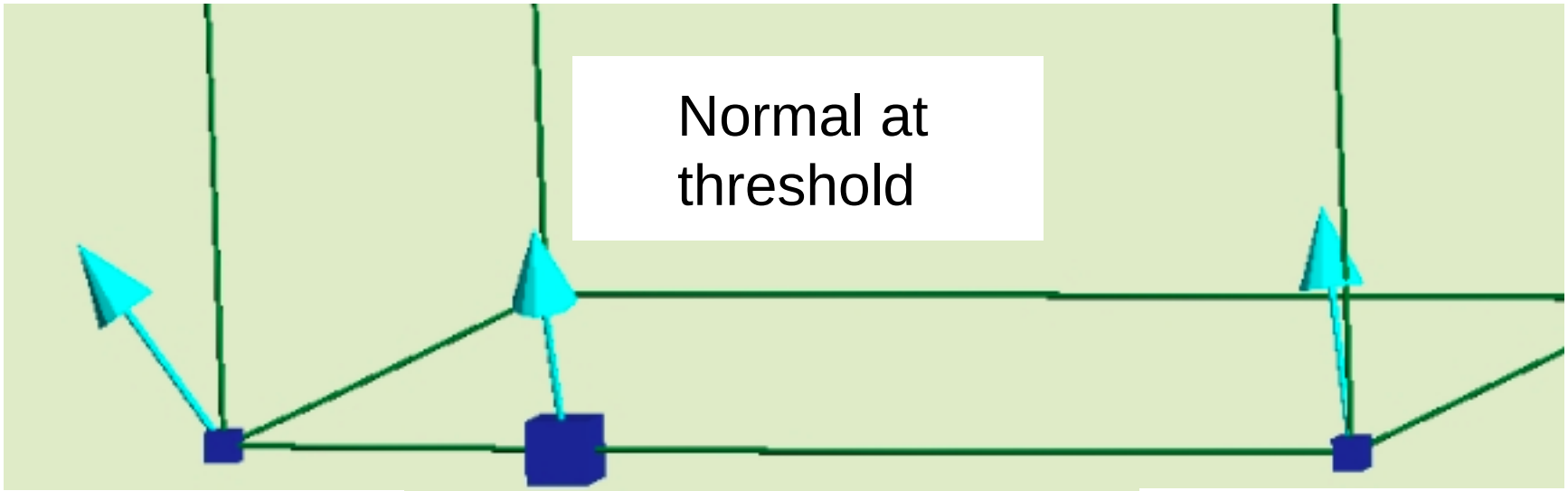
- Create **threshold points** by weight-based corner point interpolation.

Red above threshold



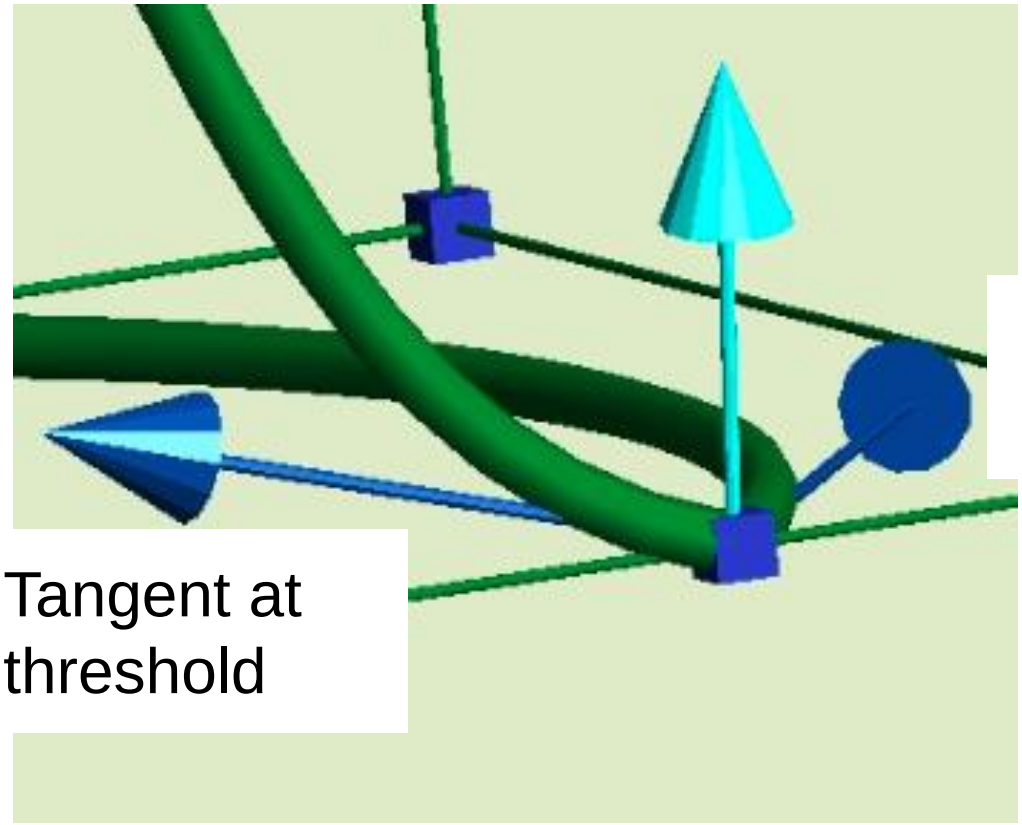
N-sided Surface Patches

Create **normals** at corners from implicit function (gradient), or from central difference approximation. Interpolate to threshold points.



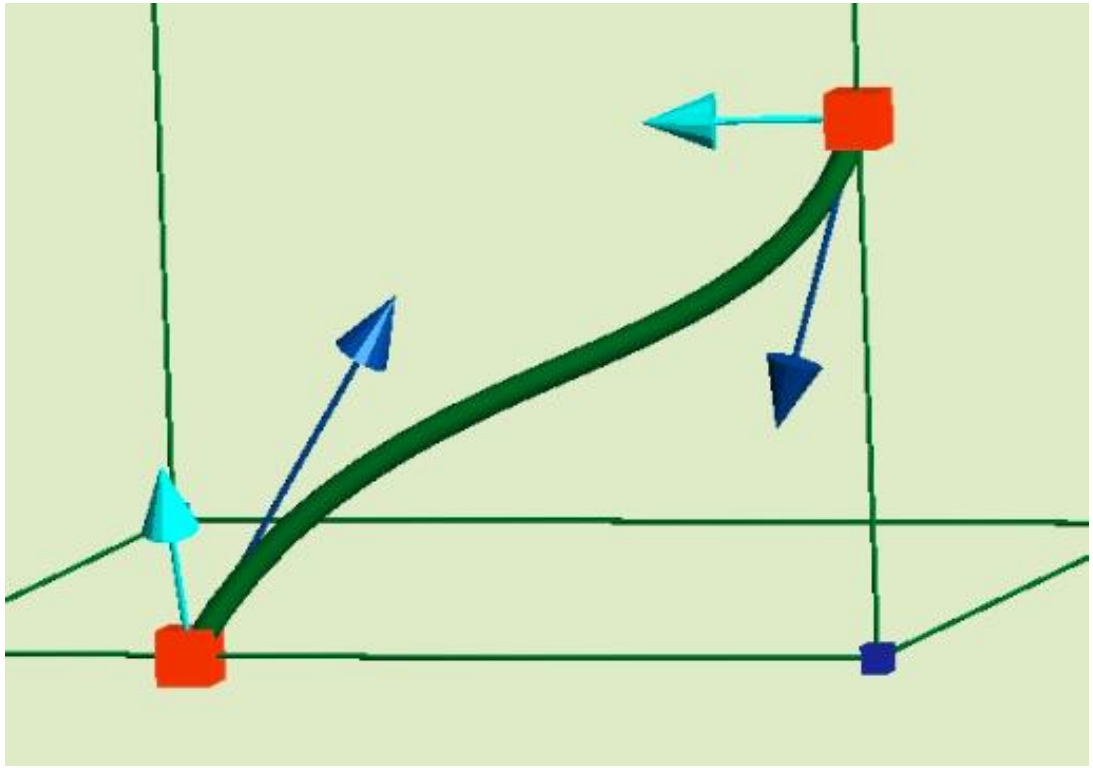
N-sided Surface Patches

- Create (two) **tangents** for curves from normals and axial planes.



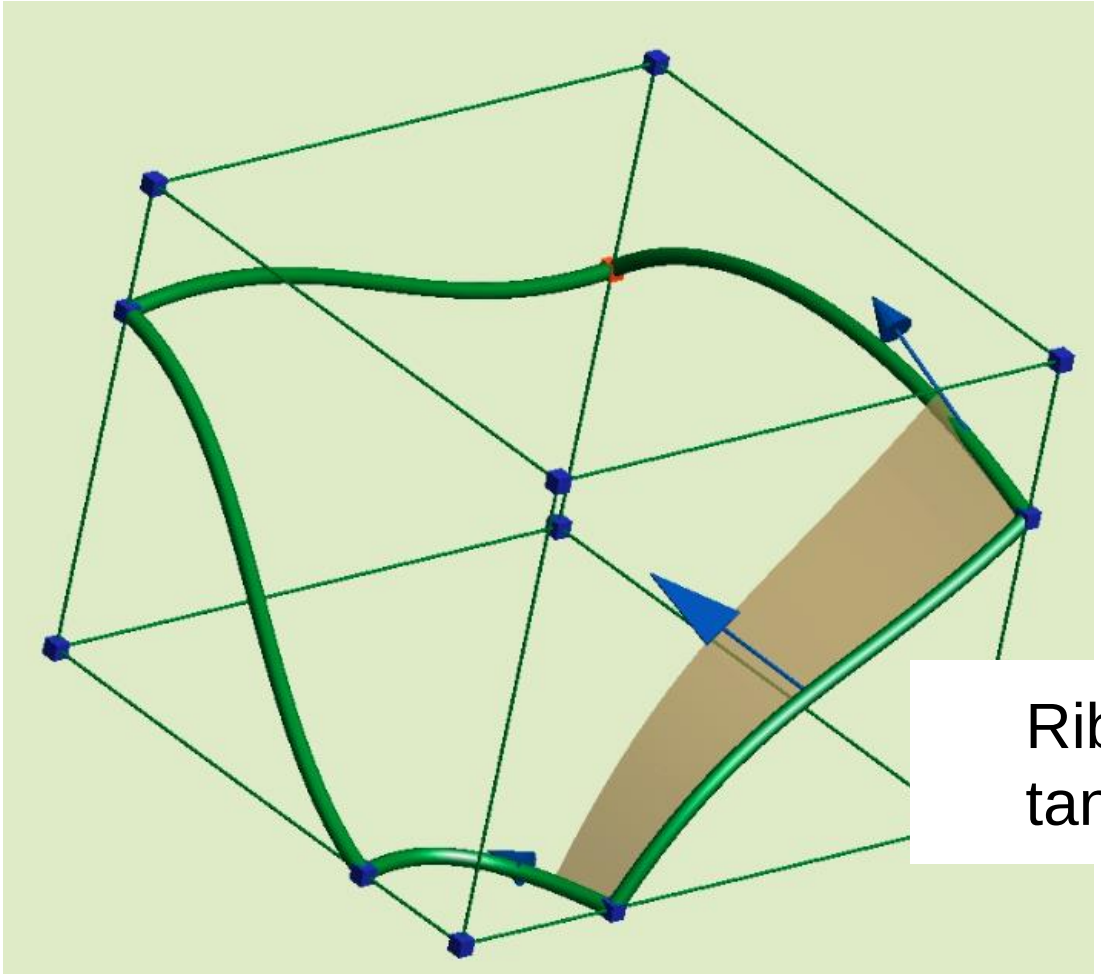
N-sided Surface Patches

Create **boundary curves** with points and tangents (cubic interpolation on face).



N-sided Surface Patches

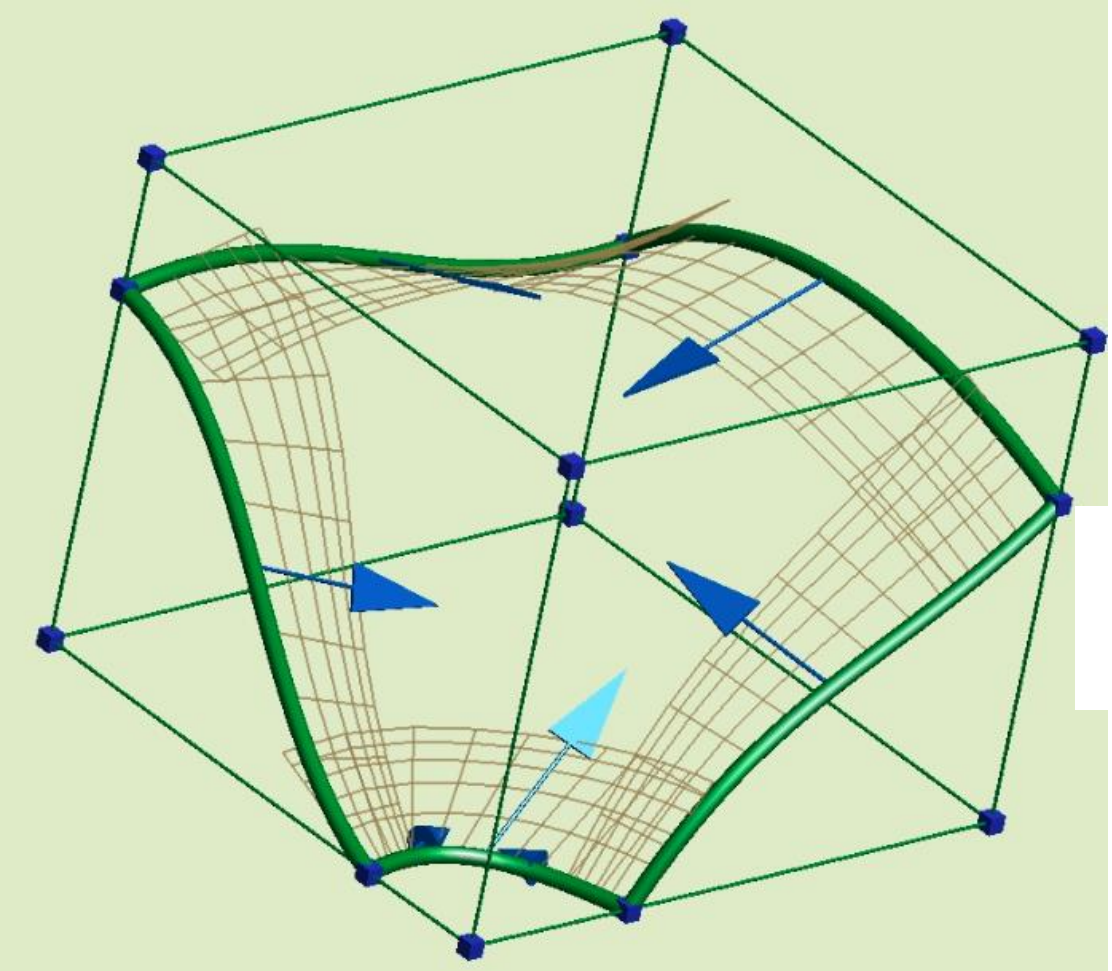
- Create **cross-tangents** function for adjacent patch smoothness (cubic interpolation on 4-sided face)



Ribbon from tangents

N-sided Surface Patches

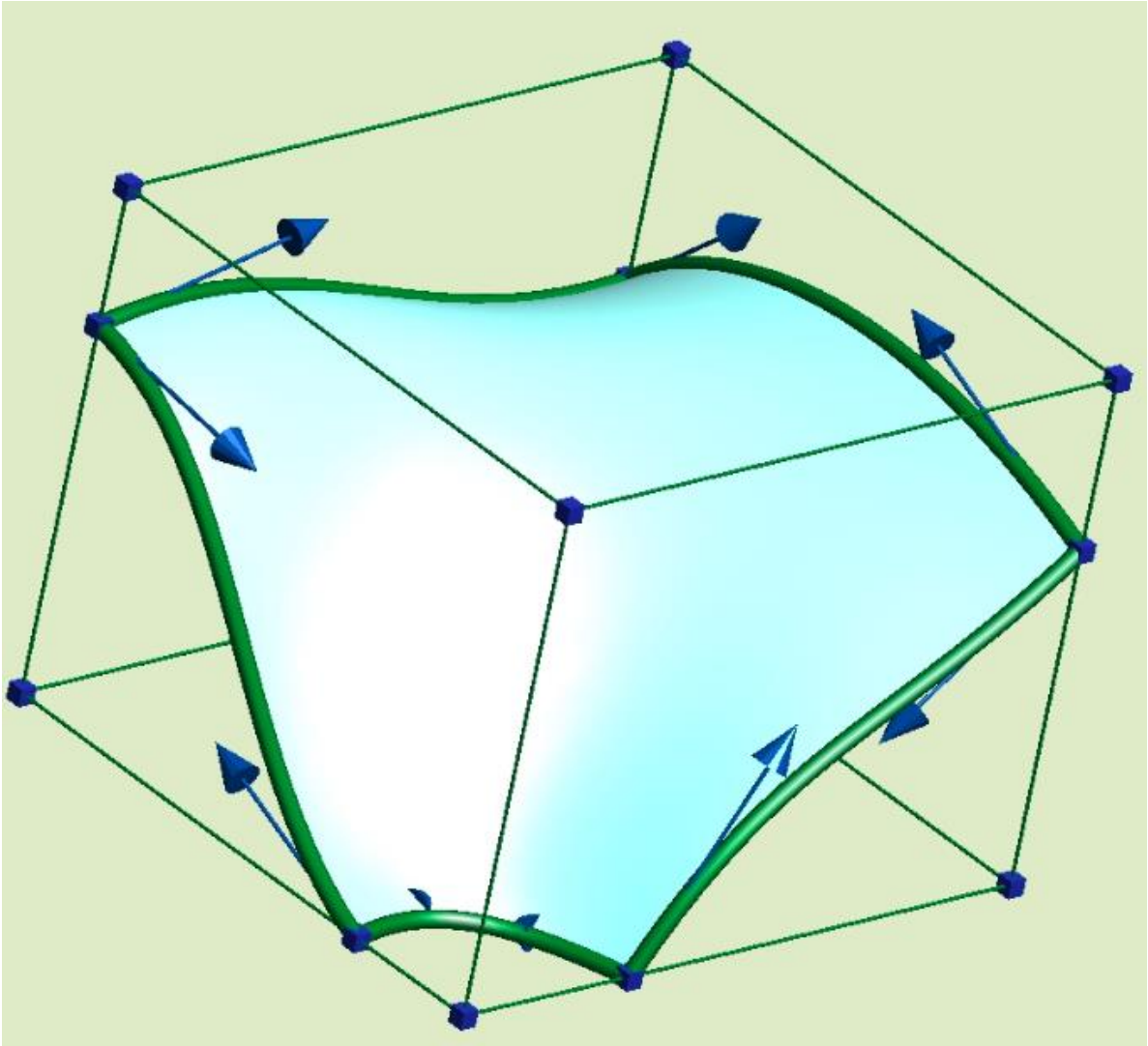
- Create **face loops** of curves (disambiguate)



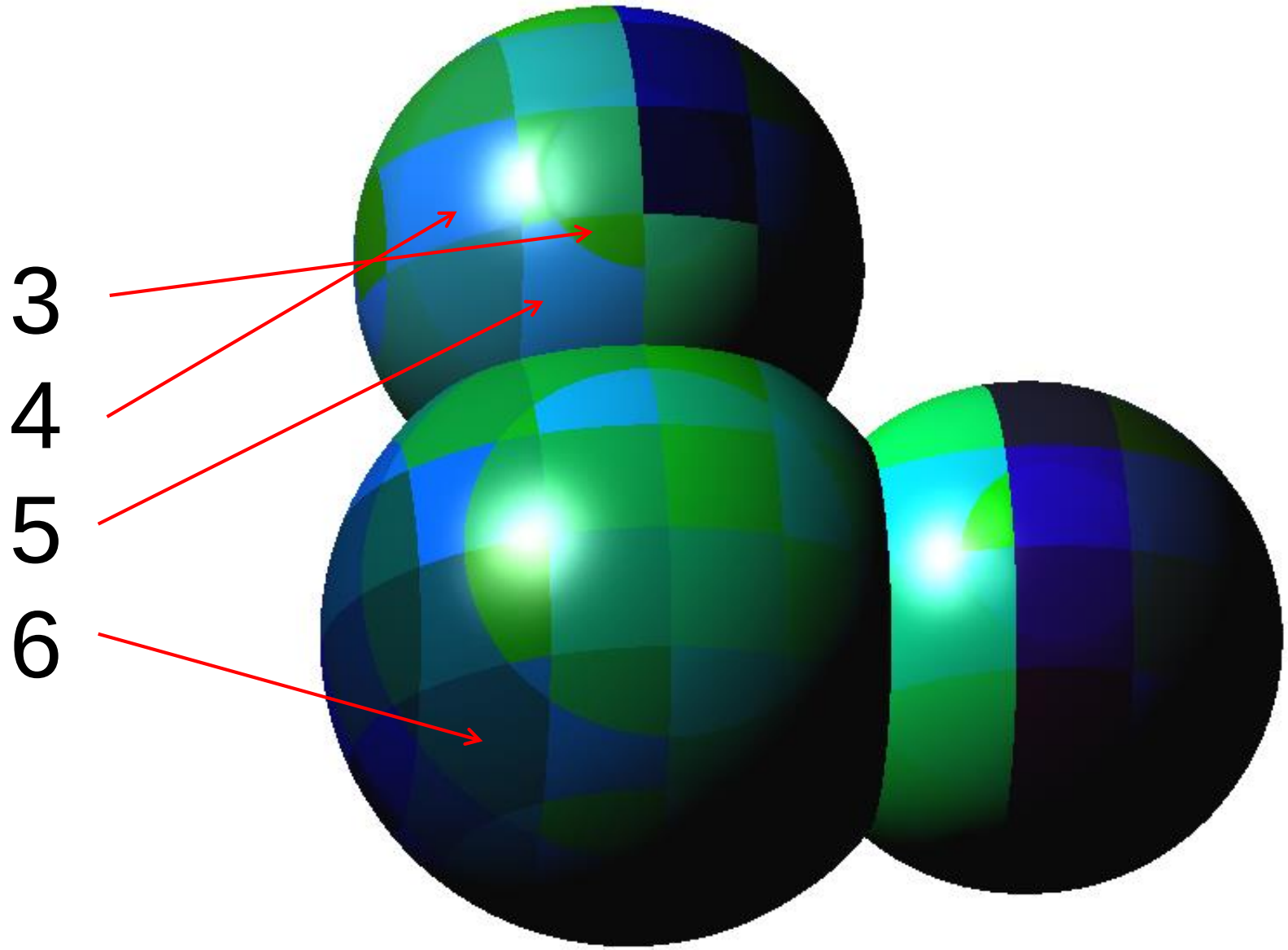
Five ordered curves with ribbons

N-sided Surface Patches

- Create approximate and fair **surface patch** for face.



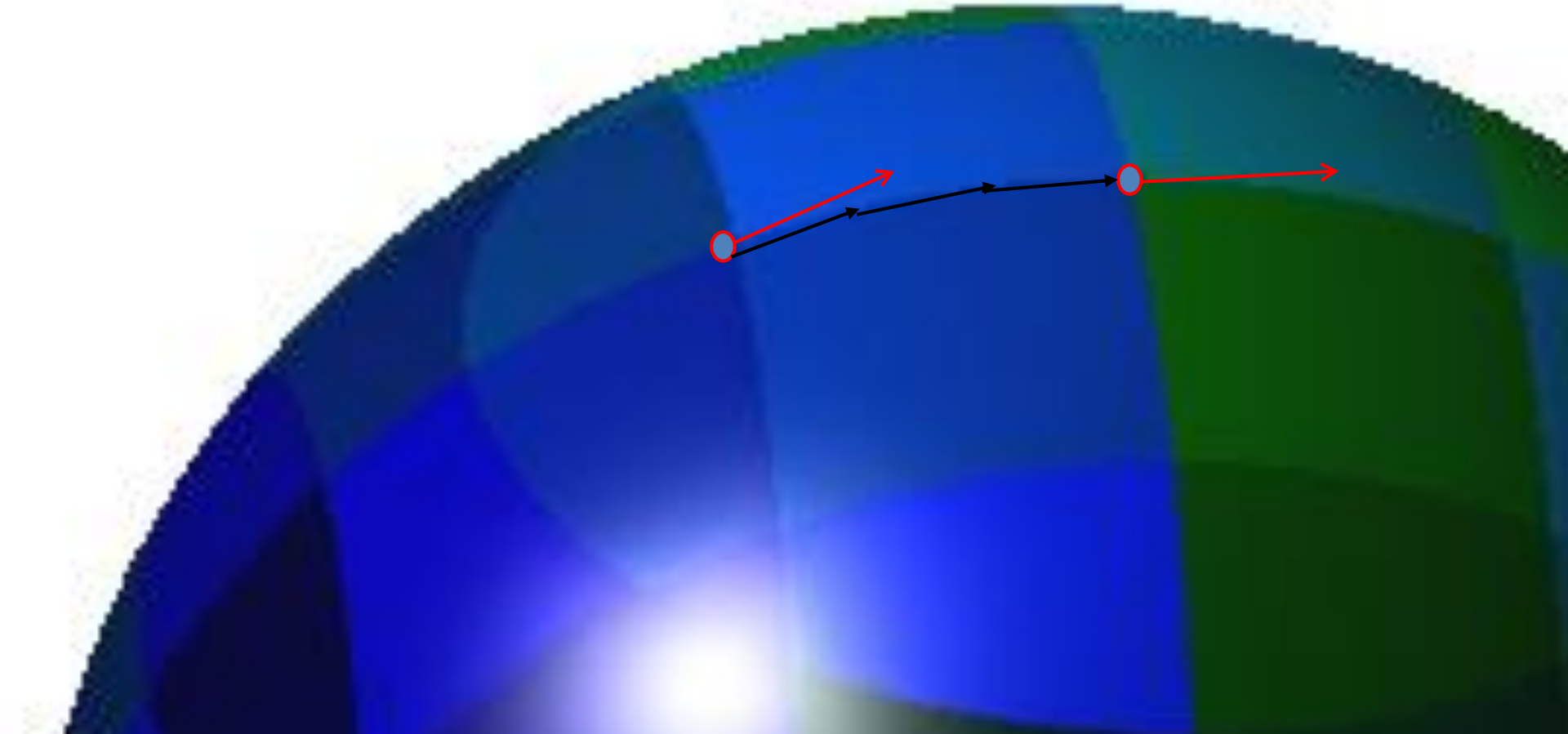
Examples on spheres



Examples on spheres

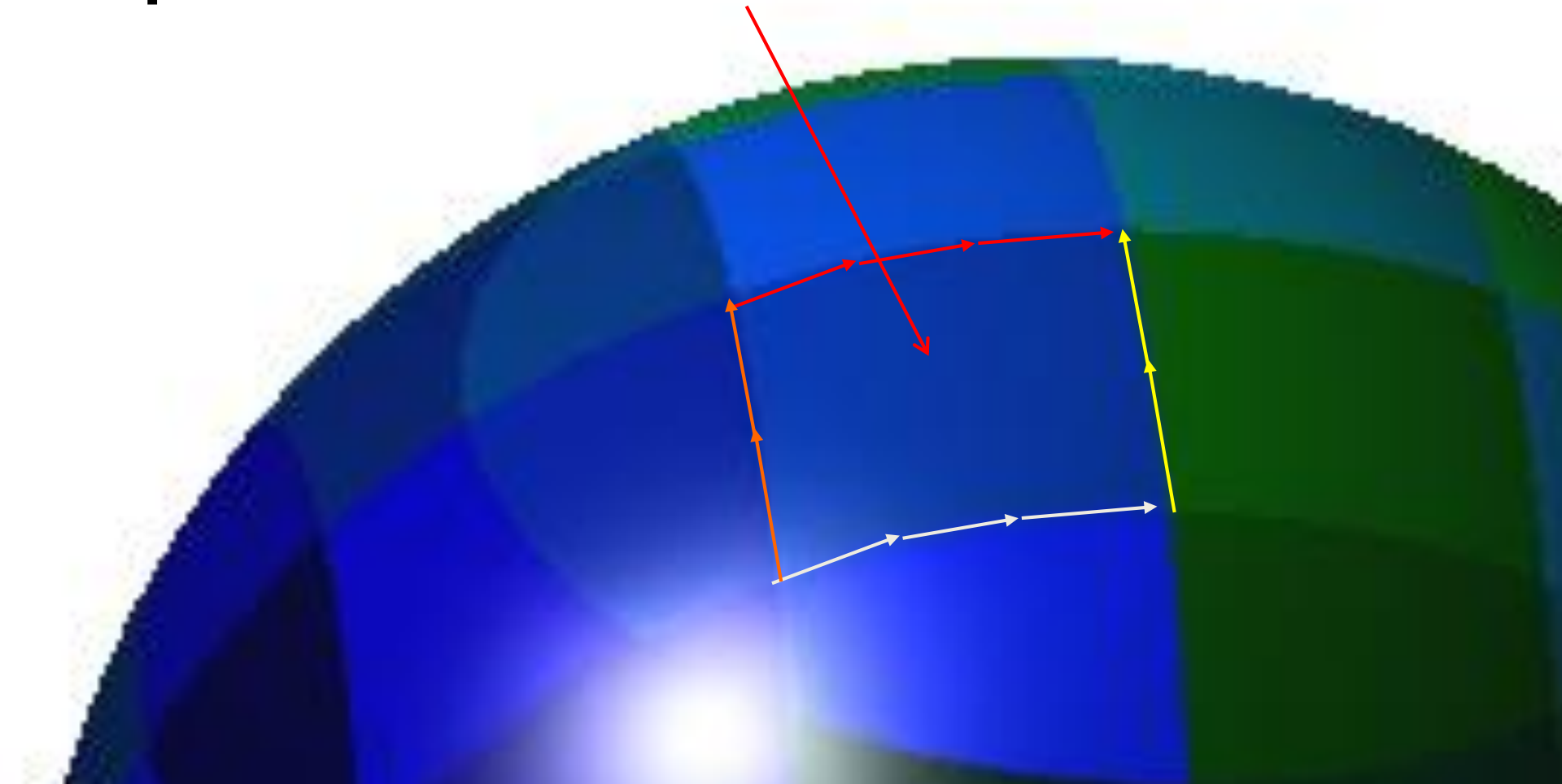
Issues:

- Create approximate **boundary curves** with points and tangents.



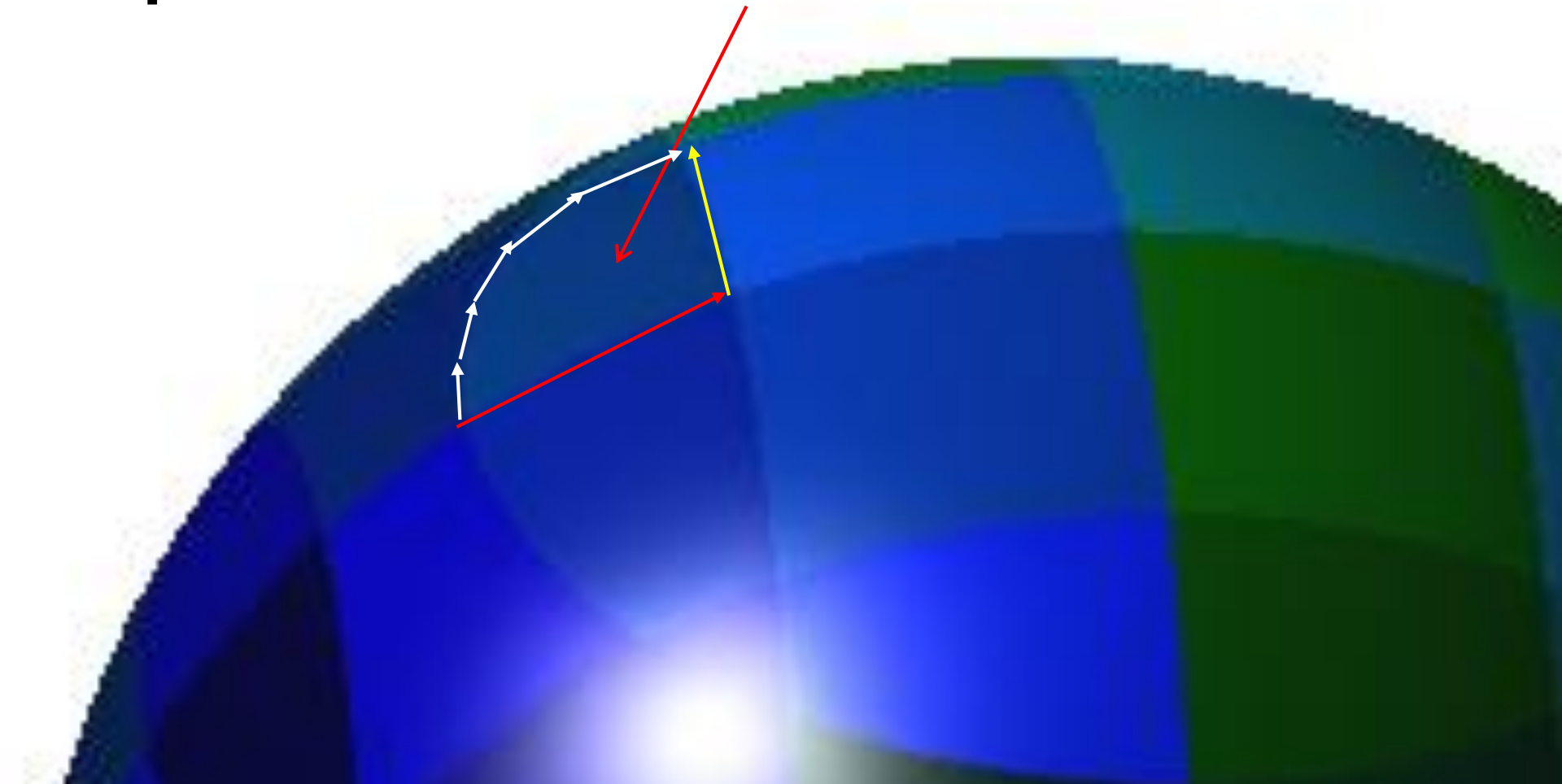
Examples on spheres

- Create approximate and fair **surface patches** for faces



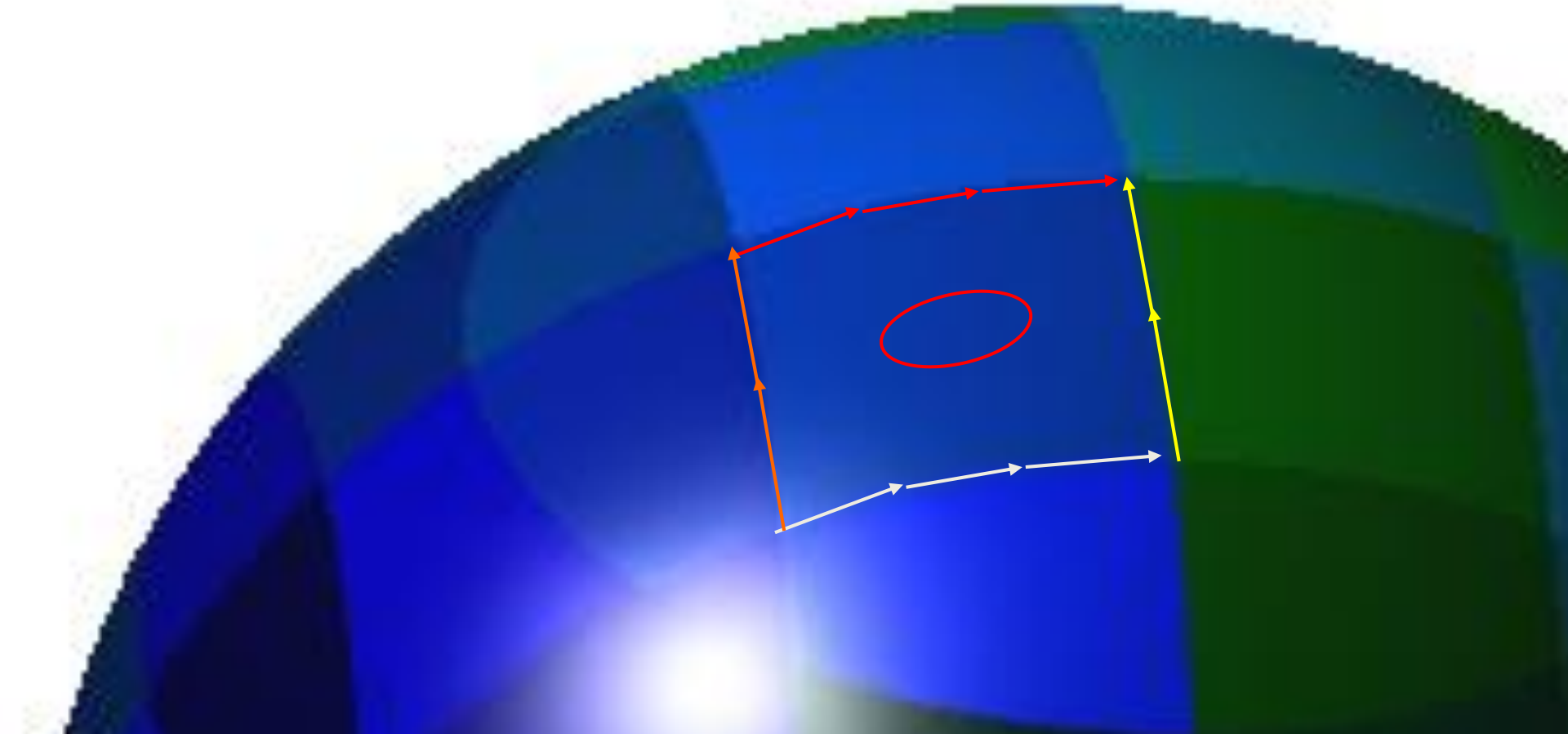
Examples on spheres

- Create approximate and fair **surface patches** for faces



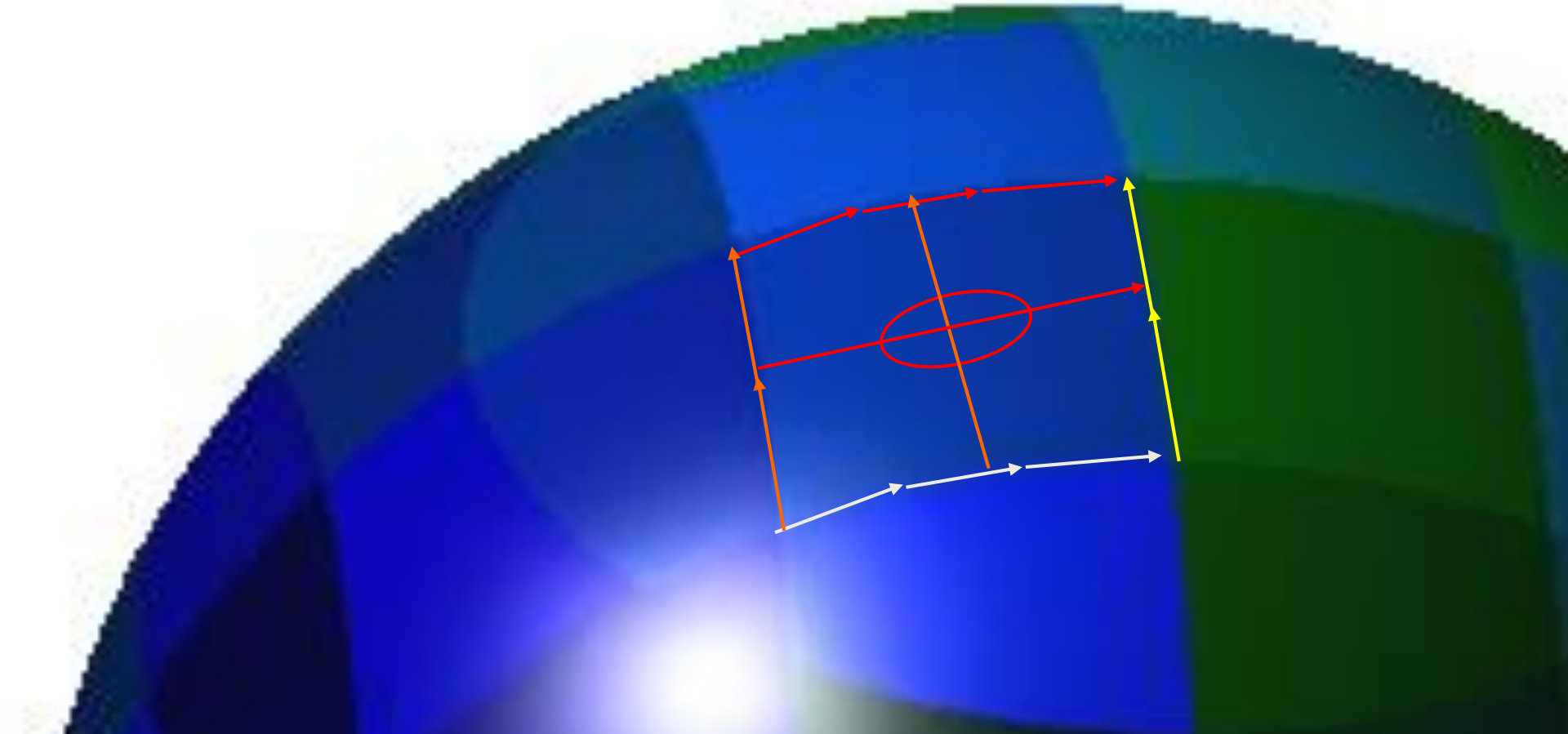
Examples on spheres

- Adaptively **refine**, if needed



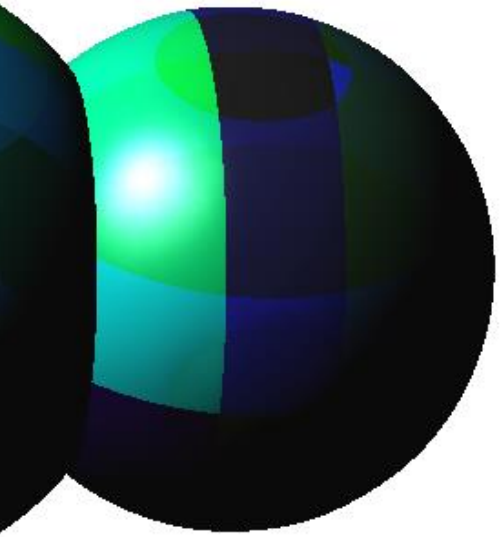
N-sided Surface Patches

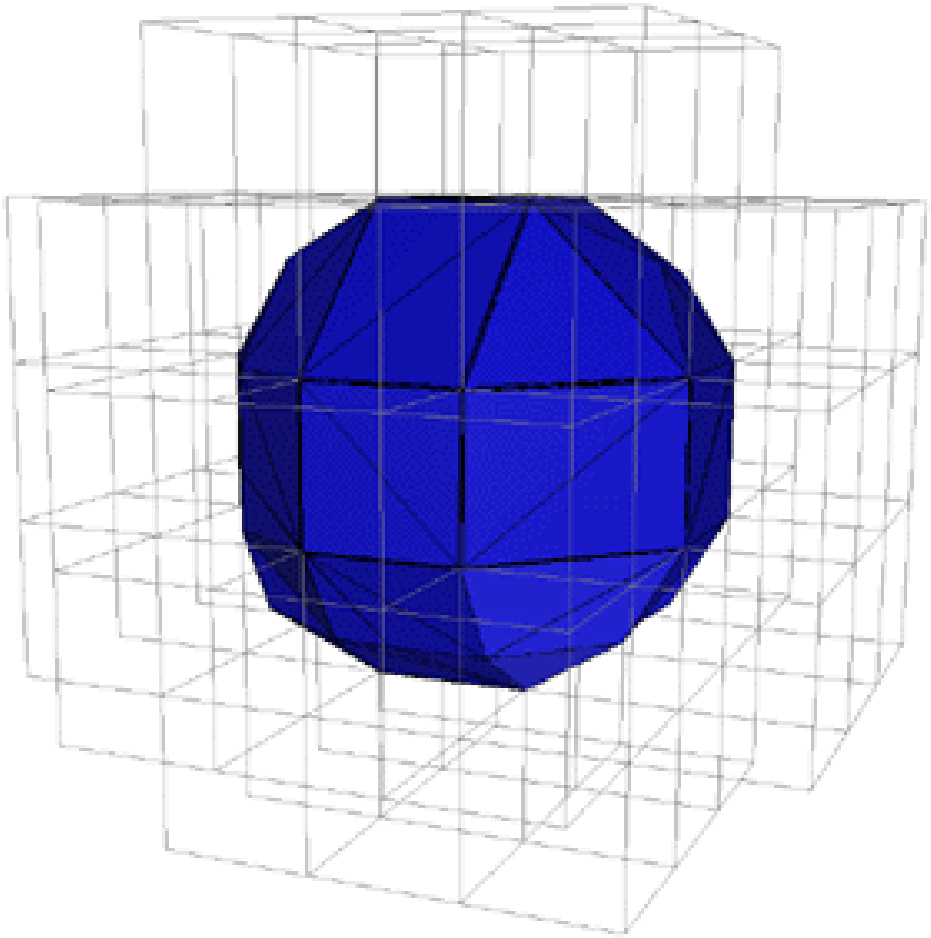
- Adaptively **refine**, if needed



Generalized Coons

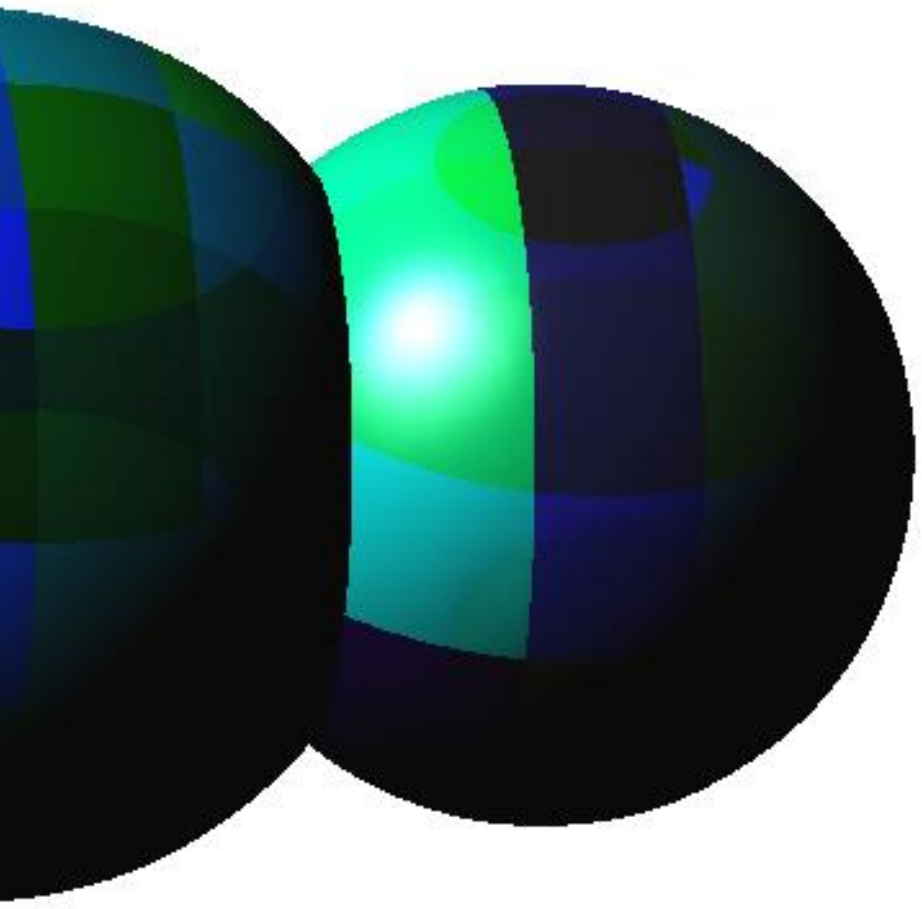
Patches vs marching cubes for spheres

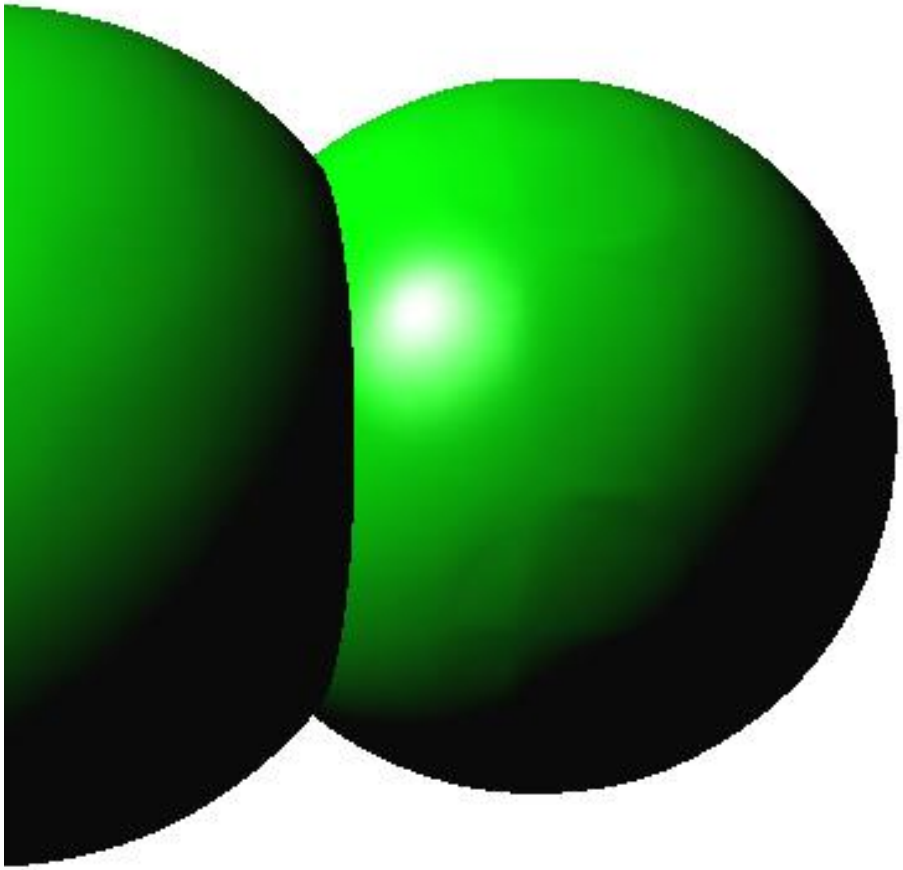




Generalized Coons

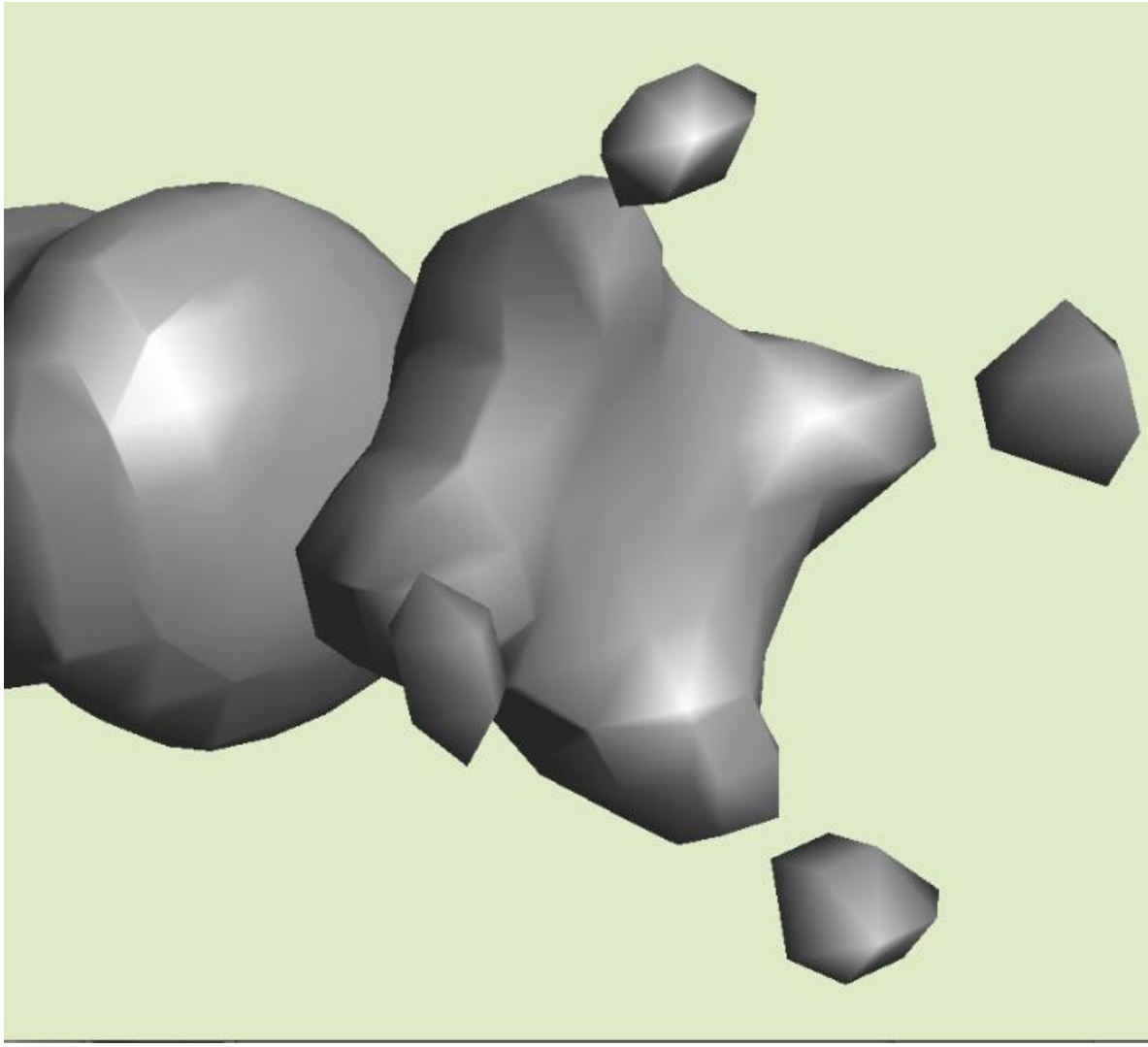
Comparing Spheres





Combustion experiment

Data from KAUST combustion lab – burn front of a fuel injection. Marching cubes 1500 triangles.

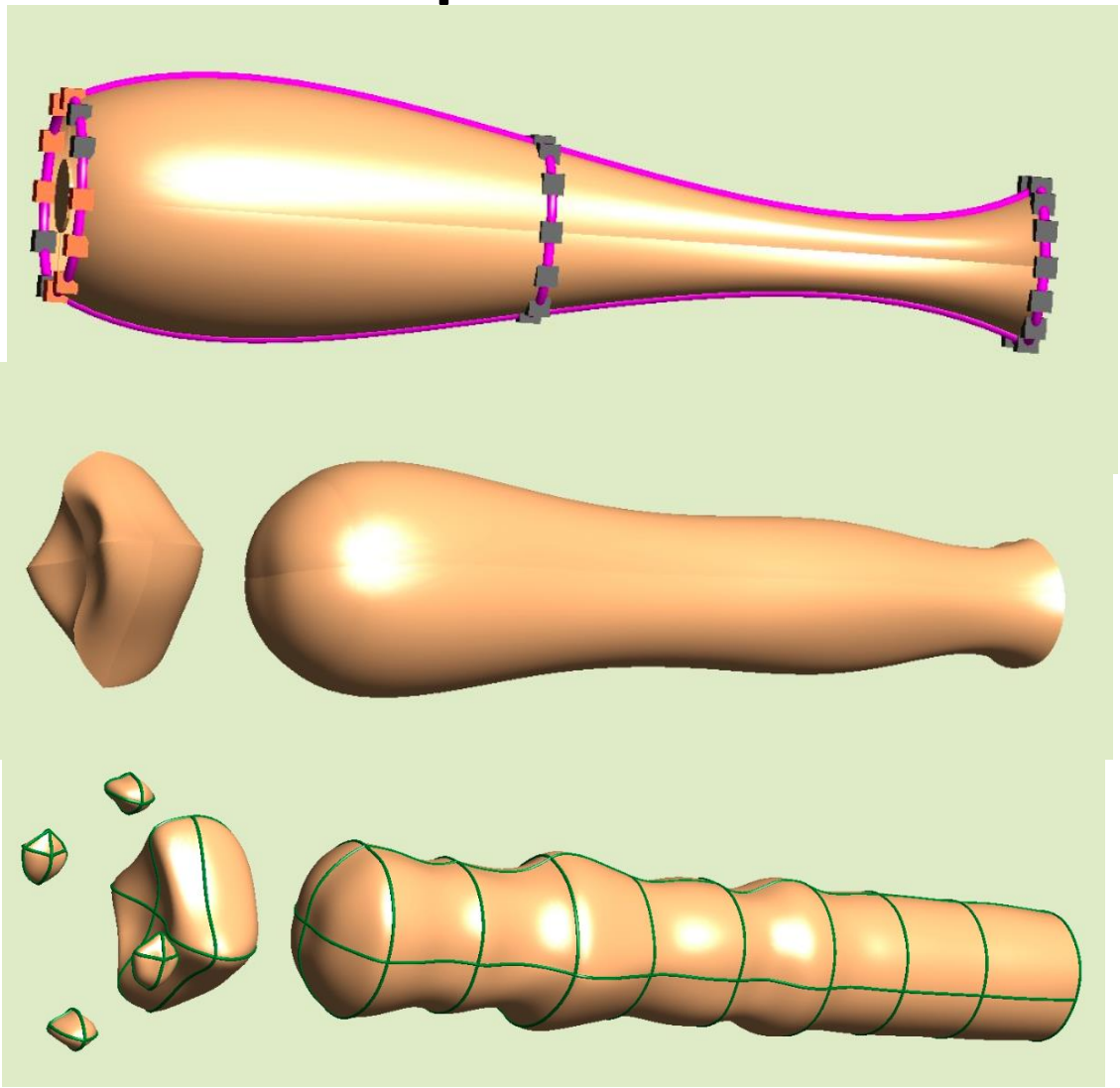


Combustion experiment

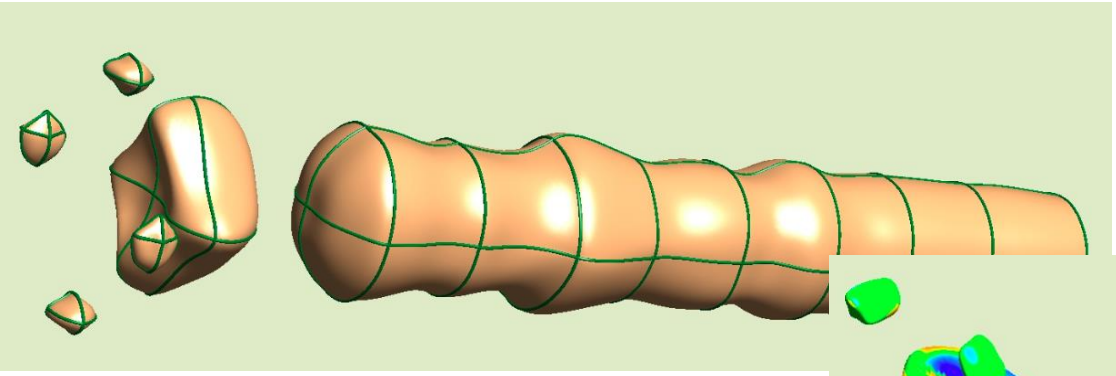
2^2 cells
8 surfaces

2^3 cells
28 surfaces

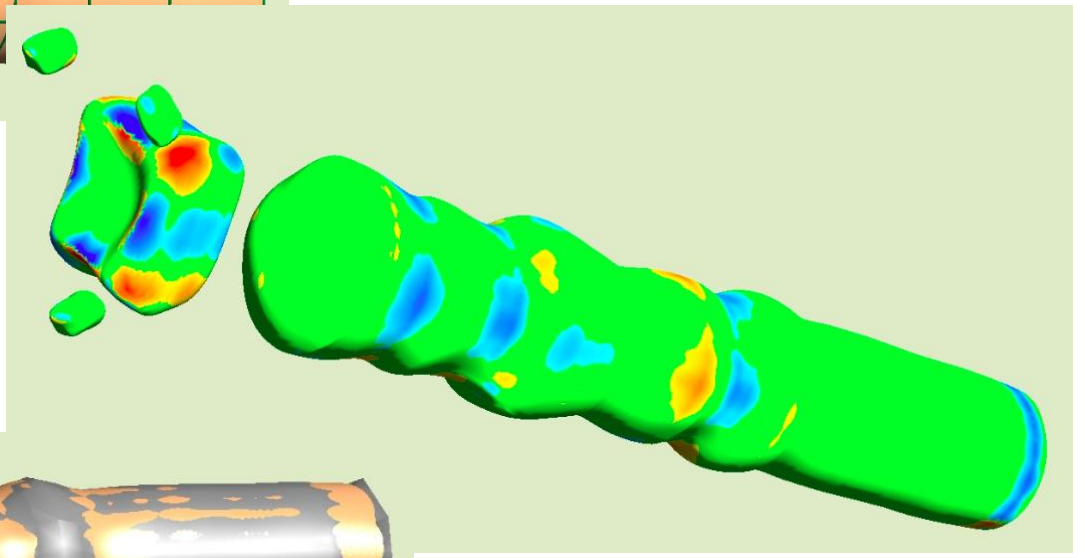
2^4 cells
72 surfaces

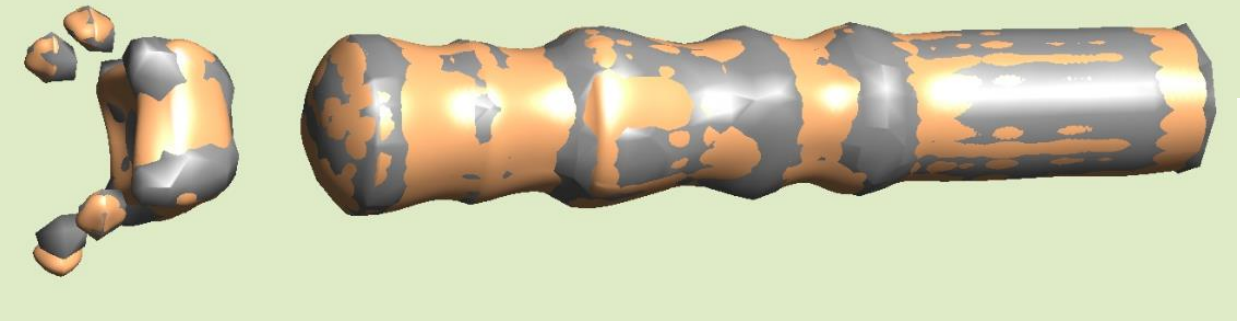


Comparison to Marching Cubes



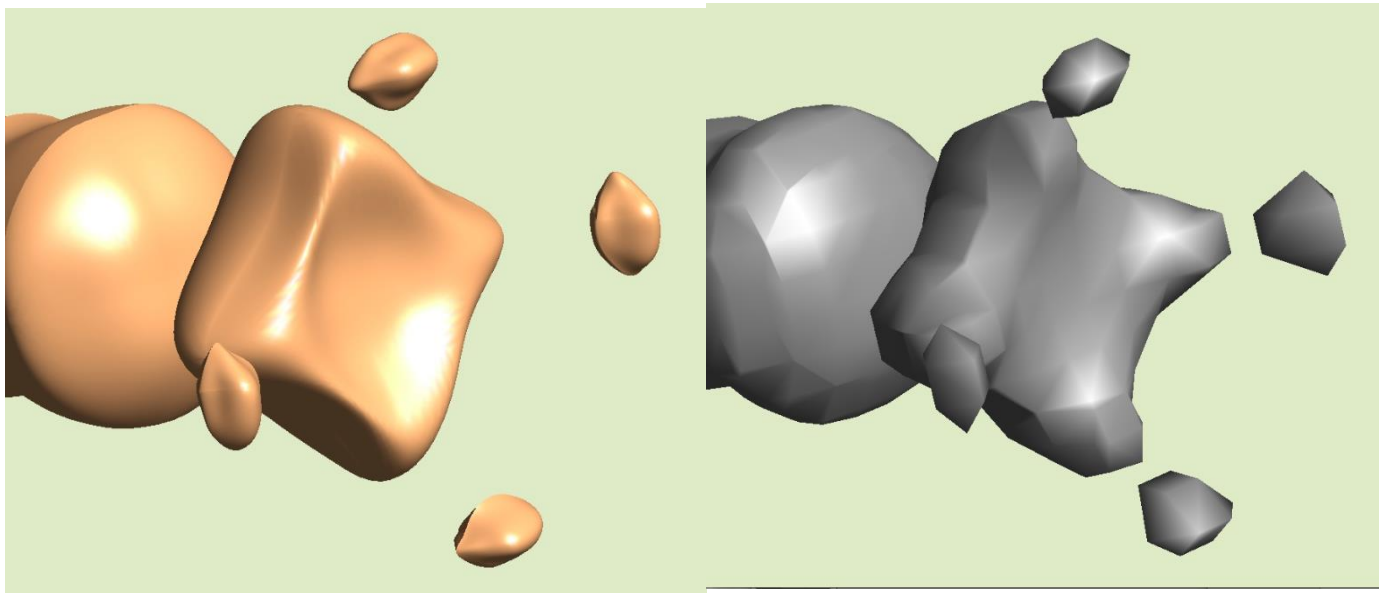
2^4 cells
72 surfaces

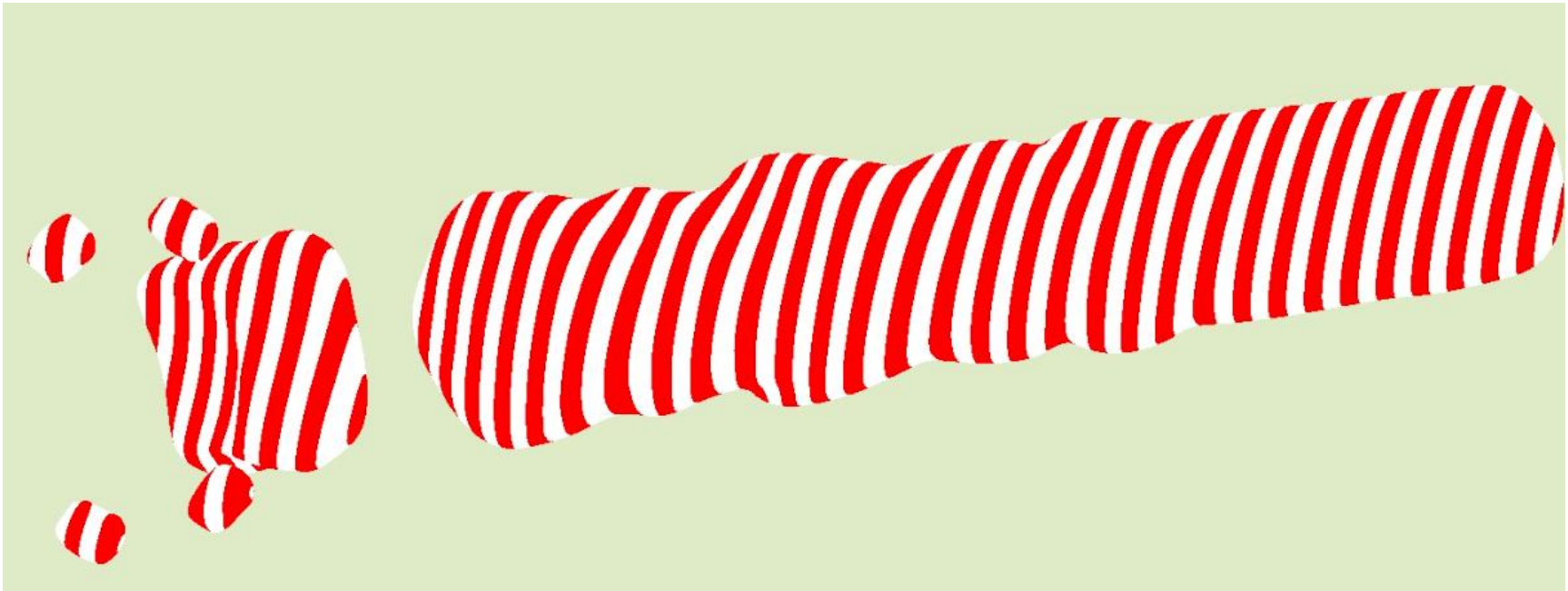




Smoothness

2^4 cells
72 surfaces



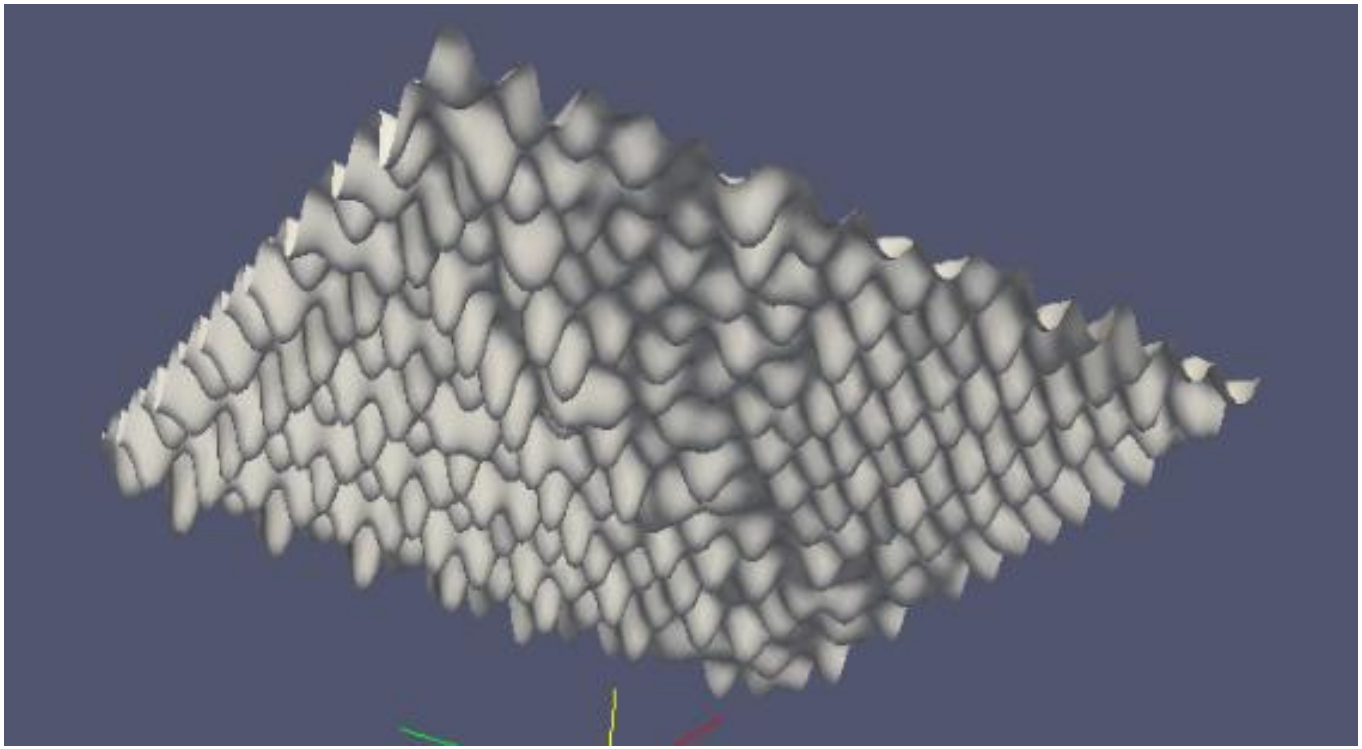


N-sided Surface Patches

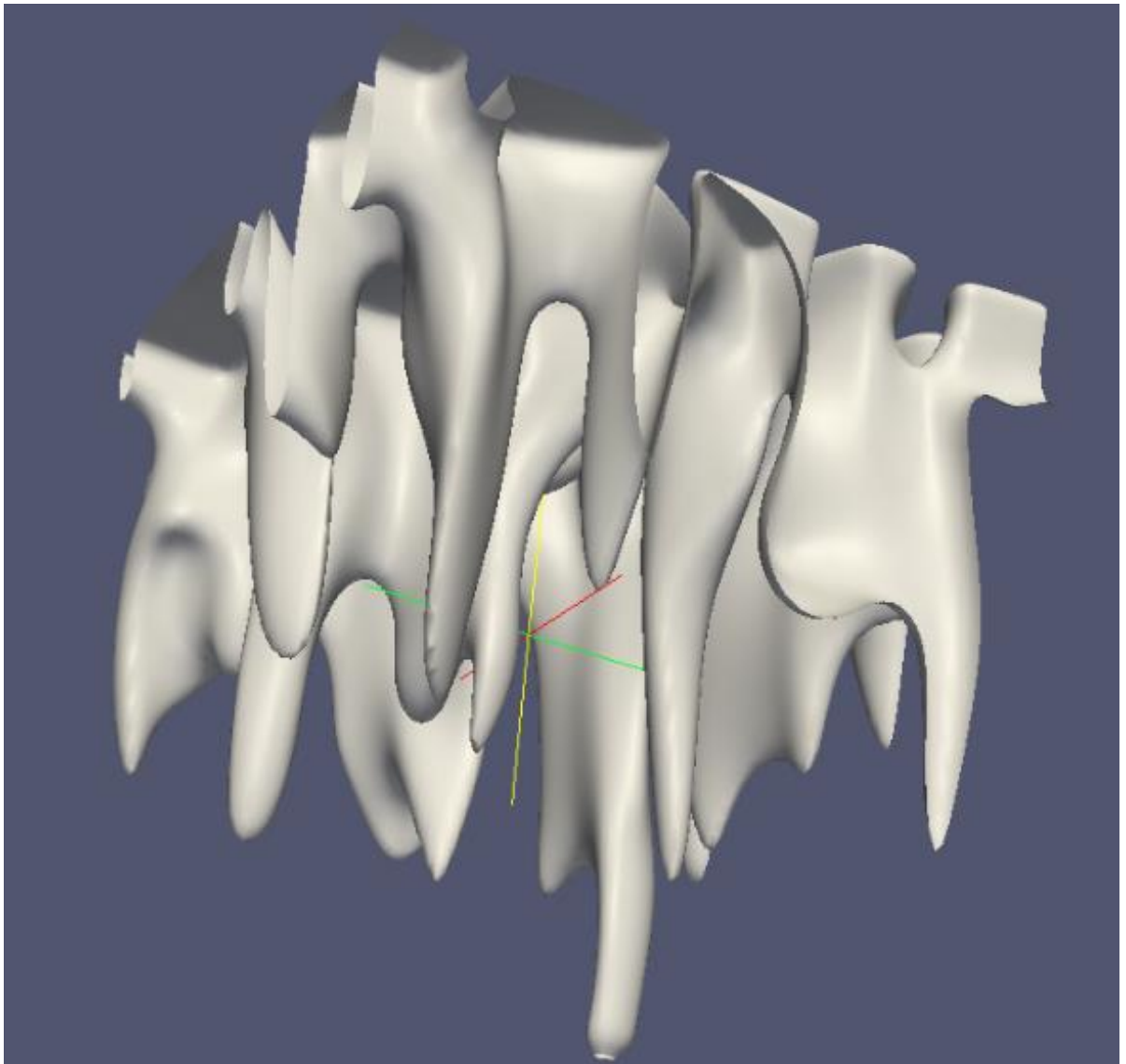
20 by 20 Isosurface of C02 sequestration

(Data from Shuyu Sun, KAUST)

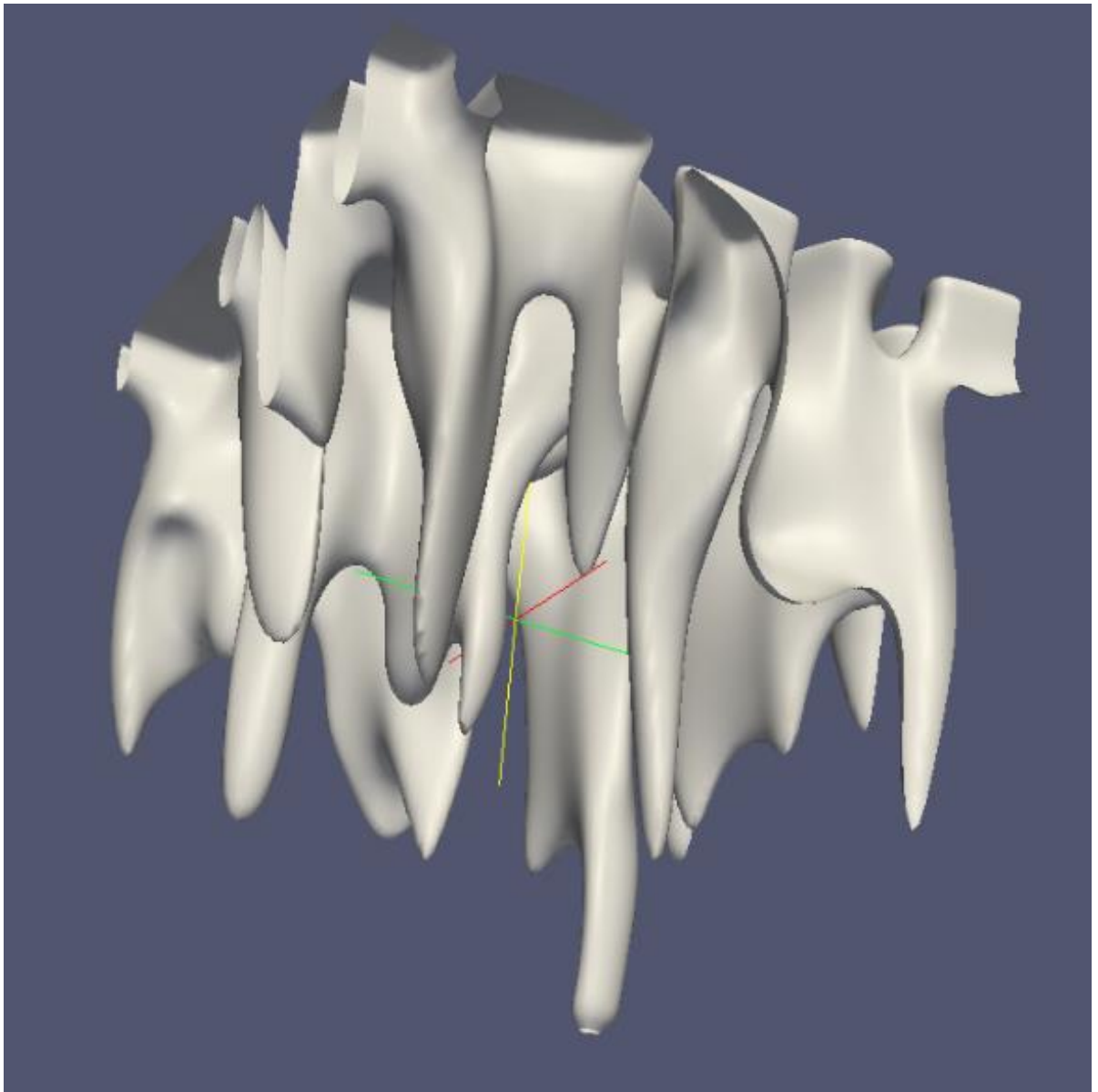
1000 patches



N-sided Surface Patches
1000 patches



Thank you !



Current Status and Call for Research:

- Setting parameters
- On mobile
- On GPU
- Testing and User Studies
- Using I-patches (demo)

Serendipity

$R^3 \rightarrow R$, Alyn

Serendipity

$R^3 \rightarrow R$, Alyn

$R^2 \rightarrow R^3 \rightarrow R$, Tamas

$R^1 \rightarrow R^3 \rightarrow R$, Other?

$R^0 \rightarrow R^3 \rightarrow R$, Why?

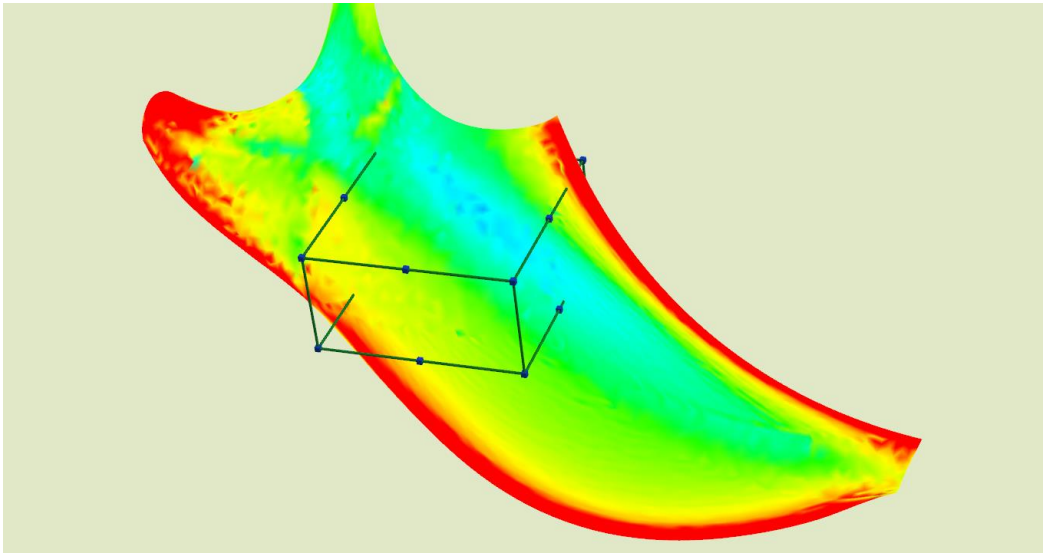
$R^3 \rightarrow R$:

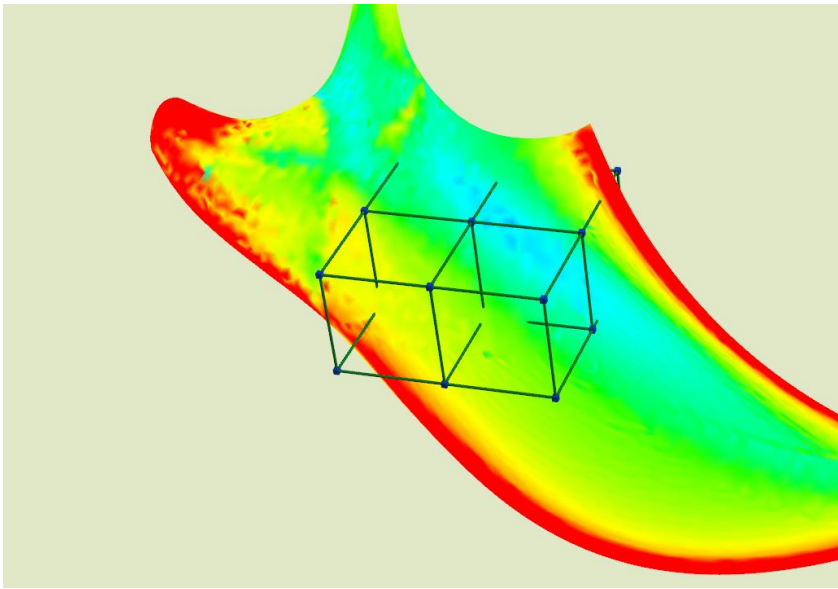
Smooth, Interpolate, Compress, Volume

$R^2 \rightarrow R^3 \rightarrow R$:

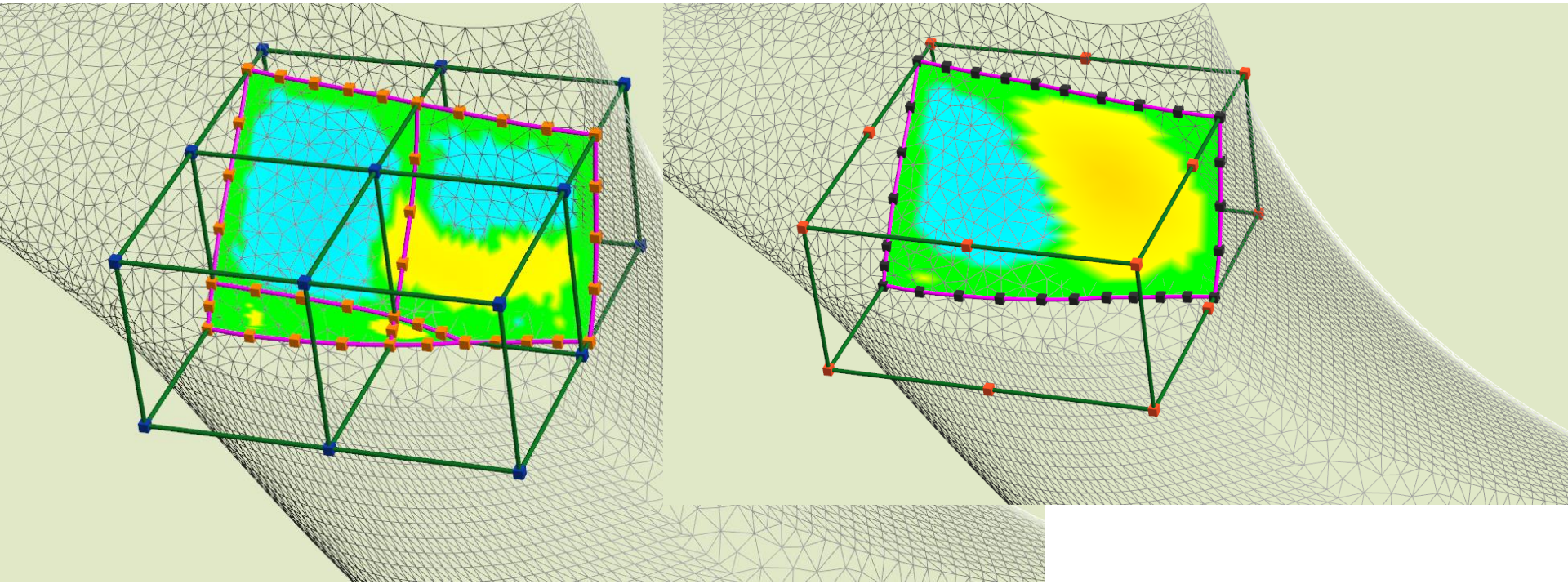
Smooth, Filter, Offset, Compress, Raytrace, Manifold

$R^2 \rightarrow R^3 \rightarrow R$: Curvature maps

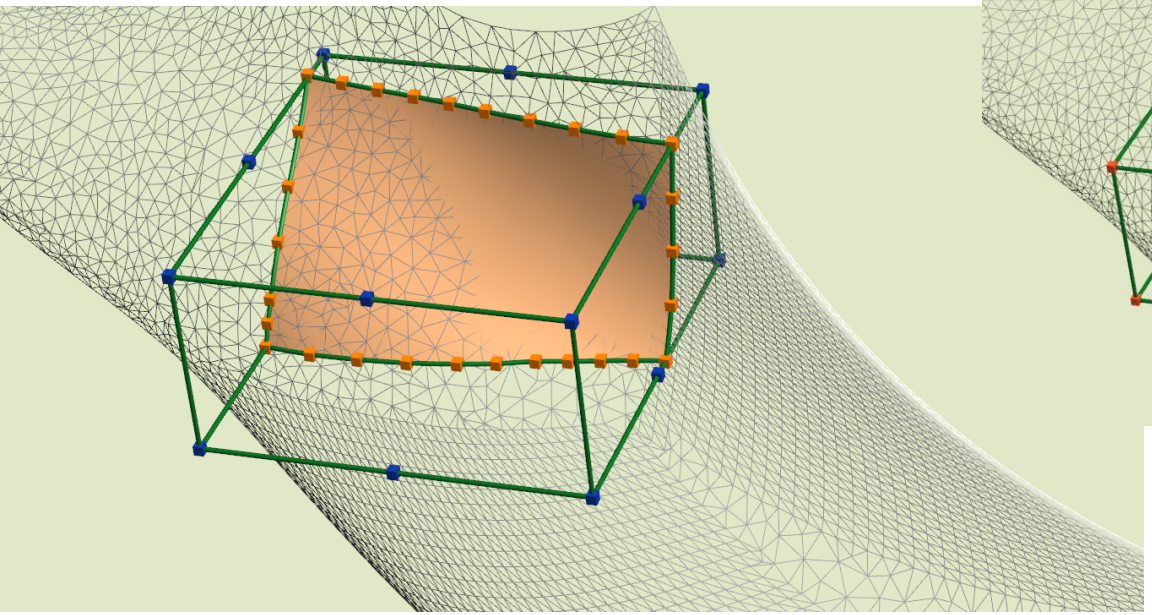


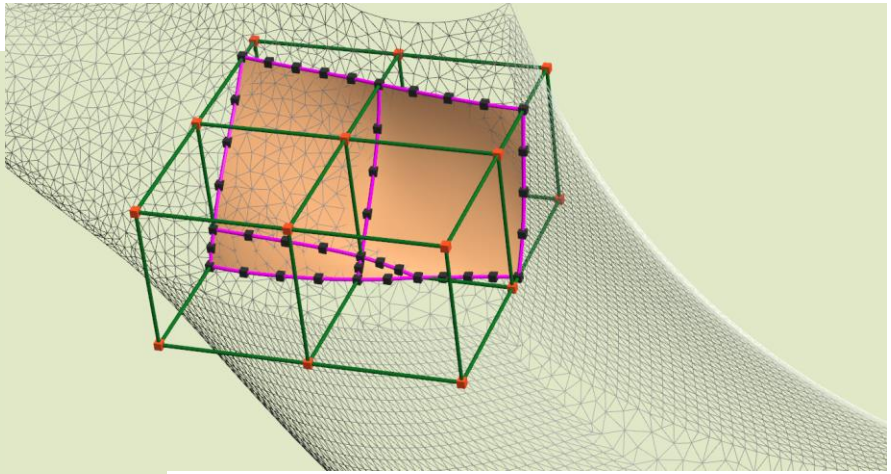


$R^2 \rightarrow R^3 \rightarrow R$: Distance maps



$R^2 \rightarrow R^3 \rightarrow R$: surfaces





Current Status and Call for Research:

- Using I-patches (demo)

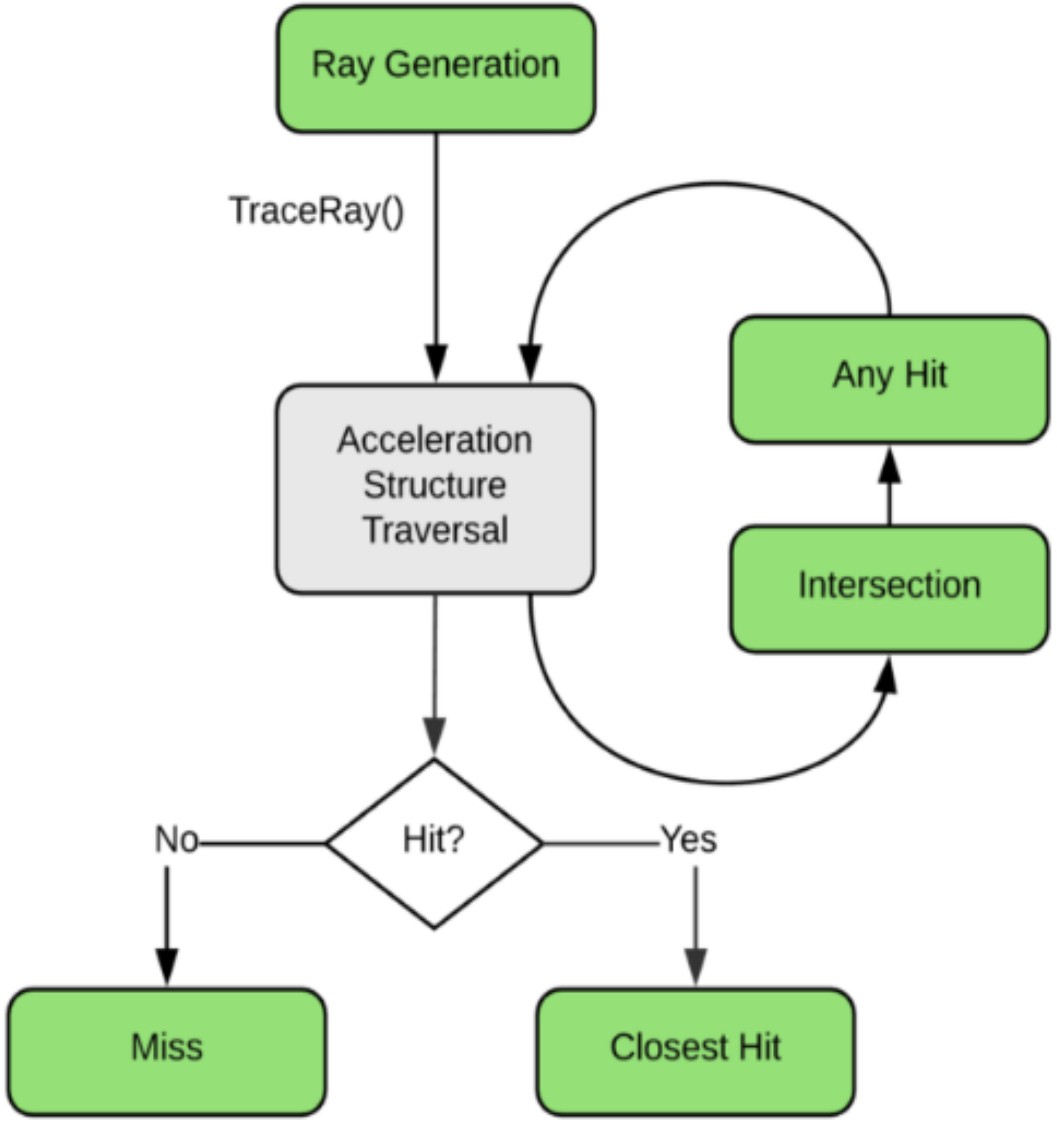
Raytracing!
https://devblogs.nvidia.com/introduction-nvidia-rtx-directx-ray-tracing/

RTX 20 raytracing hardware

Ray generation

Intersection and ***any hit*** are invoked whenever *TraceRay()* finds a *potential* a bounding volume, an individu geometric primitive like a sphere, a subdivision surface, or any primitive type you can code up!

closest hit or a ***miss*** is where most shading operations take place: material evaluation, texture lookups, and so on. Both closest hit and miss shaders can recursively trace rays by calling *TraceRay()* themselves.



RTX 20 raytracing hardware

Payload

Is how to communicate with other rays. The payload is a user-defined struct that's passed as an inout parameter to TraceRay(). Any hit, closest hit, and miss shaders can read and write the payload, and pass back the result of to the caller of TraceRay().