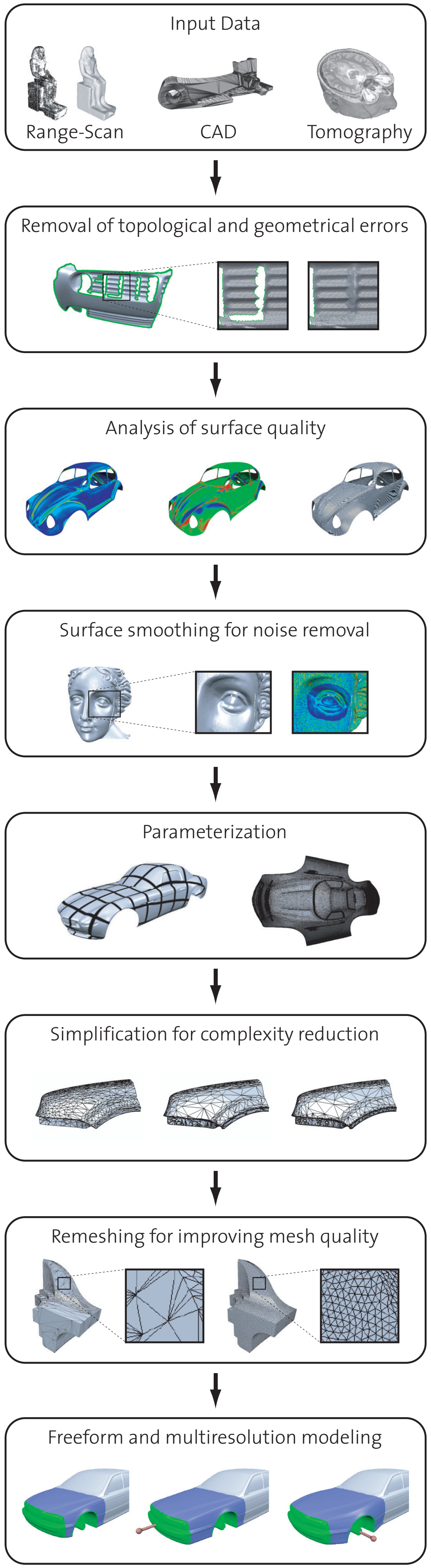


Surface Representations

Leif Kobbelt
RWTH Aachen University

Outline

- (mathematical) geometry representations
  – parametric vs. implicit
- approximation properties
- types of operations
  – distance queries
  – evaluation
  – modification / deformation
- data structures

Outline

- (mathematical) geometry representations
  - parametric vs. implicit
- approximation properties
- types of operations
  - distance queries
  - evaluation
  - modification / deformation
- data structures

Mathematical Representations

- parametric
  - range of a function
  - surface patch

$$\mathbf{f} : R^2 \rightarrow R^3, \quad \mathcal{S}_\Omega = \mathbf{f}(\Omega)$$

- implicit
  - kernel of a function
  - level set

$$F : R^3 \rightarrow R, \quad \mathcal{S}_c = \{\mathbf{p} : F(\mathbf{p}) = c\}$$

2D-Example: Circle

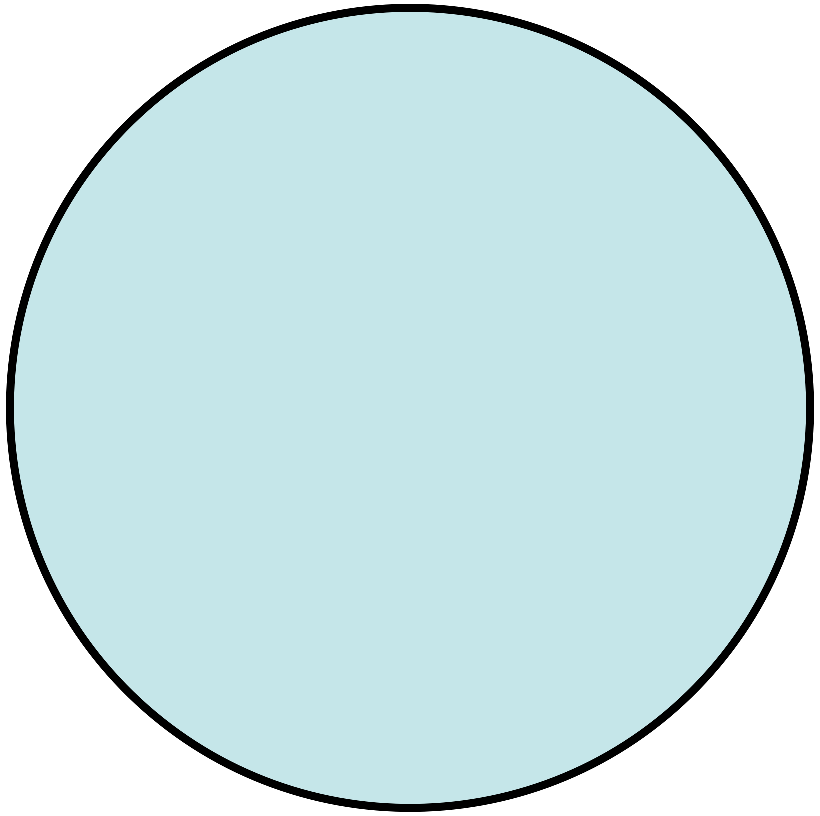
- parametric

$$\mathbf{f} : t \mapsto \begin{pmatrix} r\cos(t) \\ r\sin(t) \end{pmatrix}, \quad \mathcal{S} = \mathbf{f}([0, 2\pi])$$

- implicit

$$F(x,y) = x^2 + y^2 - r^2$$

$$\mathcal{S} = \{(x,y) : F(x,y) = 0\}$$



2D-Example: Island

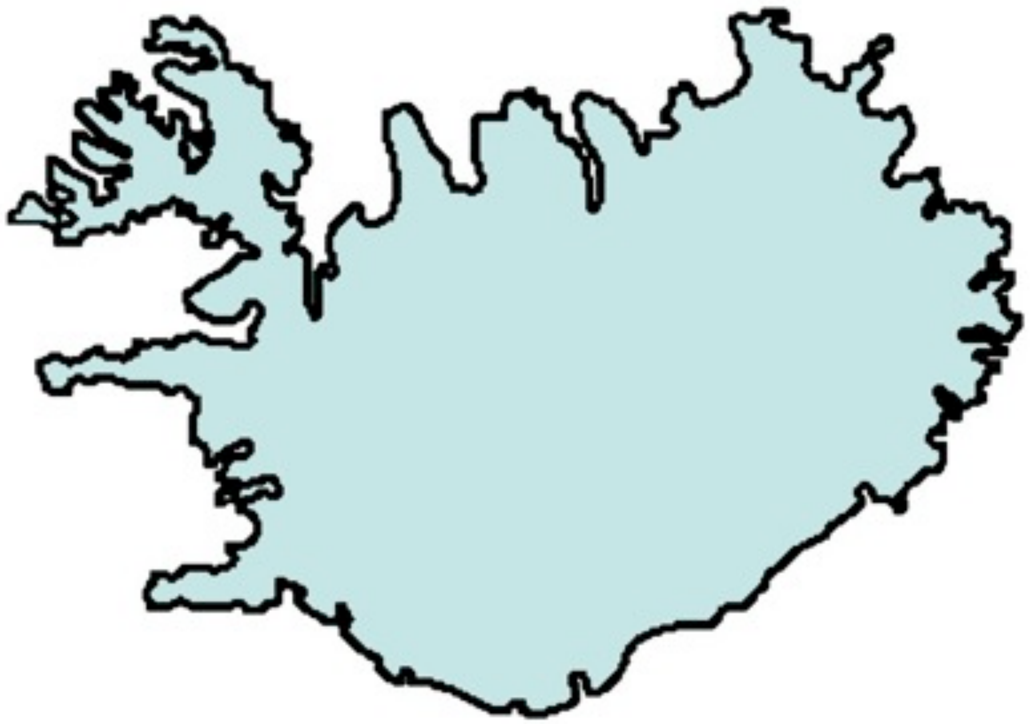
- parametric

$$\mathbf{f} : t \mapsto \begin{pmatrix} ??? \\ ??? \end{pmatrix}, \quad \mathcal{S} = \mathbf{f}([0, 2\pi])$$

- implicit

$$F(x, y) = \quad ???$$

$$\mathcal{S} = \{(x, y) : F(x, y) = 0\}$$



Approximation Quality

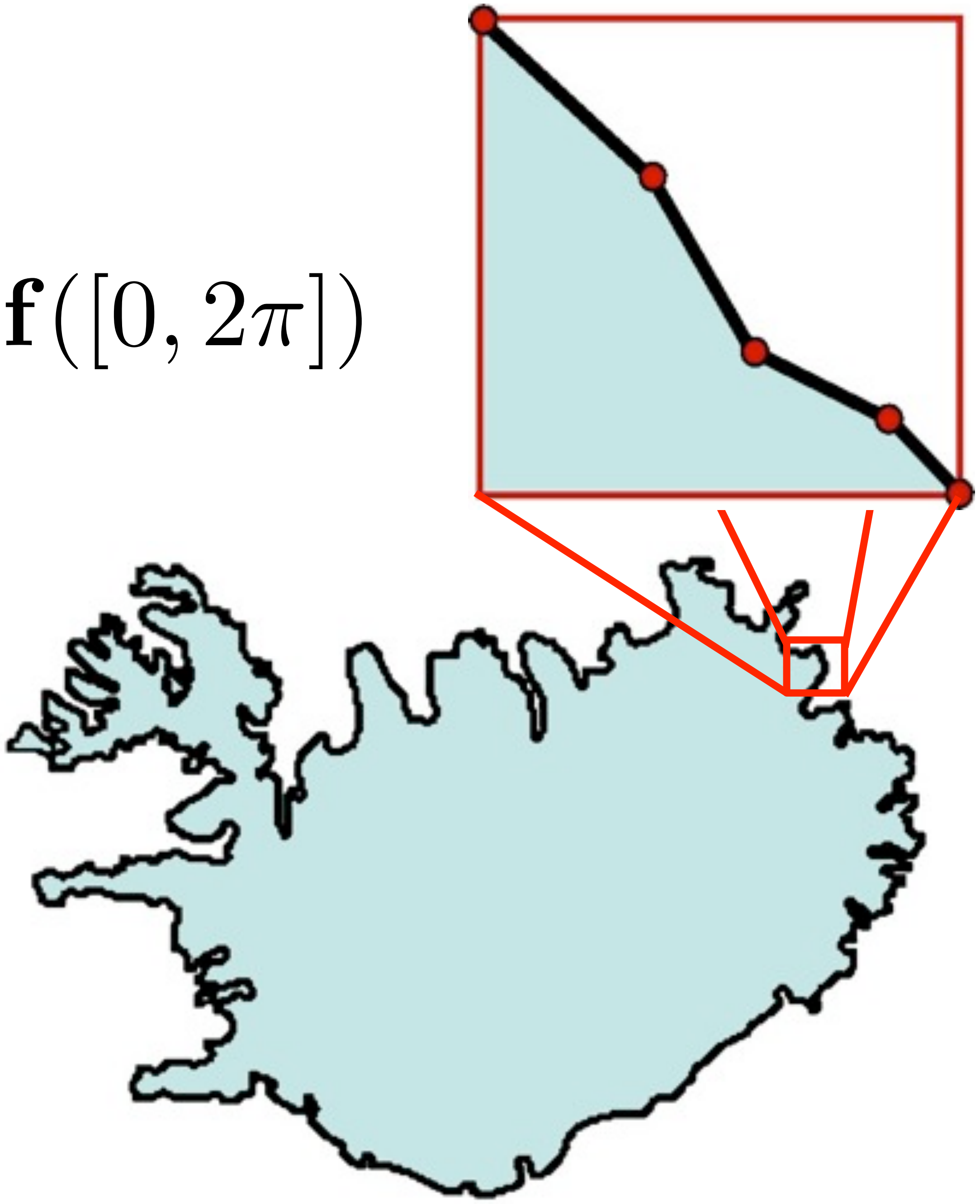
- ***piecewise*** parametric

$$\mathbf{f} : t \mapsto \begin{pmatrix} ??? \\ ??? \end{pmatrix}, \quad \mathcal{S} = \mathbf{f}([0, 2\pi])$$

- piecewise implicit

$$F(x, y) = \quad ???$$

$$\mathcal{S} = \{(x, y) : F(x, y) = 0\}$$



Approximation Quality

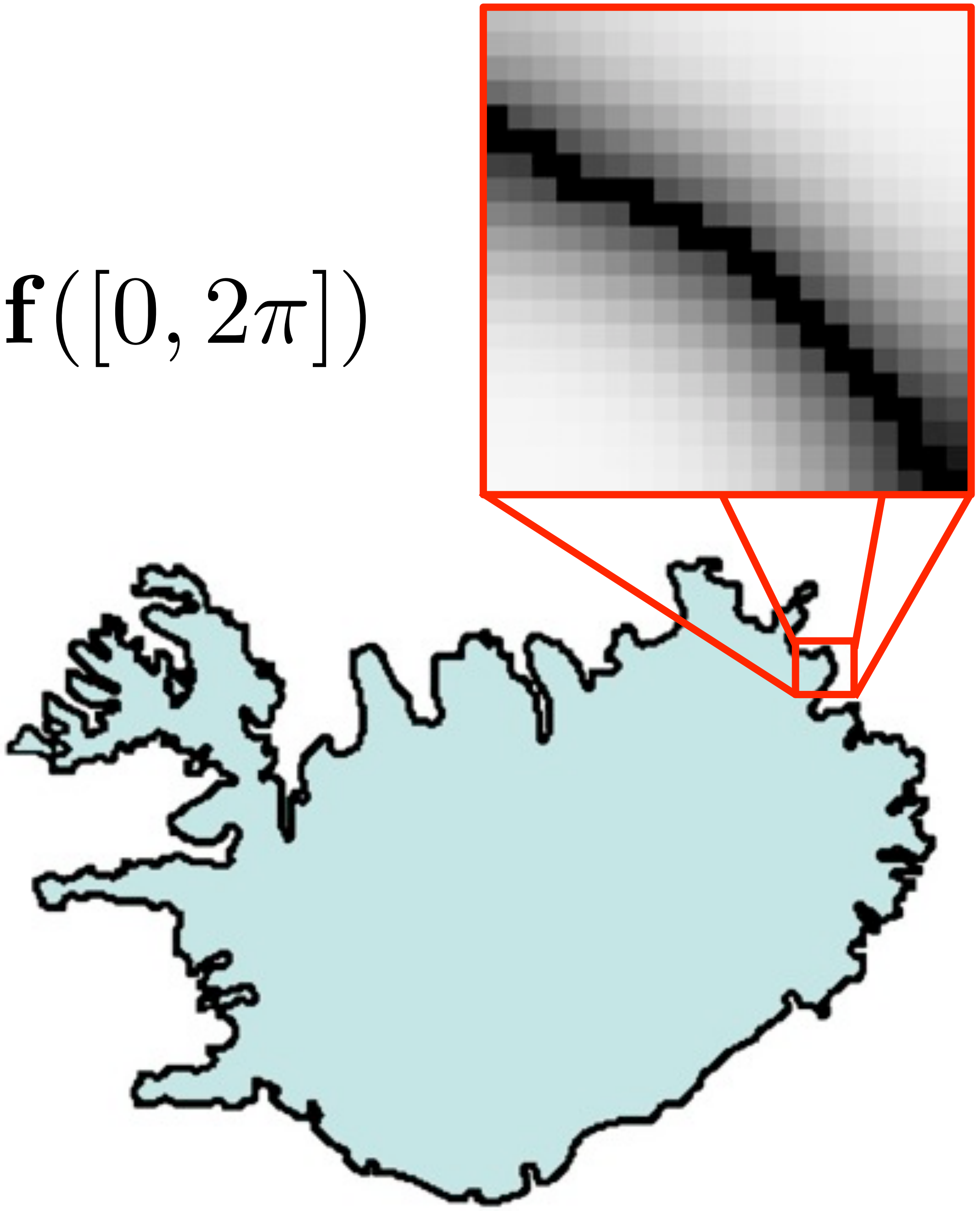
- piecewise parametric

$$\mathbf{f} : t \mapsto \begin{pmatrix} ??? \\ ??? \end{pmatrix}, \quad \mathcal{S} = \mathbf{f}([0, 2\pi])$$

- ***piecewise*** implicit

$$F(x, y) = \quad ???$$

$$\mathcal{S} = \{(x, y) : F(x, y) = 0\}$$

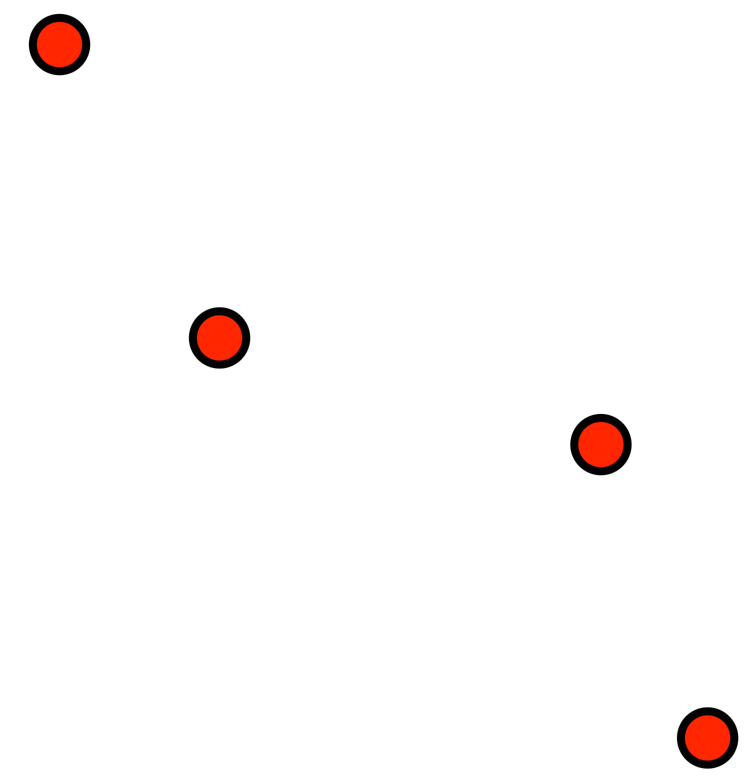


Requirements / Properties

- continuity
  - interpolation / approximation $\mathbf{f}(u_i, v_i) \approx \mathbf{p}_i$
- topological consistency
  - manifold-ness
- smoothness
  - C^0, C^1, C^2, ... C^k
- fairness
  - curvature distribution

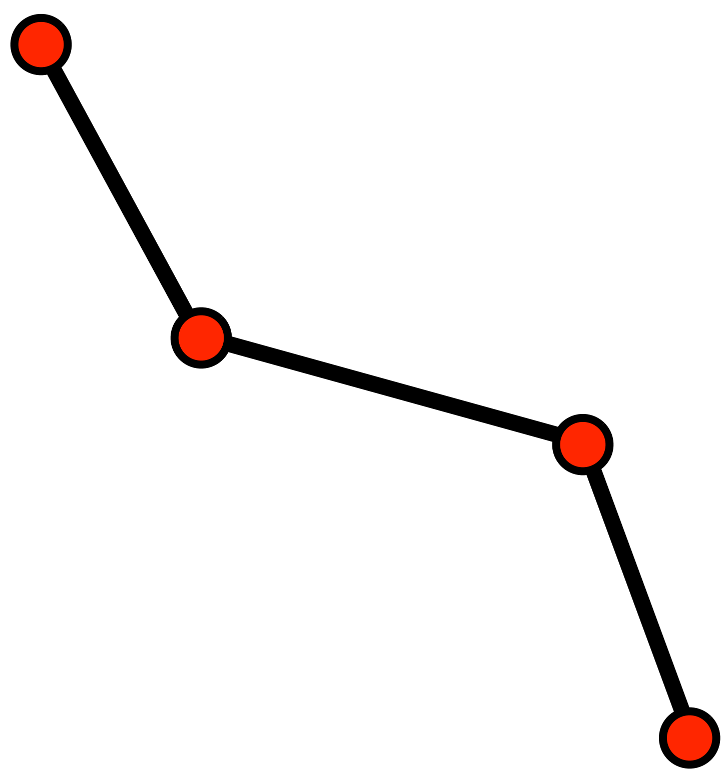
Requirements / Properties

- continuity
  - interpolation / approximation $\mathbf{f}(u_i, v_i) \approx \mathbf{p}_i$
- topological consistency
  - manifold-ness
- smoothness
  - C^0, C^1, C^2, ... C^k
- fairness
  - curvature distribution



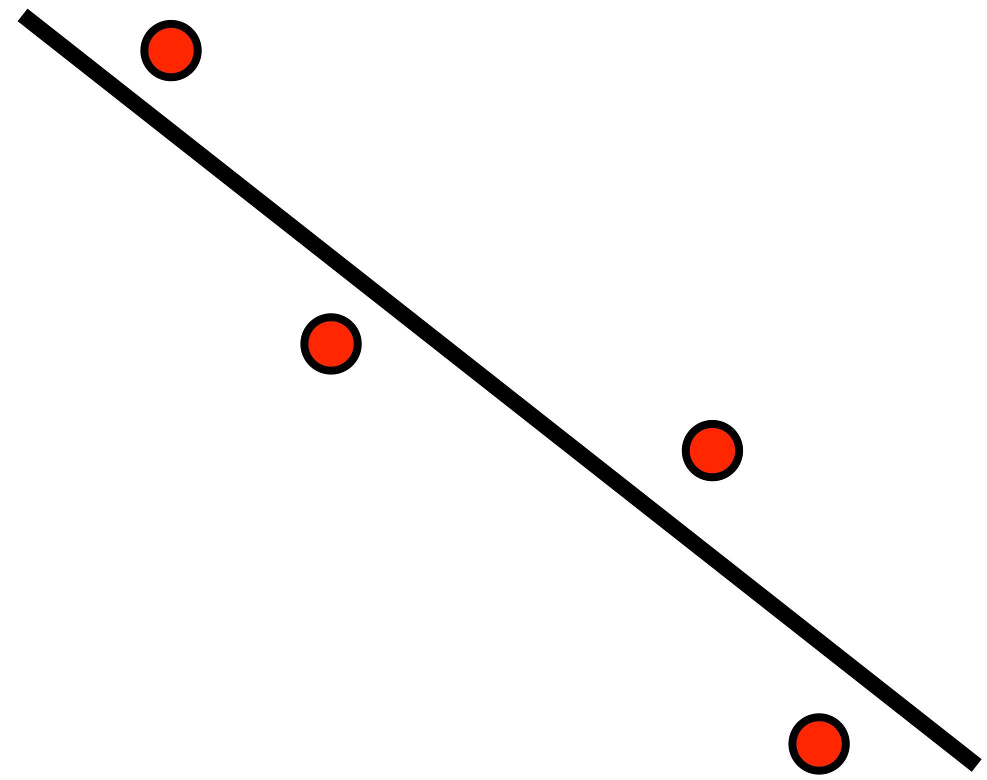
Requirements / Properties

- continuity
  - interpolation / approximation $\mathbf{f}(u_i, v_i) \approx \mathbf{p}_i$
- topological consistency
  - manifold-ness
- smoothness
  - C^0, C^1, C^2, ... C^k
- fairness
  - curvature distribution



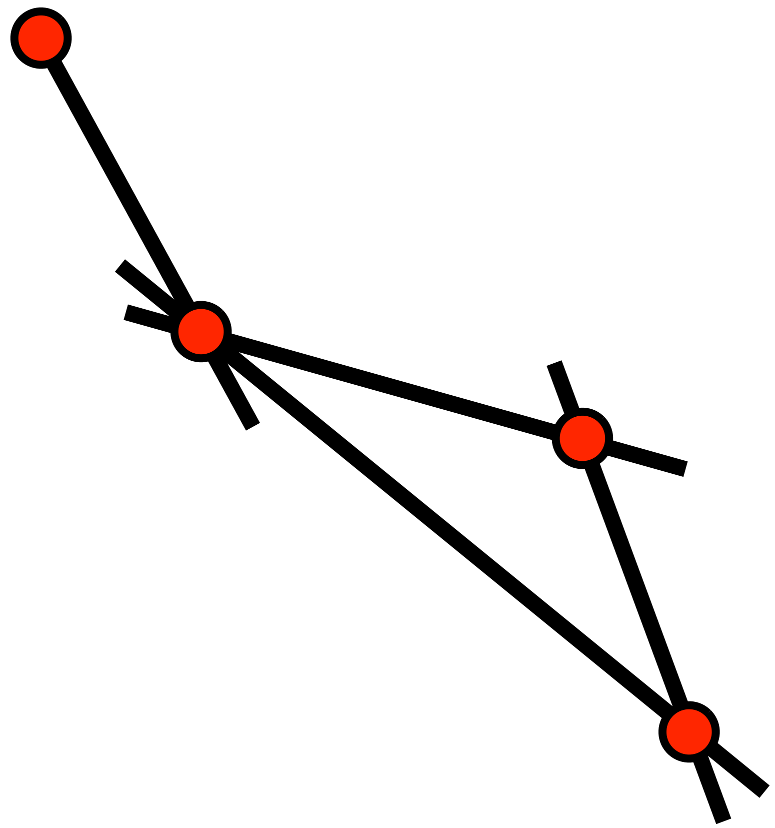
Requirements / Properties

- continuity
  - interpolation / approximation $\mathbf{f}(u_i, v_i) \approx \mathbf{p}_i$
- topological consistency
  - manifold-ness
- smoothness
  - C^0, C^1, C^2, ... C^k
- fairness
  - curvature distribution

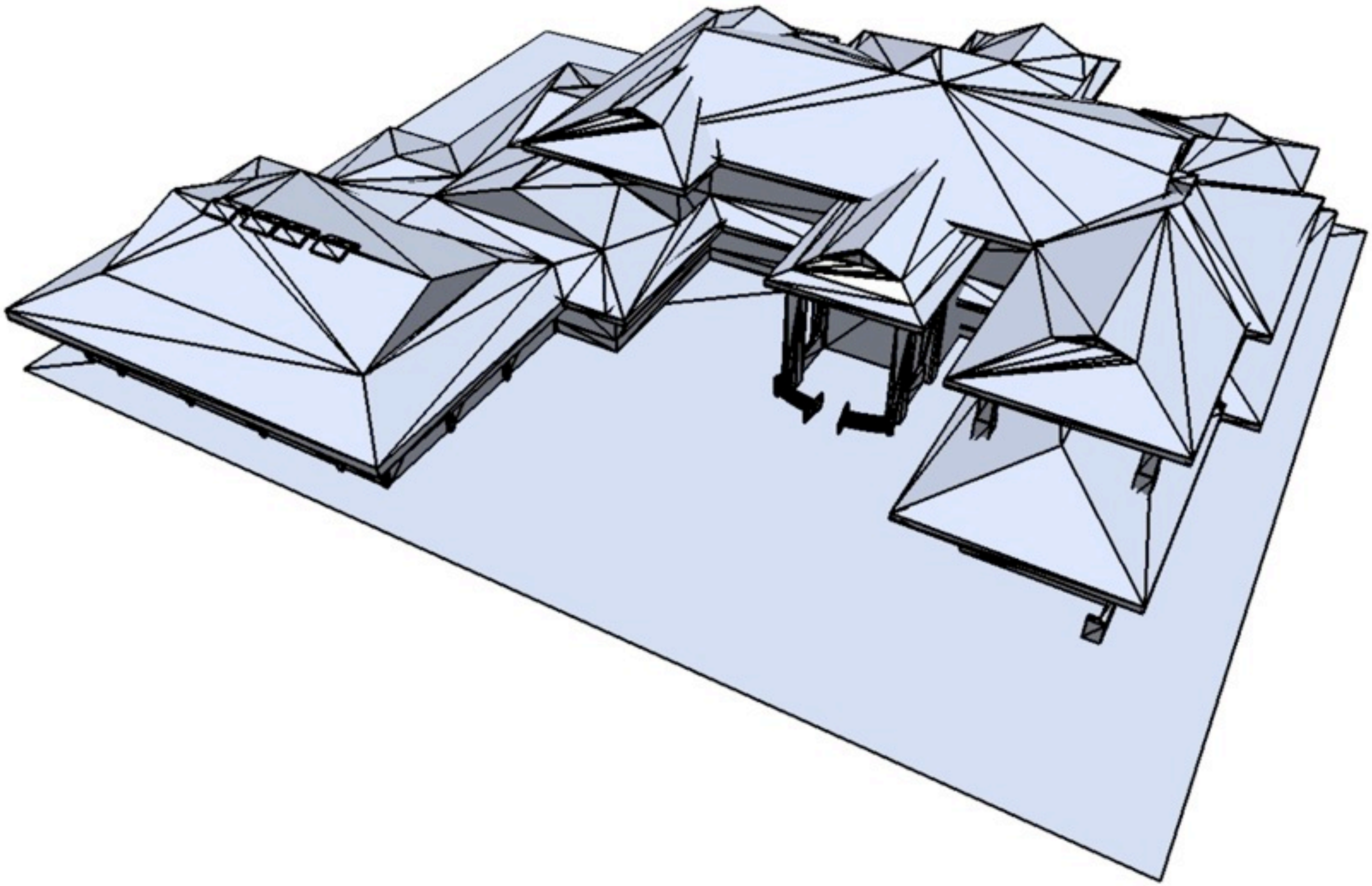


Requirements / Properties

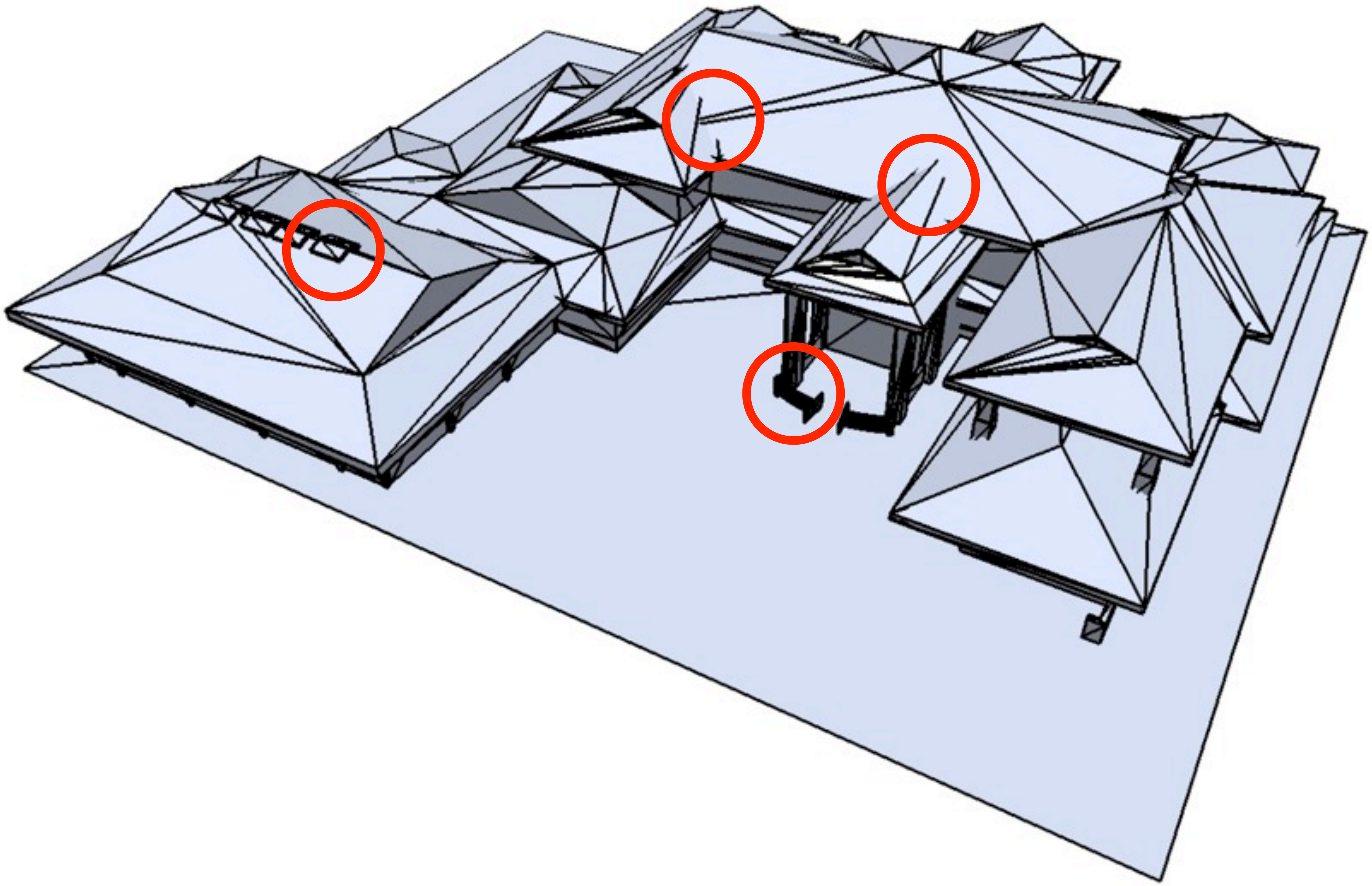
- continuity
  - interpolation / approximation $\mathbf{f}(u_i, v_i) \approx \mathbf{p}_i$
- topological consistency
  - manifold-ness
- smoothness
  - C^0, C^1, C^2, ... C^k
- fairness
  - curvature distribution



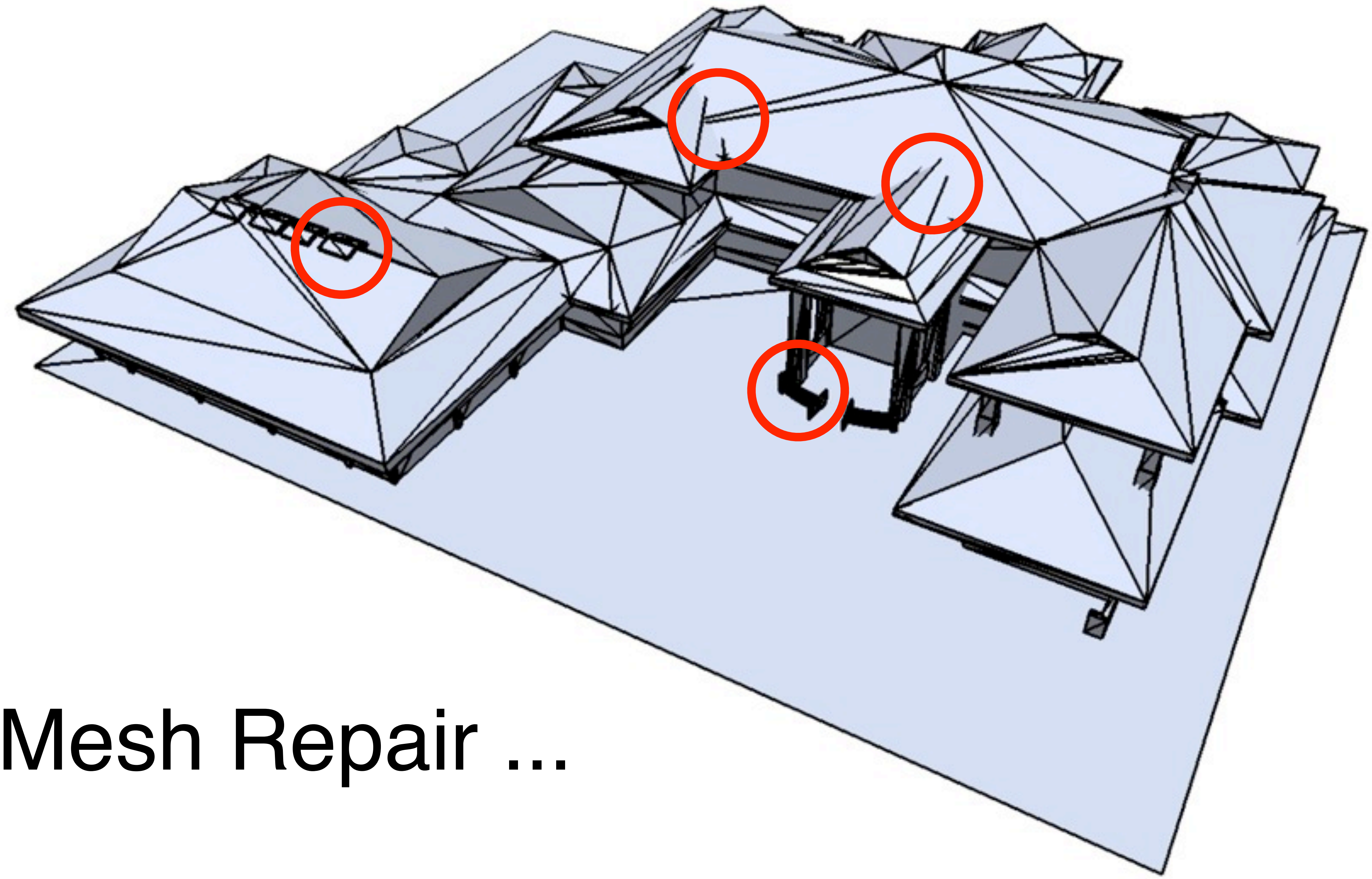
Topological Consistency



Topological Consistency



Topological Consistency



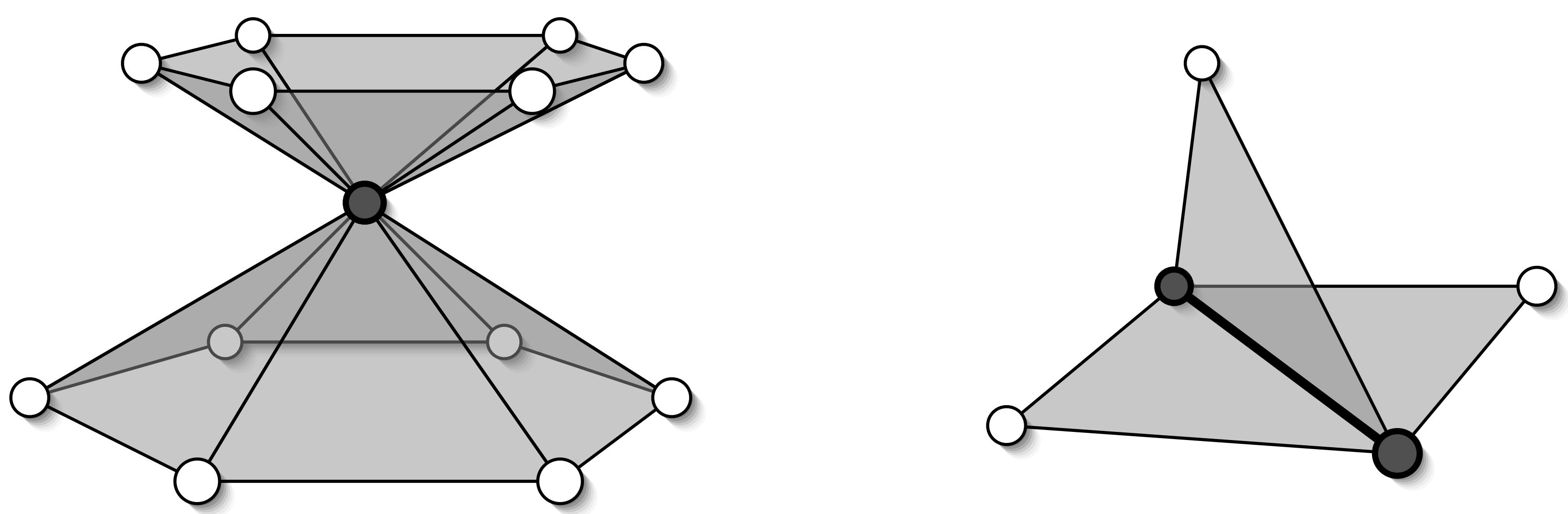
Mesh Repair ...

Closed 2-Manifolds

- parametric
  - disk-shaped neighborhoods
  - $\mathbf{f}(D_\varepsilon[u,v]) = D_\delta[\mathbf{f}(u,v)]$ + injectivity
- implicit
  - surface of a "physical" solid
  - $F(x,y,z) = c, \quad \|\nabla F(x,y,z)\| \neq 0$

Closed 2-Manifolds

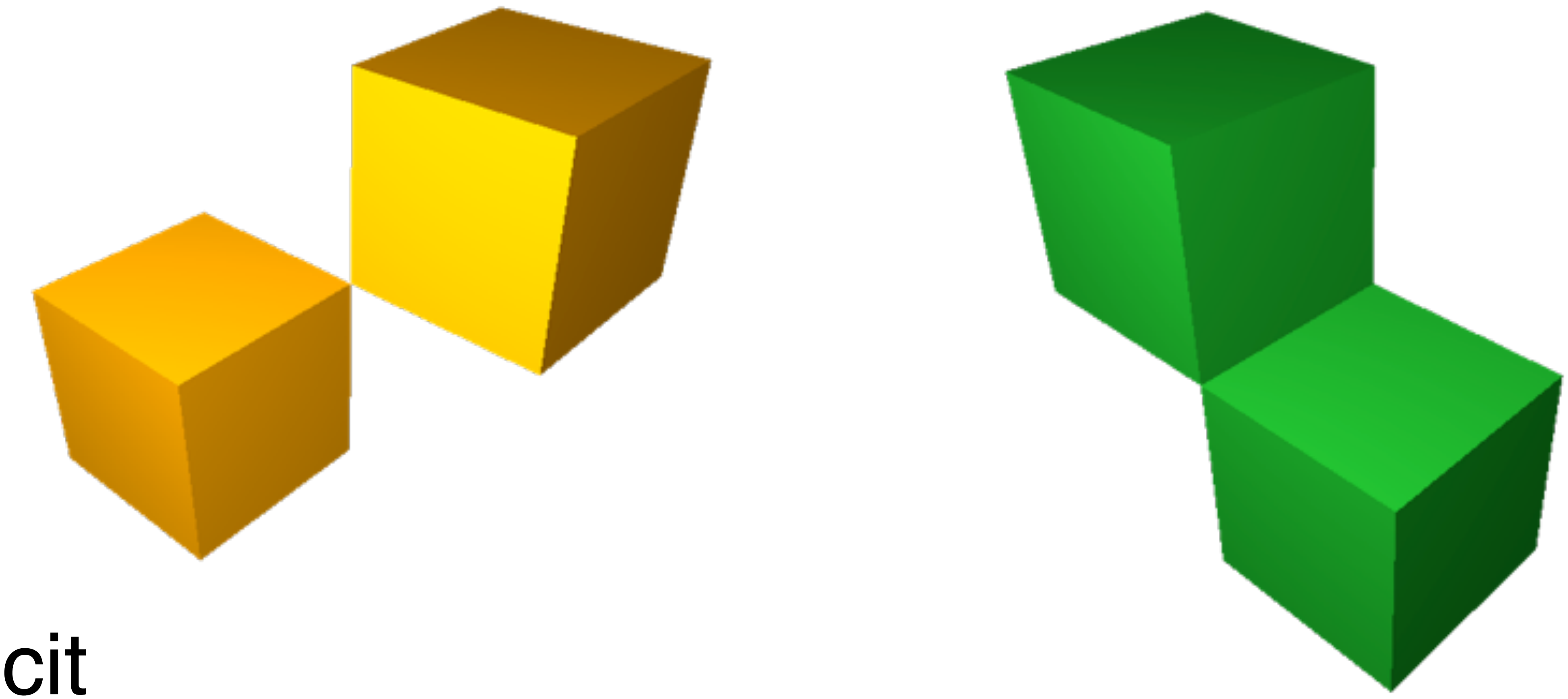
- parametric
  - disk-shaped neighborhoods
  - $\mathbf{f}(D_\varepsilon[u,v]) = D_\delta[\mathbf{f}(u,v)]$ + injectivity



Closed 2-Manifolds

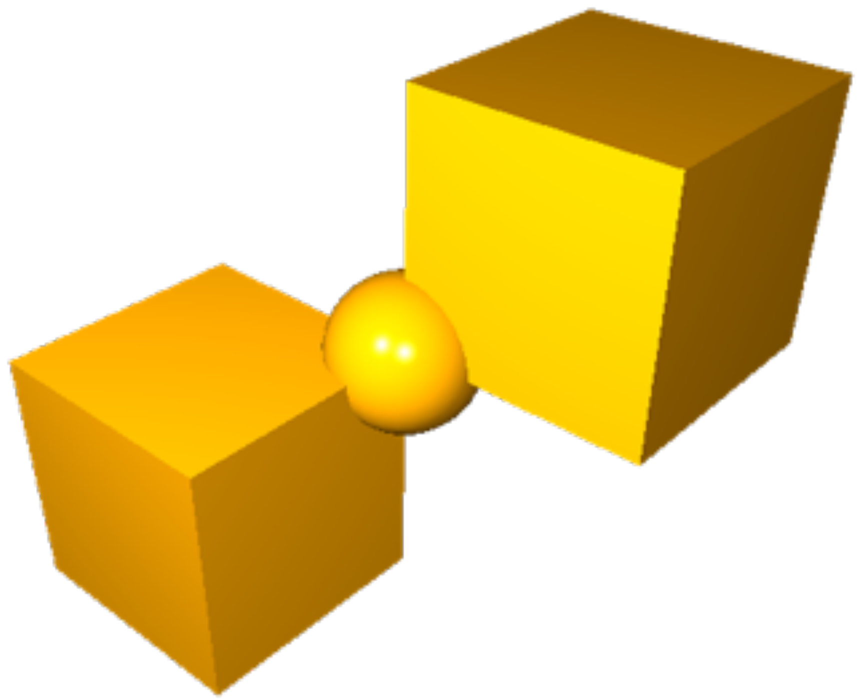
- parametric
  - disk-shaped neighborhoods
  - $\mathbf{f}(D_\varepsilon[u,v]) = D_\delta[\mathbf{f}(u,v)]$ + injectivity
- implicit
  - surface of a "physical" solid
  - $F(x,y,z) = c, \quad \|\nabla F(x,y,z)\| \neq 0$

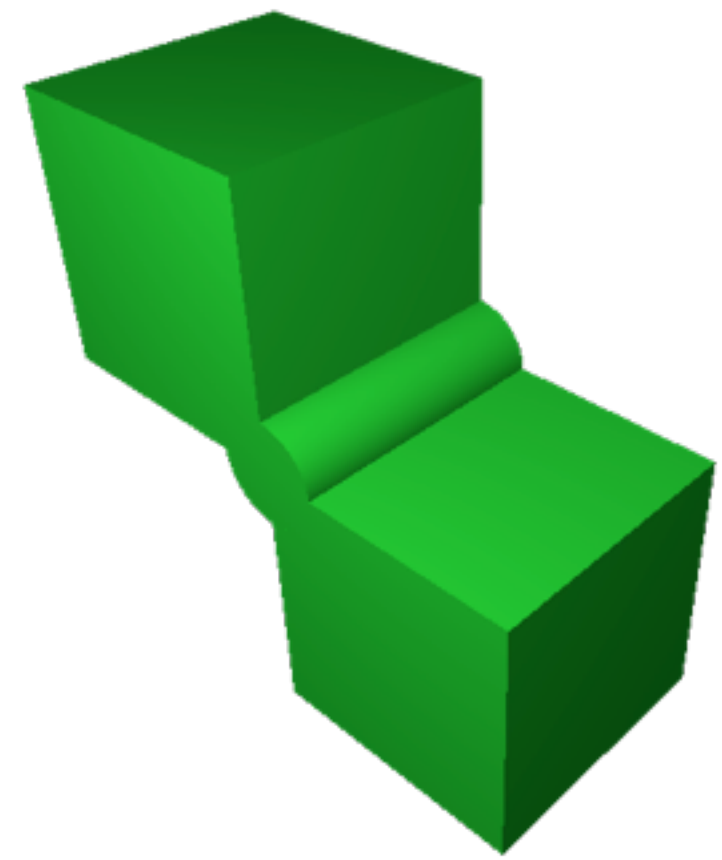
Closed 2-Manifolds



- implicit
  - surface of a “physical” solid
  - $F(x, y, z) = c, \quad \|\nabla F(x, y, z)\| \neq 0$

Closed 2-Manifolds

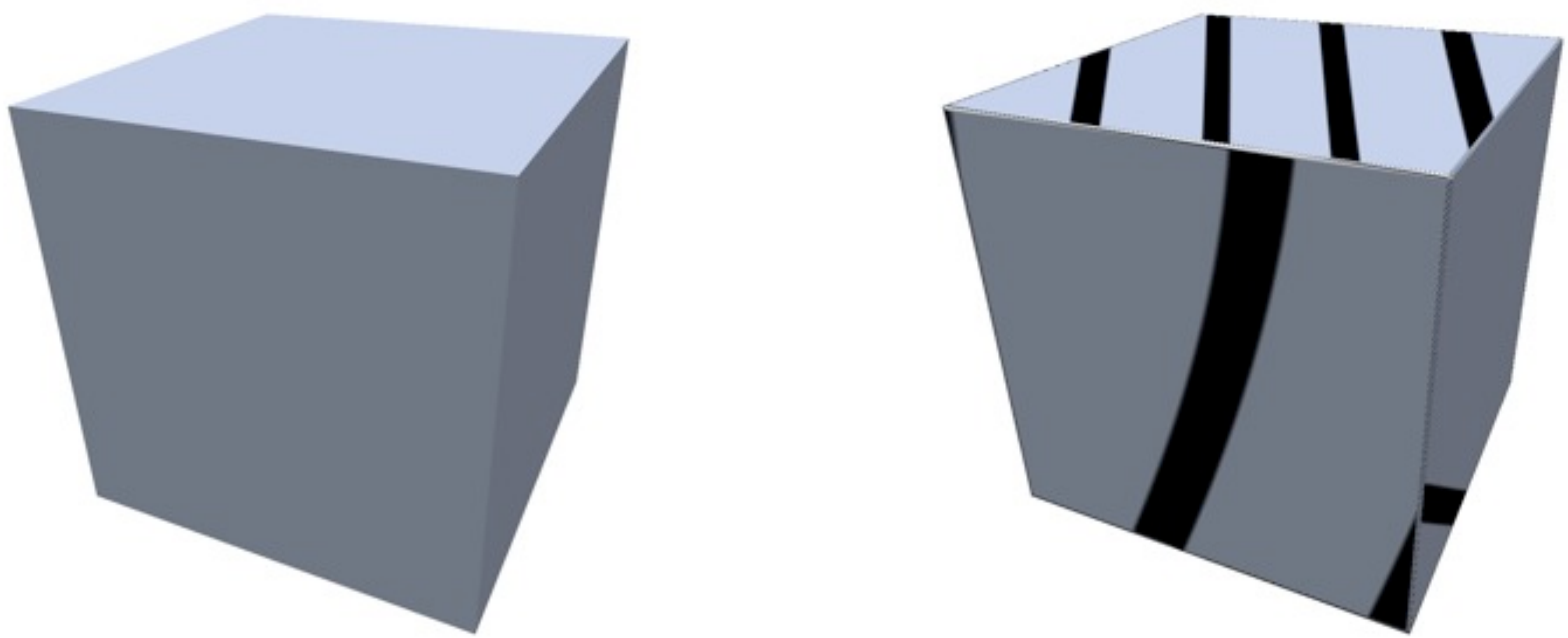




- implicit
  - surface of a “physical” solid
  - $F(x, y, z) = c, \quad \|\nabla F(x, y, z)\| \neq 0$

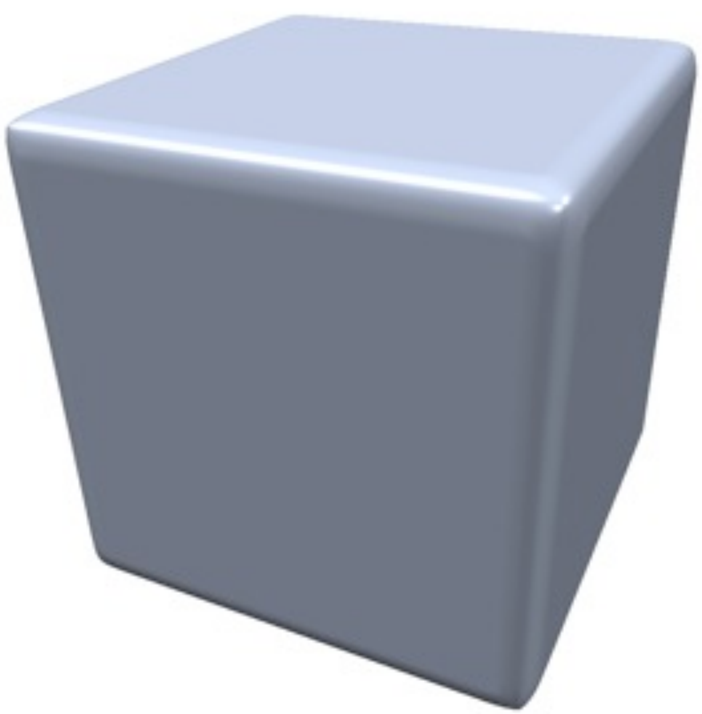
Smoothness

- position continuity : C^0
- tangent continuity : C^1
- curvature continuity : C^2



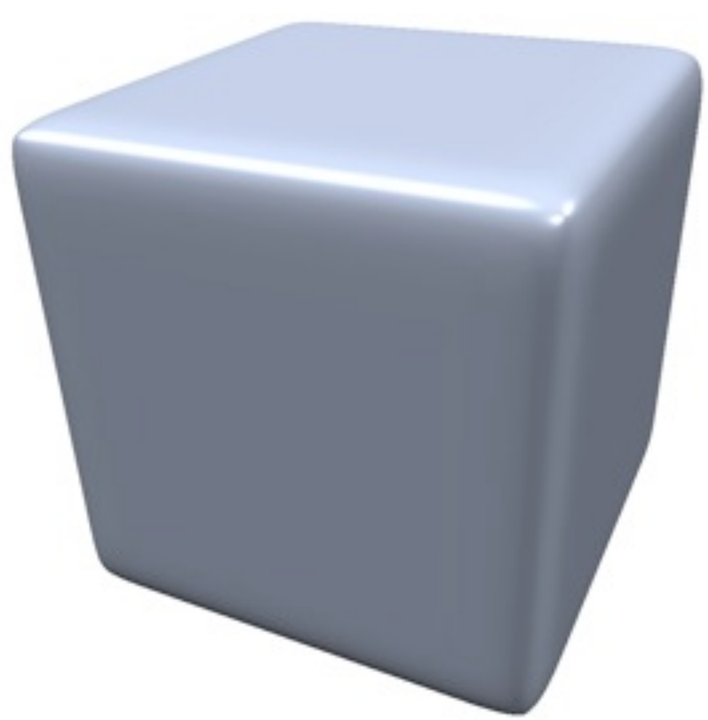
Smoothness

- position continuity : C^0
- tangent continuity : C^1
- curvature continuity : C^2



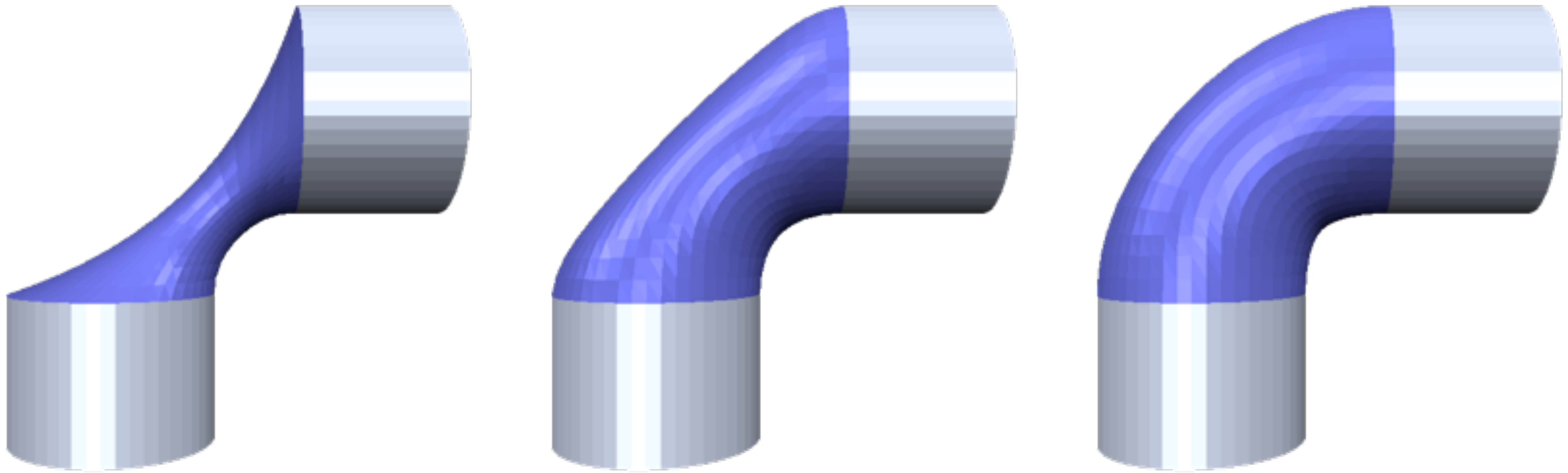
Smoothness

- position continuity : C^0
- tangent continuity : C^1
- curvature continuity : C^2



Fairness

- minimum surface area
- minimum curvature
- minimum curvature variation



Outline

- (mathematical) geometry representations
  – parametric vs. implicit
- approximation properties
- types of operations
  – distance queries
  – evaluation
  – modification / deformation
- data structures

Polynomials

- computable functions

$$\mathbf{p}(t) \;=\; \sum_{i=0}^{p} \mathbf{c}_i \, t^i \;=\; \sum_{i=0}^{p} \mathbf{c}'_i \, \Phi_i(t)$$

- Taylor expansion

$$\mathbf{f}(h) \;=\; \sum_{i=0}^{p} \frac{1}{i!} \, \mathbf{f}^{(i)}(0) \, h^i \;+\; O(h^{p+1})$$

- interpolation error (mean value theorem)

$$\mathbf{p}(t_i) = \mathbf{f}(t_i), \quad t_i \in [0, h]$$

$$\|\mathbf{f}(t) - \mathbf{p}(t)\| \;=\; \frac{1}{(p+1)!} \, \mathbf{f}^{(p+1)}(t^*) \, \prod_{i=0}^{p} (t - t_i) \;=\; O(h^{(p+1)})$$

Polynomials

- computable functions

$$\mathbf{p}(t) = \sum_{i=0}^{p} \mathbf{c}_i \, t^i = \sum_{i=0}^{p} \mathbf{c}'_i \, \Phi_i(t)$$

- Taylor expansion

$$\mathbf{f}(h) = \sum_{i=0}^{p} \frac{1}{i!} \mathbf{f}^{(i)}(0) \, h^i + O(h^{p+1})$$

- interpolation error (mean value theorem)

$$\mathbf{p}(t_i) = \mathbf{f}(t_i), \quad t_i \in [0, h]$$

$$\|\mathbf{f}(t) - \mathbf{p}(t)\| = \frac{1}{(p+1)!} \mathbf{f}^{(p+1)}(t^*) \prod_{i=0}^{p} (t - t_i) = O(h^{(p+1)})$$

Polynomials

- computable functions

$$\mathbf{p}(t) \;=\; \sum_{i=0}^{p} \mathbf{c}_i\, t^i \;=\; \sum_{i=0}^{p} \mathbf{c}'_i\, \Phi_i(t)$$

- Taylor expansion

$$\mathbf{f}(h) \;=\; \sum_{i=0}^{p} \frac{1}{i!}\, \mathbf{f}^{(i)}(0)\, h^i \;+\; O(h^{p+1})$$

- interpolation error (mean value theorem)

$$\mathbf{p}(t_i) = \mathbf{f}(t_i), \quad t_i \in [0,h]$$

$$\|\mathbf{f}(t) - \mathbf{p}(t)\| \;=\; \frac{1}{(p+1)!}\, \mathbf{f}^{(p+1)}(t^*) \prod_{i=0}^{p}(t - t_i) \;=\; O(h^{(p+1)})$$

Implicit Polynomials

- interpolation error of the function values

$$\|F(x,y,z) - P(x,y,z)\| \;=\; O(h^{(p+1)})$$

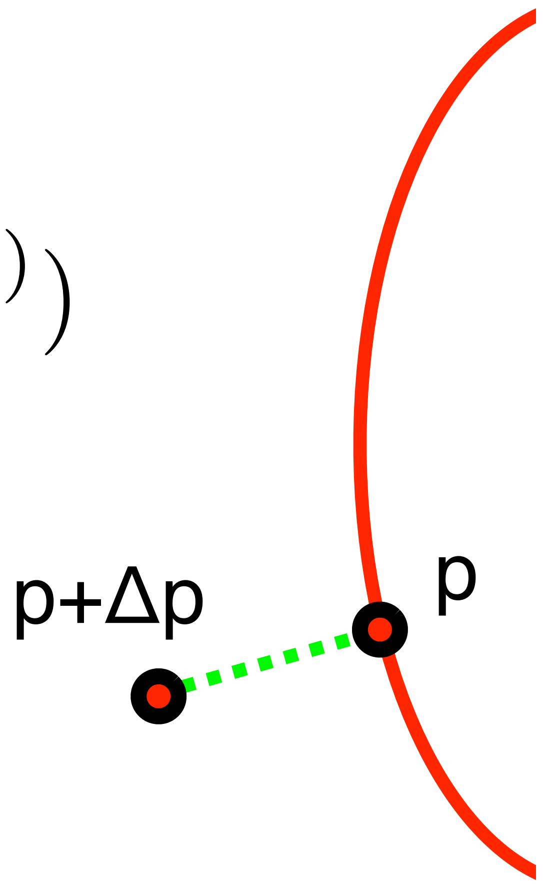
- approximation error of the contour

$$\triangle\mathbf{p} = \lambda\,\nabla F(\mathbf{p}) \qquad \frac{F(\mathbf{p}+\triangle\mathbf{p}) - F(\mathbf{p})}{\|\triangle\mathbf{p}\|} \;\approx\; \|\nabla F(\mathbf{p})\|$$

Implicit Polynomials

- interpolation error of the function values

$$\|F(x,y,z) - P(x,y,z)\| = O(h^{(p+1)})$$



- approximation error of the contour

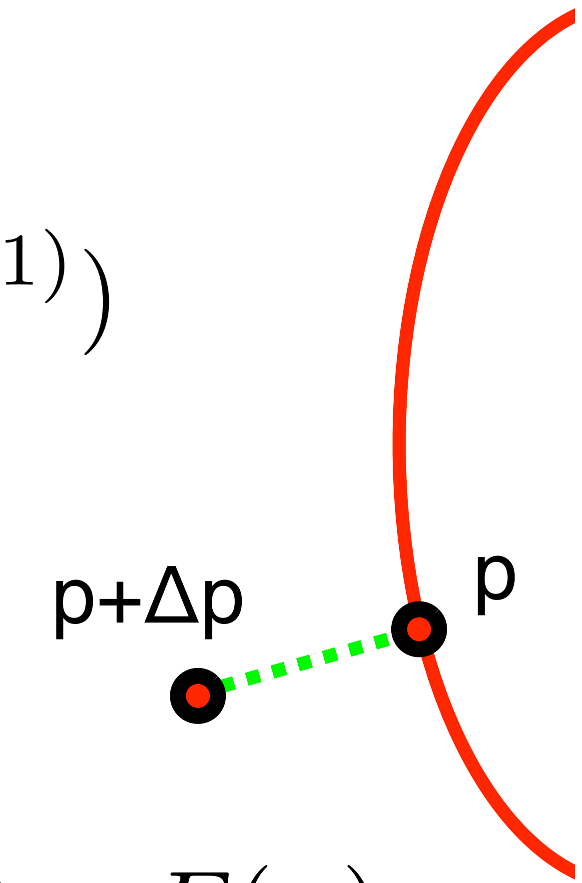
$$\triangle \mathbf{p} = \lambda \nabla F(\mathbf{p}) \qquad \frac{F(\mathbf{p}+\triangle\mathbf{p}) - F(\mathbf{p})}{\|\triangle\mathbf{p}\|} \approx \|\nabla F(\mathbf{p})\|$$

Implicit Polynomials

- interpolation error of the function values

$$\|F(x,y,z) - P(x,y,z)\| = O(h^{(p+1)})$$

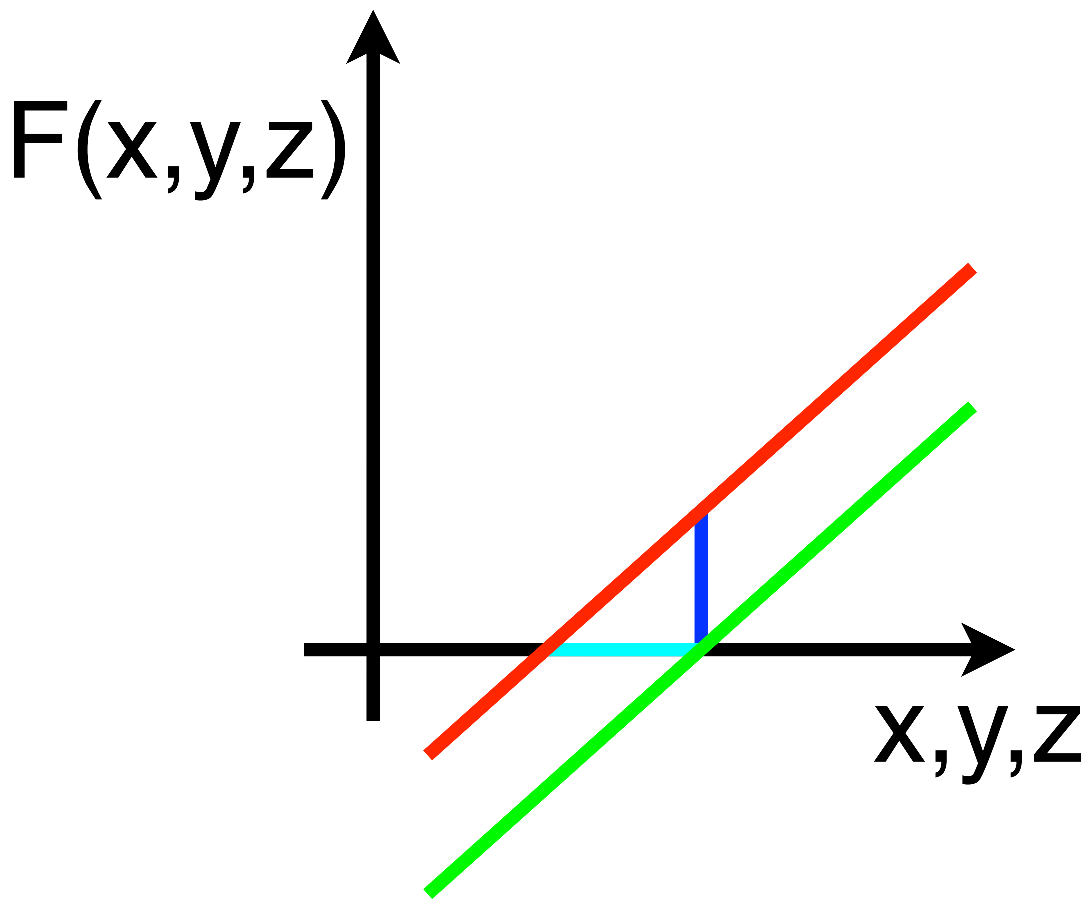
- approximation error of the contour



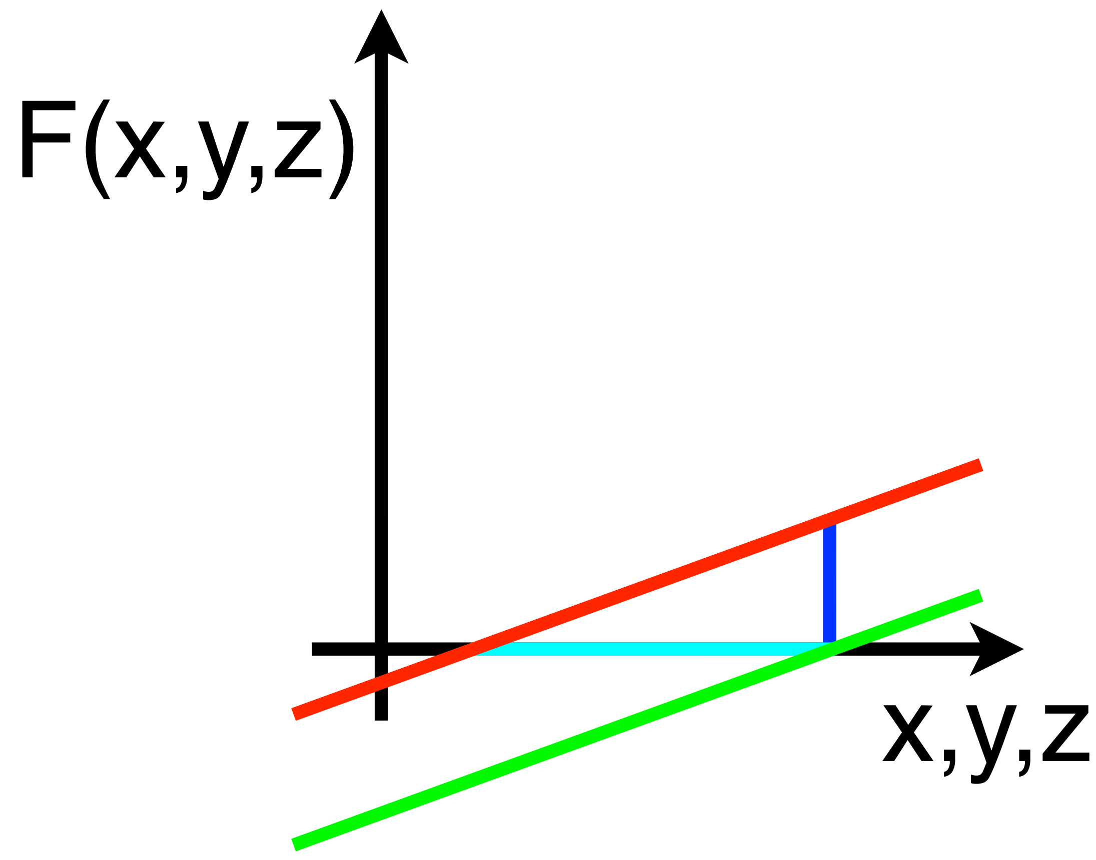
$$\triangle \mathbf{p} = \lambda \nabla F(\mathbf{p}) \qquad \|\triangle \mathbf{p}\| \approx \frac{F(\mathbf{p} + \triangle \mathbf{p}) - F(\mathbf{p})}{\|\nabla F(\mathbf{p})\|}$$

(gradient bounded from below)

Implicit Polynomials



large
gradient



small
gradient

Polynomial Approximation

- approximation error is $\mathbf{O(h^{p+1})}$
- improve approximation quality by
  - increasing **p** ... higher order polynomials
  - decreasing **h** ... smaller / more segments
- issues
  - smoothness of the target data ($\max_t f^{(p+1)}(t)$)
  - handling higher order patches (e.g. boundary conditions)

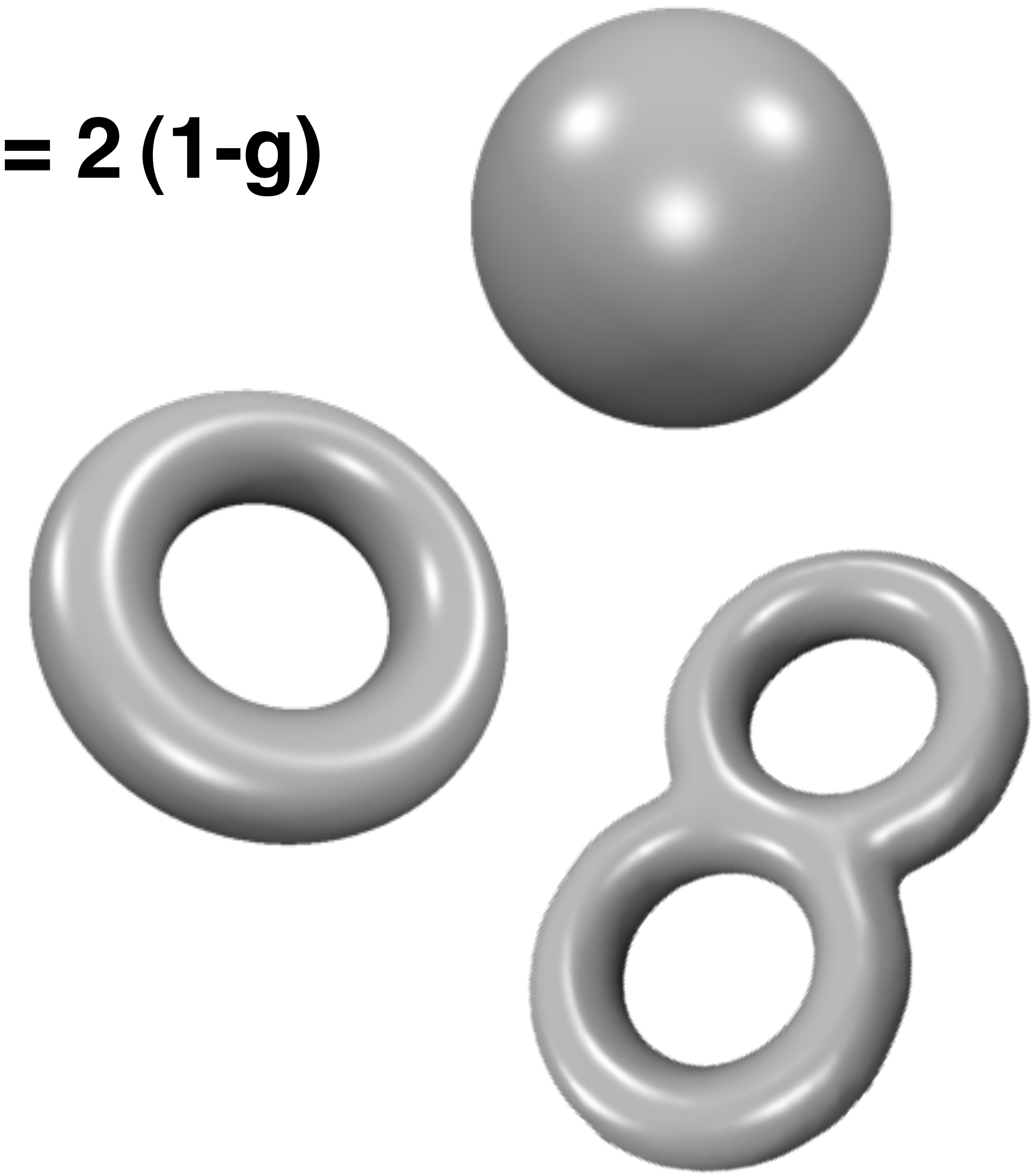
Polynomial Approximation

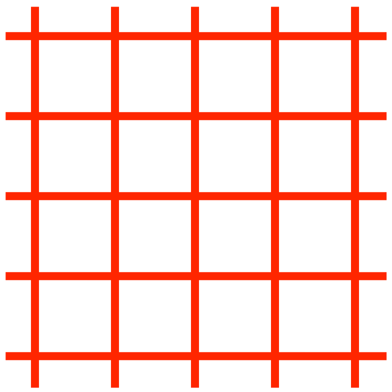
- approximation error is $\mathbf{O(h^{p+1})}$
- improve approximation quality by
  - increasing **p** ... higher order polynomials
  - decreasing **h** ... smaller / more segments
- issues
  - smoothness of the target data ($\max_t f^{(p+1)}(t)$)
  - handling higher order patches (e.g. boundary conditions)

MOTD: p = 1

Piecewise Definition

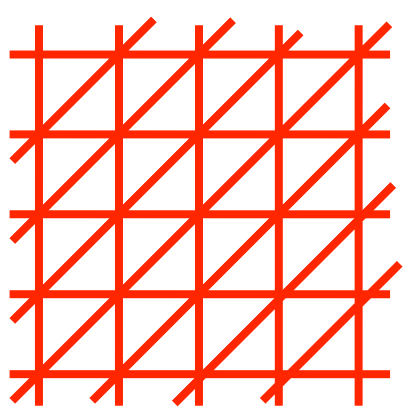
- parametric
  - Euler formula: **V - E + F = 2 (1-g)**
  - regular **quad** meshes
    - F ≈ V
    - E ≈ 2V
    - average valence = 4
  - quasi-regular
  - semi-regular

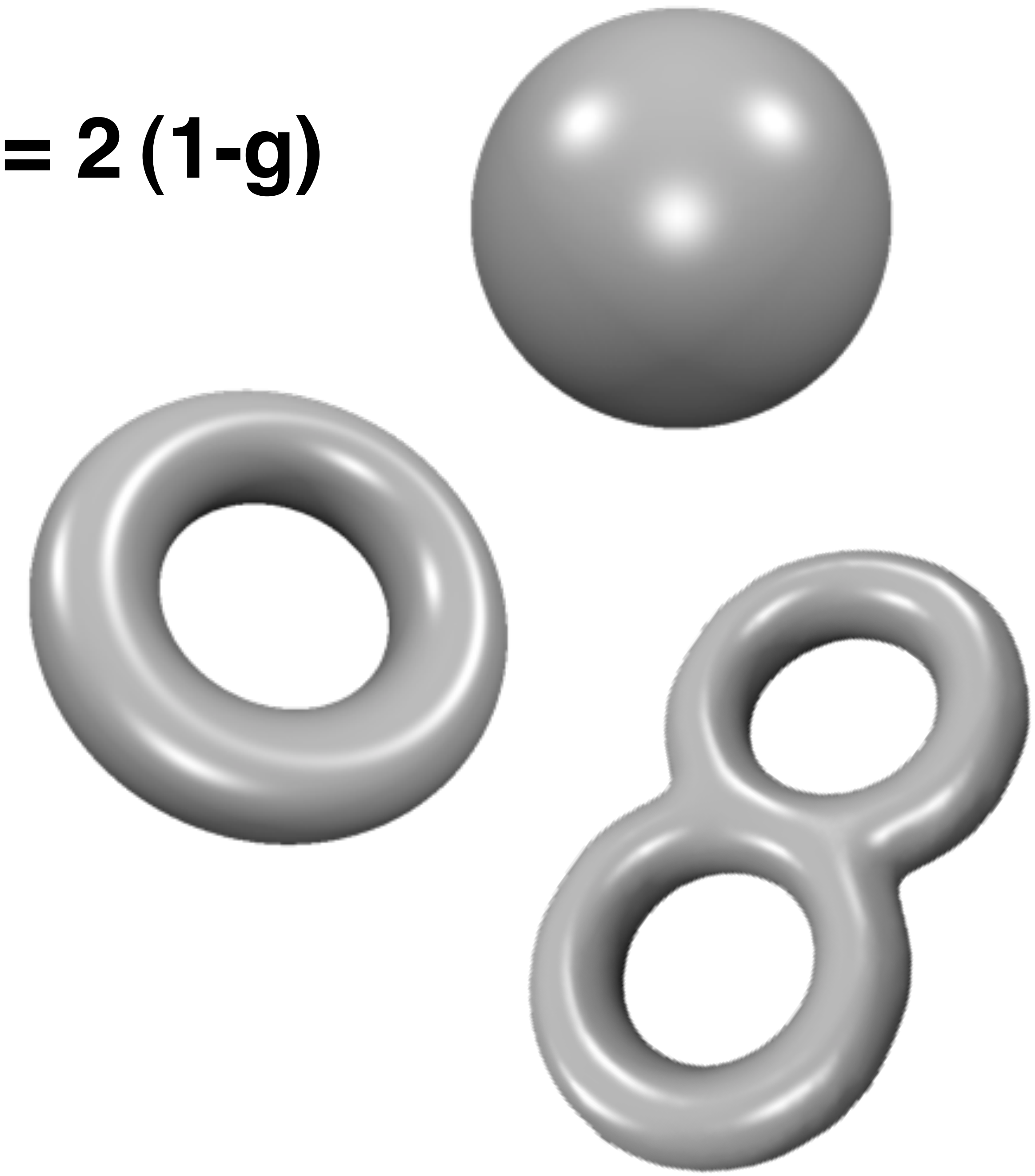




Piecewise Definition

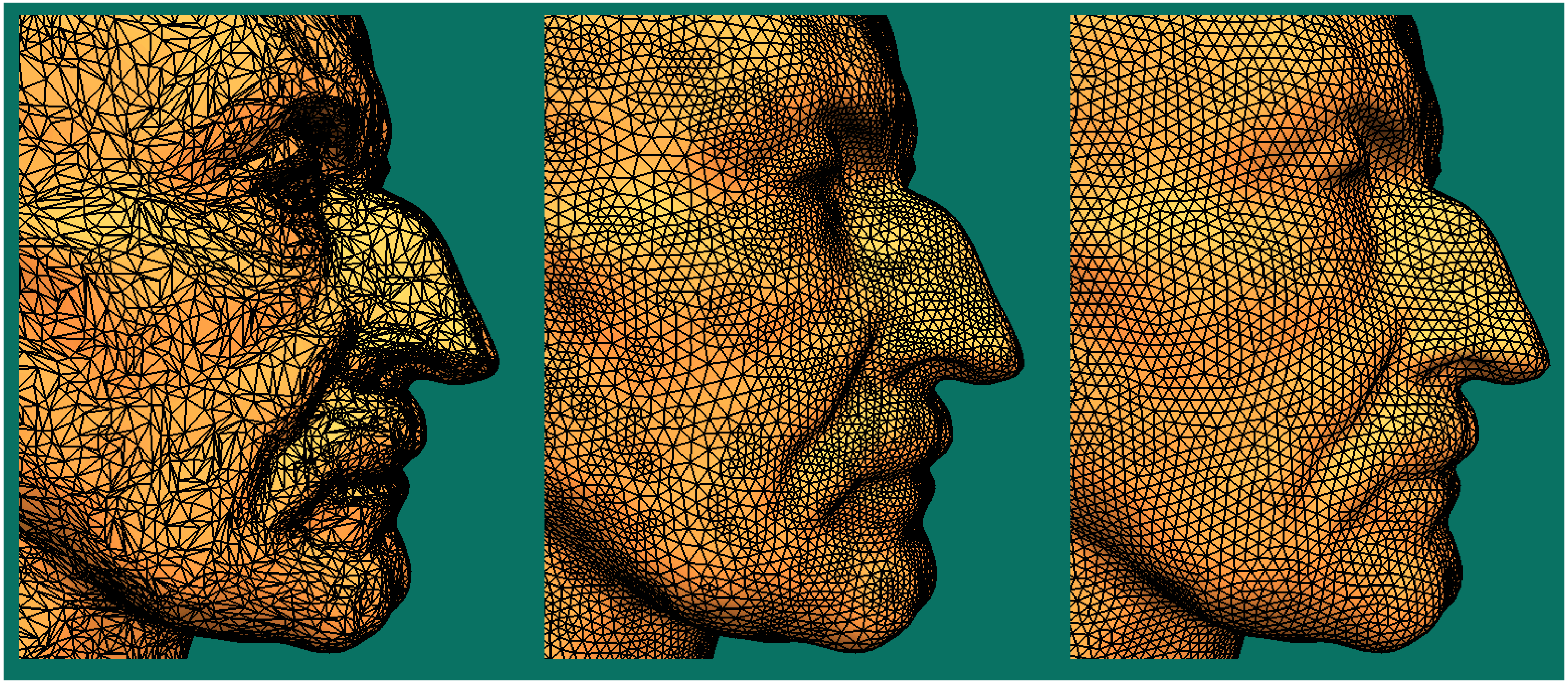
- parametric
  - Euler formula: **V - E + F = 2 (1-g)**
  - regular **triangle** meshes
    - F ≈ 2V
    - E ≈ 3V
    - average valence = 6
  - quasi-regular
  - semi-regular





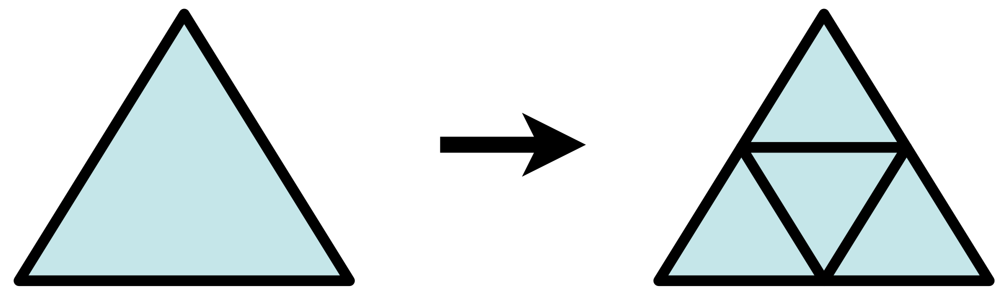
Piecewise Definition

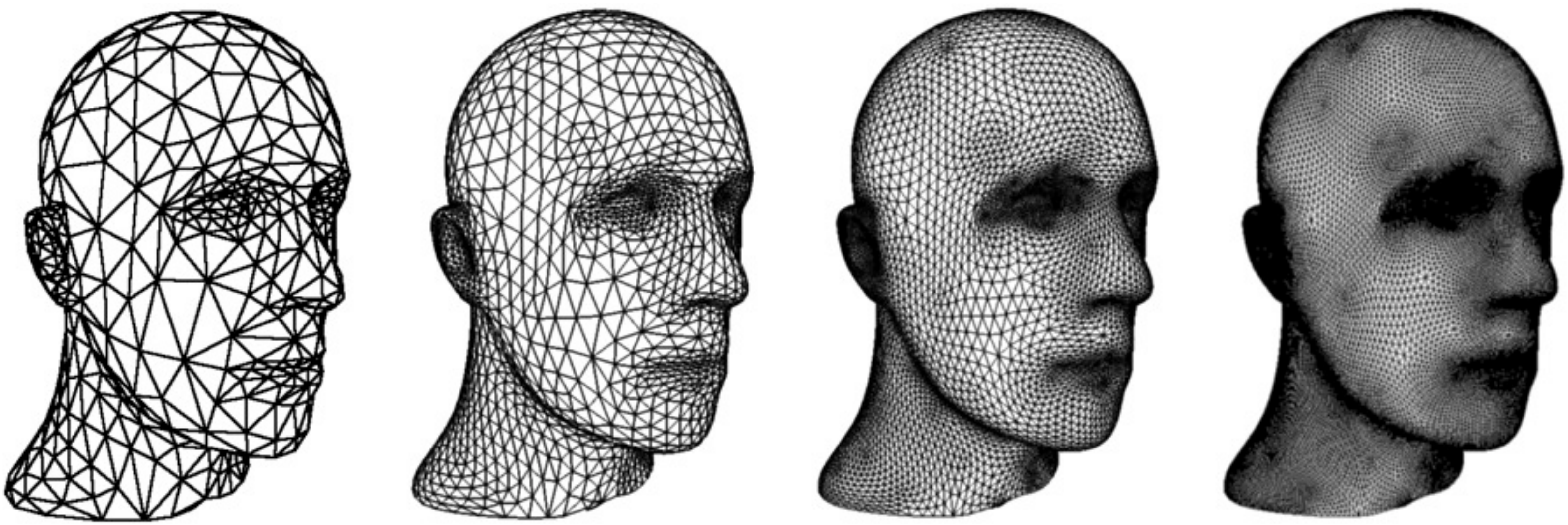
- quasi regular (→ remeshing, Pierre)



Piecewise Definition

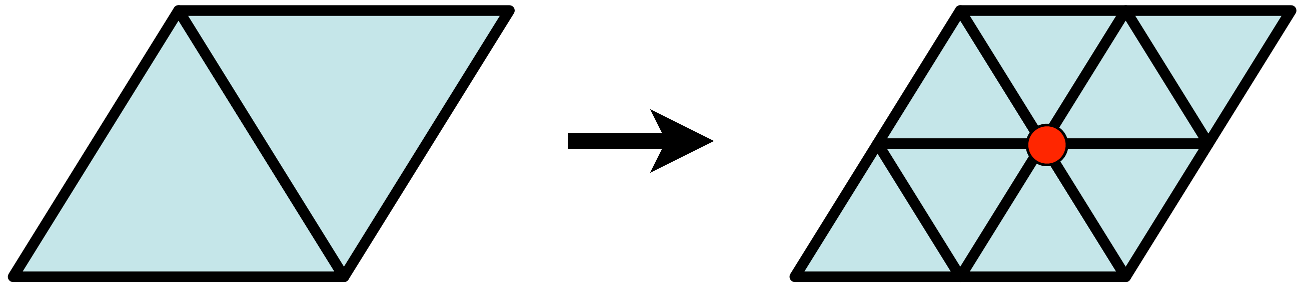
- semi regular

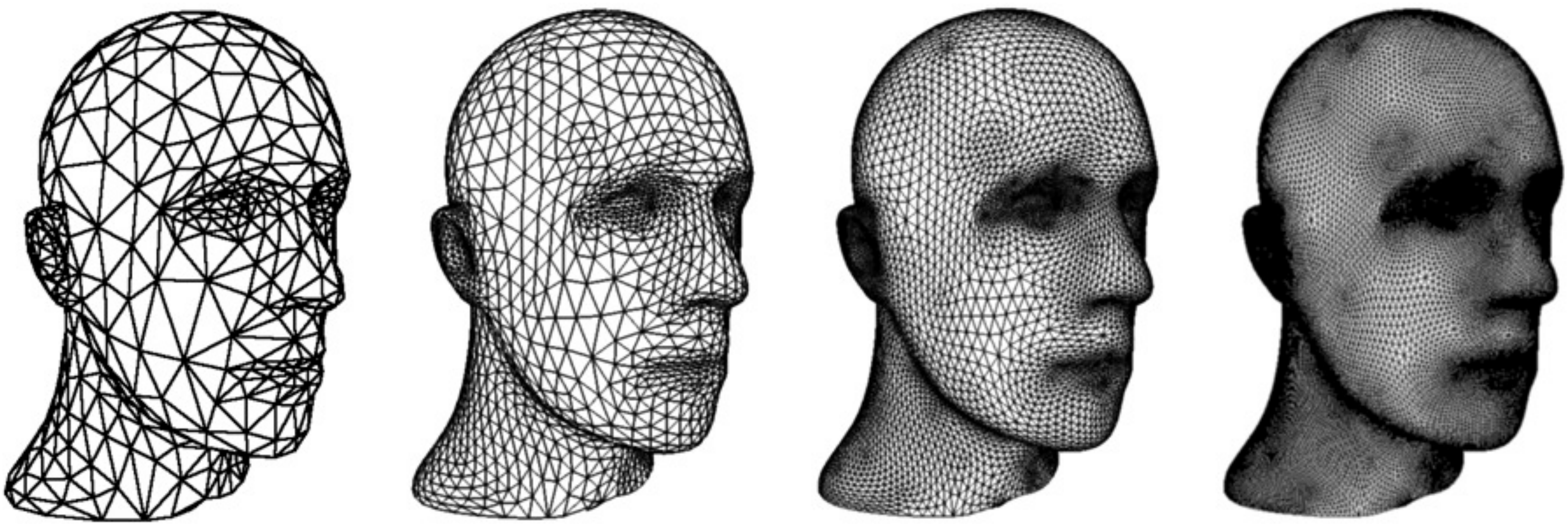




Piecewise Definition

- semi regular

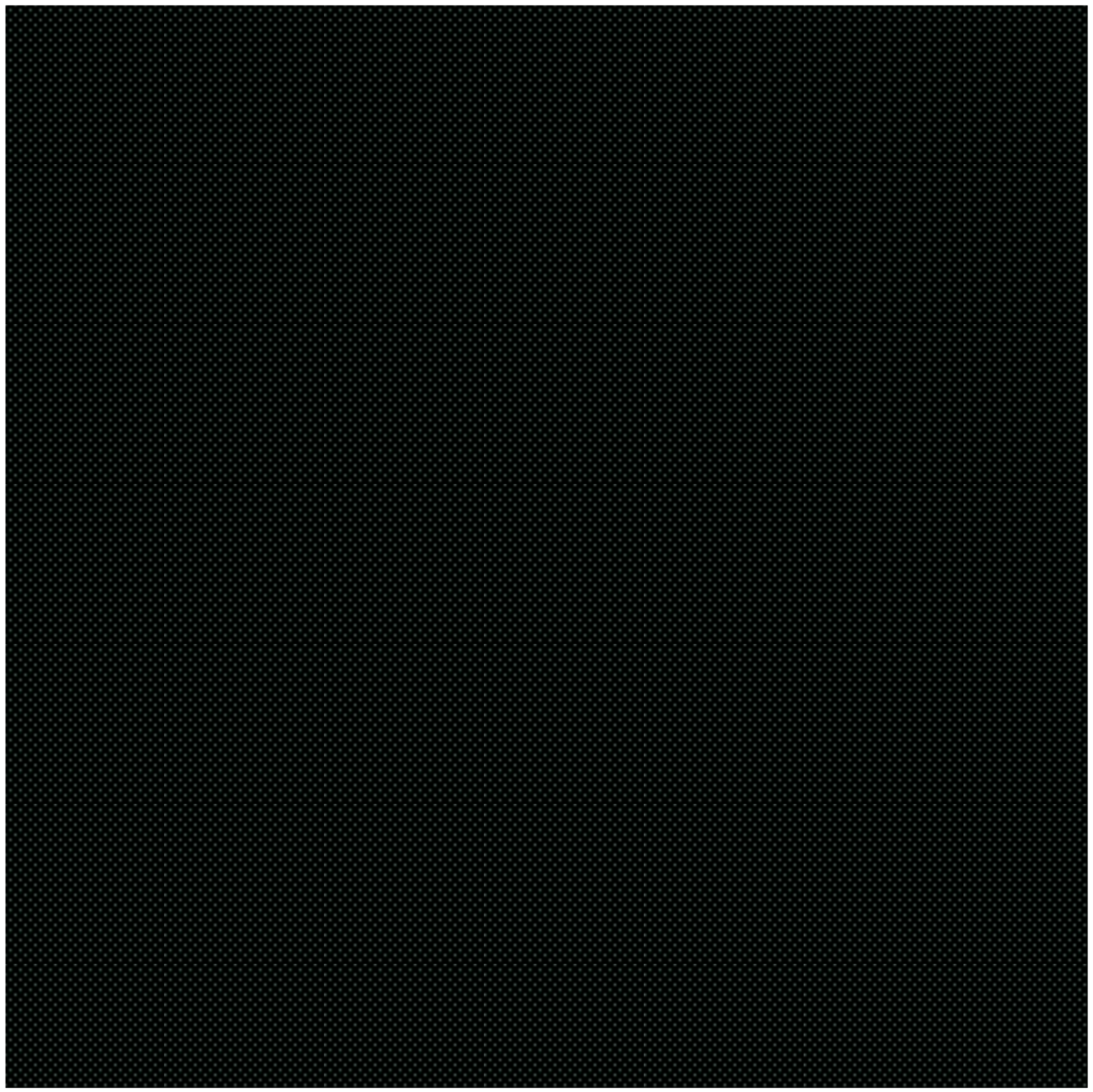




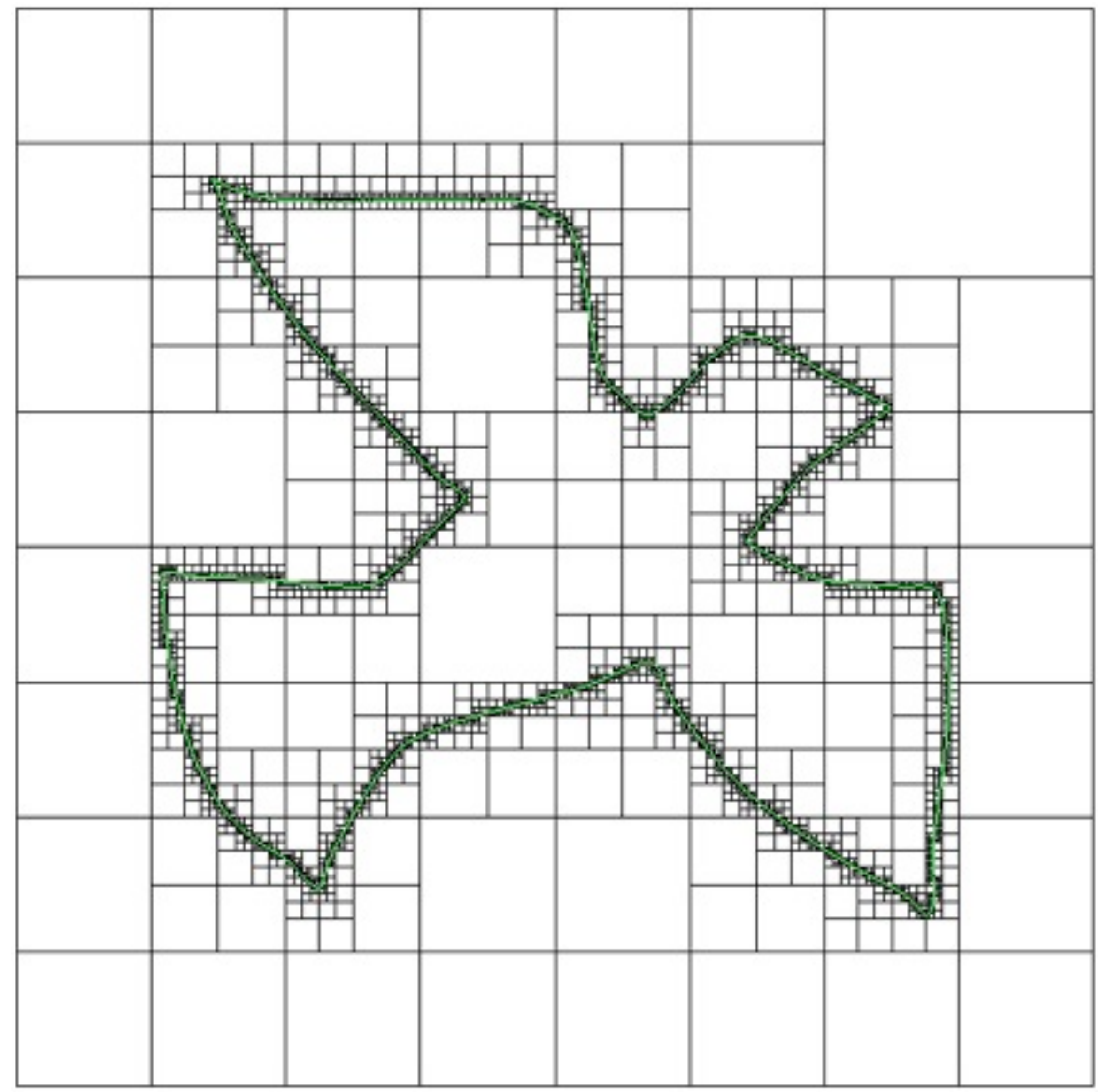
Piecewise Definition

- implicit
  - regular voxel grids $O(h^{-3})$
  - three color octrees
    - surface-adaptive refinement $O(h^{-2})$
    - feature-adaptive refinement $O(h^{-1})$
  - irregular hierarchies
    - binary space partition $O(h^{-1})$
      (BSP)

3-Color Octree

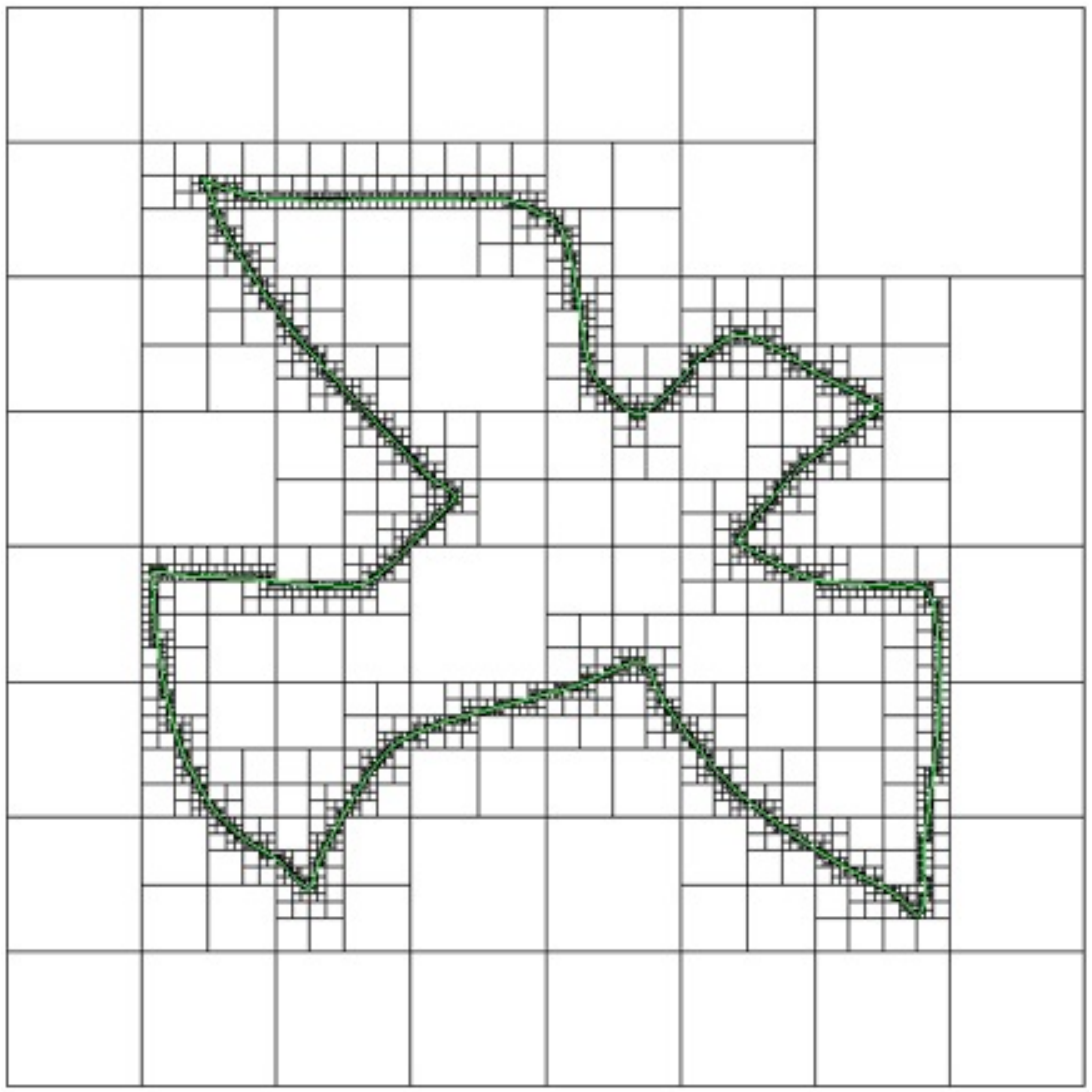


1048576 cells

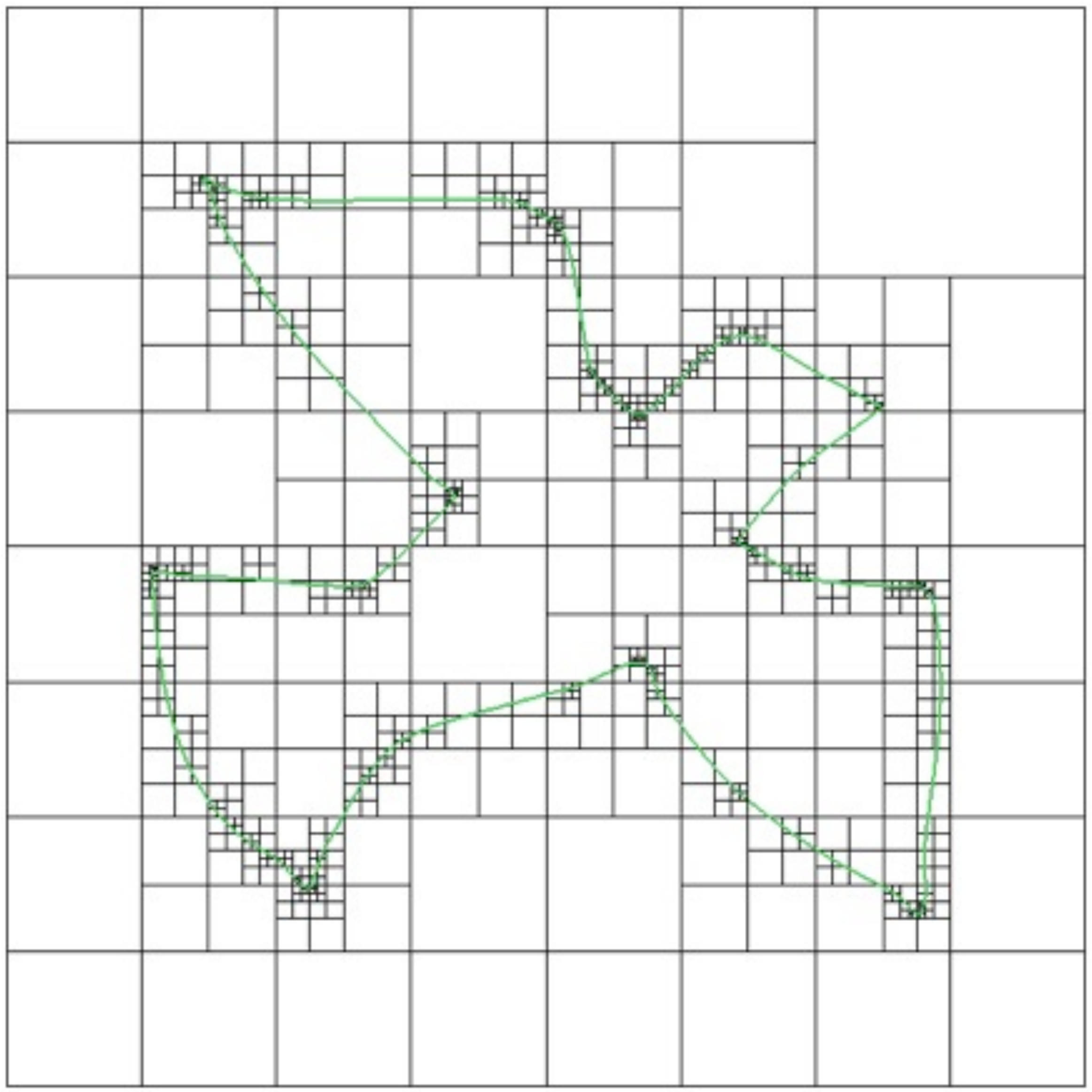


12040 cells

Adaptively Sampled Distance Fields

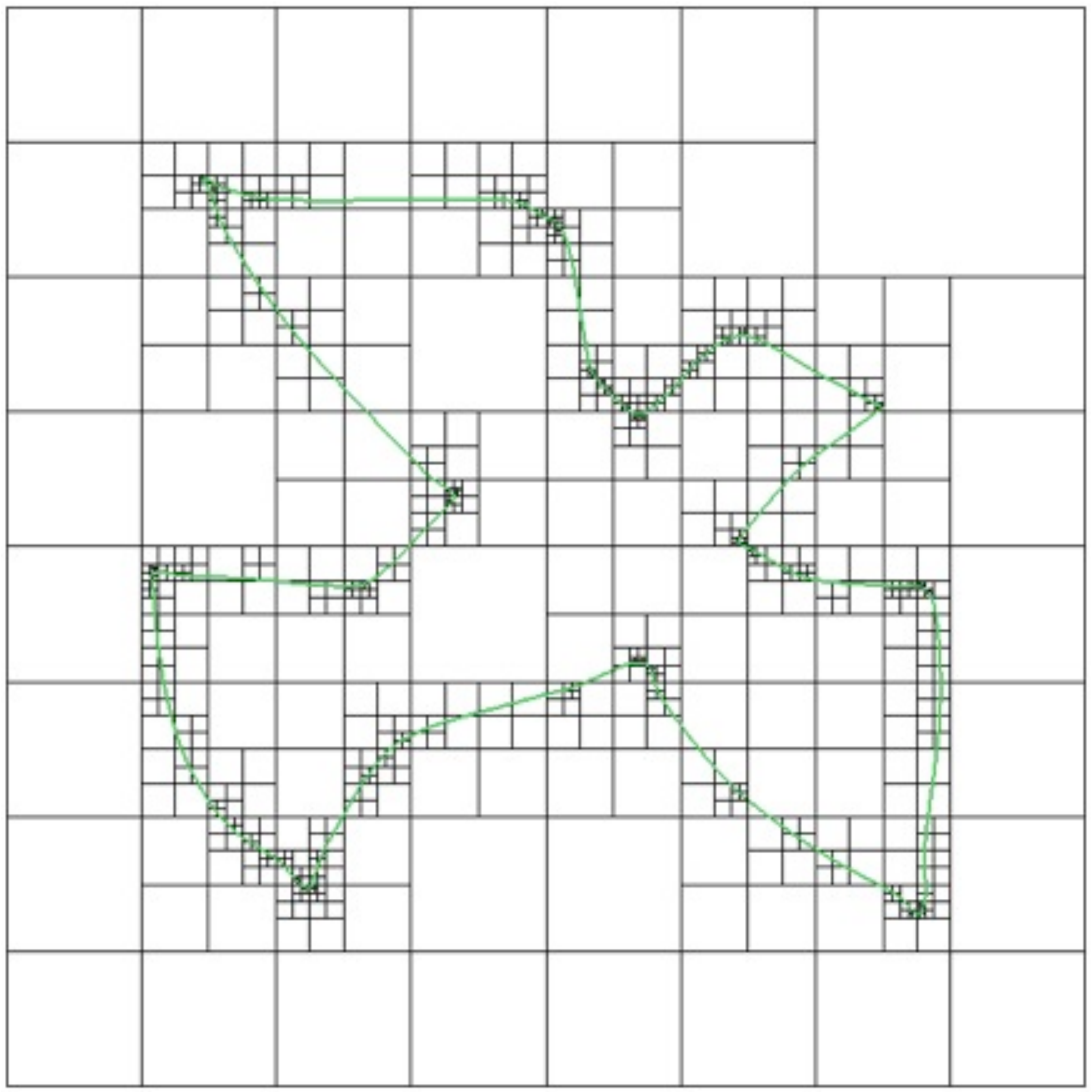


12040 cells

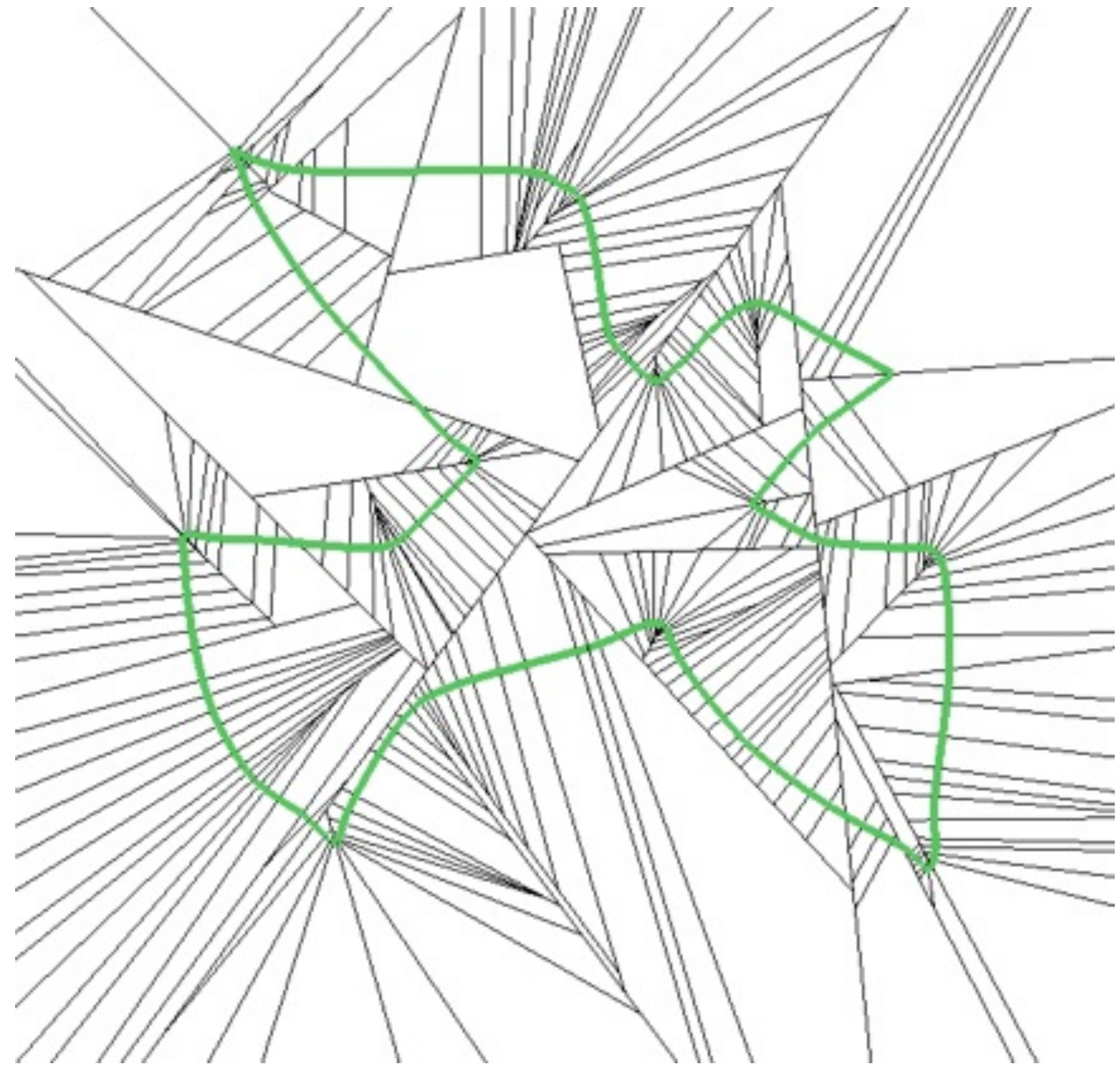


895 cells

Binary Space Partitions



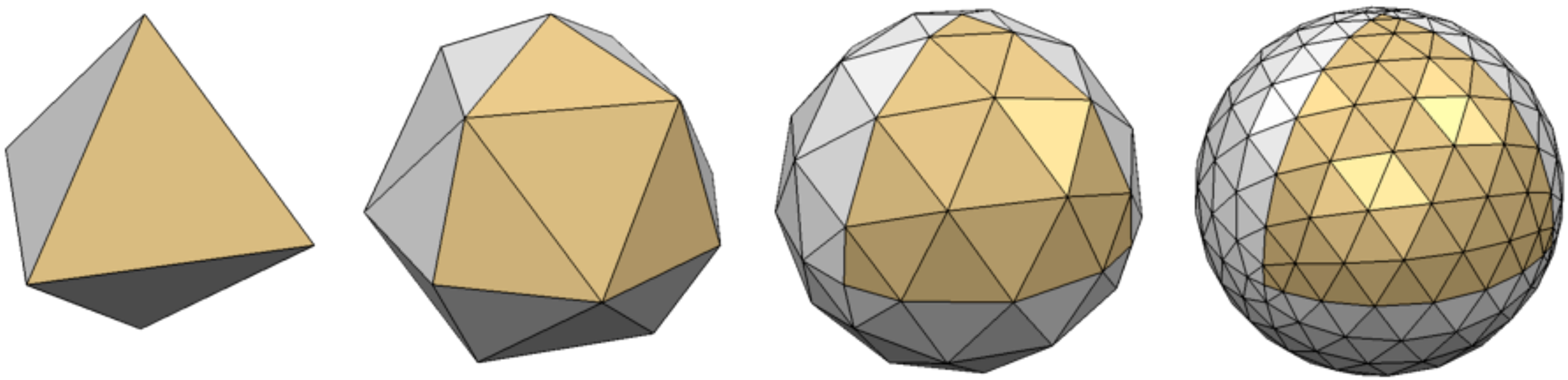
895 cells



254 cells

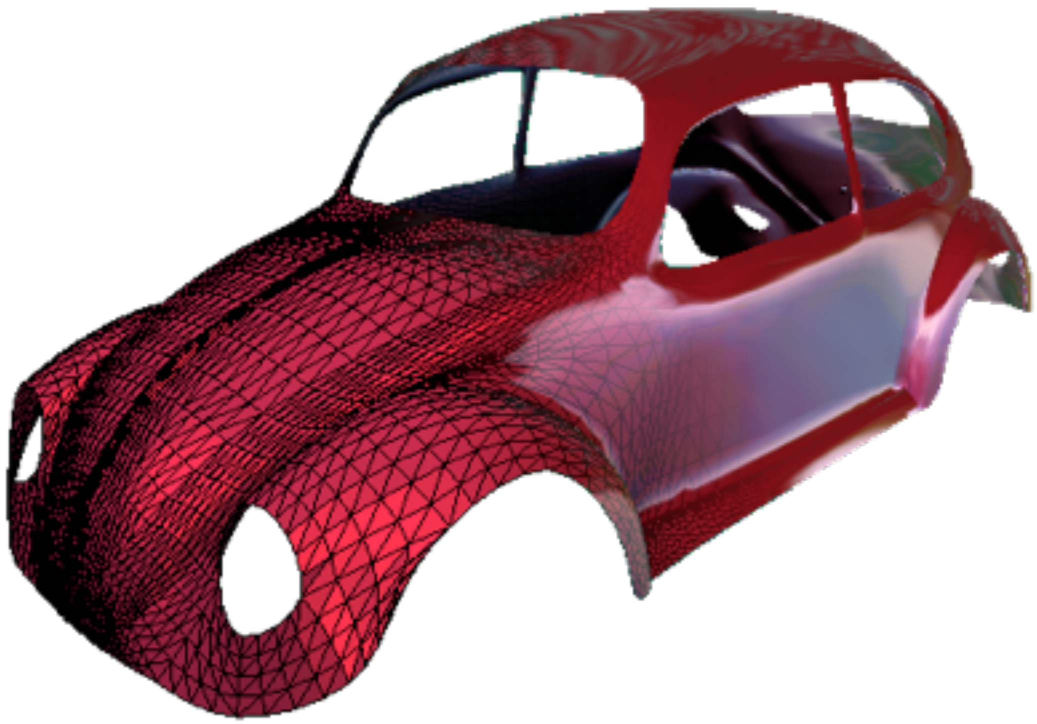
Message of the Day ...

- polygonal meshes are a good compromise
  - approximation $O(h^2)$ ... error * #faces = const.
  - arbitrary topology
  - flexibility for piecewise smooth surfaces
  - flexibility for adaptive refinement



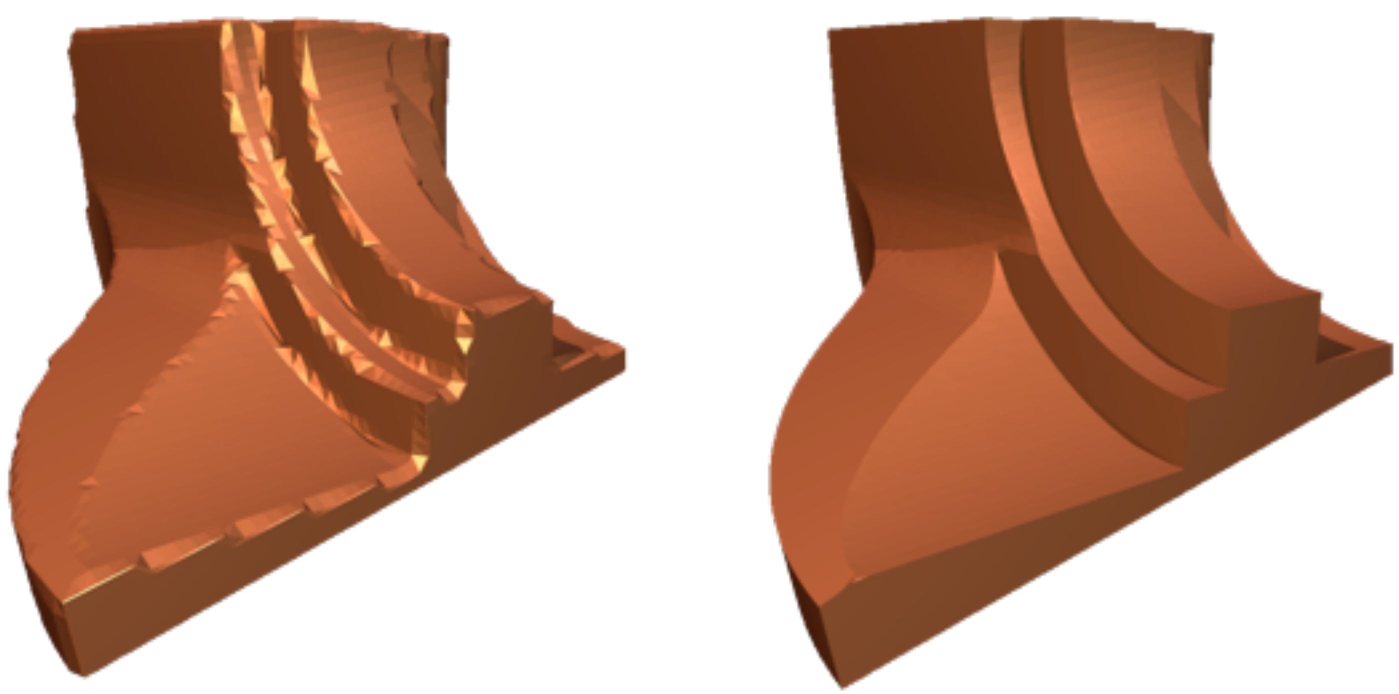
Message of the Day ...

- polygonal meshes are a good compromise
  - approximation $O(h^2)$ ... error * #faces = const.
  - arbitrary topology
  - flexibility for piecewise smooth surfaces
  - flexibility for adaptive refinement



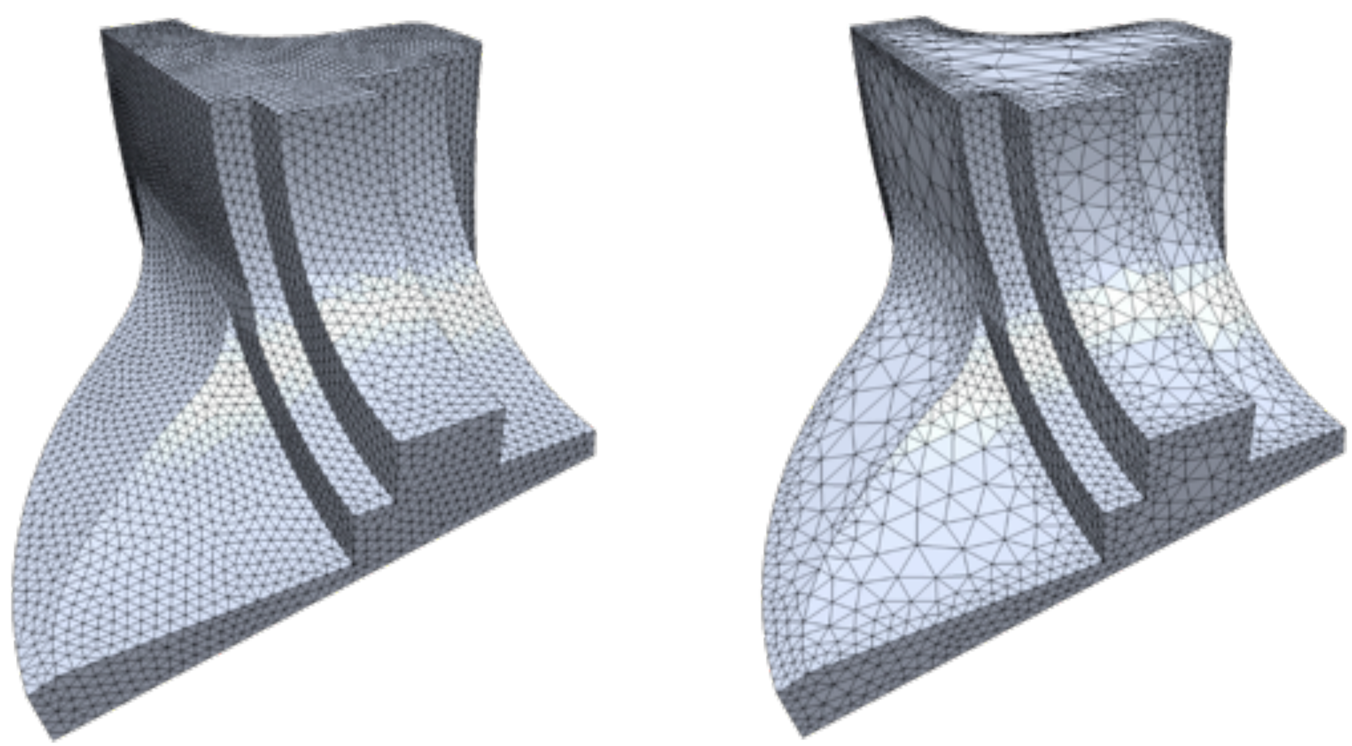
Message of the Day ...

- polygonal meshes are a good compromise
  - approximation $O(h^2)$ ... error * #faces = const.
  - arbitrary topology
  - flexibility for piecewise smooth surfaces
  - flexibility for adaptive refinement



Message of the Day ...

- polygonal meshes are a good compromise
  - approximation $O(h^2)$ ... error * #faces = const.
  - arbitrary topology
  - flexibility for piecewise smooth surfaces
  - flexibility for adaptive refinement



Message of the Day ...

- polygonal meshes are a good compromise
  - approximation $O(h^2)$ ... error * #faces = const.
  - arbitrary topology
  - flexibility for piecewise smooth surfaces
  - flexibility for adaptive refinement
  - efficient rendering

Message of the Day ...

- polygonal meshes are a good compromise
  - approximation $O(h^2)$ ... error * #faces = const.
  - arbitrary topology
  - flexibility for piecewise smooth surfaces
  - flexibility for adaptive refinement
  - efficient rendering

- implicit representation can support efficient access to vertices, faces, ...

Outline

- (mathematical) geometry representations
  - parametric vs. implicit
- approximation properties
- types of operations
  - evaluation
  - distance queries
  - modification / deformation
- data structures

Evaluation

- smooth parametric surfaces
  - positions $\mathbf{f}(u,v)$
  - normals $\mathbf{n}(u,v) = \mathbf{f}_u(u,v) \times \mathbf{f}_v(u,v)$
  - curvatures $\mathbf{c}(u,v) = C\Big(\mathbf{f}_{uu}(u,v), \mathbf{f}_{uv}(u,v), \mathbf{f}_{vv}(u,v)\Big)$

Evaluation

- smooth parametric surfaces
  - positions $\mathbf{f}(u,v)$
  - normals $\mathbf{n}(u,v) \;=\; \mathbf{f}_u(u,v) \times \mathbf{f}_v(u,v)$
  - curvatures $\mathbf{c}(u,v) \;=\; C\Big(\mathbf{f}_{uu}(u,v), \mathbf{f}_{uv}(u,v), \mathbf{f}_{vv}(u,v)\Big)$
- generalization to triangle meshes
  - positions (barycentric coordinates)

$$(\alpha,\beta) \;\mapsto\; \alpha\,\mathbf{P}_1 + \beta\,\mathbf{P}_2 + (1-\alpha-\beta)\,\mathbf{P}_3$$

$$0 \leq \alpha, \quad 0 \leq \beta, \quad \alpha + \beta \leq 1$$

Evaluation

- smooth parametric surfaces
  - positions $\mathbf{f}(u,v)$
  - normals $\mathbf{n}(u,v) = \mathbf{f}_u(u,v) \times \mathbf{f}_v(u,v)$
  - curvatures $\mathbf{c}(u,v) = C\Big(\mathbf{f}_{uu}(u,v), \mathbf{f}_{uv}(u,v), \mathbf{f}_{vv}(u,v)\Big)$
- generalization to triangle meshes
  - positions (barycentric coordinates)

$$(\alpha, \beta, \gamma) \mapsto \alpha\,\mathbf{P}_1 + \beta\,\mathbf{P}_2 + \gamma\,\mathbf{P}_3$$

$$\alpha + \beta + \gamma = 1$$

Evaluation

- smooth parametric surfaces
  - positions $\mathbf{f}(u,v)$
  - normals $\mathbf{n}(u,v) \;=\; \mathbf{f}_u(u,v) \times \mathbf{f}_v(u,v)$
  - curvatures $\mathbf{c}(u,v) \;=\; C\Big(\mathbf{f}_{uu}(u,v), \mathbf{f}_{uv}(u,v), \mathbf{f}_{vv}(u,v)\Big)$
- generalization to triangle meshes
  - positions (barycentric coordinates)

$$\alpha\,\mathbf{u} + \beta\,\mathbf{v} + \gamma\,\mathbf{w} \;\mapsto\; \alpha\,\mathbf{P}_1 + \beta\,\mathbf{P}_2 + \gamma\,\mathbf{P}_3$$

$$\alpha + \beta + \gamma \;=\; 0$$

Evaluation

- smooth parametric surfaces
  - positions $\mathbf{f}(u,v)$
  - normals $\mathbf{n}(u,v) = \mathbf{f}_u(u,v) \times \mathbf{f}_v(u,v)$
  - curvatures $\mathbf{c}(u,v) = C\Big(\mathbf{f}_{uu}(u,v), \mathbf{f}_{uv}(u,v), \mathbf{f}_{vv}(u,v)\Big)$
- generalization to triangle meshes
  - positions (barycentric coordinates)

→ Parametrization
Bruno

$$\alpha\,\mathbf{u} + \beta\,\mathbf{v} + \gamma\,\mathbf{w} \mapsto \alpha\,\mathbf{P}_1 + \beta\,\mathbf{P}_2 + \gamma\,\mathbf{P}_3$$

$$\alpha + \beta + \gamma = 0$$

Evaluation

- smooth parametric surfaces
  - positions $\mathbf{f}(u,v)$
  - normals $\mathbf{n}(u,v) = \mathbf{f}_u(u,v) \times \mathbf{f}_v(u,v)$
  - curvatures $\mathbf{c}(u,v) = C\Big(\mathbf{f}_{uu}(u,v), \mathbf{f}_{uv}(u,v), \mathbf{f}_{vv}(u,v)\Big)$
- generalization to triangle meshes
  - positions (barycentric coordinates)
  - normals (per face, Phong)

$$\mathbf{N} = (\mathbf{P}_2 - \mathbf{P}_1) \times (\mathbf{P}_3 - \mathbf{P}_1)$$

Evaluation

- smooth parametric surfaces
  - positions $\mathbf{f}(u,v)$
  - normals $\mathbf{n}(u,v) = \mathbf{f}_u(u,v) \times \mathbf{f}_v(u,v)$
  - curvatures $\mathbf{c}(u,v) = C\Big(\mathbf{f}_{uu}(u,v), \mathbf{f}_{uv}(u,v), \mathbf{f}_{vv}(u,v)\Big)$
- generalization to triangle meshes
  - positions (barycentric coordinates)
  - normals (per face, Phong)

$$\alpha\,\mathbf{u} + \beta\,\mathbf{v} + \gamma\,\mathbf{w} \mapsto \alpha\,\mathbf{N}_1 + \beta\,\mathbf{N}_2 + \gamma\,\mathbf{N}_3$$

Evaluation

- smooth parametric surfaces
  - positions $\mathbf{f}(u,v)$
  - normals $\mathbf{n}(u,v) = \mathbf{f}_u(u,v) \times \mathbf{f}_v(u,v)$
  - curvatures $\mathbf{c}(u,v) = C\Big(\mathbf{f}_{uu}(u,v), \mathbf{f}_{uv}(u,v), \mathbf{f}_{vv}(u,v)\Big)$
- generalization to triangle meshes
  - positions (barycentric coordinates)
  - normals (per face, Phong)
  - curvatures ... (→ smoothing, Mark)

Distance Queries

- parametric
  - for smooth surfaces: find orthogonal base point

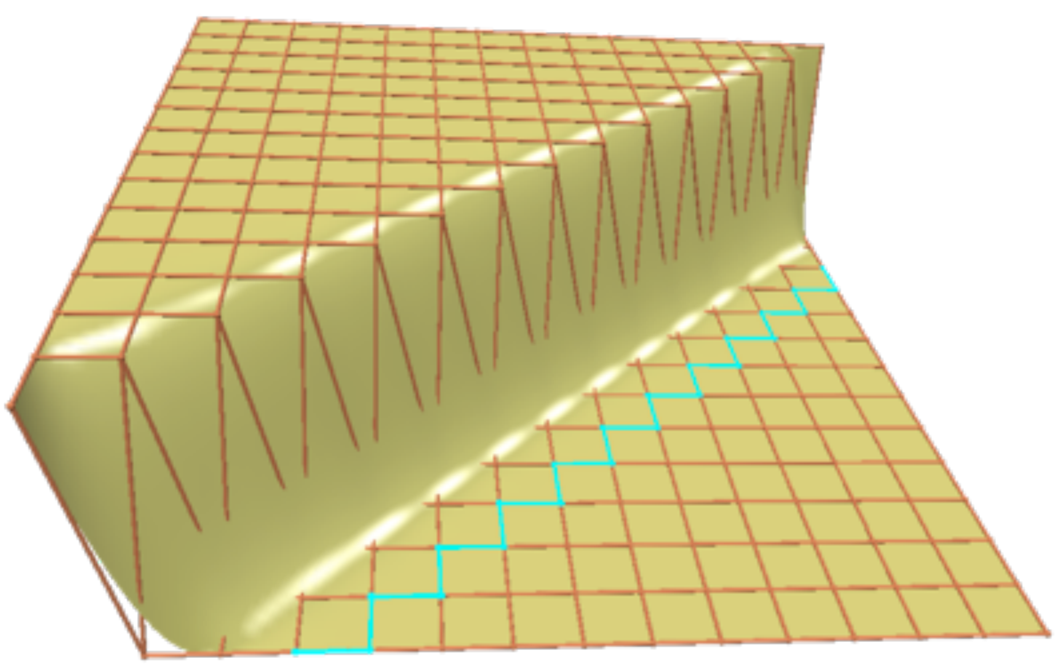
$$[\mathbf{p} - \mathbf{f}(u,v)] \times \mathbf{n}(u,v) \;=\; \mathbf{0}$$

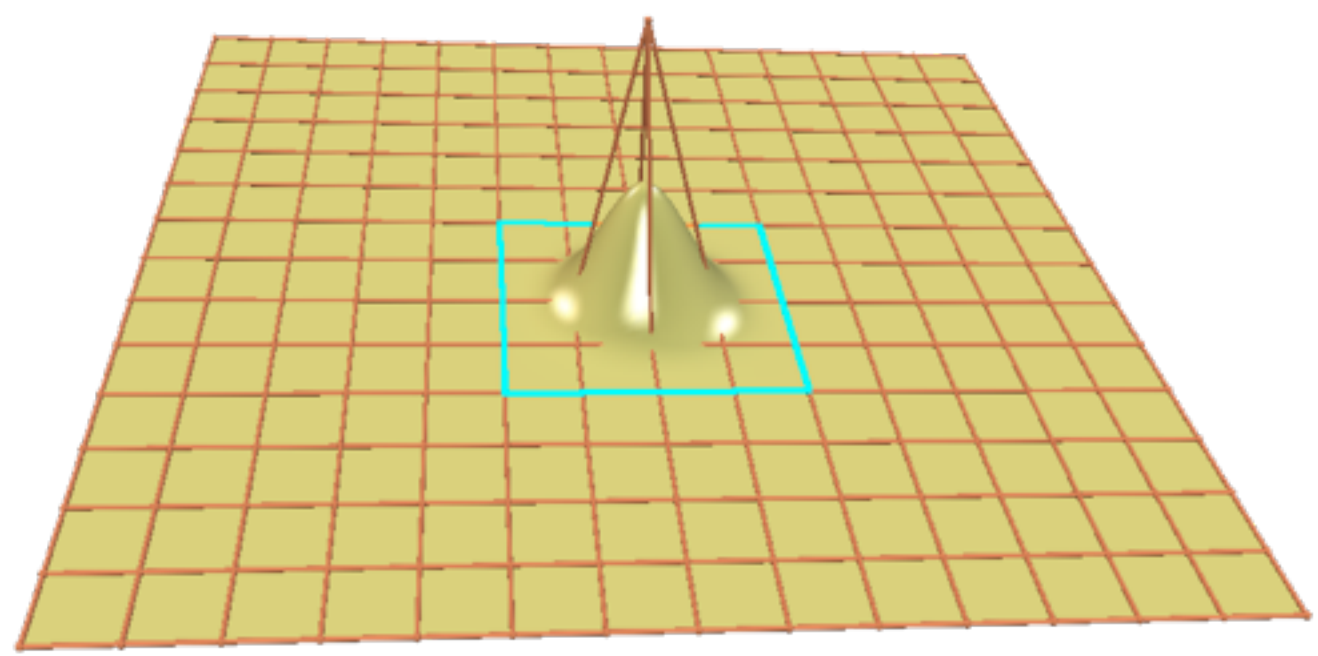
  - for triangle meshes
    - use kd-tree or BSP to find closest triangle
    - find base point by Newton iteration
      (use Phong normal field)

Modifications

- parameteric
  - control vertices
  - free-form deformation
  - boundary constraint modeling

$$\mathbf{f}(u,v) = \sum_{i=0}^{n} \sum_{j=0}^{m} \mathbf{c}_{ij} N_i^n(u) N_j^m(v)$$

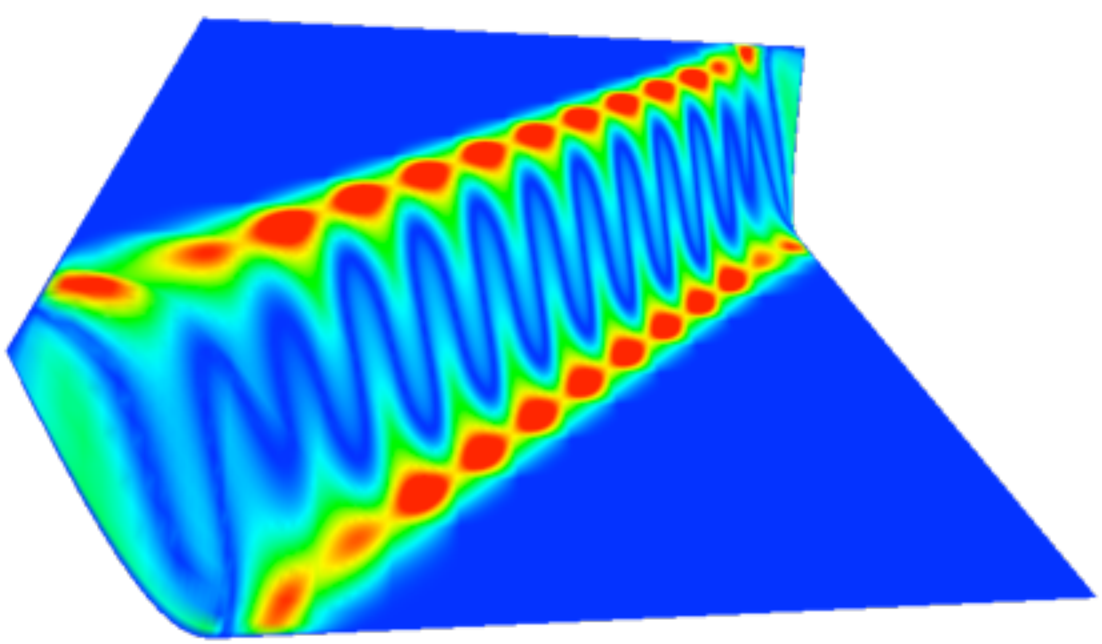


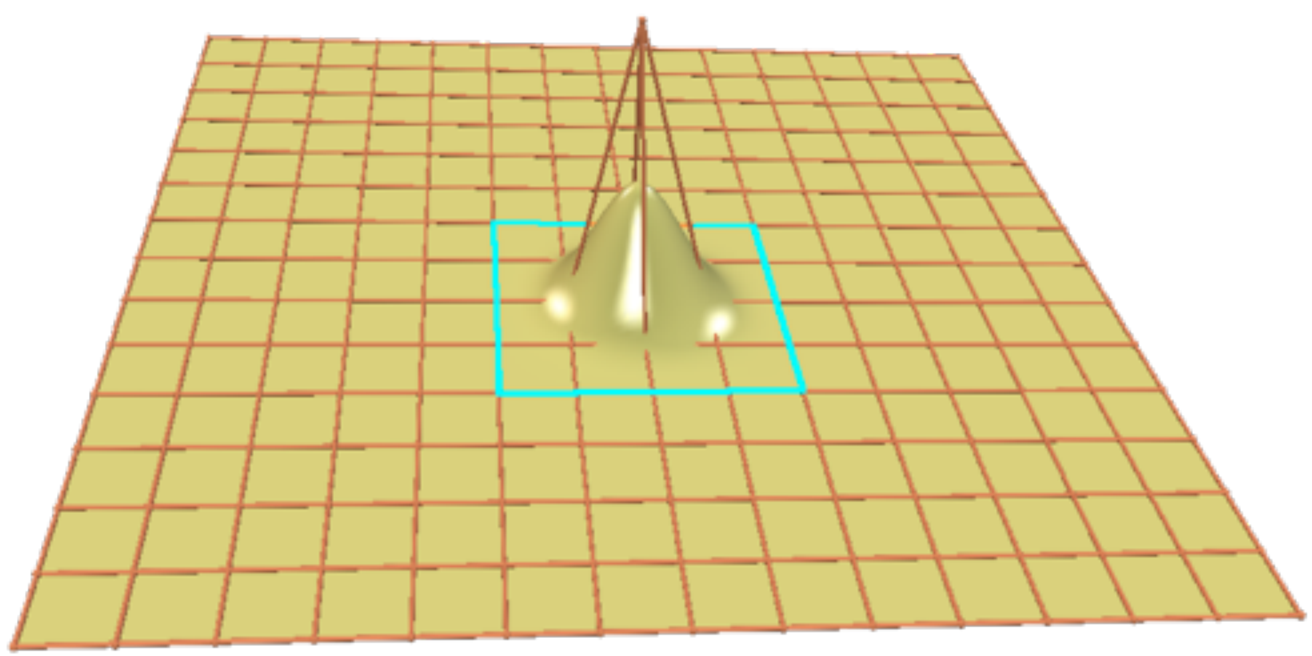


Modifications

- parameteric
  - control vertices
  - free-form deformation
  - boundary constraint modeling

$$\mathbf{f}(u,v) = \sum_{i=0}^{n}\sum_{j=0}^{m} \mathbf{c}_{ij} N_i^n(u) N_j^m(v)$$

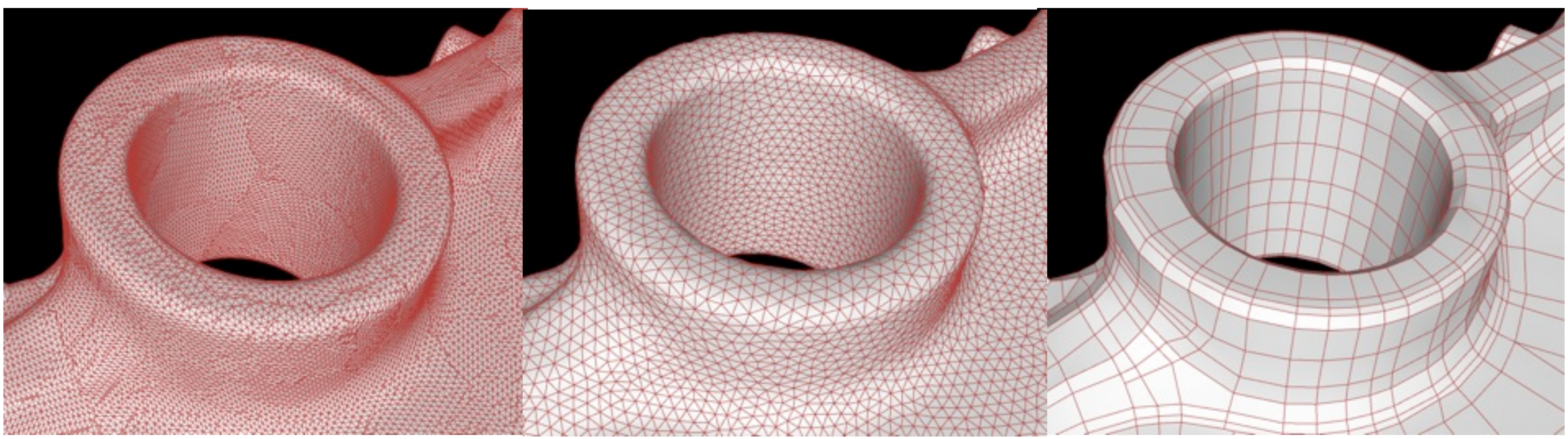




Modifications

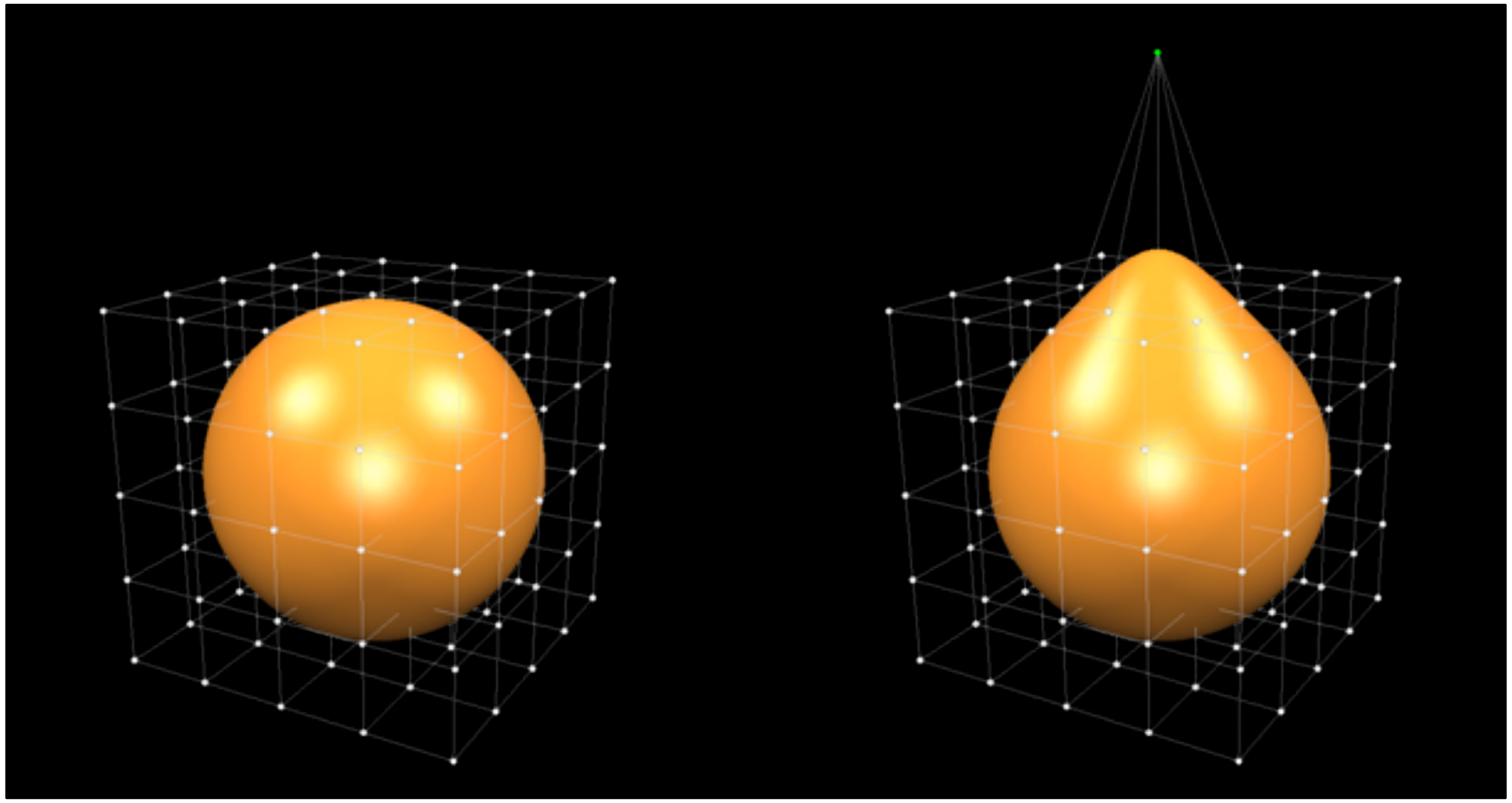
- parameteric
  - control vertices
  - free-form deformation
  - boundary constraint modeling

→ Remeshing
Pierre



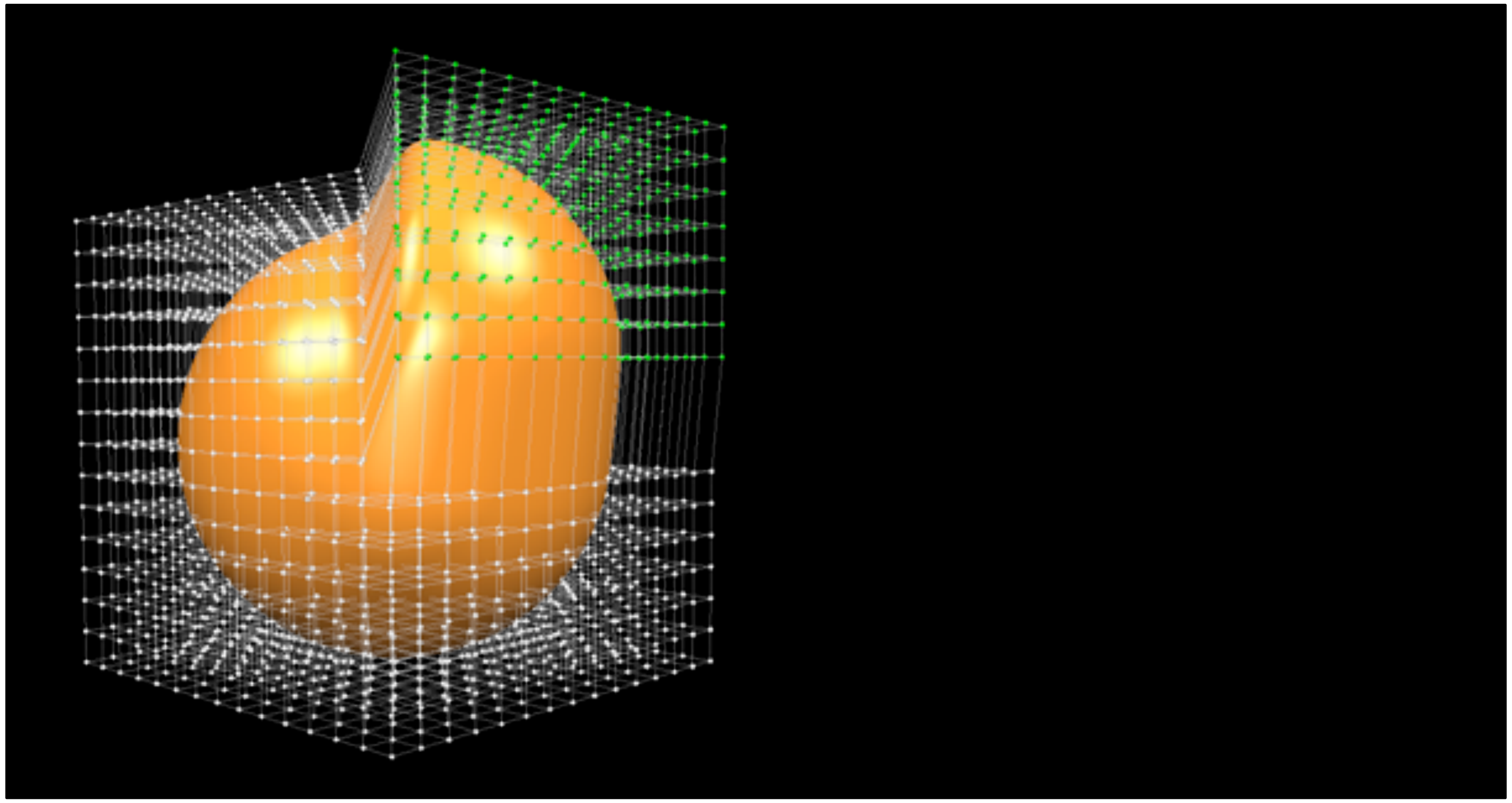
Modifications

- parameteric
  - control vertices
  - free-form deformation
  - boundary constraint modeling



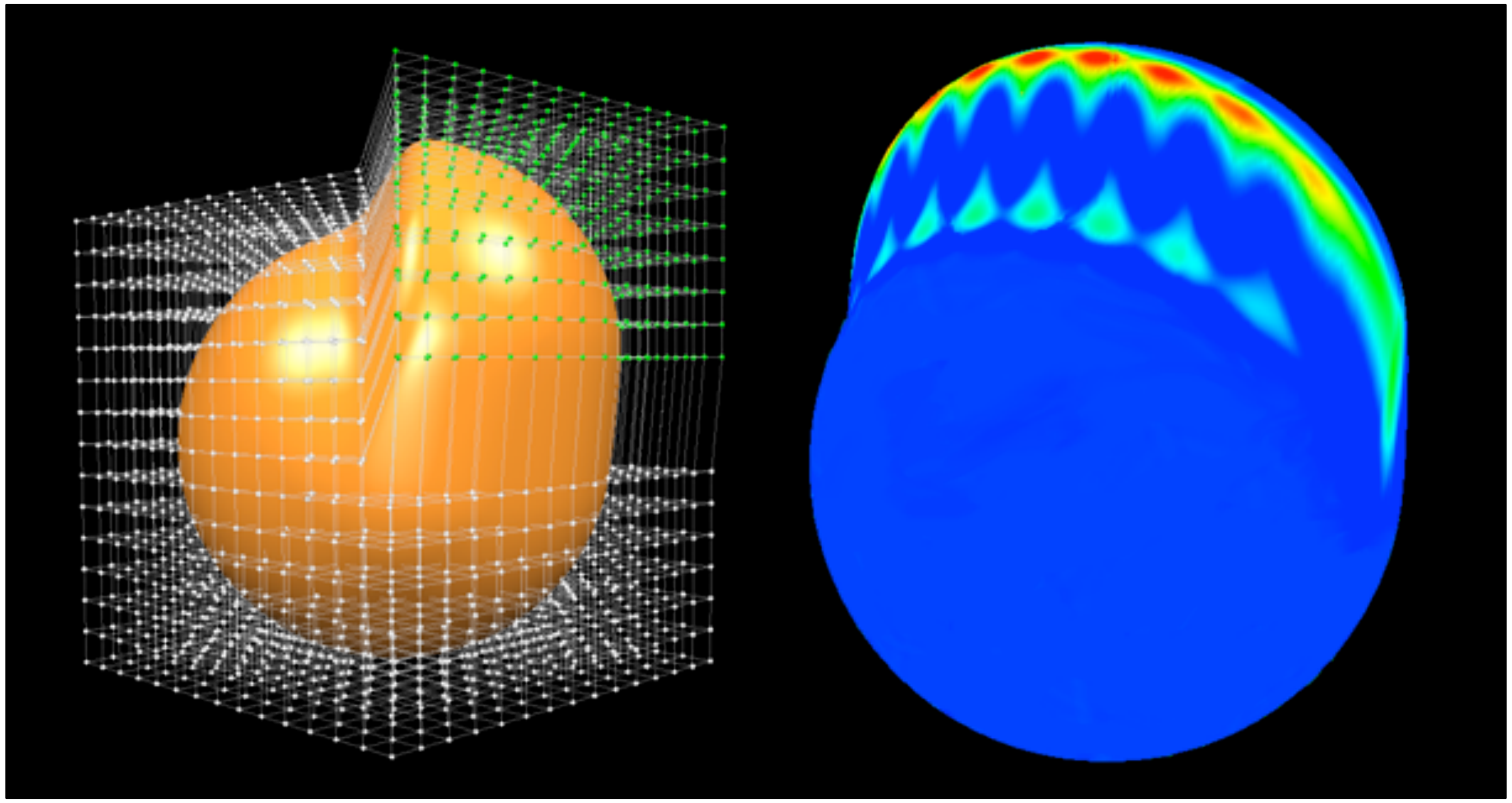
Modifications

- parameteric
  - control vertices
  - free-form deformation
  - boundary constraint modeling



Modifications

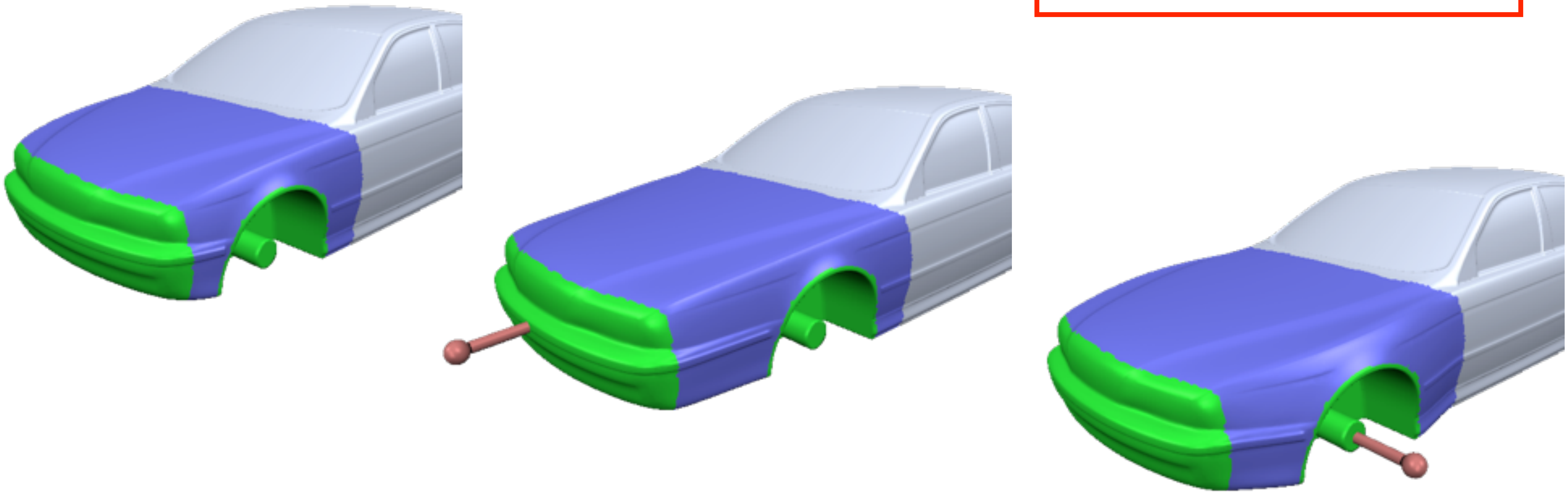
- parameteric
  - control vertices
  - free-form deformation
  - boundary constraint modeling



Modifications

- parameteric
  - control vertices
  - free-form deformation
  - boundary constraint modeling

→ Mesh Editing
Mario



Outline

- (mathematical) geometry representations
  - parametric vs. implicit
- approximation properties
- types of operations
  - distance queries
  - evaluation
  - modification / deformation
- data structures

Mesh Data Structures

- how to store geometry & connectivity?
- compact storage
  - file formats
- efficient algorithms on meshes
  - identify time-critical operations
  - all vertices/edges of a face
  - all incident vertices/edges/faces of a vertex

Face Set (STL)

- face:
  - 3 positions

Triangles		
x_{11} y_{11} z_{11}	x_{12} y_{12} z_{12}	x_{13} y_{13} z_{13}
x_{21} y_{21} z_{21}	x_{22} y_{22} z_{22}	x_{23} y_{23} z_{23}
...	...	...
x_{F1} y_{F1} z_{F1}	x_{F2} y_{F2} z_{F2}	x_{F3} y_{F3} z_{F3}

36 B/f = 72 B/v

no connectivity!

Shared Vertex (OBJ, OFF)

- vertex:
  - position
- face:
  - vertex indices

Vertices
x_1 y_1 z_1
...
x_V y_V z_V

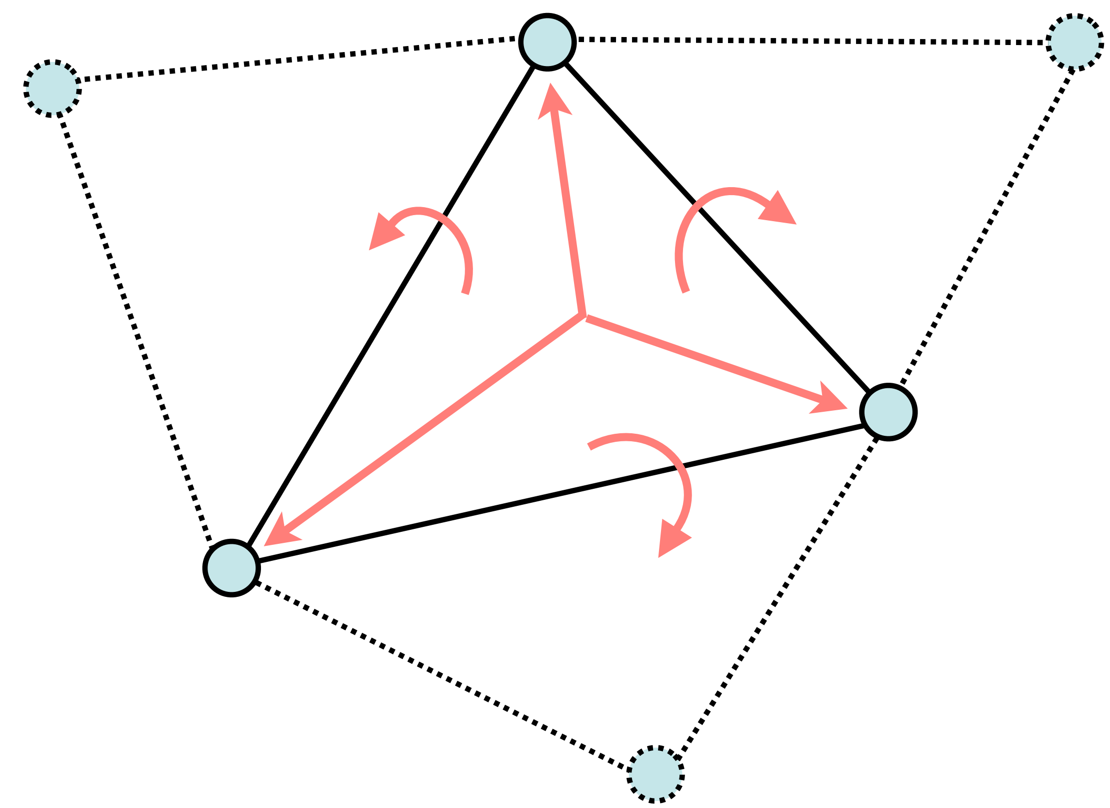
Triangles
v_{11} v_{12} v_{13}
...
...
...
...
v_{F1} v_{F2} v_{F3}

12 B/v + 12 B/f = 36 B/v

no neighborhood info

Face-Based Connectivity

- vertex:
  - position
  - 1 face
- face:
  - 3 vertices
  - 3 face neighbors

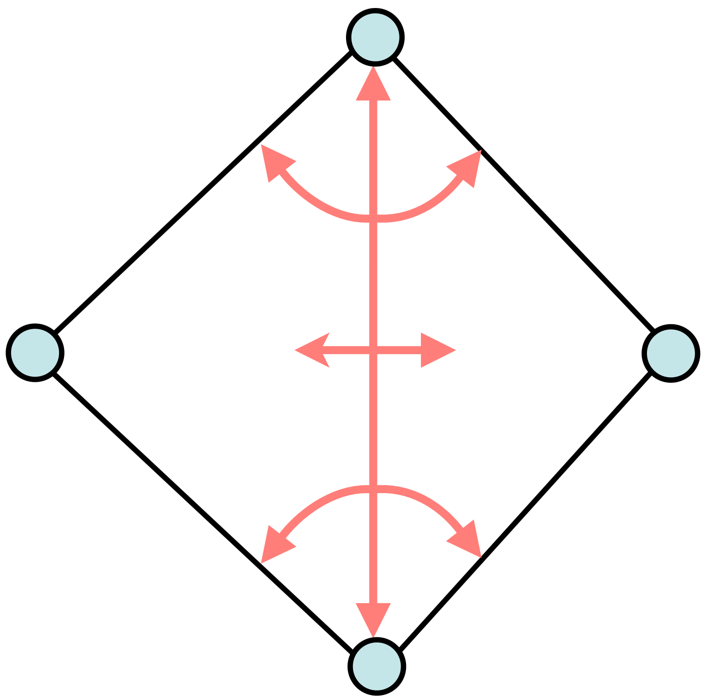


64 B/v

no edges!

Edge-Based Connectivity

- vertex
  - position
  - 1 edge
- edge
  - 2 vertices
  - 2 faces
  - 4 edges
- face
  - 1 edge

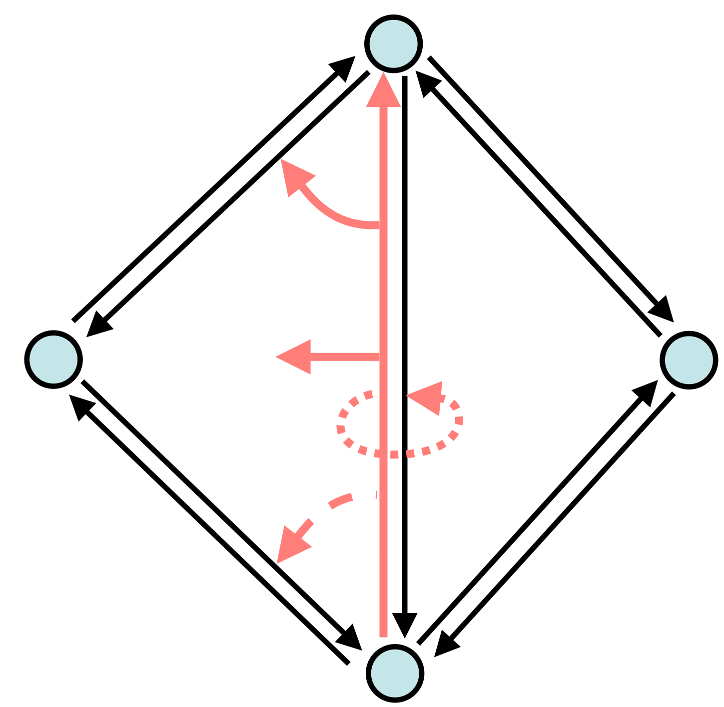


120 B/v

edge orientation?

Halfedge-Based Connectivity

- vertex
  - position
  - 1 halfedge
- halfedge
  - 1 vertex
  - 1 face
  - 1, 2, or 3 halfedges
- face
  - 1 halfedge

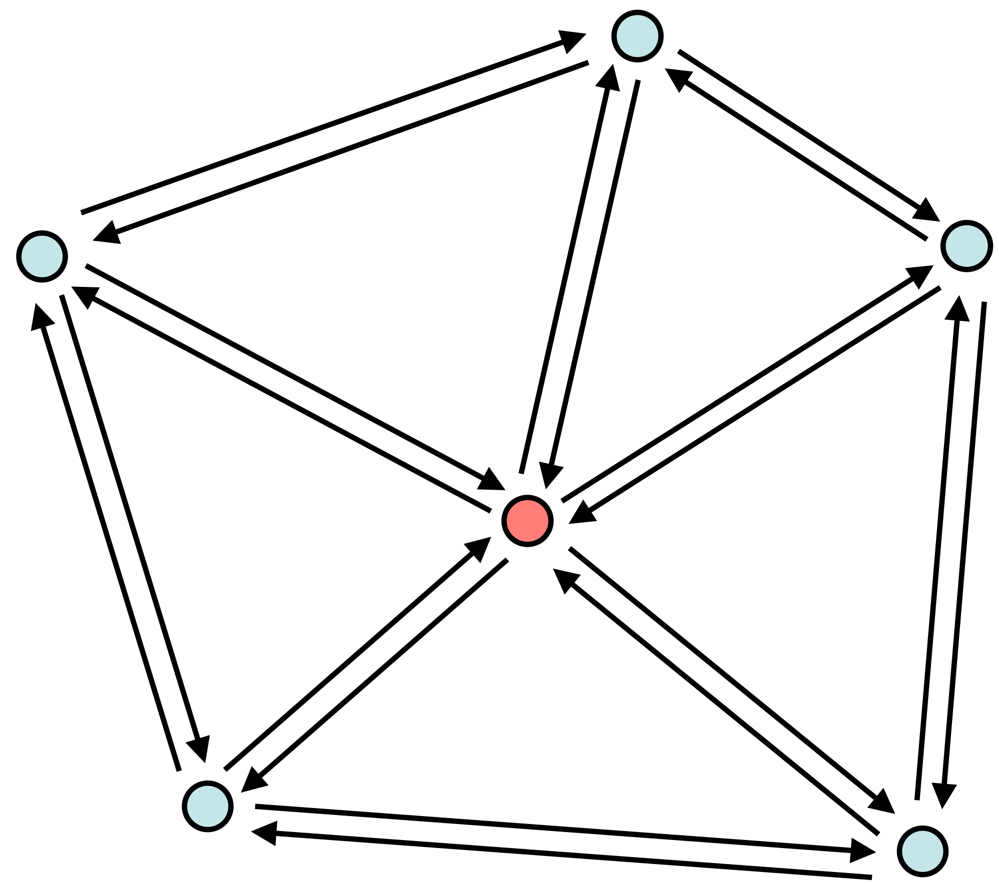


96 to 144 B/v

no case distinctions during traversal

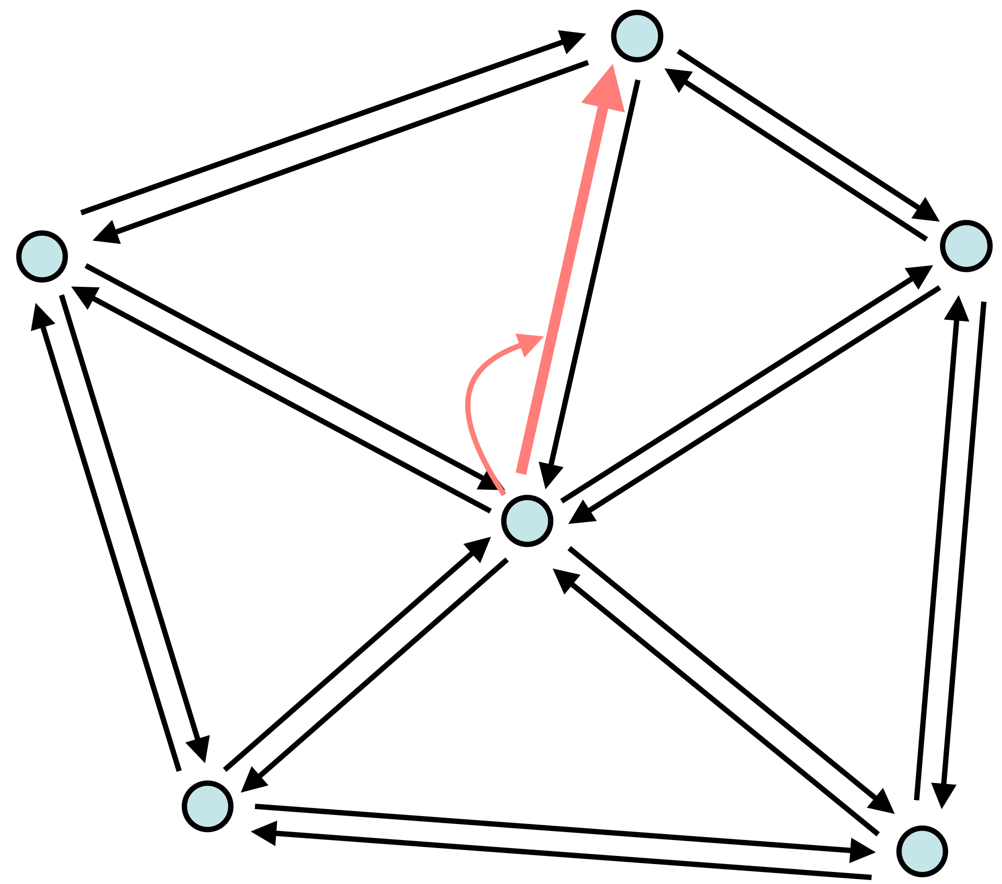
One-Ring Traversal

1. Start at vertex



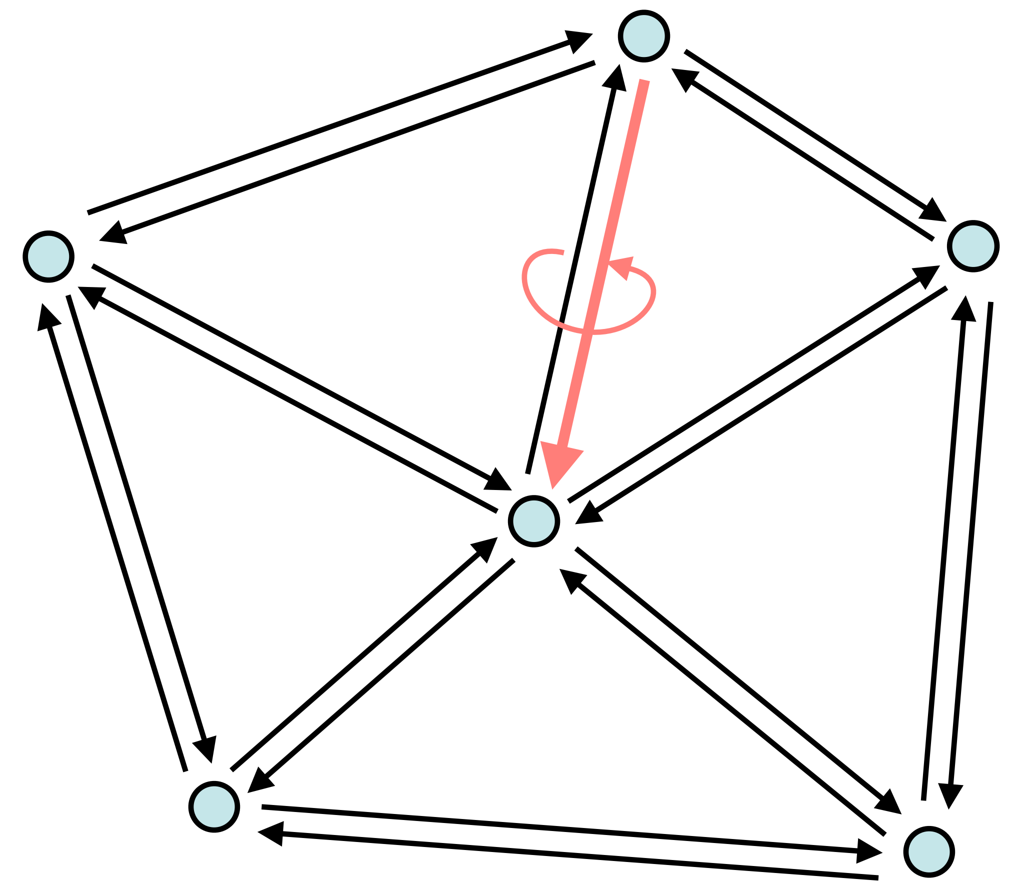
One-Ring Traversal

1. Start at vertex
2. Outgoing halfedge



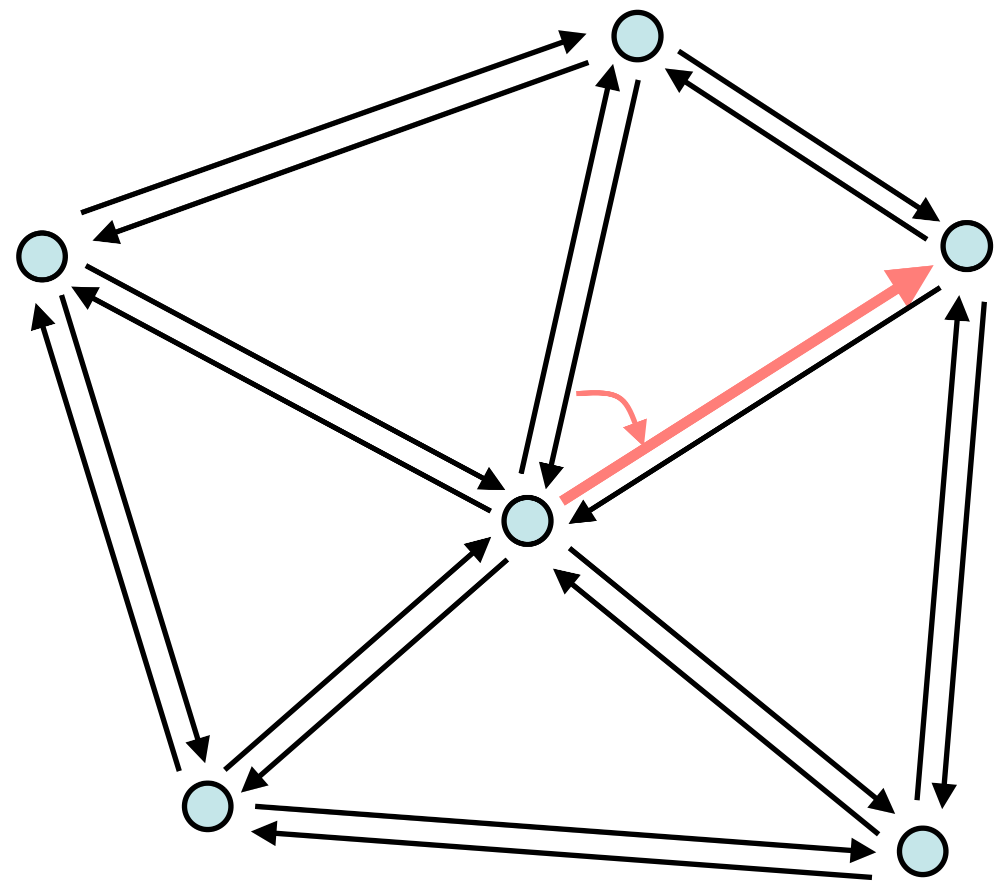
One-Ring Traversal

1. Start at vertex
2. Outgoing halfedge
3. Opposite halfedge



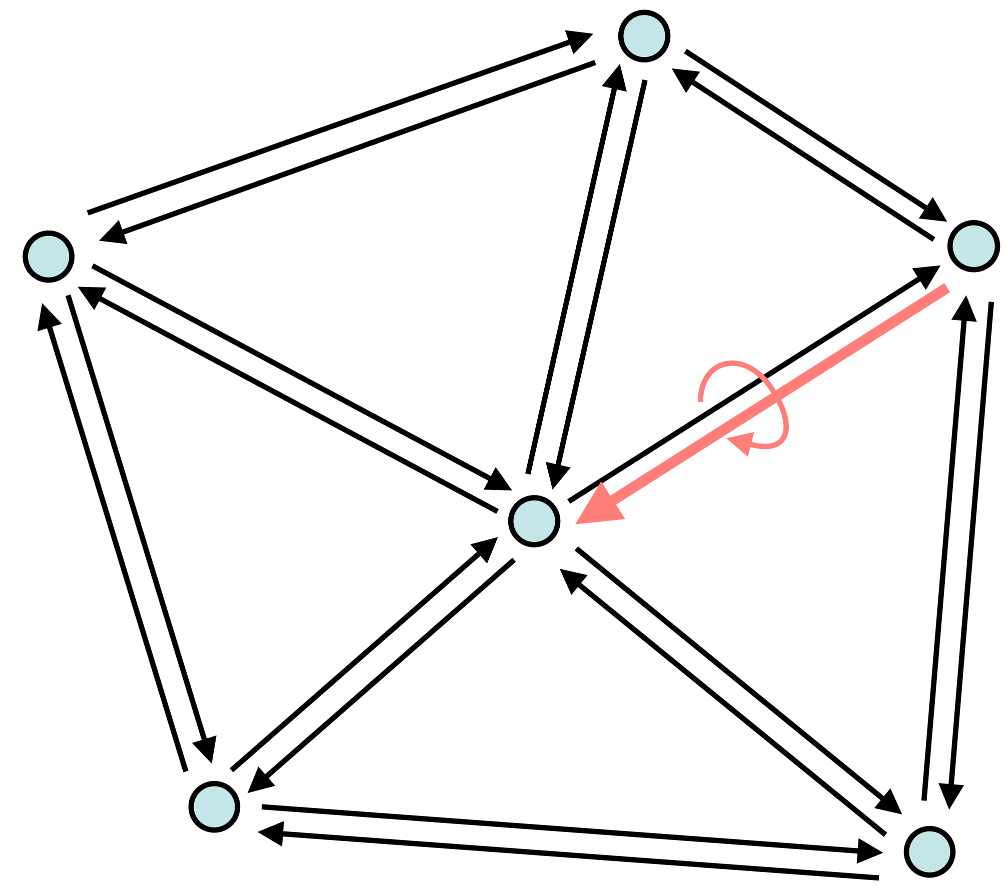
One-Ring Traversal

1. Start at vertex
2. Outgoing halfedge
3. Opposite halfedge
4. Next halfedge



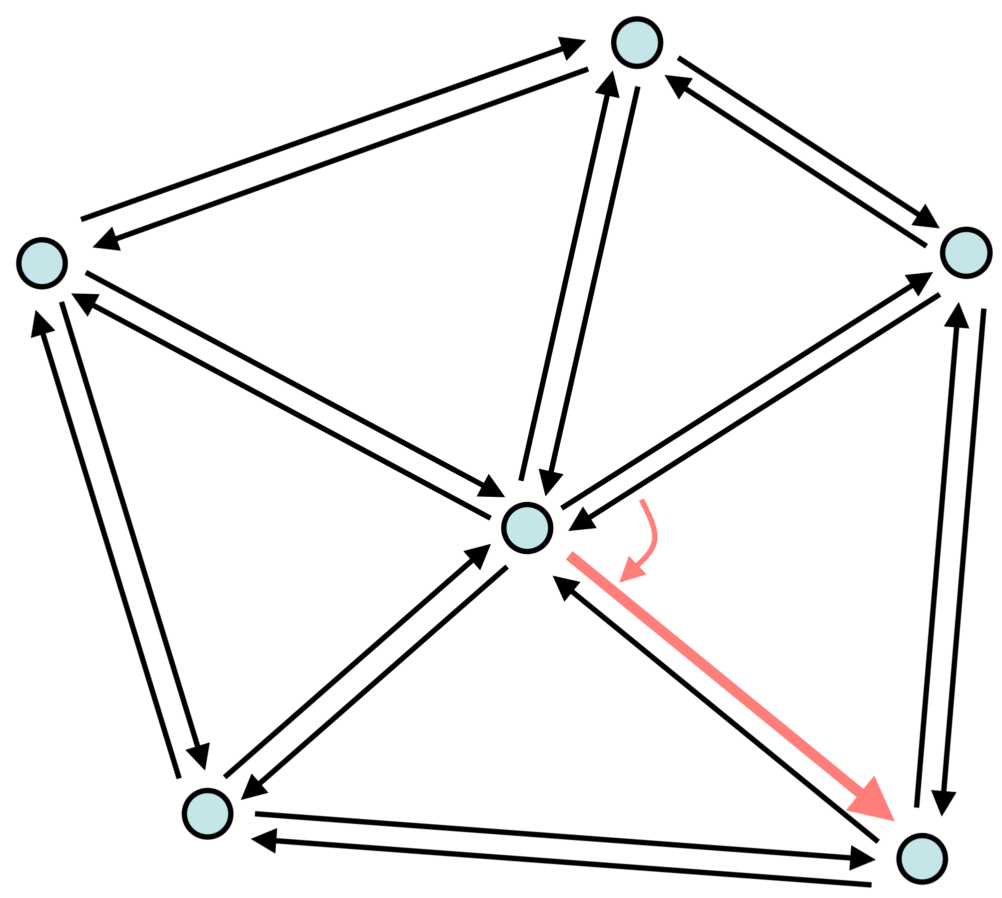
One-Ring Traversal

1. Start at vertex
2. Outgoing halfedge
3. Opposite halfedge
4. Next halfedge
5. Opposite



One-Ring Traversal

1. Start at vertex
2. Outgoing halfedge
3. Opposite halfedge
4. Next halfedge
5. Opposite
6. Next
7. ...



Halfedge-Based Libraries

- CGAL
  - `www.cgal.org`
  - computational geometry
  - free for non-commercial use

- OpenMesh
  - `www.openmesh.org`
  - mesh processing
  - free, LGPL licence

Literature

- Kettner, *Using generic programming for designing a data structure for polyhedral surfaces*, Symp. on Comp. Geom., 1998
- Campagna et al, *Directed Edges - A Scalable Representation for Triangle Meshes*, Journal of Graphics Tools 4(3), 1998
- Botsch et al, *OpenMesh - A generic and efficient polygon mesh data structure*, OpenSG Symp. 2002

Outline

- (mathematical) geometry representations
  - parametric vs. implicit
- approximation properties
- types of operations
  - distance queries
  - evaluation
  - modification / deformation
- data structures