Mesh Repairing and Simplification

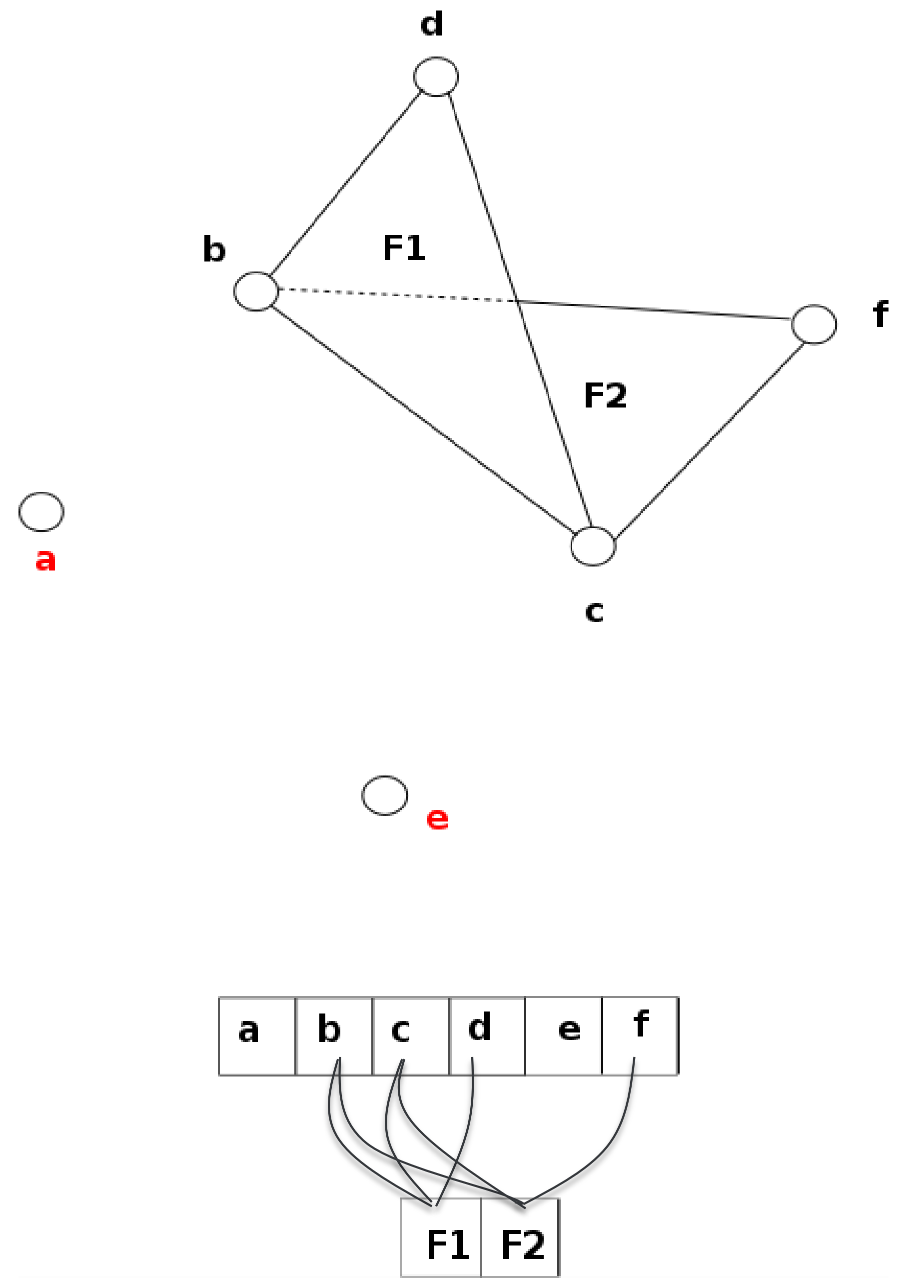
Gianpaolo Palma

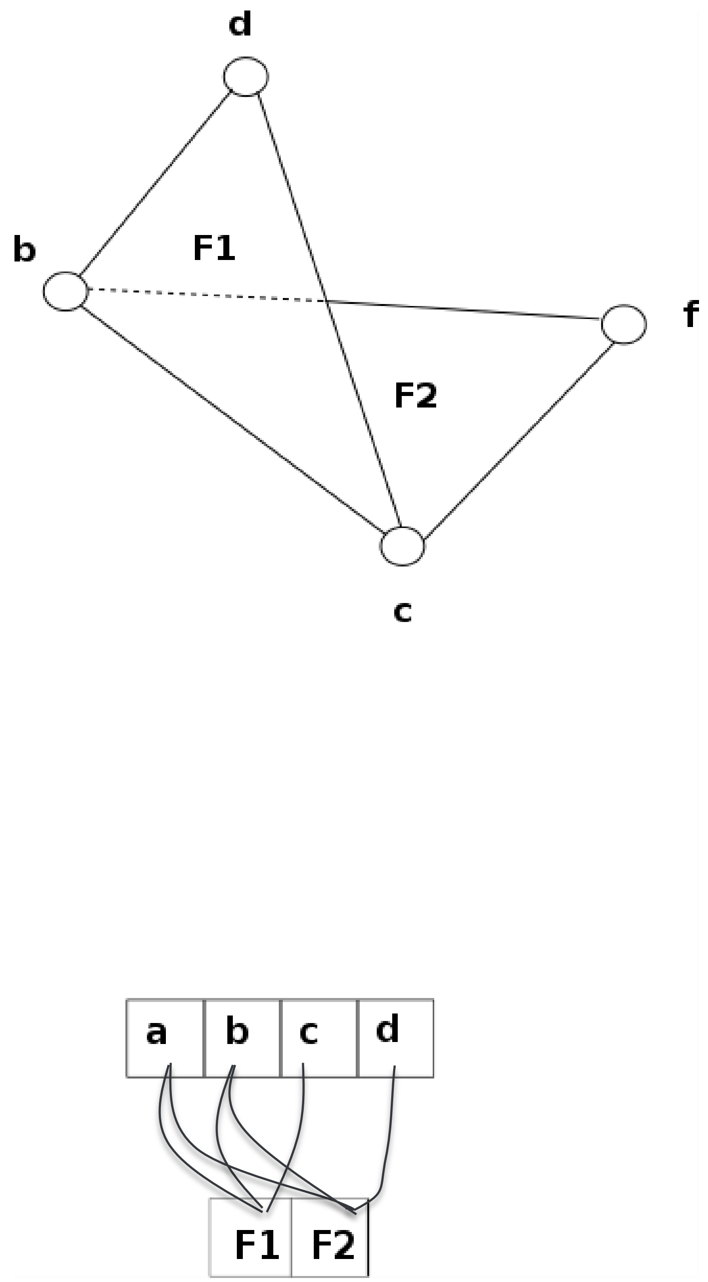
Mesh Repairing

- Removal of artifacts from geometric model such that it becomes suitable for further processing
  - Input: a generic 3D model
  - Output: (hopefully)a manifold and watertight model
- It doesn’t yet exist an algorithm able to handle all the kind of topological or geometric issue
- We see how to detect and correct these artifacts

Unreferenced Vertices

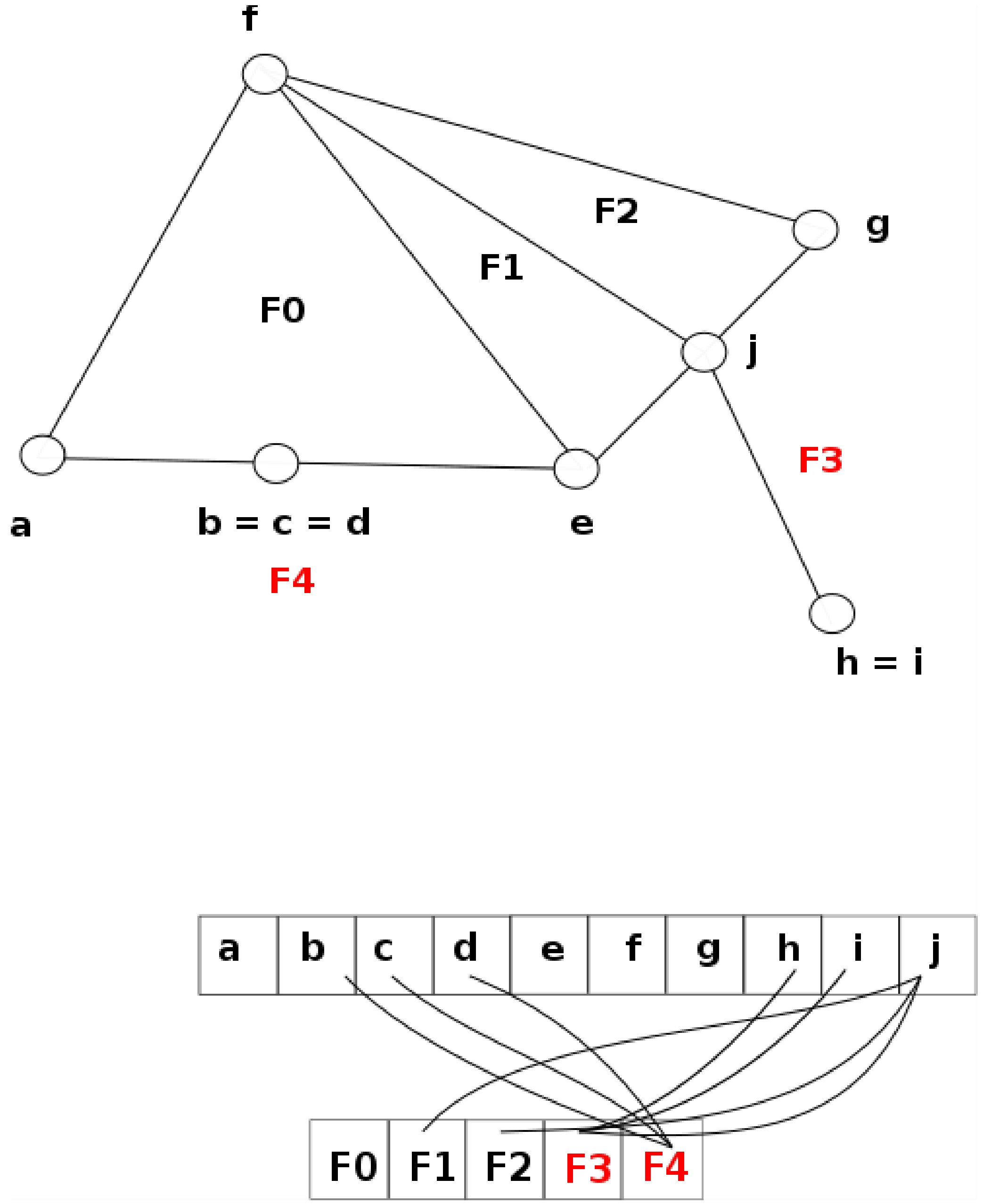
- Delete vertices not referred by any face





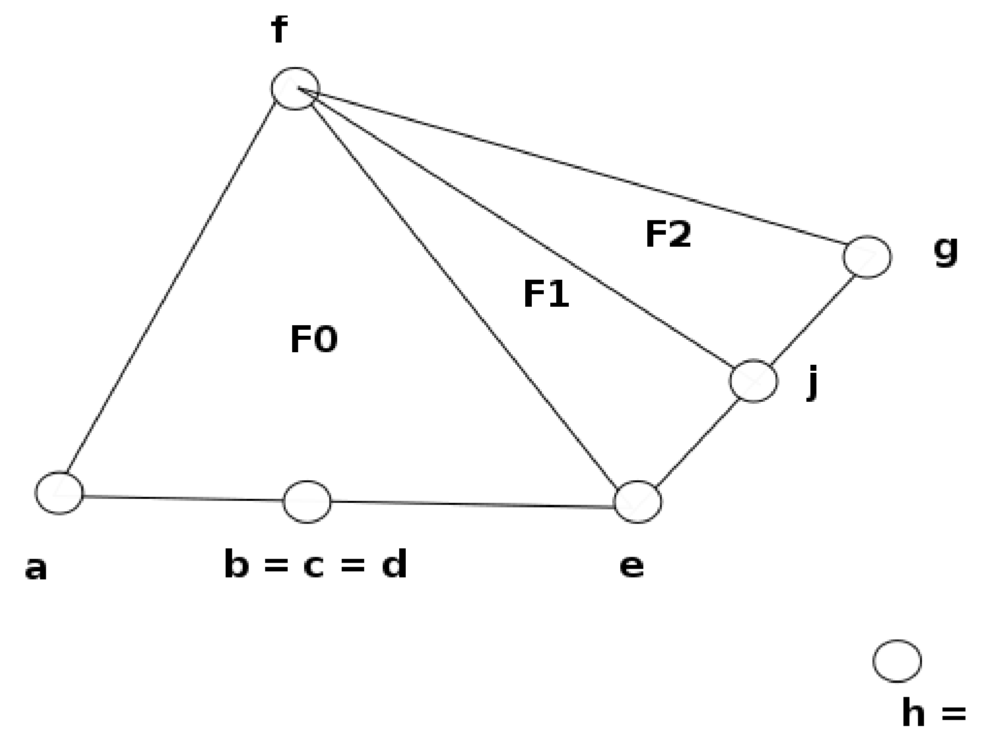
Zero Area Faces

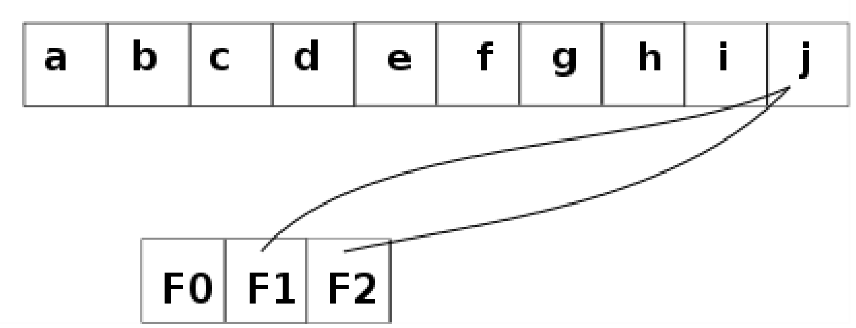
- Causes
  - Duplicate vertices (vertices with the same position)



Zero Area Faces

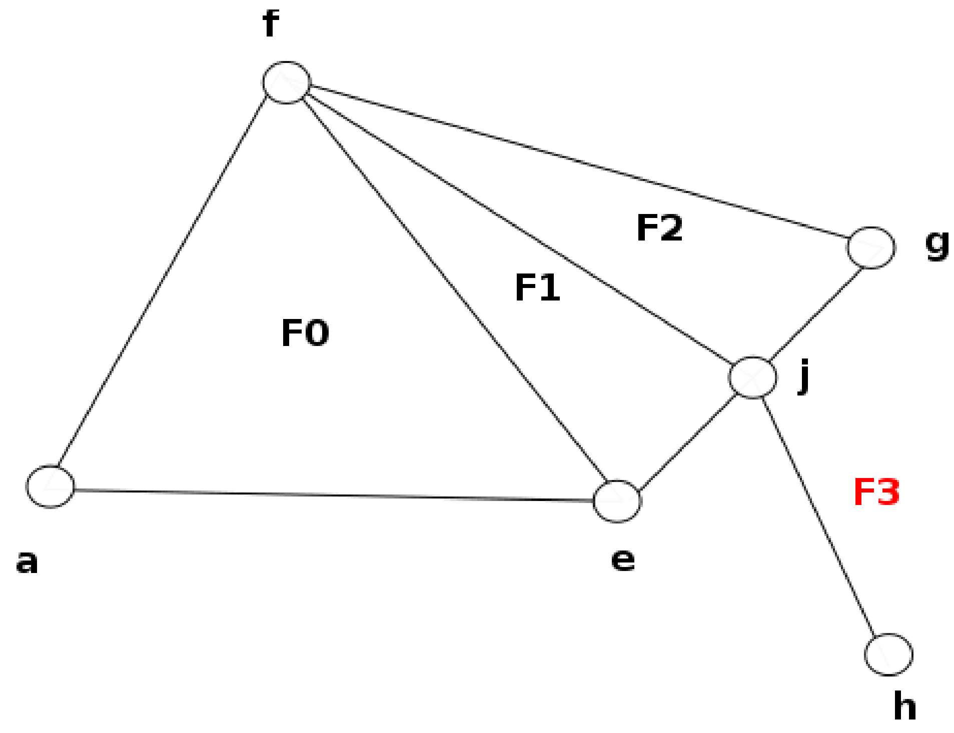
- Causes
  - Duplicate vertices (vertices with the same position)
- Compute the area of all the faces and remove the ones with zero area
- Side Effect
  - Unreferenced vertices

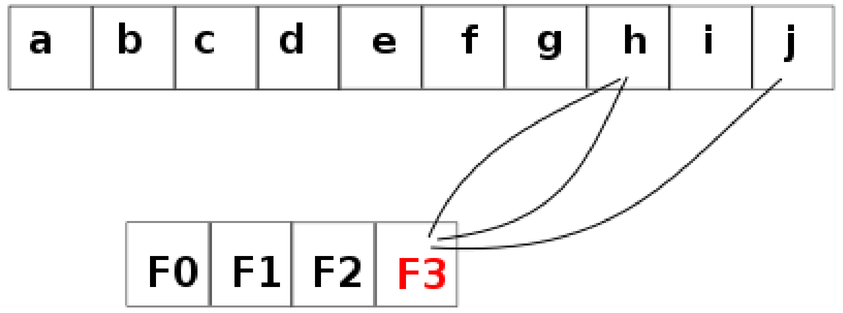




Degenerated Faces

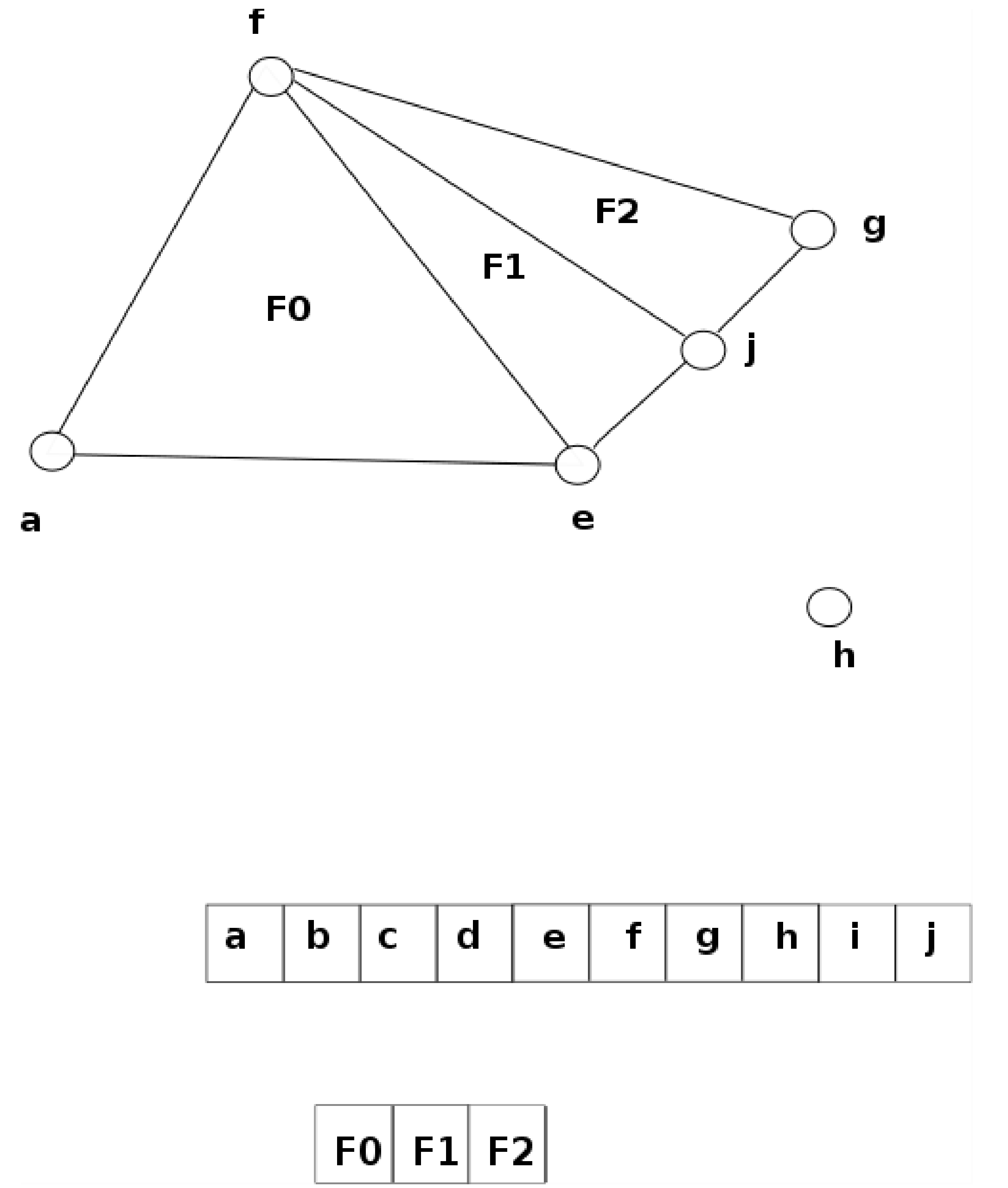
- Faces having at least two vertex pointers referring the same vertex





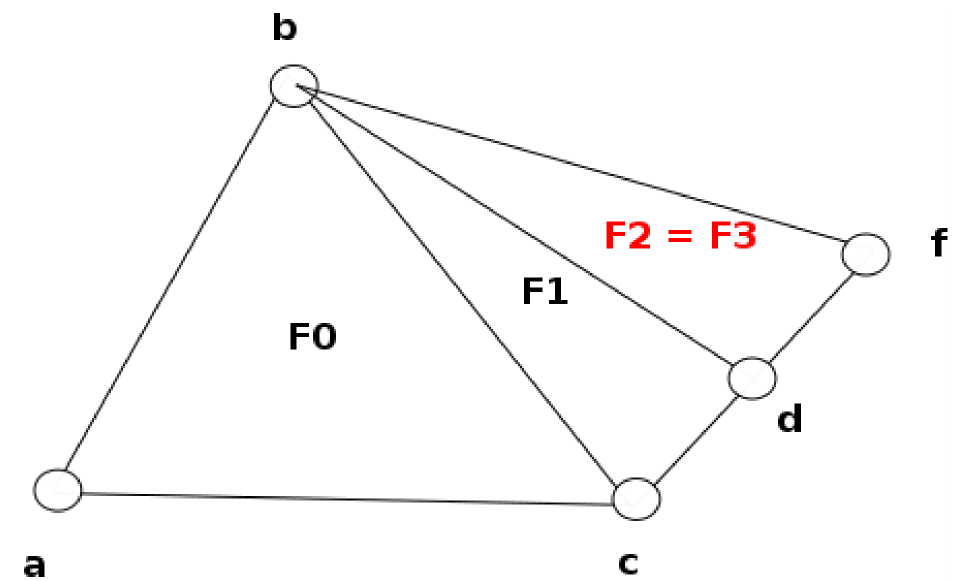
Degenerated Faces

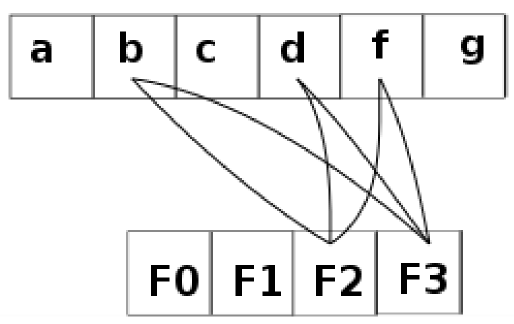
- Faces having at least two vertex pointers referring the same vertex
- Degenerated faces are zero area faces…but not all zero area faces are degenerated faces
- Side effect
  - Unreferenced vertices



Duplicated Faces

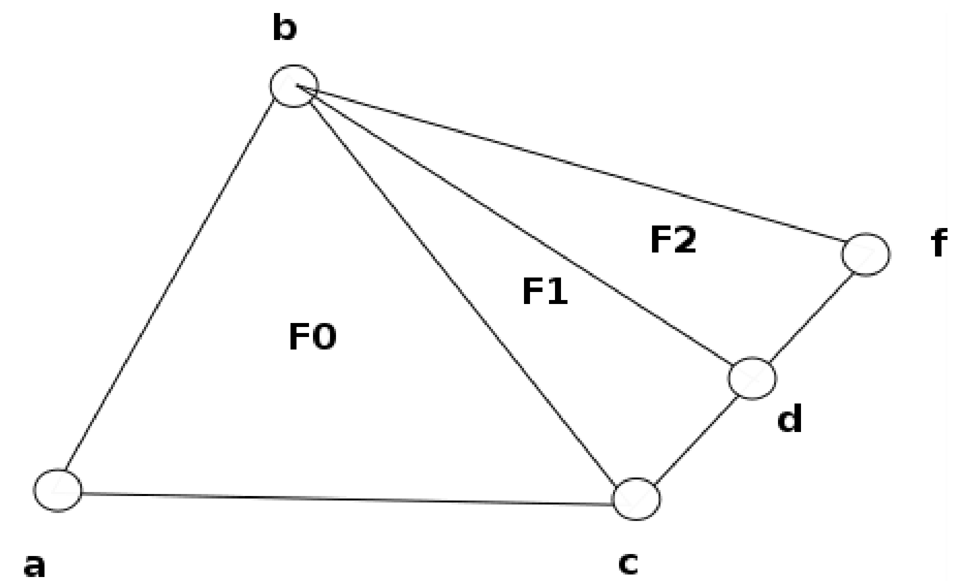
- Merge faces with the same vertex references

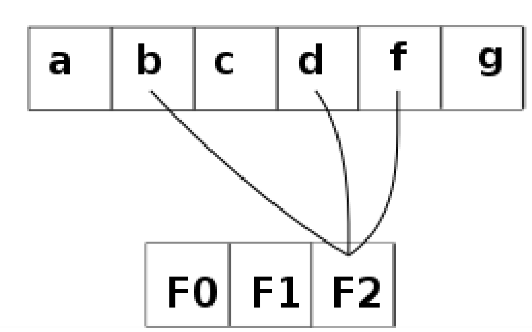




Duplicated Faces

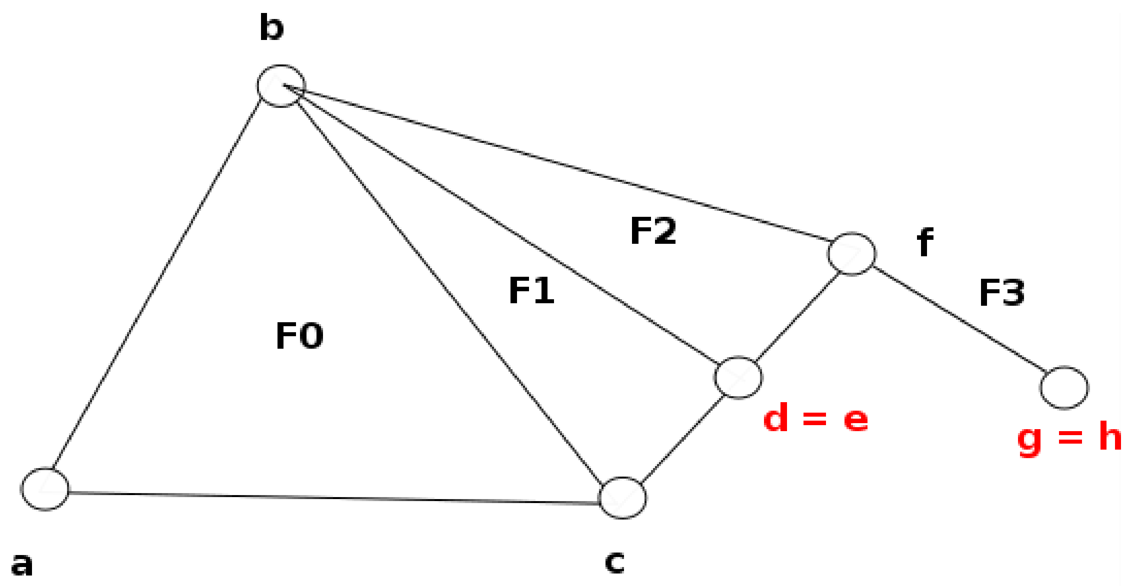
- Merge faces with the same vertex references

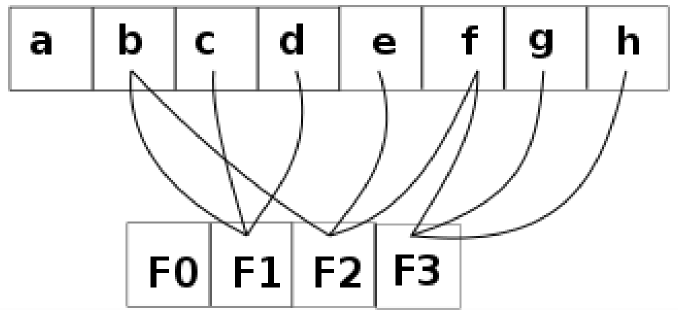




Duplicated Vertices

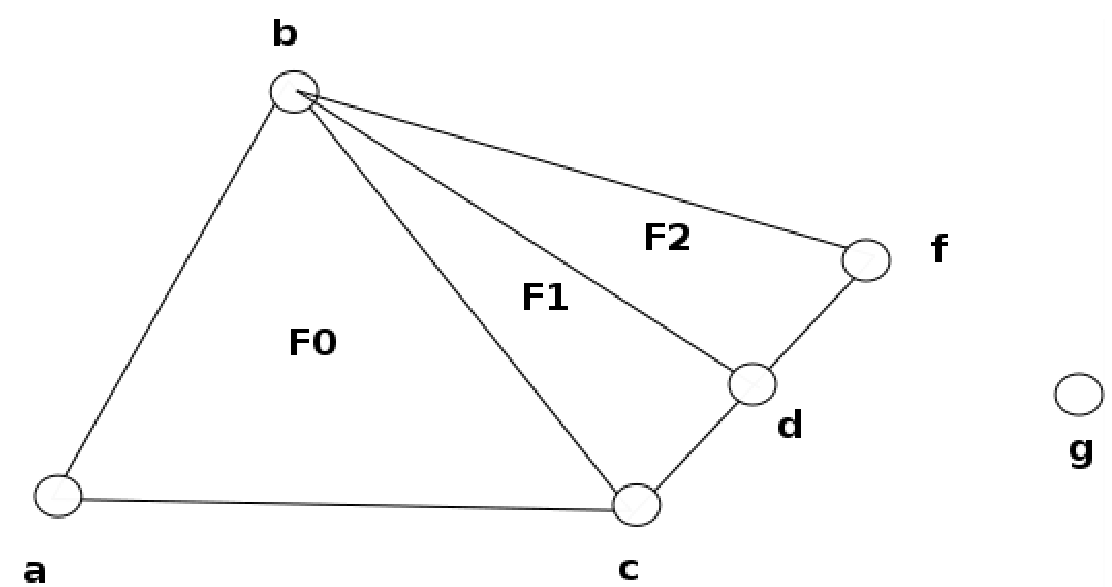
- Vertices with the same coordinates positions

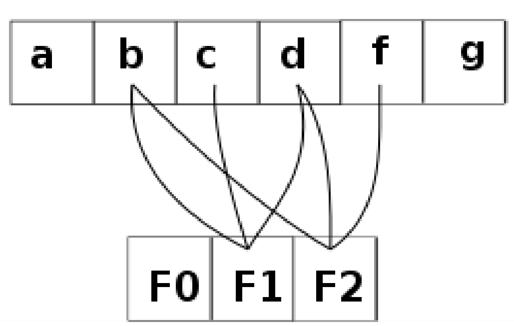




Duplicated Vertices

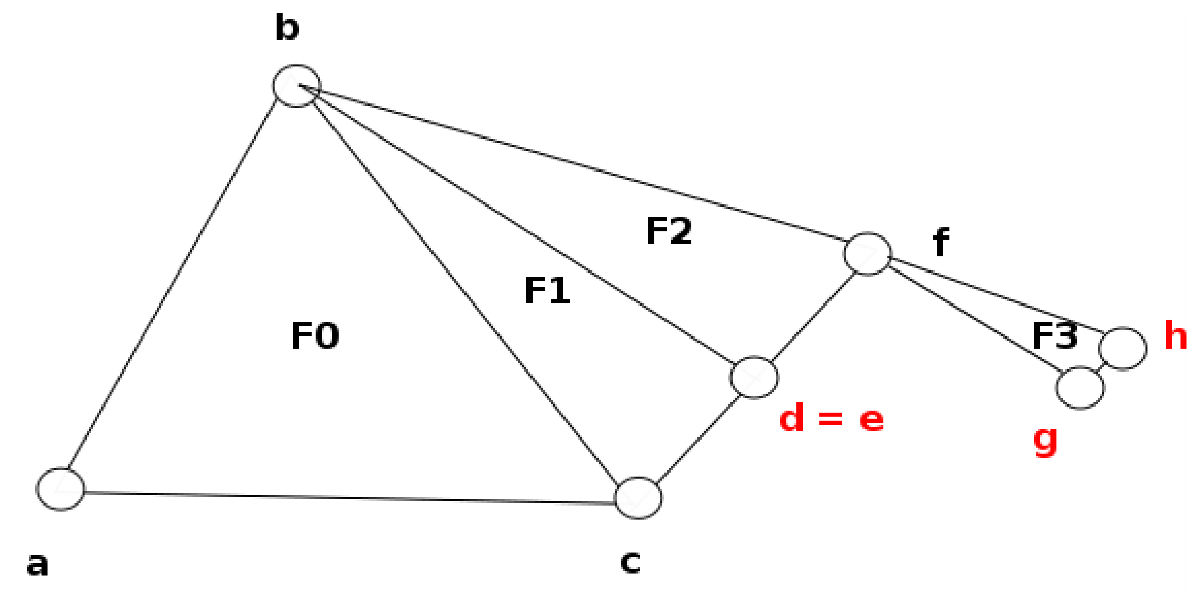
- Vertices with the same coordinates positions
- Merge the vertices and update the references in the incident face
- Side effect
    - Unreferenced vertices and degenerated faces

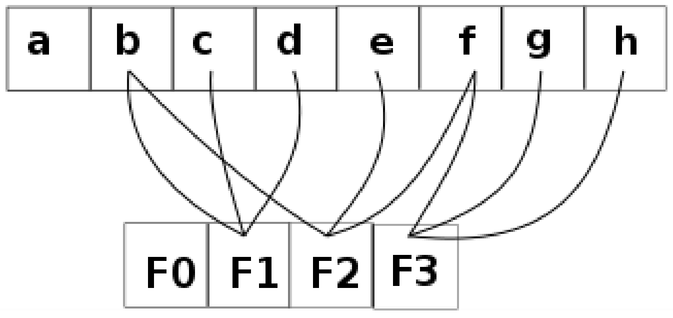




Close Vertices

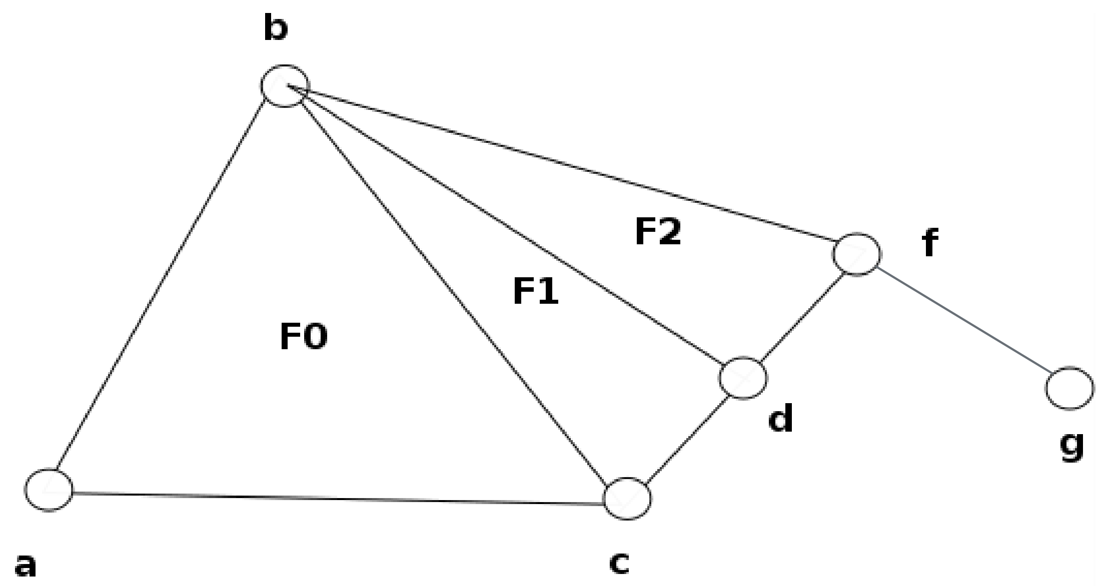
- Merge vertices faraway each other less than a threshold

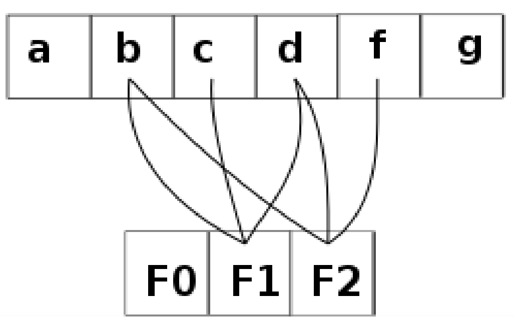




Close Vertices

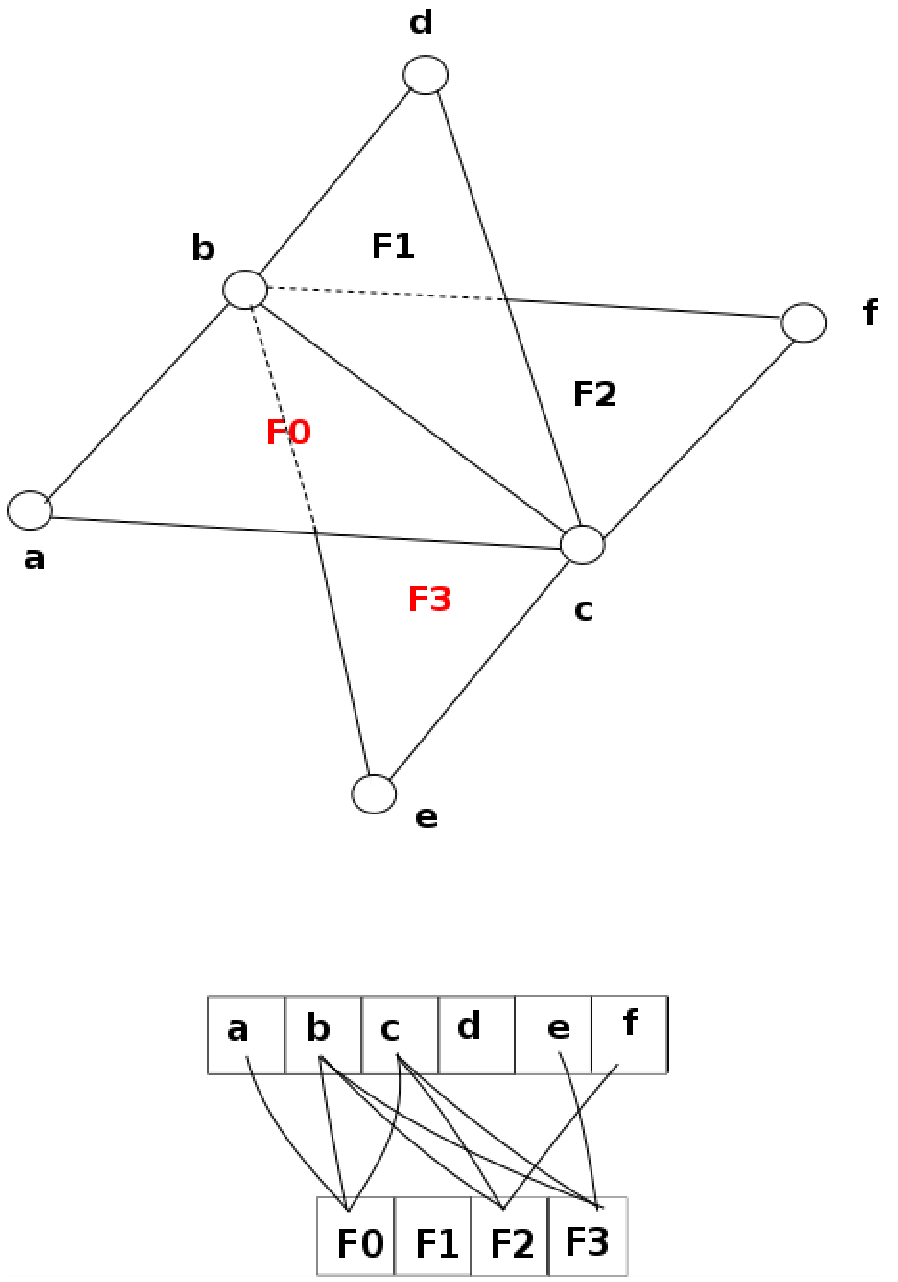
- Merge vertices faraway each other less than a threshold
- Side effect
  - Degenerated faces





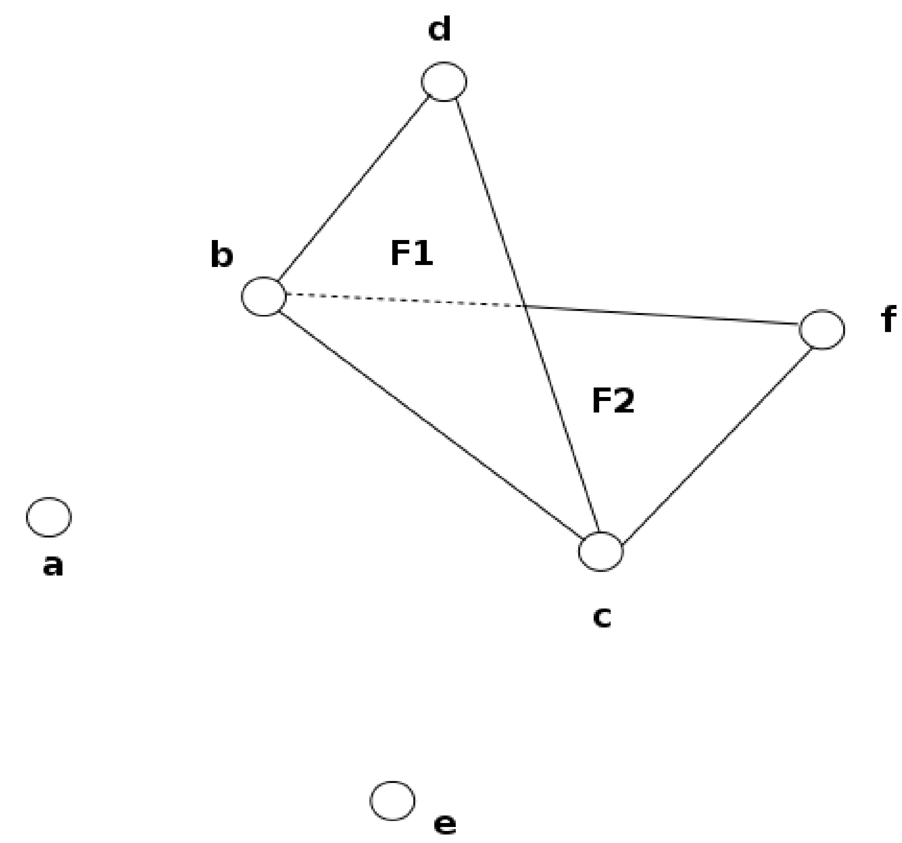
Non 2-manifold Edge

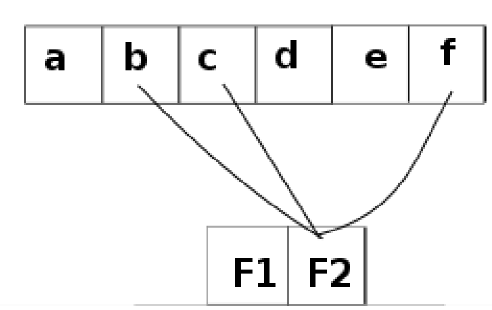
- Edge with more than two incident faces



Non 2-manifold Edge

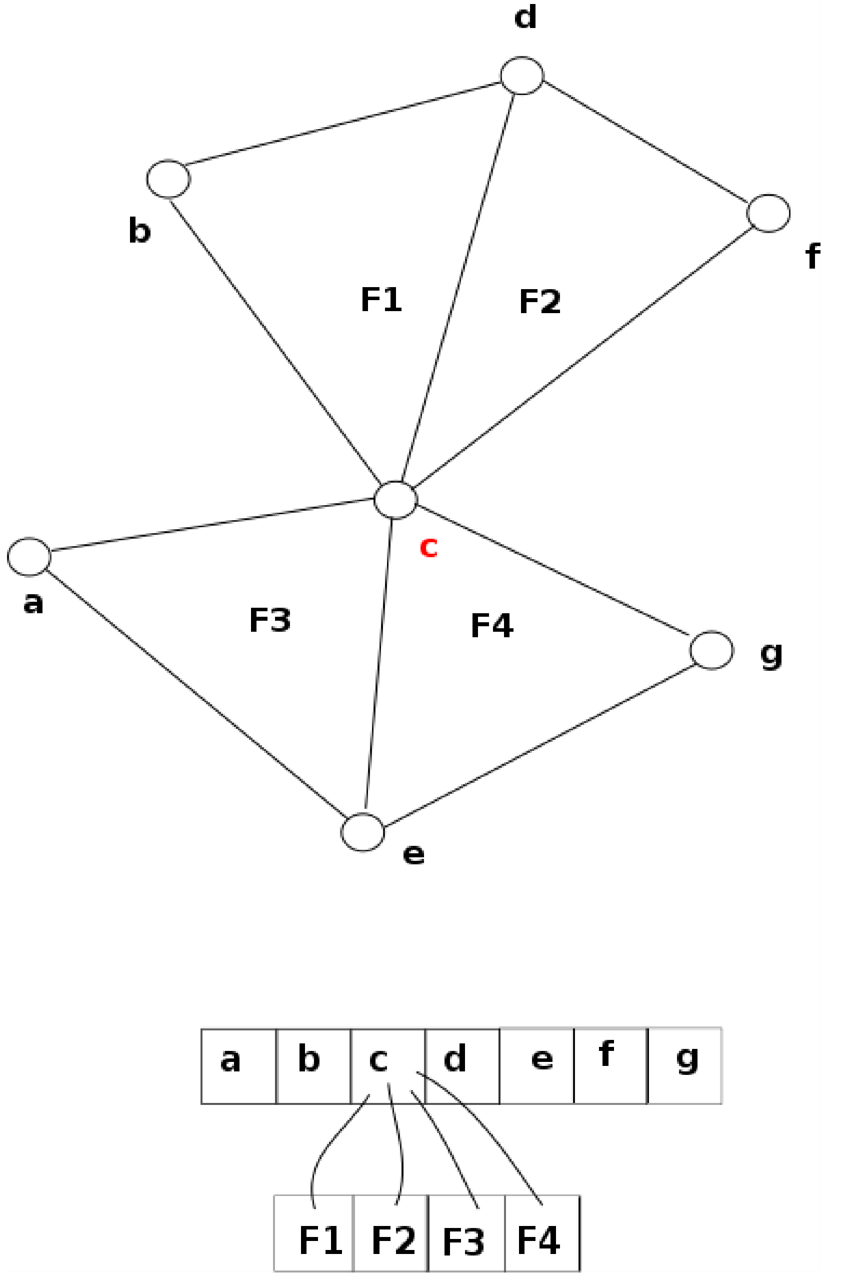
- Edge with more than two incident faces
- Delete face from non-2manifold edges until the edges have at most two faces incident on them (delete iteratively the faces with smaller area)
- Side effect
  - Unreferenced vertices





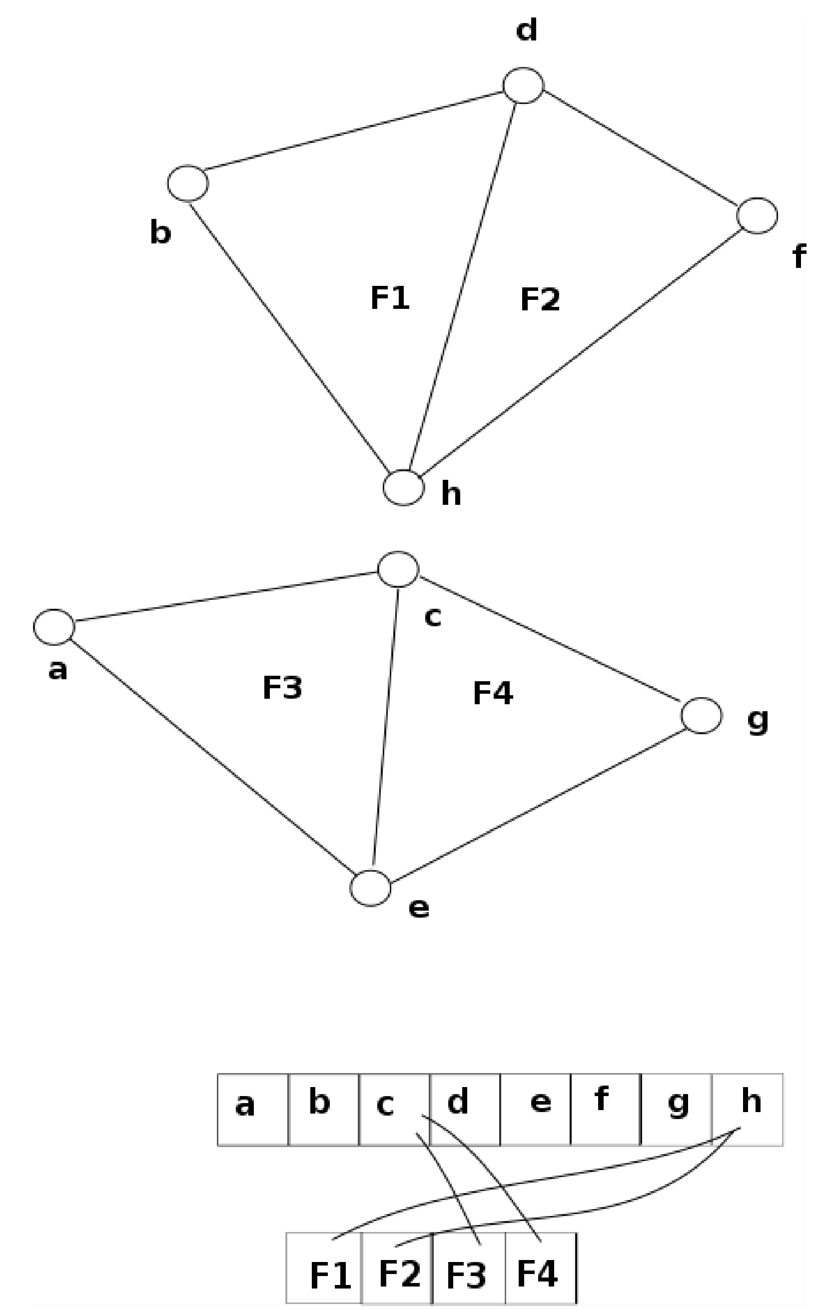
Non 2-manifold Vertex

- The vertex doesn't have a single complete loop of triangle around it using the FF adjacency



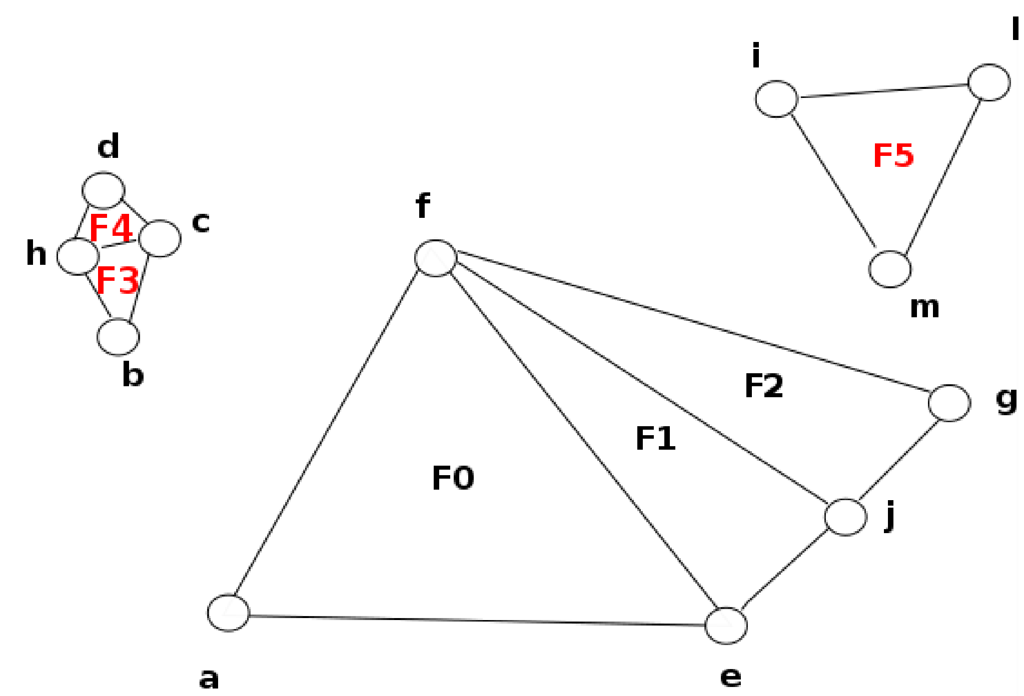
Non 2-manifold Vertex

- The vertex doesn't have a single complete loop of triangle around it using the FF adjacency
- Splits non-2manifold vertices and move them of threshold distance or delete vertex
- Side effect
    - Duplicated vertices



Isolated Pieces

- Clusters of isolated faces that cannot be reached with the navigation using the FF adjacency

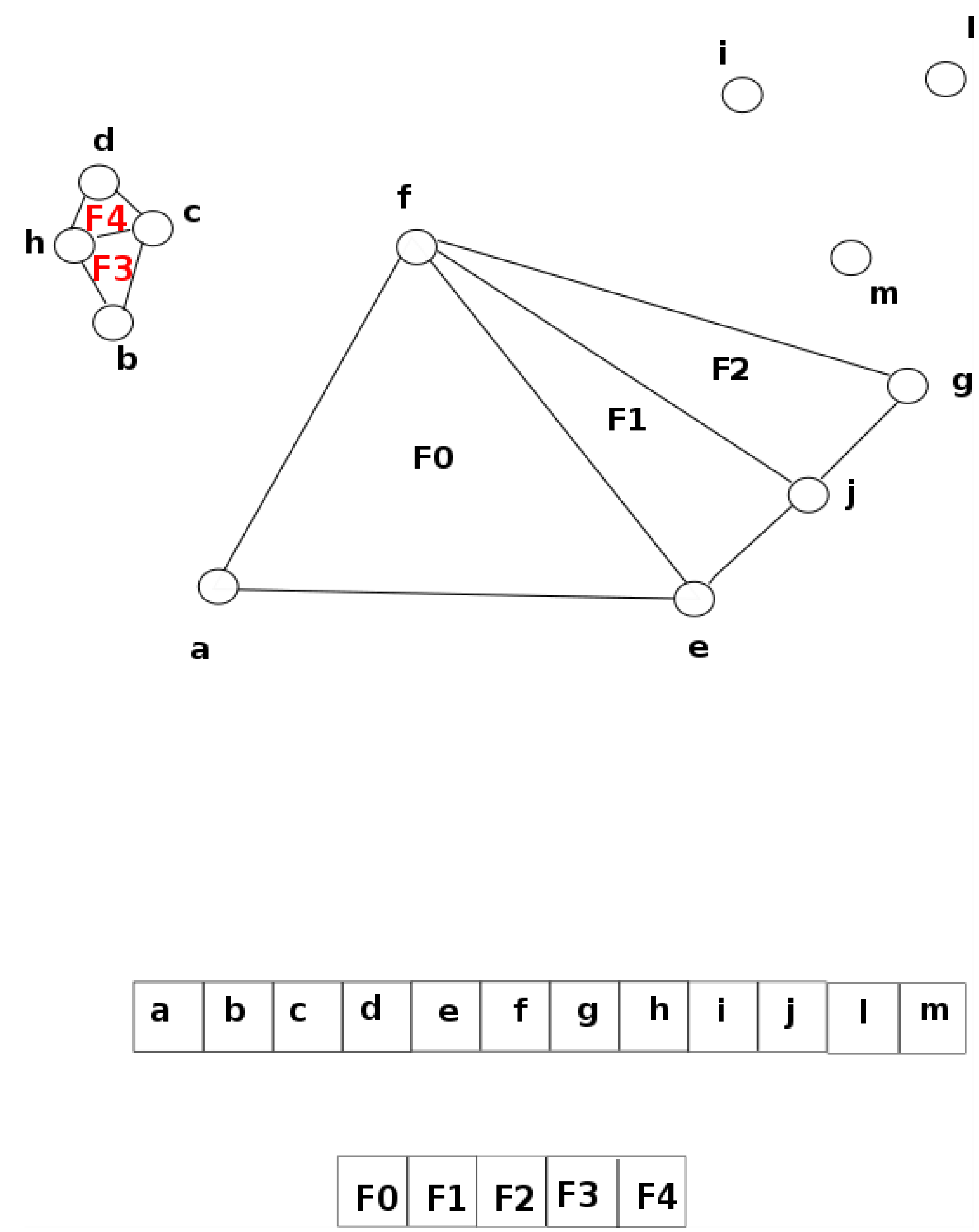


a	b	c	d	e	f	g	h	i	j	l	m

F0	F1	F2	F3	F4	F5

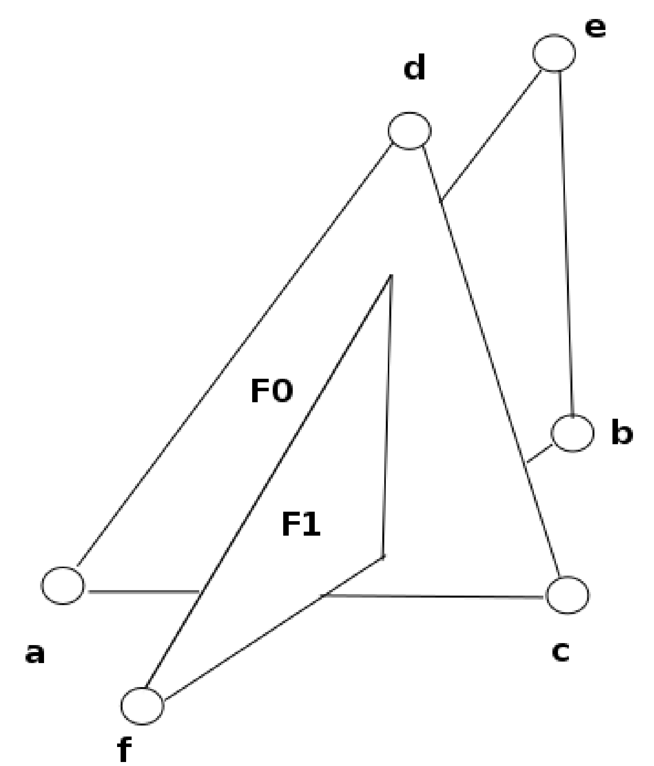
Isolated Pieces

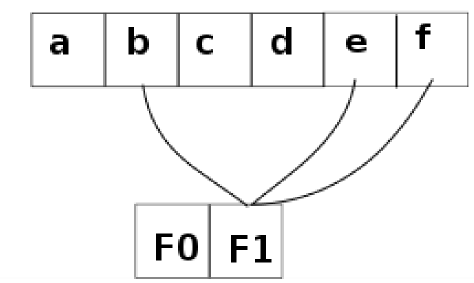
- Clusters of isolated faces that cannot be reached with the navigation using the FF adjacency
- Remove isolated connected components of the mesh composed by less than n faces or with a bbox diagonal less than a threshold
- Side effect
  - Unreferenced vertices



Self Intersecting Face

- Faces into the mesh intersecting with others
- Select the faces, delete them and eventually close the hole

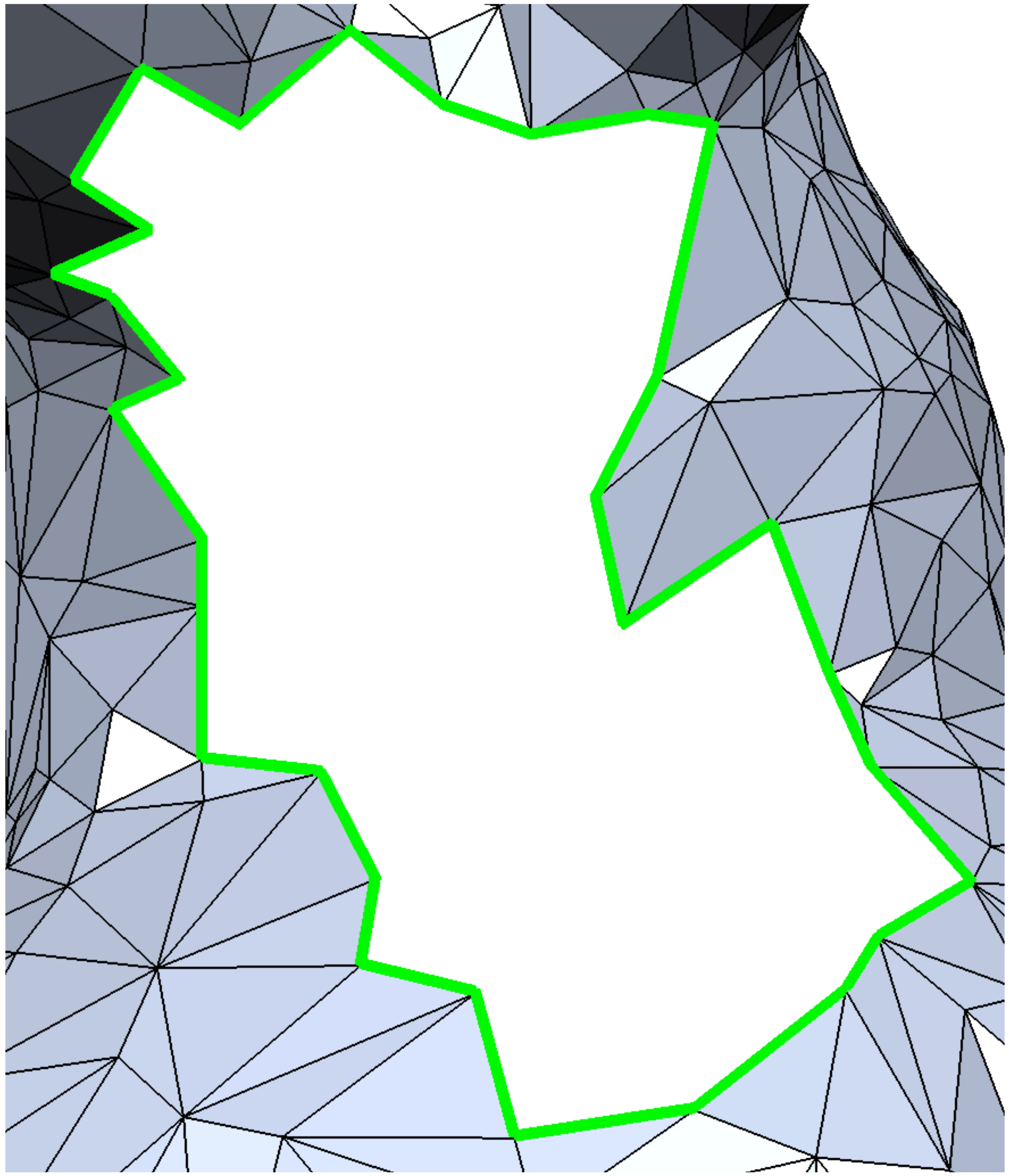




Filling Holes

- Missing data
  1. Detect holes border
     - Close loop of boundary edge
  2. Triangulate hole
  3. Mesh Refinement
  4. Mesh Fairing

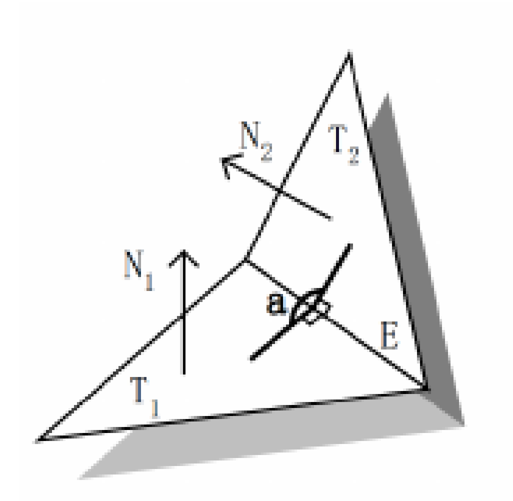
[Liepa, SGP 03]

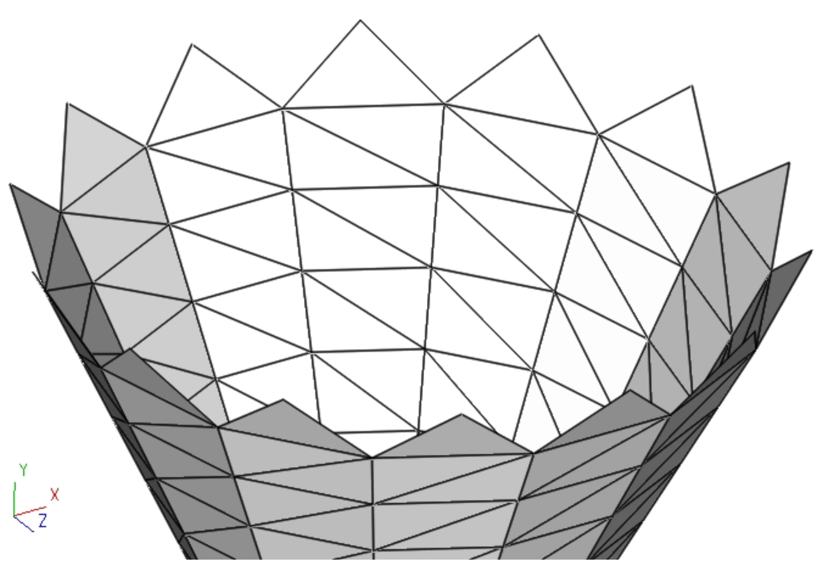


Filling Holes

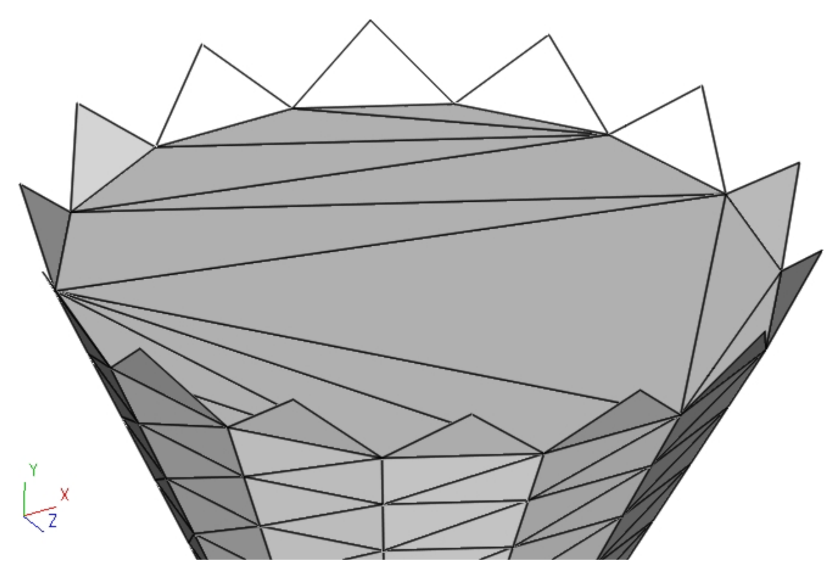
[Liepa, SGP 03]

- Triangulate Hole
  - Minimize triangulation area
  - Minimize the maximum dihedral angle

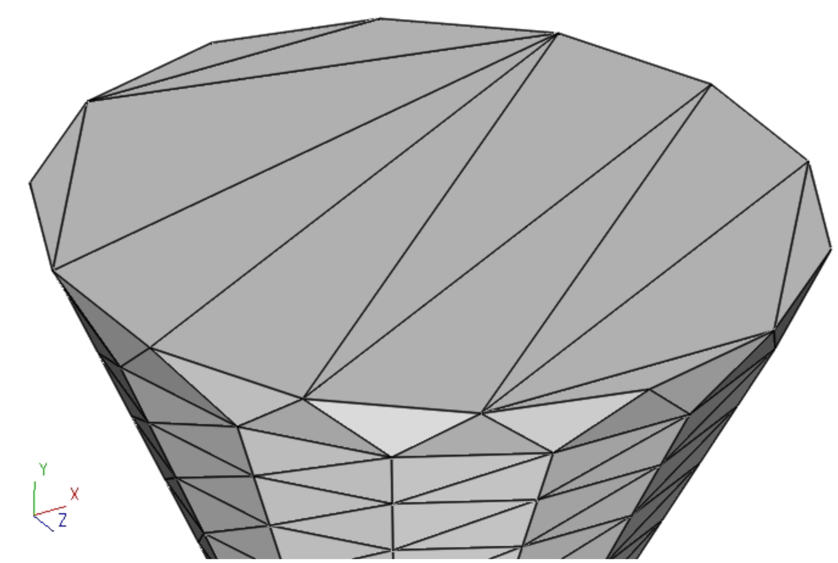




HOLE



MIN TRIANGLE AREA

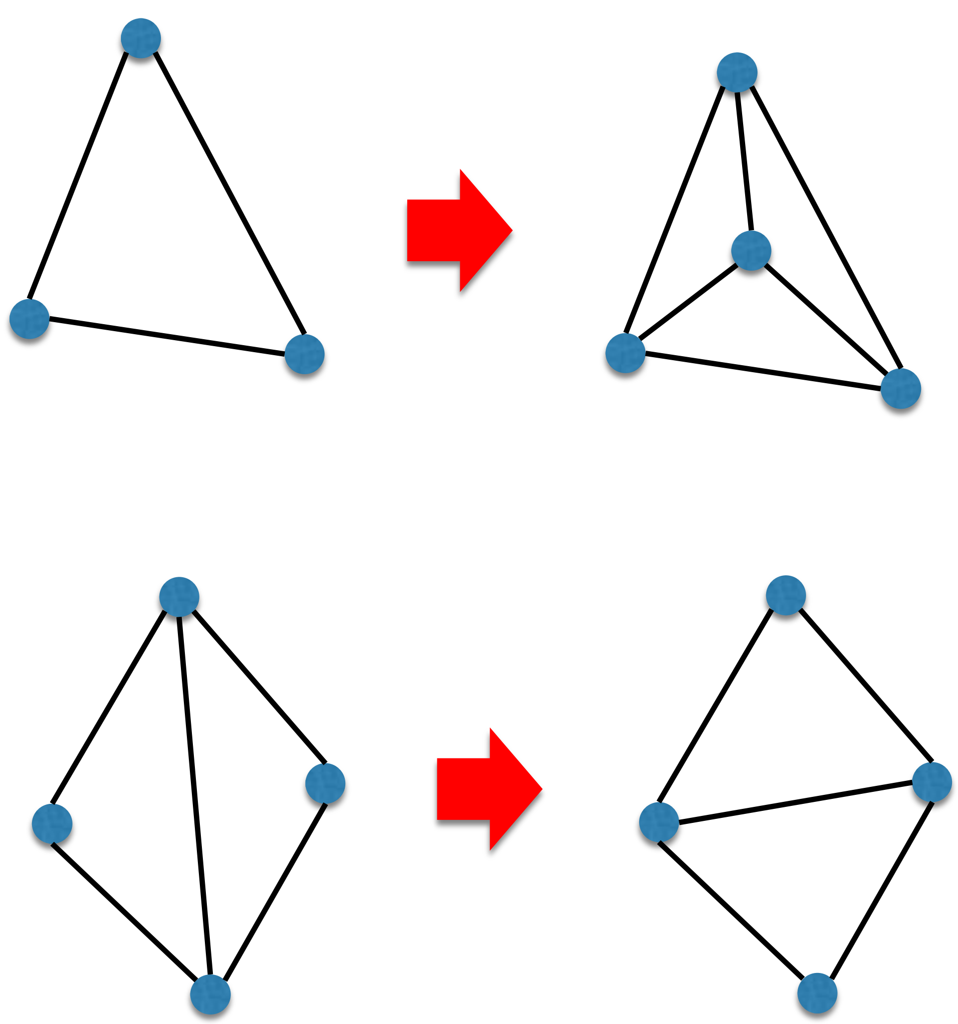


DIHEDRAL ANGLE

Filling Holes

[Liepa, SGP 03]

- Mesh refinement
  - Refine the triangulation to match the triangulation of the surrounding triangles
  - Relaxing interior edges to maintain a Delaunay-like triangulation (edge flip)



Filling Holes

[Liepa, SGP 03]

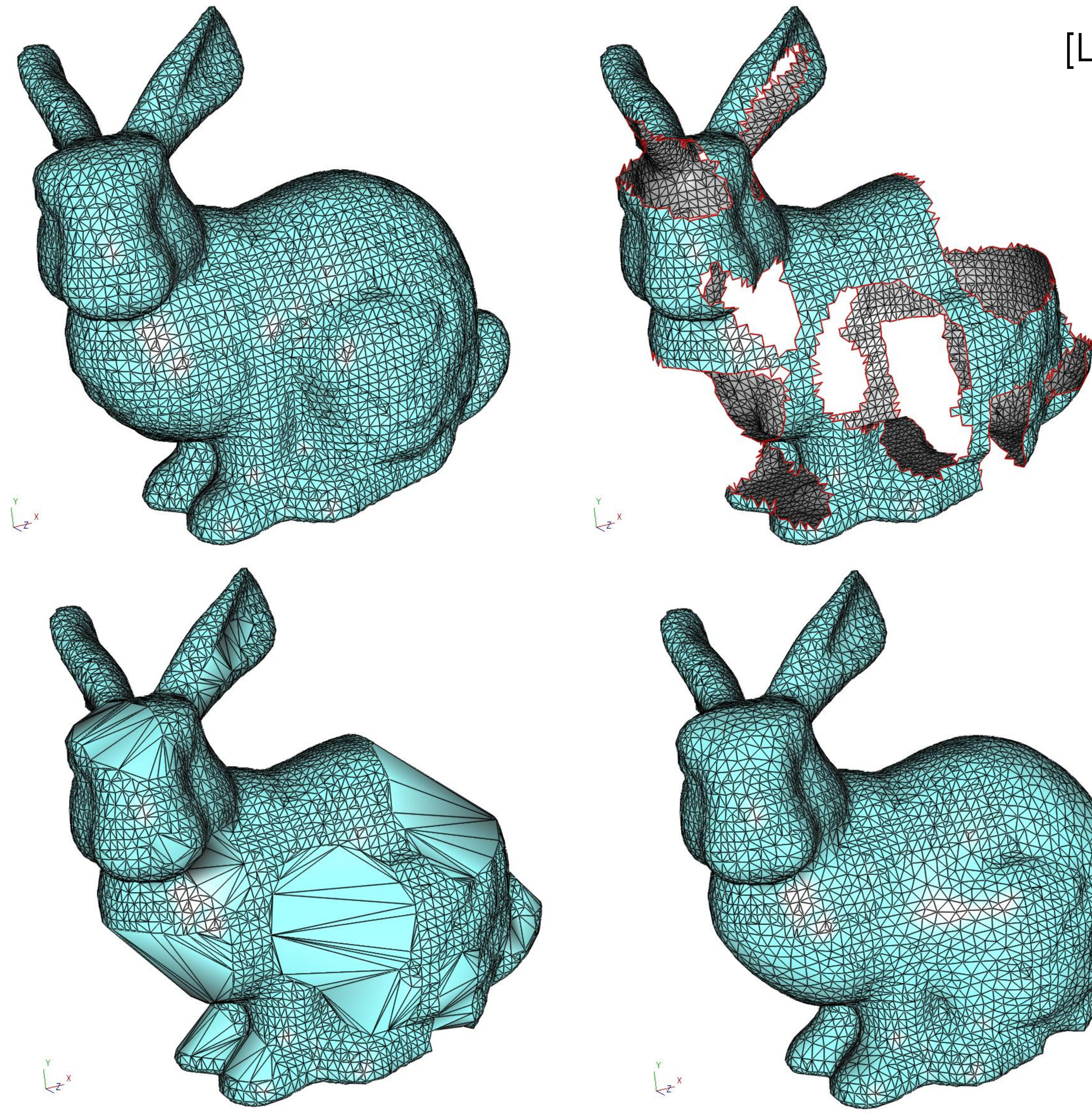
- Mesh Fairing
  - Making a surface smooth by minimizing a fairness functional
  - Set a linear system where for non-boundary vertices we constrain that $U^2(\mathbf{p}_i) = 0$

$$U(\mathbf{p}_i) = \mathbf{p}_i + \frac{1}{W} \sum_{j \in N_i} w_{ij}(\mathbf{p}_j - \mathbf{p}_i)$$

$$U^2(\mathbf{p}_i) = U(\mathbf{p}_i) + \frac{1}{W} \sum_{j \in N_i} w_{ij}(U(\mathbf{p}_j) - U(\mathbf{p}_i))$$

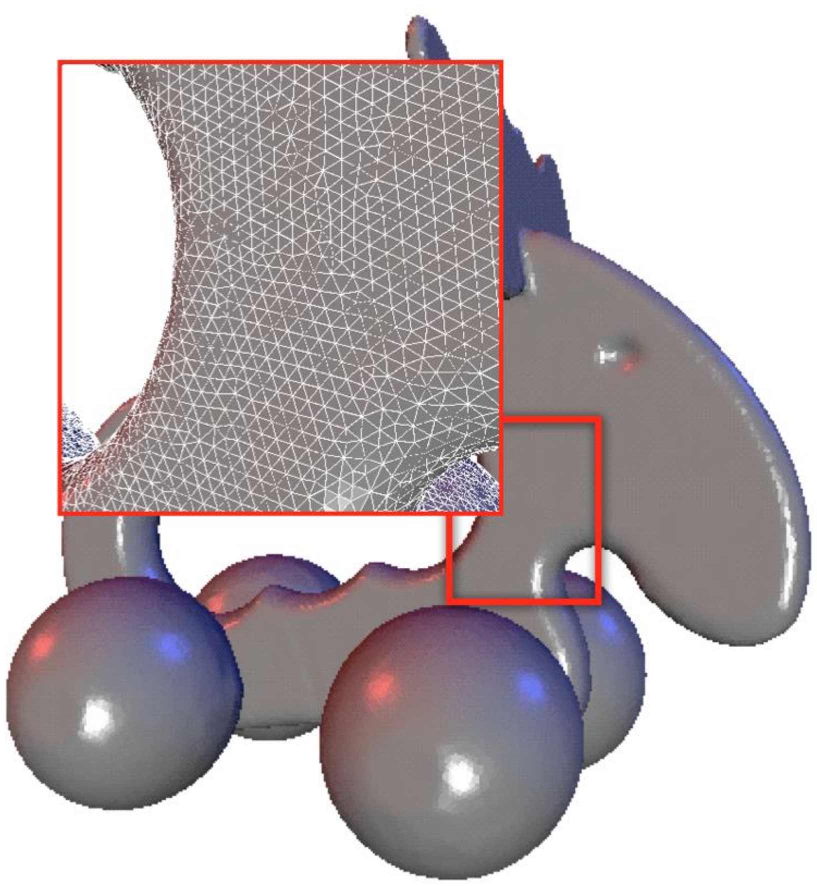
Filling Holes

[Liepa, SGP 03]

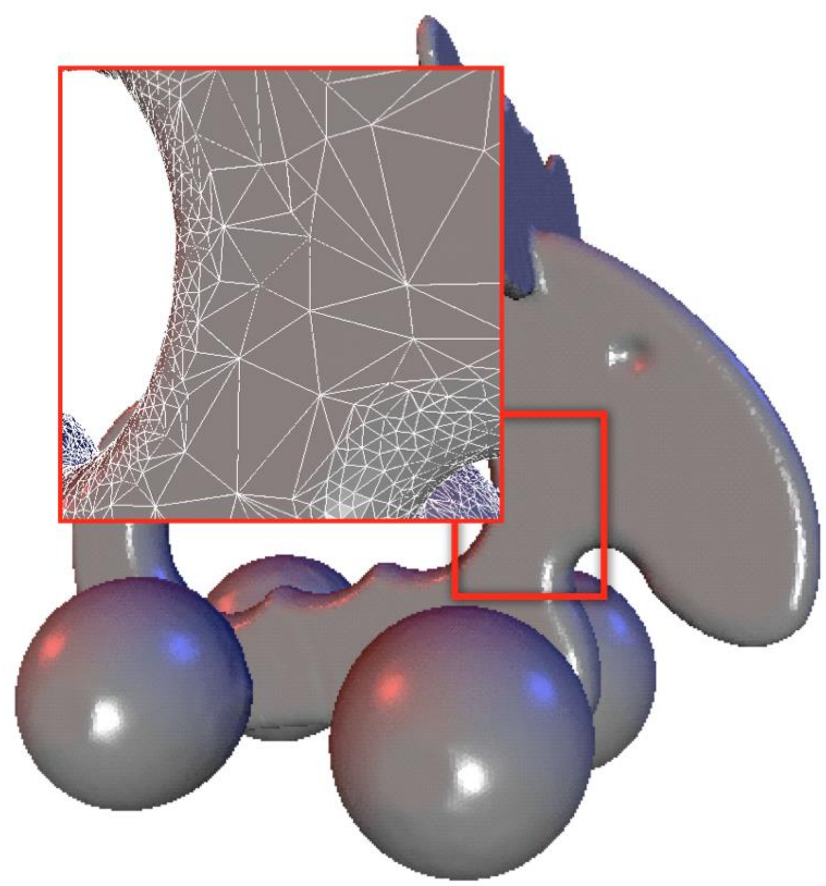


Mesh Simplification

- Reduce the amount of polygons of a mesh with minimal effect on the geometry



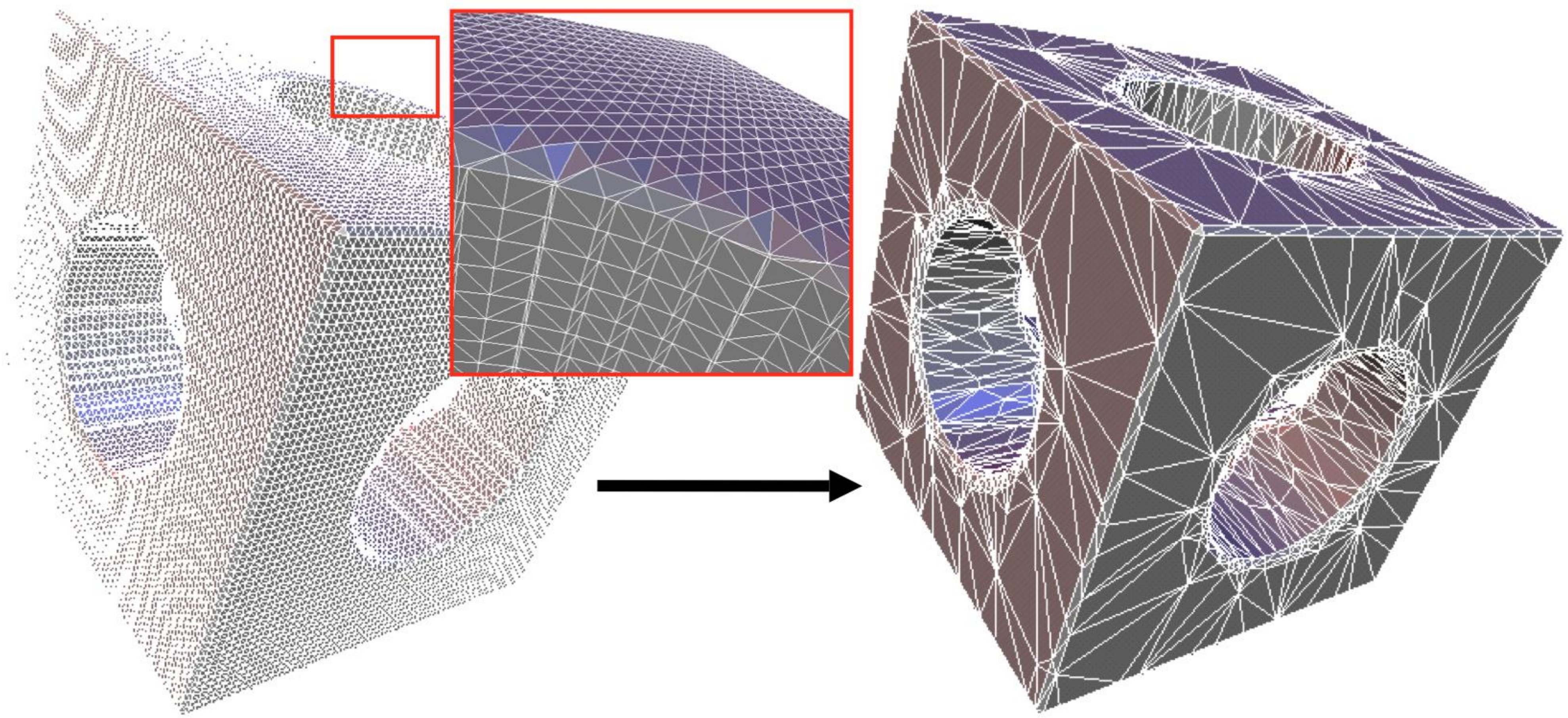
150 K triangles



80 K triangles

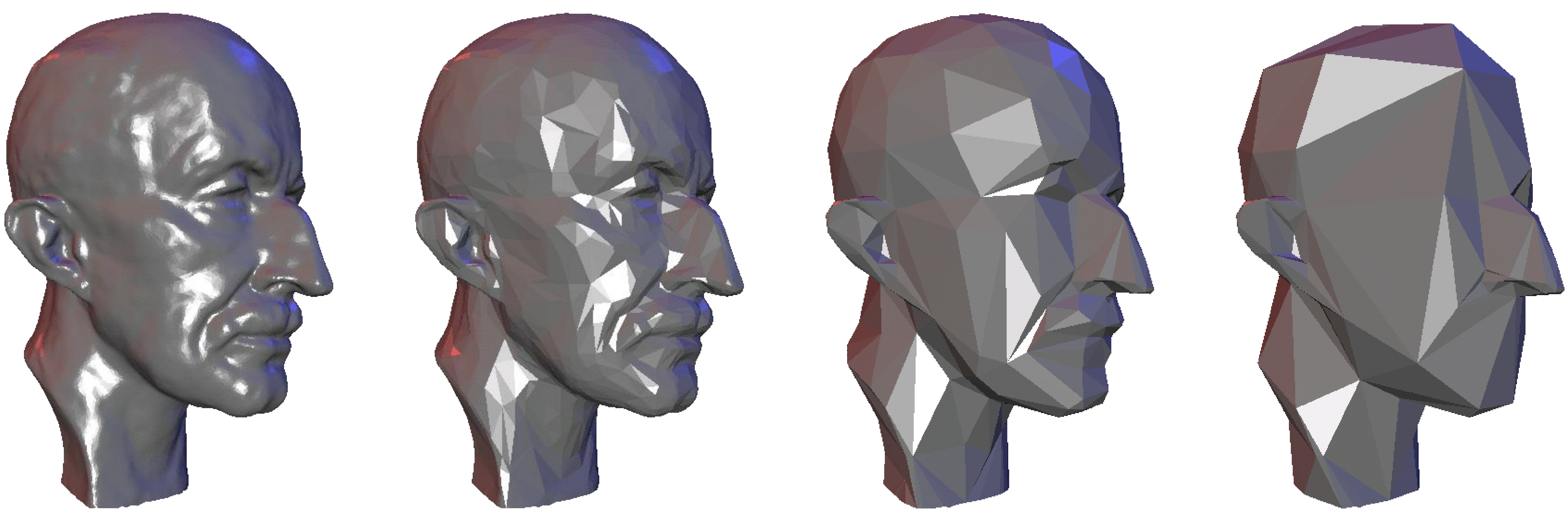
Mesh Simplification

- Erase redundant information with minimal effect on the geometry (in case of iso-surface extraction)



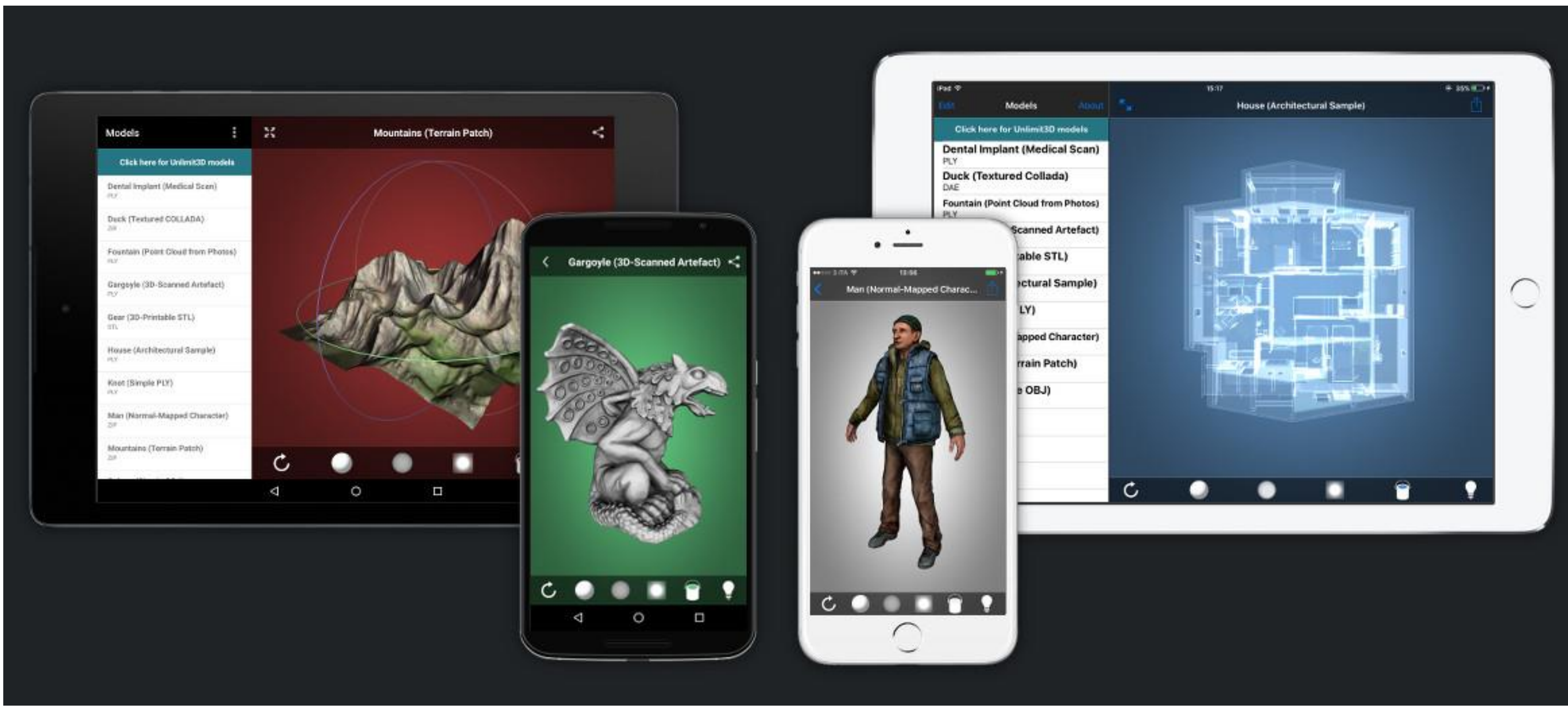
Mesh Simplification

- Multi-resolution hierarchies for
  - Efficient geometry processing
  - Level-of-detail (LOD) rendering

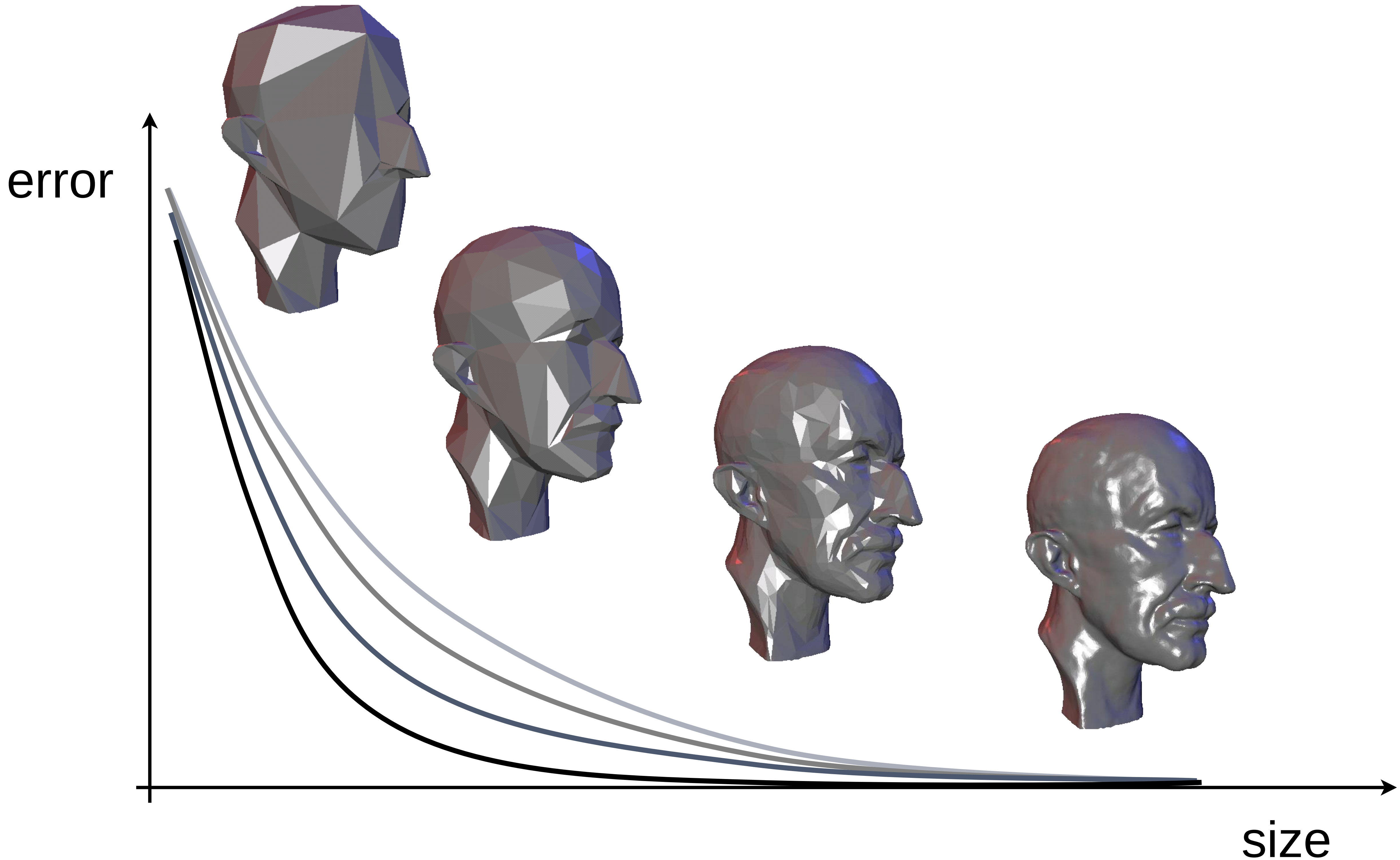


Mesh Simplification

- Adaptation to hardware capabilities

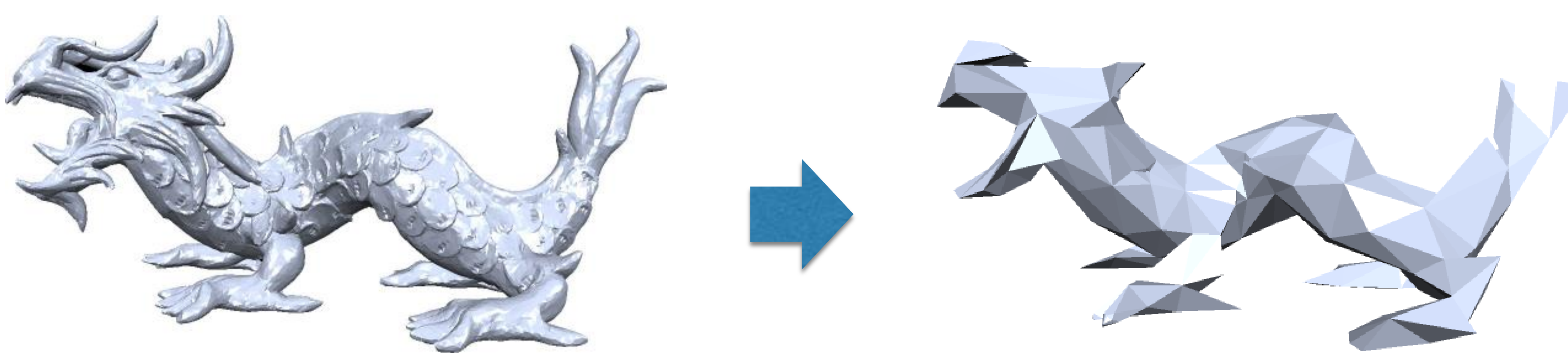


Size-Quality Tradeoff



Problem Statement

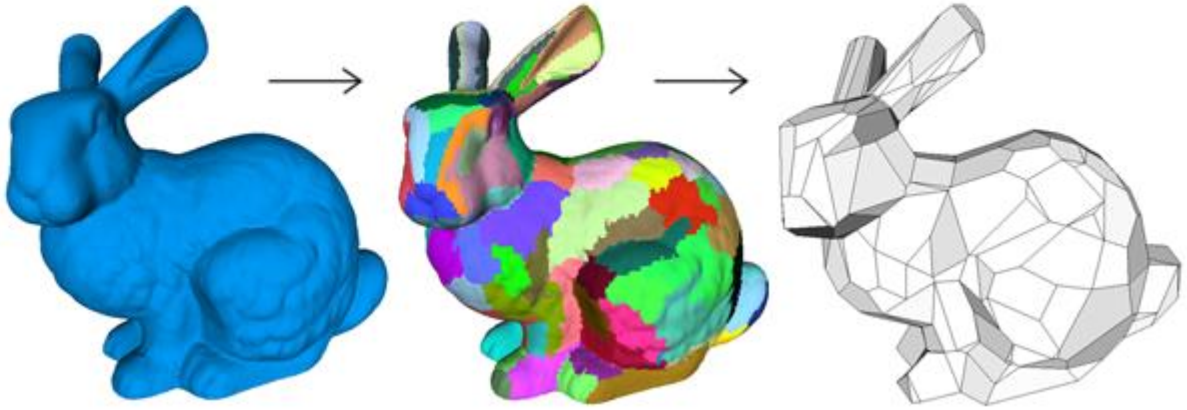
- Given $\mathcal{M} = (\mathcal{V}, \mathcal{F})$, find $\mathcal{M}' = (\mathcal{V}', \mathcal{F}')$ such that
  - $|\mathcal{V}'| = n < |\mathcal{V}|$ and $\|\mathcal{M}' - \mathcal{M}\|$ is minimal, or
  - $\|\mathcal{M}' - \mathcal{M}\| < \epsilon$ and $|\mathcal{V}'|$ is minimal
- Reduce the amount of vertices minimizing the error, or keep the error below a threshold and minimize the number of vertices



$\mathcal{M}$ $\qquad\qquad$ $\mathcal{M}'$

Mesh Decimation Methods

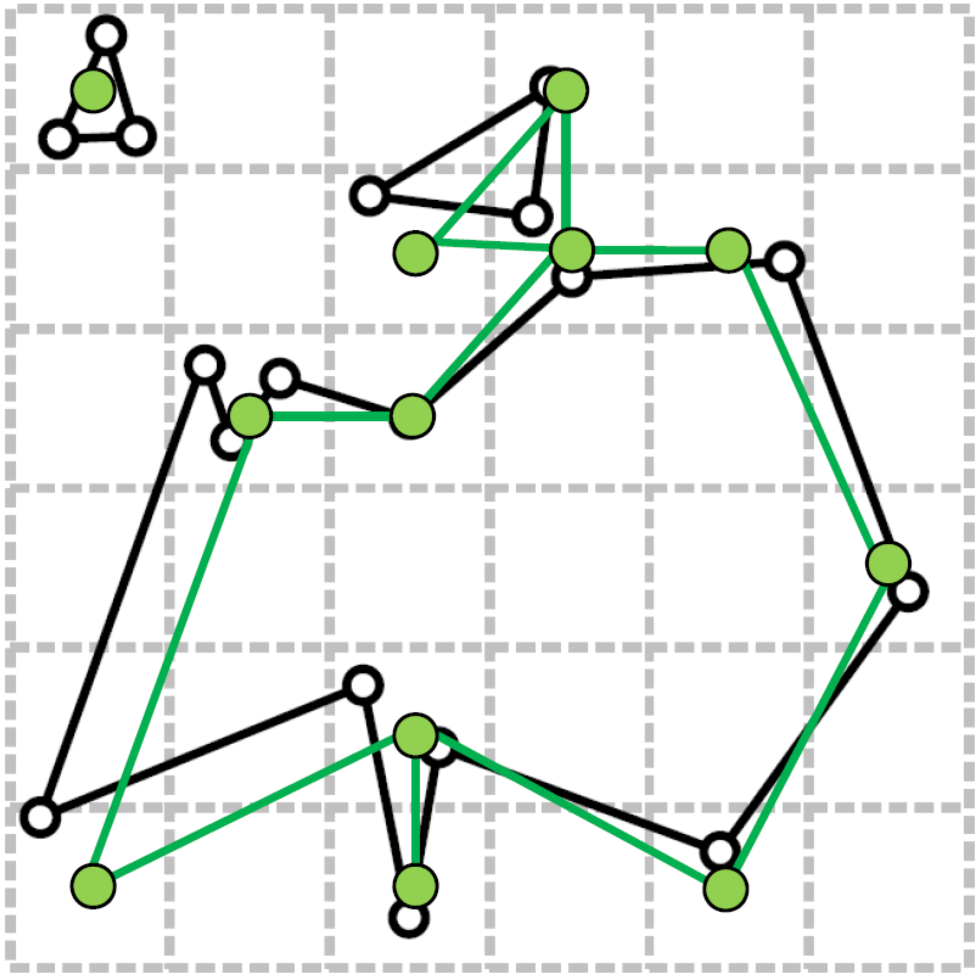
- **Vertex clustering**
- **Incremental decimation**
- Resampling
- Mesh approximation



Vertex Clustering

[Rossignac et al., MCG 93]

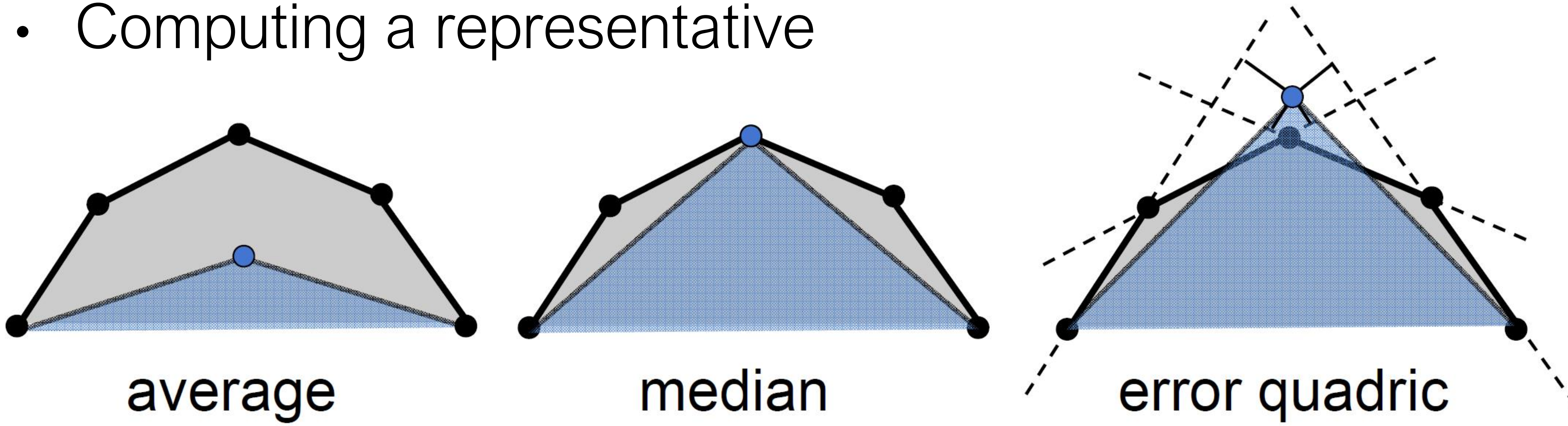
- Cluster Generation
  - Uniform 3D grid
  - Map vertices to cluster cells

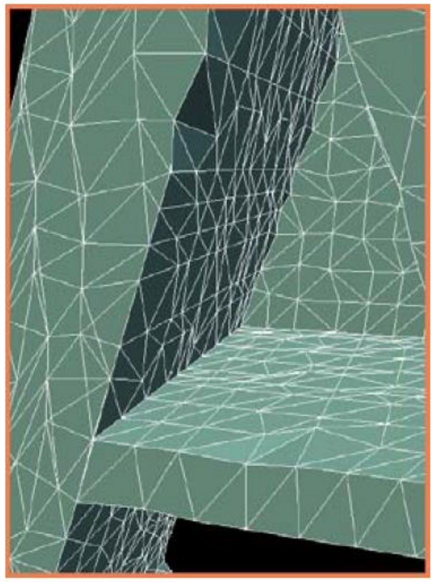


Vertex Clustering

[Rossignac et al., MCG 93]

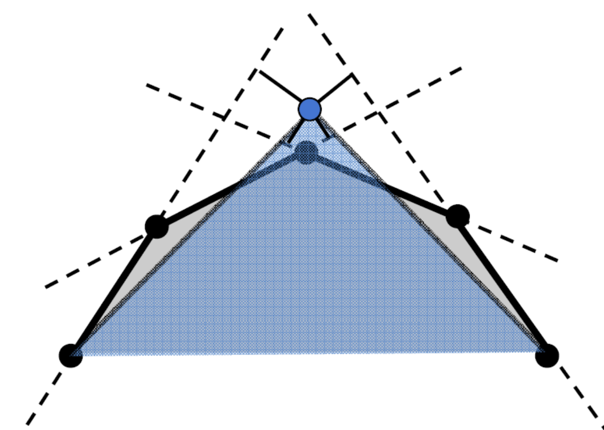
- Cluster Generation
- Computing a representative





Quadric Error Metrics

- Minimize distance to neighboring triangles' planes
- Squared distance to plane



$$\mathbf{p} = (x, y, z, 1)^T$$

$$ax + by + cz + d = 0 \Rightarrow \mathbf{q} = (a, b, c, d)^T$$

$$dist(\mathbf{p}, \mathbf{q})^2 = (ax + by + cz + d)^2 = (\mathbf{q}^T\mathbf{p})^2 = (\mathbf{p}^T\mathbf{q})(\mathbf{q}^T\mathbf{p})$$
$$= \mathbf{p}^T(\mathbf{q}\mathbf{q}^T)\mathbf{p} = \mathbf{p}^T\mathbf{Q}_\mathbf{q}\mathbf{p}$$

$$\mathbf{Q}_\mathbf{q} = \begin{bmatrix} a^2 & ab & ac & ad \\ ab & b^2 & bc & bd \\ ac & bc & c^2 & cd \\ ad & bd & cd & d^2 \end{bmatrix}$$

Quadric Error Metrics

- Sum of distances of the vertex from the plane of all the incident faces planes

$$\sum_i dist(\mathbf{p}, \mathbf{q}_i)^2 = \sum_i \mathbf{p}^T \mathbf{Q}_{\mathbf{q}_i} \mathbf{p} = \mathbf{p}^T \left(\sum_i \mathbf{Q}_{\mathbf{q}_i} \right) \mathbf{p} = \mathbf{p}^T \mathbf{Q}_{\mathbf{p}} \mathbf{p}$$

- Point that minimizes the error, setting the partial derivative to zero

$$\mathbf{p}^T \mathbf{Q}_{\mathbf{p}} \mathbf{p} = a^2x^2 + b^2y^2 + c^2z^2 + 2abxy + 2acxz + 2bczy + 2adx + 2bdy + 2cdy + d^2 = 0$$

$$\begin{cases} \frac{\partial}{\partial x} = a^2x + aby + acz + ad = 0 \\ \frac{\partial}{\partial y} = abx + b^2y + bcz + bd = 0 \\ \frac{\partial}{\partial z} = acx + bcy + c^2z + cd = 0 \end{cases}$$

Vertex Clustering

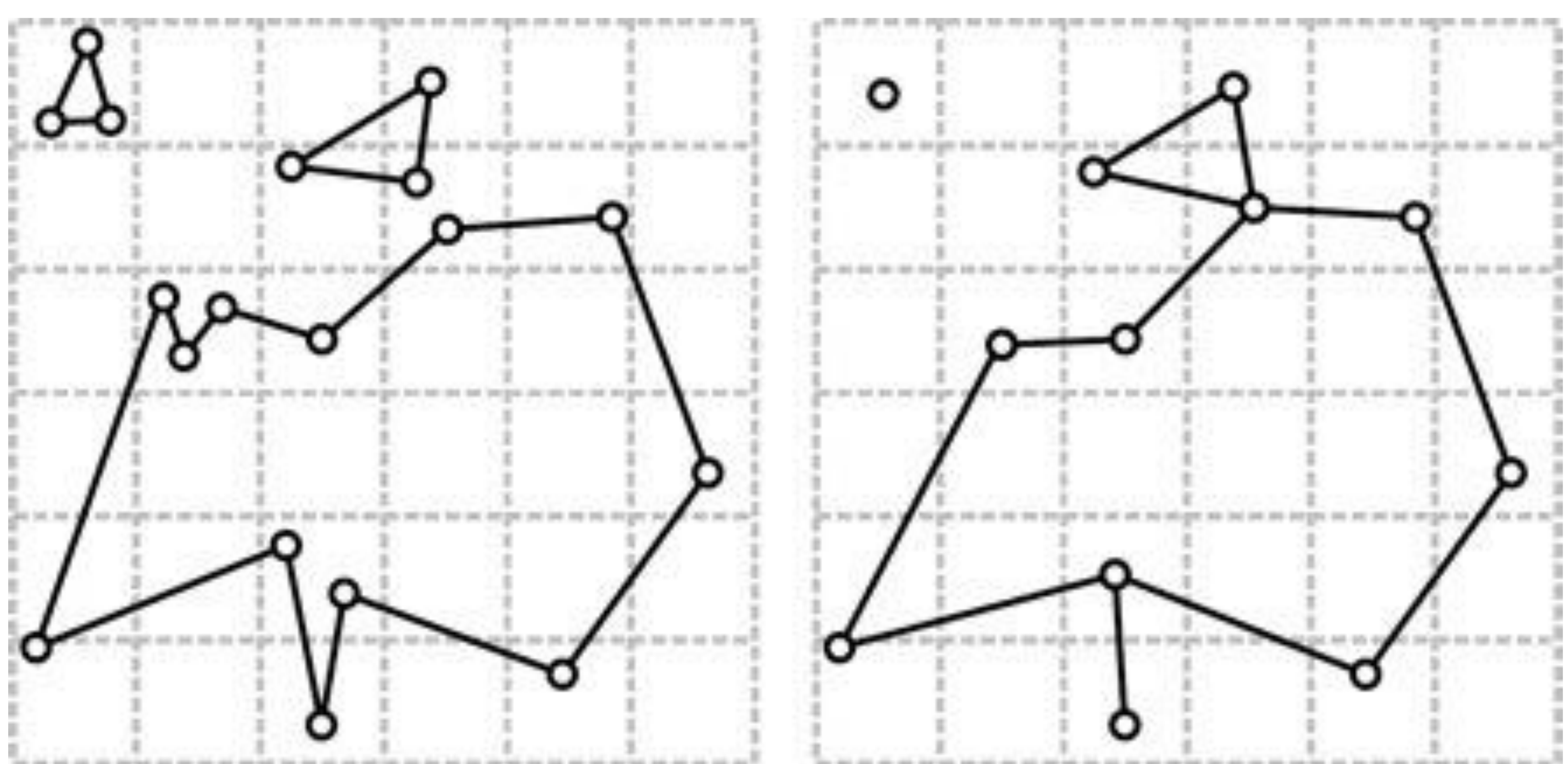
[Rossignac et al., MCG 93]

- Cluster Generation
- Computing a representative
- Mesh generation
  - Clusters $\mathbf{p} \leftrightarrow \{\mathbf{p}_0, \ldots, \mathbf{p}_n\}$ $\mathbf{q} \leftrightarrow \{\mathbf{q}_0, \ldots, \mathbf{q}_m\}$
  - Connect $(\mathbf{p}, \mathbf{q})$ if there was an edge $(\mathbf{p}_i, \mathbf{q}_j)$

Vertex Clustering

[Rossignac et al., MCG 93]

- Does not preserve topology (faces may degenerate to edges, genus may change, non-manifold geometry)
- Approximation depends on grid resolution

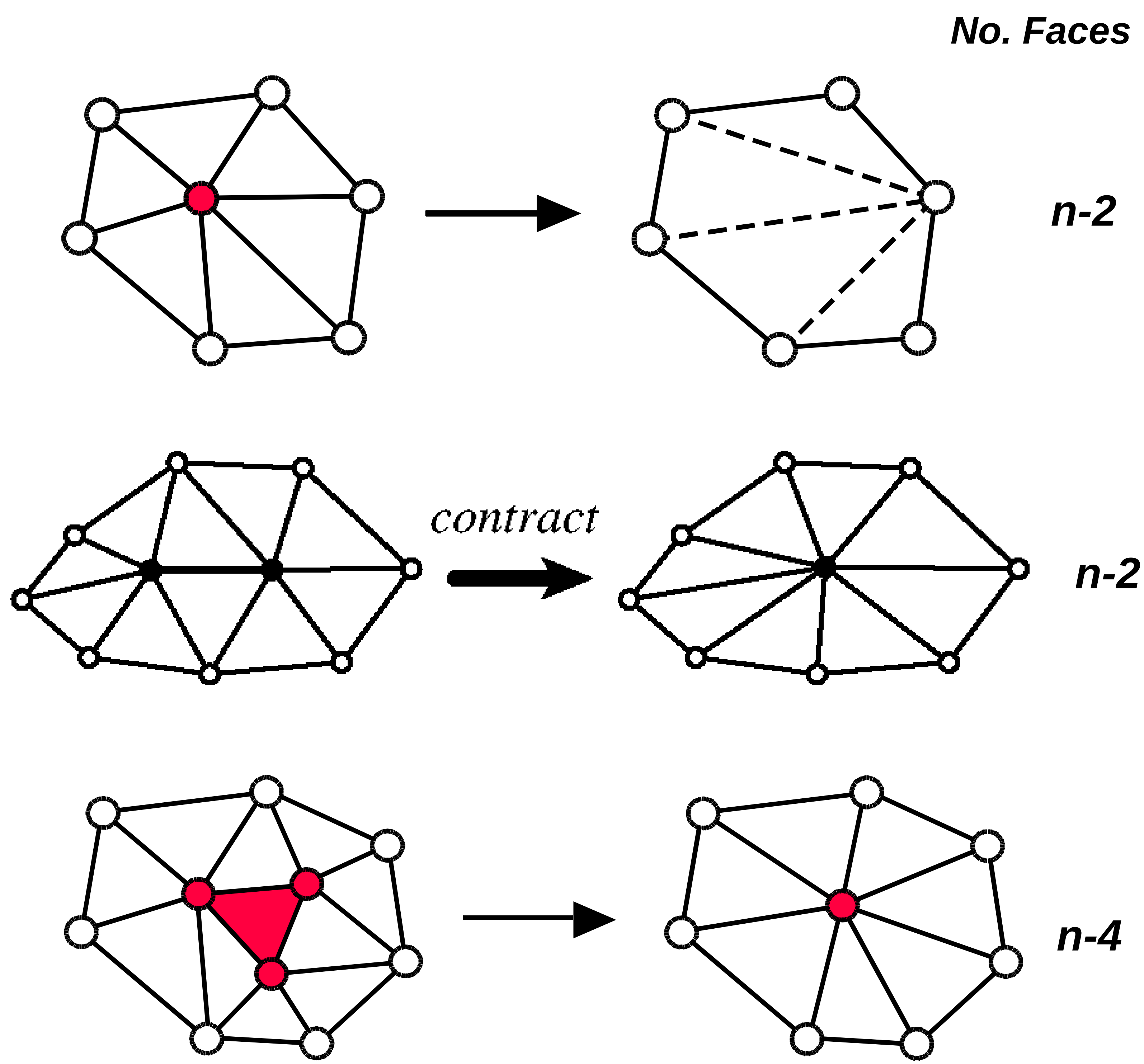


Incremental Decimation

- Based on Local Updates Operations
- All of the methods such that:
    - Simplification proceeds as a sequence of small changes of the mesh (in a greedy way)
    - Each update reduces mesh size and [~monotonically] decreases the approximation precision

Local Operation

- Vertex removal
- Edge collapse
- Triangle collapse



General Setup

```
Repeat:
   Pick a profitable operation
   Apply local operator
Until no further reduction possible
```

Greedy Optimization

```
For each region
   Evaluate quality after simulated operation
   Put the operation in a queue(quality, region)

Repeat:
  Pick best operation from the heap
  Execute the operation
  Update queue
Until no further reduction possible
```

Greedy Optimization with Global Error control

```
For each region
   Evaluate quality after simulated operation
   Put the operation in a queue(quality, region)

Repeat:
  Pick best operation from the heap
  If introduced error < ε
     Execute the operation
     Update queue
Until no further reduction possible
```

Quadric Edge Collapse

[Garland et al., SIGGRAPH 97]

- Initialization
  - Assign each vertex the quadric built from all its incident triangles' planes

- Decimation
  - Collapse the edge $(\mathbf{p}_1, \mathbf{p}_2) \rightarrow \mathbf{p}_3$ where $\mathbf{p}_3$ is the point the minimize the quadric error using the quadric $\mathbf{Q}_{\mathbf{p}_3} = \mathbf{Q}_{\mathbf{p}_1} + \mathbf{Q}_{\mathbf{p}_2}$
  - The point $\mathbf{p}_3$ receives the quadric error $\mathbf{Q}_{\mathbf{p}_1} + \mathbf{Q}_{\mathbf{p}_2}$
  - We start to collapse the edge that introduce less approximation in the shape

References

- Liepa, Peter. "Filling holes in meshes." *Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing*. Eurographics Association, 2003.
- Rossignac, Jarek, and Paul Borrel. "Multi-resolution 3D approximations for rendering complex scenes." *Modeling in computer graphics*. Springer Berlin Heidelberg, 1993. 455-465.
- Garland, Michael, and Paul S. Heckbert. "Surface simplification using quadric error metrics." *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*. ACM Press/Addison-Wesley Publishing Co., 1997.