Summary

- Topics:

1. Bio-medical imaging
2. Image processing
3. Geometric modeling and computer graphics
4. Mesh generation
    - Marching Cubes/Dual Contouring
    - Tri/Tet Meshing
    - Quad/Hex Meshing
    - Quality Improvement
5. Computational mechanics
6. Bio-medical applications

Mechanical Engineering
Carnegie Mellon

Topic 4: Mesh Generation – Dual Contouring

Jessica Zhang
Department of Mechanical Engineering
Courtesy Appointment in Biomedical Engineering
Carnegie Mellon University
jessicaz@andrew.cmu.edu
http://www.andrew.cmu.edu/user/jessicaz

Introduction

- The dual contouring method was developed in Prof. Joe Warren's group in Rice University.

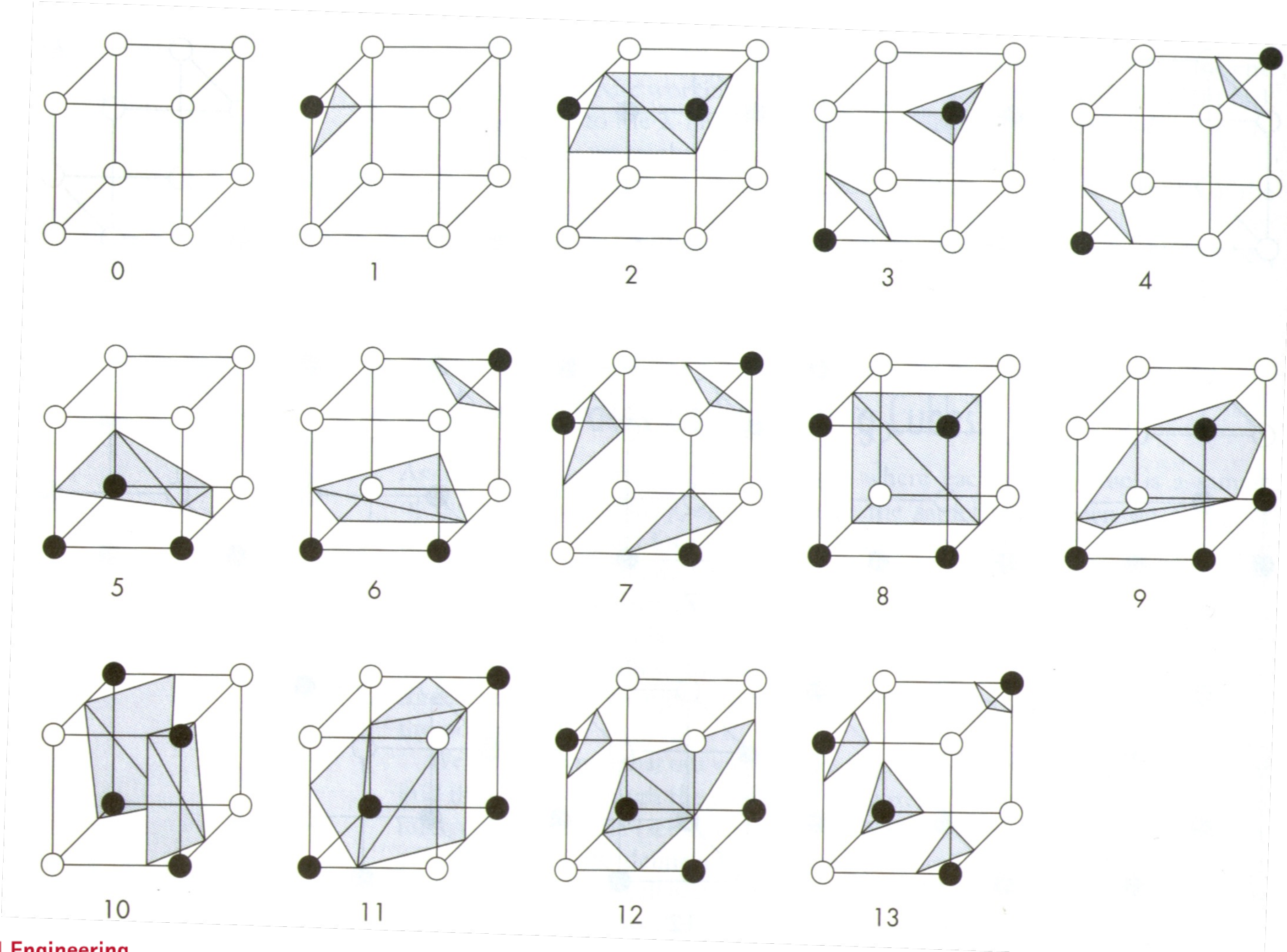
- A paper was published in SIGGRAPH 2002.

  Dual Contouring of Hermite Data. Ju T., Losasso F., Schaefer S. and Warren J. ACM SIGGRAPH 2002, pages 339-346.

- This is an octree-based method for contouring a signed grid whose edges are tagged by Hermite data (i.e., exact intersection points and normals), using a numerically stable representation for quadratic error functions.

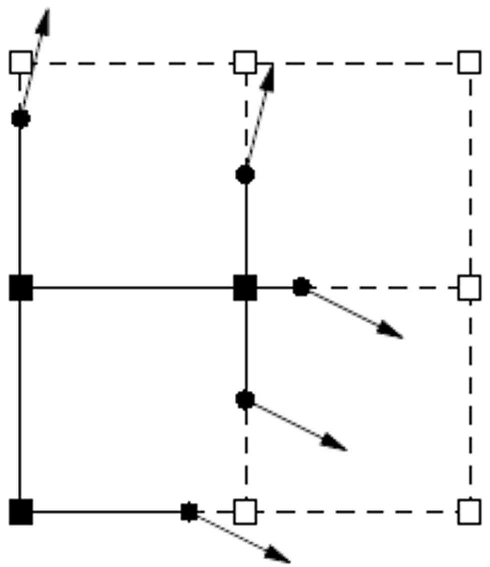
Marching Cubes

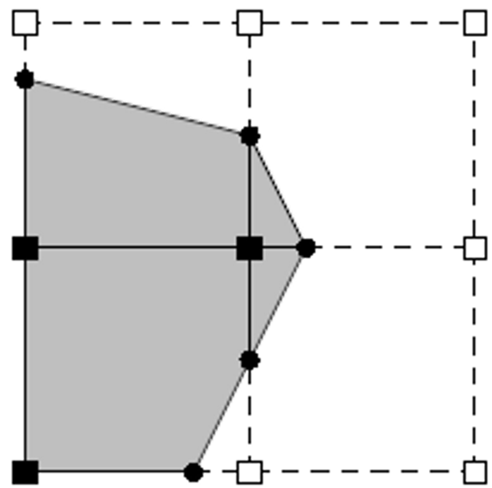
- Marching Cubes and its variants visit each cell, and generate one or more polygons for each cube in the grid that intersects the contour.

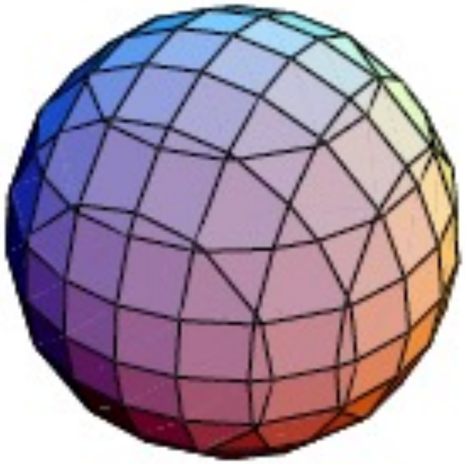


Marching Cubes

- In Marching Cubes, the vertices of these polygons are positioned at the intersection of the contour with the edges of the cubes.
- The constructed contour consists of a collection of polygons that approximate the restriction of the contour to individual cubes in the grid.







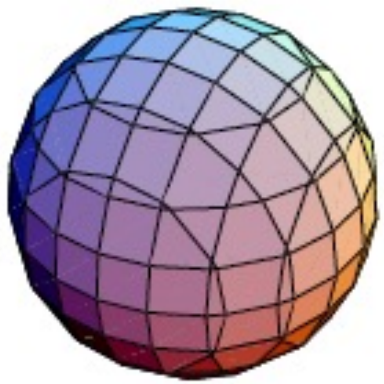
Mechanical Engineering
Carnegie Mellon

Dual Methods

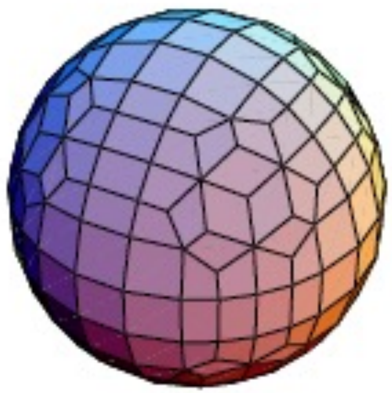
- Dual methods such as the *SurfaceNets* algorithm [Gibson 1998] generate one vertex lying on or near the contour of each cube that intersects the contour.
- Each sign change edge is shared by four cells, and one point is calculated for each cell. Four points construct a quad.
- The constructed mesh is a dual to the mesh generated by Marching Cubes.

Dual Methods

- Dual methods produce meshes with better aspect ratios since the vertices of the mesh are free to move inside the cube as oppose to being restricted to edges of the grid as in cube-based methods (Marching Cubes).



Marching Cubes

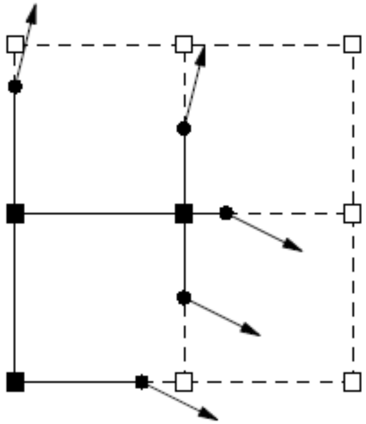


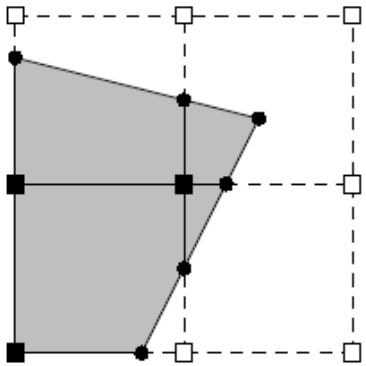
Dual Methods

Mechanical Engineering
Carnegie Mellon

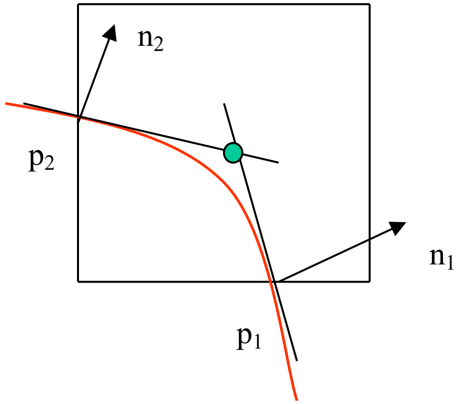
Extended Marching Cubes (EMC)

- The EMC method is a hybrid between a cube-based method and a dual method.
- EMC detects sharp features inside a cube by examining normals associated with the intersection points on the edges of the cube.
- For those cubes that do contain a feature, the method generates a vertex positioned at the minimizer of the quadratic error function.





$$QEF[x] = \sum_i (n_i \cdot (x - p_i))^2$$



Extended March Cubes

- The main advantage of EMC is that it uses Hermite data and QEFs in positioning the vertices associated with cubes that contain features.
- The Hermite approach can generate contours with sharp vertices and sharp edges.
- One drawback of EMC is the need to explicitly test for such features and to perform some type of special processing in these cases.
- Dual contouring of Hermite data is an alternative to EMC.

Dual Contouring (DC) from Hermite Data

- The Hermite data can be computed directly from the implicit definition of the contour or from a closed polygonal mesh [Foley *et al.* 1995].

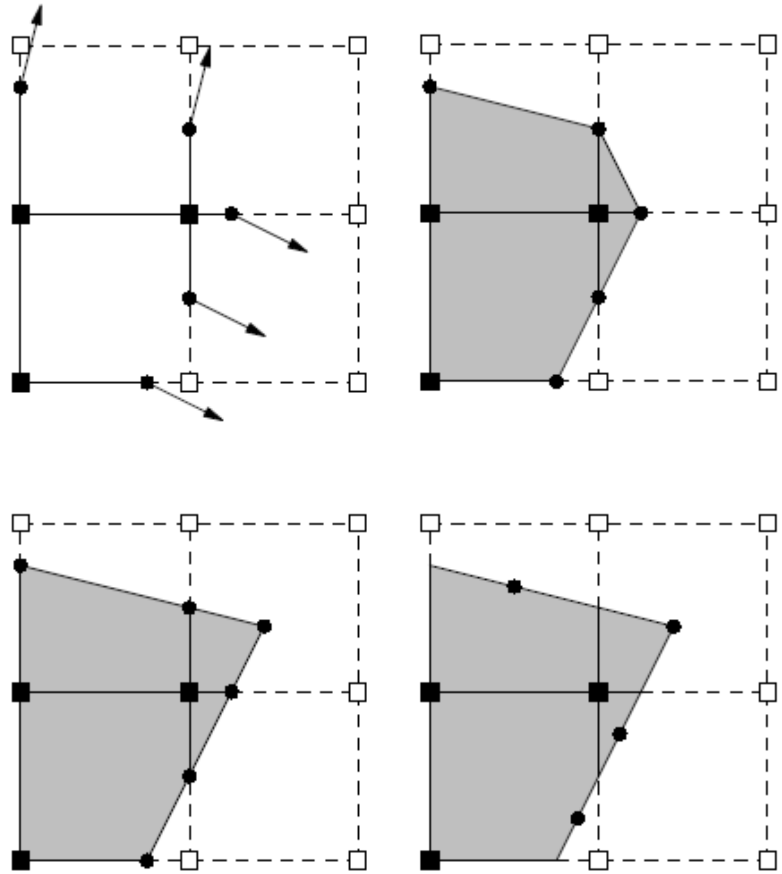
- This DC method is an interesting hybrid of the EMC method and the *SurfaceNets* method.

  – For each cube that exhibits a sign change, generate a vertex positioned at the minimizer of the quadratic error function.

  – For each edge that exhibits a sign change, generate a quad connecting the minimizing vertices of the four cubes containing the edge.

Dual Contouring (DC) from Hermite Data

- The DC method uses EMC’s feature vertex rule for positioning all vertices of the contour while using the *SurfaceNets* to determine the connectivity of these vertices.
- By using QEFs to position all of the vertices of the contour, the DC method avoids the need to explicitly test for features.



Upper left: A signed grid with edges tagged by Hermite data

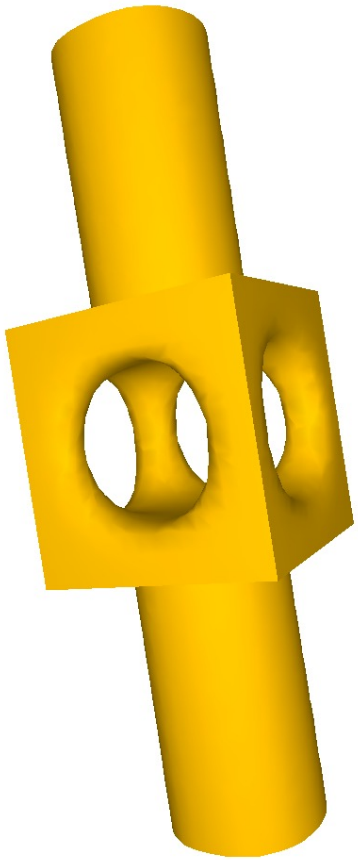
Upper right: Marching Cubes contour

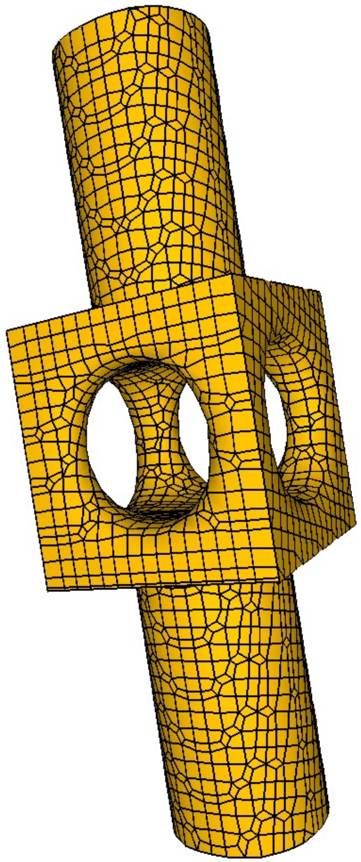
Lower left: Extended Marching Cubes contour

Lower right: Dual contour

Mechanical Engineering
Carnegie Mellon

Examples





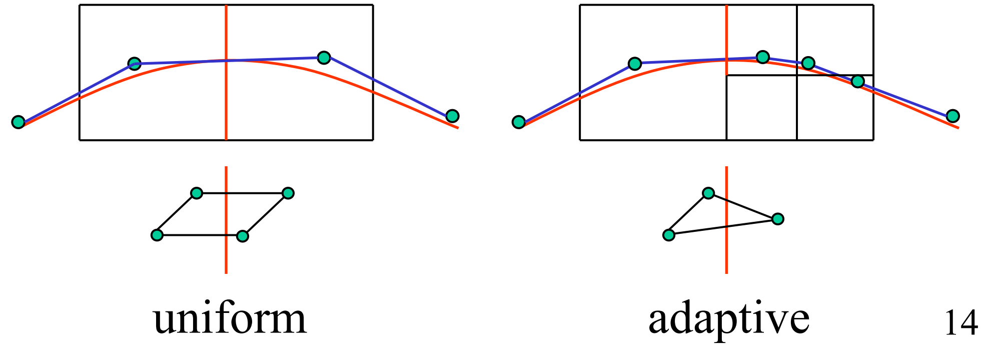
A mechanical part modeled by dual contouring Hermite data on a 64^3 grid. The left image shows a smooth shaded version of the part while the right image shows the polygonal mesh produced by dual contouring.

Adaptive Dual Contouring

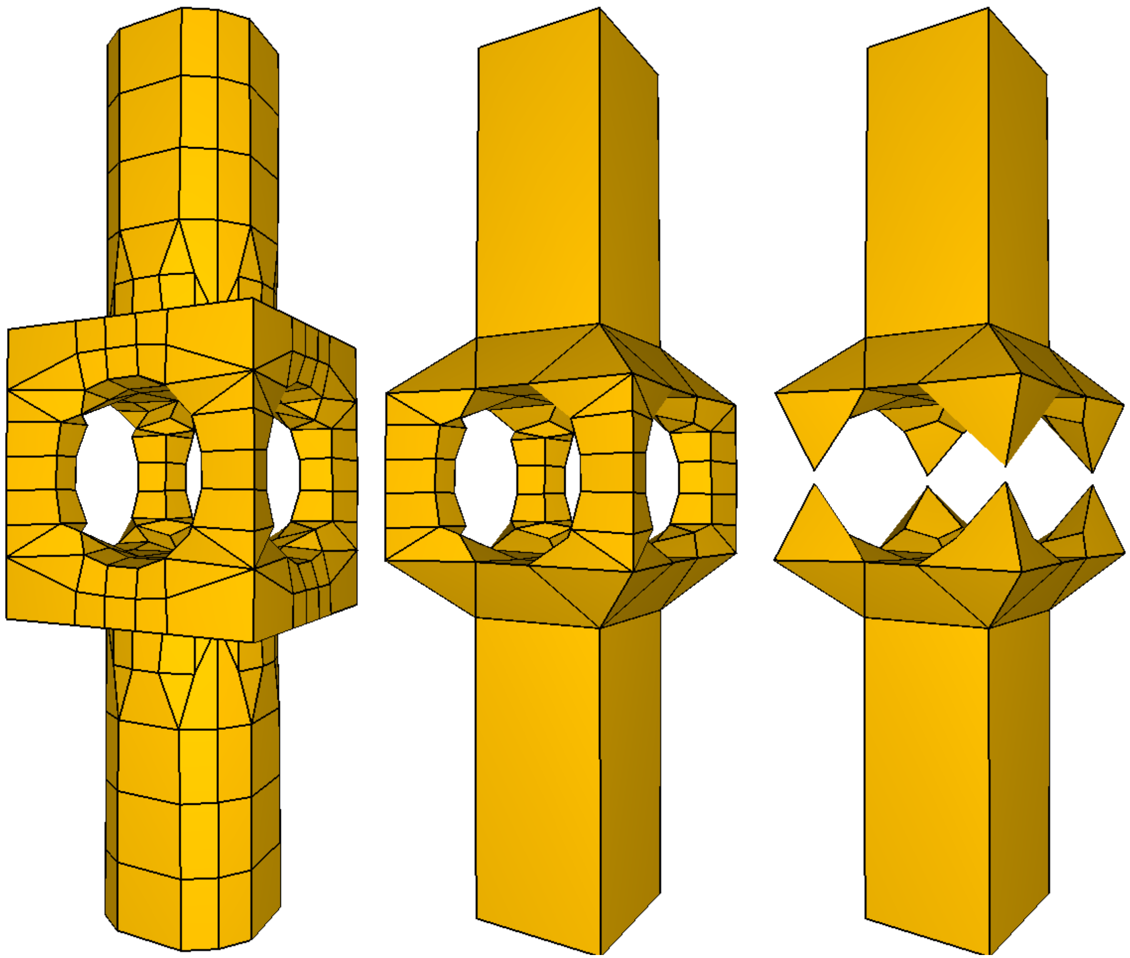
- Octree data structure is chosen to replace uniform grids and handle adaptive cases.
- Adaptive dual contouring is based on simplifying an octree whose leaves contain QEFs. Three steps:
  – Generate a signed octree whose homogeneous leaves are maximally collapsed.
  – Construct a QEF for each heterogeneous leaf and simplify the octree using these QEFs.
  – Recursively generate polygons for this simplified octree.

Polygon Generation for Simplified Octrees

- Typically we restrict the octree to have neighboring leaves that differ by at most one/two levels in order to generate meshes with good aspect ratio.
- For each sign change edge shared by three/four leave cells, generate a triangle/quadrilateral by connecting the calculated three/four minimizers.
- Minimal edge rule: only minimal edges are analyzed to construct polygons. One minimal edge is an edge of leaf cubes that don't properly contain an edge of its neighboring leaf cells.



Adaptive Meshes



Three adaptive meshes of the mechanical part

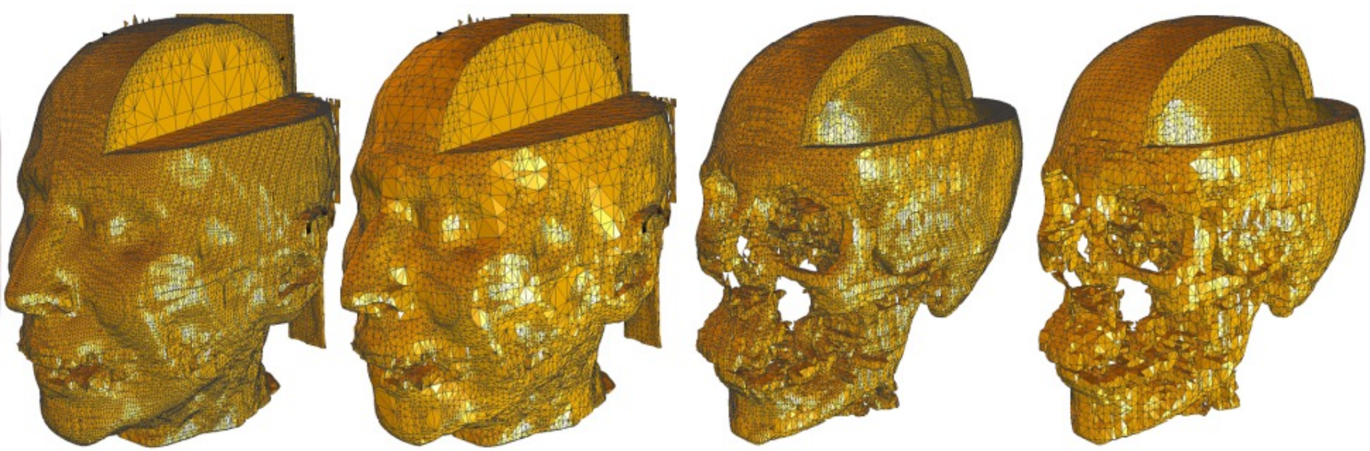
Isocontour-based Meshing

- Automatically generating adaptive and quality boundary (tri/quad) and finite (tet/hex) element from volumetric data conforming to boundaries defined as level sets of a scalar field of the volumetric domain. A boundary/finite element mesh generation software called LBIE-Mesher (Level set Boundary and Interior-Exterior Mesher) has been developed.

Requirements:

- Conforming to boundaries
- Adaptive meshes
- Quality meshes
- Correct topology
- Sharp feature preservation
- Feature sensitivity
- Crack free
- No hanging nodes

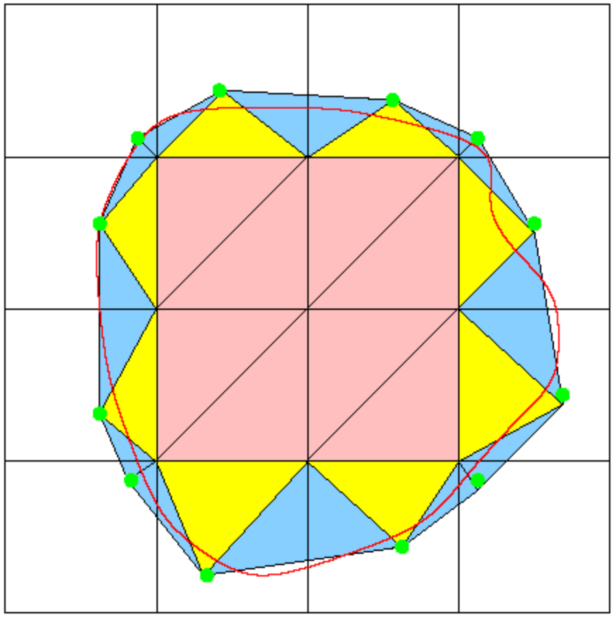

volume rendering


(a) (b) (c) (d)

Isocontour-based Tri/Tetra Meshing

* 2D Uniform Triangulation

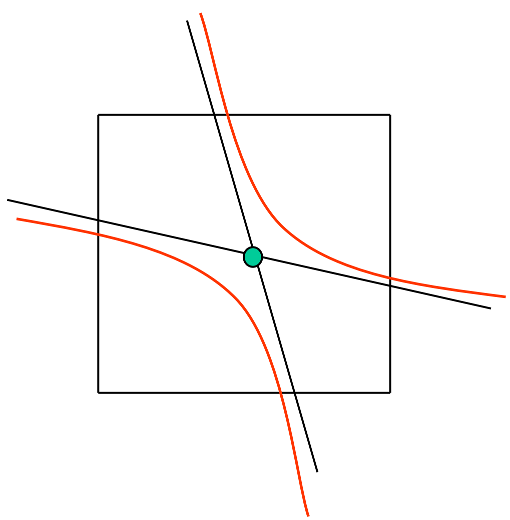
- Sign change edge – blue triangles
- Interior edge in boundary cell – yellow triangles
- Interior cell – pink triangles (Neighboring level difference <= 2)



Uniform Triangulation

Manifold Surface Construction

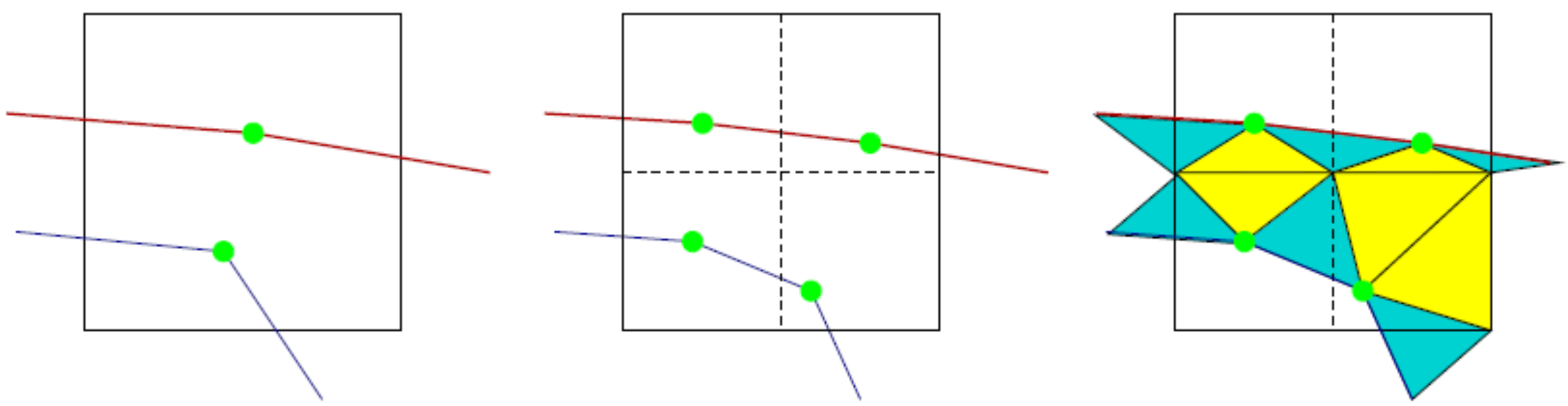
- We assume that there is only one minimizer point calculated in a cell, therefore a non-manifold situation happens if two different boundary isosurfaces of an interval volume pass through the same cell.



Mechanical Engineering
Carnegie Mellon

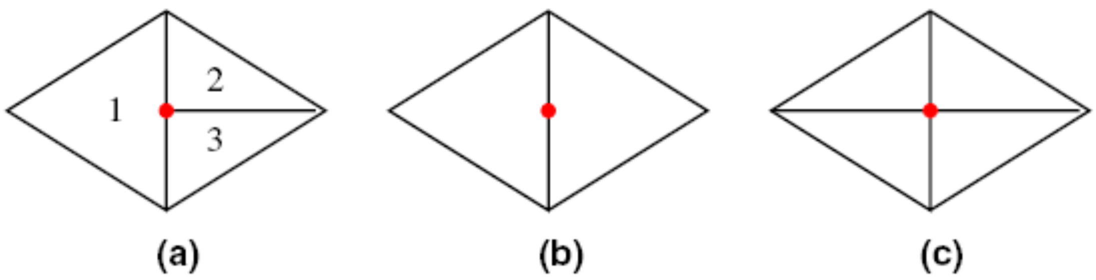
Manifold Surface Construction

- If a cell contains two boundary isosurfaces, we recursively subdivide the cell into eight identical sub-cells until each sub-cell contains at most one boundary isosurface.



Hanging Nodes

- Hanging node: a hanging node is one that is attached to the corner of one triangle but does not attach to the corners of the adjacent triangles. Generally, a hanging node is a point that is a vertex for some elements (e.g., triangle, quad, tetra, hex), but it is not for its other neighbor elements that share it. It lies on one edge or one face of its neighbors, for example, a T-Vertex.



Hanging Nodes

- Lemma: Only the interior cell needs to be modified if the splitting method is adopted.

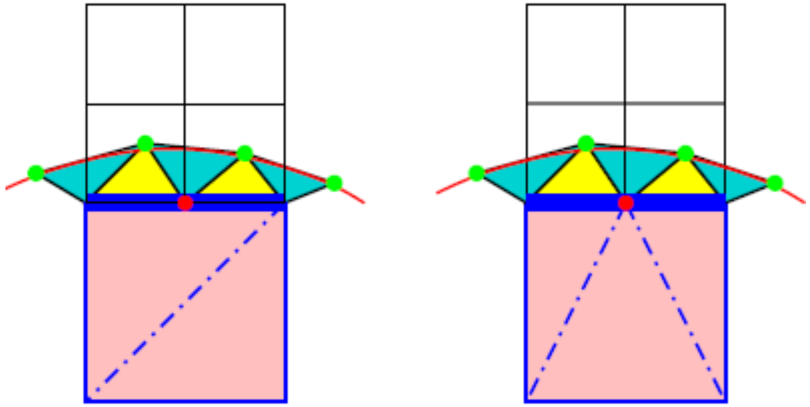
  Proof: All the leaf cells can be divided into two groups: the boundary cell and the interior cell.

  – Interior cell—Since its neighbor cells may have higher resolution levels, hanging nodes are unavoidable. There is a hanging node if we triangulate the interior cell as in the uniform case. We need to re-triangulate it to remove the hanging node.

  – Boundary cell—There are two rules for the sign change edge and the interior edge/face in boundary cells. The two rules guarantee that no hanging nodes need to be removed for the boundary cell if only the splitting method is chosen.

    - Minimal edge/face rule — only minimal edges/faces are chosen. This rule keeps the analyzed edges/faces owning the highest resolution level compared with their neighbors.

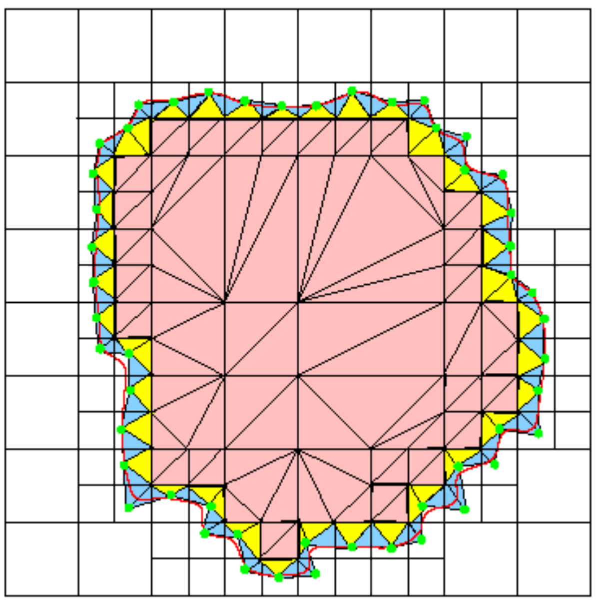
    - Only one minimizer is generated for each leaf cell.



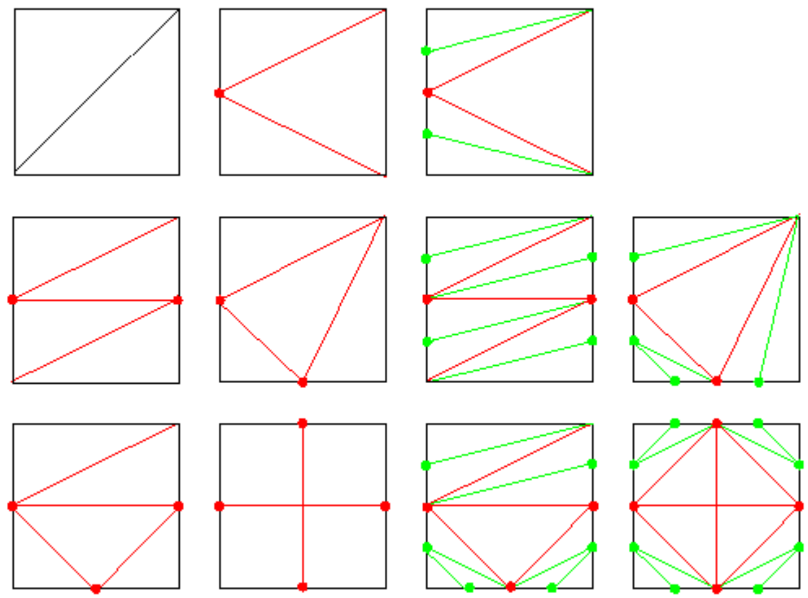
Isocontour-based Tri/Tet Meshing

* 2D Adaptive Triangulation

- Sign change edge – blue triangles
- Interior edge in boundary cell – yellow triangles
- Interior cell – pink triangles (Neighboring level difference <= 2)



Adaptive Triangulation



Case Table – interior cell (2D)

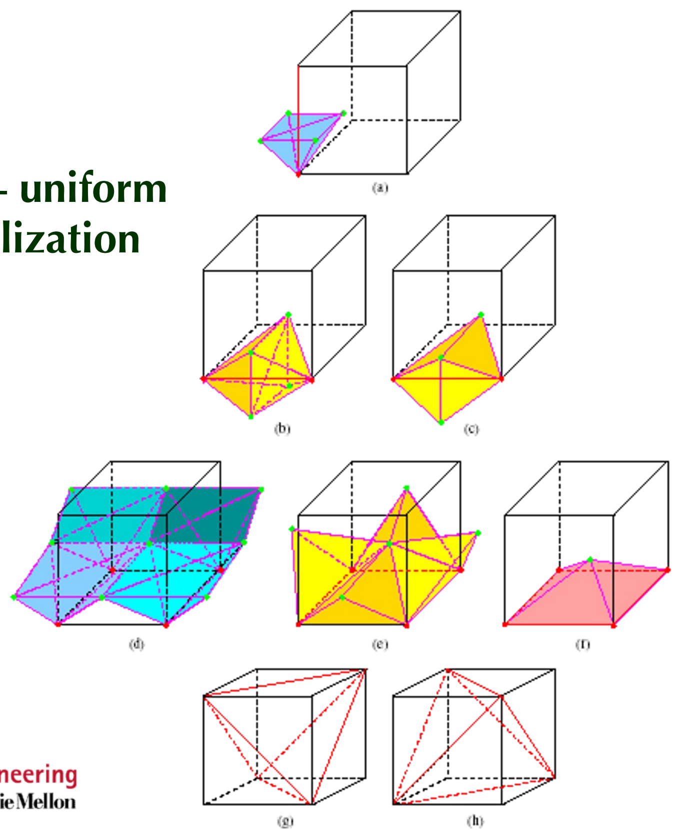
Mechanical Engineering
Carnegie Mellon

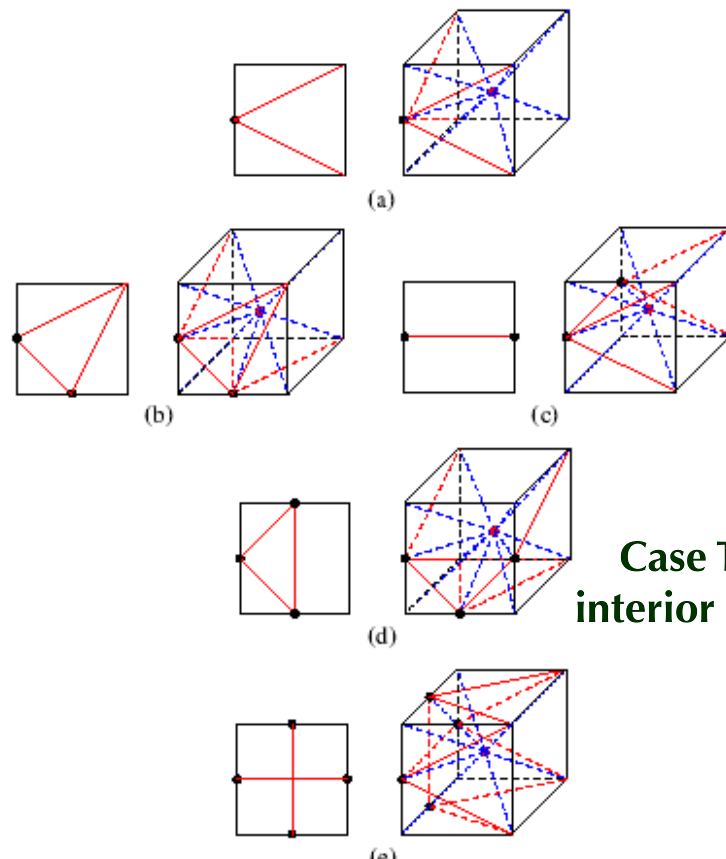
Isocontour-based Tri/Tet Meshing

* 3D Tetrahedralization

- Sign change edge – blue tets (a)
- Interior edge in boundary cell – yellow tets (b&c)
- Interior face in boundary cell – (d,e&f)
- Interior cell (g&h)

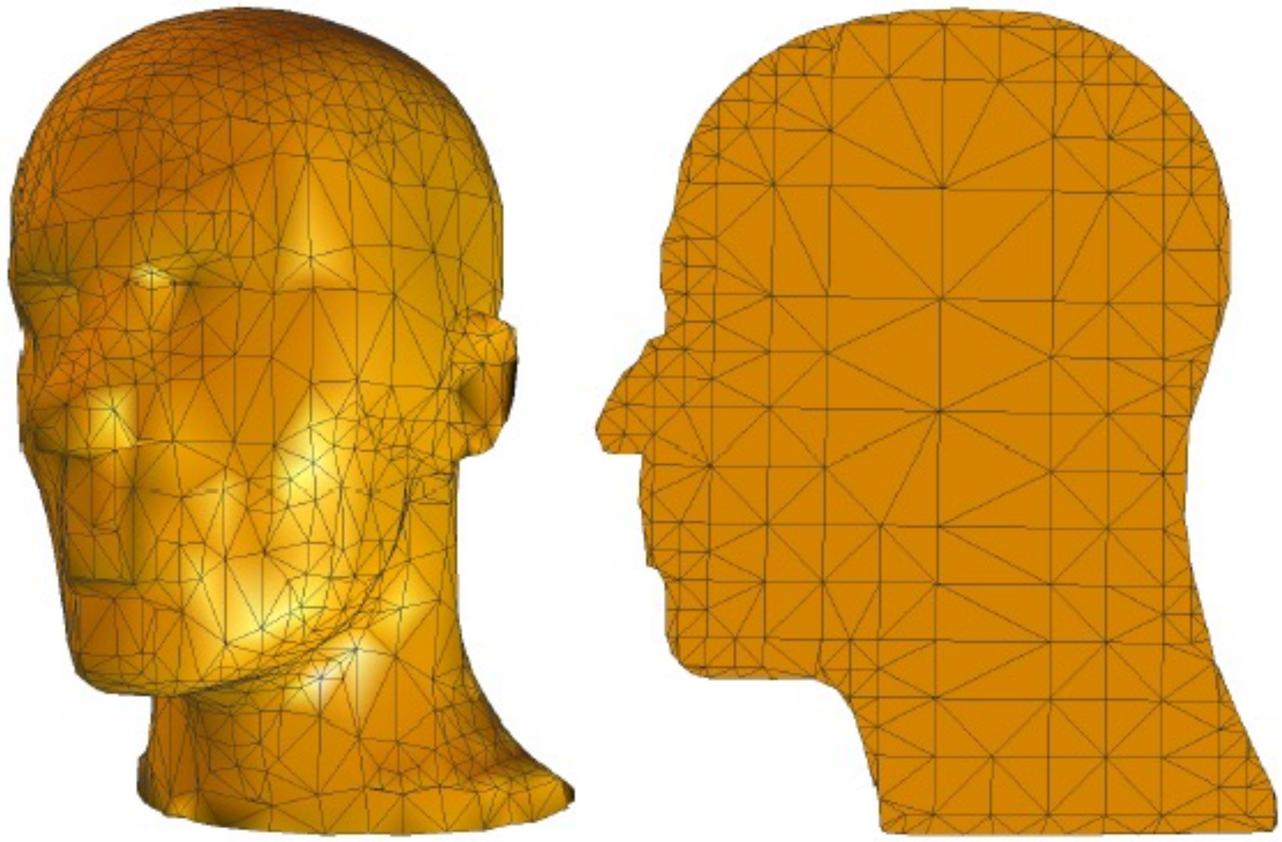
Case Table – uniform tetrahedralization



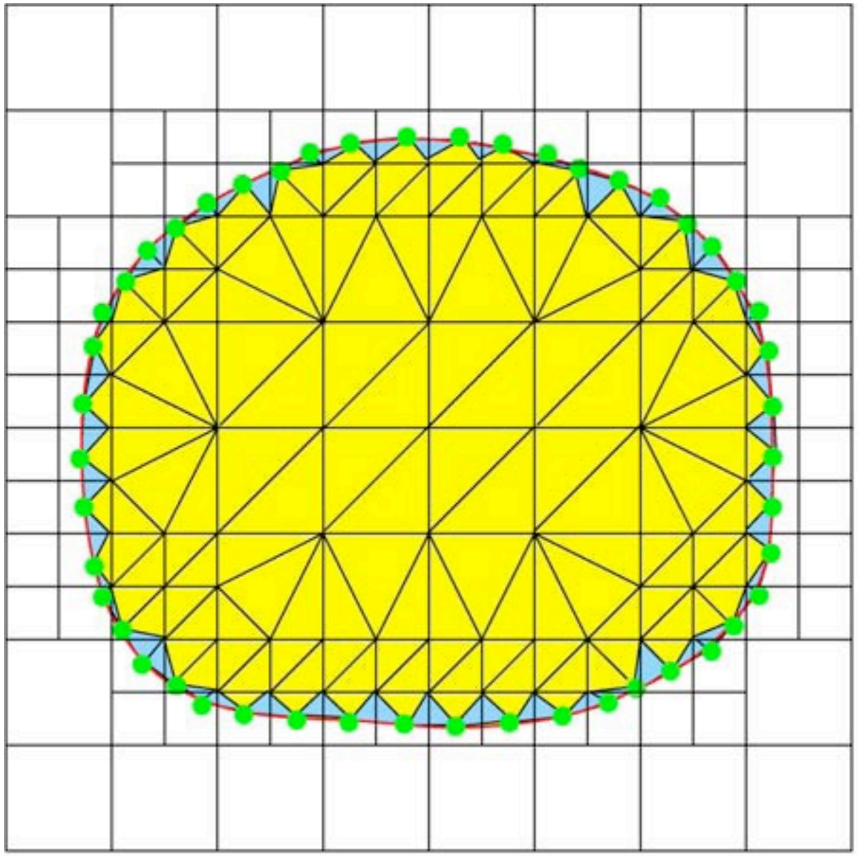


Case Table – interior cell (3D)

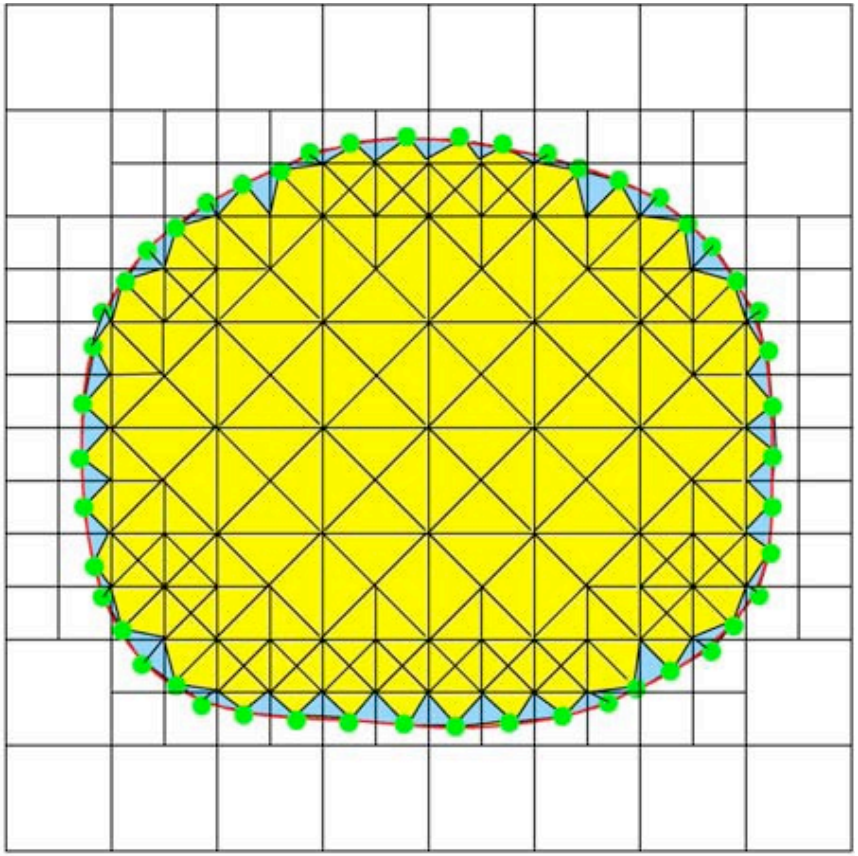
Isocontour-based Tri/Tet Meshing



Another Way for Triangle Mesh Generation



Sign change edge, interior edge in boundary cell and interior cell are analyzed.

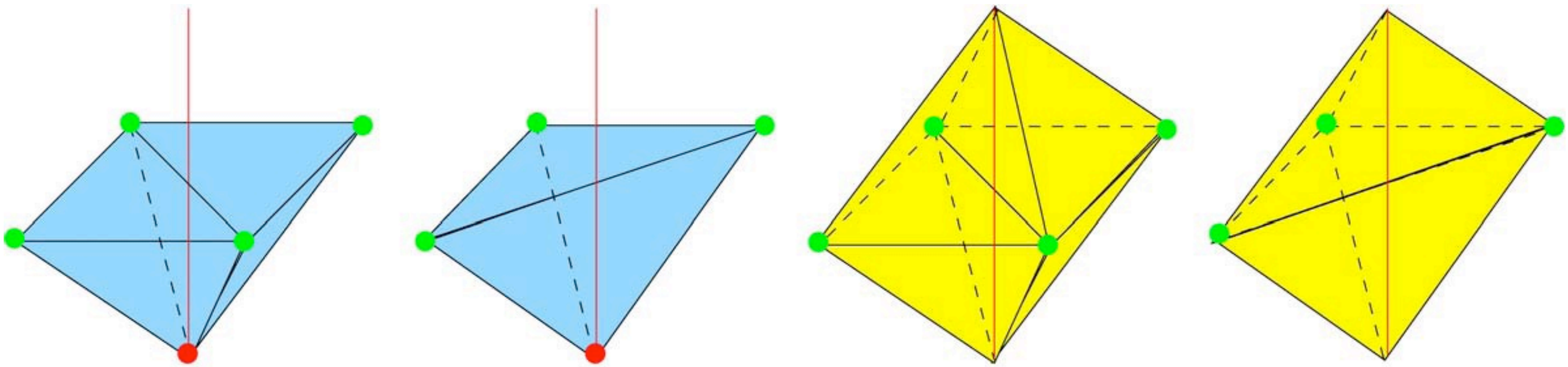


Only sign change edge and interior edge are analyzed.

Mechanical Engineering
Carnegie Mellon

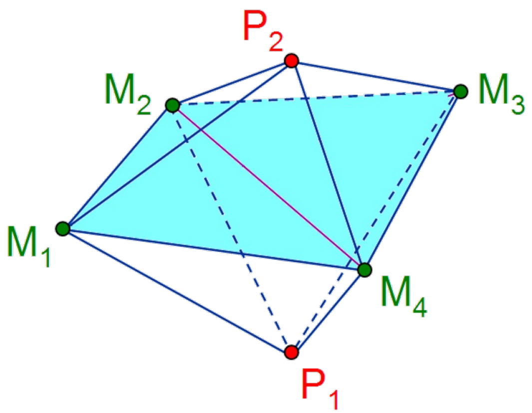
Another Way for Tetrahedral Mesh Generation

Only sign change edge and interior edge are analyzed.

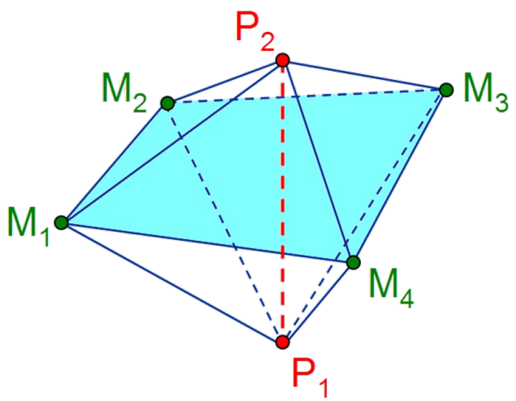


(a) (b) (c) (d)

Sign change edge Interior edge



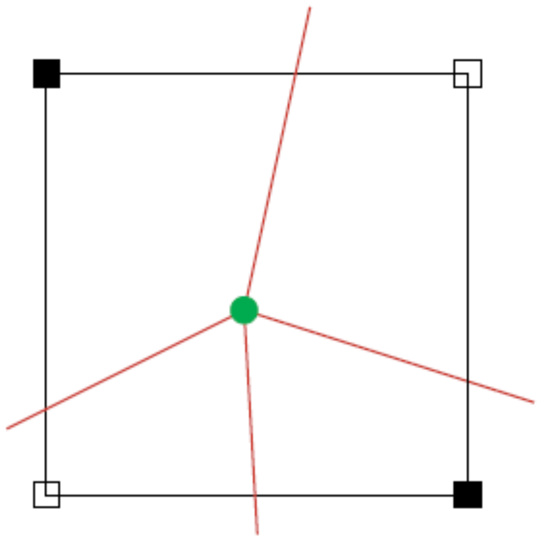
Method 1 -- split a diamond into tets



Method 2 -- split a diamond into tets (robust)

Mesh Topology

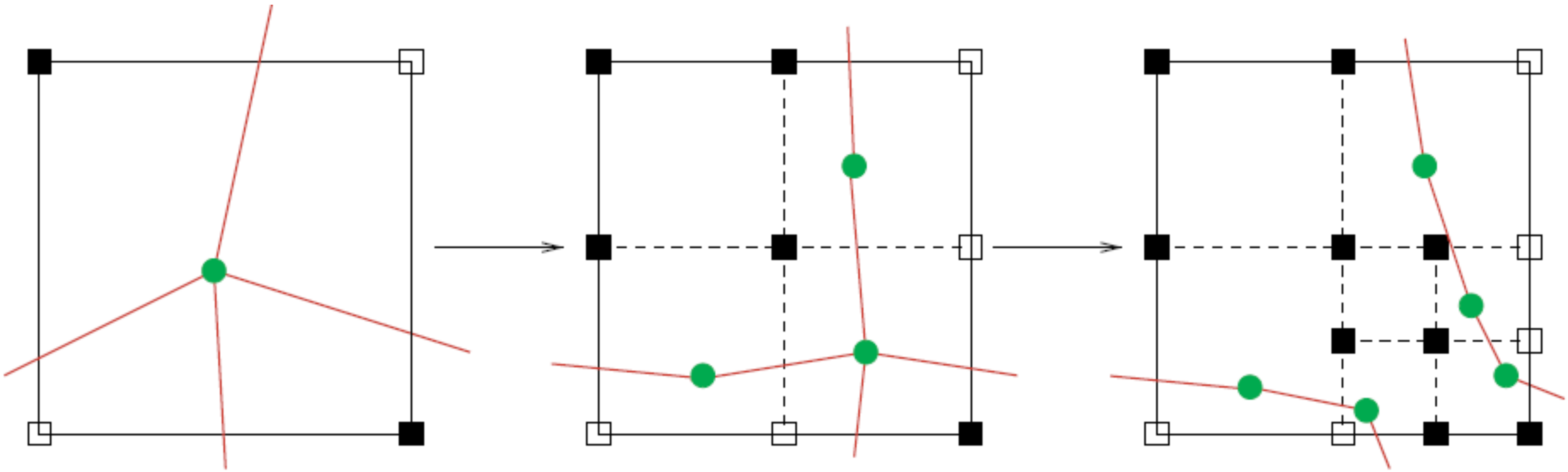
- Constructing an adaptive 3D mesh with correct topology plays an important role in accurate and efficient finite element calculations.
- Assume the function F within a cubic cell is defined by the trilinear interpolation of the eight vertex values. The sign of a vertex is defined to be positive when its value is greater than or equal to an isovalue, and negative when its value is less than the isovalue. An isosurface within a cube, defined as $F(x, y, z) = c$, may have different local topology depending on the configuration of signs at the eight vertices and the isovalue.
- Ambiguity: If the cube can be collapsed into an edge, then the cube has no ambiguity and the topologically correct dual contour is generated in the cube.



Mechanical Engineering
Carnegie Mellon

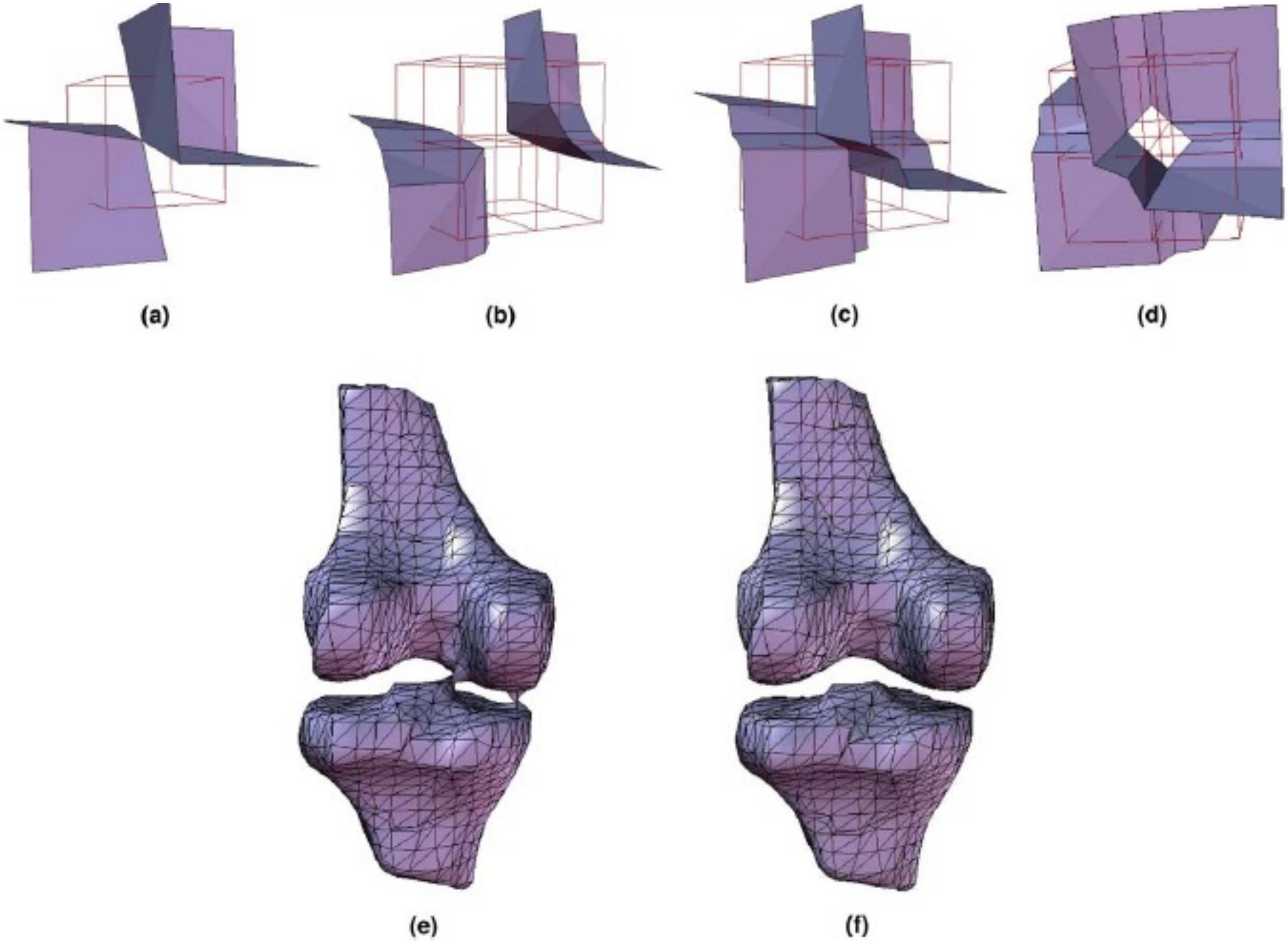
Mesh Topology

- We use the recursive cell subdivision in the finest level to reconstruct a dual contour with correct topology when the cell contains a non-manifold dual contour.
- The subdivision algorithm is very similar with what we used for enforcing each cell to have at most one boundary isosurface.



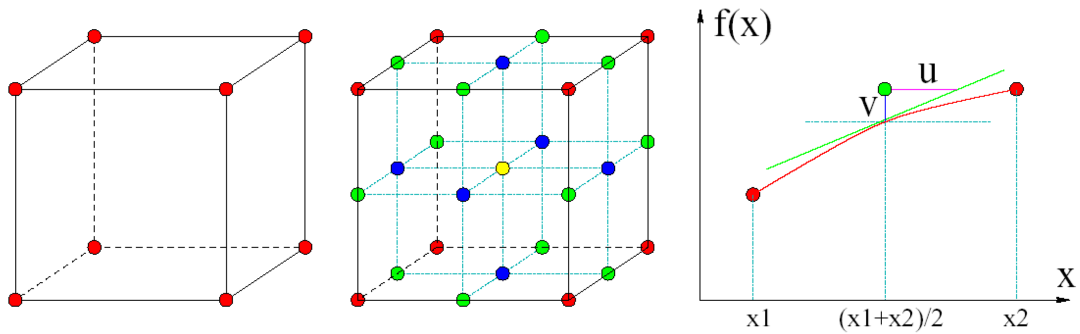
Mechanical Engineering
Carnegie Mellon

Mesh Topology



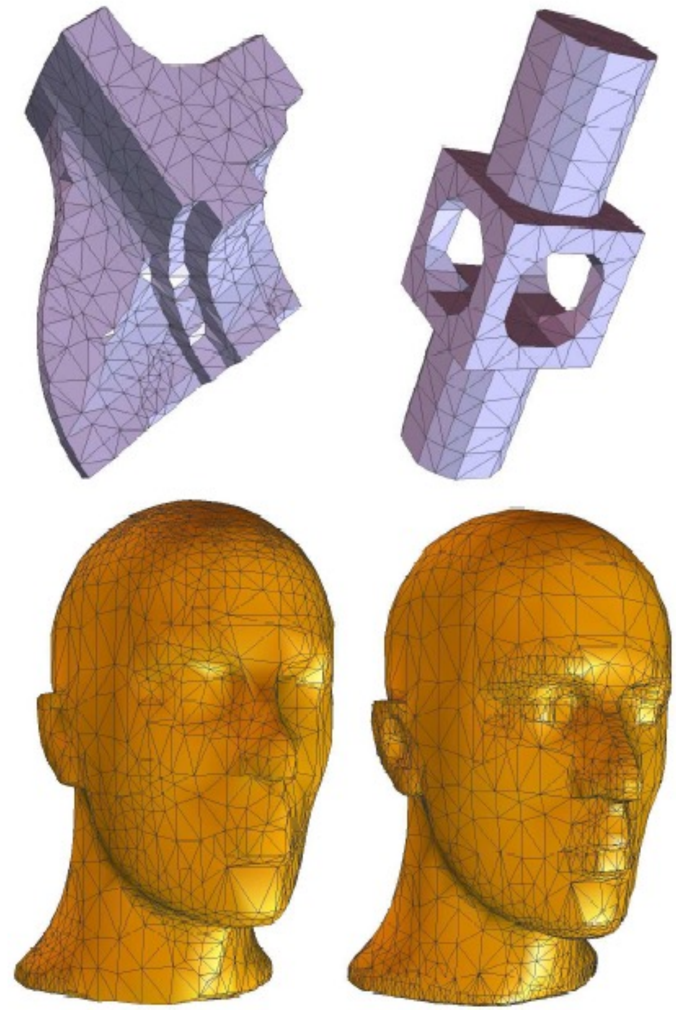
Tri/Tetra Meshing: Error Function

Feature Sensitive Error Metric: measuring the surface difference between two neighboring octree levels. It is used to decide where we should generate dense mesh, where we should keep coarse mesh.



$$
\begin{aligned}
f^i(x,y,z) &= f_{000}(1-x)(1-y)(1-z) + f_{011}(1-x)yz \\
&+ f_{001}(1-x)(1-y)z + f_{101}x(1-y)z \\
&+ f_{010}(1-x)y(1-z) + f_{110}xy(1-z) \\
&+ f_{100}x(1-y)(1-z) + f_{111}xyz
\end{aligned} \qquad (4)
$$

$$
error = \sum \frac{|f^{i+1} - f^i|}{|\nabla f^i|}
$$



Upper row – sharp features; Bottom row – facial features, left: QEF (2952 tris), right: EDerror (2734 tris).

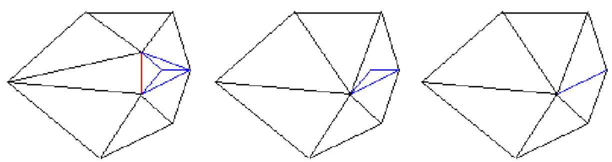
Mechanical Engineering
Carnegie Mellon

Edge-Contraction and Smoothing (Tri/Tet)

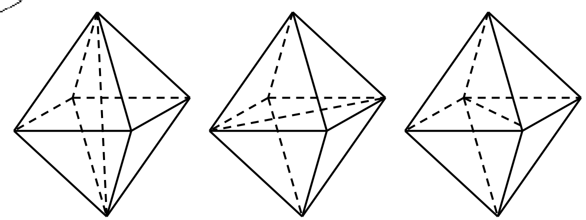
- Criteria:
  - Triangle: the aspect ratio = inscribed circle radius / circum-circle radius
  - Tetrahedron:
    - Edge-ratio
    - Joe-Liu parameter
    - Minimum volume bound

$$F(s,d) = \frac{f(s,d)}{g(s,d)} = \frac{2^{2(1-\frac{1}{d})} \times 3^{\frac{d-1}{2}} \times |s|^{\frac{2}{d}}}{\sum_{0\le i<j\le 3} |e_{ij}|^2}$$

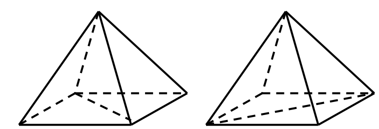
- Edge contraction is used to reduce the worst edge-ratio.



- Face/edge swapping



Diamond (interior edge)

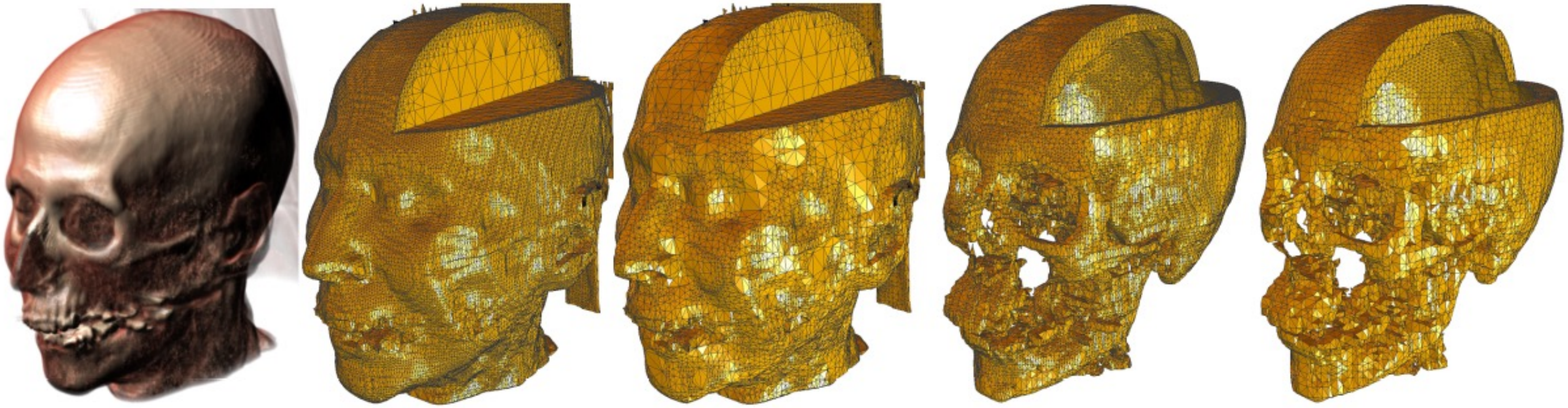


pyramid (sign change edge)

- Smoothing by relocating the node position
  - Interior node – use weighted averaging method (mass center)
  - Boundary node – use various nonlinear geometric flows to improve the quality of the surface. The discretized formula of the Laplace-Beltrami operator is used.

Summary: Isocontour-based Tri/Tet Meshing

- A top-down octree subdivision method
- Extending the dual contouring isosurface extraction method to crack-free interval volume tetrahedral meshing
- Generating adaptive and quality tri/tet meshes with the correct topology and feature sensitive adaptation
- Using edge-contraction and smoothing method to improve mesh quality

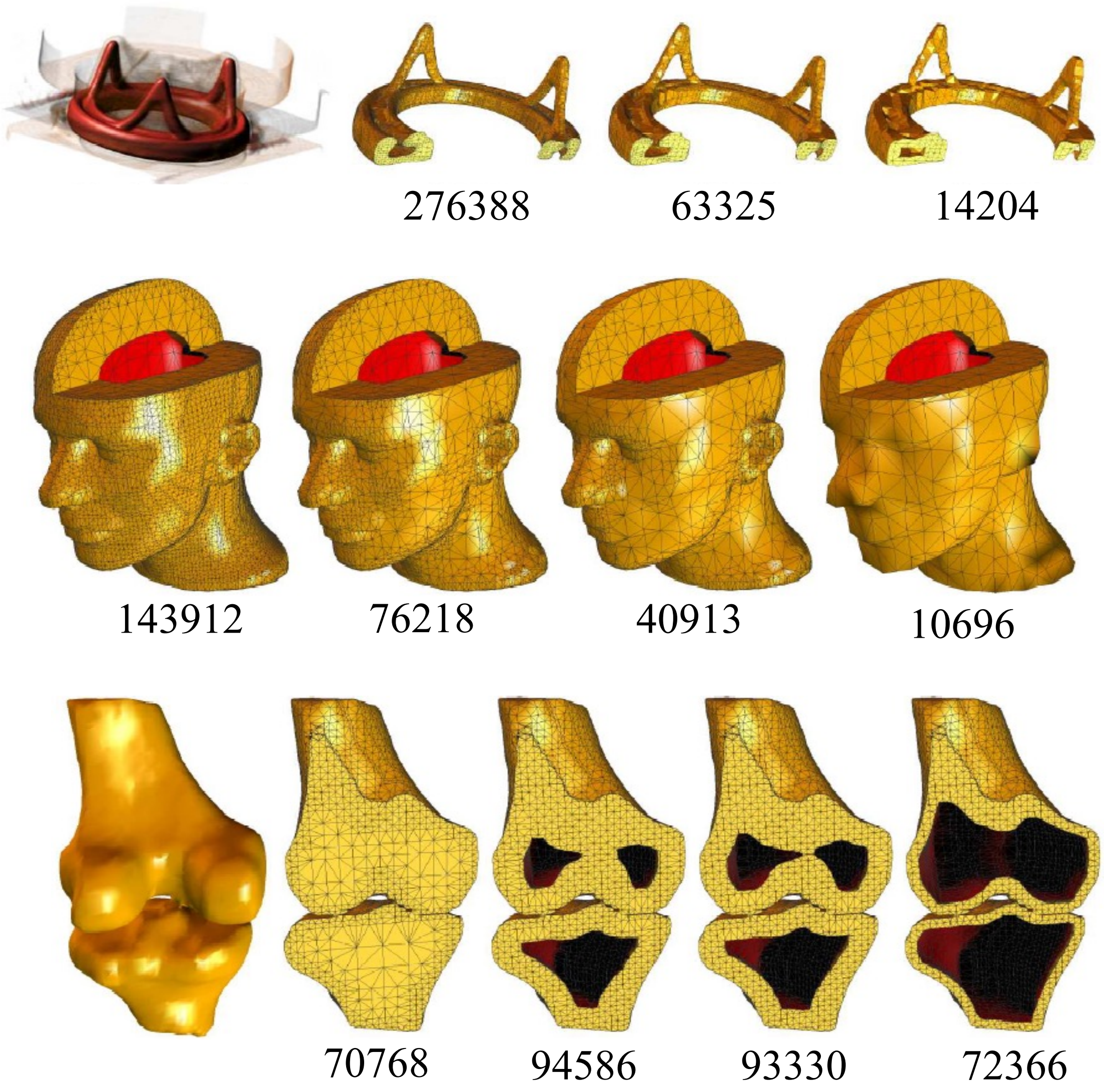


volume rendering (a) (b) (c) (d)

References:

- Y. Zhang, C. Bajaj, B-S. Sohn. **3D Finite Element Meshing from Imaging Data**. *The special issue of Computer Methods in Applied Mechanics and Engineering (CMAME) on Unstructured Mesh Generation*, 194(48-49):5083-5106, 2005.
- Y. Zhang, C. Bajaj, B-S. Sohn. **Adaptive and Quality 3D Meshing from Imaging Data**. *Proceedings of 8th ACM Symposium on Solid Modeling and Applications*, pp. 286-291. Seattle, WA. June 16-20, 2003.

Tri/Tet Meshing: Results



Isocontour-based Quad/Hex Meshing

We have extended the dual contouring method to quad/hex mesh generation. There are four main steps:

1. Starting Octree Level Selection - to guarantee the correct topology of boundary isosurfaces.

2. Adaptive Quad Meshing: The extracted uniform quad mesh can be made adaptive by local refinement.
   - How to decompose a quad into finer quads
   - How to calculate the new position of vertices

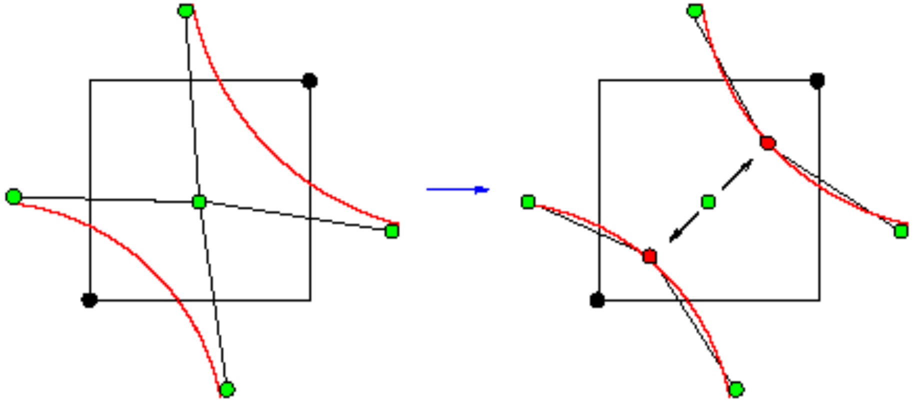
3. Adaptive Hex Meshing

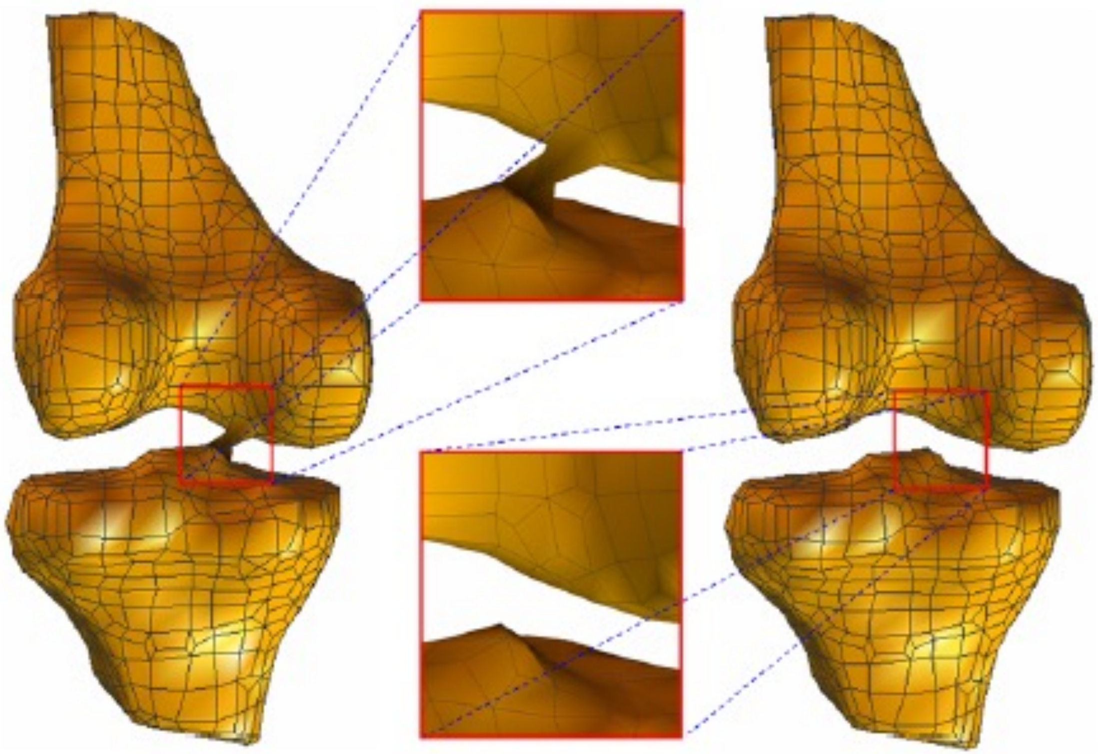
4. Quality Improvement

Step 1: Starting Octree Level Selection

- We guarantee the correct topology of boundary isosurfaces by choosing a suitable starting octree level.
- The bottom-up surface topology preserving octree-based algorithm is used to select a starting level.
- Topological equivalence checking [Ju et al. 2002] - the fine and coarse isocontour is topologically equivalent with each other if and only if the sign of the middle vertex of a coarse edge/face/cube is the same as the sign of at least one vertex of the edge/face/cube which contains the middle vertex.

Non-manifold situation





a non-manifold example

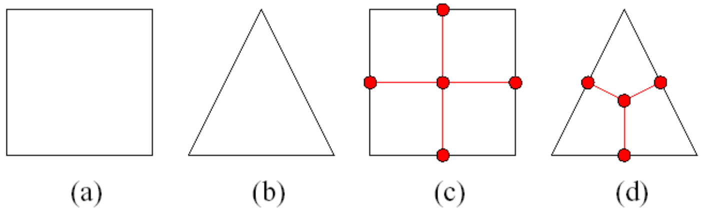
the non-manifold is removed

Step 2: Quad Isosurface Extraction

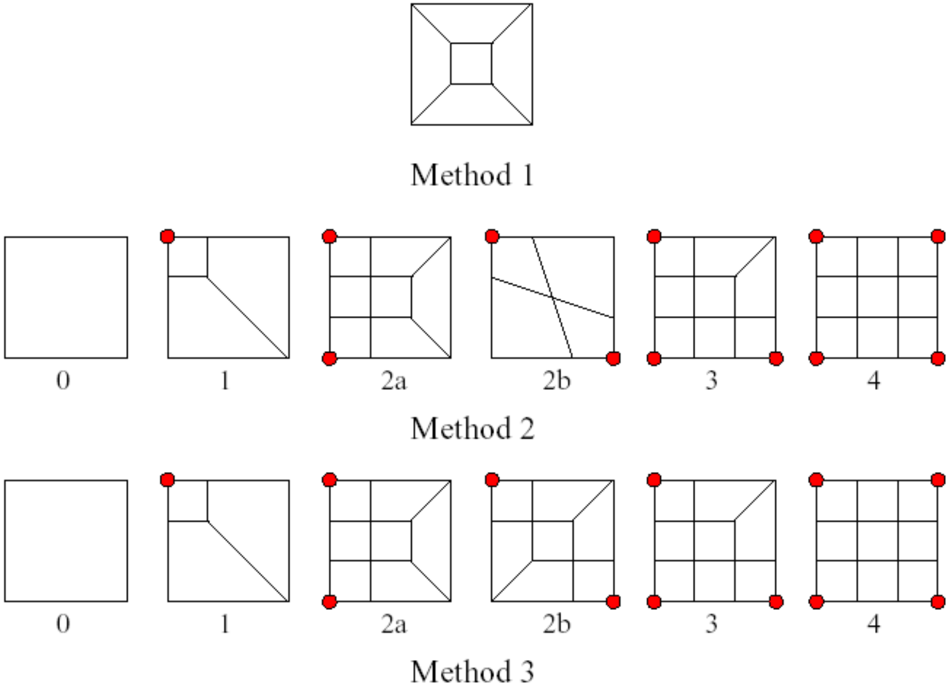
- The dual contouring method approximates isosurfaces with uniform quad meshes. Adaptive quad meshes are more preferable than uniform ones. The extracted uniform quad mesh can be made adaptive by local refinement. There are two problems in the adaptive quad mesh extraction.
  - How to decompose a quad into finer quads
  - How to calculate the new position of vertices

Step 2.1: Mesh Decomposition

- Indirect Method



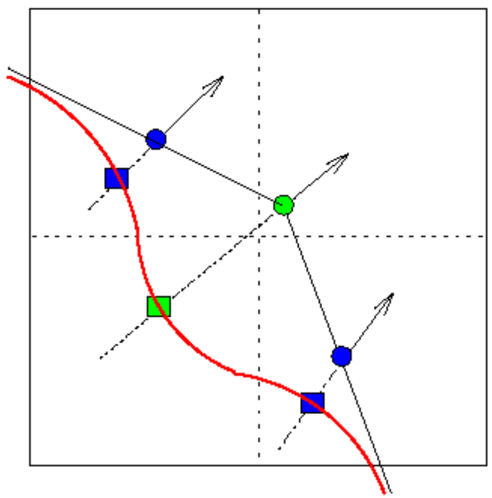
- Direct Method
  - All resulting elements are quads.
  - No hanging nodes exist.
  - The resulting mesh approximates the object surface accurately.
  - The resulting elements have good aspect ratio.
  - The resulting mesh introduces small number of new elements and vertices.



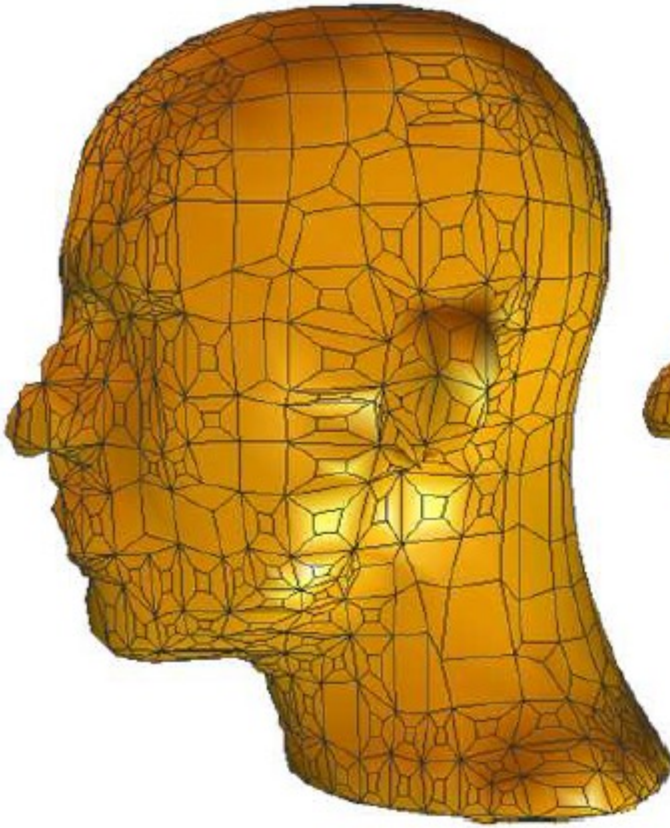
Method	Number of	0	1	2a	2b	3	4
2	elements	1	3	7	4	8	9
	vertices	0	3	8	4	10	12
3	elements	1	3	7	7	8	9
	vertices	0	3	8	8	10	12

Step 2.2: Vertex Position Calculation

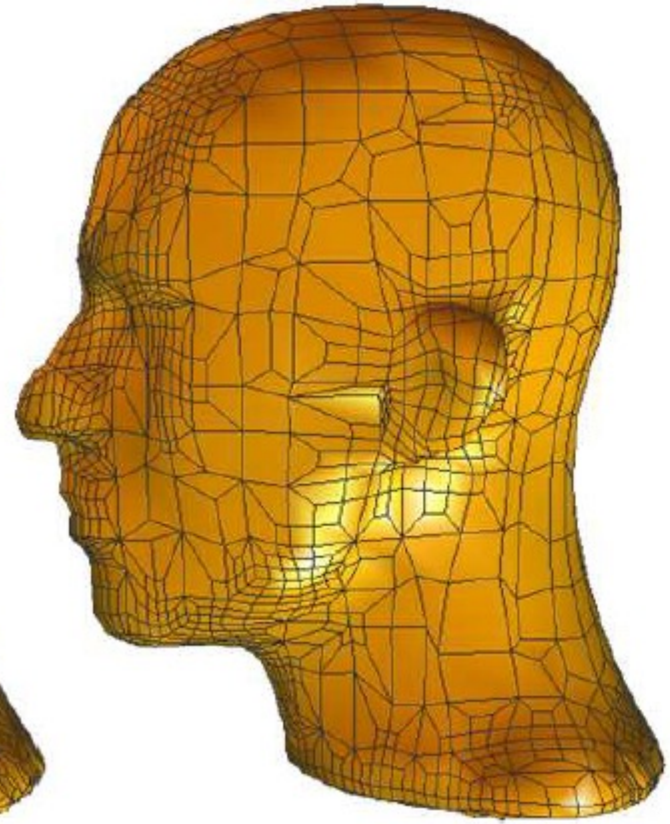
- Move the minimizer point toward the isosurface along its normal direction to find the intersection point.
- The isosurface is represented by a union of trilinear functions within the finest octree level.



Comparison of Various Methods



Direct Method (1)

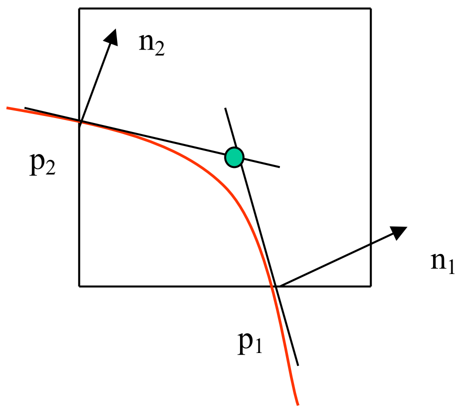


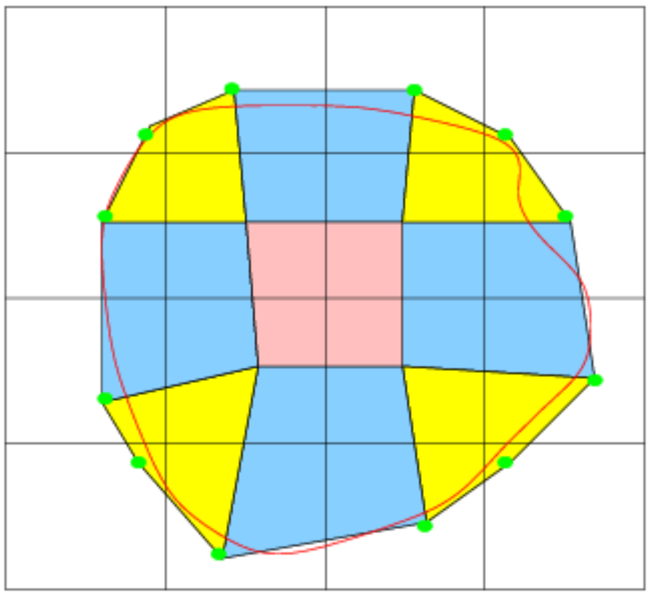
Direct Method (3)

Step 3: Hexahedral Mesh Extraction

- Uniform 2D quad - analyze each interior vertex, which is shared by four cells. One minimizer point is generated by minimizing QEF for boundary cells, the center point is chosen for interior cells.

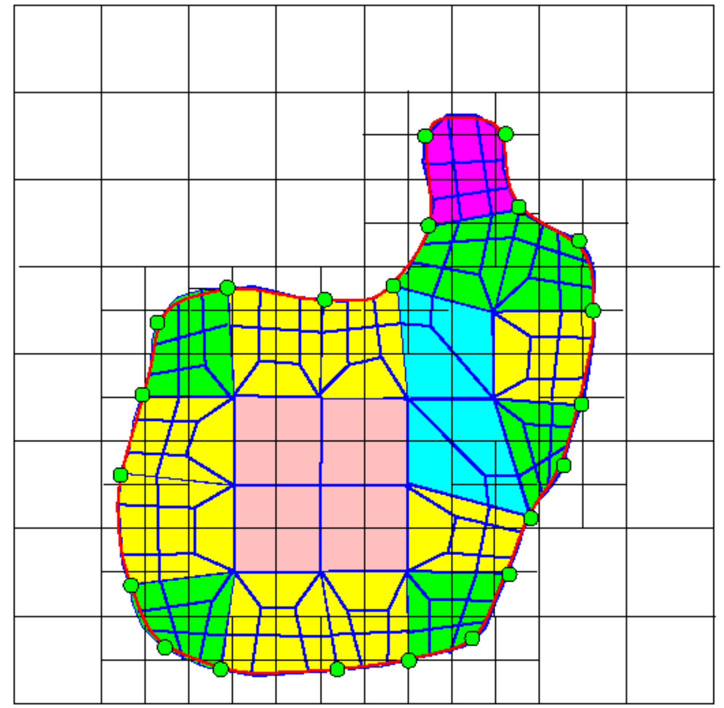
$$QEF[x] = \sum_i (n_i \cdot (x - p_i))^2$$

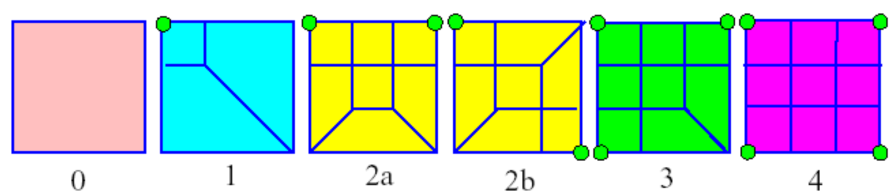




Step 3.1: 2D Decomposition

- It is important to choose a suitable starting level in order to guarantee the correct topology in the uniform mesh.
- Adaptive meshes are generated using templates.

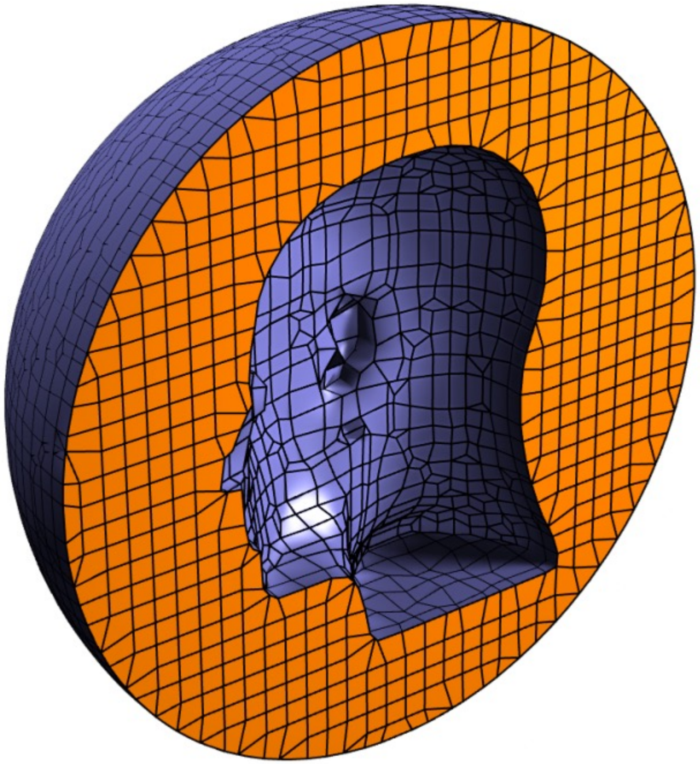




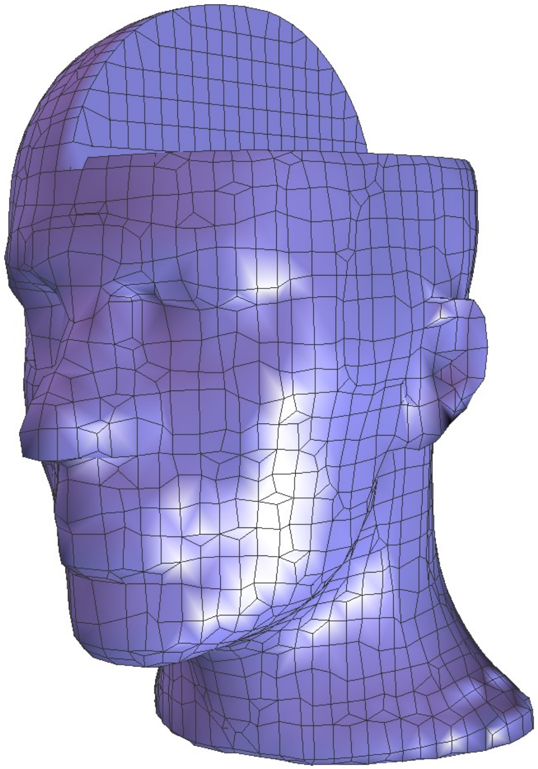
Adaptive quad

Uniform Hexahedral Mesh Extraction

- Analyze each interior vertex, which is shared by eight cells. One minimizer point is generated by minimizing QEF for boundary cells, the center point is chosen for interior cells.



An exterior mesh (13,552 hex)



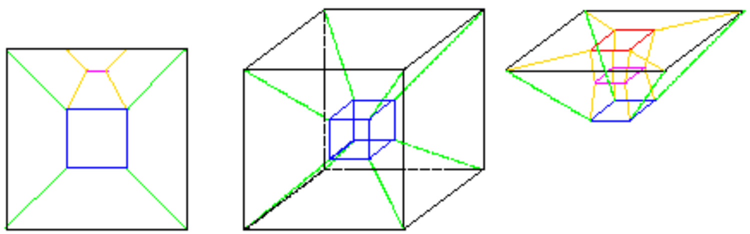
An interior mesh (23,204 hex)

References:

- Y. Zhang, C. Bajaj, B-S. Sohn. **3D Finite Element Meshing from Imaging Data**. *The special issue of Computer Methods in Applied Mechanics and Engineering (CMAME) on Unstructured Mesh Generation*, 194(48-49):5083-5106, 2005.
- Y. Zhang, C. Bajaj, B-S. Sohn. **Adaptive and Quality 3D Meshing from Imaging Data**. *Proceedings of 8th ACM Symposium on Solid Modeling and Applications*, pp. 286-291. Seattle, WA. June 16-20, 2003.

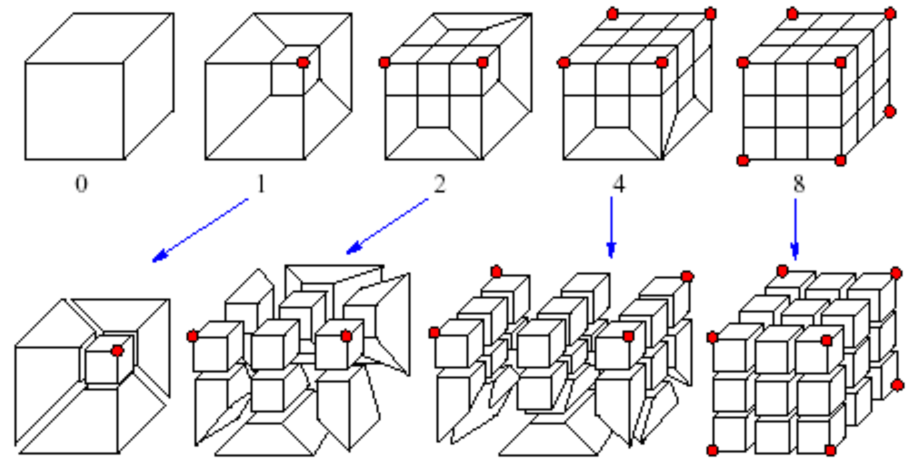
Step 3.2: 3D Decomposition

- Indirect Method: generate tetrahedral mesh first, then split each tet into hexes.
- Direct Method: directly split each hex into smaller hexes.



No. of elements = 6*i*+(6-*i*)+1 = 5*i*+7

No. of vertices = 8(*i*+1)



Method	Number of	0	1	2	4	8
2	elements	1	4	11	22	27
	vertices	0	7	19	39	56

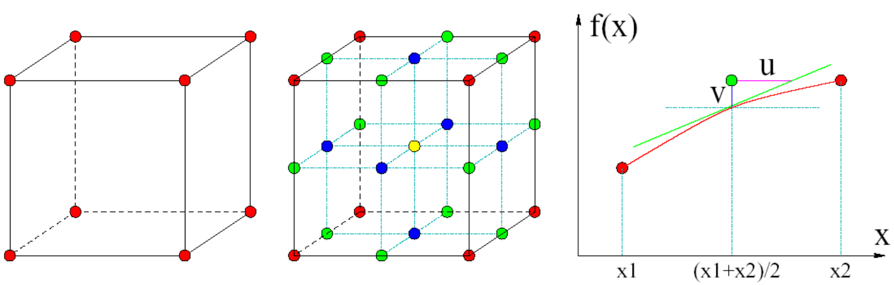
Method 1

Method 2

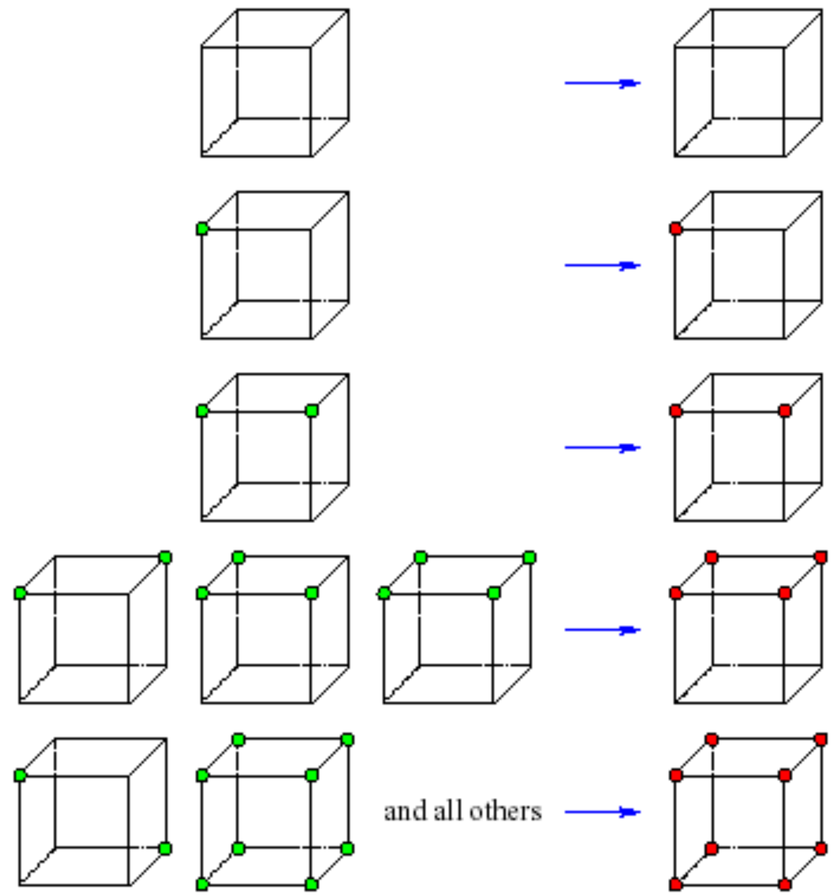
Mechanical Engineering
Carnegie Mellon

Look-Up Table

1. Set a sign for each cell in the uniform level indicating if this cell needs to be refined or not.
2. Check each hexahedron if it belongs to one of five basic templates. If not, Keep updating the sign using the Look-Up table.
3. Construct adaptive hex meshes.

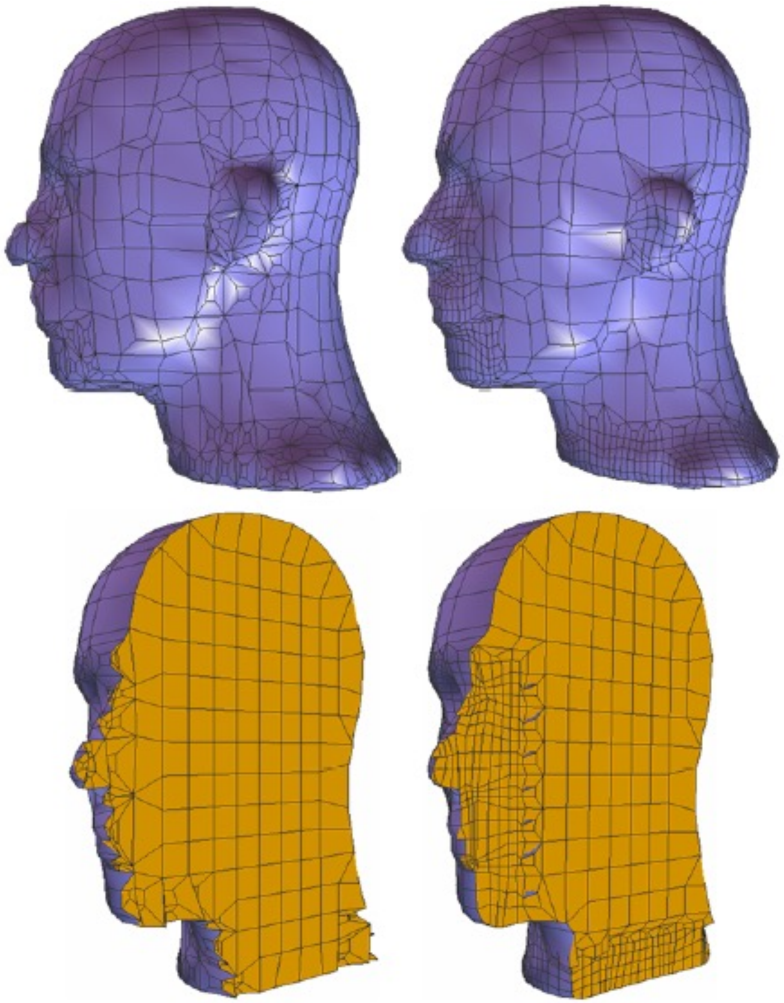


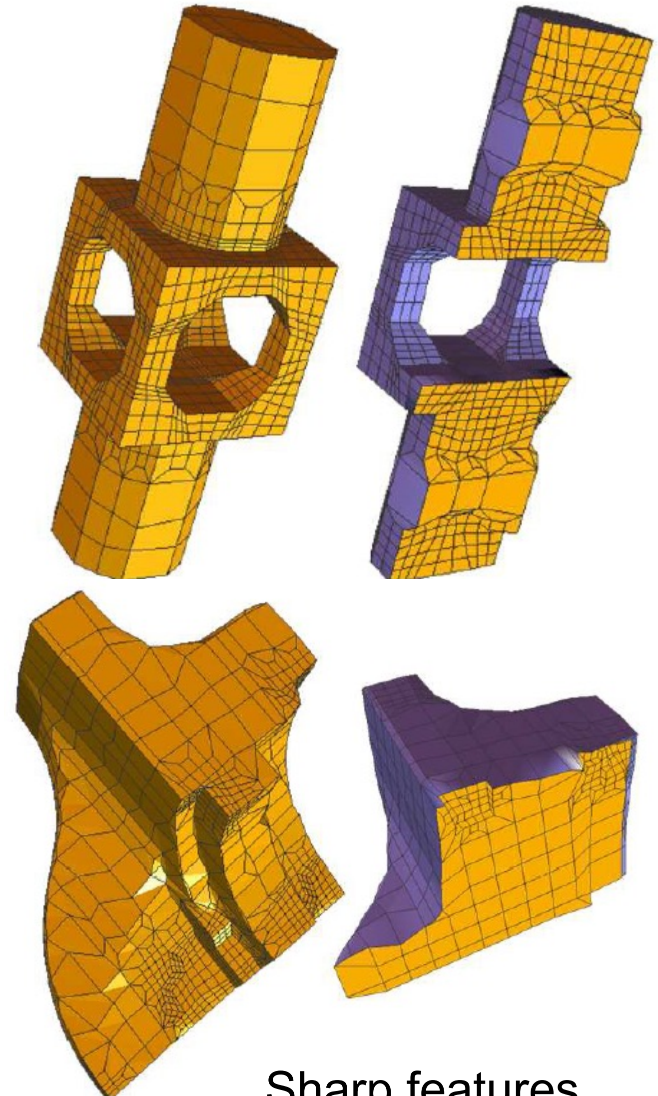
$$
\begin{aligned}
f^i(x,y,z) &= f_{000}(1-x)(1-y)(1-z) + f_{011}(1-x)yz \\
&+ f_{001}(1-x)(1-y)z + f_{101}x(1-y)z \\
&+ f_{010}(1-x)y(1-z) + f_{110}xy(1-z) \\
&+ f_{100}x(1-y)(1-z) + f_{111}xyz
\end{aligned}
$$



$$
error = \sum \frac{|f^{i+1} - f^i|}{|\nabla f^i|}
$$

Adaptive Hexahedral Meshes





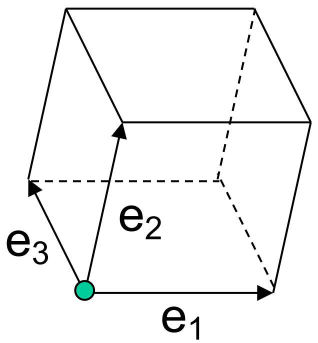
Sharp features

Mesh Adaptation

- There are three main ways to control the mesh adaptation. Users can also design an error function based on their specific requirements.
  - Feature sensitive error function
  - Various areas users are interested in
  - Finite element solutions
  - User defined

Step 4: Optimization Method

- Three error metrics [Kober et. al. 2000] [Knupp 2000] [Oddy et. al. 1988]
  Edge vector $e_i = x_i - x,\ i = 1,\ \ldots,\ m$
  Jacobian matrix $J = [e_1\, e_2\, _{\ldots}\, e_m]$



$$
\begin{aligned}
Jacobian(x) &= det(J) = \sqrt{det(J^T J)} && (1)\\
\kappa(x) &= \frac{1}{m}|J^{-1}||J| && (2)\\
Oddy(x) &= \frac{(|J^T J|^2 - \frac{1}{m}|J|^4)}{det(J)^{\frac{4}{m}}} && (3)
\end{aligned}
$$

- Quality metric in Finite Element Method [Finite Elements, Oden 1981]:

$$
x = \sum_{i=1}^{8} x_i \Phi_i \qquad y = \sum_{i=1}^{8} y_i \Phi_i \qquad z = \sum_{i=1}^{8} z_i \Phi_i \qquad \mathrm{J} = \begin{vmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} & \frac{\partial x}{\partial \zeta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} & \frac{\partial y}{\partial \zeta} \\ \frac{\partial z}{\partial \xi} & \frac{\partial z}{\partial \eta} & \frac{\partial z}{\partial \zeta} \end{vmatrix} \qquad |\,\mathrm{J}\,| = \det \mathrm{J}
$$

- Set an object function, use the conjugate gradient method to find an optimized position for a node with the worst condition number.

Summary: Isocontour-based Quad/Hex Meshing

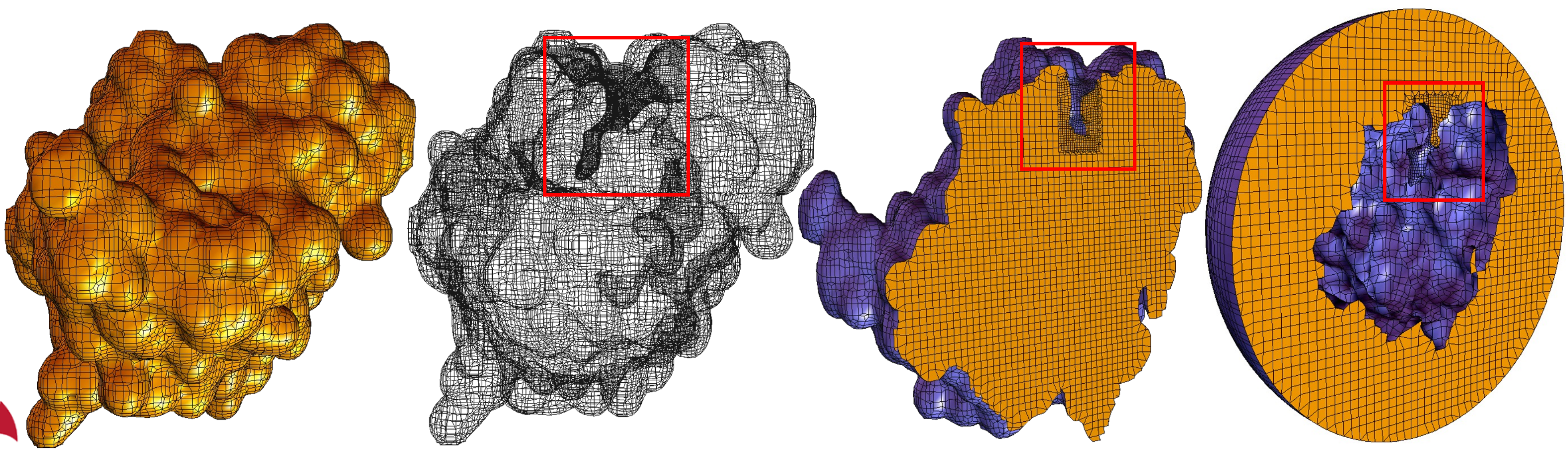
- A top-down octree-based method
- Extending the dual contouring method to adaptive and quality quad/hex meshing
  - The selection of the starting octree level to guarantee the correct topology
  - Crack-free and adaptive quad/hex meshing without hanging nodes
  - Using the optimization method to improve mesh quality

* **Mesh adaptivity**:
1. Feature sensitive error function
2. Various areas users are interested in
3. Finite element solutions
4. User defined

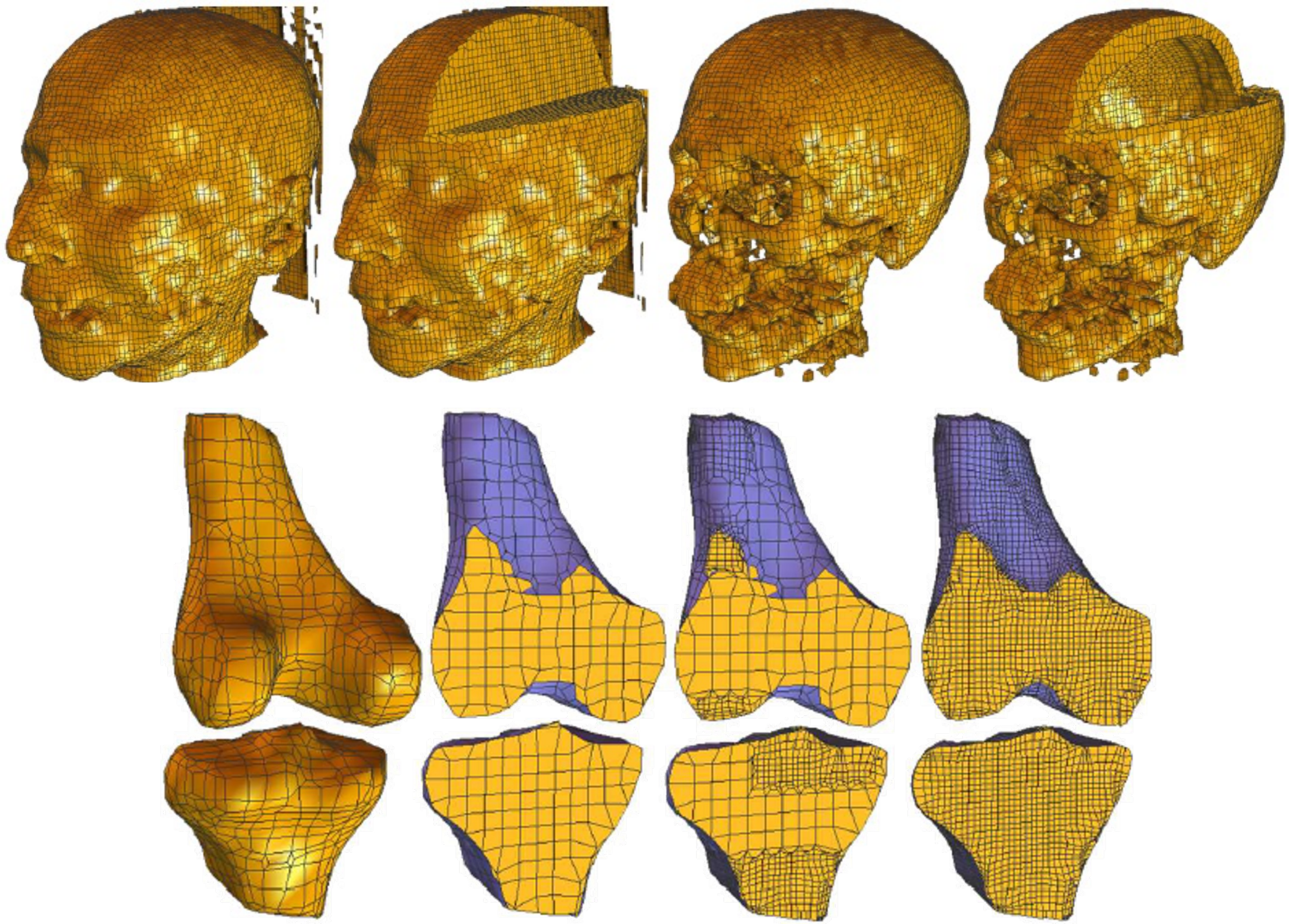
References:

• Y. Zhang, C. Bajaj. **Adaptive and Quality Quadrilateral/Hexahedral Meshing from Volumetric Data**. *Computer Methods in Applied Mechanics and Engineering (CMAME)*, 195(9-12):942-960, 2006.

•Y. Zhang, C. Bajaj. **Adaptive and Quality Quadrilateral/Hexahedral Meshing from Volumetric Data**. *Proceedings of 13th International Meshing Roundtable*. Willamsburg, VA. September 19-22, 2004.



Quad/Hex Meshing: Results

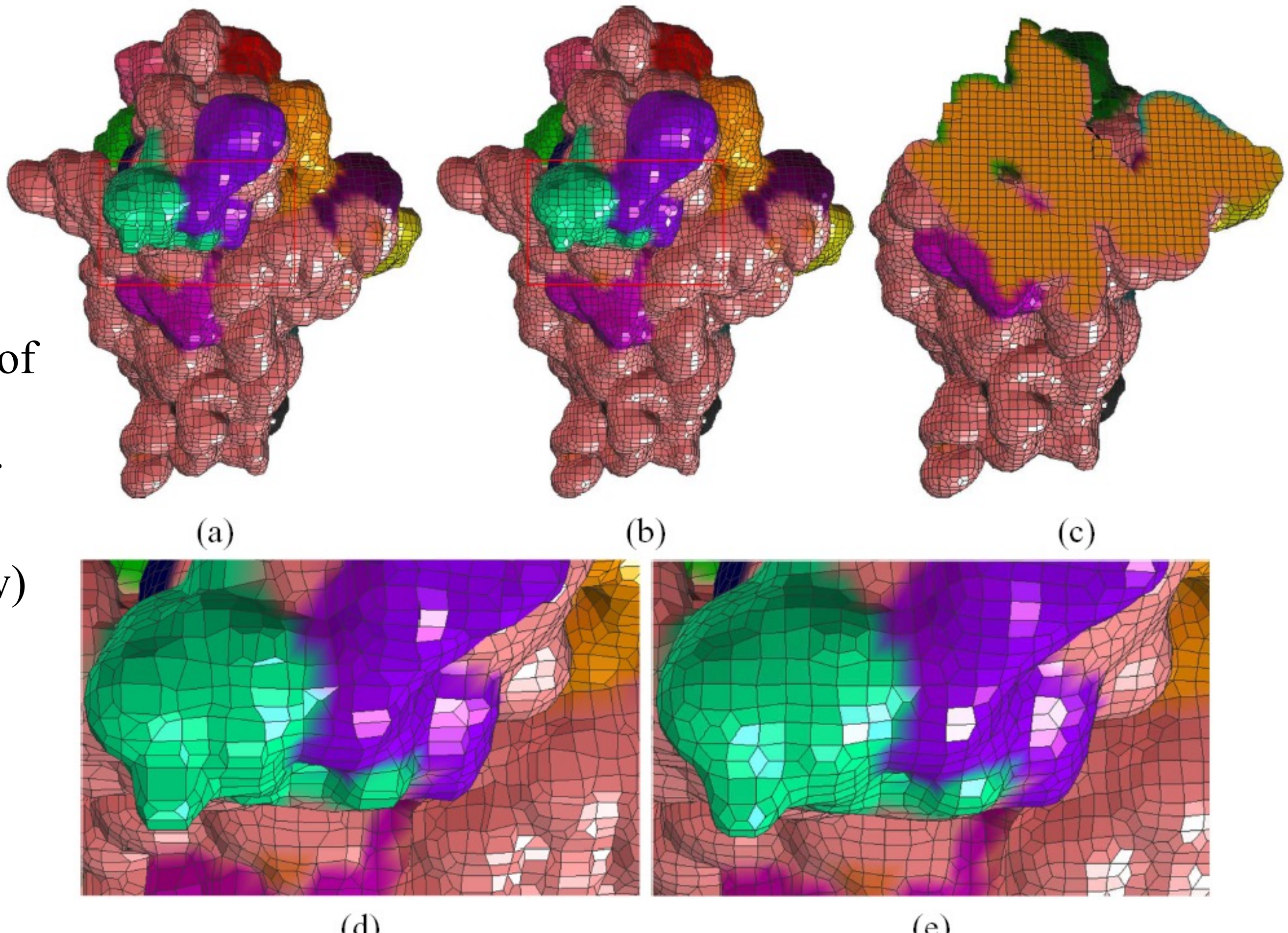


Geometric Flow

- The objective is to smooth the surface and improve the quality of quad/hex meshes with feature preserved using geometric flow.

Properties:

1. Noise removal
2. Feature preservation
3. Quality improvement of quad/hex meshes
4. Especially suitable for biomolecular meshes (surface diffusion flow)



Geometric Flow

- Nonlinear geometric PDEs have been used to efficiently solve surface modeling problems: surface blending, N-sided filling and free-form surface fitting. These nonlinear equations are discretized based on discrete differential geometry operators [Xu *et al.* 2003, 2004].

$$\frac{\partial x}{\partial t} = V_n(k_1, k_2, x)N(x), \quad M(0) = M_0.$$

where $x(t)$ – a surface point on a closed surface $M(t)$

$V_n(k_1, k_2, x)$ – the normal velocity of $M(t)$

$N(x)$ – the unit normal of the surface at $x(t)$

1. Mean Curvature Flow: $V_n = -H = -(k_1 + k_2)/2$ — area shrinking
2. Average Mean Curvature Flow: $V_n = h(t) - H(t)$ — volume preserving & area shrinking

   where $h(t) = \int_{M(t)} H d\sigma / \int_{M(t)} d\sigma$
3. Surface Diffusion Flow: $V_n = \Delta H$ — volume preserving & area shrinking
4. High Order Flow: $\frac{\partial x}{\partial t} = (-1)^{k+1}\Delta^k H N(x)$ — volume preserves if $K >= 1$

Mechanical Engineering
Carnegie Mellon

Review: Discretized LBO over Triangles

- Discretized Laplace-Beltrami operator (LBO) over triangles [Meyer *et al.* 2002]

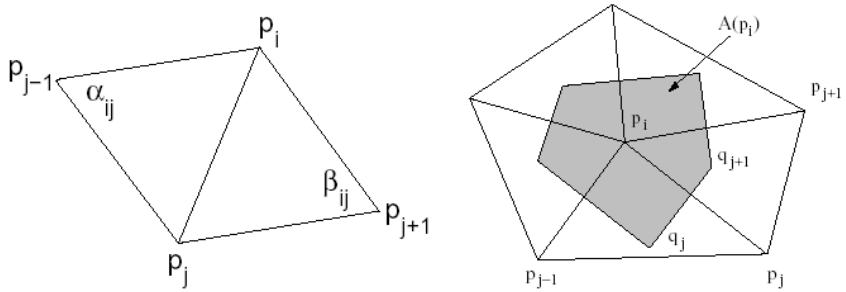


Fig 3.1: Left: The definition of the angles α_{ij} and β_{ij}. Right: The definition of the area $A(p_i)$.

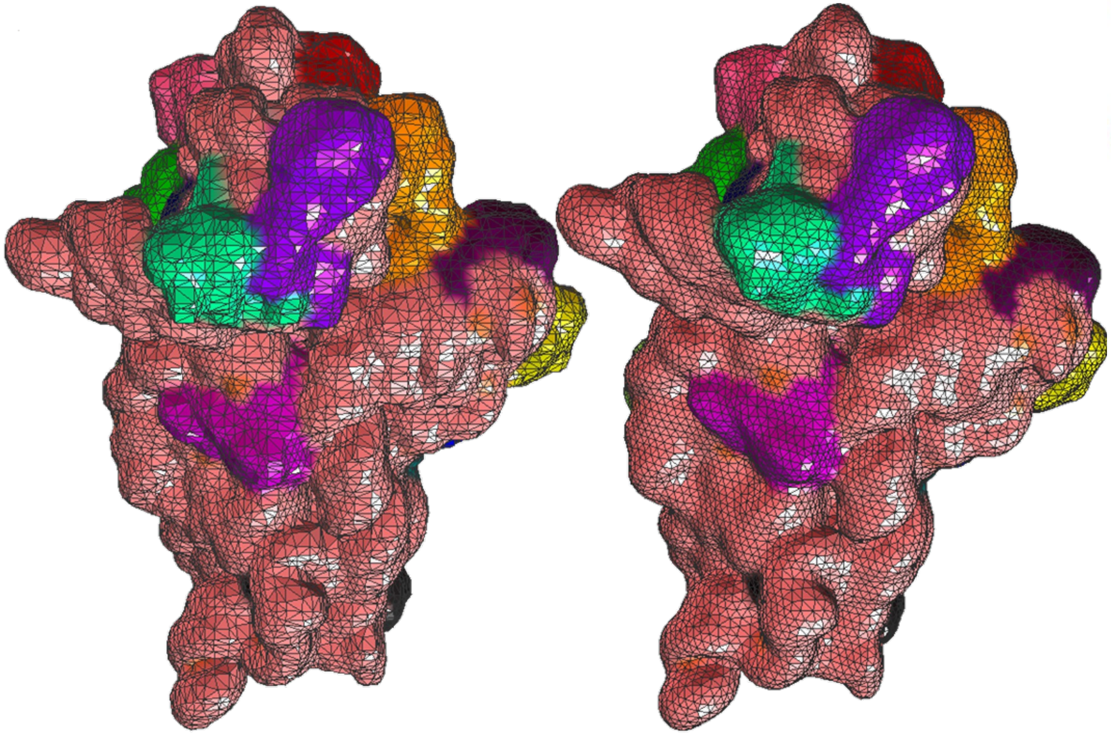
$$\Delta f(p_i) = \frac{1}{A(p_i)} \sum_{j \in N_1(i)} \frac{\cot \alpha_{ij} + \cot \beta_{ij}}{2} [f(p_j) - f(p_i)],$$

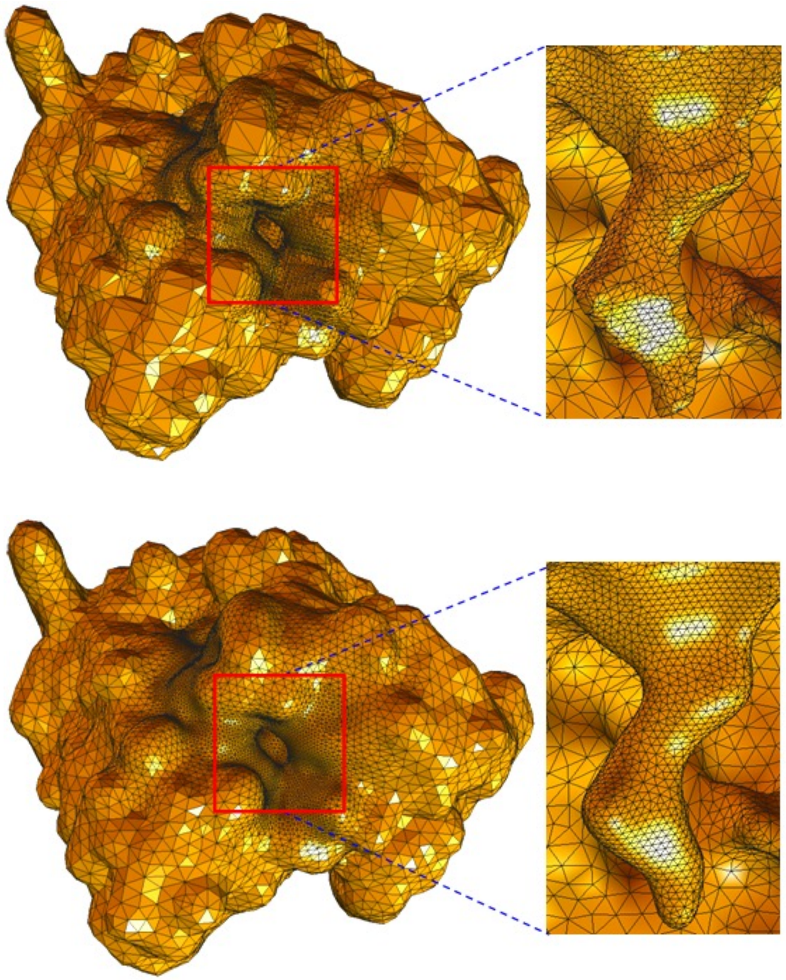
$$H(p_i)N(p_i) = \frac{1}{2A(p_i)} \sum_{j \in N_1(i)} \frac{\cot \alpha_{ij} + \cot \beta_{ij}}{2} (p_i - p_j)$$

$$\Delta^k f(p_i) = \Delta(\Delta^{k-1} f)(p_i) = \frac{1}{A(p_i)} \sum_{j \in N_1(i)} \frac{\cot \alpha_{ij} + \cot \beta_{ij}}{2} [\Delta^{k-1} f(p_j) - \Delta^{k-1} f(p_i)]$$

Review: Quality Improvement with LBO

- The discretized LBO was used in quality improvement of tri/tet meshes.





- Y. Zhang, G. Xu, C. Bajaj. Quality Meshing of Implicit Solvation Models of Biomolecular Structures. *ICES Technical Report 04-61, the University of Texas at Austin,* 2004.

Mechanical Engineering
Carnegie Mellon

Surface Diffusion Flow

- Surface diffusion flow

$$\frac{\partial x}{\partial t} = \Delta H(x)\mathbf{n}(x)$$

where Δ- the Laplace-Beltrami (LB) operator
H - the mean curvature
$\mathbf{n}(x)$ - the unit normal vector at the node x

Properties:

- Volume preserving and area shrinking
- Preserve sphere accurately if the initial mesh is embedded and close to a sphere.

Quad Area Calculation

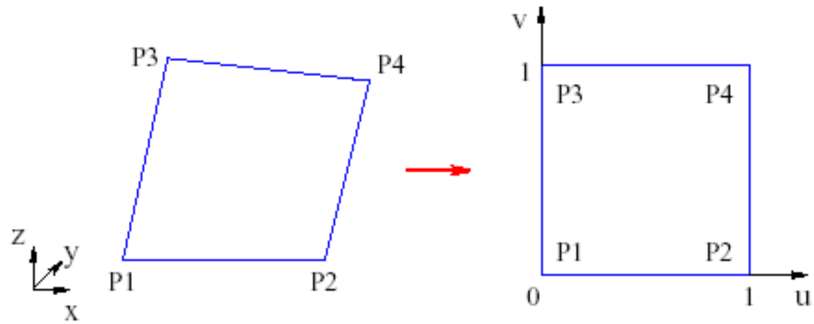


Fig. 2. A quad $[p_1 p_2 p_4 p_3]$ is mapped into a bilinear parametric surface.

$$S(u,v) = (1-u)(1-v)p_1 + u(1-v)p_2 + (1-u)vp_3 + uvp_4$$

The tangents:

$$S_u(u,v) = (1-v)(p_2 - p_1) + v(p_4 - p_3),$$
$$S_v(u,v) = (1-u)(p_3 - p_1) + u(p_4 - p_2).$$

The area:

$$A = \int_0^1 \int_0^1 \sqrt{\| S_u \times S_v \|^2} du dv = \int_0^1 \int_0^1 \sqrt{\| S_u \|^2 \| S_v \|^2 - (S_u, S_v)^2} du dv.$$

Four-point Gaussian quadrature rule:

$$\int_0^1 \int_0^1 f(u,v) du dv \approx \frac{f(q_1) + f(q_2) + f(q_3) + f(q_4)}{4}$$

$$q^- = \frac{1}{2} - \frac{\sqrt{3}}{6}, \qquad q^+ = \frac{1}{2} + \frac{\sqrt{3}}{6},$$

$$q_1 = (q^-, \ q^-), \qquad q_2 = (q^+, \ q^-),$$
$$q_3 = (q^-, \ q^+), \qquad q_4 = (q^+, \ q^+).$$

Discretized Laplace-Beltrami Operator (I)

$$\lim_{diam(R)\to 0} \frac{2\nabla A}{A} = \mathbf{H}(p)$$

where A is the area of a region R over the surface around the surface point p, $diam(R)$ denotes the diameter of the region R, and $\mathbf{H}(p)$ is the mean curvature normal.

$$\begin{aligned}
\nabla A &= \int_0^1 \int_0^1 \nabla \sqrt{\| S_u \|^2 \| S_v \|^2 - (S_u, S_v)^2}\, du dv \\
&= \int_0^1 \int_0^1 \frac{S_u(S_v, (v-1)S_v - (u-1)S_u))}{\sqrt{\| S_u \|^2 \| S_v \|^2 - (S_u, S_v)^2}} du dv \\
&+ \int_0^1 \int_0^1 \frac{S_v(S_u, (u-1)S_u - (v-1)S_v)}{\sqrt{\| S_u \|^2 \| S_v \|^2 - (S_u, S_v)^2}} du dv \\
&= \alpha_{21}(p_2 - p_1) + \alpha_{43}(p_4 - p_3) \\
&+ \alpha_{31}(p_3 - p_1) + \alpha_{42}(p_4 - p_2),
\end{aligned}$$

$$\alpha_{21} = \int_0^1 \int_0^1 \frac{(1-v)(S_v, (v-1)S_v - (u-1)S_u))}{\sqrt{\|S_u\|^2 \|S_v\|^2 - (S_u, S_v)^2}} du\, dv$$

$$\alpha_{43} = \int_0^1 \int_0^1 \frac{v(S_v, (v-1)S_v - (u-1)S_u))}{\sqrt{\|S_u\|^2 \|S_v\|^2 - (S_u, S_v)^2}} du\, dv$$

$$\alpha_{31} = \int_0^1 \int_0^1 \frac{(1-u)(S_u, (u-1)S_u - (v-1)S_v)}{\sqrt{\|S_u\|^2 \|S_v\|^2 - (S_u, S_v)^2}} du\, dv$$

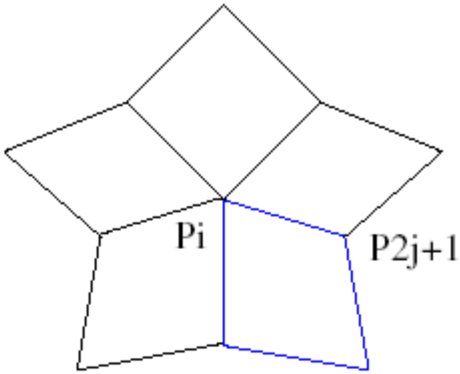
$$\alpha_{42} = \int_0^1 \int_0^1 \frac{u(S_u, (u-1)S_u - (v-1)S_v)}{\sqrt{\|S_u\|^2 \|S_v\|^2 - (S_u, S_v)^2}} du\, dv$$

$$\begin{aligned}
\nabla A &= \alpha_1 p_1 + \alpha_2 p_2 + \alpha_3 p_3 + \alpha_4 p_4 \\
&= \alpha_2(p_2 - p_1) + \alpha_3(p_3 - p_1) + \alpha_4(p_4 - p_1).
\end{aligned}$$

$$\alpha_1 = -\alpha_{21} - \alpha_{31}, \qquad \alpha_2 = -\alpha_{21} + \alpha_{42},$$
$$\alpha_3 = \alpha_{31} - \alpha_{43}, \qquad \alpha_4 = \alpha_{43} + \alpha_{42}.$$

Discretized Laplace-Beltrami Operator (II)

$$\mathbf{H}(p_i) \approx \frac{2}{A(p_i)} \sum_{j=1}^{n} [\alpha^i_j (p_{2j-1} - p_i)$$
$$+ \beta^i_j (p_{2j+1} - p_i) + \gamma^i_{j+1} (p_{2j} - p_i)]$$
$$= \sum_{k=1}^{2n} w^i_k (p_k - p_i)$$



$$w^i_{2j} = \frac{2\gamma^i_j}{A(p_i)}, \quad w^i_{2j-1} = \frac{2(\alpha^i_j + \beta^i_{j-1})}{A(p_i)}, \quad w^i_{2j+1} = \frac{2(\alpha^i_{j+1} + \beta^i_j)}{A(p_i)}.$$

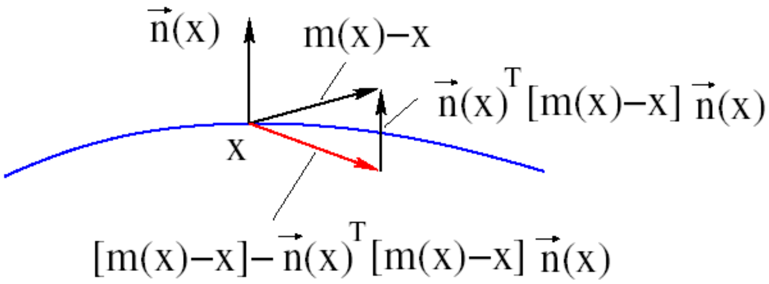
Using the relation $\Delta x = 2H(p_i)$ ([32], page 151), we obtain

$$\Delta f(p_i) \approx 2 \sum_{k=1}^{2n} w^i_k (f(p_k) - f(p_i)).$$

$$\Delta H(p_i)\mathbf{n}(p_i) \approx 2 \sum_{k=1}^{2n} w^i_k (H(p_k) - H(p_i))\mathbf{n}(p_i)$$
$$= 2 \sum_{k=1}^{2n} w^i_k \left[\mathbf{n}(p_i)\mathbf{n}(p_k)^T \mathbf{H}(p_k) - \mathbf{H}(p_i) \right]$$

Tangent Movement

$$\frac{\partial x}{\partial t} = \Delta H(x)\mathbf{n}(x) + v(x)\mathbf{T}(x)$$



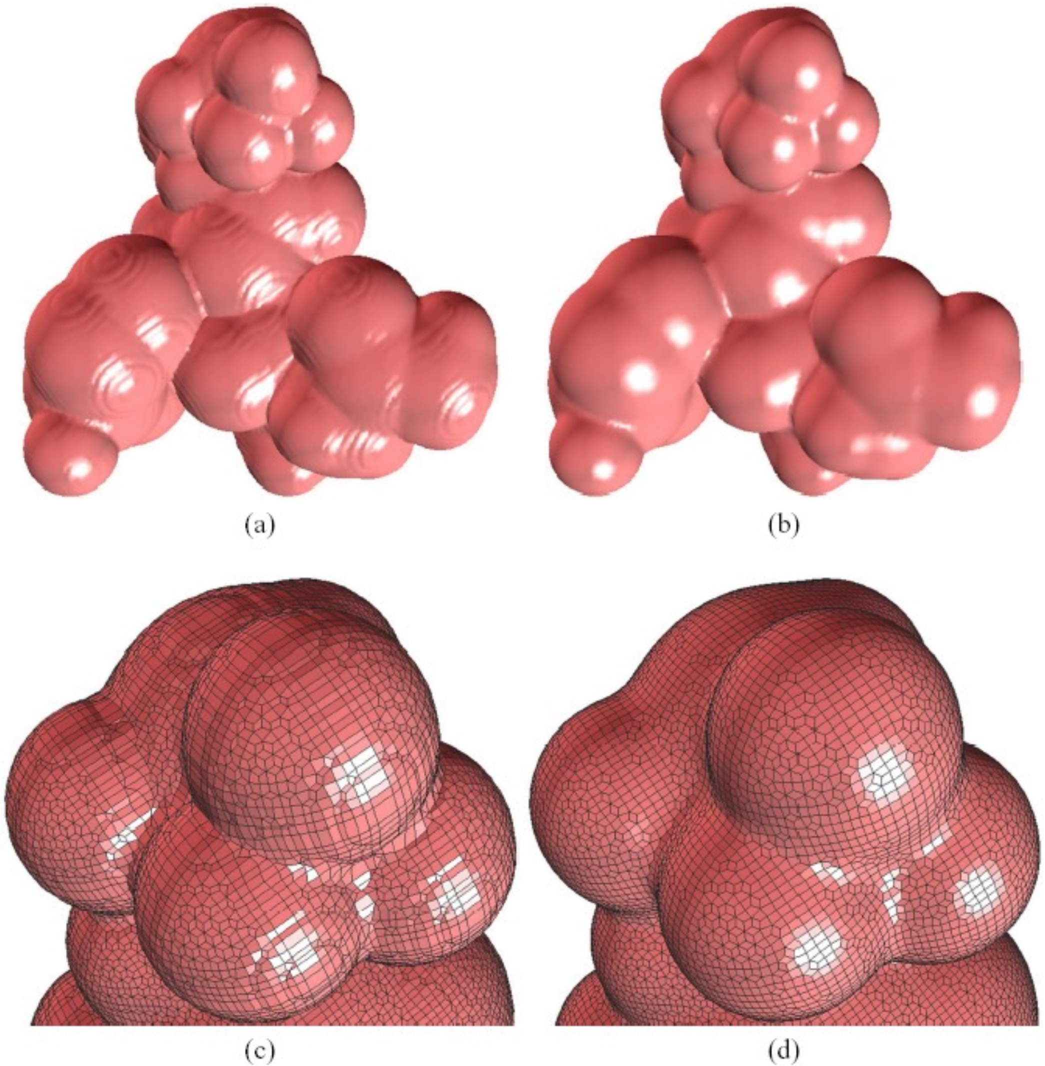
Mass center:

$$\int_S \| y - p \|^2 \, d\sigma = min$$

$$m(p_i) = \sum_{j=1}^{n} \left(\frac{p_i + p_{2j-1} + p_{2j} + p_{2j+1}}{4} A_j\right) / A^i_{total}$$

Temporal discretization:

- Semi-implicit Euler scheme
- Conjugate gradient iterative method



Discussion

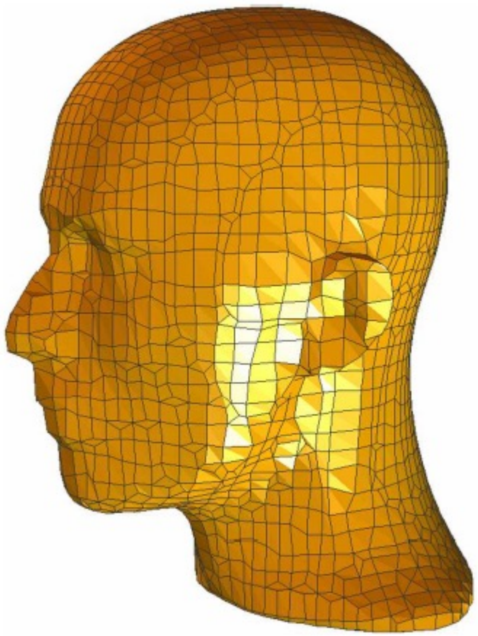
1. Vertex normal movement

    - The surface diffusion flow preserves volume
    - The surface diffusion flow preserves a sphere accurately if the initial mesh is embedded and close to a sphere.

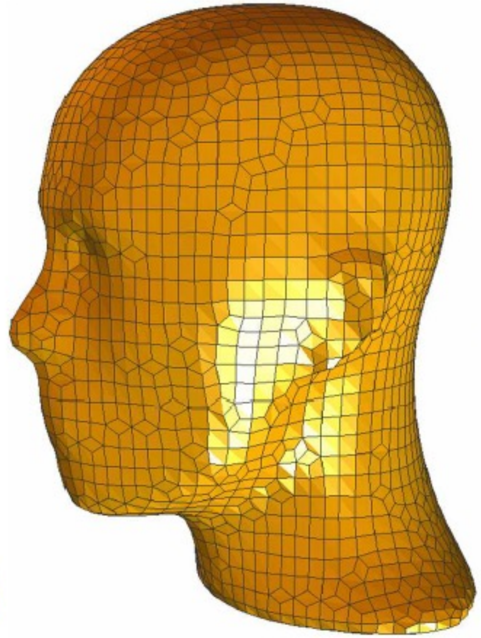
2. Vertex tangent movement

    - The tangent movement doesn't change the surface shape
    - Area-weighted averaging method, suitable for adaptive meshes.

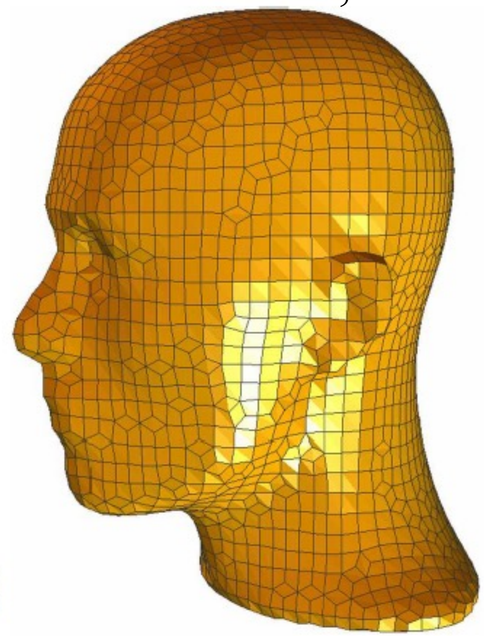
$\Delta t = 0.01$, 100 iterations



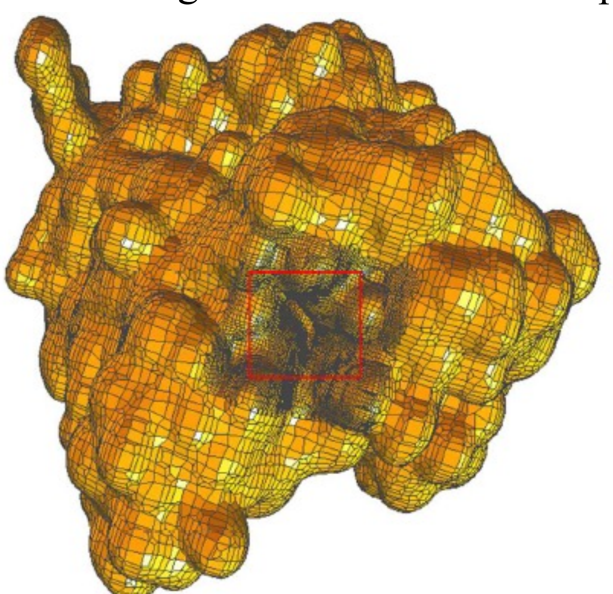
Original mesh



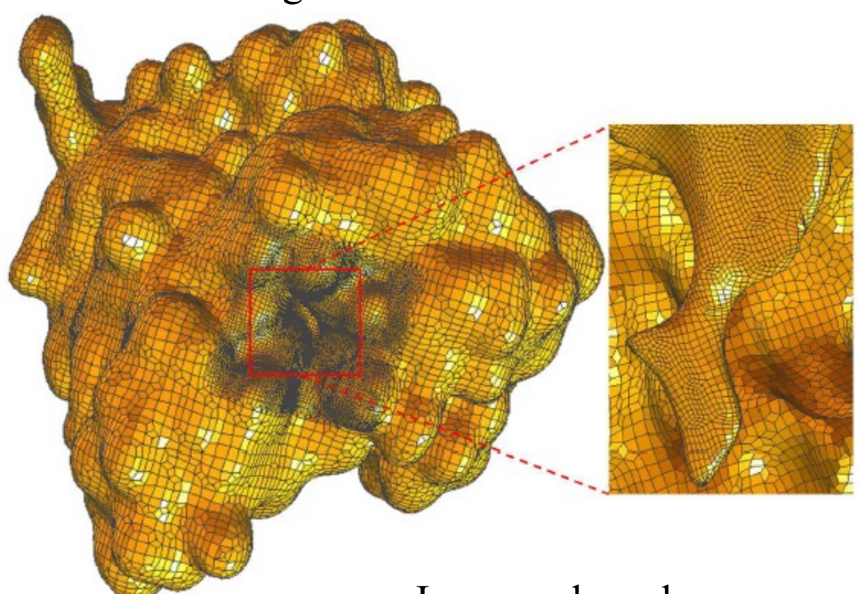
Laplacian smoothing



Surface Diffusion Flow

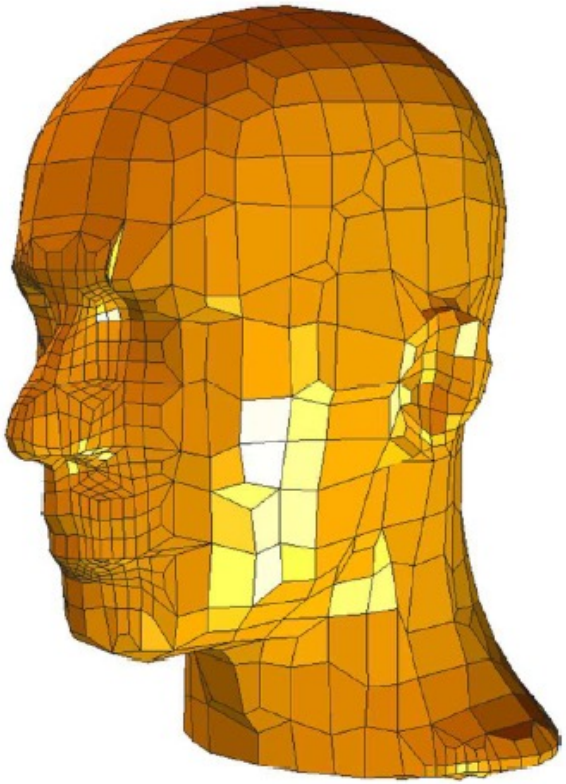


Original mesh



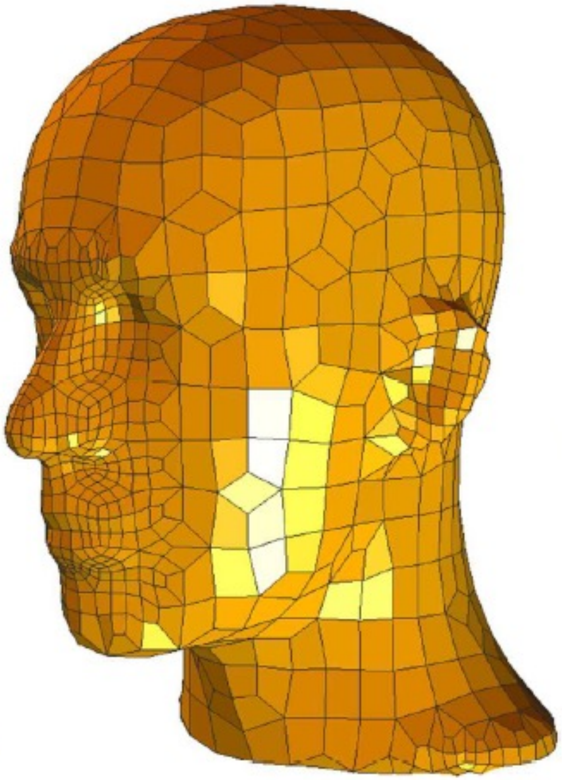
Improved mesh

Adaptive Meshes



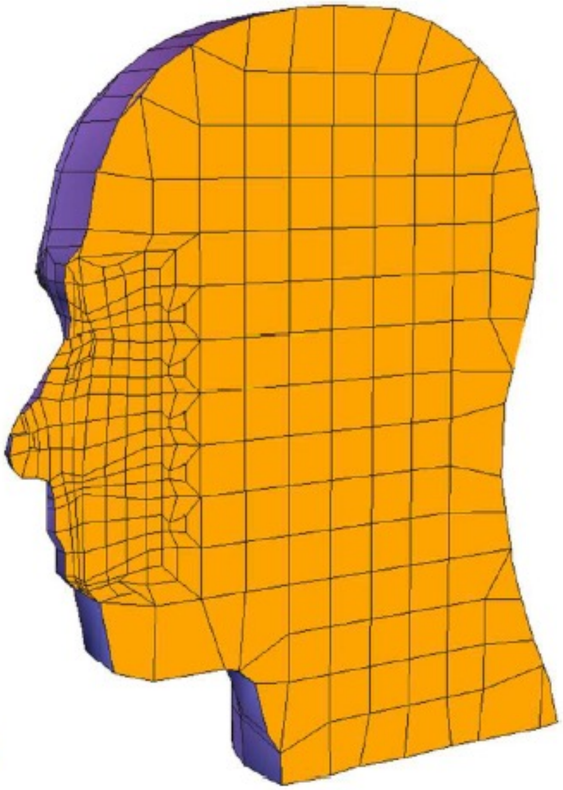
(a)

Original mesh



(b)

Improved mesh



(c)

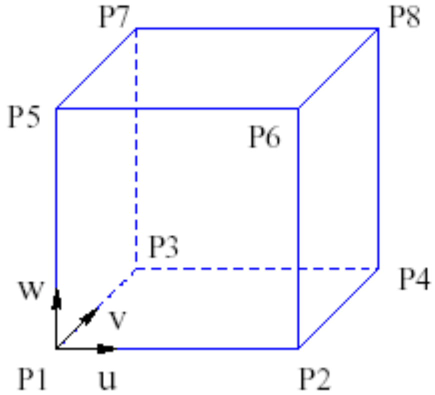
One cross-section of the improved mesh

Hexahedral Mesh

- Surface Vertex Movement
  - The dual contouring hex meshing method provides a boundary sign for each vertex and each face.
  - The boundary sign can also be decided by checking the connectivity information.
    1. A face is shared by two hexes – interior
    2. A face belongs to only one hex – boundary
    3. Four vertices of each boundary face is on the boundary
  - For each boundary vertex, find all its neighboring vertices and faces lying on the boundary for the mass center calculation.

Hexahedral Mesh

- Interior Vertex Movement



$$
\begin{aligned}
V(u,v,w) = {} & (1-u)(1-v)(1-w)p_1 \\
& + u(1-v)(1-w)p_2 + (1-u)v(1-w)p_3 \\
& + uv(1-w)p_4 + (1-u)(1-v)wp_5 \\
& + u(1-v)wp_6 + (1-u)vwp_7 \\
& + uvwp_8.
\end{aligned}
$$

$$
\begin{aligned}
V_u(u,v,w) &= (1-v)(1-w)(p_2-p_1) + v(1-w)(p_4-p_3) \\
& + (1-v)w(p_6-p_5) + vw(p_8-p_7), \\
V_v(u,v,w) &= (1-u)(1-w)(p_3-p_1) + u(1-w)(p_4-p_2) \\
& + (1-u)w(p_7-p_5) + uw(p_8-p_6), \\
V_w(u,v,w) &= (1-u)(1-v)(p_5-p_1) + u(1-v)(p_6-p_2) \\
& + (1-u)v(p_7-p_3) + uv(p_8-p_4).
\end{aligned}
$$

$$
V = \int_0^1 \int_0^1 \int_0^1 \sqrt{\bar{V}}\, dudvdw \qquad \bar{V} = \| (V_u \times V_v) \cdot V_w \|^2
$$

eight-point Gaussian quadrature rule

$$
\int_0^1 \int_0^1 \int_0^1 f(u,v,w)\, dudvdw \approx \frac{\sum_{j=1}^{8} f(q_j)}{8},
$$

$$
q^- = \frac{1}{2} - \frac{\sqrt{3}}{6}, \qquad q^+ = \frac{1}{2} + \frac{\sqrt{3}}{6},
$$

$$
\begin{aligned}
q_1 &= (q^-,\ q^-,\ q^-), & q_2 &= (q^+,\ q^-,\ q^-), \\
q_3 &= (q^-,\ q^+,\ q^-), & q_4 &= (q^+,\ q^+,\ q^-), \\
q_5 &= (q^-,\ q^-,\ q^+), & q_6 &= (q^+,\ q^-,\ q^+), \\
q_7 &= (q^-,\ q^+,\ q^+), & q_8 &= (q^+,\ q^+,\ q^+).
\end{aligned}
$$

$$
\int_V \| y - p \|^2 \, d\sigma = min.
$$

$$
m(p_i) = \sum_{j \in N(i)} \left(\frac{1}{8} \sum_{j=1}^{8} p_j V_j\right) / V_{total}^{i},
$$

Collaborators: Prof. Guoliang Xu (Chinese Academy of Sciences, China)

Summary: Quality Improvement with Geometric Flow

- **Quality Improvement**

$$\frac{\partial x}{\partial t} = V_n(k_1, k_2, x)\vec{n}(x) + v(x)\vec{T}(x)$$

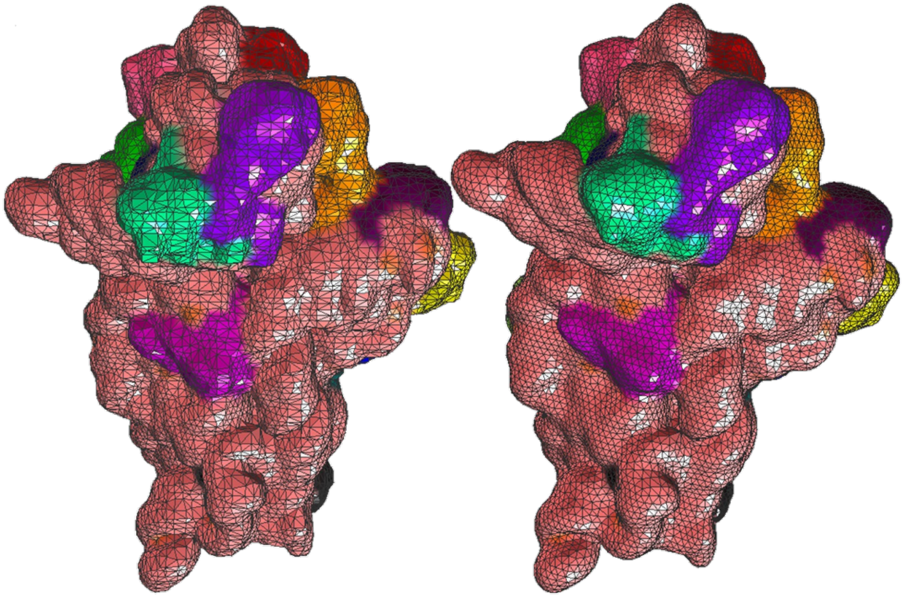
1. Denoising the surface mesh (vertex adjustment in the normal direction).
2. Improving the aspect ratio of the surface mesh (vertex adjustment in the tangent direction).
3. Improving the aspect ratio of the volumetric mesh (vertex adjustment inside the volume).

- Mean Curvature Flow: $V_n = -H = -(k_1 + k_2)/2$
- Average Mean Curvature Flow: $V_n = h(t) - H(t)$

  where $h(t) = \int_{M(t)} H d\sigma / \int_{M(t)} d\sigma$
- Surface Diffusion Flow: $V_n = \Delta H$
- High Order Flow: $\frac{\partial x}{\partial t} = (-1)^{k+1}\Delta^k H N(x)$

Properties:

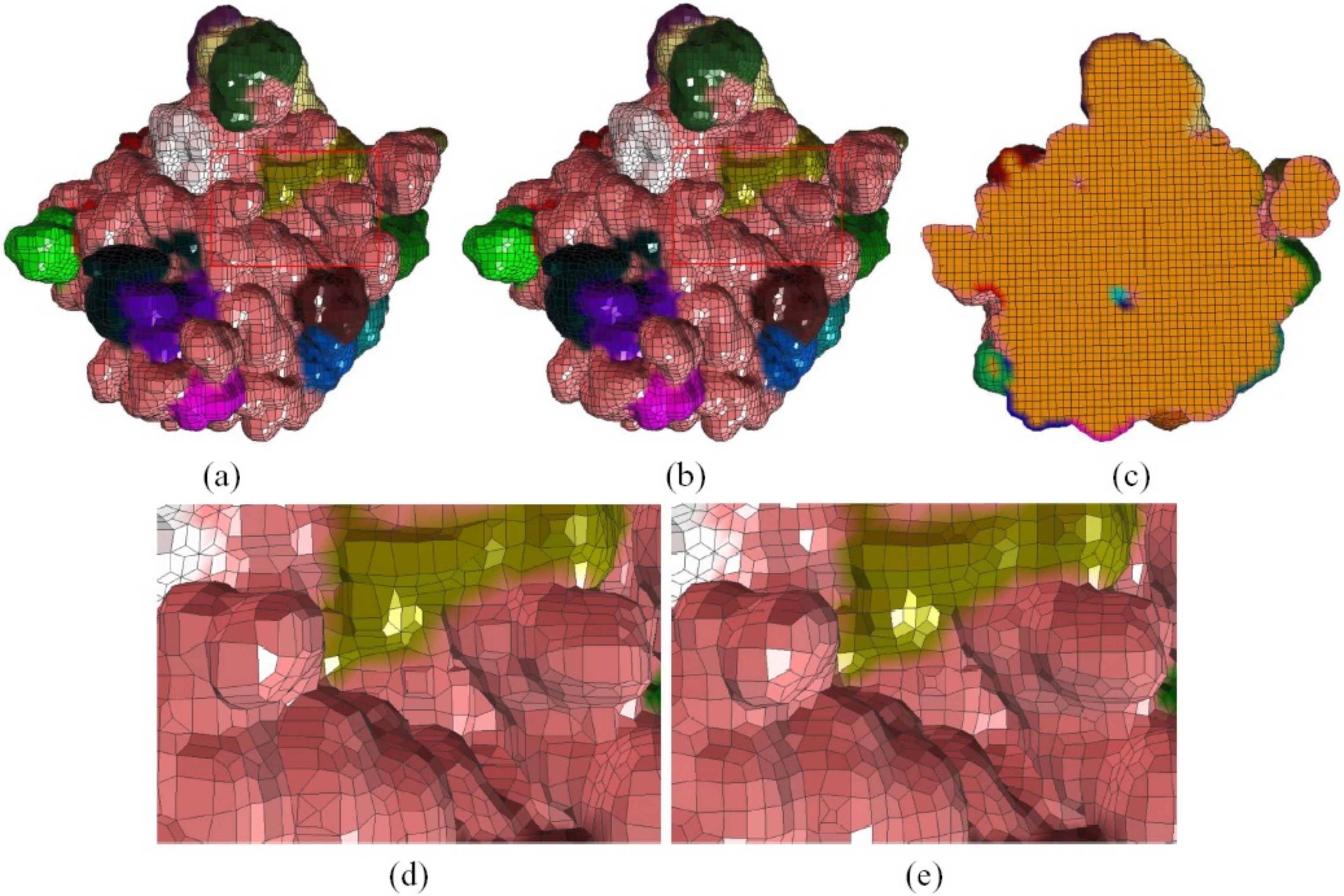
1. Noise removal – normal movement
2. Feature preservation
   - Tangent movement doesn't change the shape
   - The surface diffusion flow is volume preserving
3. Quality improvement
4. Especially suitable for molecular meshes because the surface diffusion flow preserves sphere accurately if the initial mesh is close to a sphere.



References:

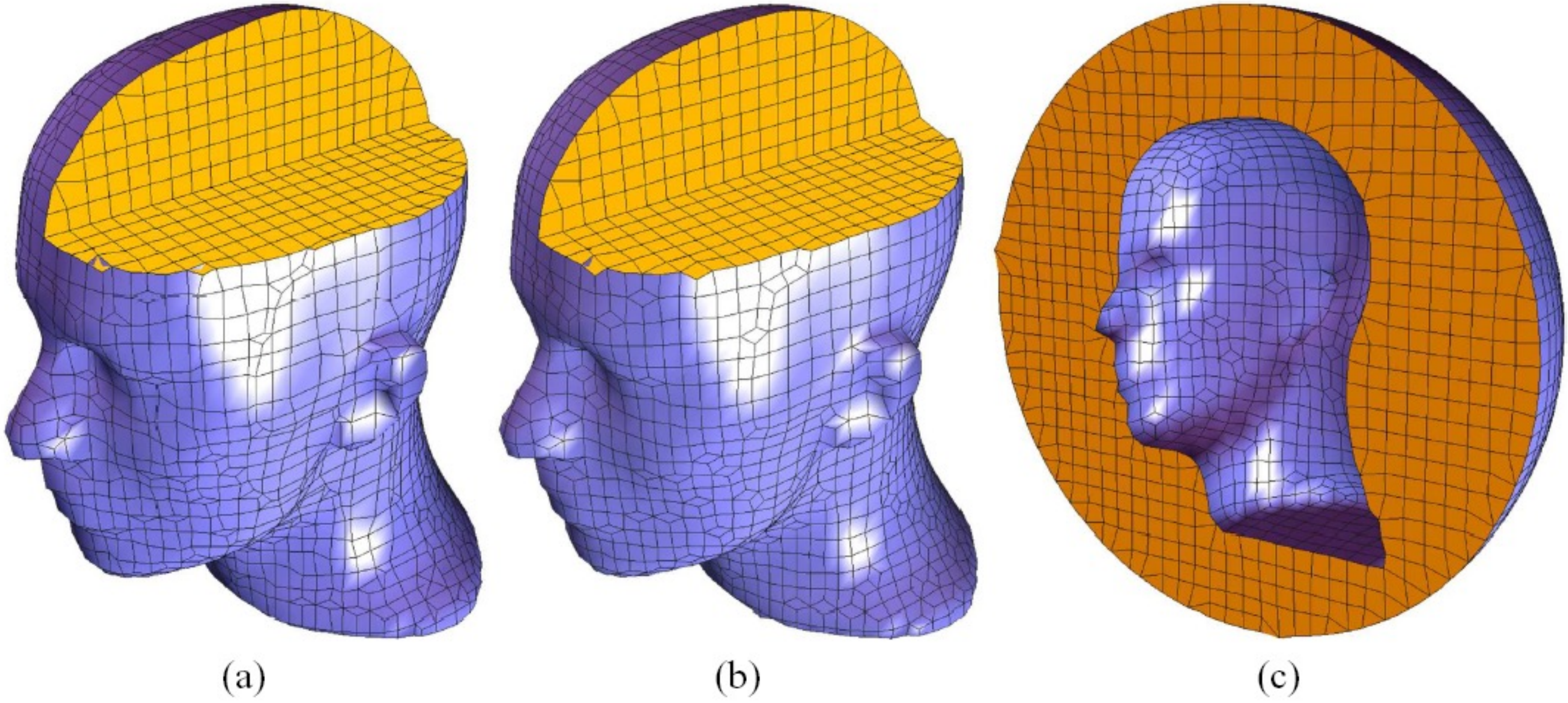
1. Y. Zhang, C. Bajaj, G. Xu. Surface Smoothing and Quality Improvement of Quadrilateral/Hexahedral Meshes with Geometric Flow. *14th International Meshing Roundtable,* 2005.
2. Y. Zhang, G. Xu, C. Bajaj. Quality Meshing of Implicit Solvation Models of Biomolecular Structures. *CAGD,* 2006.
3. Xu G., Pan Q., Bajaj C. Discrete Surface Modeling using Geometric Flows. ICES Technical Report 03-38, the University of Texas at Austin, 2003.
4. Meyer M., Desbrun M., Schroder P., Burr A. Discrete Differential-Geometry Operators for Triangulated 2-Manifolds. VisMath'02, 2002.

Results and Applications



(a) Original mesh; (b) Improved mesh; (c) One cross-section of the improved mesh; (d-e) the zoom-in pictures in the red boxes of (a-b).

Results and Applications

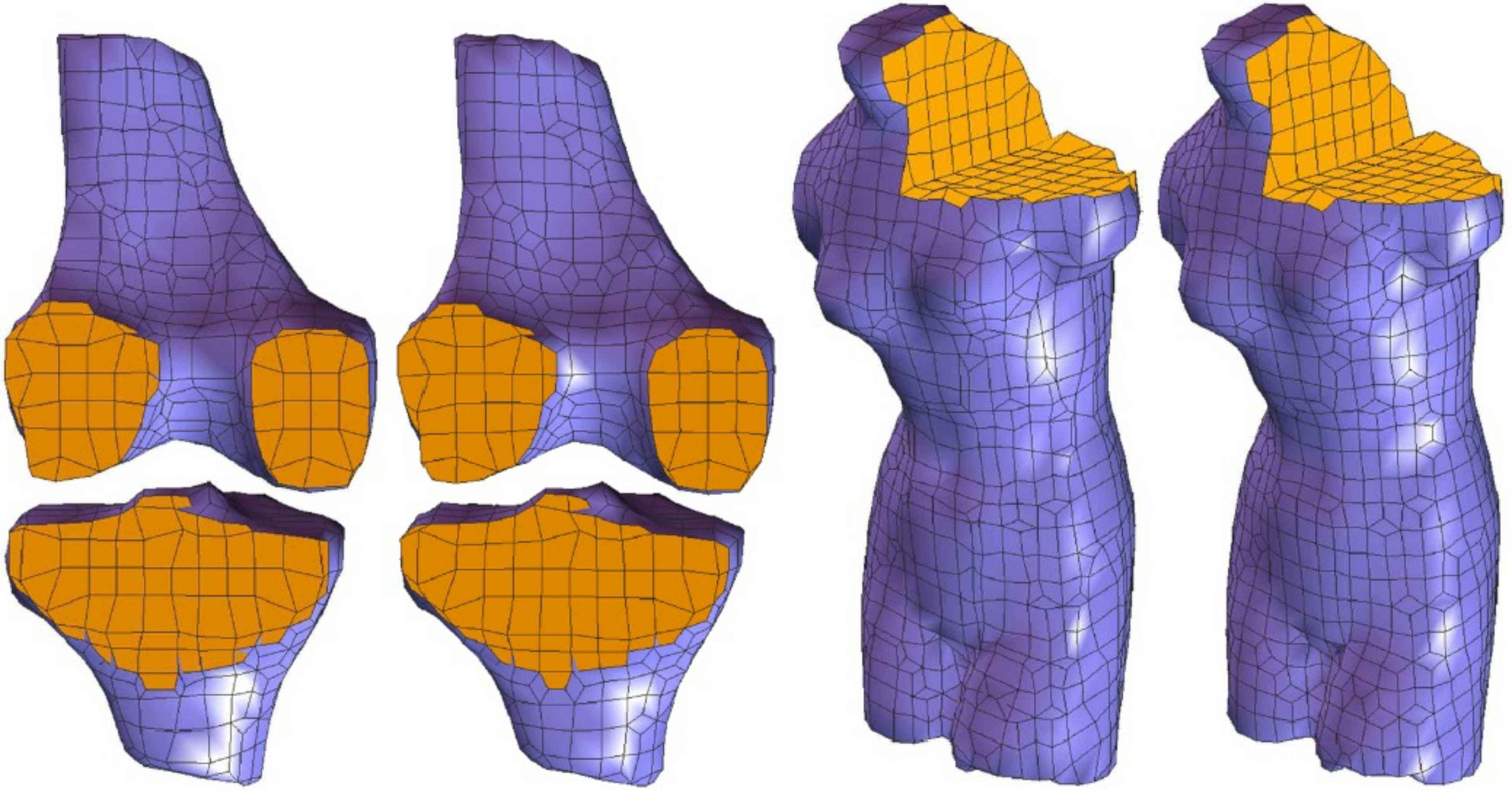


Original mesh

Improved interior mesh

Improved exterior mesh

Results and Applications



(a) Original mesh

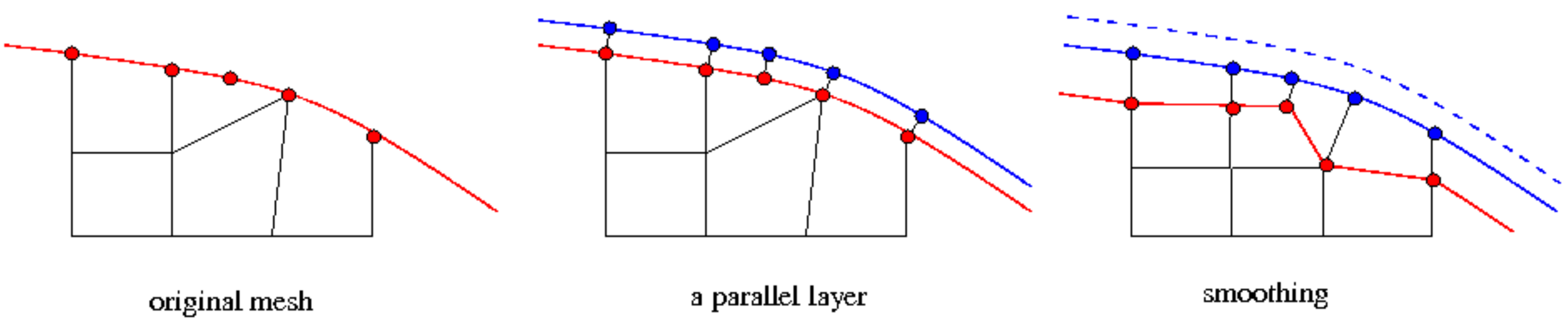
(b) Improved interior mesh

(c) Original mesh

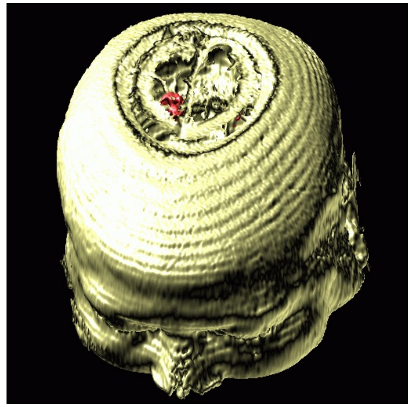
(d) Improved interior mesh

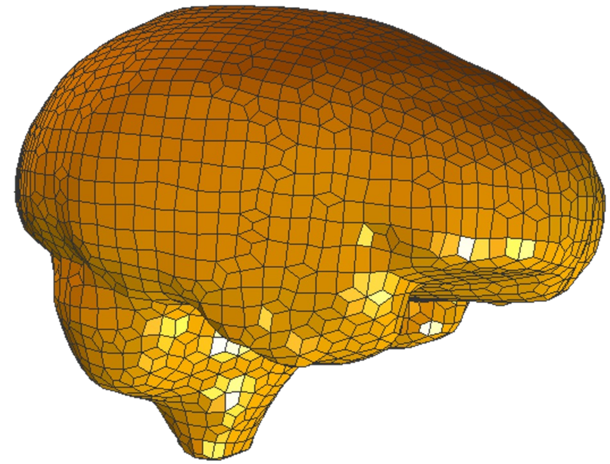
Quality Improvement: Pillowing

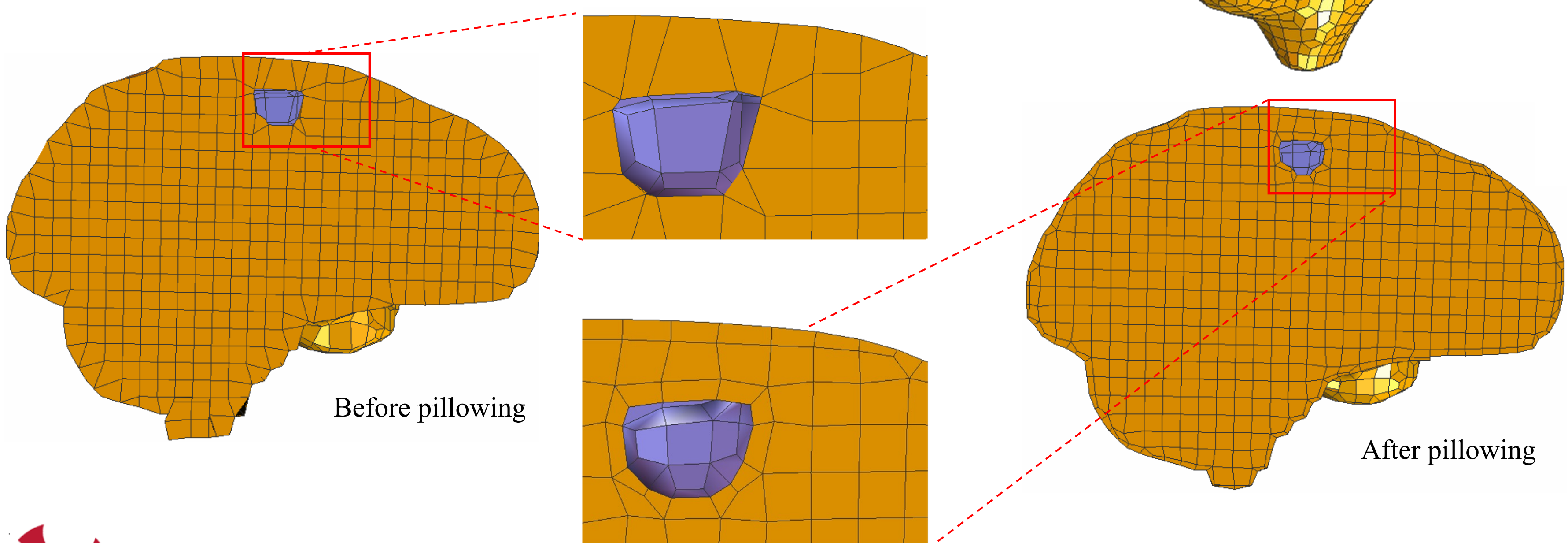
* Pillowing is a quality improvement method for quadrilateral or hexahedral meshes
* Pillowing guarantees that for each element there is at most one face lying on the boundary surface. This gives us a lot of freedom to further improve the aspect ratio of the hexahedral mesh.
* Four steps:
  1. Find out the surface layer L1 (red)
  2. Generate another layer L2 (blue) parallel to L1
  3. Connect corresponding points on L1 and L2 to form hexes
  4. Move L2 to the surface, and use geometric flows to improve the quality



Quality Improvement: Pillowing



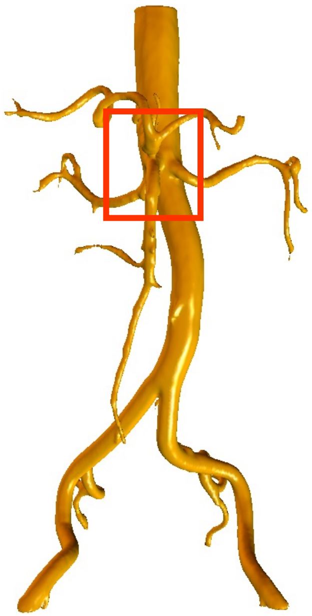
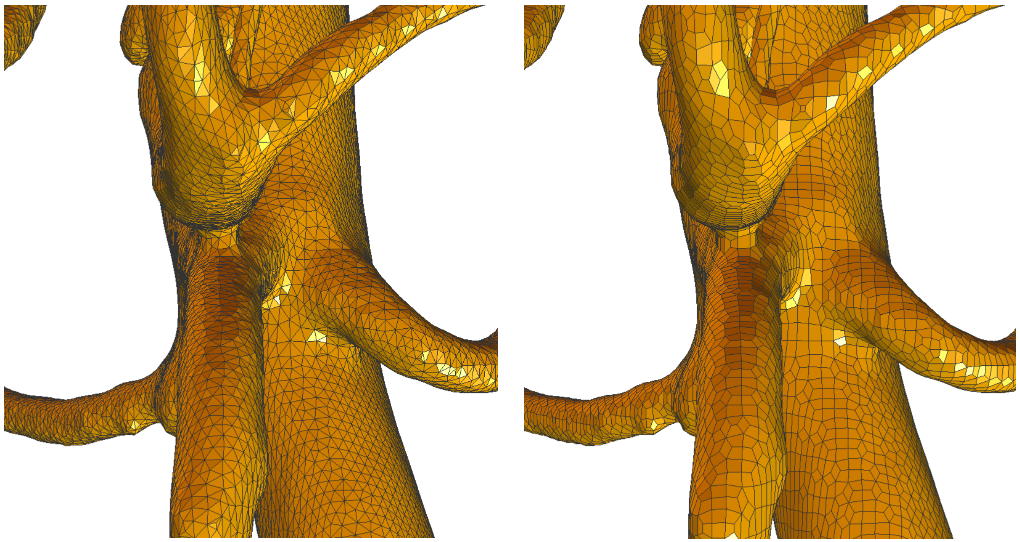
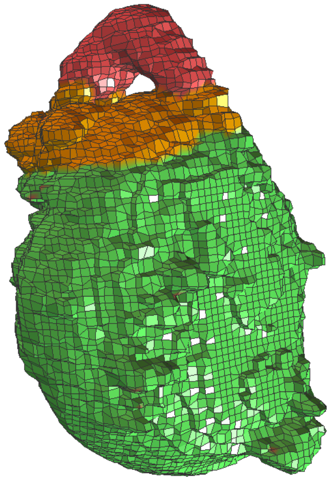
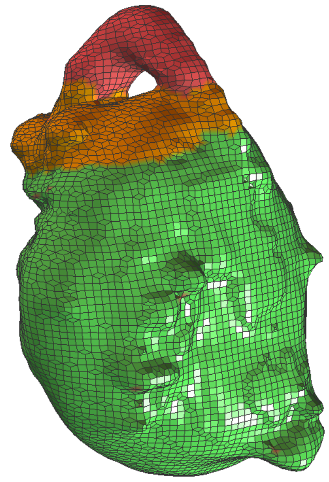
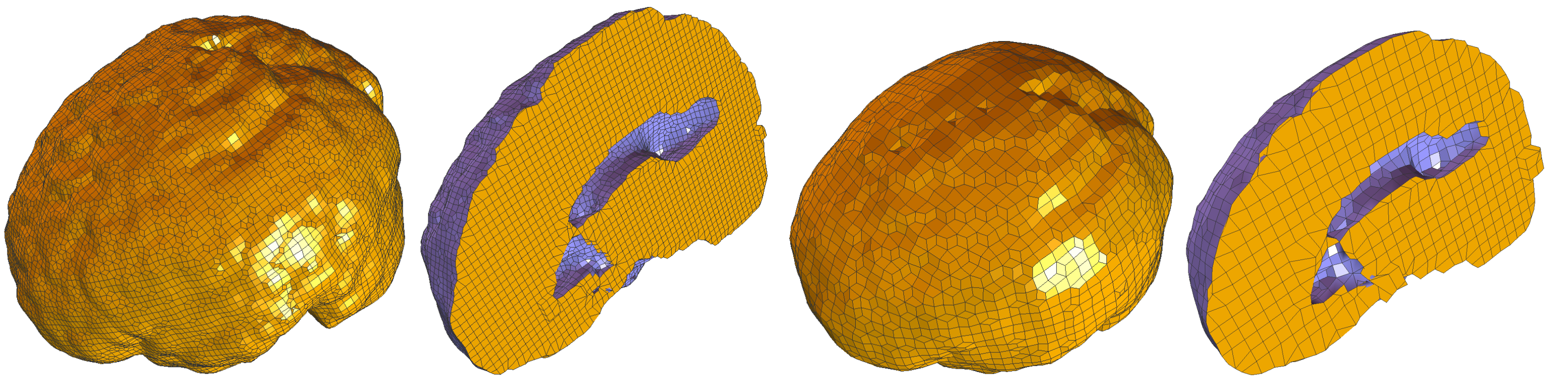




Quality Improvement

- A combination of pillowing, Geometric Flow and Optimization techniques can be used together to improve the quality of hexahedral meshes.

Patient-Specific Geometric Modeling

References

- Ju T., Losasso F., Schaefer S. and Warren J. Dual Contouring of Hermite Data. ACM SIGGRAPH 2002, pages 339-346.
- Y. Zhang, C. Bajaj, B-S. Sohn. 3D Finite Element Meshing from Imaging Data. The special issue of Computer Methods in Applied Mechanics and Engineering (CMAME) on Unstructured Mesh Generation, 194(48-49):5083-5106, 2005.
- Y. Zhang, C. Bajaj. Adaptive and Quality Quadrilateral/Hexahedral Meshing from Volumetric Data. Computer Methods in Applied Mechanics and Engineering (CMAME), 195(9-12):942-960, 2006.
- Y. Zhang, C. Bajaj, G. Xu. Surface Smoothing and Quality Improvement of Quadrilateral/Hexahedral Meshes with Geometric Flow. The special issue of the Journal Communications in Numerical Methods in Engineering (CNME), Vol 24, 2008.
- Foley, J., Van Dam, A., Feiner, S. and Hughes, J. 1995. Computer Graphics: Principles and Practice. Addison Wesley.
- Gibson, S. F. F. 1998. Using distance maps for accurate surface reconstruction in sampled volumes. In 1998 Volume Visualization Symposium, IEEE, 23-30.