W I A S

Weierstraß-Institut für Angewandte Analysis und Stochastik

3D Boundary Conforming Delaunay Mesh Generation

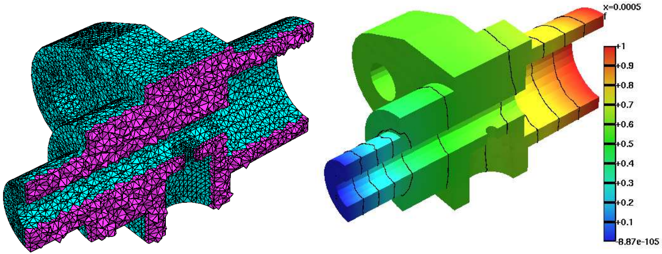
Hang Si

Research Group "Numerical Mathematics and Scientific Computing"
Weierstrass Institute for Applied Analysis and Stochastics
(WIAS) Berlin

Institutskolloquium, WIAS
Juni 25, 2007

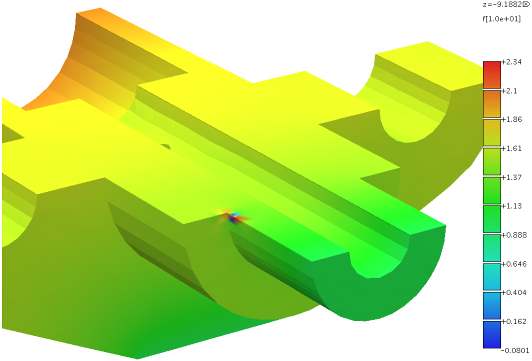
Motivation

- Using numerical methods (such as finite element and finite volume methods) to solve partial differential equations.
- The simulation domain Ω must be subdivided into many simple cells – mesh generation.
- This talk focuses on tetrahedral mesh generation for $\Omega \in \mathbb{R}^3$.



A tetrahedral mesh and the numerical solution of a heat equation.

Motivation



A wrong solution caused by a bad-quality and non-Delaunay mesh.

Motivation

What is a "good" quality mesh?

- Problem-dependent: isotropic, anisotropic, etc.
- Method-dependent: finite element, finite volume, etc.

What is a "good" quality mesh?

- ▷ Problem-dependent: isotropic, anisotropic, etc.
- ▷ Method-dependent: finite element, finite volume, etc.

How to efficiently generate it?

- ▷ Guarantee the quality theoretically.
- ▷ Complete it in polynomial time.

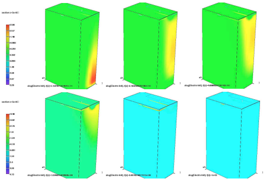
Outline

1. Introduction
2. Delaunay Refinement
3. Adaptive Refinement and Coarsening
4. Application Examples
5. Conclusion

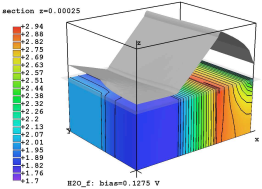
Outline

FVM is a discretization method well suited for numerical simulation of PDEs.

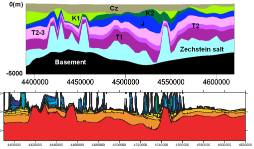
Semiconductor devices



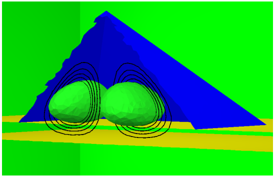
Fuel cells



Thermohaline convection

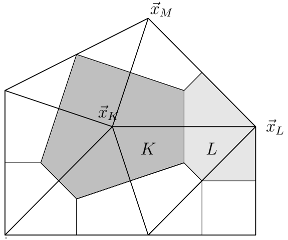


Quantum dots



Eymard R., Gallouët T., and Herbin R., *The Finite Volume Method*. In Ciarlet P.G. and Lions J.L., editors, *Handbook of Numerical Analysis*, Vol. VII, pages, 715–1022. North-Holland, 2000.

Voronoi Finite Volumes

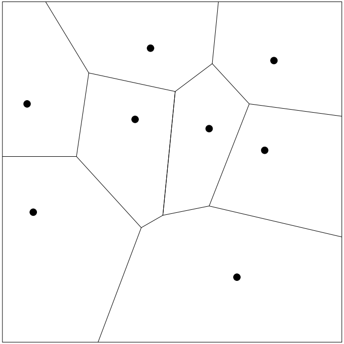


- L^∞ stability, local maximum principle
- Existence of discrete solution
- L^1 contraction, uniquness of the discrete solution
- Discrete $L^2(0, T; H^1(\Omega))$ estimate depending on $\operatorname{reg}\mathcal{D}$ and not on $\operatorname{size}\mathcal{D}$
- Space and time translate estimate not depending on $\mathcal{D}$
- Convergence to weak solution for $\operatorname{size}(\mathcal{D}) \to 0$ while $\operatorname{reg}(\mathcal{D}) \geq \rho$

Fuhrmann J., and Langmach H., *Stability and existence of solutions of time-implicit finite volume schemes for viscous nonlinear conservation laws*. App. Num. Math., **37**:201–230, 2001.

Eymard R., Fuhrmann J., and Gärtner K., *A finite volume scheme for nonlinear parabolic equations derived from 1D local Dirichlet problem*. Numerische Mathematic, **102**(3):463–495, 2006.

Given a set of points $S \subset \mathbb{R}^d$. For each $p \in S$, the Voronoi cell of p, $V(p)$, is:
$V(p) = \{x \in \mathbb{R}^d \mid \forall q \in S \; |x - p| \leq |x - q|\}$.

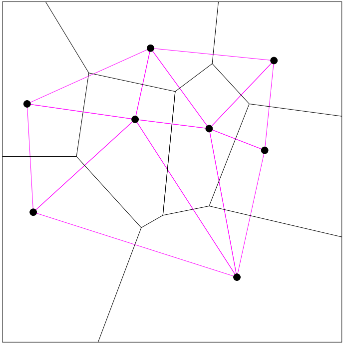


Georgy F. Voronoy (1868-1908)

Voronoi G., *Nouvelles applications des parametrès continus à la théorie de formas quadratiques*. J. Reine Angew. Math. (1907) **133**:97–178, and (1908) **134**:198–287.

The Voronoi Diagram

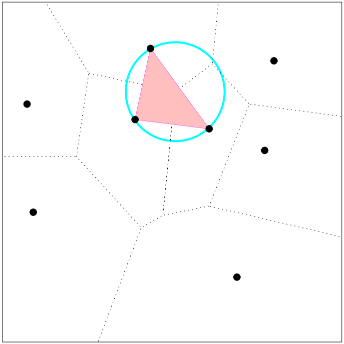
Given a set of points $S \subset \mathbb{R}^d$. For each $p \in S$, the Voronoi cell of p, $V(p)$, is:
$V(p) = \{x \in \mathbb{R}^d \mid \forall q \in S \; |x - p| \leq |x - q|\}$.



Georgy F. Voronoy (1868-1908)

Voronoi G., *Nouvelles applications des parametrès continus à la théorie de formas quadratiques*. J. Reine Angew. Math. (1907) **133**:97–178, and (1908) **134**:198–287.

Given a point set $S \in \mathbb{R}^d$. Any simplex is Delaunay if it has a circumscribed ball B, such that $int(B) \cap S = \emptyset$. The Delaunay triangulation of S, $\mathcal{D}(S)$, is formed by Delaunay simplices.

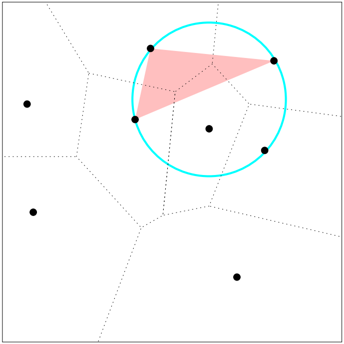


Boris N. Delaunay (1890-1980)

Delaunay B.N., *Sur la sphère vide*. Izvestia Akademii Nauk SSSR, Otdelenie Matematicheskikh i Estestvennykh Nauk. (1934) **7**:793–800.

Delaunay Triangulation

Given a point set $S \in \mathbb{R}^d$. Any simplex is Delaunay if it has a circumscribed ball B, such that $int(B) \cap S = \emptyset$. The Delaunay triangulation of S, $\mathcal{D}(S)$, is formed by Delaunay simplices.

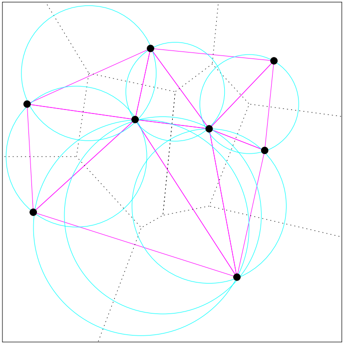


Boris N. Delaunay (1890-1980)

Delaunay B.N., *Sur la sphère vide*. Izvestia Akademii Nauk SSSR, Otdelenie Matematicheskikh i Estestvennykh Nauk. (1934) **7**:793–800.

Delaunay Triangulation

Given a point set $S \in \mathbb{R}^d$. Any simplex is Delaunay if it has a circumscribed ball B, such that $int(B) \cap S = \emptyset$. The Delaunay triangulation of S, $\mathcal{D}(S)$, is formed by Delaunay simplices.



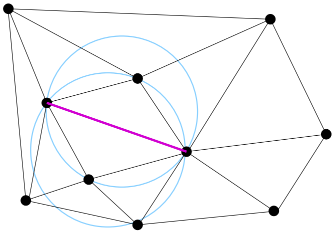
Boris N. Delaunay (1890-1980)

Delaunay B.N., *Sur la sphère vide*. Izvestia Akademii Nauk SSSR, Otdelenie Matematicheskikh i Estestvennykh Nauk. (1934) **7**:793–800.

Delaunay Triangulation

Some properties of Delaunay triangulation. (Explained in 2D, generalized to higher dimensions.)

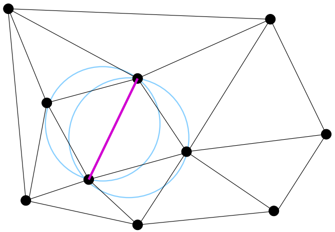
- A simplex is locally Delaunay if it has an empty circumcircle.
- edge flip - local transformation between Delaunay and non-Delaunay simplices.
- Incremental construction and updating.
- Generalize to 3 and higher dimensions.



Delaunay Triangulation

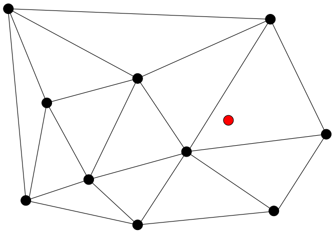
Some properties of Delaunay triangulation. (Explained in 2D, generalized to higher dimensions.)

- A simplex is locally Delaunay if it has an empty circumcircle.
- edge flip - local transformation between Delaunay and non-Delaunay simplices.
- Incremental construction and updating.
- Generalize to 3 and higher dimensions.



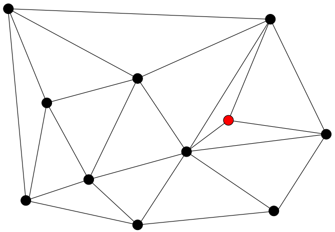
Some properties of Delaunay triangulation. (Explained in 2D, generalized to higher dimensions.)

- A simplex is locally Delaunay if it has an empty circumcircle.
- edge flip - local transformation between Delaunay and non-Delaunay simplices.
- Incremental construction and updating.
- Generalize to 3 and higher dimensions.



Some properties of Delaunay triangulation. (Explained in 2D, generalized to higher dimensions.)

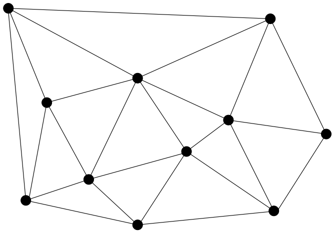
- A simplex is locally Delaunay if it has an empty circumcircle.
- edge flip - local transformation between Delaunay and non-Delaunay simplices.
- Incremental construction and updating.
- Generalize to 3 and higher dimensions.



Delaunay Triangulation

Some properties of Delaunay triangulation. (Explained in 2D, generalized to higher dimensions.)

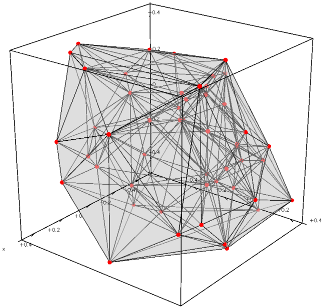
- A simplex is locally Delaunay if it has an empty circumcircle.
- edge flip - local transformation between Delaunay and non-Delaunay simplices.
- Incremental construction and updating.
- Generalize to 3 and higher dimensions.



Delaunay Triangulation

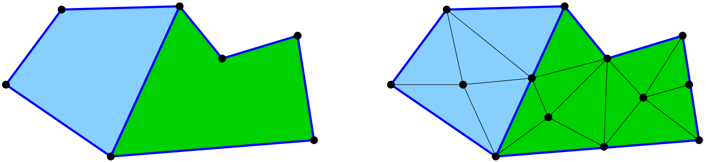
Some properties of Delaunay triangulation. (Explained in 2D, generalized to higher dimensions.)

- ▷ A simplex is locally Delaunay if it has an empty circumcircle.
- ▷ edge flip - local transformation between Delaunay and non-Delaunay simplices.
- ▷ Incremental construction and updating.
- ▷ Generalize to 3 and higher dimensions.



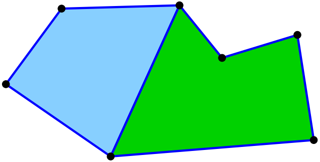
Boundary Conforming Delaunay Mesh

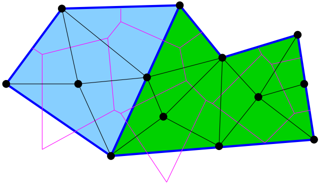
- Given any domain $\Omega \in \mathbb{R}^d$. The Delaunay mesh $\mathcal{T}$ is a partition of Ω by a set of Delaunay simplices and the boundary $\partial\Omega$ is represented by a union of simplices of $\mathcal{T}$.
- The dual Voronoi diagram of a Delaunay mesh may not conform to the input boundary.
- $\mathcal{T}$ is a boundary conforming Delaunay mesh of Ω if the diametric sphere of every boundary simplex of $\mathcal{T}$ is empty.



Boundary Conforming Delaunay Mesh

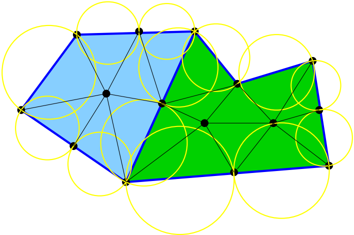
- Given any domain $\Omega \in \mathbb{R}^d$. The Delaunay mesh $\mathcal{T}$ is a partition of Ω by a set of Delaunay simplices and the boundary $\partial\Omega$ is represented by a union of simplices of $\mathcal{T}$.
- The dual Voronoi diagram of a Delaunay mesh may not conform to the input boundary.
- $\mathcal{T}$ is a boundary conforming Delaunay mesh of Ω if the diametric sphere of every boundary simplex of $\mathcal{T}$ is empty.

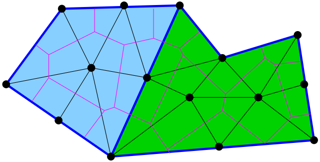




Boundary Conforming Delaunay Mesh

- Given any domain $\Omega \in \mathbb{R}^d$. The Delaunay mesh $\mathcal{T}$ is a partition of Ω by a set of Delaunay simplices and the boundary $\partial\Omega$ is represented by a union of simplices of $\mathcal{T}$.
- The dual Voronoi diagram of a Delaunay mesh may not conform to the input boundary.
- $\mathcal{T}$ is a boundary conforming Delaunay mesh of Ω if the diametric sphere of every boundary simplex of $\mathcal{T}$ is empty.





The Task

For a given 3D domain Ω, find a tetrahedral mesh $\mathcal{T}$, such that

1. $\mathcal{T}$ is a boundary conforming Delaunay mesh (conformity).
2. Tetrahedra of $\mathcal{T}$ are well-shaped (quality guarantee).
3. The number of tetrahedra of $\mathcal{T}$ is small (size guarantee).

The Task

For a given 3D domain Ω, find a tetrahedral mesh $\mathcal{T}$, such that

1. $\mathcal{T}$ is a boundary conforming Delaunay mesh (conformity).
2. Tetrahedra of $\mathcal{T}$ are well-shaped (quality guarantee).
3. The number of tetrahedra of $\mathcal{T}$ is small (size guarantee).

State-of-the-art:

- ▷ Most of the mesh generation methods can satisfy both *2* and *3*, but do not respect the conformity.
- ▷ Methods that theoretically guarantee the *1* have strong limitations.
- ▷ The big gap: lack of implementation.

The Task

For a given 3D domain Ω, find a tetrahedral mesh $\mathcal{T}$, such that

1. $\mathcal{T}$ is a boundary conforming Delaunay mesh (conformity).
2. Tetrahedra of $\mathcal{T}$ are well-shaped (quality guarantee).
3. The number of tetrahedra of $\mathcal{T}$ is small (size guarantee).

State-of-the-art:

- Most of the mesh generation methods can satisfy both *2* and *3*, but do not respect the conformity.
- Methods that theoretically guarantee the *1* have strong limitations.
- The big gap: lack of implementation.

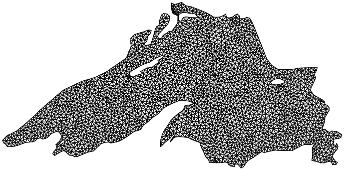
The Goals:

- Further the theoretical work for this problem.
- Implement robust and efficient program for various applications.

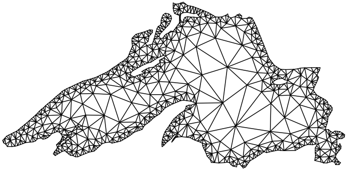
Outline

1. Introduction
2. **Delaunay Refinement**
3. Adaptive Refinement and Coarsening
4. Application Examples
5. Conclusion

Delaunay refinement – mesh refinement based on Delaunay triangulations.
The output is a boundary conforming Delaunay mesh.



$30° \leq \theta_{out}$, **uniform size** [Chew]



$20.7° \leq \theta_{out}$, **graded size** [Ruppert]
Implemented in *Triangle* [Shewchuk]

Chew P.L., *Guaranteed-quality triangular meshes*. Technical Report TR-89-983, Department of Computer Science, Cornell University, 1989.

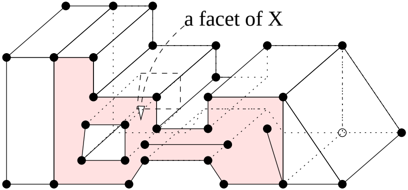
Ruppert J., *A Delaunay refinement algorithm for quality 2-dimensional mesh generation*. J. Algorithms, **18**(3):548–585, 1995.

Shewchuk J.R., *Delaunay refinement mesh generation*. PhD thesis, Department of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania, 1997.

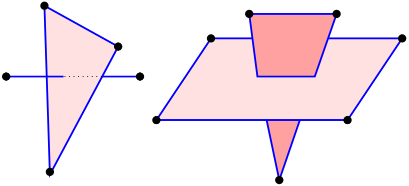
Piecewise Linear Complex

A piecewise linear complex (PLC) [Miller *et al.*'1996] is a set of polytopes X with the following properties:

1. The set X is closed under taking boundaries, i.e., for each $P \in X$ the boundary of P is a union of polytopes in X.
2. X is closed under intersection.
3. If $dim(P \cap Q) = dim(P)$ then $P \subseteq Q$, and $dim(P) < dim(Q)$.

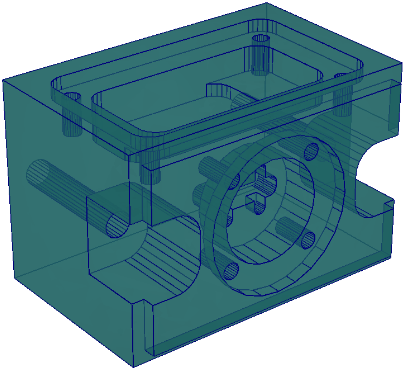
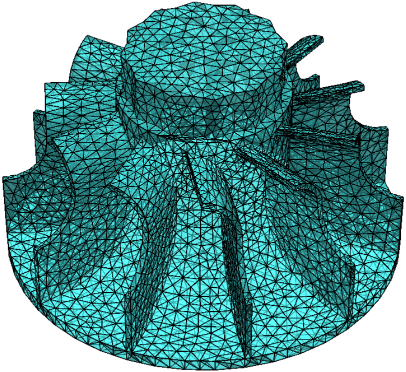


A PLC



non-PLCs

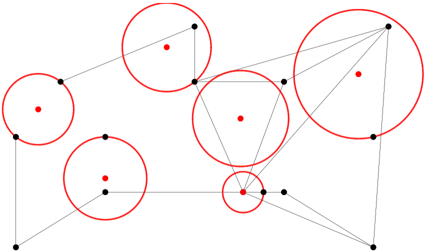
Piecewise Linear Complex

Local Feature Size

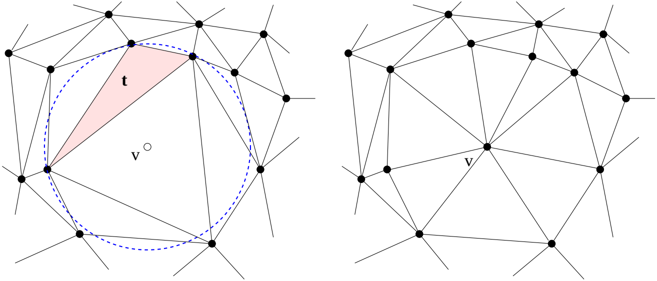
Given a PLC X, the local feature size [Ruppert'1995] at a point $p \in X$, lfs(p), is the radius of the smallest ball centered at p that intersects 2 non-incident boundaries of X.

- Bounded minimum, i.e., for any $p \in X$, lfs$(p) \geq$ lfs$_{min} > 0$.
- Lipschitz function, i.e, for $p, q \in X$, lfs$(p) -$ lfs$(q) \leq |p - q|$.



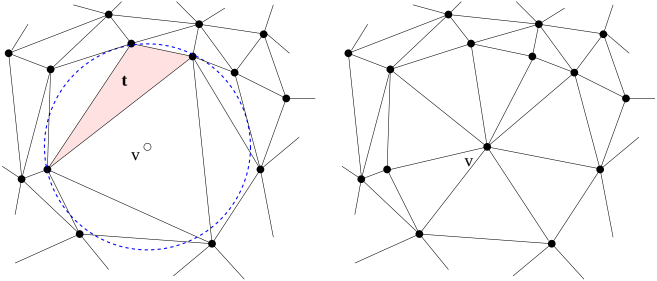
The Basic Idea

- Add the circumcenter of each badly-shaped element. Update the Delaunay triangulation with the new point.
- Prove termination: show that new edges at v never get too short, i.e., $|v - w| \geq \text{lfs}_{min}$.
- Prove well-graded: show that lfs(v) is bounded, i.e., for $D > 0$, lfs$(v) \leq D\ |v - w|$.



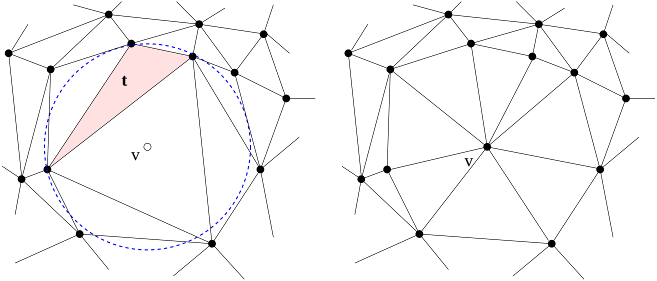
The Basic Idea

- ▷ Add the circumcenter of each badly-shaped element. Update the Delaunay triangulation with the new point.
- ▷ Prove termination: show that new edges at v never get too short, i.e., $|v - w| \geq \mathrm{lfs}_{min}$.
- ▷ Prove well-graded: show that lfs(v) is bounded, i.e., for $D > 0$, lfs$(v) \leq D\,|v - w|$.



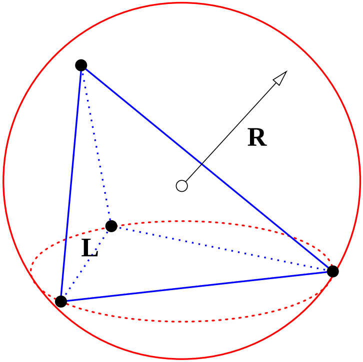
The Basic Idea

- Add the circumcenter of each badly-shaped element. Update the Delaunay triangulation with the new point.
- Prove termination: show that new edges at v never get too short, i.e., $|v - w| \geq \mathsf{lfs}_{min}$.
- Prove well-graded: show that $\mathsf{lfs}(v)$ is bounded, i.e., for $D > 0$, $\mathsf{lfs}(v) \leq D\,|v - w|$.



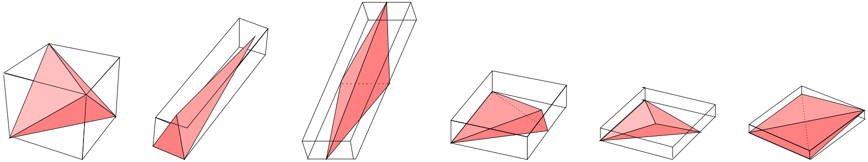
A Quality Measure for Tetrahedron

The radius-edge ratio of a tetrahedron t is the ratio between the radius R of its circumsphere and the length l of the shortest edge, i.e., $Q(t) = R/L$.



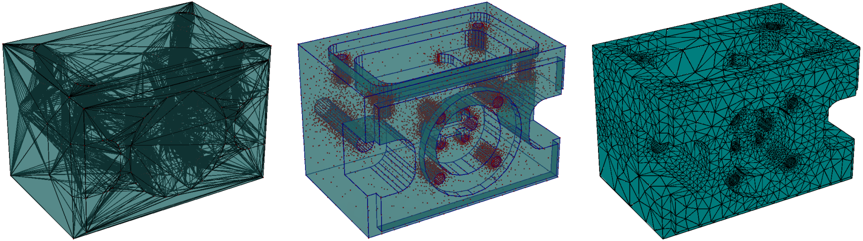
A Quality Measure for Tetrahedron

The radius-edge ratio of a tetrahedron t is the ratio between the radius R of its circumsphere and the length l of the shortest edge, i.e., $Q(t) = R/L$.

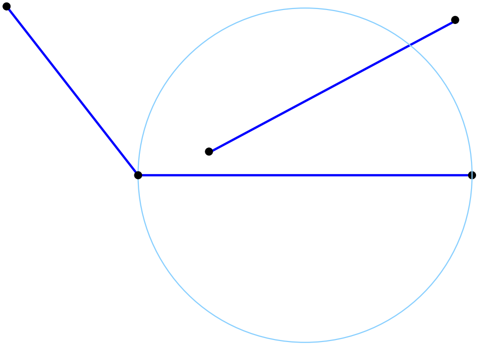


Regular	Needle	Spindle	Wedge	Cap	Sliver
0.612	4.1	26.27	6.74	3.57	0.707

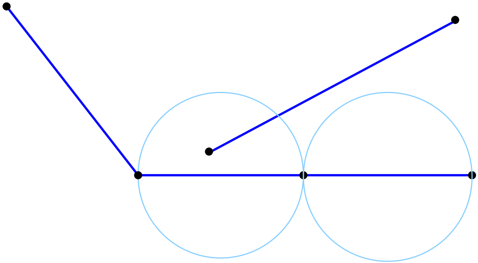
Algorithm DelaunayRefine(X: PLC, ρ_0: radius-edge ratio bound)
Initialize a set $\mathcal{P}$ of vertices of X;
Initialize a Delaunay tetrahedralization, $\mathcal{D}(\mathcal{P})$;
repeat:
Find a new point v by the point generating rules;
Add v to $\mathcal{P}$, update $\mathcal{D}(\mathcal{P})$;
until {no new point can be inserted}.
return current $\mathcal{D}(\mathcal{P})$;



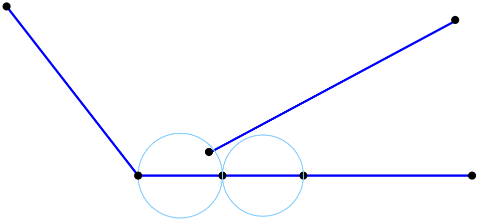
R1. If a subsegment is *encroached*, split it at its midpoint.



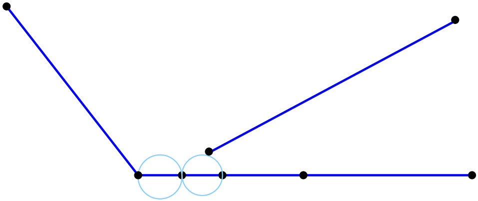
R1. If a subsegment is *encroached*, split it at its midpoint.



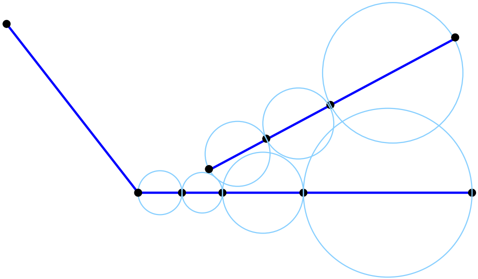
R1. If a subsegment is *encroached*, split it at its midpoint.



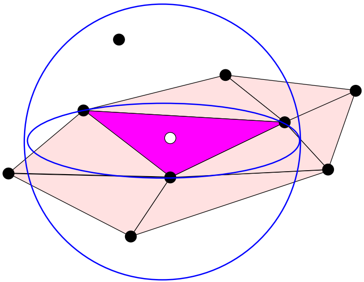
R1. If a subsegment is *encroached*, split it at its midpoint.



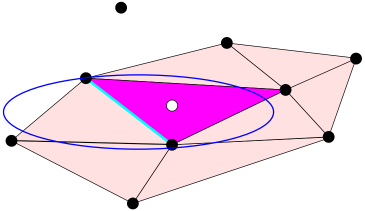
R1. If a subsegment is *encroached*, split it at its midpoint.



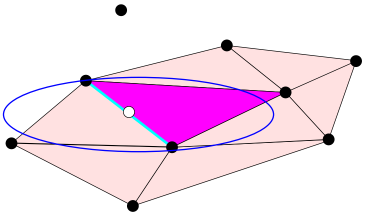
R2. If a subface f is encroached, try to insert its circumcenter c.
If c encroaches upon any subsegment, then reject c.
Instead, use **R1** to split all encroached subsegments.



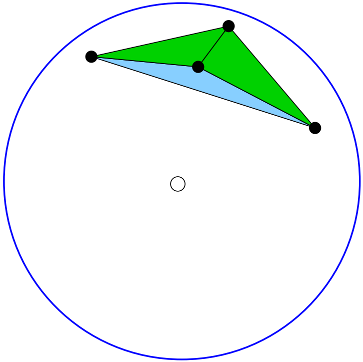
R2. If a subface f is encroached, try to insert its circumcenter c.
If c encroaches upon any subsegment, then reject c.
Instead, use **R1** to split all encroached subsegments.



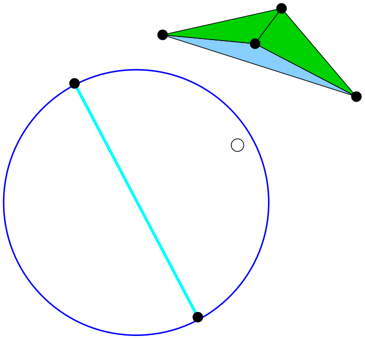
R2. If a subface f is encroached, try to insert its circumcenter c.
If c encroaches upon any subsegment, then reject c.
Instead, use **R1** to split all encroached subsegments.



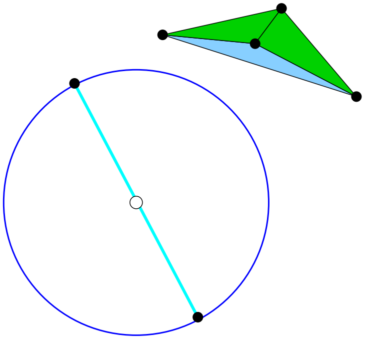
R3. If a tet t is bad ($Q(t) > \rho_0$), try to insert its circumcenter c.
If t encroaches upon any subsegment or subface, then reject c.
Instead, use **R1** and **R2** to split all encroached subsegments and subfaces.



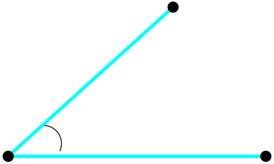
R3. If a tet t is bad ($Q(t) > \rho_0$), try to insert its circumcenter c.
If t encroaches upon any subsegment or subface, then reject c.
Instead, use **R1** and **R2** to split all encroached subsegments and subfaces.



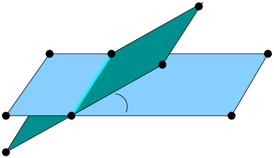
R3. If a tet t is bad ($Q(t) > \rho_0$), try to insert its circumcenter c.
If t encroaches upon any subsegment or subface, then reject c.
Instead, use **R1** and **R2** to split all encroached subsegments and subfaces.



Given a PLC X, define two types of input angles of X.

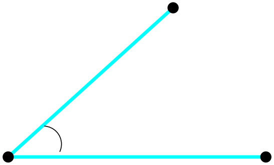


segment-segment angle

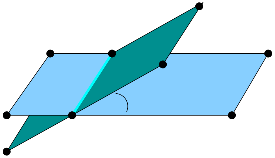


facet-facet (dihedral) angle

Given a PLC X, define two types of input angles of X.



segment-segment angle



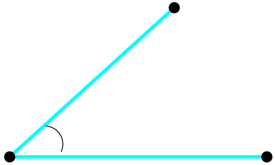
facet-facet (dihedral) angle

The input angle condition:

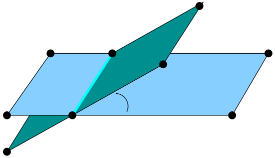
(1) No segment-segment is less than $60°$.

(2) No facet-facet angle is less than $90°$.

Given a PLC X, define two types of input angles of X.



segment-segment angle



facet-facet (dihedral) angle

The input angle condition:

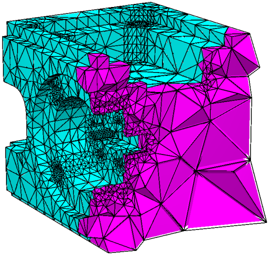
(1) No segment-segment is less than 60°.

(2) No facet-facet angle is less than 90°.

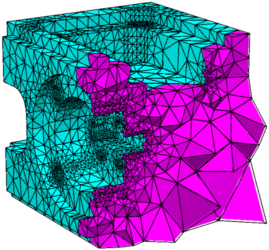
Bounded edge length. For any newly inserted vertex v, lfs$(v) \leq D\,|v - w|$, where $D = \frac{(3+\sqrt{2})\rho_0}{\rho_0 - 2}$.

Analysis [Shewchuk'1997]

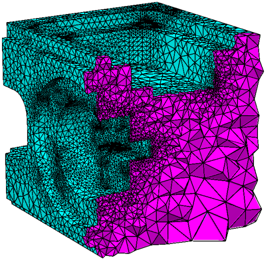
Termination. Assume that X satisfies the input angle condition. Then the algorithm terminates with a radius-edge ratio ρ_0, where $\rho_0 > 2$.



$Q(t) \geq 2.0$
$7,862$ nodes, 1.5 sec.

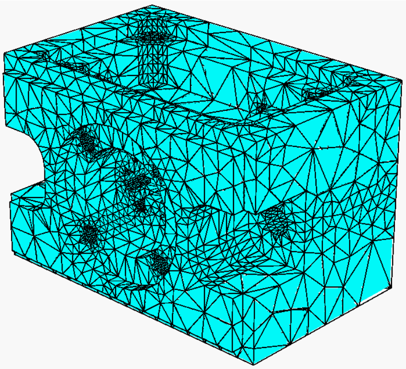


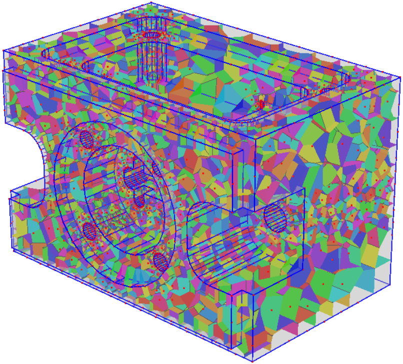
$Q(t) \geq \sqrt{2}$
$14,653$ nodes, 2.5 sec.



$Q(t) \geq 1.1$
$54,560$ nodes, 8.7 sec.

Conformity. The output is a boundary conforming Delaunay mesh.



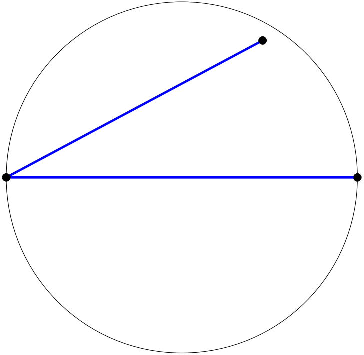


The problem of small angles

The input angle condition:

(1) No segment-segment is less than $60°$.

(2) No facet-facet angle is less than $90°$.

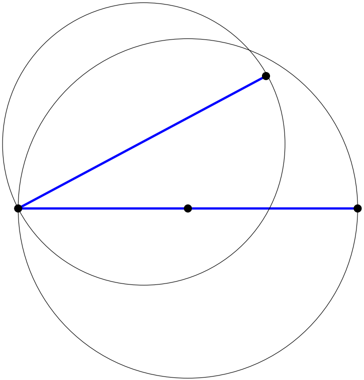


The problem of small angles

The input angle condition:

(1) No segment-segment is less than 60°.

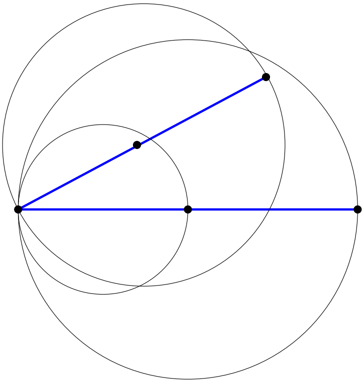
(2) No facet-facet angle is less than 90°.



The input angle condition:

(1) No segment-segment is less than $60°$.

(2) No facet-facet angle is less than $90°$.

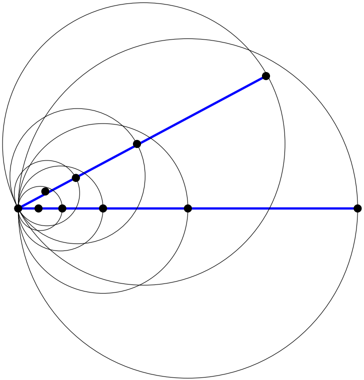


The problem of small angles

The input angle condition:

(1) No segment-segment is less than 60°.

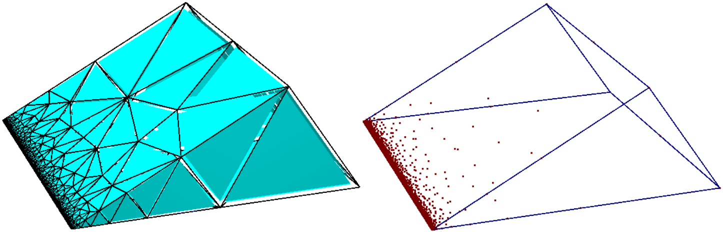
(2) No facet-facet angle is less than 90°.



The input angle condition:

(1) No segment-segment is less than 60°.

(2) No facet-facet angle is less than 90°.

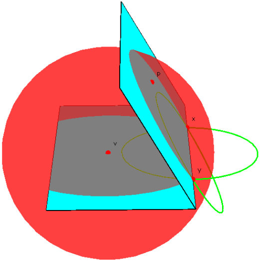


Improvement

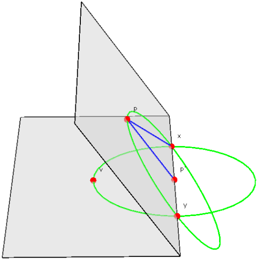
The relaxed input angle condition:

(1) No segment-segment angle is less than $60°$.

(2) No facet-facet angle is less than $\arccos \frac{1}{2\sqrt{2}} \approx 69.3°$.



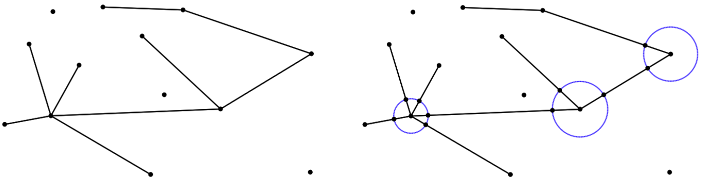
$|p - v| < |p - x|$



$|p - v| \geq \frac{1}{2\sqrt{2}\cos\theta}|p - x|$

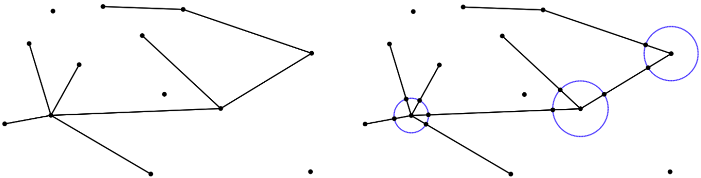
Termination. Assume that X satisfies the relaxed input angle condition. Then the algorithm terminates with a radius-edge ratio ρ_0, where $\rho_0 \geq 2$.

The input segment condition: For any two segments S_1 and S_2 of X, if they are not collinear, then $|S_1| = |S_2|$.



Improvement

The input segment condition: For any two segments S_1 and S_2 of X, if they are not collinear, then $|S_1| = |S_2|$.



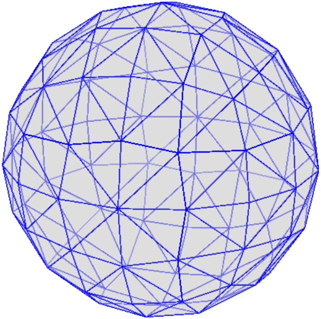
Improved Mesh Quality. Assume that X satisfies the relaxed input angle condition and the input segment condition. Then the algorithm terminates with a radius-edge ratio ρ_0, where $\rho_0 > \sqrt{2}$.

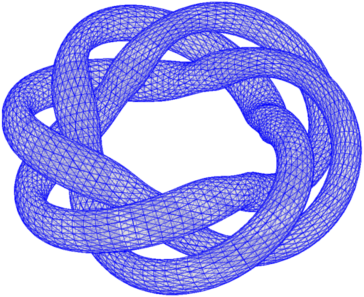
Improvement

The relaxed input angle condition:

(1) No segment-segment angle is less than 60°.

(2) No facet-facet angle is less than $\arccos \frac{1}{2\sqrt{2}} \approx 69.3^\circ$.





minimum segment-segment angle $> 38^\circ$ $> 21^\circ$

Improvement

The modified point generation rule.

$R1^*$ If a segment S is encroached. Let v be its midpoint. Insert v only If:

(1) S is not *sharp*, or

(2) S is *sharp* and the cause of splitting s is an existing mesh vertex.

The modified point generation rule.

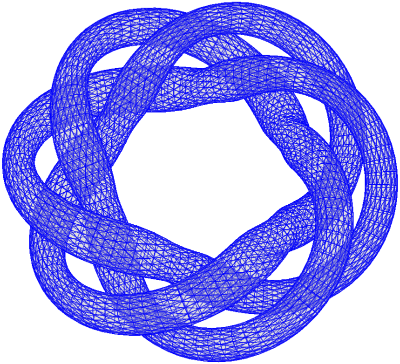
$R1^*$ If a segment S is encroached. Let v be its midpoint. Insert v only If:

- **(1)** S is not *sharp*, or
- **(2)** S is *sharp* and the cause of splitting s is an existing mesh vertex.

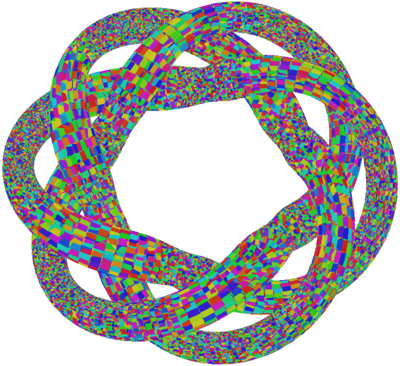
Termination. Assume that X has no facet-facet angle is less than $69.3°$, X satisfies the input segment condition, and $R1^*$ is used. Then the algorithm terminates with a radius-edge ratio ρ_0, where $\rho_0 > \sqrt{2}$.

Improvement

Termination. Assume that X has no facet-facet angle is less than $69.3°$, X satisfies the input segment condition, and $R1^*$ is used. Then the algorithm terminates with a radius-edge ratio ρ_0, where $\rho_0 > \sqrt{2}$.



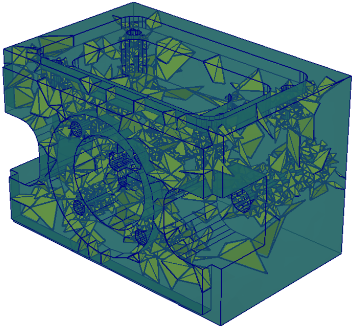
Surface mesh



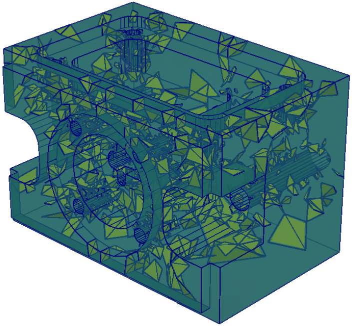
Boundary conforming Voronoi diagram

The problem of slivers

- Slivers ("round" and very "flat" tetrahedra) are not removed.
- Bound the largest (or smallest) dihedral angle, θ_{dihed} (open question).
- In practice, Delaunay refinement works very well by considering θ_{dihed} as an additional quality measure.



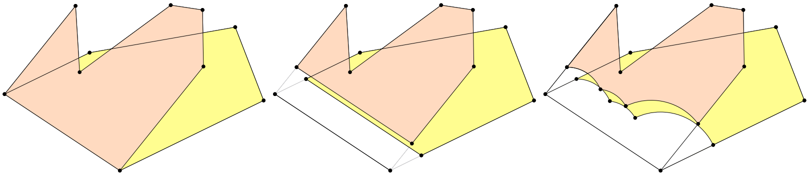
$Q(t) > 2$ ($\theta_{max} = 179.75°$)



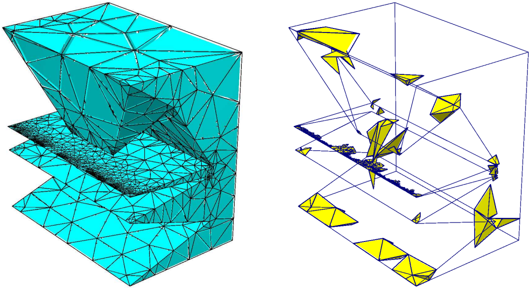
$Q(t) > \sqrt{2}$ ($\theta_{max} = 179.6°$)

Cheng & Poon'03, Cheng & Day'04,05, Pav & Walkington'04, ...

- ▷ Create protect regions to separate small input angles.
- ▷ Use Delaunay refinement to mesh the interior.



- Good mesh quality inside the mesh domain.
- Remaining bad quality tets are close to small input angles.
- No bound on time and space usage – not practical!
- No support of user-defined mesh sizing functions – not adaptive!



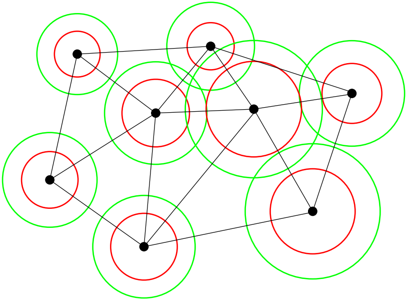
Quality tet mesh generated by *QualMesh* (T. Day).

Outline

1. Introduction
2. Delaunay Refinement
3. **Adaptive Refinement and Coarsening**
4. Application Examples
5. Conclusion

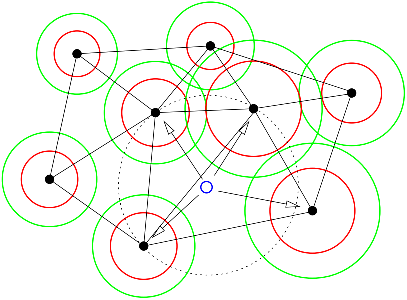
The Idea

- For each point p, assume there are two virtual balls, one protect ball (shown in red), and one sparse ball (shown in green).
- Generate candidates by the Delaunay refinement rules.
- Insert points if they are outside the neighboring protect balls.



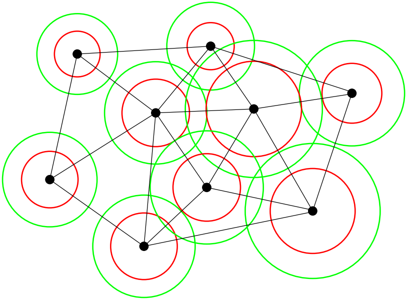
The Idea

- For each point p, assume there are two virtual balls, one protect ball (shown in red), and one sparse ball (shown in green).
- Generate candidates by the Delaunay refinement rules.
- Insert points if they are outside the neighboring protect balls.



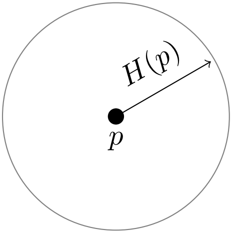
The Idea

- For each point p, assume there are two virtual balls, one protect ball (shown in red), and one sparse ball (shown in green).
- Generate candidates by the Delaunay refinement rules.
- Insert points if they are outside the neighboring protect balls.



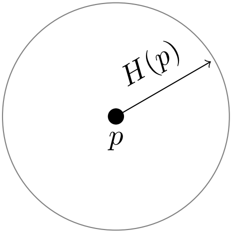
How to Decide the Radii?

- Use a sizing function, $H : \Omega \to \mathbb{R}^+$.
- Introduce two parameters: α_1 and α_2.
- The radii of the sparse and protect balls are $\alpha_1 H(p)$ and $\alpha_2 H(p)$, respectively.



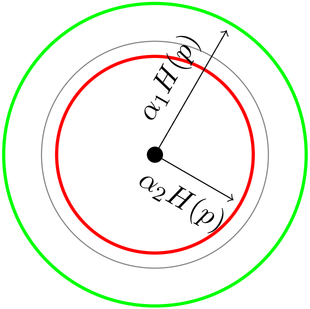
How to Decide the Radii?

- Use a sizing function, $H : \Omega \to \mathbb{R}^+$.
- Introduce two parameters: α_1 and α_2.
- The radii of the sparse and protect balls are $\alpha_1 H(p)$ and $\alpha_2 H(p)$, respectively.



How to Decide the Radii?

- Use a sizing function, $H : \Omega \to \mathbb{R}^+$.
- Introduce two parameters: α_1 and α_2.
- The radii of the sparse and protect balls are $\alpha_1 H(p)$ and $\alpha_2 H(p)$, respectively.



Outline of the algorithm

Algorithm: Adaptive Delaunay refinement.

Input: $\mathcal{T}$, ρ_0, H, α_1, α_2;

Repeat:

generate a new point v by the point generating rules[1];

If $|v - p| > \alpha_2 H(p), \forall p \in \mathcal{T}$ **then**

insert v and update $\mathcal{T}$;

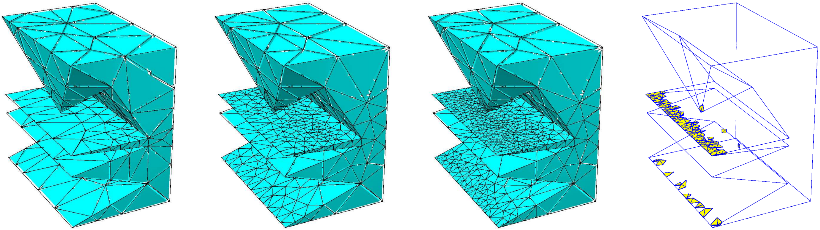
endif

Until no new point can be inserted;

1. $R3$ is modified: if $Q(t) > \rho_0$ or $|v - p| > \alpha_1 H(p), p \in \mathcal{T}$.

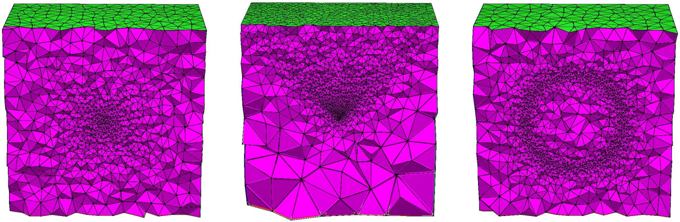
Analysis

- ▷ The algorithm terminates as long as $\alpha_2 > 0$ (No limitation on θ_{input}.)
- ▷ (Mesh quality) Most of the output tetrahedra have $Q(t) > \sqrt{2}$.
  The circumcenter of any bad quality tetrahedron is within distance $\sqrt{2}\alpha_2 H(p)$, where p is a point at sharp features.



$\alpha_2 = 0.5$ $\alpha_2 = 0.2$ $\alpha_2 = 0.1$ $\alpha_2 = 0.1$

Test of mesh conformity - $B = 2.0$, $\alpha_1 = \sqrt{2}$, $\alpha_2 = 0.05$

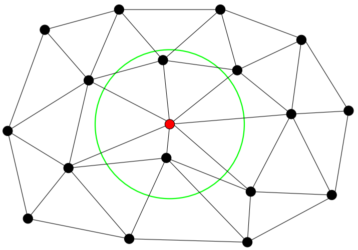


			S_v	L_v	S_v	L_v	S_v	L_v
	$<$	0.5	0	0	0	0	0	0
0.5	$-$	$1/\sqrt{2}$	58	0	0	0	0	0
$1/\sqrt{2}$	$-$	1	3221	1	283	0	0	0
1	$-$	$\sqrt{2}$	15062	113	10778	14	1927	49
$\sqrt{2}$	$-$	2	4246	3867	1187	1044	94186	12594
2	$-$	$2\sqrt{2}$	0	18606	0	11190	12276	95746
	$>$	$2\sqrt{2}$	0	0	0	0	0	0

$S_V = S(v)/H(v)$, $L_v = L(v)/H(v)$, where $S(v)$ and $L(v)$ denote the lengths of the shortest edge and longest edge among all edges connecting at v.

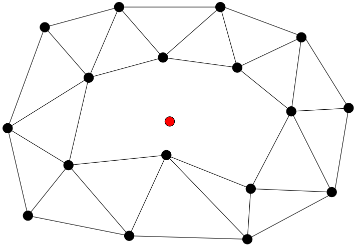
Coarsening

- For each point p, if there exist a point q which is inside the sparse ball of p, e.g., $|p - q| < \alpha_1 H(p)$, then remove it.



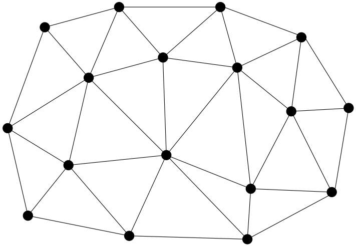
Coarsening

- For each point p, if there exist a point q which is inside the sparse ball of p, e.g., $|p - q| < \alpha_1 H(p)$, then remove it.



Coarsening

- For each point p, if there exist a point q which is inside the sparse ball of p, e.g., $|p - q| < \alpha_1 H(p)$, then remove it.



Algorithm: Adaptive Delaunay coarsening and refinement.

Input: $\mathcal{T}$, ρ_0, H, α_1, α_2;

---

for each $v \in \mathcal{T}$, **do**

  if $|v - p| < \alpha_1 H(v), p \in \mathcal{T}$, **then** remove v;

endfor

---

repeat:

  generate a new point v by the point generating rules[1];

  if $|v - p| > \alpha_2 H(p), p \in \mathcal{T}$ **then**

    insert v and update $\mathcal{T}$;

  endif

until no new point can be inserted;

1. $R3$ is modified: if $Q(t) > \rho_0$ or $|v - p| > \alpha_1 H(p), p \in \mathcal{T}$.

Outline

1. Introduction
2. Delaunay Refinement
3. Adaptive Refinement and Coarsening
4. **Application Examples**
5. Conclusion

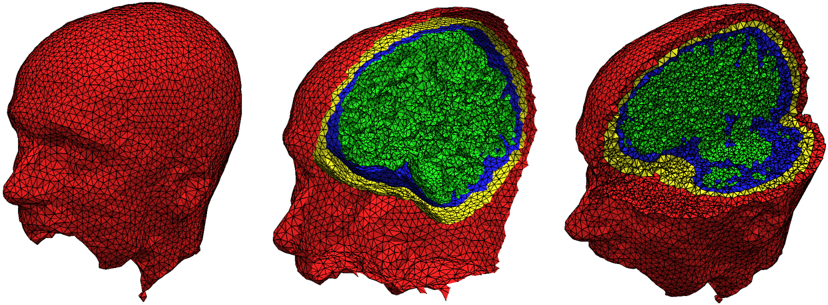
TetGen – A Delaunay Tetrahedral Mesh Generator.

- H can be automatically estimated based on the input geometric data; alternatively, H can be supplied by the user.
- Parameters ρ_0, α_1, and α_2 can be adjusted at the run time.
- Remove slivers by (1) using a dihedral angle bound (adjustable), and (2) mesh optimization and smoothing.
- Capable of dealing with arbitrary 3D PLCs.
- Robust algorithms and implementation.
- Memory efficient.

Freely available for academic and research use (http://tetgen.berlios.de) .

EEG/MEG-source localization

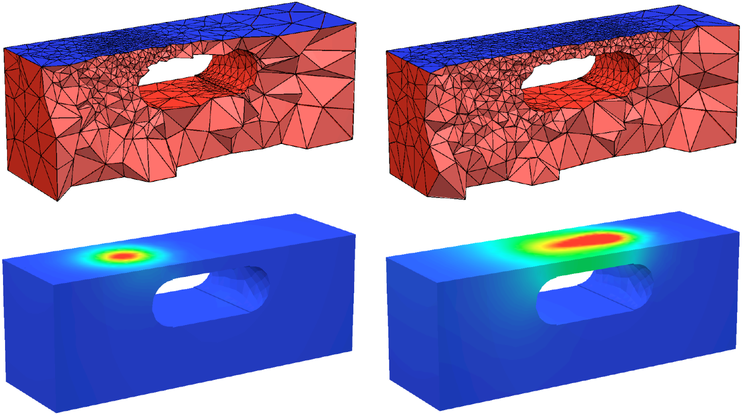
A PLC model of human brain which consists of four surface meshes: skin (red), outer and inner skull (yellow and blue), and cortex (green). (Institut für Biomagnetismus und Biosignalanalyse, Uni. Münster).



Input: $20,301$ points and $40,638$ triangles. Output: $85,312$ points, $528,727$ tetrahedra. CPU time: 18 sec.

Surface Hardening Simulation

Adaptive boundary conforming Delaunay mesh refinement and coarsening applied in the program WIAS Sharp.



Simulation of transient heat conduction (at different times).

Outline

1. Introduction
2. Delaunay Refinement
3. Adaptive Refinement and Coarsening
4. Application Examples
5. **Conclusion**

Conclusion

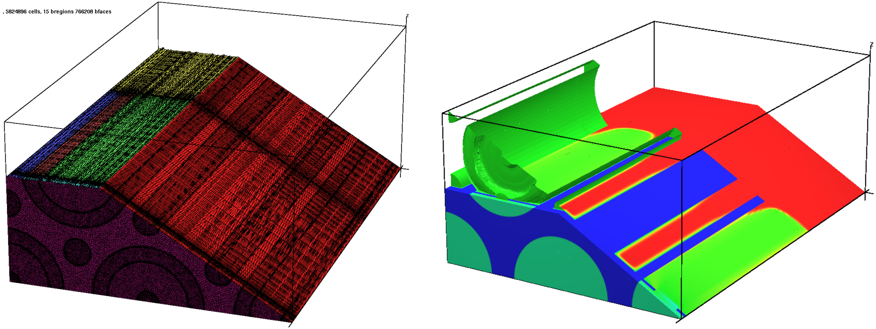
- Boundary conforming Delaunay meshes are well-suited for solving non-linear convection-diffusion problem by finite volume method.
- In 3D, many theoretical questions for creating such mesh are still open. Big gap remains between theory and practice.
- The Delaunay refinement algorithm has been extended:
  1. mesh quality and mesh size remain provable;
  2. no limitation on the input angle;
  3. adaptive refinement and coarsening.
- The **TetGen** program which implements fast, robust, quality-guaranteed algorithms has been used in applications.

Main References

- ▷ Eymard R, Gallouët T, and Herbin R, *Finite Volume Methods*. Handbook of Numerical Analysis, Vol. VII, Elsevier Science B. V., 2000.
- ▷ Shewchuk J, *Tetrahedral Mesh Generation by Delaunay Refinement*. In Proc. 14th Annu. Sympos. Comput. Geom., 1998.
- ▷ Si H, Gärtner K, *Meshing Piecewise Linear Complex by Constrained Delaunay Tetrahedralizations*. In Proc. 14th International Meshing Roundtable, San diego, CA, 2005.
- ▷ Si H, *Adaptive Tetrahedral Mesh Generation by Constrained Delaunay Refinement*. WIAS Preprint 1176, 2006.
- ▷ TetGen, A Tetrahedral Mesh Generator and 3D Delaunay Triangulator. *http://tetgen.berlios.de*

Future Work

- ▷ Sliver removal.
- ▷ Boundary conforming Delaunay for $\theta_{input} < 69.3^\circ$.
- ▷ Anisotropic boundary conforming Delaunay mesh generation.



An anisotropic mesh (left) and the numerical solution (right).