W I A S

Weierstraß-Institut für Angewandte Analysis und Stochastik

TetGen for TEN Computations

Hang Si

Weierstrass Institute for Applied Analysis and Stochastics
Berlin, Germany

3D Topo International Top-Up Day

Outline

1. **Piecewise Linear Complexes**
   - Representation
   - Validation

2. **Constrained Delaunay Tetrahedralizations**
   - Algorithms
   - Examples

Outline

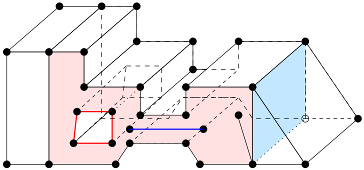
1. **Piecewise Linear Complexes**
   - Representation
   - Validation

2. **Constrained Delaunay Tetrahedralizations**
   - Algorithms
   - Examples

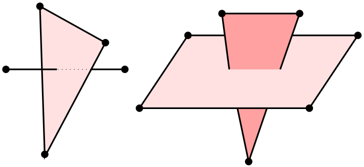
Piecewise Linear Complexes

Definition. A piecewise linear complex (PLC) is a collection X of polytopes (possibly non-convex) 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, i.e., for each $P, Q \in X$ the intersection $P \cap Q$ is a polytope in X.
3. If $\dim(P \cap Q) = \dim(Q)$ then $P \subseteq Q$, and $\dim(P) < \dim(Q)$.

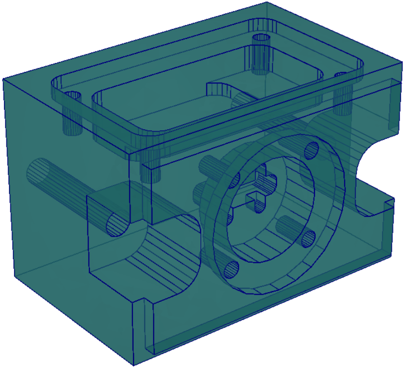


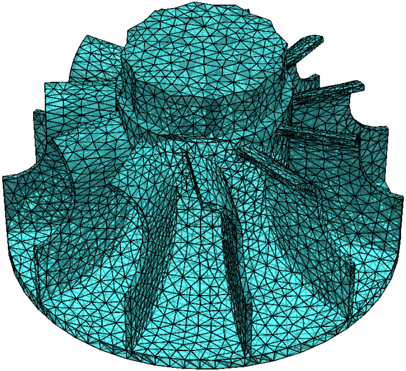
A PLC



non-PLCs

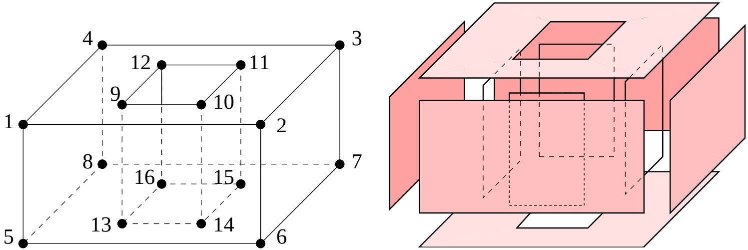
How to Describe PLCs?



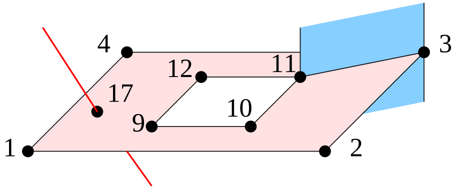


A Simplified B-Rep Description

- A PLC is described by a list of vertices and a list of facets.
  - Each "vertex" contains index, coordinates, attributes, ...
  - Each "facet" is a list of polygons and holes.



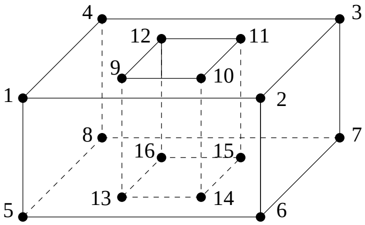
The Facet Description



The facet (shown in pink) consists of four polygons and one hole. The polygons are: $(1,2,3,4)$, $(9,10,11,12)$, $(11,3)$, and (17). The last two polygons are degenerate. The polygon $(9,10,11,12)$ is a hole.

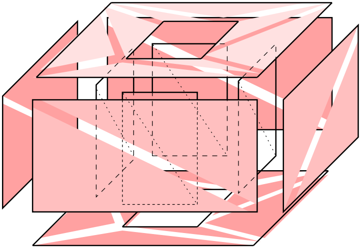
Data File Description (.poly format)

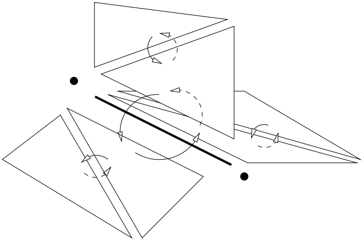
```
# The list of vertices
16 3 0 0
1 0.0 0.0 0.6 # index, x, y, z
2 1.0 0.0 0.6
3 1.0 1.0 0.6
4 0.0 1.0 0.6
...
# The list of facets
10 0
# The top facet
2 1 0 # 2 polygons, 1 hole, no boundary marker
4 1 2 3 4 # A polygon.
4 9 10 11 12
1 0.5 0.5 0.6 # A hole point
# Other facets
1 0 0
4 1 2 6 5
...
# The list of volume holes
0
```



TetGen Internal Representation

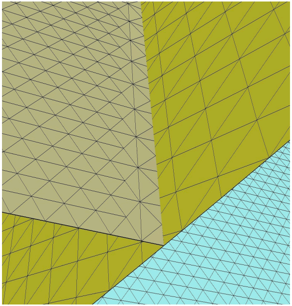
- The boundary of a PLC is stored as a 2D simplicial complex $\mathcal{F}$.
  - Triangulates each facet separately.
  - Connects facets through their common boundaries.
- The triangle-edge data structure **[Mücke'93]** is adapted to represent $\mathcal{F}$.



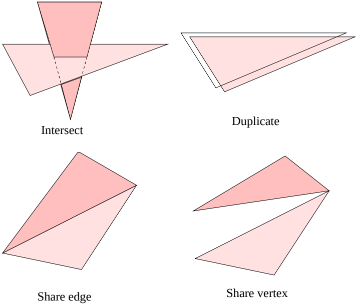


Boundary Self-Intersection Detect

- The primitive operation is the triangle-triangle test. Fast algorithms are known, see **[Möller'97]**, **[Guigue'03]**, etc. However, the implementations are found less robust. Moreover, they generally do not distinguish the type of intersection.
- TetGen has its own triangle-triangle test (the same idea as **[Guigue'03]**) which reports all types of intersections. The robustness is achieved by using exact floating-point arithmetic.



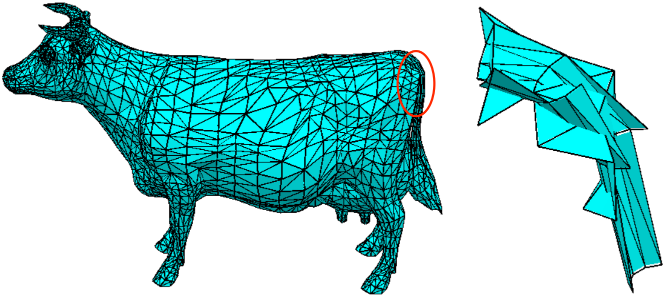
Invalid PLC



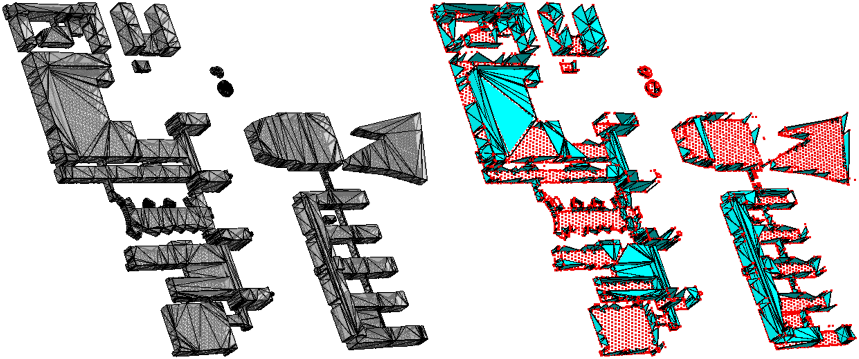
Types of intersection

Self-Intersection Detect Algorithms

- Goal: given a set of m triangles in 3D, find all pairwise intersected triangles.
- A trivial approach: test the intersection of triangles by pairs, needs $O(m^2)$ tests.
- TetGen implemented a hybrid algorithm: initially takes a divide-and-conquer approach and switches to the trivial approach for low number of triangles.
- This algorithm runs in time $O(m \log m + I^2)$, where I is the largest number of undividable triangles.



Self-intersections in the Campus Model



Open Issues

- ▷ Repair self-intersections (one of the goals of the project).
- ▷ Validate the closeness of the PLC boundary.

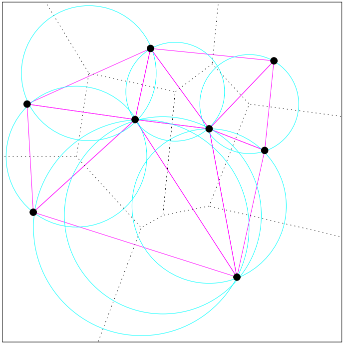
Outline

1. **Piecewise Linear Complexes**
   - Representation
   - Validation

2. **Constrained Delaunay Tetrahedralizations**
   - Algorithms
   - Examples

Delaunay Triangulations

Let S be a set of finite points in $\mathbb{R}^d$. Any simplex in S 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.

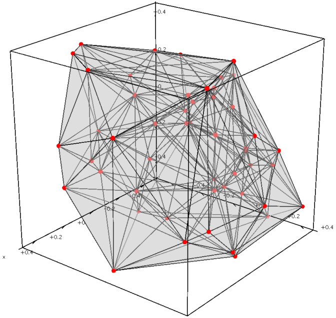


Boris N. Delaunay (1890-1980)

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

Delaunay Triangulations

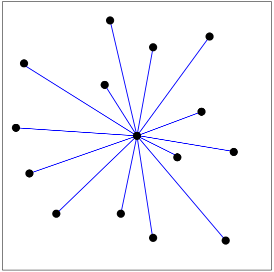
Let S be a set of finite points in $\mathbb{R}^d$. Any simplex in S is Delaunay if it has a circumscribed ball B, such that $int(B) \cap S = \emptyset$. The Delaunay triangulation of S, $\mathcal{D}(S)$, is formed by Delaunay simplices.



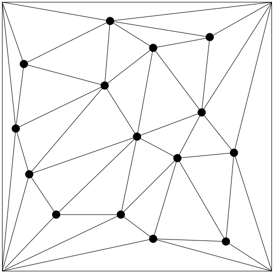
Boris N. Delaunay (1890-1980)

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

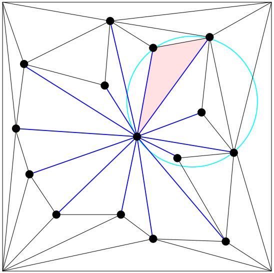
- The Delaunay triangulations (DTs) do not respect the boundaries.
- Constrained Delaunay triangulations (CDTs) well-solved the problem in 2D. (**[Lee & Lin'86]** and **[Chew'89]**)
- Work in progress in 3D, **[Shewchuk'00, 02, 03]**, **[Si & Gärtner'04, 05]**, ...



A 2D PLC



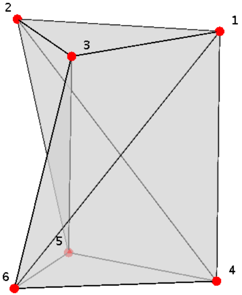
The DT

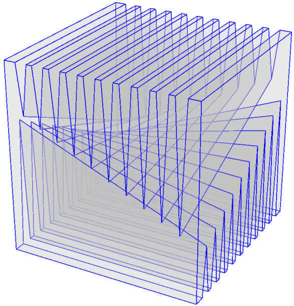


The CDT

Tetrahedralizing Polyhedra

- A simple polyhedron P may not have a tetrahedralization without using additional points (so-called Steiner points). **[Schönhardt'28]**
- The problem of deciding whether P can be tetrahedralized is NP-complete. **[Rupper and Seidel'92]**
- A simple polyhedron with n vertices may require $\Omega(n^2)$ Steiner points. **[Chazelle'84]**

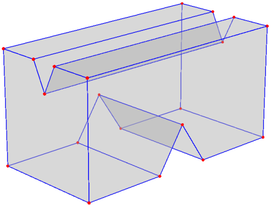




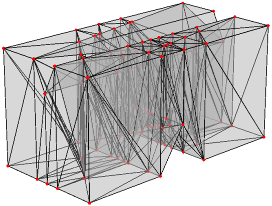
3D Tetrahedralization Methods

- Convex decomposition: [**Chazelle and Palios'90**], [**Bajaj and Dey'92**], etc.
  - Have theoretical guarantees on the complexities $O(n^2)$.
  - Very complicated, require large number of Steiner points.
- Constrained Boundary Recovery: [**George, Hecht, and Saltel'91**], [**Weatherill and Hassan'94**], [**George, Borouchaki, and Saltel'03**], etc.
  - Restriction: no Steiner points are on boundary.
  - complicated, complexities are ad hoc.
- Conforming Delaunay Methods: [**Murphy, Mount, and Gable'00**], [**Cohen-Steiner, de Verdière, and Yvinec'02**], [**Cheng and Poon'03**], etc.
  - May need too many Steiner points.
- Constrained Delaunay Methods: [**Shewchuk'00,02,03**], [**Si and Gärtner'04,05**], etc.
  - Use less Steiner points than conforming Delaunay methods.
  - Have complexity guarantees.

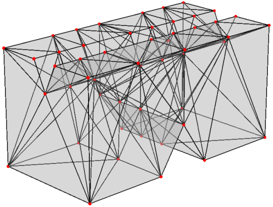
A Comparison of Various Approaches



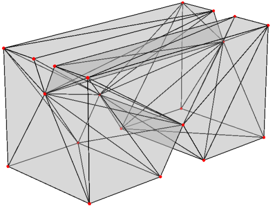
A simple ployhedron
20 vertices, 2 reflex edges



Convex decomposition
138 nodes, 280 tetrahedra



Conforming Delaunay method
51 vertices, 103 tetrahedra

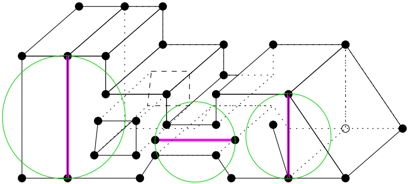


Constrained Delaunay method
20 vertices, 29 tetrahedra.

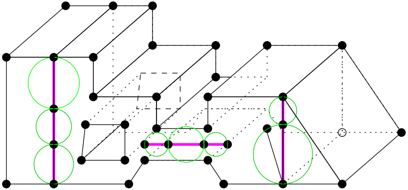
CDTs with no Steiner point

Let X be a 3D PLC. A simplex t is strongly Delaunay if there exists a circumscribed sphere Σ of t, such that no vertex of X lies inside and on Σ.

Theorem ([Shewchuk'98]). If all segment of X are strongly Delaunay, then X has a CDT with no Steiner point.



A PLC X



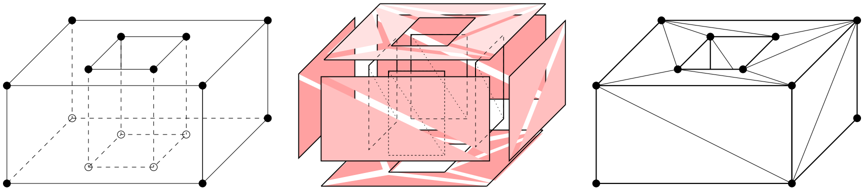
A PLC X'

Corollary. If no 5 vertices of X share a common sphere, and all segment of X are Delaunay, then X has a CDT with no Steiner point, and it is unique.

Constructing the CDT of a PLC

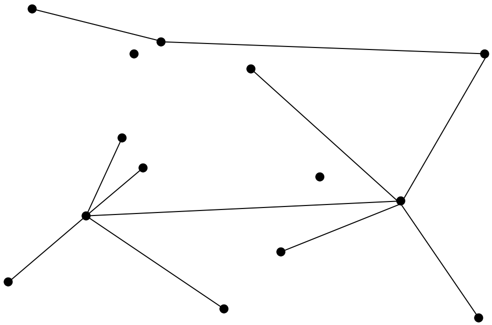
Given a 3D PLC X, a CDT is constructed in the following subsequent phases:

(1) Form the Delaunay tetrahedralization $\mathcal{T}$ of the vertices of X.
(2)* Form the surface triangulation $\mathcal{F}$ from the facets of X.
(3)* Perturb the vertices in $\mathcal{F}$ and $\mathcal{T}$ (add Steiner points).
(4) Recover the segments of $\mathcal{F}$ in $\mathcal{T}$ (add Steiner points).
(5) Recover the facets of $\mathcal{F}$ in $\mathcal{T}$.
(6) Remove tetrahedra outside $|X|$ from $\mathcal{T}$.
(7)* Remove Steiner points from ∂X (for constrained boundary recovery).



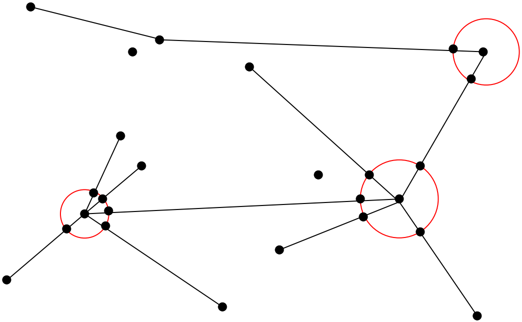
Segment Recovery – Protecting Sharp Corners

Corner lopping **[Ruppert'95, Shewchuk'02, Pav and Walkington'05].**



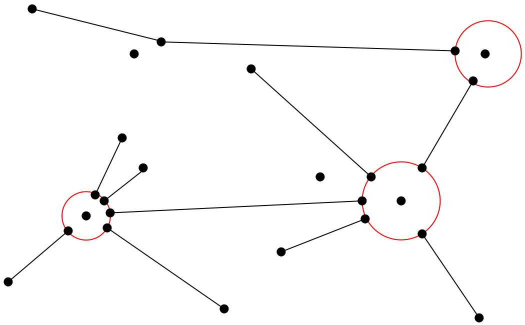
Segment Recovery – Protecting Sharp Corners

Corner lopping **[Ruppert'95, Shewchuk'02, Pav and Walkington'05].**



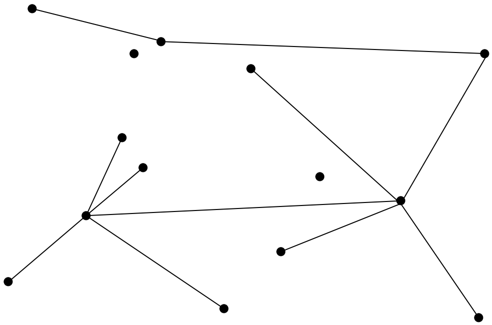
Segment Recovery – Protecting Sharp Corners

Corner lopping **[Ruppert'95, Shewchuk'02, Pav and Walkington'05].**



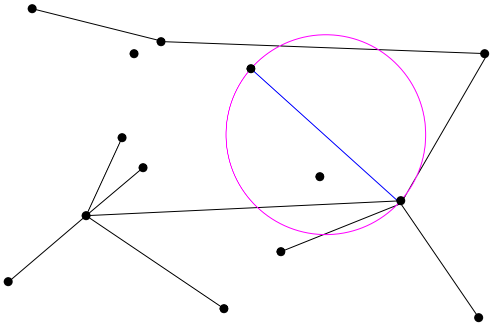
Segment Recovery – Protecting Sharp Corners

Protect sharp corner adaptively **[Si and Gärtner’05]**.



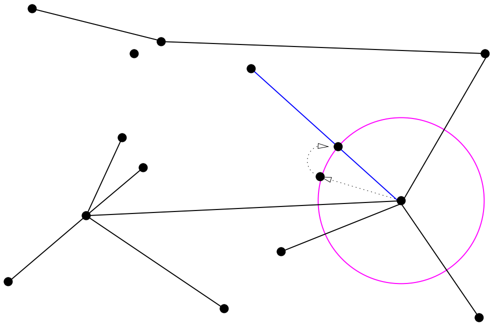
Segment Recovery – Protecting Sharp Corners

Protect sharp corner adaptively **[Si and Gärtner'05]**.



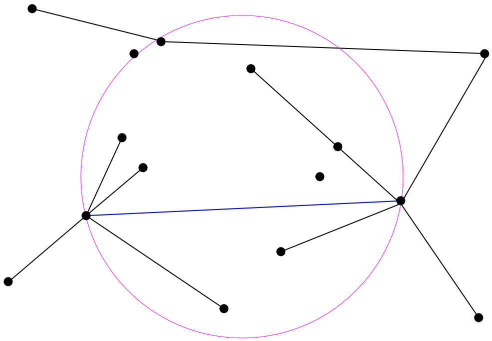
Segment Recovery – Protecting Sharp Corners

Protect sharp corner adaptively **[Si and Gärtner'05].**



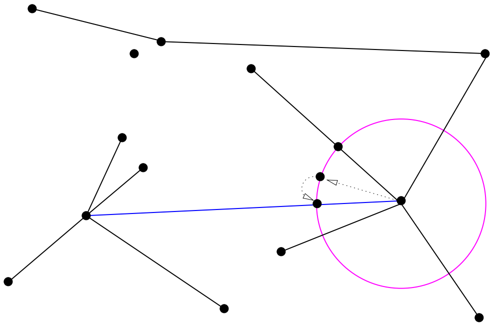
Segment Recovery – Protecting Sharp Corners

Protect sharp corner adaptively **[Si and Gärtner'05].**



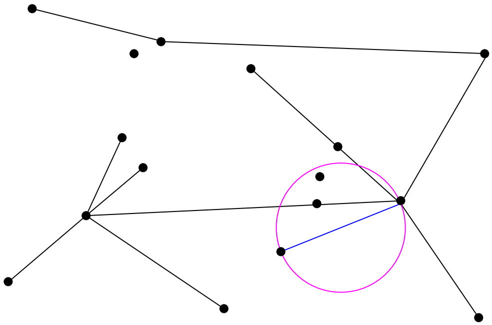
Segment Recovery – Protecting Sharp Corners

Protect sharp corner adaptively **[Si and Gärtner'05].**



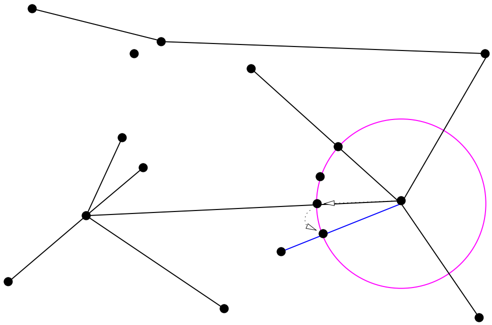
Segment Recovery – Protecting Sharp Corners

Protect sharp corner adaptively **[Si and Gärtner'05]**.



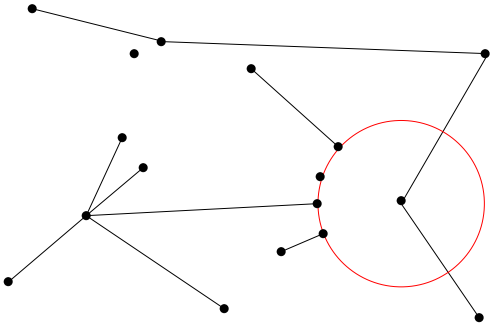
Segment Recovery – Protecting Sharp Corners

Protect sharp corner adaptively **[Si and Gärtner'05].**



Segment Recovery – Protecting Sharp Corners

Protect sharp corner adaptively **[Si and Gärtner'05]**.

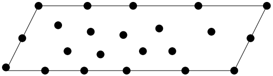


Facet Recovery

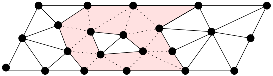
Some subfaces of a facet are missing - they must be non-Delaunay and crossed by Delaunay edges.

A missing region Ω is formed by a set of missing subfaces which are connected to each other.

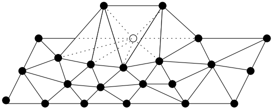
From each Ω one can form two cavities in a DT, one at each side of Ω. Each cavity C is a polyhedron bounded by triangular faces.



A facet



A missing region (shaded area)

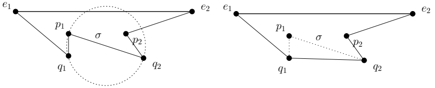


A cavity

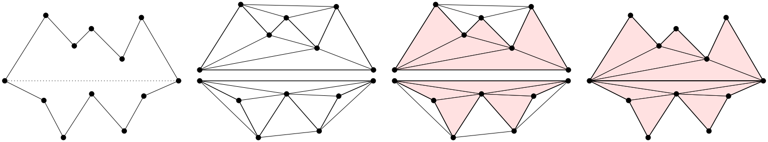
Cavity Tetrahedralization

A cavity C is tetrahedralized by the following procedure.

1. Verify C. Enlarge C until all faces of C are Delaunay.



2. Tetrahedralize C.

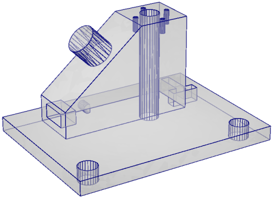


Complexity Issues

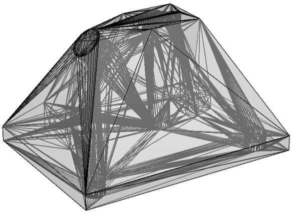
n – the number of input points,
s – the number of Steiner points,
m – the number of output points (i.e., m = n + s).

Steps	Algorithms	Worst case	General case
(1)	Delaunay tetrahedralization	$O(n^2)$	$O(n \log n)$
(2)	Surface triangulation	$O(n \log n)$	
(3)	Vertex perturbation [Si *et al*'05]	$O(n \log n)$	
(4)	Segment recovery [Si *et al*'05]	$O(sn^2 \log n)$	$O(s \log n)$
(5)	Facet recovery [Shewchuk'03]	$O(m^2 \log m)$	

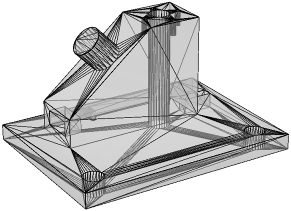
Example 1 – Cami1a



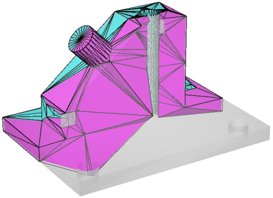
A 3D PLC
460 vertices, 328 facets



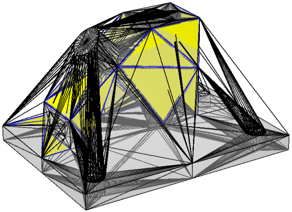
The DT
2637 tetrahedra



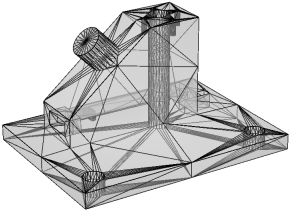
The surface mesh
954 subfaces, 706 segments



A CDT
with 505 Steiner points

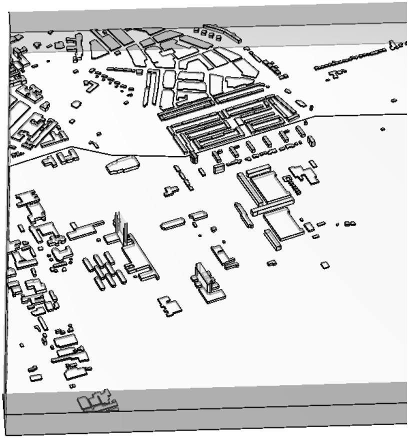


Facet recovery
22 missing subfaces

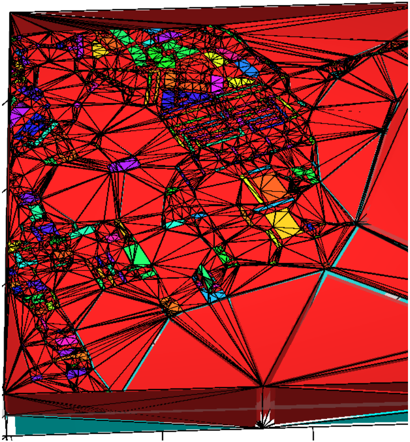


Segment recovery
$213 + 292$ Steiner points

Example 2 – Campus TU Delft



Input PLC
5184 vertices
3229 facets



The CDT
9921 vertices
54338 tetrahedra