Mesh Generation

Pierre Alliez

Inria Sophia Antipolis - Mediterranee
2D Delaunay Refinement
2D Triangle Mesh Generation

Input:
- PSLG C (planar straight line graph)
- Domain \( \Omega \) bounded by edges of C

Output:
- triangle mesh \( T \) of \( \Omega \) such that
  - vertices of C are vertices of T
  - edges of C are union of edges in T
  - triangles of T inside \( \Omega \) have controlled size and quality
Key Idea

- Break bad elements by inserting circumcenters (Voronoi vertices) [Chew, Ruppert, Shewchuk,...] "bad" in terms of size or shape
Basic Notions

C: PSLG describing the constraints
T: Triangulation to be refined

Respect of the PSLG
- Edges a C are split until constrained subedges are edges of T
- Constrained subedges are required to be Gabriel edges
  - An edge of a triangulation is a Gabriel edge if its smallest circumcircle encloses no vertex of T
  - An edge e is encroached by point p if the smallest circumcircle of e encloses p.
Refinement Algorithm

C: PSLG bounding the domain to be meshed.
T: Delaunay triangulation of the current set of vertices
\( T|_\Omega \): \( T \cap \Omega \)
Constrained subedges: subedges of edges of C

Initialise with \( T = \) Delaunay triangulation of vertices of C

Refine until no rule apply

- Rule 1
  if there is an encroached constrained subedge \( e \)
    insert \( c = \) midpoint(\( e \)) in \( T \) (refine-edge)

- Rule 2
  if there is a bad facet \( f \) in \( T|_\Omega \)
    \( c = \) circumcenter(\( f \))
    if \( c \) encroaches a constrained subedge \( e \)
      refine-edge(\( e \)).
    else
      insert(\( c \)) in \( T \)
2D Delaunay Refinement
Background
Constrained Delaunay Triangulation
Delaunay Edge

An edge is said to be a Delaunay edge, if it is inscribed in an empty circle.
Gabriel Edge

An edge is said to be a Gabriel edge, if its diametral circle is empty.
Conforming Delaunay Triangulation

A constrained Delaunay triangulation is a conforming Delaunay triangulation, if every constrained edge is a Delaunay edge.

non conforming

conforming
Conforming Gabriel Triangulation

A constrained Delaunay triangulation is a **conforming Gabriel triangulation**, if every constrained edge is a Gabriel edge.
Steiner Vertices

Any constrained Delaunay triangulation can be refined into a conforming Delaunay or Gabriel triangulation by adding Steiner vertices.
Delaunay Refinement

Rule #1: break **bad** elements by inserting circumcenters (Voronoi vertices)

- “bad” in terms of **size** or **shape** (too big or skinny)

![Diagram showing Delaunay refinement process with bad elements being broken by inserting circumcenters.](Picture taken from [Shewchuk])
Delaunay Refinement

Rule #2: Midpoint vertex insertion
A constrained segment is said to be encroached, if there is a vertex inside its diametral circle.
Delaunay Refinement

Encroached subsegments have priority over skinny triangles

Picture taken from [Shewchuk]
Surface Mesh Generation
Mesh Generation

Key concepts:
- Voronoi/Delaunay filtering
- Delaunay refinement
Voronoi Filtering

- The Voronoi diagram restricted to a curve $S$, $\text{Vor}_{|S}(E)$, is the set of edges of $\text{Vor}(E)$ that intersect $S$. 
The restricted Delaunay triangulation restricted to a curve $S$ is the set of edges of the Delaunay triangulation whose dual edges intersect $S$. 

(2D)
Delaunay Filtering

Delaunay triangulation restricted to surface $S$

Dual Voronoi edge

Voronoi edge $\cap$ surface $S$

facet
Delaunay Refinement

Steiner point

**Bad facet** = big or badly shaped or large approximation error
repeat
{
  pick bad facet \( f \)
  insert furthest \( (\text{dual}(f) \cap S) \) in Delaunay triangulation
  update Delaunay triangulation restricted to \( S \)
}
until all facets are good
Surface Meshing at Work
Isosurface from 3D Grey Level Image
Output Mesh Properties

Termination
Parsimony

Output mesh properties:

- Well shaped triangles
  - Lower bound on triangle angles
- Homeomorphic to input surface
- Manifold
  - not only combinatorially, i.e., no self-intersection
- Faithful Approximation of input surface
  - Hausdorff distance
  - Normals
Delaunay Refinement vs Marching Cubes

Delaunay refinement

Marching cubes in octree
Guarantees

- Produces a good approximation of the surface
  - \( \hat{S} \) is isotopic to \( S \)
  - \( d_H(\hat{S}, S) = O_S(\varepsilon^2) \), error on normals, area = \( O(\varepsilon) \)
  - \( S \) is covered by the surface Delaunay balls of \( \hat{S} \)

- Produces sparse samples of optimal size
  - the set \( E \) of vertices of \( \hat{S} \) is a sparse \( 2\varepsilon \)-sample of \( S \)
  - \( |E| = O(\frac{\text{area}(S)}{\varepsilon^2}) \)

- The aspect ratio of the facets can be controlled
Volume Mesh Generation
Volume Meshing

- **Couple** the latter algorithm with 3D Delaunay refinement
  (insert circumcenters of “bad” tetrahedra)
- Remove slivers at post-processing with sliver exudation
More Delaunay Filtering

Delaunay triangulation restricted to domain

Dual Voronoi vertex inside domain ("oracle")
Delaunay Filtering

domain boundary

restricted Delaunay triangulation
3D Restricted Delaunay Triangulation
Delaunay Refinement

Steiner point
Volume Mesh Generation Algorithm

repeat
{
    pick bad simplex
    if(Steiner point encroaches a facet)
        refine facet
    else
        refine simplex
        update Delaunay triangulation restricted to domain
}
until all simplices are good
Exude slivers
Delaunay Refinement

Apply the following rules with priority order:

**Rule 1:** While there is a facet $f$ in $\text{Del}_{\partial O}(\mathcal{P})$ with vertices $\notin \partial O$

\[ \text{refine_facet}(f) \]

**Rule 2:** While there is a bad facet $f$ in $\text{Del}_{\partial O}(\mathcal{P})$

\[ \text{refine_facet}(f) \]

**Rule 3:** While there is a bad tetrahedron $t$ in $\text{Del}_{O}(\mathcal{P})$

\[ \text{refine_tetrahedron_or_facet}(t) \]
Example
Example
Multi-Domain Volume Mesh
Tetrahedron Zoo
4 well-spaced vertices near the equator of their circumsphere
Slivers
Sliver Exudation [Edelsbrunner-Guoy]

- Delaunay triangulation turned into a regular triangulation with null weights.
- Small increase of weights triggers edge-facets flips to remove slivers.
Sliver Exudation Process

- **Try** improving all tetrahedra with an aspect ratio lower than a given bound
- **Never** flips a boundary facet
Example Sliver Exudation
Piecewise Smooth Surfaces
Input: Piecewise smooth complex
### More Delaunay Filtering

<table>
<thead>
<tr>
<th>primitive</th>
<th>dual of</th>
<th>test</th>
<th>against</th>
</tr>
</thead>
<tbody>
<tr>
<td>Voronoi vertex</td>
<td>tetrahedron</td>
<td>inside</td>
<td>domain</td>
</tr>
<tr>
<td>Voronoi edge</td>
<td>facet</td>
<td>intersect</td>
<td>domain boundary</td>
</tr>
<tr>
<td>Voronoi face</td>
<td>edge</td>
<td>intersect</td>
<td>crease</td>
</tr>
</tbody>
</table>
Delaunay Refinement

- Steiner points
Enrich Set of Rules...

Delaunay refinement
Apply the following rules with priority order

**Rule 1+2:** While there is an edge $e$ in some $\text{Del}_{L_j}(P)$ with vertices $\notin L_j$
  While there is a bad edge $e$ in some $\text{Del}_{L_j}(P)$
  \text{refine_edge}(\langle e \rangle)

**Rule 3+4:** While there is a facet $f$ in some $\text{Del}_{S_i}(P)$ with vertices $\notin S_i$
  While there is a bad facet $f$ in $\text{Del}_{bdO}(P)$
  \text{refine_facet_or_edge}(f)

**Rule 5:** While there is a bad tetrahedron $t$ in $\text{Del}_{O}(P)$
\text{refine_tetrahedron_or_facet_or_edge}(t)

Sliver exudation
Delaunay refinement is followed by a sliver exudation phase
Example

init

5

20

50

end

slivers
Summary

- **Meshes**
  - Definition, variety
  - Background
    - Voronoi
    - Delaunay
    - constrained Delaunay
    - restricted Delaunay

- **Generation**
  - 2D, 3D, Delaunay refinement