Delaunay Triangulation: Applications

Reconstruction

Meshing
Reconstruction

From points
Reconstruction

From points

to shape
Reconstruction

From points
Reconstruction

From points

to shape
Reconstruction

From points
Reconstruction

From points
to shape
Reconstruction

Context

Delaunay is a good start (wanted result $\subset$ Delaunay)

Crust 2D Algorithm

0.4 sample $\Rightarrow$ wanted result $\subset$ crust

0.25 sample $\Rightarrow$ crust $\subset$ wanted result

3D
Reconstruction

Context

Sensor $\rightarrow$ Point set (no structure or unknown)
Reconstruction

Medical Images
Reconstruction

Medical Images

Context
Reconstruction

Context

Childbirth simulation
Reconstruction

Context

Childbirth simulation

Surgery planning

Radiotherapy planing

Endoscopy simulation

• • •
Reconstruction

Sensor ➔ Point set (no structure or unknown)

Scanner
Reconstruction

Context

Sensor ➔ Point set (no structure or unknown)

Scanner

Endoscope

• • •
Reconstruction

Context

Cultural heritage
Reconstruction

Context

Cultural heritage
Reconstruction

Context
Reconstruction  

Reverse engineering
Reconstruction

Reverse engineering

Prototyping (3D print)

Quality control
Reconstruction

Context

Sensor → Point set (no structure or unknown)
Reconstruction

Sensor ———> Point set (no structure or unknown)

Laser illuminate in a plane
Reconstruction

Sensor → Point set (no structure or unknown)

Laser illuminate in a plane

Camera
Reconstruction

Sensor → Point set (no structure or unknown)

Laser illuminate in a plane

Camera

Image
Reconstruction

Context

Sensor → Point set (no structure or unknown)

Laser illuminate in a plane

Camera

Get 3D position
Reconstruction

Context

Sensor → Point set (no structure or unknown)

Geology
Reconstruction

Context

Sensor ➔ Point set (no structure or unknown)

Abstract 3D problem that we can solve in 2D section
Reconstruction

Sensor → Point set (no structure or unknown)

Abstract 3D problem that we can solve in 2D section
Reconstruction

Context

Sensor → Point set (no structure or unknown)

Abstract 3D problem that we can solve in 2D section
Reconstruction

Context

Sensor → Point set (no structure or unknown)

Abstract 3D problem that we can solve in 2D section
Reconstruction

Sensor → Point set (no structure or unknown)

Abstract 3D problem that we can solve in 2D section

Can be solve using Voronoi diagrams
Reconstruction

Context

Sensor → Point set (no structure or unknown)

Abstract 3D problem that we can solve in 2D section
Reconstruction

Sensor \rightarrow Point set (no structure or unknown)

Abstract 3D problem that we can solve in 2D section
Reconstruction

Context

Sensor → Point set (no structure or unknown)

Abstract 3D problem that we can solve in 2D section
Reconstruction

Context

Sensor → Point set (no structure or unknown)

Abstract 3D problem that we can solve in 2D section
Reconstruction

Context

Sensor  →  Point set (no structure or unknown)

Abstract 3D problem that we can solve in 2D section
Reconstruction

Context

Sensor → Point set (no structure or unknown)

Abstract 3D problem that we can solve in 2D section
Reconstruction

Delaunay is a good start

Medial axis of a curve (surface in 3D)

Locus of center of bitangent spheres
Reconstruction

Delaunay is a good start

Medial axis of a curve (surface in 3D)

Locus of center of bitangent spheres
Reconstruction

Delaunay is a good start

Medial axis of a curve (surface in 3D)

Locus of center of bitangent spheres
Reconstruction  Delaunay is a good start

Medial axis of a curve (surface in 3D)

Locus of center of bitangent spheres
Reconstruction  Delaunay is a good start

Medial axis of a curve (surface in 3D)

Locus of center of bitangent spheres
Reconstruction

Delaunay is a good start

Medial axis of a curve (surface in 3D)

Locus of center of bitangent spheres
Reconstruction

Delaunay is a good start

Medial axis of a curve (surface in 3D)

Locus of center of bitangent spheres
Reconstruction

Delaunay is a good start

Medial axis of a curve (surface in 3D)

Locus of center of bitangent spheres
Reconstruction

Delaunay is a good start

$\epsilon$-sample of a curve

Local feature size:
Reconstruction

Delaunay is a good start

$\epsilon$-sample of a curve

Local feature size: $\text{lfs}(x) =$
Reconstruction

Delaunay is a good start

$\epsilon$-sample of a curve

Local feature size: $\text{lfs}(x) = \text{distance}(x, \text{medial axis})$
Reconstruction

Sample is an $\epsilon$-sample of a curve

Local feature size:
Reconstruction

Delaunay is a good start

Sample is an \( \epsilon \)-sample of a curve if \( \forall x, \text{Disk}(x, \epsilon \cdot \text{lfs}(x)) \cap \text{Sample} \neq \emptyset \)

Local feature size: \( \text{lfs}(x) = \text{distance}(x, \text{medial axis}) \)
Reconstruction

Delaunay is a good start

Lemma:

\( \forall \text{ Disk}, \text{ Disk} \cap \text{Curve} \) has a single connected component

or \( \text{Disk} \cap \text{Medial axis} \neq \emptyset \)
Reconstruction

Delaunay is a good start

Lemma:

∀ Disk, Disk ∩ Curve has a single connected component

or Disk ∩ Medial axis ≠ ∅
Reconstruction

Delaunay is a good start

Lemma:

\[ \forall \text{Disk}, \text{Disk} \cap \text{Curve} \text{ has a single connected component} \]

or \[ \text{Disk} \cap \text{Medial axis} \neq \emptyset \]
Reconstruction

Delaunay is a good start

Lemma:

\( \forall \text{ Disk, Disk} \cap \text{Curve has a single connected component} \)

or \( \text{Disk} \cap \text{Medial axis} \neq \emptyset \)

\( \text{Disk} \cap \text{Curve has 2 cc } A \text{ and } B \)
Reconstruction

Delaunay is a good start

Lemma:

\[ \forall \text{ Disk, Disk} \cap \text{Curve} \text{ has a single connected component} \]

\[ \text{or Disk} \cap \text{Medial axis} \neq \emptyset \]
Reconstruction

Delaunay is a good start

Lemma:

\( \forall \text{ Disk, Disk} \cap \text{Curve has a single connected component} \)

or \( \text{Disk} \cap \text{Medial axis} \neq \emptyset \)

Disk \cap \text{Curve has 2 cc } A \text{ and } B

\( a = \text{closest of } c \text{ on Curve} (wlog \text{ on } A) \)

\( b = \text{closest of } c \text{ on } B \)

Moving from \( c \) to \( a \) dist to \( B \)
Reconstruction

Delaunay is a good start

Lemma:

∀ Disk, Disk ∩ Curve has a single connected component

or Disk ∩ Medial axis ≠ ∅

Disk ∩ Curve has 2 cc A and B

\( a = \text{closest of c on Curve (wlog on A)} \)

\( b = \text{closest of c on B} \)

Moving from c to a dist to \( B \uparrow \)

reach center of bitangent disk
Reconstruction

Delaunay is a good start

Theorem
If Sample is a $\varepsilon$-sample, $\varepsilon < 1$

neighboring points along Curve are Delaunay neighbors
Reconstruction

Delaunay is a good start

Theorem
If Sample is a $\varepsilon$-sample, $\varepsilon < 1$

neighboring points along Curve are Delaunay neighbors

Two neighboring points along curve

disks centered on Curve, through $x$
Reconstruction

Delaunay is a good start

Theorem
If Sample is a $\epsilon$-sample, $\epsilon < 1$

neighboring points along Curve are Delaunay neighbors

Two neighboring points along curve

 disks centered on Curve, through $x$
Reconstruction \hspace{1cm} \text{Delaunay is a good start}

Theorem
If Sample is an $\epsilon$-sample, $\epsilon < 1$

neighboring points along Curve are Delaunay neighbors

Two neighboring points along curve

disks centered on Curve, through $x$
Reconstruction

Delaunay is a good start

Theorem
If Sample is a $\varepsilon$-sample, $\varepsilon < 1$

neighboring points along Curve are Delaunay neighbors
Reconstruction

Delaunay is a good start

Theorem
If Sample is a $\epsilon$-sample, $\epsilon < 1$

neighboring points along Curve are Delaunay neighbors

$xx'$ neighbors on curve
$\Rightarrow$ no points on cc $xx'$ in $\bigcirc$

1-sampling $\Rightarrow \bigcirc \subset \bigcirc$
Reconstruction

Delaunay is a good start

Theorem
If Sample is a $\varepsilon$-sample, $\varepsilon < 1$

neighboring points along Curve are Delaunay neighbors

$xx'$ neighbors on curve

$\Rightarrow$ no points on cc $xx'$ in

1-sampling $\Rightarrow$ ⊂ ⊂ $\Rightarrow$ no other cc $\cap$
Reconstruction

Delaunay is a good start

Theorem
If Sample is a $\epsilon$-sample, $\epsilon < 1$

neighboring points along Curve are Delaunay neighbors

Lemma
$xx'$ neighbors on curve
$\Rightarrow$ no points on cc $xx'$ in

1-sampling $\Rightarrow$ no other cc $\cap$ empty

$\Rightarrow$ empty
Reconstruction

Delaunay is a good start

Given a sampling
Reconstruction

Delaunay is a good start

Given a sampling

Compute Delaunay
Reconstruction

Delaunay is a good start

Given a sampling

Compute Delaunay

Search the good sequence of edges there
Reconstruction

Delaunay is a good start

1-sample is not enough
Reconstruction

Delaunay is a good start

1-sample is not enough
Reconstruction

Delaunay is a good start

1-sample is not enough
Reconstruction

Delaunay is a good start

1-sample is not enough

10 - 4
Reconstruction

Crust 2D Algorithm

Compute Voronoi diagram
Reconstruction

Crust 2D

Algorithm

Keep Voronoi vertices
Reconstruction

Crust 2D Algorithm

Keep Voronoi vertices

Compute Delaunay triangulation
Reconstruction

Crust 2D Algorithm

Keep Voronoi vertices

Compute Delaunay triangulation

Keep edges between original points
Reconstruction  Crust 2D  Algorithm

Keep edges between original points
Reconstruction

Crust 2D

Algorithm
Reconstruction

Crust 2D

Algorithm
Reconstruction Crust 2D Algorithm
Reconstruction  
Crust 2D  
Algorithm
Reconstruction

Crust 2D

Algorithm
Reconstruction

Crust 2D

Algorithm
Reconstruction

Crust 2D

Algorithm
Reconstruction

Crust 2D

Theorem: $0.4$ sample $\Rightarrow$ wanted result $\subset$ crust
Reconstruction

Crust 2D 0.4 sample ⇒ wanted result ⊂ crust

Theorem: 0.4 sample ⇒ wanted result ⊂ crust

\(x, x'\) two neighboring points on Curve
Circle thru \(x\) and \(x'\) centered on Curve
Theorem: \( x, x' \) two neighboring points on Curve

Circle thru \( x \) and \( x' \) centered on Curve

By contradiction assume \( v \in \bigcirc \)
Reconstruction

Crust 2D  
0.4 sample ⇒ wanted result ⊂ crust

Theorem:  
0.4 sample ⇒ wanted result ⊂ crust

$x, x'$ two neighboring points on Curve

Circle thru $x$ and $x'$ centered on Curve

By contradiction assume $v ∈ \bigcirc$ intersects another cc of curve  
(by Lemma)
Reconstruction

Crust 2D 0.4 sample \(\Rightarrow\) wanted result \(\subset\) crust

Theorem: 0.4 sample \(\Rightarrow\) wanted result \(\subset\) crust

\(x, x'\) two neighboring points on Curve

Circle thru \(x\) and \(x'\) centered on Curve

By contradiction assume \(v \in\) (by Lemma)

\(\bullet\) intersects another cc of curve

\(R \leq 2r \sin \frac{\theta}{2}\)
Reconstruction

Theorem: \( 0.4 \text{ sample } \Rightarrow \text{ wanted result } \subset \text{ crust} \)

- \( x, x' \) two neighboring points on Curve
- Circle thru \( x \) and \( x' \) centered on Curve
- By contradiction assume \( v \in \circ \)
  - Intersects another cc of curve
  - \( R \leq 2r \sin \frac{\theta}{2} \)
  - \( \theta \leq \)
Theorem: \(0.4\) sample \(\Rightarrow\) wanted result \(\subset\) crust

\(x, x'\) two neighboring points on Curve

Circle thru \(x\) and \(x'\) centered on Curve

By contradiction assume \(v \in \circ\)

\(\circ\) intersects another cc of curve

\(\text{Ifs}\) (by Lemma)

\(R \leq 2r \sin \frac{\theta}{2}\)

\(\theta \leq\)

tangent disk is empty
Theorem: 0.4 sample $\Rightarrow$ wanted result $\subset$ crust

$x, x'$ two neighboring points on Curve
Circle thru $x$ and $x'$ centered on Curve
By contradiction assume $v \in \bigcirc$ intersects another cc of curve
(by Lemma)

$R \leq 2r \sin \frac{\theta}{2}$
$\theta \leq$

wlog $\text{lfs}=1$ and $r \leq \epsilon$
Theorem: \( 0.4 \) sample \( \Rightarrow \) wanted result \( \subset \) crust

\( x, x' \) two neighboring points on Curve
Circle thru \( x \) and \( x' \) centered on Curve

By contradiction assume \( v \in \) intersects another cc of curve (by Lemma)

\[ R \leq 2r \sin \frac{\theta}{2} \]

\( \theta \leq \)

wlog \( \text{lfs} = 1 \) and \( r \leq \epsilon \)
Reconstruction

Crust 2D

Theorem: 0.4 sample \( \Rightarrow \) wanted result \( \subset \) crust

\( x, x' \) two neighboring points on Curve

Circle thru \( x \) and \( x' \) centered on Curve

By contradiction assume \( v \in \) intersects another cc of curve (by Lemma)

\[ R \leq 2r \sin \frac{\theta}{2} \]

\[ \theta \leq \]

\[ r = 2 \sin \frac{\alpha}{2} \]
Reconstruction

Crust 2D

Theorem:

0.4 sample $\Rightarrow$ wanted result $\subset$ crust

$x, x'$ two neighboring points on Curve

Circle thru $x$ and $x'$ centered on Curve

By contradiction assume $v \in \bigcirc$ intersects another cc of curve (by Lemma)

$$R \leq 2r \sin \frac{\theta}{2}$$

$$\theta \leq \beta = \pi - \frac{\pi - \alpha}{2}$$

$$r = 2 \sin \frac{\alpha}{2}$$
Reconstruction

Crust 2D

0.4 sample ⇒ wanted result ⊂ crust

Theorem: 0.4 sample ⇒ wanted result ⊂ crust

$x, x'$ two neighboring points on Curve

Circle thru $x$ and $x'$ centered on Curve

By contradiction assume $v \in$ intersects another cc of curve (by Lemma)

\[
R \leq 2r \sin \frac{\theta}{2}
\]

\[
\theta \leq \beta = \pi - \frac{\pi - \alpha}{2}
\]

\[
\leq \frac{\pi}{2} + \arcsin \frac{r}{2}
\]

\[
r = 2 \sin \frac{\alpha}{2}
\]
Theorem: \( 0.4 \text{ sample } \Rightarrow \text{ wanted result } \subset \text{ crust} \)

\( x, x' \) two neighboring points on Curve

Circle thru \( x \) and \( x' \) centered on Curve

By contradiction assume \( v \in \) intersects another \( cc \) of curve (by Lemma)

\[
R \leq 2r \sin \frac{\theta}{2} \\
\theta \leq \frac{\pi}{2} + \arcsin \frac{r}{2} \\
\| \bullet \times \| \leq \| \bullet \bullet \| + \| \bullet \bullet \|
\]
Theorem: 0.4 sample $\Rightarrow$ wanted result $\subset$ crust

$x, x'$ two neighboring points on Curve

Circle thru $x$ and $x'$ centered on Curve

By contradiction assume $v \in \bigcirc$ intersects another cc of curve (by Lemma)

$$R \leq 2r \sin \frac{\theta}{2}$$
$$\theta \leq \frac{\pi}{2} + \arcsin \frac{r}{2}$$

$$\|\bullet \times\| \leq \|\bullet \bullet\| + \|\bullet \times\|$$
$$\leq r + 2r \sin \left(\frac{\pi}{4} + \frac{1}{2} \arcsin \frac{r}{2}\right)$$
Theorem: \(0.4\) sample \(\Rightarrow\) wanted result \(\subset\) crust

\(x, x'\) two neighboring points on Curve

Circle thru \(x\) and \(x'\) centered on Curve

By contradiction assume \(v \in \bigcirc\) intersects another cc of curve (by Lemma)

\[ R \leq 2r \sin \frac{\theta}{2} \]

\[ \theta \leq \frac{\pi}{2} + \arcsin \frac{r}{2} \]

\[ \|\bullet \times\| \leq \|\bullet \circ\| + \|\bullet \times\| \]

\[ \leq r + 2r \sin \left( \frac{\pi}{4} + \frac{1}{2} \arcsin \frac{r}{2} \right) \]

if \(\|\bullet \times\| \leq \text{lfs} = 1\) contradiction is reached
Reconstruction

Theorem: 0.4 sample \(\Rightarrow\) wanted result \(\subset\) crust

\(x, x'\) two neighbors on Curve

Circle thru \(x\)

By contradiction assume \(v\) intersects another cc of curve

\(R \leq 2r \sin \frac{\theta}{2}\)

\(\theta \leq \frac{\pi}{2} + \arcsin\)

\[ ||\bullet\times|| \leq ||\bullet\bullet|| + ||\bullet\times|| \]

\[ \leq r + 2r \sin \left(\frac{\pi}{4} + \frac{1}{2}\arcsin \frac{r}{2}\right) \]

if \(||\bullet\times|| \leq lfs = 1\) contradiction is reached

Crust 2D 0.4 sample \(\Rightarrow\) wanted result \(\subset\) crust

Plot

\[ r + 2r \sin \left(\frac{\pi}{4} + \frac{1}{2}\arcsin \frac{r}{2}\right) \]
Reconstruction

Theorem: \(0.4 \text{ sample} \Rightarrow \text{wanted result} \subseteq \text{crust}\)
Reconstruction

Crust 2D

0.25 sample \Rightarrow crust \subset \text{wanted result}

Theorem: \hspace{1cm} 0.25 \text{ sample} \Rightarrow \text{crust} \subset \text{wanted result}
Reconstruction

Crust 2D

Theorem: $0.25$ sample $\Rightarrow$ crust $\subset$ wanted result
Reconstruction

Crust 2D

0.25 sample $\Rightarrow$ crust $\subset$ wanted result

Theorem: 0.25 sample $\Rightarrow$ crust $\subset$ wanted result

Assume empty circle
Reconstruction

Theorem: \[ 0.25 \text{ sample} \Rightarrow \text{crust} \subset \text{wanted result} \]

Assume empty circle

No Voronoi vertices there
Reconstruction

Crust 2D

0.25 sample $\Rightarrow$ crust $\subseteq$ wanted result

Theorem: 0.25 sample $\Rightarrow$ crust $\subseteq$ wanted result

Assume empty circle

No Voronoi vertices there

$\Rightarrow$

No sample points there
Reconstruction 3D
Reconstruction

3D

Difficulty: sliver
Reconstruction 3D

Difficulty: sliver
small sphere
Reconstruction  3D

Difficulty: sliver

small sphere  four sample points
Reconstruction 3D

Difficulty: sliver

small sphere four sample points

almost flat Delaunay tetrahedron
Reconstruction

Difficulty: sliver

small sphere  four sample points

almost flat Delaunay tetrahedron

Which triangle belongs to reconstruction?
Reconstruction

3D

Difficulty: sliver

small sphere four sample points

almost flat Delaunay tetrahedron

Which triangle belongs to reconstruction?

Crust: Voronoi vertices may kill useful triangles
Reconstruction 3D
Reconstruction 3D
Reconstruction 3D
Reconstruction

3D

Pole = farthest of seed
Reconstruction 3D

Pole = farthest of seed

2nd pole = farthest of 1st pole
Reconstruction 3D

Pole = farthest of seed

2nd pole = farthest of 1st pole

Approximate normal

Approximate medial axis $\rightarrow$ crust
Reconstruction

3D

Pole = farthest of seed

2nd pole = farthest of 1st pole

Approximate normal

Approximate medial axis $\rightarrow$ crust

Do not kill slivers
Meshing
Meshing

Discretize space to solve (differential) equations

Finite elements

Finite differences
Meshing

Discretize space to solve (differential) equations

Finite elements

Finite differences

Good mesh:

Control shape of elements (no small angles)

Control size of elements (adjust to function variability)

Minimize number of elements
Meshing

Gallery

Structured meshes (advancing front, deformation)

Delaunay mesh refinement

[Ruppert]

protecting small angles

off-centers

Delaunay mesh optimization

3D
Meshing

Gallery
Meshing Gallery
Meshing

Structured meshes

Regular grid
Meshing

Structured meshes

Regular grid

Shape
Meshing

Structured meshes

Regular grid

Shape

Deform
to fit the grid in the shape
Meshing

Structured meshes

Regular grid

Shape

Deform
to fit the grid in the shape
Meshing

Structured meshes

Shape
Meshing

Structured meshes

Shape

Advancing front
Meshing

Shape

Advancing front

Structured meshes
Meshing

Structured meshes

Shape

Advancing front
Meshing

Structured meshes

Shape
Meshing

Structured meshes

Shape

Add grid
Meshing

Structured meshes

Shape

Add grid

Triangulate
Meshing

Structured meshes

Shape

Triangulate

Uniform mesh
Meshing

Structured meshes

Shape
Meshing

Structured meshes

Shape

Triangulate

Adaptive mesh
Meshing

Delaunay mesh refinement [Ruppert]

Unstructured mesh

Use Delaunay (good angles property)

Add vertices
Meshing

Delaunay mesh refinement [Ruppert]

Input: PSLG
Meshing

Input: PSLG

Delaunay mesh refinement

[Ruppert]
Meshing

Delaunay mesh refinement

[Def: Edge encroached by vertex if inside diametral circle]
Meshing

Input: PSLG

Delaunay

Split at middle

Delaunay mesh refinement [Ruppert]
Meshing
Input: PSLG
Delaunay
Delaunay mesh refinement [Ruppert]
Meshing

Input: PSLG

Delaunay mesh refinement [Ruppert]
Meshing

Input: PSLG

Delaunay

Delaunay mesh refinement [Ruppert]
Meshing

Input: PSLG

Delaunay refinement

Delaunay mesh refinement

[Ruppert]
Meshing

Input: PSLG

Delaunay refinement

Delaunay mesh refinement [Ruppert]
Meshing

Delaunay mesh refinement

Input: PSLG

Delaunay refinement

[Ruppert]
Meshing

Input: PSLG

Delaunay mesh refinement [Ruppert]
Meshing

Delaunay mesh refinement

[ Ruppert ]

Input: PSLG

Delaunay refinement
Meshing

Delaunay mesh refinement

[Ruppert]

Input: PSLG

Delaunay refinement

Small angle

Add circumcenter
Meshing

- Delaunay mesh refinement
- Input: PSLG
- Delaunay refinement
- Small angle
- Add circumcenter

[Ruppert]
Meshing

Input: PSLG

Delaunay refinement

Delaunay mesh refinement  [Ruppert]
Meshing

Input: PSLG

Delaunay refinement

Angle is multiplied by 2

Delaunay mesh refinement [Ruppert]
Meshing

Delaunay mesh refinement

Input: PSLG

Delaunay refinement

Small angle
But circumcircle encroached
Split edge
Meshing

Delaunay mesh refinement [Ruppert]

Input: PSLG

Delaunay refinement

Small angle

But circumcircle encroached

Split edge
Meshing

Delaunay mesh refinement  [Ruppert]

Input: PSLG

Delaunay refinement
Meshing

Delaunay mesh refinement

Input: PSLG

Delaunay refinement

[Ruppert]
Meshing

Input: PSLG

Delaunay mesh refinement

[Ruppert]
Meshing

Input: PSLG

Delaunay mesh refinement

[Ruppert]
Meshing

Delaunay mesh refinement

Input: PSLG

Delaunay refinement

Output: Mesh with angle guarantees
Small angles means \( < \alpha < 20^\circ \)

Theorem: algorithm terminates with mesh of size \( O(\text{optimal}) \)
Meshing

Delaunay mesh optimization

Lloyd iteration
Meshing

Delaunay mesh optimization

Lloyd iteration
Meshing

Delaunay mesh optimization

Lloyd iteration
Meshing

Delaunay mesh optimization

LLoyd iteration

Move to barycenter

Clip by some boundary
Meshing

Delaunay mesh optimization

LLoyd iteration
Meshing

Delaunay mesh optimization

Lloyd iteration
Meshing

Delaunay mesh optimization

Lloyd iteration
Meshing

Delaunay mesh optimization

LLoyd iteration

Reach a nice point distribution
Meshing

Delaunay mesh optimization

Alternate

Delaunay mesh refinement

Lloyd smoothing or different kind of smoothing
Figure 1.6: CVT mesh optimization. In 2D (top), (left) a 2D Delaunay mesh $M_2$ generated by Delaunay refinement, (center) $M_2$ optimized with CVT, and (right) $M_2$’s Voronoi diagram. In 3D (bottom), (left) a 3D Delaunay mesh $M_3$ generated by Delaunay refinement, (center) $M_3$ optimized with CVT, and (right) $M_3$’s slivers (tetrahedra with dihedral angles smaller than 5°).
Meshing

Delaunay mesh optimization

Delaunay Refinement (DR)
Approximation: 0.001
0.72 178.56

DR + Optimization (NODT)
5.06 171.2

DR + Optimization (NODT) + Sliver perturbation
15.03 157.22

1256 slivers < 15 deg

55 slivers < 15 deg

0 sliver < 15 deg
Meshing

Delaunay mesh optimization

Delaunay Refinement

DR & Lloyd

DR & ODT

Us

Random

17°

22°

25°
Meshing

Constraints: edges and faces

Point to insert may be encroached by edges or faces