Incremental Construction of Constrained Delaunay Triangulations

Jonathan Richard Shewchuk

Computer Science Division
University of California
Berkeley, California, USA
The Delaunay Triangulation

An edge is \textit{locally Delaunay} if the two triangles sharing it have no vertex in each others’ circumcircles.

A \textit{Delaunay triangulation} is a triangulation of a point set in which every edge is locally Delaunay.
Constraining Edges

Sometimes we need to force a triangulation to contain specified edges.

- Nonconvex shapes; internal boundaries
- Discontinuities in interpolated functions
2 Ways to Recover Segments

Conforming Delaunay triangulations

- Edges are all locally Delaunay.
- Worst-case input needs $\Omega(n^2)$ to $O(n^{2.5})$ extra vertices.

Constrained Delaunay triangulations (CDTs)

- Edges are locally Delaunay or are domain boundaries.
Goal

Input: planar straight line graph (PSLG)

Output: constrained Delaunay triangulation (CDT)

Every edge is locally Delaunay except segments.
Randomized Incremental CDT Construction

Start with Delaunay triangulation of vertices.
Randomized Incremental CDT Construction

Start with Delaunay triangulation of vertices.
Do segment location. Insert segment.
Start with Delaunay triangulation of vertices. Do segment location. Insert segment.
Randomized Incremental CDT Construction

Start with Delaunay triangulation of vertices.
Do segment location.
Insert segment.
Randomized Incremental CDT Construction

Start with Delaunay triangulation of vertices. Do segment location. Insert segment.
Randomized Incremental CDT Construction

Start with Delaunay triangulation of vertices.
Do segment location.
Insert segment.
Randomized Incremental CDT Construction

Start with Delaunay triangulation of vertices.
Do segment location.
Insert segment.
Randomized Incremental CDT Construction

Start with Delaunay triangulation of vertices. Do segment location. Insert segment.
Topics

Inserting a segment in expected linear time.

Randomized incremental CDT construction in expected $\Theta(n \log^2 k)$ time.

- Upper bound: Agarwal/Arge/Yi.
- Lower bound: new.

Inserting a polygon into a 3D CDT.
CDT Construction Algorithms

\( O(n \log n) \) time.

Plane sweep [Seidel 1988].
\( O(n \log n) \) time.

Randomized incremental segment insertion.
Expected \( O(n \log n + n \log^2 k) \) time.

\( \# \) of vertices \( \# \) of segments
Why Incremental?

It’s what everyone implements in practice. (Easiest to implement.)

Leverages the best DT implementations.

The $n \log^2 k$ term is pessimistic.
Topics

Inserting a segment in expected linear time.

Randomized incremental CDT construction in expected $\Theta(n \log^2 k)$ time.

- Upper bound: Agarwal/Arge/Yi.
- Lower bound: new.

Inserting a polygon into a 3D CDT.
Segment Insertion

cavity
cavity
Edge Flips
Chew’s Algorithm

Computes DT of a convex polygon in expected linear time.

Randomized incremental vertex insertion
Backward Analysis

Idea: analyze an algorithm as if it were running backward in time.

Pretend we remove a randomly chosen vertex from the final triangulation. Its expected degree is $< 4$.

Randomized incremental vertex insertion
Backward Analysis

Idea: analyze an algorithm as if it were running backward in time.

Pretend we remove a randomly chosen vertex from the final triangulation. Its expected degree is < 4.

Expected time to insert one vertex = constant.

Expected time to compute Del tri = linear.
Retriangulating a Segment Cavity
How Our Algorithm Differs from Chew’s

Polygons with dangling edges.

Always insert segment endpoints first.

An intermediate polygon can self-intersect.

CDT triangles are deleted for 2 reasons: new vertex in circumcircle; new edges cross.
Topics

Inserting a segment in expected linear time.

Randomized incremental CDT construction in expected $\Theta(n \log^2 k)$ time.

- Upper bound: Agarwal/Arge/Yi.
- Lower bound: new.

Inserting a polygon into a 3D CDT.
Coupon Collecting

Shuffle numbered coupons in random order.

How often do we collect a coupon preceded by every larger coupon?

Expected $\Theta(\log k)$ times.
Coupon Collecting

Shuffle numbered coupons in random order.

How often is the first collection of a coupon preceded by (one of) every larger coupon?

Expected $\Theta(\log^2 k)$ times.
Lower Bound Example

$\Theta(\sqrt{k})$ pulling vertices

$k$ segments

$\Theta(n)$ pushing vertices
Lower Bound Example

$\Theta(\sqrt{k})$ pulling vertices

$k$ segments

$\Theta(n)$ pushing vertices
Lower Bound Example

\( \Theta(\sqrt{k}) \) pulling vertices

\( k \) segments

\( \Theta(n \log^2 k) \) edges deleted.

\( \Theta(n) \) pushing vertices
Topics

Inserting a segment in expected linear time.

Randomized incremental CDT construction in expected $\Theta(n \log^2 k)$ time.

- Upper bound: Agarwal/Arge/Yi.
- Lower bound: new.

Inserting a polygon into a 3D CDT.
Polygon Insertion
Some Polyhedra Have No Tetrahedralization

Schönhardt’s polyhedron
Some Polyhedra Have No Tetrahedralization

Any four vertices of Schönhhardt’s polyhedron yield a tetrahedron that sticks out a bit.
Input: A Piecewise Linear Complex

PLC
A set of vertices, segments, and polygons.

CDT
Each polygon appears as a union of triangular faces.
The Delaunay Triangulation

An edge is *locally Delaunay* if the two triangles sharing it have no vertex in each others’ circumcircles.

A triangular face is *locally Delaunay* if the two tetrahedra sharing it have no vertex in each others’ circumspheres.
Constrained Delaunay Triangulations (CDTs)

A tetrahedralization of a polyhedron/PLC is a CDT if every triangular face not included in an input polygon is locally Delaunay.
The CDT Theorem (makes 3D CDTs useful)

Say that a PLC is *edge–protected* if every segment has an enclosing ball that contains no vertex but the segment’s endpoints.

**Theorem:**
Every edge–protected PLC has a CDT.
Bistellar Flips
Seidel’s Parabolic Lifting Map

The 3D DT matches the lower convex hull of the vertices lifted onto a paraboloid in $E^4$. 
As the lifted vertices move vertically, use flips to maintain the lower convex hull.
A Lifted CDT

constraining segment

reflex edge

Lifted triangulation is convex everywhere except at constraining segments.
“Polygon” Insertion in 1D by Flips

Let’s insert this “polygon”!

Lifted vertices rise at a velocity proportional to distance from “polygon.”

1D triangulation

CDT
“Polygon” Insertion in 2D by Flips

Lift vertices at speed proportional to distance from new segment.

Maintain lower convex hull with flips.
Running Time for Polygon Insertion

Worst-case time to insert one polygon:

\( O(n^2 \log n) \)
Running Time: Lower Bound

$\Theta(n^2)$ tetrahedra cut
Running Time for Polygon Insertion

Worst–case time to insert one polygon:

\[ O(n^2 \log n) \]

Worst–case time to insert any number of polygons and construct a CDT:

\[ O(n^2 \log n) \]
Every bistellar flip either creates or deletes an edge. Once deleted, an edge never reappears.
Open Problems (Theoretical)

Does any incremental insertion algorithm run in $\Theta(n \log n)$ time? E.g. biased random?

Upper bound does not depend on Delaunay property! Other optimal triangulations?