CS6100: Topics in Design and Analysis of Algorithms

Delaunay Triangulation

John Augustine
Triangulation of a Planar Point Set

A triangulation of $P$ is planar subdivision of the plane in which (i) end points of line segments are in $P$ and (ii) no line segment (connecting points in $P$) can be added without destroying planarity.

→ Bounded faces are triangles — hence triangulation.

→ Convex hull edges always included.

**Theorem 1.** Let $P$ have $n$ points (not all collinear) and $k$ points on the convex hull.

- Number of triangular faces $m$ is $2n - 2 - k$.
- Number of edges is $3n - 3 - k$.

**Proof Idea.** Total # of faces is $m + 1$ where 1 is unbounded and others are triangles. The total number of edges is $\frac{3m+k}{2}$ (why?). Results follow by applying Euler’s formula.
How do we compare the skinness triangulations?

Define **angle-vector** $A(T)$ of a triangulation $T$ as vector of angles sorted in non-decreasing order.

We use lexicographic comparison on the angle vectors to compare skinness of triangulations.

More precisely, let $A(T) = \{\alpha_1, \alpha_2, \ldots\}$ and $A(T') = \{\alpha'_1, \alpha'_2, \ldots\}$. We say that $A(T) > A(T')$ if for some $i$,

$$\forall j < i, \quad \alpha_j = \alpha'_j \quad \text{and} \quad \alpha_i > \alpha'_i.$$ 

We say that a triangulation $T$ is **angle-optimal** if $\nexists T'$ such that $A(T') > A(T)$.
Flipping an Illegal Edge

Let $e$ be a non-boundary edge between two triangles that form a convex quadrilateral. This edge can be “flipped” to $e'$. If this flip locally improves the angle vector, then we call $e$ illegal.

![Diagram of edge flip]

**Observation 2.** If $e$ is an illegal edge in $\mathcal{T}$ and $\mathcal{T}'$ is obtained by flipping $e$, then,

$$A(\mathcal{T}') > A(\mathcal{T}).$$
How to check if an edge is illegal?

**Lemma 3.** In the above figure, if $p_l$ is inside circle through $p_i$, $p_k$, and $p_j$, then the edge (as indicated) is illegal. (This follows from Thales’ Theorem.)

**Theorem 4 (Thales’ Theorem).** In Figure 1,

$$\angle arb > \angle apb = \angle aqb > \angle asb.$$
Legal Triangulation

A legal triangulation does not contain any illegal edge.

Recall Observation 2.

⇒ A repeated flipping of illegal edges will terminate and make eventually make the triangulation legal.

Observation 5. Every angle-optimal triangulation is a legal triangulation.

Is the converse true?

Towards Delaunay Graph

Consider a set $P$ of points/sites in the plane. Let $\text{Vor}(P)$ be its Voronoi diagram. Each site $p$ has a cell $\mathcal{V}(p)$ associated with it.

The **dual graph** of $\text{Vor}P$ in which the vertices are the sites and two sites $p_i$ and $p_j$ are connected iff their cells $\mathcal{V}(p_i)$ and $\mathcal{V}(p_j)$ share an edge in $\text{Vor}(P)$. 

\[ \text{Vor}(P) \]
Delaunay Graph

A straight line embedding of the dual graph on the set $P$ of points is called the Delaunay Graph (denoted $DG(P)$).

Theorem 6. The Delaunay graph is a plane graph.
Proof of Theorem 6

Proof. Recall that the perpendicular bisector between two sites appears in Vor(P) iff $\exists$ a circle touching the sites but not enclosing any other site.

In other words, $\overline{p_ip_j}$ is in $DG(P)$ iff $\exists$ closed disc $C_{ij}$ with $p_i$ and $p_j$ on the boundary and no other site is contained in it.

Let $t_{ij}$ be the triangle formed by $p_i$, $p_j$, and center of (some) $C_{ij}$.

Note that edge of $t_{ij}$ between $p_i$ and center of $C_{ij}$ is inside $\mathcal{V}(p_i)$.
Suppose there is another edge \( p_k p_l \) also in \( DG(P) \) such that \( p_i p_j \) and \( p_k p_l \) intersect.

(As defined for \( p_i p_j \), define \( C_{kl} \) and \( t_{kl} \) for \( p_k p_l \).)

If \( p_k p_l \) intersected \( p_i p_j \), it must also intersect one other edge \( e \) of \( t_{ij} \). \( \text{Why?} \) (Because \( p_k \) and \( p_l \) are outside \( C_{ij} \) and therefore outside \( t_{ij} \).

Likewise an edge \( e' \) of \( t_{kl} \) must intersect \( p_i p_j \).

Notice that this implies that one of the edges of \( t_{ij} \) incident to center of \( C_{ij} \) and one of the edges of \( t_{kl} \) incident to center of \( C_{kl} \) must intersect.

But, those edges must be contained within their respective Voronoi cells, which is a contradiction. \( \square \)
A vertex $v \in \text{Vor}(P)$ corresponds to a face in $\mathcal{DG}(P)$. If $v$ has degree $k$, then, the corresponding face is a $k$-gon. Furthermore, it is a \underline{convex} $k$-gon. \textbf{Why?}

A \underline{Delaunay Triangulation} is a triangulation obtained by adding edges to a delaunay graph.
Let $P$ be a set of points in the plane.

**Theorem 7.**

1. Three points form a triangle in $DG(P)$ iff the circle through those three points does not enclose any other point in $P$.
2. Two points form an edge in $DG(P)$ iff $\exists$ a circle through those two points that does not enclose any other point in $P$.

**Theorem 8.** A triangulation $\mathcal{T}$ is a Delaunay triangulation iff the circumcircle of any triangle in $\mathcal{T}$ does not contain any point in its interior.

**Theorem 9.** A triangulation $\mathcal{T}$ is legal iff $\mathcal{T}$ is a Delaunay triangulation.
Proof of Theorem 9

Proof. Easy to see that any Delaunay triangulation is legal. So focus on opposite direction.

Assume for contradiction that $T$ is a legal triangulation, but not a Delaunay triangulation.

From Theorem 7, $\exists p_i, p_j, p_k$ such that circumcircle $C(p_i, p_j, p_k)$ contains point $p_l \in P$ in its interior.

Let $e = p_ip_j$ be the edge chosen so that $\triangle p_ip_jp_l$ does not intersect $\triangle p_ip_jp_k$.

Notice that $\triangle p_ip_jp_l$ cannot be a triangle in $T$ as $e$ can be flipped to improve angle-optimality. Therefore,
∃ \( p_m \neq p_l \) such that \( \triangle p_ip_jp_m \) is in \( \mathcal{T} \). (Note that \( e \) cannot be a boundary edge.)

From Lemma 3 (since \( \mathcal{T} \) is a legal triangulation), \( p_m \) is outside \( C(p_i, p_j, p_j) \).

Let \( \overline{p_jp_m} \) be the edge such that \( \triangle p_jp_mp_l \) does not intersect \( p_jp_mp_l \). But, by Thales’ Theorem,

\[
\angle p_jp_lp_i > \angle p_jp_mp_i,
\]

a contradiction as it allows \( \overline{p_jp_m} \) to be flipped. \( \Box \)

**Theorem 10.**

1. Any angle optimal triangulation is a Delaunay triangulation.
2. Any Delaunay triangulation maximizes the minimum angle.
A Randomized Incremental Construction of Delaunay Triangulation

Let \( P = \{p_0, p_1, \ldots, p_n\} \) be a set of points in the plane and let \( p_0 \) be the highest point in \( P \), whereas, the rest of the points are randomly permuted.

Enclose \( P \) in \( \triangle p_0p_{-1}p_{-2} \), where \( p_{-1} \) and \( p_{-2} \) are dummy points. \( \triangle p_0p_{-1}p_{-2} \) must be large enough so that \( p_{-1} \) and \( p_{-2} \) don’t lie in any circle defined by three points in \( P \).

We start with \( \triangle p_0p_{-1}p_{-2} \) as the current triangulation and incrementally add points taken from a random permutation \( \{p_1, p_2, \ldots, p_n\} \) of the remaining points.
Two Cases

$p_r$ lies in the interior of a triangle

$p_r$ falls on an edge

In either case, illegal edges can be introduced, so we legalize them by testing each potentially illegal edge and flipping illegal edges.

Note that all edges created are incident to the newly inserted point $p_r$. 
Pseudocode

Algorithm DELAUNAY TRIANGULATION(P)

Input. A set P of n + 1 points in the plane.
Output. A Delaunay triangulation of P.

1. Let p₀ be the lexicographically highest point of P, that is, the rightmost among the points with largest y-coordinate.

2. Let p⁻¹ and p⁻² be two points in ℝ² sufficiently far away and such that P is contained in the triangle p₀p⁻¹p⁻².

3. Initialize ℱ as the triangulation consisting of the single triangle p₀p⁻¹p⁻².

4. Compute a random permutation p₁, p₂, ..., pₙ of P \ {p₀}.

5. for r ← 1 to n

6.   do (* Insert pᵣ into ℱ: *)

7.     Find a triangle pᵢpⱼpₖ ∈ ℱ containing pᵣ.

8.     if pᵣ lies in the interior of the triangle pᵢpⱼpₖ

9.         then Add edges from pᵣ to the three vertices of pᵢpⱼpₖ, thereby splitting pᵢpⱼpₖ into three triangles.

10.        LEGALIZEEDGE(pᵣ, pᵢpⱼ, ℱ)

11.        LEGALIZEEDGE(pᵣ, pⱼpₖ, ℱ)

12.        LEGALIZEEDGE(pᵣ, pₖpᵢ, ℱ)

13.     else (* pᵣ lies on an edge of pᵢpⱼpₖ, say the edge pᵢpⱼ *)

14.         Add edges from pᵣ to pₖ and to the third vertex pᵢ of the other triangle that is incident to pᵢpⱼ, thereby splitting the two triangles incident to pᵢpⱼ into four triangles.

15.        LEGALIZEEDGE(pᵣ, pᵢpᵢ, ℱ)

16.        LEGALIZEEDGE(pᵣ, pᵢpⱼ, ℱ)

17.        LEGALIZEEDGE(pᵣ, pⱼpₖ, ℱ)

18.        LEGALIZEEDGE(pᵣ, pₖpᵢ, ℱ)

19.     Discard p⁻¹ and p⁻² with all their incident edges from ℱ.

20. return ℱ
In line 7 of the pseudocode in the previous page, we perform a point location.

We use a DAG $\mathcal{D}$ similar to the DAG used in point location in planar subdivision.

Each node in $\mathcal{D}$ corresponds to a $\triangle$ that was created at some point.

Searching for a point $p_r$ would entail going down the DAG $\mathcal{D}$ through a sequence of nodes corresponding to all triangles created before step $r$ (and possibly destroyed) that contain $p_r$. 
split $\Delta_1$

flip $p_i p_j$

flip $p_i p_k$

$\Delta_1$ $\Delta_2$ $\Delta_3$

$\Delta_1$ $\Delta_2$ $\Delta_3$

$\Delta_1$ $\Delta_2$ $\Delta_3$

$\Delta_1$ $\Delta_2$ $\Delta_3$
To show correctness, recall that every new edge added after the insertion of a point \( p_r \) is incident to \( p_r \).

The part that gets changed is limited to triangles with two edges incident to \( p_r \). Therefore, the correctness of the algorithm follows from the following lemma.
Lemma 11. Every new edge created when \( p_r \) is added is a Delaunay edge of \( \{p_{-2}, p_{-1}, p_0, p_1, \ldots, p_r\} \).

Proof Sketch.

\( \triangle p_j p_i p_l \) is a Delaunay triangle of \( \{p_{-2}, p_{-1}, p_0, p_1, \ldots, p_r-1\} \). Therefore \( \exists \) a circle \( C \) through \( p_j p_i p_l \) devoid of points in \( \{p_{-2}, p_{-1}, p_0, p_1, \ldots, p_{r-1}\} \). Shrink \( C \) until it touches only \( p_r \) and \( p_l \) — this can be done — thus proving that \( p_r p_l \) is a Delaunay Edge. \( \square \)
Lemma 12. Expected number of triangle created by the algorithm is $9n + 1$.

Proof. Recall what happens when we insert $p_r$. First, one of the following two cases, and subsequently, we repeatedly legalize edges (until none are left).

First, three (or four) triangles are formed. Subsequently, each edge we “legalize” will add 2 triangles.

Therefore, if $k$ is the number of edges incident to $p_r$ then, $\leq 2(k - 3) + 3 = 2k - 3$ new triangles are formed.
When \( p_r \) is added, the degree of \( p_r \) is 6. This follows from:

- \( 6r \) being the total degree, which in turn follows from
- the bound on the number of edges in the Delaunay Graph,
- which equals the number of Voronoi edges.

\[
\begin{align*}
E[\# \text{ of } \triangle \text{s created in step } r] & \leq E[2k - 3] \\
& = 2 \times 6 - 3 = 9.
\end{align*}
\]  

Using linearity of expectation over all iterations, the expected number of triangles added is \( 9n + 1 \), where the extra triangle is \( \triangle p_0p_{-1}p_{-2} \).
Theorem 13. The Randomized Incremental Algorithm for the Delaunay Triangulation of $n$ points takes $O(n \log n)$ time and $O(n)$ space, both on expectation.

Proof Sketch. Space follows from nodes in $D$ representing triangles created, which is $9n + 1$ on expectation.

Not counting point location, the creation of each triangle takes $O(1)$ time, so the total time will be $O(n) + \text{time for point locations}$ (on expectation).

Recall: Searching for a point $p_r$ would entail going down the DAG $D$ through a sequence of nodes corresponding to all triangles created before step $r$ (and possibly destroyed) that contain $p_r$.

So each triangle that was created is visited once for each point in it. In other words, . . .

Let $K(\Delta) \subseteq P$ be the points inside triangle $\Delta$. Time
for point locations is

\[ \sum_{\Delta} K(\Delta). \]

Roughly speaking, each triangle created in round \( r \) has \( O(n/r) \) points of \( P \) in it. Therefore,

\[ \sum_{\Delta} K(\Delta) \leq \sum_{r} O(n/r) \]

\[ \in O(n \log n), \]