Randomized Incremental Construction

`Topics:` incremental algorithm, growing star, number of flips, history dag, search time.

Incremental algorithm. We obtain a faster algorithm for constructing Delaunay triangulations if we interleave flipping edges with adding points. Denote the points in $S \subseteq \mathbb{R}^2$ as $p_1, p_2, \ldots, p_n$ and assume general position. When we add a point to the triangulation it can either lie inside or outside the convex hull of the preceding points. To reduce the outside to the inside case we start with a triangulation D_0 that consists of a single and sufficiently large triangle xyz. Define

$$S_i \quad = \quad \{x, y, z, p_1, p_2, \ldots, p_i\},$$

and let D_i be the Delaunay triangulation of S_i. The algorithm is a `for`-loop which adds the points in sequence. After adding a point we use flips to get to the Delaunay triangulation before the next point is added.

```
for i = 1 to n do
  find triangle τ_{i-1} ∈ D_{i-1} that contains p_i;
  add p_i by splitting τ_{i-1} into three triangles;
  while ∃ab that is not locally delone do
    flip ab to other diagonal cd
  endwhile
endfor.
```

The two elementary operations used by the algorithm are shown in Figure 14. Both pictures can be interpreted as the projection of a tetrahedron, though from different angles. For this reason the addition of a point inside a triangle is sometimes called a 1-to-3 flip, while an edge-flip is sometimes also called a 2-to-2 flip.

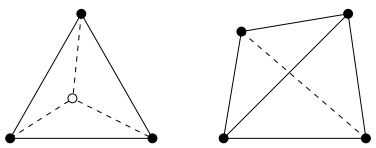

Figure 14: To the left the hollow vertex is added to the triangle. To the right the solid diagonal is flipped to the dashed one.

Growing star. Note that every new triangle in D_i has p_i as one of its vertices. Indeed, abc is a triangle in D_i iff $a, b, c \in S_i$ and the circumcircle is empty of points in S_i. But if p_i is not one of the vertices then $a, b, c \in S_{i-1}$ and if the circumcircle is empty of points in S_i then it is also empty of points in S_{i-1}. So abc is also a triangle in D_{i-1}. This implies that all flips during the insertion of p_i happen right around p_i.

We need some definitions. The *star* of p_i consists of all triangles that contain p_i. The *link* of p_i consists of all edges of triangles in the star that are disjoint from p_i. Both concepts are illustrated in Figure 15. Right after

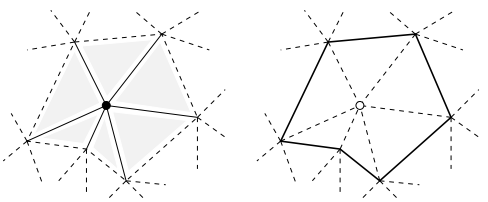

Figure 15: The star of the solid point to the left and the link of the same point to the right.

p_i is added, the link consists of only three edges, namely the edges of the triangle that contains p_i. These edges are marked and pushed on the stack to start the edge-flipping `while`-loop. Each flip replaces a link edge by an edge with endpoint p_i. At the same time it removes one triangle in the star and one outside the star and it adds the two triangles that cover the same quadrangle to the star. The net effect is one more triangle in the star. The number of flips is therefore 3 less than the number of edges in the final link, which is the same as 3 less than the degree of p_i in D_i.

Number of flips. For now we ignore the time needed to find the triangles τ_{i-1}. The rest of the time is proportional to the number of flips needed for p_1 through p_n. We assume $p_1, p_2, \ldots, p_n$ is a randomly chosen in-

put sequence. Random does not mean arbitrary but rather that every permutation of the n points is equally likely. The expected number of flips is the total number of flips needed to construct the Delaunay triangulation for all $n!$ input permutations divided by $n!$.

Consider inserting the last point, p_n. The sum of degrees of all possible last points is the same as the sum of degrees of all points p_i in D_n. The latter is equal to twice the number of edges and therefore

$$\sum_{i=1}^{n} \deg p_i \quad \leq \quad 6n.$$

The number of flips needed to add all last points is therefore at most $6n - 3n = 3n$. The total number of flips is

$$\begin{aligned} F(n) &\leq n \cdot F(n-1) + 3n \\ &\leq 3n \cdot n!. \end{aligned}$$

Indeed, if we assume $F(n-1) \leq 3(n-1) \cdot (n-1)!$ we get $n \cdot F(n-1) + 3n = 3(n-1) \cdot n! + 3n \leq 3n \cdot n!$. The expected number of flips needed for n points is therefore at most $3n$.

There is a simple way to say the same thing. The expected number of flips for the last point is at most 3, and therefore the expected number of flips to add any point is at most 3.

History dag. To find the triangle τ_{i-1} that contains point p_i we use the evolution of the Delaunay triangulation. Instead of deleting a triangle when it is split or flipped away, we just make it the parent of the new triangles. Figure 16 shows the two operations to the

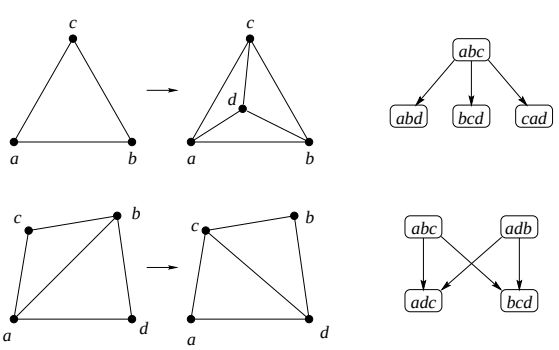

Figure 16: Splitting a triangle generates a parent with three children. Flipping an edge generates two parents which share the same two children.

left and the corresponding parent-child relations to the right. Each time we split or flip we add triangles or nodes to the growing data structure that records the history of the construction. The evolution from D_0 to D_n consists of n splits and an expected number of at most $3n$ flips. The resulting directed acyclic graph, or dag for short, therefore has an expected size of at most $1 + 3n + 2 \cdot 3n = 9n + 1$ nodes. It has a unique source, the triangle xyz, and its sinks are the triangles in D_n.

Search time. Consider adding the point p_i. To find the triangle $\tau_{i-1} \in D_{i-1}$ we search a path of triangles in the history dag that all contain p_i. The path begins as xyz and ends at τ_{i-1}. The history dag of D_{i-1} consists of i layers. Layers 0 through j represent the dag of D_j. Its sinks are the triangles in D_j and we let $\sigma_j \in D_j$ be the triangle that contains p_i. Triangles σ_0 through σ_j form a scattered subsequence of the search path. It is quite possible that some of the triangles in the scattered subsequence are the same. Let G_j be the set of triangles removed from D_j during the insertion of p_{j+1} and let H_j be the set of triangles removed from D_j during the (independent) insertion of p_i, see Figure 17. We have

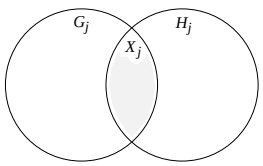

Figure 17: Regions of triangles removed from D_j by insertion of p_{j+1} and of p_i.

$\sigma_j = \sigma_{j+1}$ if G_j and H_j are disjoint. Suppose $\sigma_j \neq \sigma_{j+1}$. Then $X_j = G_j \cap H_j \neq \emptyset$ and all triangles on the portion of the path from σ_j to σ_{j+1} are generated by flips that remove triangles in X_j. The cost for searching with p_i is therefore at most proportional to the sum of card X_j, for j from 0 through $i-2$.

To write X_j in terms of other sets we consider what happens if we insert p_i into D_j and then insert p_{j+1} into the Delaunay triangulation of $S_j \cup \{p_i\}$. Let Y_j be the triangles removed during the insertion of p_{j+1}, and let Z_j be the subset of triangles that do not belong to D_j. Each triangle in Z_j is created during the insertion of p_i, so p_i must be one of its vertices. We have

$$X_j \quad = \quad G_j - (Y_j - Z_j).$$

We use lowercase letters to denote cardinalities. Because $Z_j \subseteq Y_j$ and $Y_j - Z_j \subseteq G_j$ we have

$$x_j \quad = \quad g_j - y_j + z_j\,.$$

The expected values of g_j and y_{j-1} are the same because

both count triangles removed by inserting a random j-th point. Hence

$$\begin{aligned} \mathrm{E}[\sum_{j=0}^{i-2} x_j\,] &= \sum_{j=0}^{i-2} \mathrm{E}[\,g_j\,] - \mathrm{E}[\,y_j\,] + \mathrm{E}[\,z_j\,] \\ &= \mathrm{E}[\,g_0 - g_{i-1}\,] + \sum_{j=0}^{i-2} \mathrm{E}[\,z_j\,]. \end{aligned}$$

To compute the expected value of z_j we use the fact that among $j+2$ points every pair is equally likely to be p_{j+1} and p_i. For example, if p_{j+1} and p_i are not connected by an edge in the Delaunay triangulation of $S_j \cup \{p_{j+1}, p_i\}$ then $Z_j = \emptyset$. In general, a triangle in the Delaunay triangulation of $S_j \cup \{p_i\}$ has probability at most $\frac{3}{j+1}$ to be in the star of p_i. The expected number of triangles removed by inserting p_{j+1} is at most 4. The expectation of z_j is the product: $\mathrm{E}[\,z_j\,] \leq \frac{4 \cdot 3}{j+1}$. The expected length of the search path for p_i is

$$\sum_{j=0}^{i-2} \mathrm{E}[\,x_j\,] \;\leq\; \sum_{j=0}^{i-2} \frac{12}{j+1} \;\leq\; 1 + 12\ln(i-1).$$

The expected total time spend on searching in the history dag is $\sum \mathrm{E}[\,x_j\,] \leq c \cdot n \log n$. We summarize the result.

THEOREM 7. Given any set of n points in the plane, the randomized incremental algorithm constructs the Delaunay triangulation in expected time $\mathrm{O}(n \log n)$ and in expected memory $\mathrm{O}(n)$.

Bibliographic notes. The randomized incremental algorithm of this lecture is due to Guibas, Knuth and Sharir [3]. It is generalized to three and higher dimensions by Edelsbrunner and Shah [2]. All this is based on earlier work on randomized algorithms and in particular on the methods introduced by Clarkson and Shor [1]. The arguments used to bound the expected number of flips and the expected search time are examples of the backwards analysis introduced by Raimund Seidel [4].

[1] K. L. CLARKSON AND P. W. SHOR. Applications of random sampling in computational geometry. *Discrete Comput. Geom.* **4** (1989), 387–421.

[2] H. EDELSBRUNNER AND N. R. SHAH. Incremental topological flipping works for regular triangulations. *Algorithmica* **15** (1996), 223–241.

[3] L. J. GUIBAS, D. E. KNUTH AND M. SHARIR. Randomized incremental construction of Delaunay and Voronoi diagrams. *Algorithmica* **7** (1992), 381–413.

[4] R. SEIDEL. Backwards analysis of randomized geometric algorithms. In *New Trends in Discrete and Computational Geometry*, J. Pach (ed.), Springer-Verlag, Berlin, 1993, 37–67.