A NOTE ON POINT LOCATION IN RANDOM DELAUNAY TRIANGULATIONS

Luc Devroye and Binhai Zhu
School of Computer Science
McGill University
Montreal, Canada H3A 2A7

ABSTRACT. We consider a standard data structure for a Delaunay triangulation in the plane based upon n independent points drawn from a common density f that is bounded away from 0 and ∞ on a convex set. A simple easy-to-implement (but, of course, suboptimal) heuristic for point location is shown to take expected time $O(n^{1/3})$.

KEYWORDS AND PHRASES. Voronoi diagram, Delaunay triangulation, point location, probabilistic analysis of algorithms, computational geometry.

CR CATEGORIES: 3.74, 5.25, 5.5.

Address: School of Computer Science, McGill University, 3480 University Street, Montreal, Canada H3A 2K6. Research sponsored by NSERC Grant A3456. Email: `lucdevroye@gmail.com`

Introduction and main result.

Assume that we are given n points $X_1, \ldots, X_n$ in the plane together with the standard implementation of the Delaunay triangulation for these points (Okabe, Boots and Sugihara, 1992). That is, in PASCAL terminology, the information is stored as points, edges and triangles, linked by neighborhood information and would have the following rough `type` definition:

```
point:      RECORD         x,y:  real
                           neighbors:  edgelist END
edgelist:   RECORD         next:  ↑edgelist
                           key:  ↑edge END
edge:       RECORD         pt1,pt2:  ↑point
                           tr1,tr2:  ↑triangle END
triangle:   RECORD         ed1,ed2,ed3:  ↑edge END
delaunay:   ARRAY[1..n]OF ↑point
```

No other structure is assumed on top of this simple graph-like object. The objective is to investigate how fast we can perform point location for a query point x when we are not allowed to further preprocess the data. A simple heuristic is proposed that has an acceptable expected complexity and is very easy to implement:

STEP 1. Select m points $Y_1, \ldots, Y_m$ at random and without replacement from $X_1, \ldots, X_n$.

STEP 2. Determine the index $i \in \{1, \ldots, m\}$ for which $\|Y_i - x\|$ is minimal, and call it j. Define $Y = Y_j$.

STEP 3. Locate the triangle to which x belongs by traversing all triangles crossed by the line segment (Y, x): this is easy to do by the adjacency list implementation mentioned above.

The time T taken by the algorithm is $\Theta(m)$ (for the distance computations) plus $\Theta(N)$, where N is the number of triangles visited by the line segment (Y, x). The notation $\Theta(.)$, $O(.)$ and $\Omega(.)$ is as in standard textbooks on data structures (see, e.g., Cormen, Leiserson and Rivest, 1990). Since Y is random, T is random as well. Our main result is the following.

THEOREM. *If $X_1, \ldots, X_n$ are independently drawn from a distribution with density f on a compact set $C \subseteq \mathrm{R}^2$, if $\alpha \stackrel{\text{def}}{=} \inf_C f(x) > 0$ and $\beta \stackrel{\text{def}}{=} \sup_C f(x) < \infty$, and if the query point X is independent of $X_1, \ldots, X_n$, then the expected time of the simple algorithm given above is bounded by*

$$c_1 m + c_2 \sqrt{n/m} \ ,$$

where $c_1, c_2 > 0$ are universal constants possibly depending upon the geometrical properties of C. In particular, the expected time is $O(n^{1/3})$ if $m = \Theta(n^{1/3})$.

Proof of the Theorem.

The proof rests on the following Lemma.

LEMMA (BOSE AND DEVROYE, 1995). *If $X_1, \ldots, X_n$ are as in the Theore, and if $\mathcal{L}$ is a fixed line segment of length $|\mathcal{L}|$, then the expected number of triangles or edges of the Deaunay triangulation (for $X_1, \ldots, X_n$) crossed by $\mathcal{L}$ is bounded by*

$$c_3 + c_4 |\mathcal{L}| \sqrt{n} \ ,$$

where c_3, c_4 are universal positive constants not depending upon $\mathcal{L}$ or n.

To use this Lemma for a random line segment $\mathcal{L}$, we must make sure that $\mathcal{L}$ is independent of $X_1, \ldots, X_n$. This is not the case in our example. For this reason, we make a small detour. Let $\mathcal{D}$ be the Delaunay triangulation for $X_1, \ldots, X_n$, and let $\mathcal{D}_m$ be the Delaunay triangulation for $\{X_1, \ldots, X_n\} - \{Y_1, \ldots, Y_m\}$. Then $\mathcal{L} = (Y, X)$ is independent of the $n - m$ data points defining $\mathcal{D}_m$. Thus, by the Lemma, the expected number of triangles or edges of $\mathcal{D}_m$ crossed by $\mathcal{L}$ is not more than

$$c_3 + c_4 \mathbf{E}\|Y - X\| \sqrt{n - m} \ .$$

The number of triangles or edges of $\mathcal{D}$ crossed by $\mathcal{L}$ is not more than that for $\mathcal{D}_m$ plus the sum S of the degrees of $Y_1, \ldots, Y_m$ in the Delaunay triangulation $\mathcal{D}$. The expected value of S is, by symmetry, m times the expected degree of Y_1. By the planarity of $\mathcal{D}$, we know that sum of all degrees of $X_1, \ldots, X_n$ is twice the number of edges, which does not exceed $6n$. Therefore, the expected degree of X_1 or Y_1 does not exceed 6. Combining all this shows that

$$\mathbf{E}T \leq O(m) + O(\sqrt{n}) \mathbf{E}\|Y - X\| \ .$$

We note here that the maximal degree of the X_i's is known to be $\Theta(\log n / \log \log n)$ on average (Bern, Eppstein and Yao, 1991), so that the detour suggested above was indeed necessary to avoid an additional logarithmic factor.

We conclude the proof by showing that $\mathbf{E}\|Y - X\| = O(1/\sqrt{m})$. Let $S_{x,t}$ denote the circle of radius t centered at x, and let $\lambda(A) = \int_A dx$. Fix $x \in C$. Note that there exist positive constants γ and t_0 such that for $t < t_0$, $\lambda(S_{x,t} \cap C) \geq \gamma t^2$ (this is obvious if C is a circle or rectangle, and is easy to show for general convex C of nonzero area). Then, if D is the maximal distance between any two points of C,

$$\begin{aligned}
\mathbf{E}\|Y - x\| &= \int_0^\infty \mathbf{P}\{\|Y - x\| > t\} \, dt \\
&= \int_0^\infty \mathbf{P}^m \{\|X_1 - x\| > t\} \, dt \\
&= \int_0^\infty (1 - \mathbf{P}\{\|X_1 - x\| \leq t\})^m \, dt \\
&\leq \int_0^\infty e^{-m\mathbf{P}\{\|X_1 - x\| \leq t\}} \, dt \\
&\leq \int_0^\infty e^{-m\alpha\lambda(S_{x,t} \cap C)} \, dt
\end{aligned}$$

$$
\begin{aligned}
&\leq \int_0^{t_0} e^{-m\alpha\gamma t^2}\,dt + \int_{t_0}^{D} e^{-m\alpha\gamma {t_0}^2}\,dt \\
&\leq \int_0^{\infty} e^{-m\alpha\gamma t^2}\,dt + De^{-m\alpha\gamma {t_0}^2} \\
&= \sqrt{\frac{\pi}{2\alpha\gamma m}} + De^{-m\alpha\gamma {t_0}^2}\ .
\end{aligned}
$$

This shows that $\mathbf{E}T = O(m + \sqrt{n/m})$.

Remarks.

The Theorem may also be used to obtain a very simple on-line algorithm for insertion and deletion in a Delaunay triangulation with $O(n^{1/3})$ expected time per operation. Clearly, this is not as good as the $O(\log n)$ expected time fully dynamic algorithms of Devillers, Meiser and Teillaud (1991, 1992), but the data structure is also less complicated.

Using the given point location, a Delaunay triangulation can be constructed in $O(n^{4/3})$ expected time. Again, this is theoretically slower than some well-known $O(n \log n)$ algorithms (Shamos and Hoey, 1975; Lee and Schachter, 1980; Guibas and Stolfi, 1985) or some $O(n \log n)$ expected time randomized algorithms (Guibas, Knuth and Sharir, 1990; Boissonnat and Teillaud, 1993).

It should be noted also that we do not make use of the power of truncation and bucketing, so that the algorithm cannot be expected to compete against fine-tuned bucket methods such as those of Maus (1984), Dwyer (1986, 1987), Katajainen and Koppinen (1988), or the research group at the University of Tokyo (Ohya, Iri and Murota, 1984; Sugihara, Ooishi and Imai, 1990; Ooishi and Sugihara, 1991) which all achieve $O(n)$ expected time under certain conditions on the distribution of the data.

References.

M. Bern, D. Eppstein, and F. F. Yao, "The expected extremes in a Delaunay triangulation," in: *18th ICALP Conference, held at Madrid, Spain*, 1991.

J.-D. Boissonnat and M. Teillaud, "On the randomized construction of the Delaunay tree," *Theoretical Computer Science*, vol. 112, pp. 339–354, 1993.

P. Bose and L. Devroye, "Intersections with random Voronoi diagrams," Manuscript, School of Computer Science, McGill University, Montreal, 1995.

T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms*, MIT Press, Boston, 1990.

O. Devillers, S. Meiser, and M. Teillaud, "Fully dynamic Delaunay triangulation in logarithmic expected time per operation," in: *Proc. 2nd Workshop Algorithms Data Struct.*, vol. 519, pp. 42—53, Lecture Notes in Computer Science, Springer-Verlag, 1991.

O. Devillers, S. Meiser, and M. Teillaud, "Fully dynamic Delaunay triangulation in logarithmic expected time per operation," *Computational Geometry: Theory and Applications*, vol. 2, pp. 55–80, 1992.

R. A. Dwyer, "A simple divide-and-conquer algorithm for constructing Delaunay triangulations in $O(n \log \log n)$ expected time," in: *Proceedings of the 2nd Annual Symposium on Computational Geometry*, pp. 276–284, ACM, New York, 1986.

R. A. Dwyer, "A faster divide-and-conquer algorithm for constructing Delaunay triangulations," *Algorithmica*, vol. 2, pp. 137–151, 1987.

L. J. Guibas, D. E. Knuth, and M. Sharir, "Randomized incremental construction of Delaunay and Voronoi diagrams," in: *ICALP90*, pp. 414–431, Lecture Notes in Computer Science, vol. 443, Springer-Verlag, Berlin, 1990.

L. Guibas and J. Stolfi, "Primitives for the manipulation of general subdivisions and the computation of Voronoi diagrams," *ACM Transactions on Graphics*, vol. 4, pp. 74–123, 1985.

J. Katajainen and M. Koppinen, "Constructing Delaunay triangulations by merging buckets in quadtree order," *Fundamenta Informaticae*, vol. 11, pp. 275–288, 1988.

D.-T. Lee and B. J. Schachter, "Two algorithms for constructing the Delaunay triangulation," *International Journal of Computer and Information Sciences*, vol. 9, pp. 219–242, 1980.

A. Maus, "Delaunay triangulation and the convex hull of n points in expected linear time," *BIT*, vol. 24, pp. 151–163, 1984.

J. Moller, "Random tessellations in R^d," *Advances in Aplied robability*, vol. 21, pp. 37–73, 1989.

T. Ohya, M. Iri, and K. Murota, "Improvements of the incremental method for the Voronoi diagram with computational comparison of various algorithms," *Journal of Operations Research*, vol. 27, pp. 306–336, 1984.

T. Ohya, M. Iri, and K. Murota, "A fast Voronoi-diagram algorithm with quaternary tree bucketing," *Information Processing Letters*, vol. 18, pp. 227–231, 1984.

A. Okabe, B. Boots, and K. Sugihara, *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*, John Wiley, Chichester, England, 1992.

Y. Ooishi and K. Sugihara, "Numerically robust divide-and-conquer algorithm for constructing Voronoi diagrams," *Transactions of Information Processing Society of Japan*, vol. 32, pp. 709–720, 1991.

M. I. Shamos and D. Hoey, "Closest-point problems," in: *Proceedings of the 16th IEEE Symposium on the Foundations of Computer Science*, pp. 151–162, 1975.

K. Sugihara, Y. Ooishi, and H. Imai, "Topology-oriednted approach to robustness and its applications to several Voronoi-diagram algorithms," in: *Abstracts of the Second Canadian Conference on Computational Geometry, Ottawa*, pp. 36–39, 1990.