Algorithmica (1987) 2: 131-134

Algorithmica

Problem Section

Some Problems in Computational Geometry

Claire Mathieu[1]

In recent years, with the rise of computational geometry and an increasing number of people working on the subject (see Lee and Preparata [5] for a survey), many formerly neglected problems were solved efficiently and elegantly, but at the same time, as is natural for every expanding field, even more arose and became of interest: a few are hereafter offered.

Although the minimum spanning tree problem (MST) has long been solved (at least in theory), related questions still remain unanswered. For example, let us consider n points in the plane, with which we wish to form a "reasonable" polygon. In practice these points are on the boundary of an unknown object, and the polygon is "reasonable" if the order of the points is roughly the same on the boundary of the other object and on the polygon. A traveling salesman tour (TST) would be a nice solution, but finding it is an NP-hard problem, and so we would like an easier criterion for constructing a good polygon. Suppose that such a polygon is part of the Delaunay triangulation of the points. It contains $(n-2)$ Delaunay triangles, and the (graph-theoretic) dual of these triangles is a subtree of the Voronoi diagram, and covers $(n-2)$ Voronoi nodes (Figure 1). A good criterion of "reasonableness" might be to minimize the cost (cumulated edge-length) of this tree: whence the question of choosing $k=n-2$ nodes among $N=2n-e-2$ (number of nodes of the Voronoi diagram, e being the number of edges on the convex hull of the n data points), in order to minimize the cost of their MST (problem proposed by J. D. Boissonnat).

Another problem in connection with Delaunay graphs, appealing but deceptive in its apparent simplicity, was proposed to me by J. D. Boissonnat and H. Crapo: this is the question of proving that a planar Delaunay triangulation always has a Hamiltonian cycle—which, if true, might lead to nice heuristics for the TST problem. For the definition of a Delaunay triangulation see, for example, Shamos and Hoey [8].

Algorithms for solving geometrical problems in the plane abound, and their worst-case complexity has been fairly well studied, but many difficulties arise

[1] Laboratoire de Recherche en Informatique, Université de Paris-Sud (Orsay), Bât 490, 91 405 Orsay, (France).

Received May 15, 1986; revised June 1, 1986. Communicated by Bernard Chazelle.

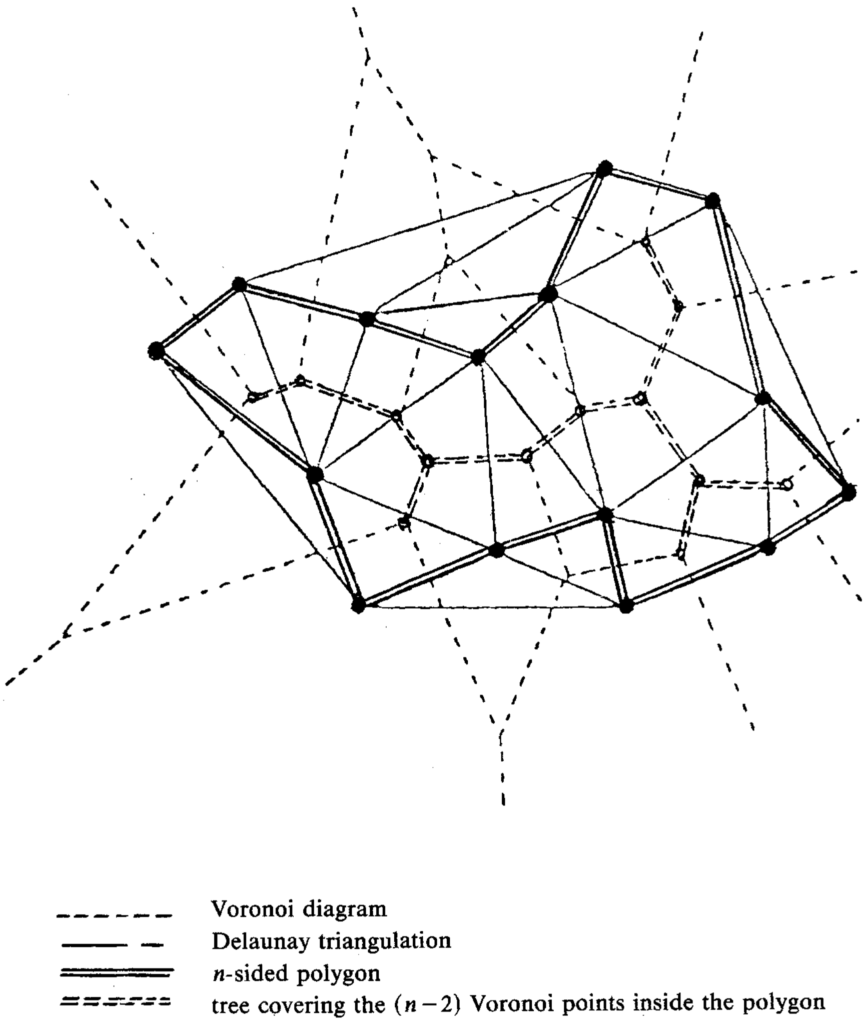

Fig. 1.

when trying to generalize them to higher dimensions: for example, the well-known convex hull planar problem has been solved in worst-case time $O(n \log h)$ in Kirkpatrick and Seidel [4], where n denotes the number of points and h the size of the convex hull, thus yielding an algorithm worst-case optimal in the size of both the input and output. It still remains to be shown, however, whether an $O(n \log h)$ algorithm can also be found for three-space.

When n data points are given, the two farthest points are both on the convex hull. This produces the pair in $O(n \log n)$ time in two dimensions. The closest pair can be found in $O(n \log n)$ in any dimension [1], but we do not know of any algorithm with complexity lower than $O(n^{2-(1/2^{k+1}}(\log n)^{1-(1/2^{k+1})})$ in k-space, $k \geq 4$ ($O((n \log n)^{1.8})$ if $k=3$), [9] for finding the farthest pair.

The same gap between dimension two and greater dimensions appears when we examine the MST problem. There is an $O(n \log n)$ algorithm for the planar case [8]. When the points are taken from a uniform distribution in $[0 \cdots 1]^k$, an average-time linear algorithm exists for $k=2$, but nothing better than $O(n \log \log n)$ is known for $k \geq 3$ [3].

When the average performance of algorithms is examined, the increase in difficulty with dimension is striking. It may be due to a lack of efficient merge procedures: for instance, no method is now known for finding the hull of the union of four-dimensional polyhedra in less than quadratic expected time [2].

Once several algorithms are known for solving a problem, their average-case complexity must be analyzed in order to compare their efficiency. (A standard reference about the study of random geometrical objects is Santalo [7]). Uniform distribution is the one that is usually easiest to study, but normal distribution is another case often encountered. Bentley *et al.* [3] have given an average-case constant time algorithm to find the nearest neighbor of a query point in the case of a uniform distribution, but analyzing the same problem for an unbounded distribution is an open problem.

The average number of points on the convex hull has been computed under various assumptions on the distribution of points, and efficient average-time algorithms are known [2]. Still, it would be interesting to have precise results on the distribution of the number of extreme points (convex-hull points): for example, what is its variance if the points are independently drawn from a uniform distribution in a disk? This may give information on the stability of the algorithms.

When a problem depends on several parameters, it is usually analyzed by making one of the variables grow boundlessly while all the others are kept constant. But almost nothing has been done so far for the study of situations where two or more parameters go to infinity together: what is the average number of points on the convex hull of n points uniformly taken in an n-sided regular polygon? (See Renyi and Sulanke [6] for related questions.) Is there a nontrivial way of finding the two nearest of n points in n-space [1]? All these questions remain open to investigation.

Acknowledgments. Thanks to Jean-Daniel Boissonnat for many suggestions.

References

[1] J. L. Bentley and M. I. Shamos, Divide-and-conquer for linear expected time, *Inform. Process. Lett.*, 7, (1978), 87–91.

[2] J. L. Bentley and M. I. Shamos, Divide-and-conquer in multidemensional space, *Eighth Annual ACM Symposium on Computing*, 1976, pp. 220–230.

[3] J. L. Bentley, B. W. Weide, and A. C. Yao, Optimal expected-time algorithms for closest-point problems, *ACM Trans. Math. Software*, **6** (1980), 563–580.

[4] D. G. Kirkpatrick and R. Seidel, The ultimate planar convex hull algorithm? *SIAM J. Comput.*, **15** (1986), 287–299.

[5] D. T. Lee and F. P. Preparata, Computational geometry—a survey, *IEEE Trans. Comput.*, **33** (1984), 1072–1101.

[6] A. Renyi and R. Sulanke, Über die konvexe Hülle von *n* zufällig gewählten Punkten, *Z. Wahrsch. Verw. Gebiete*, **2** (1963), 75–84.
[7] L. A. Santalo, *Integral Geometry and Geometric Probability*, Addison-Wesley, Reading, MA, 1976.
[8] M. I. Shamos and D. Hoey, Closest point problems, *Proceedings of the 16*th *Annual IEEE Symposium on Foundations of Computing Science,* 1975, pp. 151–161.
[9] A. C. Yao, On constructing minimum spanning trees in *k*-dimensional spaces and related problems, *SIAM J. Comput.*, **11** (1982), 721–736.