Information Processing Letters 20 (1985) 255–257
North-Holland 12 June 1985

A NOTE ON THE EXPECTED TIME REQUIRED TO CONSTRUCT THE OUTER LAYER

Luc DEVROYE *

School of Computer Science, McGill University, 805 Sherbrooke Street West, Montreal, Canada H3A 2K6

Communicated by J. Nievergelt
Received 13 July 1984

The expected time E(T) of the standard divide-and-conquer algorithm for finding the outer layer of a set of points in the plane depends upon the distribution of the points. Under the mild assumption that the points are independent random vectors and have a common bounded density with compact support, it is shown that E(T) = O(n).

Keywords: Outer layer, computational complexity, algorithms, divide-and-conquer, expected time, probability distribution

Introduction

The *outer layer* of points $(X_1, Y_1), \ldots, (X_n, Y_n)$ in R^2 is the subset of points (X_i, Y_i) having the property that one of the four quadrants centered at (X_i, Y_i) contains no (X_j, Y_j), $j \neq i$. Sometimes, these points are also called *maximal vectors*, or *admissible points*. Algorithms for finding the outer layer include:

(i) the *naive algorithm*: for each (X_i, Y_i), carry out $n-1$ comparisons to determine whether it is on the outer layer or not. The time taken by this algorithm is $\theta(n^2)$, i.e., there exist positive constants c_0, c_1, n_0 such that, for $n \geqslant n_0$, the time is bounded from below by $c_0 n^2$ and from above by $c_1 n^2$.

(ii) *one sort and one elimination pass*: sort the points according to their y-coordinates. In an extra pass through the sorted array, eliminate unwanted points by keeping only partial extrema in the x-direction. The time is basically that of the one-dimensional sort. If a comparison-based sort is used, the expected time is bounded from below by $\Omega(n \log n)$ if all permutations of $(X_1, Y_1), \ldots, (X_n, Y_n)$ are equally likely.

(iii) *divide-and-conquer* (Bentley and Shamos [2]): start with n singleton outer layers, marry (merge) all outer layers pairwise, and keep repeating the pairwise marriages until one outer layer is left. This requires about $\log_2 n$ rounds of merging. Note that outer layers of sizes k and m can be married in $O(k + m)$ time if the y-coordinates are kept in order at all times.

The divide-and-conquer algorithm throws away many unwanted points at an early stage. Because of this, there is reasonable hope of obtaining linear expected time. Unfortunately, the expected time depends very heavily on the distribution of the (X_i, Y_i)'s, considered here as independent random vectors. We know of course that the time is deterministically bounded by $O(n \log n)$ (argue as in mergesort, see, e.g., Knuth [5]). If the (X_i, Y_i)'s are independent and uniformly distributed on the diagonal of the unit square $[0, 1]^2$, we have time $\theta(n \log n)$ since no points are thrown away. In contrast, if the (X_i, Y_i)'s are independent, identically distributed, and the X_i's are independent of the Y_i's (such as for the uniform distribution on $[0, 1]^2$, or the standard normal distribution in R^2), the expected time is O(n) (see Bentley and Shamos

* This paper was written while the author was on sabbatical leave at the Department of Computer Science, Australian National University, GPO Box 4, Canberra, ACT 2601, Australia. The support of the Australian National University is gratefully acknowledged. The research was funded by NSERC Grant A3456.

[2] or Devroye [3]). In this note, we would like to point out that the latter result holds for a much larger class of distributions. From here onward, T *is the time taken by the divide-and-conquer algorithm*, and can be considered as the number of coordinatewise comparisons.

Theorem 1. *Let* $(X_1, Y_1), \ldots, (X_n, Y_n)$ *be independent* R^2*-valued random vectors with common bounded density* f *of compact support* (*i.e.*, $f \leqslant C$ *for some finite* C, *and* $f = 0$ *outside a big square*), *then* $E(T) = O(n)$.

Theorem 1 is based upon the following crucial inequality.

Theorem 2. *Let* N_n *be the number of outer layer points for* $(X_1, Y_1), \ldots, (X_n, Y_n)$, *which is a collection of independent random vectors with common density* f *bounded by a constant* C *and support contained in* $[x_{min}, x_{max}] \cdot [y_{min}, y_{max}]$. *Then*

$$E(N_n) \leqslant 8\sqrt{nDC}\,\frac{e}{e-1}\,\frac{1}{1 - 1/\sqrt{nDC}},$$

$$n > 1/(DC),$$

where

$$D = (x_{max} - x_{min})(y_{max} - y_{min}).$$

By a general theorem for the expected time analysis of divide-and-conquer algorithms [3], we have $E(T) = O(n)$ if

$$\sum_{n=1}^{\infty} E(N_n)/n^2 < \infty.$$

Thus, Theorem 1 follows from Theorem 2 without work.

Remark 1. We should note here that if we had married the outer layers by the naive algorithm (i), then

$$E(T) = \Omega\left(\sum_{j=1}^{n} E^2(N_j)/j^2\right),$$

and this is $\Omega(n \log n)$ whenever $E(N_n) = \Omega(\sqrt{n})$: this includes most uniform distributions on compact sets with very few exceptions. One notable exception is the class of uniform distributions on finite unions of rectangles because it is known that for a uniform distribution on one rectangle, $E(N_n) \sim 4 \log n$ (see, e.g., [1,3]).

Remark 2. For uniform distributions on a set $A \subseteq [x_{min}, x_{max}] \cdot [y_{min}, y_{max}]$, we have $DC = (x_{max} - x_{min})(y_{max} - y_{min})/\int_A dx$. This can be recognized as a measure of the degree of concentration of the set A in its enclosing rectangle.

Remark 3. No attempt is made to obtain an inequality in Theorem 2 that is 'best possible'. It should be stressed though that the inequality is *uniform* over the class of distributions given in the theorem. Also, the exact behavior of $E(N_n)$ for uniform distributions on compact sets A with $\int_A dx > 0$ is derived by the author elsewhere: for example, $E(N_n) \sim c\sqrt{n}$, where $c \geqslant 0$ is a constant depending upon A only. Thus, at least the order (in n) in the inequality of Theorem 2 cannot be improved upon.

Remark 4. The number of points on the convex hull of $(X_1, Y_1), \ldots, (X_n, Y_n)$ also satisfies the inequality of Theorem 2. Hence, Theorem 1 also holds for the divide-and-conquer algorithm for convex hulls given in [2] (linear time merging of two convex hulls is possible if points are always stored in clockwise order).

Remark 5. For the distributions of Theorem 1, the convex hull can be obtained from the outer layer in expected time bounded by a constant times

$$E(N_n \log N_n) \leqslant E(N_n) \log n = O(\sqrt{n} \log n),$$

at least if one of the standard 'n log n' convex hull algorithms is used (see, e.g., [7,6,4]). This time is o(n), i.e., it is asymptotically negligible compared with the time needed to find the outer layer. For the suggestion to use two-step algorithms for finding convex hulls, see [1].

Proof of Theorem 2

Let c_n be a sequence of integers to be determined later, and consider the grid of c_n^2 rectangles A_i formed by partitioning each side of $[x_{min}, x_{max}] \cdot [y_{min}, y_{max}]$ into c_n equal intervals. Let Z_i denote the number of points (X_j, Y_j) in rectangle A_i, and let p_i denote $\int_{A_i} f$. We will now obtain a collection of indices, B, by marking certain rectangles. Find the leftmost nonempty column of rectangles, and mark the northernmost occupied rectangle in this column. Let its row number be j (row numbers increase if we go north). Having marked one or more cells in column i, we mark one or more cells in column i + 1 as follows:

(i) mark the rectangle at row number j (the highest row number marked up to that point);

(ii) mark all rectangles between row number j and the northernmost occupied rectangle in column i + 1 provided that its row number is at least j + 1.

By this process, we note that at most $2c_n$ rectangles are marked. Also, any point that is a maximal vector for the north-west quadrant must be in a marked rectangle. We repeat this procedure for the tree other quadrants, so that ultimately at most $8c_n$ cells are marked. Observe that

$$N_n \leqslant \sum_{i \in B} Z_i,$$

and that Z_i is stochastically smaller than W_i, where W_i is a binomial (n, p_i) random variable conditioned on $W_i \geqslant 1$. Thus,

$$E(N_n) \leqslant \sum_{i \in B} E(W_i) = \sum_{i \in B} \frac{np_i}{1-(1-p_i)^n}. \tag{1}$$

Since each term on the right-hand side of (1) is increasing in p_i, and $p_i \leqslant CD/c_n^2$, we have

$$E(N_n) \leqslant 8c_n(nDC/c_n^2)\left(1-(1-DC/c_n^2)^n\right)^{-1}$$
$$\leqslant 8DCn \frac{1}{c_n\left(1-\exp\{-DCn/c_n^2\}\right)}. \tag{2}$$

We take $c_n = [\sqrt{nDC/a}\,]$ where $a > 0$ is a constant, and $[\cdot]$ denotes the floor function. Thus, by substituting $\sqrt{DC/a} - 1$ for c_n,

$$E(N_n) \leqslant 8DCn \frac{1}{c_n(1-\exp\{-a\})}$$
$$\leqslant 8\sqrt{nDCa}\, \frac{1}{1-\exp\{-a\}} \frac{1}{1-\sqrt{a/(DCn)}},$$

valid for $nDC > a$. Theorem 2 follows if we take $a = 1$.

References

[1] J.L. Bentley, H.T. Kung, M. Schkolnick and C.D. Thompson, On the average number of maxima in a set of vectors and applications, J. ACM 25 (1978) 536–543.

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

[3] L. Devroye, Moment inequalities for random variables in computational geometry, Comput. 20 (1983) 111–119.

[4] R.L. Graham, An efficient algorithm for determining the convex hull of a planar set, Inform. Process. Lett. 1 (1972) 132–133.

[5] D.E. Knuth, The Art of Computer Programming Vol. 3: Sorting and Searching (Addison-Wesley, Reading, MA, 1975).

[6] F.P. Preparata and S.J. Hong, Convex hulls of finite sets of points in two and three dimensions, Comm. ACM 20 (1977) 87–93.

[7] M. Shamos, Computational geometry, Ph.D. Dissertation, Yale University, 1978.