Information Processing Letters 33 (1989/90) 319–321
North-Holland

ON A TRIANGLE COUNTING PROBLEM

Samir KHULLER *
Computer Science Department, Cornell University, Ithaca, NY 14853, USA

Joseph S.B. MITCHELL * *
School of Operations Research, Cornell University, Ithaca, NY 14853, USA

Communicated by Fred B. Schneider
Received 23 May 1989
Revised 28 June 1989

We consider the following problem: given a set S of n points in the plane, we would like to compute for each point $p \in S$, how many triangles with corners at points in set S contain p. We give an $O(n^2)$ algorithm to solve the problem.

Keywords: Range query, computational geometry, combinatorics

1. Introduction

To obtain efficient algorithms for many computational geometry problems, one needs to solve efficiently the range counting problem. Given a set S of n points in the plane, preprocess the points so that one can answer efficiently range queries of the following form: given some region R of the plane, how many points of S belong to R?

The recent paper [1] studied the following problem. Given a set S of n points in the plane, answer the following types of "triangle" queries: given three points p, q, $r \in S$, how many points of S are in the interior of the triangle $\triangle pqr$? We developed an $O(n^2)$ time preprocessing algorithm that achieved an O(1) query time.

In this paper we consider an interesting variation of the triangle query problem. We want to preprocess a point set S, to support queries of the following form: given a point $p \in S$, how many triangles with their corners on points in set S contain p? The naive algorithm compares every triangle against every point, requiring $O(n^4)$ time. The objective of this note is to provide an algorithm which does only $O(n^2)$ preprocessing and achieves an O(1) query time. In fact, for each point of S we compute the number of triangles containing it.

The method also solves the following problem: given a set S of n points in the plane, and *any* point q, find the number of triangles (with corners in S) containing q. We solve this problem in $O(n \log n)$ time, or in $O(n)$ time if the points of S are given in sorted order about q.

2. Algorithm

We will assume for simplicity that no three points are collinear. First compute the angularly sorted order of the points about each point. This can be done in $O(n^2)$ time by mapping the points

* Supported by NSF Grant DCR 85-52938 and PYI matching funds from AT&T Bell Labs. and Sun Microsystems.
** Partially supported by NSF Grants IRI-8710858, ECSE-8857642 and a grant from Hughes Research Labs.

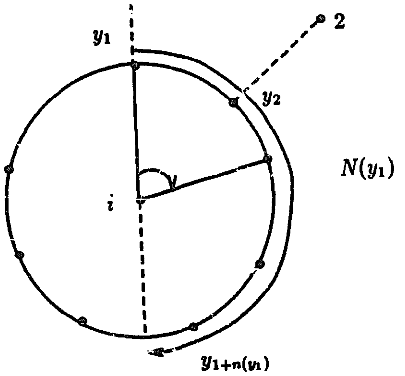

Fig. 1. Calculation of $n(y_k)$.

to their dual lines and constructing the arrangement of these lines [3].

We will denote the n points by integers $i = 1, \ldots, n$. For each point i, we wish to compute the number of triangles containing it. We now show how to perform this computation in $O(n)$ time. Draw a circle with i as the center such that all other points of S lie outside the circle. Now map each point k to the intersection point y_k of the circle with the segment $\overline{ik}$ (see Fig. 1). Note that a triangle based on the points k, l, m contains point i if and only if the triangle based on the points y_k, y_l, y_m contains i. We will refer to the points y_k sorted around i as $y_1, y_2, \ldots, y_{n-1}$ (ordered clockwise). From now on, we will consider only the points y_k.

We define

$$N(y_k) = \{ y_l \mid 0 < \angle y_k i y_l < \pi \},$$

where we measure the angle clockwise. Let $n(y_k) = |N(y_k)|$; thus $n(y_k)$ is the number of points, y_l, lying between y_k and the point diametrically across y_k on the circle. We count the points in clockwise order (see Fig. 1). Note that it is easy to compute the values $n(y_k)$, $1 \leqslant k \leqslant n-1$, in $O(n)$ time, since we compute $n(y_1)$ for point y_1 by simply marching around the sorted point set, and then we can compute $n(y_k)$ for all other points by doing an angular sweep around the points in order.

We now give a relationship for the numbers $n(y_k)$, which will simplify some calculations later.

Lemma 1.

$$\sum_{k=1}^{n-1} n(y_k) = \frac{(n-1)(n-2)}{2}.$$

Proof.

$$\sum_{k=1}^{n-1} n(y_k) = \sum_{k=1}^{n-1} \sum_{y_l \in N(y_k)} 1.$$

Notice that each pair of points y_k, y_l contributes exactly 1 to the summation. Hence, we obtain

$$\sum_{k=1}^{n-1} n(y_k) = \frac{(n-1)(n-2)}{2}. \qquad \square$$

Lemma 2. *The number of triangles containing point i is*

$$\frac{(2n-3)(n-1)(n-2)}{12} - \tfrac{1}{2} \sum_{k=1}^{n-1} n(y_k)^2.$$

Proof. Fix a point y_k as shown in Fig. 2. Now any triangle containing i, with y_k as a corner vertex will have as a second corner vertex point y_l, $k+1 \leqslant l \leqslant k+n(y_k)$. The number of triangles with y_k as a corner vertex and containing i is given by

$$\sum_{l=k+1}^{k+n(y_k)} \left[l + n(y_l) - (k + n(y_k)) \right],$$

since given that a point $y_l \in N(y_k)$ is the second corner of a triangle, we know that the third point must lie in the interval $(y_{k+n(y_k)}, y_{l+n(y_l)}]$.

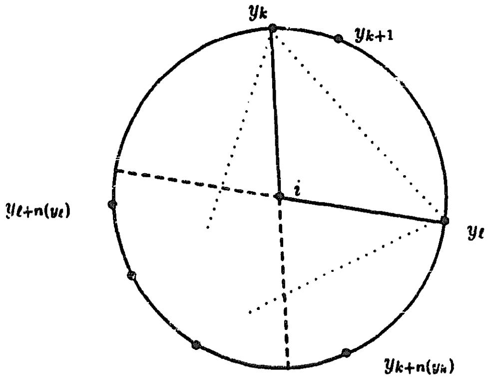

Fig. 2. Number of triangles containing i, with y_k as a corner.

Thus the total number of triangles is

$$T=\sum_{k=1}^{n-1}\sum_{l=k+1}^{k+n(y_k)}\left[l+n(y_l)-(k+n(y_k))\right],$$

where the inner sum on l is circular (i.e., taken mod$(n-1)$). By rearranging the terms we get

$$\begin{aligned}T&=\sum_k\sum_{l=k+1}^{k+n(y_k)}\left[l+n(y_l)\right]\\&\quad-\sum_k n(y_k)(k+n(y_k))\\&=\sum_k\sum_{l=k+1}^{k+n(y_k)}n(y_l)+\sum_k\sum_{l=k+1}^{k+n(y_k)}l\\&\quad-\sum_k\left[n(y_k)k+n(y_k)^2\right].\end{aligned}$$

After some algebra we get

$$\begin{aligned}&=\sum_k\sum_{l=k+1}^{k+n(y_k)}n(y_l)\\&\quad+\sum_k\frac{n(y_k)(n(y_k)+1)}{2}-\sum_k n(y_k)^2.\end{aligned}$$

Now by expanding the first (double) summation, and by combining the last two terms, we get

$$\begin{aligned}T&=\sum_k n(y_k)(n-2-n(y_k))\\&\quad-\sum_k\frac{n(y_k)(n(y_k)-1)}{2}.\end{aligned}$$

Rearranging the terms and applying Lemma 1 we get

$$\begin{aligned}T&=\left(n-\tfrac{3}{2}\right)\sum_k n(y_k)-\tfrac{3}{2}\sum_k n(y_k)^2\\&=\left(n-\tfrac{3}{2}\right)\frac{(n-1)(n-2)}{2}-\tfrac{3}{2}\sum_k n(y_k)^2.\end{aligned}$$

In this calculation each triangle was counted thrice; therefore, the total number of triangles containing the point i is given by

$$\frac{(2n-3)(n-1)(n-2)}{12}-\tfrac{1}{2}\sum_{k=1}^{n-1}n(y_k)^2. \quad \square$$

The formula of Lemma 2 enables us to compute the number of triangles containing i in $O(n)$ time, once we have computed the $n(y_k)$ values for the points around point i. Thus, it takes $O(n^2)$ time to compute the number of triangles containing each point.

Theorem 3. *Given a set S of n points in the plane, we can compute for every point the number of triangles that contain it in total $O(n^2)$ time.*

Interesting possible extensions of our work would be to count the number of convex k-gons that contain a query point q, or to generalize these problems to higher dimensions. We have recently learned of independent work [2] in which a similar algorithm is presented for the problem we address here. They use this result to define a notion of a "median" for a set of points.

Acknowledgment

We would like to thank Estie Arkin, Bob Connelly, Robert Freimer, Mark Novick, Shmuel Onn, Christine Piatko and all other members of the Cornell Computational Geometry Worksessions for very useful discussions. We also thank the referees for several useful suggestions.

References

[1] E.M. Arkin, S. Khuller and J.S.B. Mitchell, Optimal enclosure problems. In: *Proc. 1st Canadian Conf. on Computational Geometry*, Montreal, Canada (August 1989).
[2] J. Gill, W. Steiger and A. Wigderson, Some geometric medians. In: *Proc. 1st Canadian Conf. on Computational Geometry*, Montreal, Canada (August 1989).
[3] D.T. Lee and Y.T. Ching, The power of geometric duality revisited, *Inform. Process. Lett.* **21** (1985) 117–122.