A convex hull algorithm for discs, and applications

David Rappaport*
*Department of Computing and Information Science, Queen's University, Kingston, Ont., Canada K7L 3N6

Communicated by Hiroshi Imai
Submitted 15 October 1990
Accepted 30 August 1991

Abstract


We show that the convex hull of a set of discs can be determined in $\Theta(n \log n)$ time. The algorithm is straightforward and simple to implement. We then show that the convex hull can be used to efficiently solve various problems for a set of discs. This includes $O(n \log n)$ algorithms for computing the diameter and the minimum spanning circle, computing the furthest site Voronoi diagram, computing the stabbing region, and establishing the region of intersection of the discs.

Keywords. Discs; convex hull; Voronoi diagram; transversals; minimum spanning circle.

1. Introduction

In many applications using geometric data, a set of sites on a plane is modelled as a set of points. Much work has been done in solving problems efficiently for planar point sites. In this paper we consider a collection of geometric problems by modelling sites as a set of discs of various sizes.

In Section 2 we begin by presenting an algorithm to compute the convex hull of a set of discs. We then show how the convex hull algorithm can be used to solve a collection of other problems. The key to many of the results is to consider the set of discs as a cross section of a set of axis parallel cones.

An algorithm to compute the so called stabbing region for a set of discs is presented in Section 3. A stabbing line, or a line transversal, of a set of objects is a straight line that intersects every object in the set. In [7] an $O(n \log n)$
algorithm is introduced to determine whether a stabbing line exists for a set of line segments. The *stabbing region*, a concise description of all possible stabbing lines is also obtained in $O(n \log n)$ time. A simple object is an object that can be described in $O(1)$ space and pairwise tangents and pairwise intersections can be computed in $O(1)$ time. Examples of simple objects are $k$-gons ($k$ fixed), ellipses, etc. Atallah and Bajaj [2] obtain stabbing regions of a set of simple object in $O(\lambda_t(n) \log n)$ time. The function $\lambda_t(n)$ grows very slowly, i.e., $\lambda_t(n) = O(n)$, $t < 3$, $\lambda_t(n) = O(n \log^* n)$, $t \geq 3$. The parameter $t$ represents the maximum number of common support lines admissible by a pair of simple objects. For example for discs $t = 2$ and for triangles $t = 6$. The function $\lambda_t(n)$ can also be considered as the length of a Davenport–Schinzel sequence of parameter $t$. We show how the convex hull algorithm of Section 2 can be used to implement the ideas in [2] for discs. We also introduce a novel approach to the representation of the stabbing region for a set of objects.

In Section 4 we show how to compute the diameter of a set of discs by using the convex hull algorithm. In Section 5, our algorithm to compute the convex hull of discs is used to compute the intersection of a set of three dimensional right circular cones. The results of Section 5 are applied in Section 6 to obtain the intersection of a set of discs.

In Section 7, we present an algorithm to compute the furthest site Voronoi diagram for a set of discs. Given a set of point sites, $S$, in $\mathbb{E}^2$, the Voronoi diagram is a partition of $\mathbb{E}^2$ into disjoint Voronoi regions such that for each site $s \in S$, the Voronoi region $V(S)$ is the locus of points that are closest to the site $s$ than to any other site [12–13]. A generalization of the Voronoi diagram, the furthest site Voronoi diagram, is a partition of $\mathbb{E}_2$ such that for each site $s \in S$ the furthest site Voronoi region $FV(s)$ is the locus of points further from the site $s$ than from any other site. As was shown in [12], the region $FV(s)$ is non-empty if and only if $s$ is on the convex hull of $S$. Shamos and Hoey [13] introduced an $O(n \log n)$ divide and conquer algorithm for computing the Voronoi diagram for a set of point sites. Briefly, Shamos and Hoey's algorithm splits a set of points by a vertical separating line into two roughly equal parts, and recursively computes their Voronoi diagrams. A linear time algorithm is then used to merge the two diagrams. The merge step uses the geometry of Voronoi regions and requires careful transversal of fairly complex data structures. In subsequent years, similar methods have been used to find Voronoi diagrams for different types of sites. Kirkpatrick [9] examines the geometric structure of the Voronoi diagram of a set of line segments to tailor a linear time merge step for its efficient computation. Sharir [14] has analyzed the geometry of the Voronoi diagram for sites that are discs, and has developed an $O(n \log n)$ merge step, leading to an $O(n \log^2 n)$ algorithm. Yap [19] examines the case where sites are straight or curved line segments and gives an $O(n \log n)$ algorithm. Fortune [8] has presented a novel way of constructing Voronoi diagrams. Fortune's result relies on the property that vertices in the Voronoi diagram are intersection points of a suitably constructed
set of axis parallel cones. He then uses a sweeping plane to detect these intersections. The same method is modified slightly to compute the Voronoi diagram for a set of discs. A more complicated version is also presented to compute the Voronoi diagram for a set of line segments. Keeping in spirit with the method of [8], we use the algorithm for computing the intersection of a set of right circular cones, presented in Section 5, to compute the furthest site Voronoi diagram for a set of discs in $O(n \log n)$ time.

In the final section of the paper we discuss the expected time complexity of the algorithms that have been presented. We show that under certain assumptions the algorithms have an $O(n)$ expected time complexity.

2. A convex hull algorithm for a set of discs

Let $S$ be a set of closed planar discs. To simplify the presentation we assume that the discs are in general position. The convex hull of $S$, $\text{Hull}(S)$, is the smallest convex region containing $S$. Let $\partial(R)$ denote the boundary of a simple closed set $R$. Then $\partial(\text{Hull}(S))$ denotes the boundary of the convex hull of $S$, and consists of straight line segments or edges and arcs of circles. To be more precise, define $L$ as a directed line, and let $H(L)$ denote the closed right half-plane of $L$. We say that a half plane is a supporting half plane of the set $S$ if it contains $S$ in its interior. We say a line $L$ is a support line of a set of discs $S$ if $H(L)$ is a supporting half plane of $S$ and no subset of $H(L)$ is a supporting half plane of $S$. Let $A$ and $B$ be two sets of discs. The common support line of $A$ and $B$, if such a line exists, directed from $A$ to $B$, is defined as $L(A, B)$. Let $t(A, B)$ denote a closed subset of $L(A, B)$ with one endpoint on $\partial(A)$ and one endpoint on $\partial(B)$. Let $a$ and $b$ denote two different discs in $S$. We define an edge of $\text{Hull}(S)$ as $t(a, b)$ such that $L(a, b)$ is a supporting line of $\text{Hull}(S)$. We represent $\partial(\text{Hull}(S))$ by a list of discs $s \in S$, that is, $\text{CH}(S) = (s_0, s_1, \ldots, s_h)$, such that $t(s_i, s_{i+1})$ is an edge of $\text{Hull}(S)$ for $i = 0, \ldots, h - 1$. Note that a disc may appear more than once on $\text{CH}(S)$, so the list $\text{CH}(S)$ may contain elements $s_i$ and $s_j$ where $i \neq j$ but $s_i = s_j$. See Fig. 1. $\text{CH}(S)$ is viewed as a cyclic sequence. We overcome the complication of using a circular list by setting $s_0 = s_h$.

**Lemma 2.1.** $|\text{CH}(S)| \leq 2n - 1$.

**Proof.** Let $u, v$ denote any two distinct discs in $S$. We show that $u, \ldots, v, \ldots, u, \ldots, v$ is a forbidden subsequence of the list $\text{CH}(S)$. Let $S_0, S_1, S_2, S_3, S_4$ denote subsequences of $\text{CH}(S)$ so we can write $\text{CH}(S) = S_0, u, S_1, v, S_2, u, S_3, v, S_4$. First consider the case where $S_1$ and $S_3$ are of length 0. (Or $S_0, S_2$ and $S_4$ are of length 0). This implies that $t(u, v)$ appears twice as an edge of $\text{CH}(S)$, which is absurd. (Or $t(u, v)$ appears twice as an edge of $\text{CH}(S)$). Suppose on the other hand the $S_1$ is not of length 0. This implies that $L(u, v)$ is a line