TR-85-001

Geometric Complexity and Related Problems

D. T. Lee

Institute of Information Science
Academia Sinica
Nankang, Taipei, Taiwan
Republic of China 115

Abstract

In this report we briefly sketch the-state-of-the-art of computational geometry and some background material. A list of open problems is given under different categories. The interested reader may find relevant information in the references at the end of the report. If a solution is found to any of the open problems, please contact the author at the address below.

Professor D. T. Lee
Department of Electrical Engineering
and Computer Science
Northwestern University
Evanston, IL 60201
U. S. A.

Geometric Complexity and Related Problems

D. T. Lee

Introducation

We in the report shall deal with problems that arise mostly in computational geometry and other related areas. Since its introduction by Shamos [Sh75] in 1975, computational geometry has become a popular field of research and tremendous amount of progress has been made and hundreds of research papers produced by researchers in the field. There are two main reasons for such incredible activities in this field. Firstly, most geometric problems arise in various other disciplines, including computer-aided design, computer graphics, operations research, pattern recognition, robotics and statistics, and efficient algorithms for these problems are desirable. Secondly, issues of how difficult the problems are to compute under certain computation models are also of interest to theoretical computer scientists. A recent survey [LeP84a] of the state of the art of the field includes more than three hundred research articles as of August 1984. Nevertheless, the list of references is by no means complete. In this survey the field is conveniently classified into five major problem areas -- convexity, intersection, searching, proximity and optimization, and seven problem-solving paradigms -- incremental construction, plane-sweep, locus, divide-and-conquer, geometric transformation, prune-and-search and dynamization are discussed. Several open problems are included and directions for future research suggested.

2. Lower Bound

In complexity theory one usually measures the difficulty of a computational problem in terms of the number of "key" operations required by any algorithm that "solves" the problem in a certain computation model. For example, sorting of n numbers in the decision tree model [AHU74] requires Ω(nlog n) key operations, namely, comparisons between numbers in the input. When the computation model changes, the "complexity" of the problem differs. For instance, the maximum gap problem, i.e., given n numbers find the maximum difference between two successive numbers when these numbers are ordered, requires Ω(nlog n) comparisons [MT82, PrSh84] in the linear decision tree model. However, when floor functions are allowed in the model, the problem can be solved in O(n) time [G75]. On the other hand, when an algorithm is given for a problem, it provides an <u>upper bound</u> on the difficulty of the problem. A fundamental question often raised by computer theoreticians when confronted with a computational problem is whether or not an asymptotically optimal algorithm can be obtained, i.e., an algorithm whose complexity matches the lower bound of the given problem. When the answer is affirmative, the complexity is referred to as the <u>intrinsic</u> difficulty of the problem. For example, the problem of finding the intersection of n half-planes in the plane has complexity Θ(nlog n) in the algebraic computation tree (ACT, for short) model of Ben-Or [B83]. When there is a gap between upper and lower bounds of a given problem, there is potential for improvement. In the survey [LeP84] we have included a list of problems, most of which have Ω(nlog n) lower bounds in the ACT model. Some of the problems have Θ(nlog n) as their complexity, since O(nlog n) algorithms for them can be obtained. The proof techniques are mostly by problem transformations. That is, if problem A is known to require f(n) time and is g(n)-time transformable to problem B, then problem B requires at least f(n) - cg(n) time for some positive constant c. Since we have a kernel of problems that require Ω(nlog n) time in the ACT model, by problem transformation we are able to construct a class of problems that are as hard as the problems in the kernel. This has a strong analogy to the NP-hardness result [GJ79] in the sense that if a problem in the NP-complete (NP-hard) class is polynomial-time transformable to another problem, then the latter problem is as hard as the former and is therefore in the class of NP-hard problems. We hope to establish a similar catalog of problems that require $\Omega(n\log n)$ time in the ACT model. For convenience we shall refer to the problems requiring $\Omega(f(n))$ time in the ACT model or others as f(n)-hard and to those f(n)-hard problems solvable in O(f(n)) time as f(n)-complete. Specifically, the complexity of the following problems is of interest.

<u>Problem</u> 2.1 Smallest enclosing rectangle [FrS75]: Given a set of n points in the plane, determine the smallest (area) rectangle enclosing the set.

This problem is known to have an O(nlog n) solution [Tou83]. We expect this problem to be nlogn-complete.*

Problem 2.2 Weighted 1-center: Given a set of n points, p_i with a positive weight w_i, i= 1,2,...,n, find the center c such that the maximum weighted distance from c to the set is minimized, i.e., minimize $\max_i w_i d(c,p_i)$, where d(a,b) is the Euclidean distance between points a and b.

This problem has an O(nlog n) solution [Co84]; whereas its unweighted counterpart (i.e., $w_i = 1$ for all i) can be solved in O(n) time [Meg83]. Apparently the <u>weights</u> play a crucial role in determining the complexity and must be involved in the proof.

Problem 2.3 Smallest enclosing circle for line arrangement: Given n lines in the plane, find the smallest circle that encloses all the intersection points.

There are in general $O(n^2)$ intersection points determined by these n

*Y. T. Ching has proved that this problem requires $\Omega(n\log n)$ time in the linear desicion tree model.

lines, but only O(n) extreme points are relevant, namely, the two farthest intersection points on each line. With this observation we are able to show that the problem of finding the diameter of these n lines, i.e., the two farthest intersection points, is nlogn-complete [ChL83]. The O(nlog n) algorithm for finding the diameter [ChL83] can be used to solve this problem within the same time bound. Recall that the diameter problem and the smallest enclosing circle problem for n points are, respectively nlogn-complete (See, e.g. [LeP84a]) and n-complete [Meg83]. We believe, however, that this problem is also nlogn-complete.

Problem 2.4 Triangulation of a simple polygon: Given a simple polygon with n vertices in the plane, triangulate its interior into n-2 triangles without adding new points.

This problem has been studied by many authors and the worst-case complexity of the algorithms obtained is O(nlogn) [Ch82,ChI83,GJPT78,HM83]. For certain special classes of polygons o(nlog n) algorithms are known [ScV80,TouA82,ChI83]. The problem of triangulating an arbitrary set of n points has been shown to be nlogn-complete [Sh75]. Of interest is the following outstanding question: whether or not the fact that the points form a simple polygon helps reduce the complexity. As demonstrated in [ChI83], an O(nlog s) algorithm can be obtained, where s is the sinuosity of the polygon, namely the number of times the boundary alternates between complete spirals of opposite direction. The parameter s in a way gives the shape complexity of the polygon. We conjecture that this problem is nlogn-complete, i.e., the fact that the points form a simple polygon does not help. Unlike the convex hull problem for simple polygons, where certain global information of the input enables one to disregard points and obtain an O(n) algorithm

[McCA79, Le80a, GrY81] as opposed to being nlogn-complete [Sh75, vE80, Ya81, KiS82, KiS83] for general point sets, the local information of each vertex must be examined. Exactly what local information is relevant to the problem will be investigated. Along the same line the following two problems are conjectured to be nlogn-complete.

Problem 2.5 Closest pair of a simple polygon: Given a simple polygon with n vertices in the plane, find the two closest vertices.

Problem 2.6 Simplicity test: Given a sequence of n points where successive two points define a line segment, determine if the sequence of line segments defines a simple curve, i.e., if any two nonconsecutive segments intersect.

For these two problems if the input is an arbitrary set, both problems are known to be nlogn-complete [ShH75, ShH76]. Worth investigating is the crux of why an "ordered" input may not reveal enough useful information to render more efficient algorithms. There are instances, however, in which additional properties of the input have been shown asymptotically to be algorithmically "useless". For example, the problem of sorting n triangulated points in the plane, say in x-coordinates, remains nlogn-complete [Se84]. In other words, given a triangulation of a set of n points in the plane, to sort the set of points is asymptotically as hard as sorting the same set of points without triangulation. We remark here that should not necessarily restrict ourselves to proving the nlogn lower bound. Nontrivial lower bounds other than nlogn are of even greater interest. One reason for the study of the nlogn lower bound is that the bound is relatively easier to derive than other bounds and is of interest in its own right.

3. Constrained Proximity

Proximity problems that we are concerned with in this section are those with "constraints." Consider, for example, the closest pair problem in the plane. Given is a set of n points, and the problem is to determine the closest pair. In other words, we want to find the shortest line segment connecting two of the given points, where the length of a segment is measured in terms of the Euclidean distance between the two endpoints of the segment. A constrained version would be to consider a set of n points and a set of obstacles represented as polygons, or circles, and find the two points such that the path connecting these two points without crossing any of the obstacles is the shortest. The closest pair problem without constraints is known to be nlogn-complete [ShH75]. We therefore cannot hope to have an o(nlog n) solution for the constrained closest pair problem.

A special case of the constrained closest pair problem is the problem of finding a shortest path connecting two given points in the presence of obtacles. For the constrained shortest path problem we have the following results. If the obstacles are line segments which form a simple polygon with n edges, the shortest path connecting two points in the polygon can be found in O(nlog n) time [LeP84]. Note that if the polygon has been triangulated, O(n) time suffices. Thus, a lower bound for this problem is also a lower bound for the triangulation problem (Problem 2.4) If the line segments are parallel or more generally, if there exists a line such that the projections of these line segments onto the line are pairwise disjoint, then the problem has been shown to be nlogn-complete [LeP84]. If the line segments are in general position, the best known solution runs in $O(n^2\log n)$ time [Le78]. An outstanding problem has been to see if one can obtain an $o(n^2\log n)$ algorithm for the general problem. For obstacles other than line segments see [LaLi81, LeM83, ShS84]. A related problem studied by

O'Dunlaing *et al.* [OSY83], Lozano-Perez and Wesley [LoW79], and Schwartz and Sharir [ScS82] arises in motion planning, in which the objects to be moved have positive measure (rather than points) and only a feasible path is sought. Specifically, given a set of polygonal obstacles in the plane and two placements of an object, determine if one can move the object from one placement to the other. If the object is a disk, O'Dunlaing *et al.* have obtained an $O(n\log n)$ algorithm using the Voronoi diagram approach [Ki79]; if the object is a line segment then the best known solution runs in $O(n^2\log n)$ time [OSY83].

For the constrained closest pair problem one can resort to a general approach of constructing the so-called *visibility* or *viewability* graph and solving the graph-theoretic "all pair" shortest path problem. (The $O(n^2\log n)$ solution for the constrained shortest path problem is based on such a technique.) The question is of course whether or not one can do better. Since the constrained closest pair problem is much harder than the shortest path problem, unless efficient algorithms can be obtained for the latter we cannot hope for any good solutions. Thus, we shall restrict ourselves to the following problems, which are special cases of the general problem.

Problem 3.1 Shortest path problem in the presence of obstacles: Given a set of k obstacles and two distinguished points s and t, find a shortest path connecting s and t without crossing any of the obstacles.

As indicated earlier, this problem in general can be solved in $O(n^2\log n)$ time, where n is the size of the input. For example, if the obstacles are pairwise disjoint simple polygons, n is the total number of vertices. A preliminary study of a special case in which the obstacles are (nondegenerate) convex polygons indicates that the time complexity of $O(n^2 \log n)$ can be improved. When the obstacles are rectilinear rectangles, i.e., rectangles whose sides are parallel to the coordinate axes, and the underlying distance metric is the L_1-metric or Manhattan distance, we have obtained a result [DLW84] which shows that the shortest path problem is nlogn-complete. Specifically it is shown [DLW84] that there must exist a path from s to t that is monotone either in the x-direction or in the y-direction. By using the plane-sweep technique as in [LeP84], a shortest path can be found in $O(n\log n)$ time. We wish to generalize the result to the case where the obstacles are rectilinear polygons or convex rectilinear polygons.

Problem 3.2 Closest pair problem in a simple polygon: Given a simple polygon with k vertices and a set S of n points in the interior of the polygon, find the two "closest" points.

For this problem we expect that the Delaunay triangulation result of a simple polygon [LeL84] and the "dual" technique used in [LeP84] for finding a shortest path in a simple polygon may be useful. Meanwhile the notion of the Voronoi diagram with obstacles (defined below) is important for constrained proximity problems.

Constrained Voronoi diagrams

Given a set S of n points and a set K of obstacles in the plane, the **constrained nearest neighbor Voronoi diagram** VOD(S,K) is defined as follows. VOD(S,K) is a collection of regions, each of which is associated with a point of S and is the locus of points whose distance from the point is the shortest. That is, $VOD(S,K) = \{ R(p_i) \mid p_i \in S,\ i=1,2,\ldots,n\}$, where $R(p_i) = \{r \mid d(r,p_i) = \min_j d(r,p_j)\}$ and $d(r,p)$ is the length of a shortest path from r to p without crossing any obstacles in K. Figure 1 illustrates a constrained Voronoi diagram for 6 points with 3 line segments as obstacles. The dotted lines in the figure are edges of the Voronoi diagram without obstacles. The cross-lined area is the region associated with p_3 and any point r in the region has a path to p_3 that is the shortest among all paths from r to the points in S. Notice the Voronoi polygon for p_3 of the unconstrained Voronoi diagram has been altered with the presence of segments $\overline{A,B}$ and $\overline{E,F}$. In the diagram the curved segments are portions of some hyperbolae. As in the ordinary Voronoi diagram [Sh75, Le80, Le82], the constrained Voronoi diagram contains a lot of proximity information in the presence of obstacles. The diagram is a natural generalization of the ordinary diagram and is expected to play an important role in solving constrained proximity problems. The problem of how such a diagram can be constructed is thus of great interest.

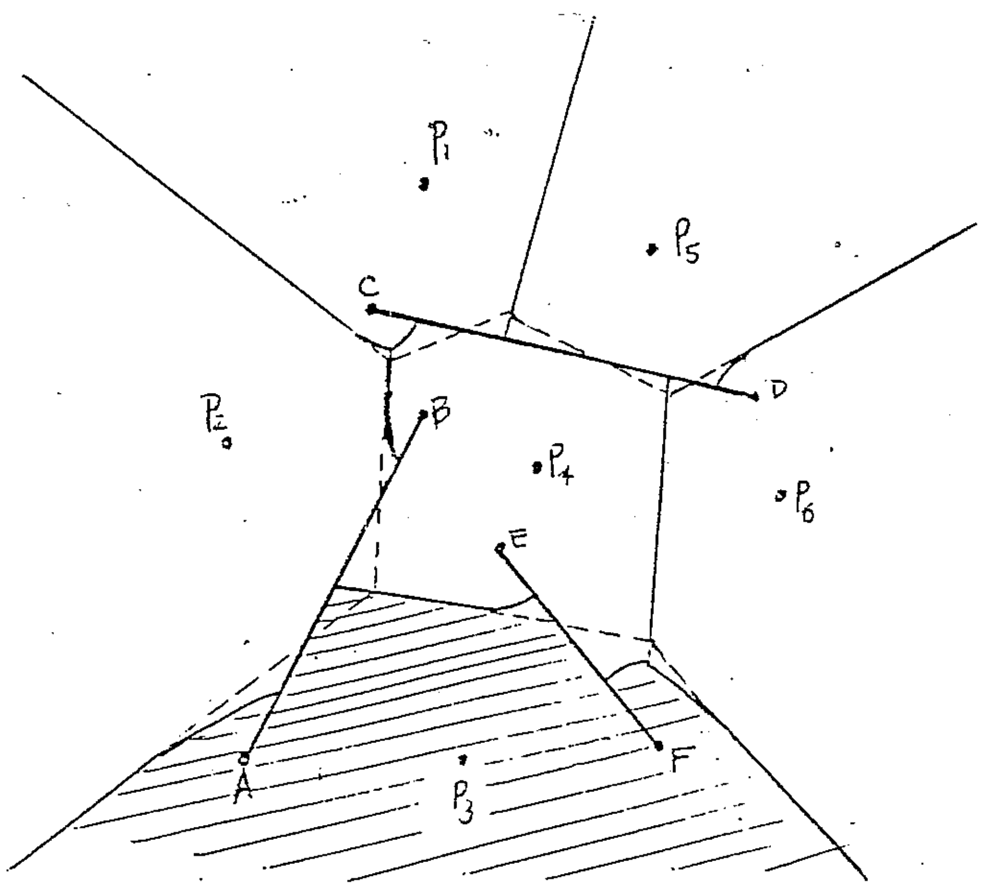


Figure 1. Constrained nearest neighbor Voronoi diagram.

Problem 3.3 Construction of constrained Voronoi diagram: Given a set S of n points and a set K of obstacles in the plane, compute the Voronoi diagram defined earlier.

We expect that this notion of the Voronoi diagram will open up a new area of research. Abundant are problems about variations of the diagram and associated proximity problems. To just give a few examples, the underlying metric for the diagram can vary, and the set S can have geometric objects other than points. The following problem has obvious applications and is worth looking into as well.

Problem 3.4 Constrained minimum spanning tree: Given a set S of points and a set K of obstacles, find a shortest tree interconnecting the points of S without crossing any of the obstacles.

A situation where this problem is of importance is that of connecting a set of e.g., computer work stations, to form a network such that the interconnecting cables should not cross existing physical barriers.

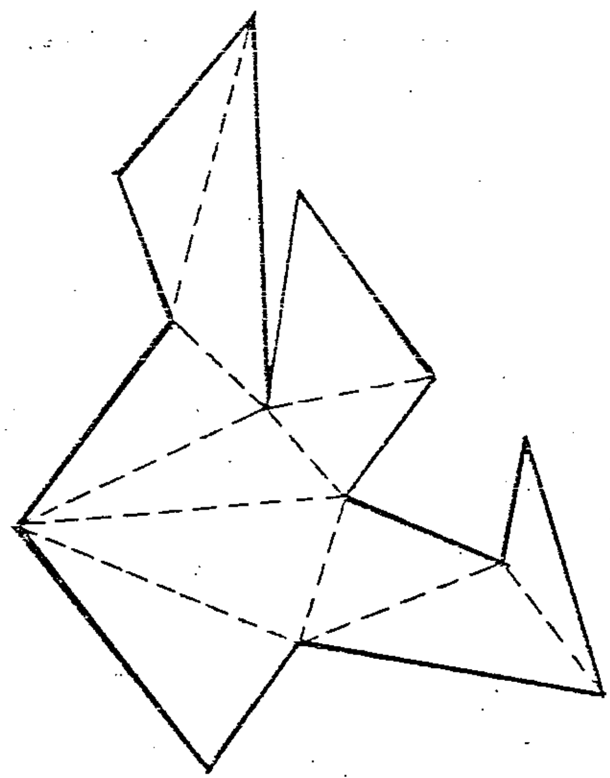
Problem 3.5 Constrained Delaunay triangulation: Given a set S of objects in the plane, e.g., points, line sgments, or polygons, represented as a graph G(V,E), find the Delaunay triangulation of S.

First of all, let us define the Delaunay triangulation of a set of objects in the plane. A triangulation of a set of objects represented as a planar straight-line graph G(V,E) is a triangulated graph G'(V,E'), in which each face of the graph, except the exterior one, is a triangle and $E \subseteq E'$. A Delaunay triangulation of G(V,E) is a triangulation such that the circumcircle of each triangle does not contain any vertex visible from the vertices of the triangle. A vertex v in V is said to be visible from another vertex w if the line segment $\overline{v,w}$ does not cross any edge in E in the interior. Triangulations have applications in finite element method [Ca74], interpolation [McL76] and terrain fitting [LaSc80]. The local extrema of a terrain surface, i.e., peaks of mountains and pits of valley, are represented as points and mountain ridges as edges of the graph G. The triangulation that we seek is one that retains all the ridge line segments and yet preserves the features of the Delaunay triangulation. Thus, the generalized Delaunay triangulation will define a piecewise planar approximation to the terrain surface.

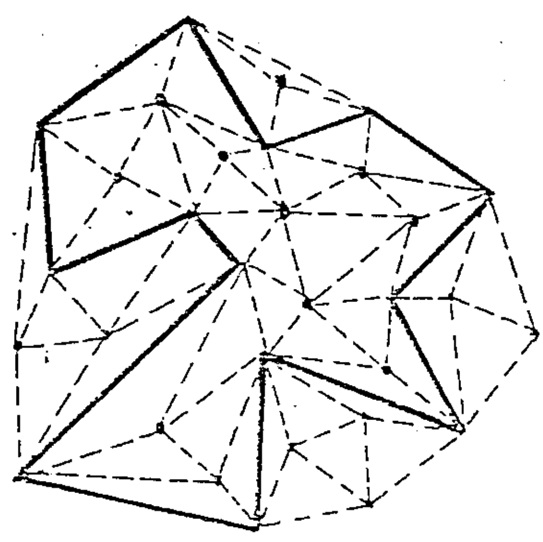
The Delaunay triangulation of a simple polygon is first studied by Lee [Le78] where an $O(n^2)$ time algorithm is given. Making use of the polygon cutting theorem of Chazelle [Ch82], we are able to improve the time bound to $O(n\log n)$ [LeL84b]. In the same paper [LeL84b] we also show that the generalized Delaunay triangulation of a set of points and a simple polygon can be obtained in $O(n\log n)$ time. Figure 2 illustrates the generalized Delaunay triangulation of a simple polygon with and without points. Problem 3.5 can be solved by the visibility graph approach mentioned earlier. It is easy to show that the generalized Delaunay triangulation is a subgraph of the visibility graph, which can be obtained in $O(n^2\log n)$ time, where $n = |V|$. (Recall that $|E|$ is $O(|V|)$, since we have a planar graph.) Once the visibility graph is available, the generalized Delaunay triangulation can be found in $O(n^2)$ time. The question is whether one can compute the generalize Delaunay triangulation in $o(n^2\log n)$ time. We hope to investigate this problem further and see if it can be solved in optimal time, i.e., $O(n\log n)$, as in the case when the set E of edges of the graph is empty.

Problem 3.6 Visibility polygon from an edge: Given a simple polygon P with n vertices on its boundary and an arbitrarily specified edge e, find the region of P that is visible from the edge. That is, find the visibility polygon $VISPOL(P,e) = \{ r \mid \overline{r,x} \in P \text{ for some } x \in e\}$.

A point r of P is (weakly) visible from an edge e if there exists a point x on e such that the line segment $\overline{r,x}$ lies completely in P. This problem is first studied by Avis and Toussaint [AvT81], in which they are concerned with the visibility issue of a simple polygon, i.e., determining whether or not the polygon is visible from a given edge. The region visible from the edge, called the visibility polygon, is the region that a guard can



(a)



(b)

Figure 2. Delaunay triangulations of simple polygons without and with points respectively.

"see" when it patrols back and forth along the edge. This is a natural generalization of the visibility problem from a fixed point (see, e.g. [ElA81, Le83]). In [AvT81] an O(n) time algorithm is given for the visibility problem and the problem of finding the visibility polygon in O(n) time is posed as an open question. Since then, O(nlog n) algorithms for computing the visibility polygon from an edge of a simple polygon with n vertices have been independently obtained [ChG84, El84, LeL84]. We would like to see if indeed O(n) time is sufficient or if the problem is actually nlogn-complete.

A related problem, posed by Klee [H76] and known as **art gallery problem** or **watchman problem**, is to find a minimum number of watchmen on the boundary of a simple polygon (an art gallery) such that the entire region is visible. This problem was partially solved by Chvatal [Chv75] that $\lfloor n/3 \rfloor$ watchmen are always sufficient and this number is the best possible in some cases (Fig. 3). A simpler proof was later found by Fisk [Fi78] and it lends itself to an O(nlogn) algorithm developed by Avis and Toussaint [AvT81a] for locating these $\lfloor n/3 \rfloor$ stationary watchmen. If the polygon is rectilinear, i.e., the edges of the polygon are either horizontal or vertical, Kahn et al. [KKK80] and O'Rourke [O82a] have shown that $\lfloor n/4 \rfloor$ watchmen are sufficient and sometimes necessary (Fig. 4). Sack [Sa82] and Edelsbrunner et al. [ECW84], based on the results of [KKK80] and [O82a], respectively, have devised an O(nlogn) algorithm for locating these $\lfloor n/4 \rfloor$ watchmen. The watchman problem can be treated as one of the polygon decomposition problems, in which a polygon is to be decomposed according to vertex visibility. Since the visibility polygon from each watchman is a star-shaped polygon, the watchman problem is equivalent to finding a minimum number of star-shaped polygons such that the union of these star-shaped polygons covers the entire polygon. For polygon decompositions into various types of "primitives" and their

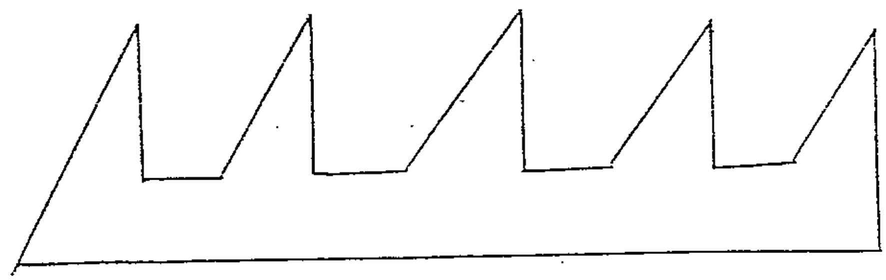


Figure 3. A polygon that requires $\lfloor n/3 \rfloor$ watchmen.

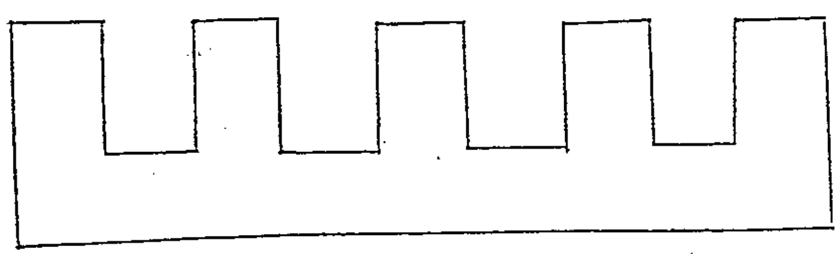


Figure 4. A polygon that requires $\lfloor n/4 \rfloor$ watchmen.

complexities, see, for example [CD79,FP75,O82]. In the context of the watchman problem O'Rourke [O83] showed that $\lfloor n/4 \rfloor$ such mobile watchmen, where the watchmen can move along fixed line segments (the edges of the polygon, for example), are always sufficient and sometimes necessary. Recently we have shown that the (stationary or mobile) watchman problem for simple polygons is NP-hard [LeL84a]. An interesting problem is to see if it still remains NP-hard for rectilinear polygons.

4. Other Related Problems

In this section we shall address ourselves to the problems that call for relationship between two sets of objects. For example, the closest pair problem between two sets of points is one in which we are given two sets of points and want to find the two closest points, one from each set. This type of problems arises as a subproblem when solving problems by, for example, divide-and-conquer. Specifically consider the problem of finding all the containment pair of a set S of n isothetic rectangles [VW80,LeP82]. After we have divided the set S into two equal subsets S_1 and S_2 and recursively solve these two subproblems, we have a subproblem of finding the containment pair (R_i, R_j) of rectangles such that $R_i \in S_1$ and $R_j \in S_2$.

Not many specific results for this class of problems are known. The problems of finding closest pair and farthest pair between two sets of points have been studied. Schwartz [Sc81] gives an $O(\log^2 n)$ time algorithm for finding the closest pair of points (not necessarily at the vertices) between two convex polygons with n vertices each; the result is later improved to $O(\log n)$ by Chin and Wang [CW83] and Edelsbrunner [E82] independently. If the closest pair of vertices between two convex polygons is sought, $O(n)$ is both necessary and sufficient [CW84, McKT83, Tou83a].

The problem of finding the closest pair of points between two arbitrary sets, however, is nlogn-complete [LeP84a]. The farthest pair of points between two sets of points can be solved in O(nlog n) time; and in O(n) time if the points form each a convex polygon [BhT83]. The farthest pair problem can also be shown to be nlogn-hard by transforming from the set disjointness problem [Re72]. Reporting of pairwise intersections between two sets of line segments is another problem in this class that has been studied. In [MaS84], Mairson and Stolfi give an optimal O(nlogn +k) time algorithm, where k is the number of such intersecting pairs reported, when the two sets of line segments are each pairwise disjoint.

In general, we are given two sets A and B of objects and want to find a relation R from A to B such that a certain property P is satisfied. That is, $R = \{\langle a,b\rangle \mid a \in A,\ b \in B,\ P(a,b)\}$, where P is a predicate. For example, the problem studied by Mairson and Stolfi can be formulated as one in which A and B are sets of line segments and no two segments in the same set intersect, and P(a,b) denotes that segments $a \in A$ and $b \in B$ intersect. The detection of whether or not any segment of A intersects a segment of B is then equivalent to testing if the relation R defined is empty. Within this class of problems we would like to study, among others, the following.

Problem 4.1 Segment intersection reporting: Given two sets A and B of line segments, find $R = \{\langle a,b\rangle \mid a \in A \text{ and } b \in B \text{ intersect}\}$.

This problem arises as a subproblem in the hidden-surface elimination problem studied by Chu and Lee [CL83]. In [CL83] the edges of polygons representing the faces of a polyhedron are projected to the view plane and the intersection of the set of boundary edges and the set of boundary and nonboundary edges need to be computed. (A boundary edge is one which borders a front and a back face with respect to a given viewpoint.) A fast solution of this problem may also be used to solve the segment intersection reporting problem defined in [LeP84a], i.e., given a set of line segments in the plane, find all intersecting pairs of segments.

Problem 4.2 Half-planar range query: Given a set A of points and a set B of half-planes, find $R = \{\langle a,b\rangle \mid a \in A \text{ lies in } b \in B\}$.

When $|B| = 1$ and the half-plane is given as a query, the reporting problem can be solved in optimal time ($O(\log n + k)$, where k is the output size) with $O(n\log n)$ preprocessing and $O(n)$ space [CGL83]. But the technique used cannot solve the counting problem, i.e., determining the size of R, optimally. We hope to investigate the possibility of having such an optimal solution. The counting counterpart of the general problem in which A is the set of vertices of a simple polygon and B is the set of half-planes determined by the edges of the polygon is an instance of the so-called discrete signature of a polygonal figure [O84]. For the signature problem, O'Rourke gives an $O(n^{3/2}\log n)$ algorithm and poses as open if $O(n\log n)$ is achievable [O84]. When B is the set of half-planes determined by $O(n^2)$ lines connecting pairs of points in an n point set A, the counting problem becomes an instance of the multidimensional sorting problem studied by Goodman and Pollack [GP83]. We would also like to study a problem posed by O'Rourke in [O84]. Given that a rectilinear polygon has a unique signature, how does one construct the polygon when its signature is given? The brute-force method takes exponential time but empirical tests show that a polynomial time algorithm may exist [O84]. We would like to see whether or not the reconstruction problem is NP-complete.

<u>Problem</u> 4.3 Shortest path problem in the presence of obstacles: Given a simple polygon P and an object, say a rectangle R in P with a certain orientation, find a <u>shortest</u> path, if it exists, from the initial position to a prespecified final position and orientation, with translation and rotation allowed. Here we assume that rotations are at no cost and the distance measure is in terms of Euclidean distance traveled by the center of the object.

As an initial step, one may want to consider the case where the obstacles are composed of n discrete points. For some configurations one may have to retract the object and then move (with possibly rotations) the object toward the goal position. As shown in Figure 5, the rectangle has to be moved backward, rotated in the final orientation and then translated to the goal position.

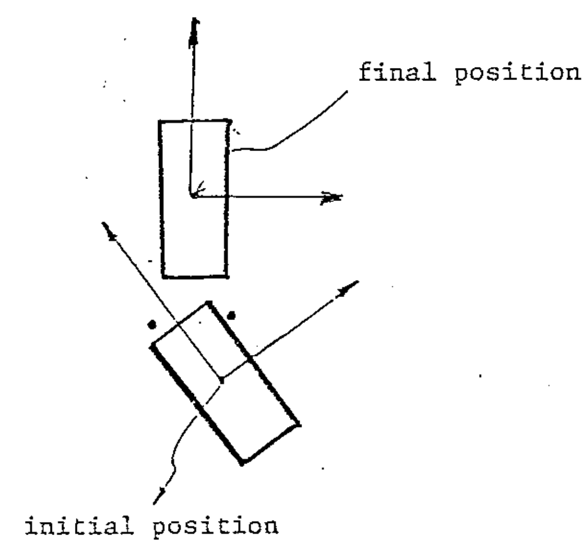


Figure 5. The rectangle must be retracted before it can be moved to the final position.

Problem 4.4 Limited covering problem: Given a set S of n intervals on the real line, suppose p is the minimum number of "probes" that are sufficient to cover the set. A probe is a position x and the subset S' of intervals covered by the probe consists of those intervals that contain x. We say p probes are sufficient and necessary if we can find a set Q of p positions such that the union of the subsets covered by each of the p probes is the set S. The problem is to determine r probes for a given r less than p such that the total number of intervals covered by these r probes is maximum.

Variations of this problem include limited covering of a set of n circular-arcs, or a set of rectangles, or a set of circles. In addition, the objects in the set may each have a certain weight, and we want to maximize the total weight of those objects covered by the probes.

Lastly the notion of dynamic computational geometry addressed in [At83,OW82] is also of interest. The objects within the framework of dynamic computational geometry are subject to motions which are low degree polynomials in time. Consider the case where the objects are points in Euclidean d-dimensional space and each coordinate of every point is a polynomial in time of degree less than or equal to k. Briefly we say that these points are in k-motion. Several results have been obtained and some open problems are listed in [At83]. In general we have two fundamental issues associated with dynamic problems, i.e., global or transient behavior of the moving objects and steady state. For instance, computation of the convex hulls of the points would mean the maintenance of the points on the hull at those instants of time when the hull changes. The last convex hull points will be said to be on the steady state hull. The steady state configuration is often not as difficult to compute as the computation of the first time at which the steady state occurs. In case of 1-motion, i.e., all the points move along a straight line, the steady state convex hull can be computed fairly easily in O(nlogn) time [At83]; whereas the time at which the steady state occurs is not easily attainable. In [At83] whether or not an $o(n^2)$ time algorithm for the latter problem exists is posed as an open problem; in [Ch84] such an algorithm, at least theoretically, has been reported. A preliminary study indicates that O(nlogn) time is sufficient. We intend to establish the above claim and investigate problems within the framework of dynamic computational geometry. For brevity we omit the description of such problems and instead, refer the reader to the article [At83], in which some interesting open problems are indicated.

[AHU74] Aho, A.V., J.E. Hopcroft, and J.D. Ullman, The Design and Analysis of Computer Algorithms, Addison-Wesley, Reading, MA., 1974.

[At83] Atallah, M.J., "Dynamic computational geometry," Proc. 24th IEEE Annual Symp. on Foundations of Comput. Sci., (Nov. 1983) 92-99.

[AvT81] Avis, D. and G.T. Toussaint, "An optimal algorithm for determining the visibility of a polygon from an edge," IEEE Trans. Comput., C-30, (1981) 910-1014.

[AvT81a] Avis, D. and G.T. Toussaint, "An efficient algorithm for decomposing a polygon into star-shaped polygons," Pattern Recognition, 13,6 (1981), 395-398.

[B83] Ben-Or, M, "Lower bounds for algebraic computation trees," Proc. 15th ACM Annual Symp. on Theory of Comput., (Apr. 1983) 80-86.

[BhT83] Bhattacharya, B.K. and G.T. Toussaint, "Efficient algorithms for computing maximum distance between two finite planar sets," J. Algorithms, 4 (June 1983) 121-126.

[Ca74] Cavendish, J.C., "Automatic triangulation of arbitrary planar domains for the finite element method," Int'l J. Numerical methods in Engineering, 8 (1974) 679-696.

[CD79] Chazelle, B. M. and D. Dobkin, "Decomposing a polygon into its convex parts," Proc. 11th ACM Symp. Theory of Comput. 1979, 38-48.

[Ch82] Chazelle, B.M., "A theorem on polygon cutting with applications," Proc. 23rd IEEE Annual Symp. on Foundations of Comput. Sci., (1982) 339-349.

[Ch84] Chazelle, B.M., "Fast searching in a real algebraic manifold with applications to geometric complexity," Tech. Rep. TR-CS-84-13, Dept. of Comput. Sci., Brown Univ., June 1984.

[ChG84] Chazelle, B.M. and L.J. Guibas, "Fractional cascading: A data structuring technique with geometric applications," manuscript, 1984.

[CGL83] Chazelle, B.M., L.J. Guibas and D.T. Lee, "The power of geometric duality," Proc. 24th IEEE Annual Symp. on Foundations of Comput. Sci., (Nov. 1983) 217-225.

[ChI83] Chazelle, B.M. and J. Incerpi, "Triangulating a polygon by divide-and-conquer," Proc. 21st Allerton Conference on Comm. Control and Comput., (Oct 1983) 447-456.

[CW83] Chin, F. and C.A. Wang, "Optimal algorithms for the intersection and the minimum distance problems between planar polygons", IEEE Trans. Comput., C-32,12 (Dec. 1983) 1203-1207.

[CW84] Chin, F. and C.A. Wang, "Minimum vertex distance between separable convex polygons," Info. Proc. Lett., 18 (Jan. 1984) 41-45.

[ChL83] Ching, Y.T. and D.T. Lee, "On finding the diameter of a set of lines," submitted for publication.

[CL83] Chu, W.H. and D.T. Lee, "The hidden line elimination problem revisited," Proc. 2nd Int'l Conference of Advanced Automation, Taipei, Taiwan, Dec. 1983, 469-476.

[Chv75] Chvatal, V., "A combinatorial theorem in plane geometry," J Comb. Theory Ser. B, 18 (1975), 39-41.

[Co84] Cole, R., "Slowing down sorting networks to obtain faster sorting algorithms," Tech. Rep. 117, Dept. of Comput. Sci., Courant Inst. of Math. Sci., April 1984.

[DLW84] DeRezende, P.J., D.T. Lee and Y.F. Wu, "Rectilinear shortest path with rectangular barriers," Tech. Rep. #84-04-FC-02, Dept. of EE/CS, Northwestern Univ., August 1984.

[E82] Edelsbrunner, H., "On computing the extreme distances between two convex polygons," Rep. 96, Technische Universitat Graz, IIG, Austria, 1982.

[EOW84] Edelsbrunner, H. J. O'Rourke and E. Welzl, "Stationing guards in rectilinear art galleries," submitted for publication.

[E184] El Gindy, H., "An efficient algorithm for computing the weak visibility polygon from an edge in simple polygons," manuscript, (Jan. 1984).

[E1A81] El Gindy, H. and D. Avis, "A linear algorithm for computing the visibility polygon from a point", J. Algorithms, 2,2 (June 1981) 186-197.

[FP75] Feng, H-Y, and T. Pavlidis, "decomposition of polygons into simpler components: feature generation for syntactic pattern recognition," IEEE Trans. Comput., C-24 (June 1975), 636-650.

[Fi78] Fisk, S., " A short proof of Chvatal's watchman theorem," J. Comb. Theory Ser. B, 24 (1978), 374.

[FrS75] Freeman, H. and R. Shapira, "Determining the minimum-area encasing rectangle for an arbitrary closed curve," Comm. ACM, 18,7 (July 1975) 409-413.

[GJ79] Garey, M.R. and D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, W.H. Freeman, San Francisco, CA 1979.

[GJPT78] Garey, M., D.S. Johnson, F.P. Preparata and R.E. Tarjan, "Triangulating a simple polygon," Info. Proc. Lett., 7,4 (1978) 175-180.

[G75] Gonzales, T., "Algorithms on sets and related problems," Tech. Rep. Dept. of Comput. Sci., Univ. of Oklahoma, 1975.

[GP83] Goodman, J.E. and R. Pollack, "Multidimensional sorting," SIAM J. Comput., 12 (1983), 484-507.

[GrY81] Graham, R.L. and F.F. Yao, "Finding the convex hull of a simple polygon," Tech. Rep. No. STAN-CS-81-887, Stanford Univ., 1981; also J. Algorithms, 4,4 (Dec 1983) 324-331.

[HM83] Hertel, S. and K. Mehlhorn, "Fast triangulation of simple polygons," Proc. FCT'83, 1983.

[H76] Honsberger, R. Mathematical Gems II, Mathematical Assoc. of America, 1976, 104-110.

[KKK80] Kahn, J., M. Klawe, and D. Kleitman, "Traditional galleries require fewer watchman," SIAM J. Alg. Disc. Method, 4,2 (1980) 194-206.

[Ki79] Kirkpatrick, D.G., "Efficient Computation of Continuous Skeletons," Proc. 20th IEEE Annual Symp. on Foundations of Comput. Sci., (Oct. 1979) 18-27.

[KiS82] Kirkpatrick, D.G. and R. Seidel, "The ultimate planar convex hull algorithm?" Proc. 20th Allerton Conference on Comm. Control and Comput., (1982) 35-42.

[KiS83] Kirkpatrick, D.G. and R. Seidel, "The ultimate planar convex hull algorithm?" Tech. Rep. 83-577, Dept. of Comput. Sci., Cornell Univ., Oct. 1983.

[LaLi81] Larson, R.C. and V.O.K. Li, "Finding Minimum Rectilinear Distance Paths in the Presence of Barriers", Networks, 11,3 (1981), 285-304.

[Le78] Lee, D.T., "Proximity and reachability in the plane," Tech. Rep. No. R-831, Coordinated Sci. Lab., Univ. of Ill., Urbana, IL 1978.

[Le80] Lee, D.T., "Two dimensional Voronoi diagram in the L_p-metric", J. ACM. (Oct. 1980) 604-618.

[Le80a] Lee, D.T., "On finding the convex hull of a simple polygon", Tech. Rep. No. 80-03-FC-01, EE/CS Dept., Northwestern Univ., 1980; see also Int'l J. Comput. and Infor. Sci., 12,2 (Apr. 1983) 87-98.

[Le82] Lee, D.T., "On k-nearest neighbor Voronoi diagrams in the plane", IEEE Trans. Comput., Vol. C-31, No. 6, (June 1982) 478-487.

[Le83] Lee, D.T., "Visibility of a Simple Polygon", Computer Vision, Graphics and Image Processing, 22 (1983), 207-221.

[LeL84] Lee, D.T. and A. Lin, "Computing visibility polygon from an edge," Tech. Rep. No. 84-02-FC-01, Dept. EE/CS, Northwestern Univ., Jan. 1984.

[LeL84a] Lee, D.T. and A.K. Lin, "Computational complexity of art gallery problems," submitted for publication.

[LeL84b] Lee, D.T. and A.K. Lin, "Finding the Delaunay triangulation of a simple polygon," in preparation.

[LeM83] Lee, D.T. and N. Megiddo, "Routing around circles," manuscript, Oct. 1983.

[LeP82] Lee, D.T. and F.P. Preparata, "An improved algorithm for the rectangle enclosure problem," J. Algorithms, 3,3 (Sept. 1982) 218-224.

[LeP84] Lee, D.T. and F.P. Preparata, "Euclidean shortest paths in the presence of rectilinear barriers", Networks, to appear.

[LeP84a] Lee, D.T. and F.P. Preparata, "Computational Geometry: A survey," IEEE Trans. on Comput., to appear. Also Tech. Rep. No. 84-05-FC-03, Dept. EE/CS, Northwestern Univ., August 1984.

[LeSc80] Lee, D.T. and B. Schachter, "Two algorithms for constructing Delaunay triangulations", Int'l. J. Comput. and Info. Sci., 9,3, (June 1980) 219-242.

[LoW79] Lozano-Perez, T. and M. A. Wesley, "An algorithm for planning collision free paths among polyhedral obstacles", Comm. ACM, 22,10 (Oct. 1979) 560-570.

[MT82] Manber, U. and M. Tompa, "Probablistic, nondeterministic and alternating decision trees," Tech. Rep. No. 82-03-01, Univ. Washington, 1982.

[MaS84] Mairson, H.G. and J. Stolfi, "Reporting and counting line segment intersection," Extended Abstract, Dept. Comput. Sci., Stanford Univ., 1984.

[McCA79] McCallum, D. and D. Avis, "A linear algorithm for finding the convex hull of a simple polygon," Info. Proc. Lett., 9 (1979) 201-206.

[McKT83] McKenna, M. and G.T. Toussaint, "Finding the minimum vertex distance between two disjoint convex polygons in linear time," Tech. Rep. No. SOCS-83.6, School of Computer Sci., McGill Univ. April 1983.

[McL76] McLain, D.H., "Two-dimensional interpolation from random data," Computer J. 19 (1976) 178-181.

[Meg83] Megiddo, N., "Linear time algorithm for linear programming in R^3 and related problems," SIAM J. Comput., 12,4 (Nov. 1983) 759-776.

[O82] O'Rourke, J., "The complexity of computing minimum convex covers for polygons," Proc 20th allerton Conference, Monticello, IL., Oct. 1982, 75-84.

[O82a] O'Rourke, J., "An alternate proof of the rectilinear art gallery theorem," Tech. Rep. 82/15, Johns Hopkins Univ., Dept. of Elec. Eng. and Comput. Sci., Dec. 1982. To appear in J. Geometry.

[O83] O'Rourke, J., "Galleries need fewer mobile guards: a variation on Chvatal's theorem," Geometricae Dedicata, 14, 1983, 273-283.

[O84] O'Rourke, J., "The signature of a plane curve," SIAM J. Comput., to appear.

[OSY83] O'Dunlaing, C., M. Sharir and C.K. Yap, "Retraction: A new approach to motion planning," Proc. 15th ACM Annual Symp. on Theory of Comput., April 1983, 207-220

[OtW82] Ottmann, T. and D. Wood, "Dynamic sets of points," Tech. Rep. No. CS-82-56, Dept. of Comput. Sci., Univ. of Waterloo, (Nov. 1982).

[PrSh84] Preparata, F.P. and M.I. Shamos, Computational Geometry, in preparation.

[Re72] Reingold, E.M., "On the complexity of some set algorithms," J. ACM, 19,4 (Oct. 1972) 649-659.

[Sa82] Sack, J., "An O(nlogn) algorithm for decomposing simple rectilinear polygons into convex quadrilaterals," Proc 20th Allerton Conference, Monticello, IL., Oct. 1982, 64-74.

[ScV80] Schoone, A.A. and J. van Leeuwen, "Triangulating a star-shaped polygon," Tech. Rep. No. RUV-CS-80-3, Univ. of Utrecht, April 1980.

[Sc81] Schwartz, J.T., "Finding the minimum distance between two convex polygons", Info. Proc. Lett., 13,4 (1981) 168-170.

[ScS82] Schwartz, J.T. and M. Sharir, "On the Piano Movers Problem", Tech. Rep. No.41, Dept. Comput., Sci, Courant Institute of Mathematical Sciences, New York Univ. Feb. 1982.

[Se84] Seidel, R., "A method for lower bounds for certain geometric problems," Tech. Rep. 84-592, Dept. of Comput. Sci., Cornell Univ., Feb. 1984.

[Sh75] Shamos, M.I., "Geometric complexity," Proc. 7th ACM Annual Symp. on Theory of Comput., (May 1975) 224-233.

[ShH75] Shamos, M.I. and D. Hoey, "Closest-point problems," Proc. 16th IEEE Annual Symp. on Foundations of Comput. Sci., (Oct. 1975) 151-162.

[ShH76] Shamos, M.I. and D. Hoey, "Geometric intersection problems," Proc. 17th IEEE Annual Symp. on Foundations of Comput. Sci., (Oct. 1976) 208-215.

[ShS84] Sharir, M. and A. Schorr, "On shortest paths in polyhedral spaces," Proc. 16th ACM Annual Symp. on Theory of Comput., April 1984, 144-153.

[Tou83] Toussaint, G.T., "Solving geometric problems with the rotating calipers," Proc. IEEE MELECON '83, Athens, Greece, May 1983.

[Tou83a] Toussaint, G.T., "An optimal algorithm for computing the minimum vertex distance between two crossing convex polygons," Proc. 21st Allerton Conference on Comm. Control and Comput., (Oct. 1983) 457-458.

[TouA82] Toussaint, G.T. and D. Avis, "On a convex hull algorithm for polygons and its application to triangulation problems," Pattern Recognition, 15,1 (1982) 23-29.

[VW80] Vaishnavi, V. and D. Wood, "Data structures for the rectangle containment and enclosure problems," Computer Graphics and Image Processing, 13 (1980), 372-384.

[vE80] van Emde Boas, P., "On the Ω(nlog n) lower bound for convex hull and maximal vector determination," Info. Proc. Lett., 10 (1980) 132-136.

[Ya81] Yao, A.C., "A lower bound to finding convex hulls," J. ACM, 28 (1981) 780-787.