Advanced Topics in Computational and Combinatorial Geometry

Prof. Micha Sharir
Spring 1993
Take Home Final Exam

Due back: July 5 before 4pm

Answer 3 of Problems 1–4 (30 points each) and 1 of Problems 5–7 (10 points)

Problem 1 (30 points)

Given n points in the plane, $p_1, \ldots, p_n$, each moving along some straight line at some fixed velocity (each point has a different line and a different velocity). Let $CH(t)$ denote the convex hull of these points at time t.

(a) Show that the number of combinatorial changes of $CH(t)$ as t varies is $O(n\lambda_s(n))$, for some constant s (give an upper bound for s). (**Hint**: For each fixed point p_i express the slope of the two edges of the hull incident to p_i (if such edges exist at all) in terms of upper or lower envelopes of appropriate functions of t, where each such function is defined by p_i and another p_j.)

(b) Describe an efficient (close to quadratic) algorithm for finding the smallest time t_0 such that $CH(t)$, for $t > t_0$, does not change combinatorially.

Problem 2 (30 points)

Given k convex polygons in the plane, $P_1, \ldots, P_n$, where P_i has n_i edges, for $i = 1, \ldots, k$. Put $n = n_1 + \cdots + n_k$. Let U denote the union of these polygons. Show that the complexity of U is $O(k^2 + n \log k)$, by applying the inductive proof technique as used in the proof of the Zone Theorem. Specifically, we want to bound the number of edges on the boundary of U. We remove a polygon P_i, consider the union U' of the remaining polygons, add P_i back, and want to estimate how many edges of the boundary of U' have been split into 2 subedges by the insertion of P_i. Show that the number of such edges is $O(n_i + k)$. (For example, charge each such split either to a vertex of P_i or to some topological change in the structure of the complement of U' that P_i generates—increase in the number of components, or merging two boundary components of the same 'hole' of the union; this part of the analysis is tricky; continue even if you don't get this part fully done.) Now obtain a recurrence relation for $\phi(k, n)$, the maximum complexity of the union of k polygons with a total of n edges, similar to that used in other proofs, and show that its solution satisfies the asserted bound.

Problem 3 (30 points)

Given a collection C of n discs in the plane, and an integer $k \leq n$.

(a) Apply the Clarkson-Shor technique to show that the number of vertices of the arrangement $\mathcal{A}(C)$, of the cicrcles bounding these discs, which are covered by at most k discs, is $O(nk)$. (Recall the bound proved in class for the complexity of the union of n discs.)

(b) If we are also given that no point of the plane is covered by more than k discs of C, show that the total combinatorial complexity of $\mathcal{A}(C)$ is $O(nk)$.

(c) Derive an algorithm that computes the maximum k for which there exists a point covered by k discs, whose complexity is close to $O(nk)$ (up to a polylogarithmic factor). (**Hint:** Use (b) in the analysis of the algorithm performance.)

Problem 4 (30 points)

Given a set S of n points in 3 dimensions. We want to preprocess the points for solving the *Post Office Problem:* Given a query point x, we want to find quickly the point of S nearest to x. Derive an algorithm that uses close to quadratic storage and preprocessing and answers queries in $O(\log n)$ time, using random sampling and the ϵ-net theory:

(a) Transform the problem to the problem of computing the intersection of n lower halfspaces in 4 dimensions, using standard techniques (we studied them in the first course). The query now asks for the hyperplane that lies vertically above the query point (in 4 dimensions) and is closest to the point along the vertical linepassing through it (the query point lies below all the hyperplens).

(b) Choose r points of S at random (r a big constant), compute the intersection of the r corresponding halfspaces (what is the complexity of the intersection?), and triangulate it into simplices.

(c) Apply the ϵ-net theorem to define appropriate subproblems for each simplex and continue the preprocessing recursively.

(d) Explain how a query is performed by searching with the query point through the recursive structure computed above.

(e) Analyze the expected complexity of the storage, preprocessing, and query time of the algorithm.

Problem 5 (10 points)

Consider the range space $(X, \mathcal{R})$, where X is a set of all points in 3 dimensions, and each range of $\mathcal{R}$ is a subset of X obtained by intersecting X with some ball. Show that the VC-dimension of this range space is finite. (**Hint:** Take a subset $A \subseteq X$ of n points, and derive an upper bound on the number of 'equivalence classes' of balls, where all balls in the same class intersect A in the same subset. Do it by defining in each class a 'canonical ball' in terms of some points of A.)

Problem 6 (10 points)

Given k convex polyhedra in 3 dimensions, with a total of n faces. Show that the complexity of their arrangement is $O(nk^2)$. (**Hint:** Show first that the complexity of the union or intersection of a pair of convex polyhedra having n_1, n_2 faces respectively is $O(n_1 + n_2)$. Sum this over all pairs of the given polyhedra. Then bound the number of vertices of the arrangement that can lie on a single edge of the union of any pair of the polyhedra (use convexity!).)

Problem 7 (10 points)

Consider a randomized incremental algorithm for constructing the Delaunay triangulation of n points in the plane. Show that the expected number of Delaunay triangles that the algorithm creates during the incremental process is at most $6n$ (we proved in class that it is $O(n)$). (**Hint**: Use the *backwards analysis* technique, which considers the algorithm as if it runs backwards in time. Show that the expected number of Delaunay triangles that are destroyed when we remove a random single point out of j given points, is at most 6.)