Assignment 3 - Computational Geometry (0368-4211)

Due: May 17, 2011

Problem 1

Let P be a set of n points in the plane. Design an algorithm that preprocesses P into some data structure, so as to answer quickly queries of the form: Given a query slab W (a region of the plane bounded by two paralle lines), count the number of points in $P \cap W$. The query time should be $O(\log n)$. How much preprocessing and storage does the algorithm need? (**Hints:** (a) Try to solve it when the orientation of the lines bounding W is fixed and known, and see when and how the solution changes as the orientation increases. (b) Try to use persistent search trees to minimize the storage and preprocessing.)

Problem 2

Let P be a convex polytope with n facets in three dimensions.

(a) Preprocess P into a linear-size data structure so that, given any query plane H, we can determine, in $O(\log n)$ time, whether H intersects P. (**Hint:** Use the normal diagram of P, as defined in a previous exercise.)

(b) Preprocess P into a linear-size data structure so that, given any query point q, we can determine, in $O(\log n)$ time, whether q lies inside P.

Problem 3

(a) Let L be a set of n lines in the plane. Preprocess L into a data structure of size $O(n^2)$, so that, given any query point q, one can compute, in $O(\log n)$ time, the number of lines of L below q.

(b) Suppose we know in advance the number m of queries. Modify the solution in (a) so that the total running time for answering the queries (including preprocessing cost) is roughly $O(n\sqrt{m} + m)$ (up to polylogarithmic factors).

(c) Modify the solution in (b) so that it also works when we do not know the number m in advance. In other words, for any m, after m queries have arrived, the total time spent so far should be roughly $O(n\sqrt{m} + m)$ (up to polylogarithmic factors).

Problem 4

Let P be a set of n points in the plane. Preprocess P into a data structure so that, given any query disk D of a fixed radius r, we can compute the number of points of P inside D. **Hint:** Use "duality", to replace the points by disks and the query by a point. The query time should be logarithmic. Express the space and preprocessing time in terms of n and of k, the number of intersections between the dual disks.

Problem 5

Let P be a set of n points in the plane in general position. Preprocess P into a data structure of linear size which can answer the following kind of queries in $O(\log n)$ time: In each query we are given a point q in the plane and a real number a, and we wish to determine whether q lies in the convex hull of the subset P_a of those points of P that lie to the left of $x = a$. (**Hints:** (a) Reduce the problem to the simpler problem where we need to determine whether q lies below the upper convex hull of P_a. (b) Think of the Graham walk, and use persistence to solve the latter problem.)