Assignment 2 - Computational Geometry (0368-4211)

Due: December 15, 2008

Problem 1

Modify the randomized incremental algorithm so that it computes the convex hull of a set of n points in $\mathbb{R}^4$. Explain how the modified algorithm works, and show that the expected number of facets (3-dimensional faces) that it generates is $O(n^2)$.

Problem 2

The *normal diagram* of a 3-D convex polyhedron K with n vertices is defined as a partition of the unit sphere $\mathbb{S}^2$ in $\mathbb{R}^3$ into regions $R(v_1), \ldots, R(v_n)$, one for each vertex v_i of K, so that a direction $u \in \mathbb{S}^2$ is in $R(v_i)$ if the plane supporting K and having u as its outward drawn normal touches K at v_i.

(a) What are the edges and vertices of this partition? How many are there? How fast can the diagram be computed from a reasonable representation (say, the DCEL) of K?

(b) Use the diagram to find efficiently all pairs (face f, vertex v) of K such that v is the vertex of K farthest from the plane containing f.

(c) The *width* of K is defined as the smallest distance between a pair of parallel supporting planes of K. Prove or disprove: The width is always attained for some pair of planes passing respectively through a face f and a vertex v of K as in (b) above.

(d) How fast can you calculate the width of K?

Problem 3

Use line sweeping to solve the following problem: Given a set S of n points in the plane, and a length a, find an axis-parallel square of side a that contains the maximum number of points of S. What is the running time of the algorithm? (**Hint:** Use a dual representation, where the roles of squares and points are interchanged.)

Problem 4

Let P and Q be two simple rectilinear polygons in the plane, with a total of n edges. That is, each of them is a closed polygonal curve that does not cross itself, and every edge of P or of Q is either horizontal (parallel to the x-axis) or vertical (parallel to the y-axis). Count the number of intersections between P and Q in $O(n \log n)$ time, using line sweeping. (Note that the actual number of intersections can be quadratic in n.)

Show that in fact this can also be done if only P is a simple rectilinear polygon and Q is an arbitrary set of n pairwise disjoint segments in the plane.

Problem 5

Let S be a set of n non-intersecting line segments in the plane. Describe an algorithm which runs in $O(n \log n)$ time to determine whether there exists a pair of segments in S at distance smaller than 1 from each other.