Assignment 5 - Computational Geometry (0368-4211)

Due: June 19, 2011 (in Tal's mailbox or in mine)

Problem 1

Let P be a given set of n point sites in three dimensions, and let H be the plane $z = x + y$. Consider the Voronoi diagram $\mathrm{Vor}(P)$ restricted to H. This is the partition of H into regions $V(p)$, for $p \in P$, such that $V(p)$ is the set of points on H whose nearest site is p.

(a) Describe the properties of these regions, and show that the resulting partition has linear complexity.

(b) Use the algebraic method in an explicit manner to obtain an $O(n \log n)$ algorithm for computing the restricted diagram on H.

Problem 2

We have reduced the problem of constructing $\mathrm{Vor}(P)$, for a set P of n points in the plane, to the problem of constructing the lower convex hull of the lifted points in three dimensions, and then we can compute the hull by the randomized incremental algorithm for convex hulls in three dimensions, in expected time $O(n \log n)$.

Translate this construction back to the plane, and describe it in terms of the original points (no lifting, no three dimensions). For example, how do the configuartions ("butterflies") that we maintain look like? What is a conflict? How do we implement an insertion of a new point? And so on. (There is no need to repeat the analysis of the storage and running time, but state the Clarkson-Shor bound that we need in terms of the setup in the plane: How many configurations of weight at most k we can have.)

Problem 3

Let S be a set of n pairwise disjoint discs $D_1, \ldots, D_n$ of arbitrary radii in the plane. For a point x not in any of the discs, the distance $d(x, D_i)$ is defined to be the length of the tangent from x to D_i. The Voronoi cell $V(D_i)$ consists of all points x which either lie in D_i or are such that $d(x, D_i) \leq d(x, D_j)$ for all j. The partition of the plane into these Voronoi cells will be called the Voronoi diagram of S.

(a) What is the shape of the Voronoi edges? of each Voronoi cell? What is the total complexity of the diagram?

(b) Assume that the plane where S lies is the xy-plane in 3-space. Show that there exists a set S^* of n points in three dimensions, such that (i) each point $p^* \in S^*$ projects vertically to the center of a disk $D \in S$, and (ii) The intersection of the xy-plane with the (standard, Euclidean) 3-dimensional Voronoi diagram of S^* is the Voronoi diagram of S, as defined above.

(c) Give an $O(n \log n)$ algorithm for computing the Voronoi diagram of S. (Remember Problem 1!)

Problem 4

Let P be a set of n points $p_1, \ldots, p_n$ in the plane. With each point p_i associate a positive weight w_i. Given a point x in the plane, define its distance from p_i to be $d(x, p_i) = w_i |xp_i|$, where $|xp_i|$ denotes the Euclidean distance between these points. Again, the Voronoi cell $V(p_i)$ of p_i is defined as

$$\{x \mid d(x, p_i) \leq d(x, p_j) \text{ for all } j\}$$

and the collection of these cells constitutes the Voronoi diagram of P.

(a) How do the bisectors between two sites look like?

(b) Give an example of a set P and weights w_i for which the resulting diagram has $\Omega(n^2)$ complexity.

Problem 5

Let L and R be two sets of points in the plane in general position, so that all the points of L lie to the left of the y-axis, and all the points of R lie to its right. Let P be the union $L \cup R$.

(a) Let ℓ be any horizontal line. Show that, as we move a point q on ℓ from left to right, its nearest neighbor in P changes through sites in L and then through sites in R. That is, once the nearest neighbor of q changes from a site in L to a site in R, it will never be again a site in L.

(b) Let C be the union of all Voronoi edges in Vor(P) that are defined by one nearest site in L and one nearest site in R. Using (a), show that each horizontal line intersects C in exactly one point. Show also that C is connected.

(c) Show that the highest edge of C is the Voronoi edge between the cells of p and q, where $p \in L$, $q \in R$, and pq is the higher edge (bridge) of the convex hull of P that is crossed by the y-axis.

(d) Using these properties, one can construct Vor(P) from Vor(L) and Vor(R) in *linear time*, by tracing C from its highest edge down. Explain *briefly* how this should be done, and obtain from this a divide-and-conquer algorithm for computing the Voronoi diagram of a set of n sites in the plane in (deterministic) $O(n \log n)$ time.