Eidgenössische Technische Hochschule Zürich
Ecole polytechnique fédérale de Zurich
Politecnico federale di Zurigo
Swiss Federal Institute of Technology Zurich

Institute for Theoretical Computer Science
Dr. B. Gärtner, Dr. M. Hoffmann and Marek Sulovský

24.11.2008

Computational Geometry Homework 4 HS08

URL: http://www.ti.inf.ethz.ch/ew/courses/CG08/

Exercise 1

Objective of this exercise is to develop a data structure for hexagonal window queries.

Given a set P on n points in $\mathbb{R}^2$, we want to preprocess it into a data struture that allows us to efficiently answer *hexagonal window queries*. To be precise, the data structure should allow us to count the number of points inside a given hexagonal window in time $O(\log n)$, and it should allow us to report the points inside a given hexagonal window in time $O(\log n + k)$, where k is the number of points inside the window.

A hexagonal window is a regular convex hexagon with arbitrary side length but fixed orientations, see the figure below.

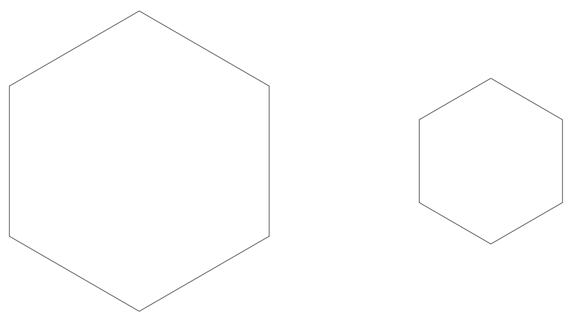

Figure 1: Hexagonal windows: regular hexagons with two vertical sides

Describe in detail the data structure that you use, including the time and space required to build it; also describe the actual window query, and how you arrive at the runtimes $O(\log n)$ and $O(\log n + k)$, respectively.

Exercise 2

Let P be a set of n points in the plane, each point $p_i \in P$ moving along a trajectory defined by a pair of polynomials $x_i(t)$ and $y_i(t)$ (giving its x and y coordinate at time t), each of maximum degree s. The nearest neighbour of $p_i \in P$ at time t is the point p_j with the minimal distance to p_i (at time t).

Show that the nearest neighbour of any point $p_i \in P$ changes at most $\lambda_{2s}(n)$ times and moreover, the closest pair in P changes at most $\lambda_{2s}(\binom{n}{2})$ times. You may assume general position in the sense, that the nearest neighbour of each point is unique except for finitely many values of t.

Show that for $s = 1$ the bound given above is asymptotically tight, i.e., give a point set P of n points moving along a line, such that the nearest neighbor of each point changes $\Omega(n)$ times and the closest pair of P changes $\Omega(n^2)$ times.

Hint: Consider very simple kinds of movement, for example where the points move at unit speed, half of of them along the x-axis and the other half along the y-axis.

Exercise 3

Consider the following problem: we are given a set I of n intervals in $\mathbb{R}$ and want to answer interval stabbing queries, i.e., given a point p, list the intervals containing p.

Describe a data structure, which you can construct in $O(n \log n)$ time with $O(n)$ storage and query time $O(\log n + s)$ where s is the number of points reported (note that the segment trees from the lecture require $O(n \log n)$ storage space).

Hint: The main idea is that given some base point x, you can divide all the intervals I into $I_<$ lying completely to the left of x, $I_>$ completely to the right of x and $I_\in$ containing x. This enables you to construct a tree-like data structure of required parameters.

Due date: 8.12.2008, 13h00