CS 6463-004 Computational Geometry, Fall 06

10/26/06

4. Homework

Due **11/9/06** before class

Always justify the runtime and the correctness of your algorithms, and try to make algorithms as efficient as possible.

1. Convex Hull of Intersections – 10 points
   Let $\mathcal{L}$ be a set of n straight lines in the plane, no two of which are parallel. Let S be the set of all $O(n^2)$ intersection points between any two lines in $\mathcal{L}$. Give an algorithm that computes $CH(S)$ in $O(n \log n)$ time, i.e., without computing all points in S explicitly. Hint: Sort all lines by slope, and prove that it is enough to consider only a certain subset of intersection points.

2. (4.16) Railway Tracks – 10 points
   On n parallel railway tracks n trains are going with constant speeds $v_1, \ldots, v_n$. At time $t = 0$ the trains are at positions $k_1, \ldots, k_n$.

   Give an $O(n \log n)$ time algorithm that detects all trains that at some moment in time are leading. To this end, use the algorithm for computing the intersection of half-planes.

3. LP in $\mathbb{R}^d$ – 10 points
   Read chapter 4.6 in the book which describes how the randomized algorithm for solving a linear program with n constraints and 2 variables can be generalized to n constraints and d variables.

   a) **(5 points)** Give a **short** description, in your own words, of the algorithm as well as a **brief** explanation of the expected runtime.

   b) **(5 points)** Analyse the worst-case runtime (in terms of n and d) and give a worst-case example.

4. Linear Separator – 10 points
   Given m red points $R = \{r_1, \ldots, r_m\}$ and n blue points $B = \{b_1, \ldots, b_n\}$ in the plane. The **linear separator problem** is to decide whether there exists a line l such that all points of R are on one side of l and all points of B are on the other side. (You may assume appropriate general position, and may disregard points that lie exactly on the line.) The task will be to use linear programming to solve the linear separator problem in expected linear time.

   (a) Use point-line duality to reformulate the problem. Then use linear programming to solve it. What is the runtime (in terms of m and n)?

   (b) Without using point-line duality, give a direct LP formulation of the problem. *(Hint: You may need more than one LP.)* How many constraints and how many variables does the LP have, and what is the objective function?