15-451/651 Algorithm Design & Analysis, Spring 2023

Recitation #12

Objectives

- Problem-Solving with Computational Geometry
- Using tools such as Convex Hull Algorithms, Sweep Line/Angle Algorithms, etc.

Recitation Problems

1. **(Circle with Most Points)** Given a set of points $S = \{p_1, \ldots, p_n\}$, and a radius $r > 0$, the goal is to find a circle of radius r that contains the maximum number of points from S. Give an $O(n^3)$ algorithm for this problem.
   Note: There exists an $O(n^2 \log(n))$ solution, try to find it!

2. **(Sophie's Walk Home)** Everyday Sophie walks home from CMU in a straight line. She also carries 64 stacks of ender pearls with her, i.e. she has an abundance of ender pearls. Her throwing radius is r, which means that the fastest way for her to get home is just to throw an ender pearl is to throw $\lfloor d/r \rfloor + 1$ ender pearls.

   For simplicity, say that CMU is at position $(0,0)$ and home is at $(d,0)$. There are n people stationed around her path, the i^{th} person is standing on point (x_i, y_i) and has a peripheral vision of radius p_i (i.e., they can see every point within distance p_i of their location). Sophie cannot ender pearl into, or out of those areas, but she can ender pearl over those areas if her path intersects their field of vision for less than a distance of r. Otherwise she must walk through those areas.

   Your goal is to find the minimum distance that Sophie has to walk to get home.

   (a) Give an $O(n)$ algorithm that constructs line segments representing the overlap of Sophie's path with each circle.

(b) Now that you have the $O(n)$ line segments, find in $O(n \log n)$ where the lines intersect to determine the regions where Sophie has to walk.

3. **(Width of a Set of Points)** You're given a set $S = \{p_1, \ldots, p_n\}$ of n points in the plane. A strip of width w is the region between two parallel lines, where the distance between the two lines is w. The goals is to find the strip of minimum width that contains all the points.

   (a) Give an $O(n^2)$ algorithm for this problem.

   (b) Give an $O(n \log n)$ algorithm for this problem.