Chapter 39

Exercises - Computational Geometry

By Sariel Har-Peled, December 4, 2014

This chapter include problems that are related to computational geometry.

39.1 Misc

39.1.1 Nearest Point to a Polygon

(20 pts.)

Given a convex polygon \( P \), its balanced triangulation is created by recursively triangulating the convex polygon \( P' \) defined by its even vertices, and finally adding consecutive diagonals between even points. For example:

Alternative interpretation of this construction, is that we create a sequence of polygons where \( P_0 \) is the highest level polygon (a quadrangle in the above example), and \( P_i \) is the refinement of \( P_{i-1} \) till \( P_{⌈\log n⌉} = P \).

1. (5 pts.) Given a polygon \( P \), show how to compute its balanced triangulation in linear time.

2. (15 pts.) Let \( T \) be the dual tree to the balanced triangulation. Show how to use \( T \) and the balanced triangulation to answer a query to decide whether point \( q \) is inside \( P \) or outside it. The query time should be \( O(\log n) \), where \( n \) is the number of vertices of \( P \). (Hint: use \( T \) to maintain the closest point in \( P_i \) to \( q \), and use this to decide in constant time what is the closest point in \( P_{i+1} \) to \( q \).)

39.1.2 Sweeping

(20 pts.)
(a) (5 pts.) Given two x-monotone polygons, show how to compute their intersection polygon (which might be made out of several connected components) in \(O(n)\) time, where \(n\) is the total number of vertices of \(P\) and \(X\).

(b) (5 pts.) You are given a set \(\mathcal{H}\) of \(n\) half-planes (a half plane is the region defined by a line - it is either all the points above a given line, or below it). Show an algorithm to compute the convex polygon \(\cap_{h \in \mathcal{H}} h\) in \(O(n \log n)\) time. (Hint: use (a).)

(c) (10 pts.) Given two simple polygons \(P\) and \(Q\), show how to compute their intersection polygon. How fast is your algorithm?

What the maximum number of connected components of the polygon \(P \cap Q\) (provide a tight upper and lower bounds)?

### 39.1.3 Robot Navigation

(20 pts.)

Given a set \(S\) of \(m\) simple polygons in the plane (called obstacles), with total complexity \(n\), and start point \(s\) and end point \(t\), find the shortest path between \(s\) and \(t\) (this is the path that a robot would take to move from \(s\) to \(t\)).

1. (5 pts.) For a point \(q \in \mathbb{R}^2\), which is not contained in any of the obstacles, the visibility polygon of \(q\), is the set of all the points in the plane that are visible from \(q\). Show how to compute this visibility polygon in \(O(n \log n)\) time.

2. (5 pts.) Show a \(O(n^3)\) time algorithm for this problem. (Hint: Consider the shortest path, and analyze its structure. Build an appropriate graph, and do a Dijkstra in this graph.))

3. (10 pts.) Show a \(O(n^2 \log n)\) time algorithm for this problem.

### 39.1.4 Point-Location

Given a \(x\)-monotone polygonal chain \(C\) with \(n\) vertices, show how to preprocess it in linear time, such that given a query point \(q\), one can decide, in \(O(\log n)\) time, whether \(q\) is below and above \(C\), and what is the segment of \(C\) that intersects the vertical line that passes through \(q\). Show how to use this to decide, in \(O(\log n)\) whether a point \(p\) is inside a \(x\)-monotone polygon \(P\) with \(n\) vertices. Why would this method be preferable to the balanced triangulation used in the previous question (when used on a convex polygon)?

### 39.1.5 Convexity revisited.

(a) Prove that for any set \(S\) of four points in the plane, there exists a partition of \(S\) into two subsets \(S_1, S_2\), such that \(CH(S_1) \cap CH(S_2) \neq \emptyset\).

(b) Prove that any point \(x\) which is a convex combination of \(n\) points \(p_1, \ldots, p_n\) in the plane, can be defined as a convex combination of three points of \(p_1, \ldots, p_n\). (Hint: use (a) and induction on the number of points.)

(c) Prove that for any set \(S\) of \(d + 2\) points in \(\mathbb{R}^d\), there exists a partition of \(S\) into two subsets \(S_1, S_2\), such that \(CH(S_1) \cap CH(S_2) \neq \emptyset\), \(S = S_1 \cup S_2\), and \(S_1 \cap S_2 = \emptyset\). (Hint: Use (a) and induction on the dimension.)
39.1.6 Covered by triangles

You are given a set of $n$ triangles in the plane, show an algorithm, as fast as possible, that decides whether the square $[0, 1] \times [0, 1]$ is completely covered by the triangles.

39.1.7 Nearest Neighbor

Let $P$ be a set of $n$ points in the plane. For a point $p \in P$, its nearest neighbor in $P$, is the point in $P \setminus \{p\}$ which has the smallest distance to $p$. Show how to compute for every point in $P$ its nearest neighbor in $O(n \log n)$ time.