Karlsruher Institut für Technologie
Fakultät für Informatik
ITI Sanders/Wagner

Computational Geometry
Summer term 2018

Exercises 4 - Voronoi-Diagrams & Delaunay Triangulations

Discussion: Friday, June 15th, 2018

Lecture 7 (30.05.2018)

Exercise 1 – Voronoi-Cells.

1. Prove the following theorem.

   Theorem 1. *Let* $P \subset \mathbb{R}^2$ *be a set of* n *points.* $\mathrm{Vor}(P)$ *consists of at most* $2n-5$ *vertices and at most* $3n-6$ *edges.*

2. Show that this theorem implies that the average number of vertices of a Voronoi Cell is less than six.

Exercise 2 – Beach Line. In the lecture we have seen that the Voronoi diagram is constructed by means of a beach line.

1. Give an example where the parabola defined by some site p contributes more than one arc to the beach line.
2. Can you give an example where a point contributes a linear number of arcs?

Exercise 3 – Next Neighbor. Let P be a set of n points in the plane, the *next neighbor* $a(p)$ of $p \in P$ is the point in P with smallest distance to p. Describe an algorithm that computes for each point $p \in P$ its next neighbor $a(p)$ using $O(n \log n)$ time in total.

Exercise 4 – Nuclear Power Plants. Suppose that your are against nuclear power plants and that you want to live as far from any nuclear power plant as possible. However, at the same time you want to live in a certain region. Where should you settle down? We can formalize the problem as follows.

FindPlaceToLive
Given: Set S of n points (the nuclear power plants) and a rectangle R (the region in which you would like to live).
Find: Point $p \in R$ (your prospective place of residence) whose distance $\min_{s \in S} d(p, s)$ to the next point $s \in S$ is maximized.

Assume that your are already given the Voronoi diagram $\mathrm{Vor}(S)$ of S.

1. Show that your prospective residence p is a vertex of $\mathrm{Vor}(S)$, a corner of R, or the intersection of R with an edge of $\mathrm{Vor}(S)$.
2. Describe an algorithm that solves FindPlaceToLive in $O(n)$ time.
3. Assume that the preferred region R is a simple polygon consisting of m corners. Describe an algorithm that solves FindPlaceToLive in $O(n+m)$ time.

Lecture 8 (06.06.2018)

Exercise 5 – Triangulations. Let P be a set of n points in the plane.

1. Show that there is no set P of points that has more than $2^{\binom{n}{2}}$ different triangulations.
2. Give an example of a point set P such that each triangulation of P contains at least one point having degree $n-1$.

Exercise 6 – Delaunay Triangulation. Let $P \subset \mathbb{R}^2$ be a finite set of points in general position. Further, let $q \in \mathbb{R}^2 \setminus P$ be a point that is contained in the convex hull of P. Let p_i, p_j, p_j denote the corners of a triangle in the Delaunay triangulation of P that contains q – there can be at most two such triangles. Show that $\overline{qp_i}$, $\overline{qp_j}$, $\overline{qp_k}$ are edges of the Delaunay triangulation of $P \cup \{q\}$.

Exercise 7 – Minimum Spanning Tree. Let P be a set of n points in the plane. The Euclidean minimum spanning tree (EMST) of P is a tree of minimum edge length connecting all points in P.

1. Show that the set of edges of a Delaunay triangulation of P contains an EMST for P.
2. Describe an algorithm to compute an EMST for P in $O(n \log n)$ time.

Exercise 8 – Gabriel Graph. We define the *Gabriel Graph* of a set P of points in the plane as follows. Two points p and q are connected by an edge of the Gabriel graph iff the disc with diameter $d(p,q)$ containing p and q does not contain any other point of P.

1. Show that the set of edges of a Delaunay triangulation of P contains an Gabriel graph for P.
2. Prove that p and q are adjacent in the Gabriel graph of P if and only if the Delaunay edge between p and q intersects its dual Voronoi edge.
3. Describe an algorithm that computes the Gabriel graph for P in $O(n \log n)$ time, where $n = |P|$.