AMS 345/CSE 355 (Spring, 2008) Joe Mitchell

COMPUTATIONAL GEOMETRY
Homework Set # 6

Due at the beginning of class on Wednesday, April 23, 2008.

Recommended Reading: O'Rourke, Chapter 5.

In all of the exercises, be sure to give at least a brief explanation or justification for each claim that you make.

(1). [30 points] Let S be a set of n points in the plane in general position (no three are collinear, no four are cocircular). Let h denote the number of points of S that are vertices of the convex hull, $CH(S)$.

(a). Consider the Delaunay diagram, $\mathcal{D}(S)$, of the set S; since no four points are cocircular, we know that the Delaunay diagram is a triangulation, with each face (except the face at infinity) being a triangle (and each point of S being a vertex of some triangle). As a function of n and h how many triangles does $\mathcal{D}(S)$ have? How many Delaunay edges are there in $\mathcal{D}(S)$?

(b). Now we are interested in decomposing the convex hull of S into pentagons (5-sided polygons, **not** necessarily convex), such that each point of S is a vertex of some pentagon. Such a decomposition is called a "pentagonalization" of S.

(i). Give an example of a set S with $|S| \geq 5$ such that S does *not* have a pentagonalization. Justify briefly your claim.

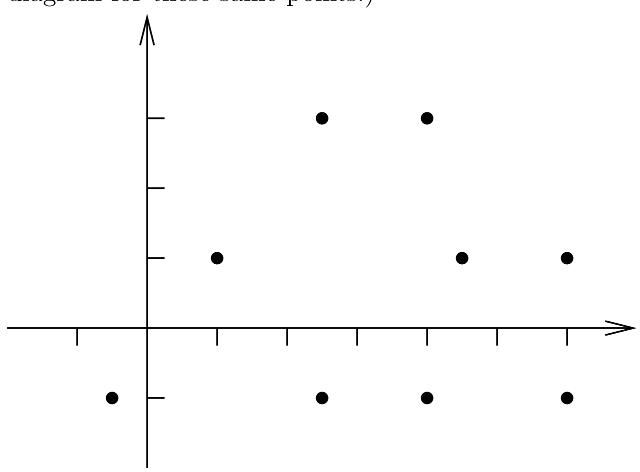
(ii). Give an example of a set S with $|S| \geq 6$ such that S *does* have a pentagonalization.

(iii). (Extra Credit) Can you characterize *when* a point set S has a pentagonalization and when it does not, based, e.g., on looking only at the two numbers $n = |S|$ and h?

(Open problem, as far as I know): Given a point set S, is there an efficient algorithm to determine if there exists a "convex pentagonalization" of S, in which each pentagon is a convex polygon?

(2). [25 points] O'Rourke, problem 7, section 5.5.6, page 178. For part (c), be sure to state the running time of your algorithm and justify.

(3). [45 points] Let S be the set of points {(-1,-2), (2,2), (5,6), (8,6), (9,2), (12,2), (12,-2), (8,-2), (5,-2)}. (In HW5 you constructed the Delaunay diagram for these same points.)



(a). Draw the (directed) NNG for S.

(b). Draw the relative neighborhood graph (RNG) for S. (See O'Rourke, problem 7, section 5.5.6, page 178, for the definition of the RNG.)

(c). Construct the MST by Kruskal's algorithm applied to the Delaunay diagram.

(d). Construct the approximate TSP tour obtained by doubling the MST and shortcutting, as in Figure 5.18. (Begin the walk around the MST at point (-1,-2) and go clockwise around the MST, as done in the example in the text.) Also determine the *optimal* TSP tour (easy to do in this case – can you justify your answer?).

(e). Construct the "furthest-point Voronoi diagram" of S. Read problem 11, section 5.5.6, and see Figure 5.19 for the definition of the diagram.