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

COMPUTATIONAL GEOMETRY
Homework Set # 6 – Solution Notes

(1). *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 $\mathcal{D}(S)$?*

Actually, the fact that we are dealing with a Delaunay triangulation is irrelevant for this question: the combinatorics depend solely on the input, T, being a triangulation. Let t denote the number of triangles in T. There is one face at infinity, so, when we view T as a planar graph, we know it has $t+1$ faces, t of which are 3-sided and one of which is h-sided. Thus, the sum of the degrees of the faces is $3t+h$, which we know is twice the number of edges; thus, $2e = 3t+h$. We also know from Euler's formula that $(t+1)-e+n = 2$. Putting $e = (3t+h)/2$ into this, we get $(t+1)-(3t+h)/2+n = 2$. We solve for t to get $t = 2n-h-2$ for the number of triangles. Then, $e = (3t+h)/2 = 3n-h-3$ is the number of edges.

(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.*

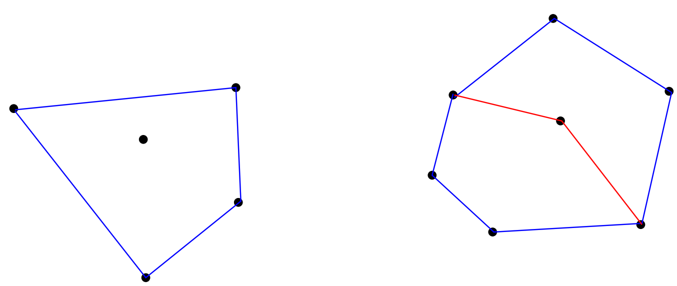
(i). *Give an example of a set S with $|S| \geq 5$ such that S does* not *have a pentagonalization. Justify briefly your claim.*

See below, left. The example has $h = 4$ points on the hull, with 1 point interior to the hull. The one interior point must be joined to at least two hull vertices in any decomposition of the hull into polygons such that each point of S is a vertex of some polygon. If the interior point is joined to two hull vertices, the hull is split into either a pentagon and a triangle or into two quadrilaterals; in either case, we do not get a pentagonalization. If the interior point is joined to 3 hull vertices, we get two triangles and a quadrilateral; if it is joined to 4 hull vertices, we get 4 triangles. Thus, in no case do we get only pentagons.

(ii). *Give an example of a set S with $|S| > 5$ such that S* does *have a pentagonalization.*

Of course, if we allow $|S| = 5$, a trivial example is just to place the 5 points in convex position: the convex hull is already a pentagon.

An example is shown below (right) in which $|S| = 7$, with $h = 6$. By joining the one interior point to the hull vertices with the two red edges, we decompose the hull into two pentagons, resulting in a pentagonalization.



(2). *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.*

(a). The brute force algorithm simply tests for each pair, (p_i, p_j) $(i \neq j)$, if the Lune(p_i, p_j) is empty of other points on its interior (this is done naively in time $O(n)$, by looping through all other points p_k and testing in time $O(1)$ if p_k is at distance less than $|p_i p_j|$ from both p_i and from p_j (actually, we would check the squared distances, so as to avoid taking square roots!)). Since there are $O(n^2)$ pairs (p_i, p_j), this brute force algorithm is $O(n^3)$ overall.

(b). We need to show that $RNG \subseteq \mathcal{D}(P)$. Let (p,q) be an edge of the RNG. Then, by definition, Lune(p,q) contains no sites of P. We claim that (p,q) is an edge of the Delaunay diagram; i.e., that there exists a site-free circle, C, passing through p and q and no other sites. Select C to be the unique circle with diameter pq (centered at the midpoint, c, of segment pq, with radius $r = |pq|/2$). It is easy to see that the interior of C is contained in Lune(p,q) (which is an open set, by definition).

(Proof: Let $z \in int(C)$. Then, $dist(z,c) < r$. By the triangle inequality, $dist(z,p) \leq dist(z,c) + dist(c,p)$, so $dist(z,p) < r + dist(c,p) = r + r = |pq|$. Thus, z is interior to the disk of radius $|pq|$ centered at p. Similarly, z is interior to the disk of radius $|pq|$ centered at q. Thus, by definition, z is in the Lune(p,q).)

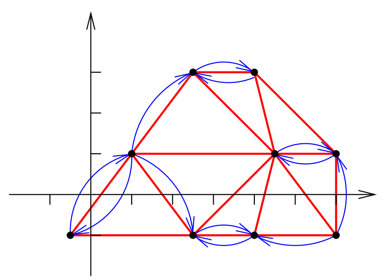
Since Lune(p,q) is site-free, and the interior of C is contained in the Lune(p,q), we get that C is a witness to the Delaunayhood of (p,q).

(c). Using part (b), we know that the edges of the RNG must be edges of the Delaunay diagram, which has only $O(n)$ edges (remember, it is a planar graph with n nodes!). Thus, instead of checking *every* pair (p_i, p_j), as we did in (a), we need only check those pairs that correspond to the $O(n)$ Delaunay edges. (The Delaunay edges are all identified in time $O(n \log n)$, using either the divide-and-conquer or the plane sweep algorithm for building the Voronoi/Delaunay.) This reduces the time bound of the brute force algorithm to $O(n^2)$.

(It is in fact possible to construct the RNG even faster – in time $O(n \log n)$ – by first building the Delaunay diagram (which takes time $O(n \log n)$) and then being even more clever in checking which edges correspond to empty lunes. I omit the details here.)

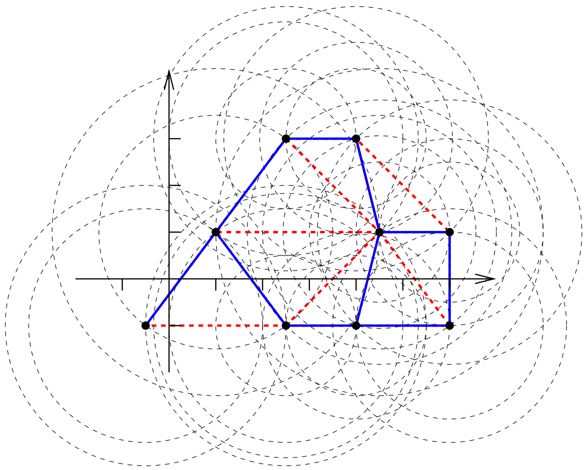
(3). *Let S be the set of points {(-1,-2), (2,2), (5,6), (8,6), (9,2), (12,2), (12,-2), (8,-2), (5,-2)}. (In HW5 you constructed the Delaunay diagram for these same points.)*

(a). *Draw the (directed) NNG for S.* See figure below. The NNG (directed) edges are drawn in blue; the Delaunay edges are shown in red.



(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.)*

I draw the diagram below, showing the RNG edges in blue. All RNG edges are edges of the Delaunay; some Delaunay edges (shown dashed red) are not RNG edges. I also draw the relevant circles (dashed black) for each Delaunay edge, so we can see the "lunes" defined for each Delaunay edge and determine if they are site-free (as needed to be RNG edges).



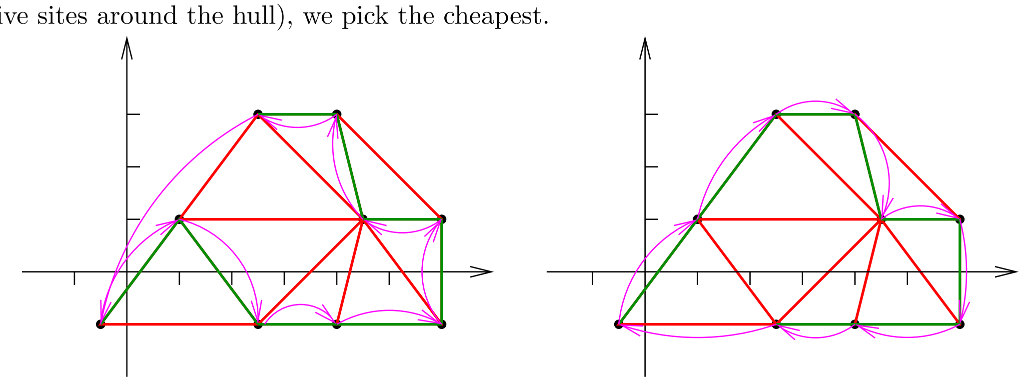
(c). *Construct the MST by Kruskal's algorithm applied to the Delaunay diagram.*

We consider the Delaunay edges in increasing order by length, adding an edge if it does not create a cycle with the edges we have already added. We end up with 3 edges of length 3, then 2 edges of length 4, then 1 edge of length $\sqrt{5}$, then 2 edges of length 5 Note that we can use EITHER edge ((2,2),(5,-2)) OR edge ((2,2,),(5,6)); I show both possibilities in the figure below. All other edges are uniquely determined. In either case, the MST is in fact a path in this example. MST edges are shown in dark green in the figures below.

(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?).*

The TSP tours obtained for each of the two MST's are shown below, with magenta arcs. Because the MST's are just paths (in this case, not in general), and we start the walk at one of the endpoint of the path, the doubling and shortcutting is particularly simple.

The optimal tour is in fact given in the figure on the right. (In general, one does NOT expect that a doubled MST tour will give optimal! We know it will always give a tour whose length is most most twice optimal.) One can argue it is optimal as follows: First, the sites on the convex hull of the sites must appear in the same order along the tour as they occur around the hull (otherwise, one can argue that the tour would self-intersect, which is not the case for an optimal tour). Now, the only question is how to insert the point (9,2) into the tour in the cheapest possible way. By checking the 8 possibilities (one for each pair of consecutive sites around the hull), we pick the cheapest.



(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.*

I show the diagram in blue below. Only sites 1, 2, 3, 4, 5 (the vertices of the convex hull of sites) end up having a nonempty region (this is always true for furthest-point Voronoi diagrams). I label the region corresponding to site i by drawing a circle with "i" inside, as the textbook does.

Also, I show the circle centered at the Voronoi vertices that pass through the corresponding 3 sites: Note that for furthest-point Delaunay (the dual of the furthest-point Voronoi), we have the “full circle” property instead of the “empty circle” property: All sites lie inside such a circle.