CS 6301.008.18S Lecture—February 15, 2017

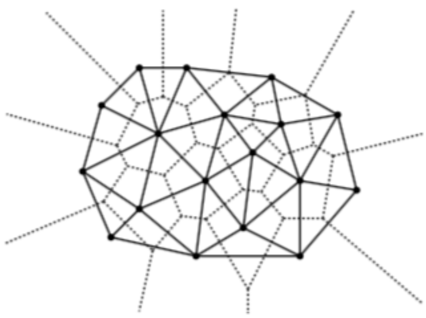
Main topics are #Delaunay_triangulations.

Prelude

- I'm sorry about taking so long to provide feedback on the homework. I'm going to make it due next Thursday February 22nd so you at least have a few week days with the feedback.
- And if you're having difficulty with Problem 2, the algorithm I give on Tuesday may inspire you.
- I'm going to ask for project proposals soon. One page describing something you want to work on for the rest of the semester, possibly with others. I think I'll give you more details on Tuesday and make them due either the following Tuesday or Thursday in leu of assigning the next homework.

Delaunay Triangulations

- Last time we discussed the Voronoi diagram: Given a set $P = \{p_1, \ldots, p_n\}$ of n sites, partition the plane into cells. The cell $V(p_i)$ of a site is the set of points closer to p_i than any other site.
- The *dual* of the Voronoi diagram is another graph. Its vertices are the sites themselves. For each edge of the Voronoi diagram between sites p_i and p_j, add an edge between p_i and p_j.



- You get another planar subdivision. Assuming general position, each Voronoi vertex had degree 3. Therefore, each face (except the outer one) of the dual graph is a triangle. We call such planar subdivisions *triangulations*.
- This particular dual subdivision the *Delaunay triangulation*.

Why Triangulations?

- Before we go into specifics on the Delaunay triangulation, let's talk about why triangulations are useful in the first place.
- Let $h : R^2 \to R$ be a continuous function over the plane called the *height* function.

- The set of points (x, y, h(x, y)) is a 2d surface in 3d called a *terrain*. Note every vertical line hits the terrain in exactly one position.
- You can naturally model the height of points on the Earth's surface using a terrain.
- However, a computer can't store the uncountably infinite set of points in R^2, so instead we just store a subset of the points, say P = {p_1, …, p_n}.
- We still need to estimate the height of any other point, so we do the following: we find some triangulation of the points of P. We then estimate h using a piecewise linear function that is linear within each triangle.
- Since we're modeling heights on the Earth's surface, we call storage strategies like this a *digital elevation models* or DEM. This particular one is called a *triangulated irregular network* (TIN).
- For reasons I'll get into later today, Delaunay triangulations give particularly nice TINs.

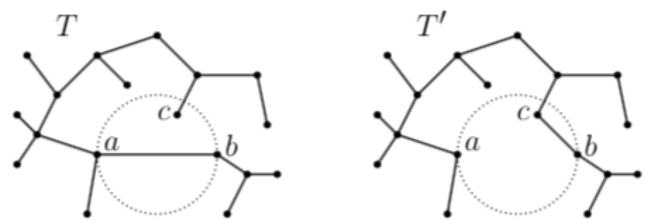
Basic Properties

- Delaunay triangulations have some nice properties that follow directly from what we know about Voronoi diagrams.
- Convex hull: The boundary of the external face is the convex hull of P.
- Circumcircle property: The circumcircle of any triangle is empty.
  - The center of the circle is a Voronoi vertex, so the triangle's vertices are nearest neighbors to the center.
- Empty circle property: Two sites p_i and p_j are connected by an edge if and only if there is an empty circle passing through p_i and p_j.
  - If they're connected, there's a Voronoi edge between them and we can center the circle on that edge.
  - If there's an empty circle, then the circle center is equidistant to p_i and p_j and they are the nearest neighbors to the center. The center is on a Voronoi edge shared by p_i and p_j.
- Closest pair property: The closest pair of sites share an edge.
- Note the way I defined the Delaunay triangulation, if four sites are cocircular and the circle is empty, then you'll actually get a square or worse around the circle center. It's only really a triangulation if you assume general position, or add additional edges to fill in these squares.
- Assuming general position, if there are h sites on the convex hull, then the Delaunay triangulation has exactly 2n - 2 - h triangles and 3n - 3 - h edges.

Minimum Spanning Trees

- Now we'll get into more surprising properties of the Delaunay triangulation.

- First off, it hides within it a good way to simply connect all the points together.
- The *Euclidean graph* of P is the complete graph with P as its vertices. Each edge p_i p_j is given a weight w(p_i p_j) = ||p_i p_j||.
- The *minimum spanning tree* of P is a subset of n - 1 edges from the Euclidean graph that connects the points (into a tree) such that the total weight is minimized.
- Suppose we want to compute the minimum spanning tree of P. We could compute the Euclidean graph explicitly and then run, say, Kruskal's algorithm. But there are (n choose 2) edges, so the algorithm will take O(n^2 log n) time.
- Instead, we could use the following theorem.
- Theorem: The minimum spanning tree T of P is a subgraph of the Delaunay triangulation.
- Proof:
    - Suppose to the contrary that there is an edge ab in T that is *not* in the Delaunay triangulation.
    - Therefore, there is a circle with diameter ab that strictly contains a site c.



    - Removing ab from T splits it into two subtrees. Reconnect them them with an edge from a or b to c to create tree T'.
    - This new edge is shorter than the old one, so T' weighs less.
- So, instead of computing the whole Euclidean graph, you can instead compute the Delaunay triangulation in O(n log n) time using Tuesday's algorithm for Voronoi diagrams or the more direct approach I'll teach next Tuesday. Then you compute an MST in the Delaunay triangulation in O(n log n) time total.
- Now, you might guess that the Delaunay triangulation is actually the triangulation of minimum total edge length. **This is wrong.** Finding the minimum weight triangulation is actually NP-hard.

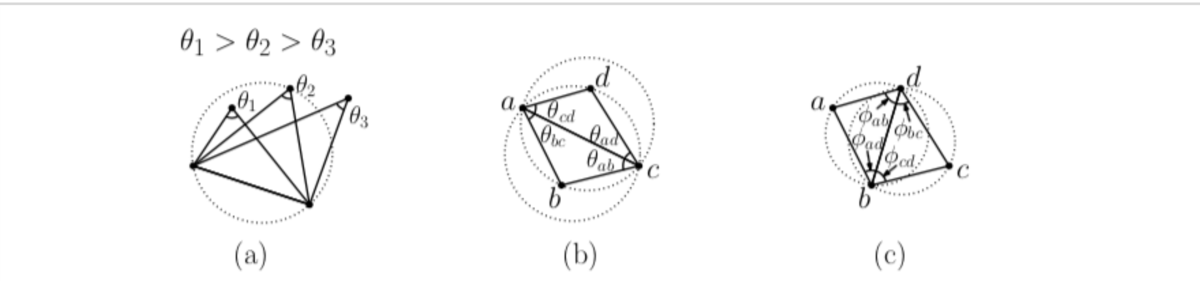
Spanners

- Second, the Delaunay triangulation provides a reasonable transportation network between sites.
- Let G be any graph with vertices from P.
- Let delta_G(p, q) be the shortest path *in G* from site p to q.
- Given some t ≥ 1, we say G is a *t-spanner* if for any p, q in P, delta_G(p, q) ≤_t ||pq||.
- For t = 1, G must be the complete graph (assuming general position).

- Theorem [Keil, Gutwin '92]: The Delaunay triangulation of P is a t-spanner with t = 4 pi sqrt(3) / 9 about equals 2.418.

Maximizing Angles

- Earlier, I mentioned how the Delaunay triangulation is a particularly good choice for making a triangulated irregular network.
- Suppose we want to build a TIN over P. One thing we'd like to avoid is very skinny triangles. In particular, a pair of skinning triangles sharing a long edge means you're basing the height of a point on that long edge based entirely on a pair of very far away sites and not either site closer by.
- It turns out the Delaunay triangulation maximizes the smallest angle over all the triangles. Even stronger, among all triangulations maximizing the smallest angle, it maximizes the second smallest. Even stronger, among all triangulations maximizing the smallest and then the second smallest angle, it maximizes the third, and so on.
- Let's make this claim more formal. Any triangulation is associated with an *angle sequence* <alpha_1, alpha_2, …, alpha_m> which is the set of triangle angles sorted in increasing order.
- If we were to compare two angle sequences lexicographically, we would compare their first angles. In the event of a tie, we would compare their second angles, and so on.
- Theorem: Among all triangulations of P, the Delaunay triangulation has the lexicographically largest angle sequence.
- The proof uses an important geometric fact:
- Consider a circle through two points. Let theta_1 be the angle formed with a middle point inside the circle, theta_2 be the angle formed with a third point on the circle, and theta_3 be the angle formed with a third point outside the circle. Then
  - theta_1 > theta_2 > theta_3
- The book proves that given any triangulation that doesn't have the empty circle property, there exists some convex quadrilateral abcd where the long diagonal ac is in the triangulation.
- We can replace this long diagonal with the short one bd. This operation is called an *edge flip*.
- Before the flip, the circumcenters for both triangles contains the other point on the quad.



- After the flip, the two circumcircles do not contain the forth point.
- But take, for example, this angle labeled theta_ab. c lies on the boundary of a circle through those points, but d lies interior. So this angle labeled phi_ab is bigger. Similarly, phi_bc > theta_bc, phi_cd > theta_cd, and phi_da > theta_da.
- You can also argue that the other two new angles are not smaller than the smallest phi angle.
- So we took six angles and replaced them six others where the smallest of the angles in the first set is bigger than the smallest in the second set.
- We can repeat this process as long as there is some non-empty circle. Since there are a finite number of triangulations, the process will terminate.
- And it will terminate with the Delaunay triangulation, meaning it has the lexicographically largest angle sequence.

- Next week, I'll describe a randomized incremental construction algorithm for constructing a Delaunay triangulation. Since it and the Voronoi diagram are duals, you can use it to construct the Voronoi diagram as well!