21 Delaunay Triangulations

The triangulations constructing by plane-sweep are typically of inferior quality, that is, there are many long and skinny triangles and therefore many small and many large angles. We study Delaunay triangulations which distinguish themselves from all other triangulations by a number of nice properties, including they have fast algorithms and they avoid small angles to the extent possible.

Plane-sweep versus Delaunay triangulation. Figures 97 and 98 show two triangulations of the same set of points, one constructed by plane-sweep and the other the Delaunay triangulation. The angles in the Delaunay triangulation seem consistently larger than those in the plane-sweep triangulation. This is not a coincidence and it can be proved that the Delaunay triangulation maximizes the minimum angle for every input set. Both triangulations contain the edges that bound the convex hull of the input set.

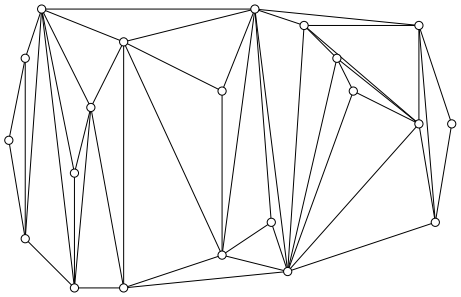

Figure 97: Triangulation constructed by plane-sweep. Points on the same vertical line are processed from bottom to top.

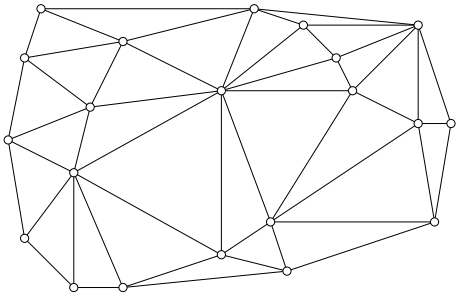

Figure 98: Delaunay triangulation of the same twenty-one points triangulated in Figure 97.

Voronoi diagram. We introduce the Delaunay triangulation indirectly, by first defining a particular decomposition of the plane into regions, one per point in the finite data set S. The region of the point u in S contains all points x in the plane that are at least as close to u as to any other point in S, that is,

$$V_u \quad = \quad \{x \in \mathbb{R}^2 \mid \|x-u\| \leq \|x-v\|, v \in S\},$$

where $\|x-u\| = [(x_1-u_1)^2 + (x_2-u_2)^2]^{1/2}$ is the Euclidean distance between the points x and u. We refer to V_u as the *Voronoi region* of u. It is closed and its boundary consists of *Voronoi edges* which V_u shares with neighboring Voronoi regions. A Voronoi edge ends in *Voronoi vertices* which it shares with other Voronoi edges. The *Voronoi diagram* of S is the collection of Voronoi regions, edges and vertices. Figure 99 illustrates the definitions. Let n be the number of points in S. We list some of the properties that will be important later.

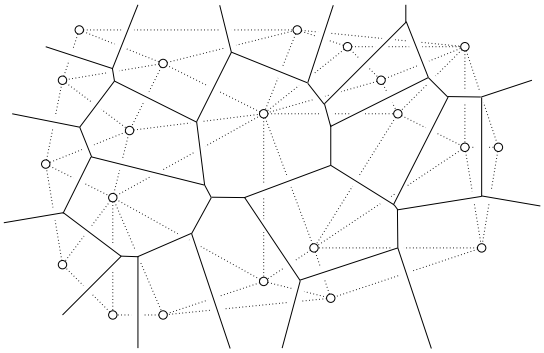

Figure 99: The (solid) Voronoi diagram drawn above the (dotted) Delaunay triangulation of the same twenty-one points triangulated in Figures 97 and 98. Some of the Voronoi edges are too far out to fit into the picture.

- Each Voronoi region is a convex polygon constructed as the intersection of $n-1$ closed half-planes.
- The Voronoi region V_u is bounded (finite) iff u lies in the interior of the convex hull of S.
- The Voronoi regions have pairwise disjoint interiors and together cover the entire plane.

Delaunay triangulation. We define the *Delaunay triangulation* as the straight-line dual of the Voronoi diagram. Specifically, for every pair of Voronoi regions V_u and V_v that share an edge, we draw the line segment from u to v. By construction, every Voronoi vertex, x, has $j \geq 3$ closest input points. Usually there are exactly three closest

points, u, v, w, in which case the triangle they span belongs to the Delaunay triangulation. Note that x is equally far from u, v, and w and further from all other points in S. This implies the *empty circle property* of Delaunay triangles: all points of $S - \{u, v, w\}$ lie outside the circumscribed circle of uvw. Similarly, for each Delaunay edge uv, there is a circle that passes through u and v such that all points of $S - \{u, v\}$ lie outside the circle. For example, the circle centered at the midpoint of the Voronoi edge shared by V_u and V_v is empty in this sense. This property can be used to prove that the edge skeleton of the Delaunay triangulation is a straight-line embedding of a planar graph.

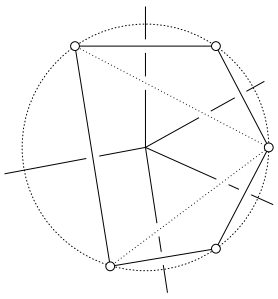

Figure 100: A Voronoi vertex of degree 5 and the corresponding pentagon in the Delaunay triangulation. The dotted edges complete the triangulation by decomposing the pentagon into three triangles.

Now suppose there is a vertex with degree $j > 3$. It corresponds to a polygon with $j > 3$ edges in the Delaunay triangulation, as illustrated in Figure 100. Strictly speaking, the Delaunay triangulation is no longer a triangulation but we can complete it to a triangulation by decomposing each j-gon into $j - 2$ triangles. This corresponds to perturbing the data points every so slightly such that the degree-j Voronoi vertices are resolved into trees in which $j - 2$ degree-3 vertices are connected by $j - 3$ tiny edges.

Local Delaunayhood. Given a triangulation of a finite point set S, we can test whether or not it is the Delaunay triangulation by testing each edge against the two triangles that share the edge. Suppose the edge uv in the triangulation T is shared by the triangles uvp and uvq. We call uv *locally Delaunay*, or *lD* for short, if q lies on or outside the circle that passes through u, v, p. The condition is symmetric in p and q because the circle that passes through u, v, p intersects the first circle in points u and v. It follows that p lies on or outside the circle of u, v, q iff q lies on or outside the circle of u, v, p. We also call uv locally Delaunay if it bounds the convex hull of S and thus belongs to only one triangle. The local condition on the edges implies a global property.

DELAUNAY LEMMA. If every edge in a triangulation K of S is locally Delaunay then K is the Delaunay triangulation of S.

Although every edge of the Delaunay triangulation is locally Delaunay, the Delaunay Lemma is not trivial. Indeed, K may contain edges that are locally Delaunay but do not belong to the Delaunay triangulation, as shown in Figure 101. We omit the proof of the lemma.

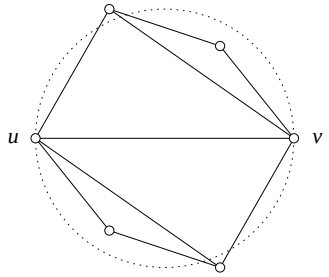

Figure 101: The edge uv is locally Delaunay but does not belong to the Delaunay triangulation.

Edge-flipping. The Delaunay Lemma suggests we construct the Delaunay triangulation by first constructing an arbitrary triangulation of the point set S and then modifying it locally to make all edges lD. The idea is to look for non-lD edges and to flip them, as illustrated in Figure 102. Indeed, if uv is a non-lD edge shared by triangles uvp and

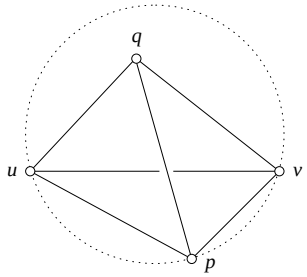

Figure 102: The edge uv is non-lD and can be flipped to the edge pq, which is lD.

uvq then $upvq$ is a convex quadrilateral and *flipping* uv means substituting one diagonal for the other, namely pq

for uv. Note that if uv is non-lD then pq is lD. It is important that the algorithm finds non-lD edges quickly. For this purpose, we use a stack of edges. Initially, we push all edges on the stack and mark them.

```
while stack is non-empty do
  pop edge uv from stack and unmark it;
  if uv is non-lD then
    substitute pq for uv;
    for ab ∈ {up, pv, vq, qu} do
      if ab is unmarked then
        push ab on the stack and mark it
      endif
    endfor
  endif
endwhile.
```

The marks avoid multiple copies of the same edge on the stack. This implies that at any one moment the size of the stack is less than $3n$. Note also that initially the stack contains all non-lD edges and that this property is maintained as an invariant of the algorithm. The Delaunay Lemma implies that when the algorithm halts, which is when the stack is empty, then the triangulation is the Delaunay triangulation. However, it is not yet clear that the algorithm terminates. Indeed, the stack can grow and shrink during the course of the algorithm, which makes it difficult to prove that it ever runs empty.

In-circle test. Before studying the termination of the algorithm, we look into the question of distinguishing lD from non-lD edges. As before we assume that the edge uv is shared by the triangles uvp and uvq in the current triangulation. Recall that uv is lD iff q lies outside the circle that passes through u, v, p. Let $f : \mathbb{R}^2 \to \mathbb{R}$ be defined by $f(x) = x_1^2 + x_2^2$. As illustrated in Figure 103, the graph of this function is a paraboloid in three-dimensional space and we write $x^+ = (x_1, x_2, f(x))$ for the vertical projection of the point x onto the paraboloid. Assuming the three points u, v, p do not lie on a common line then the points u^+, v^+, p^+ lie on a non-vertical plane that is the graph of a function $h(x) = \alpha x_1 + \beta x_2 + \gamma$. The projection of the intersection of the paraboloid and the plane back into $\mathbb{R}^2$ is given by

$$\begin{aligned} 0 &= f(x) - h(x) \\ &= x_1^2 + x_2^2 - \alpha x_1 - \beta x_2 - \gamma, \end{aligned}$$

which is the equation of a circle. This circle passes through u, v, p so it is the circle we have to compare q against. We note that q lies inside the circle iff q^+ lies below the plane. The latter test can be based on the sign of the determinant of the 4-by-4 matrix

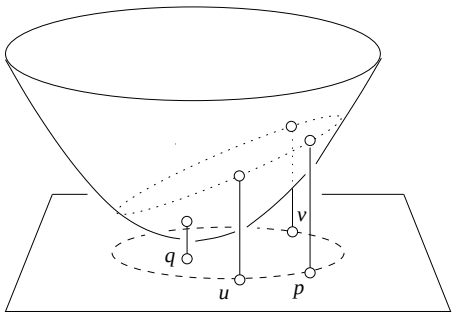

Figure 103: The plane passing through u^+, v^+, p^+ intersects the paraboloid in an ellipse whose projection into $\mathbb{R}^2$ passes through the points u, v, p. The point q^+ lies below the plane iff q lies inside the circle.

$$\Delta = \begin{bmatrix} 1 & u_1 & u_2 & u_1^2 + u_2^2 \\ 1 & v_1 & v_2 & v_1^2 + v_2^2 \\ 1 & p_1 & p_2 & p_1^2 + p_2^2 \\ 1 & q_1 & q_2 & q_1^2 + q_2^2 \end{bmatrix}.$$

Exchanging two rows in the matrix changes the sign. While the in-circle test should be insensitive to the order of the first three points, the sign of the determinant is not. We correct the change using the sign of the determinant of the 3-by-3 matrix that keeps track of the ordering of u, v, p along the circle,

$$\Gamma = \begin{bmatrix} 1 & u_1 & u_2 \\ 1 & v_1 & v_2 \\ 1 & p_1 & p_2 \end{bmatrix}.$$

Now we claim that s is inside the circle of u, v, p iff the two determinants have opposite signs:

```
boolean INCIRCLE(Points u, v, p, q)
  return det Γ · det Δ < 0.
```

We first show that the boolean function is correct for $u = (0, 0)$, $v = (1, 0)$, $p = (0, 1)$, and $q = (0, 0.5)$. The sign of the product of determinants remains unchanged if we continuously move the points and avoid the configurations that make either determinant zero, which are when u, v, p are collinear and when u, v, p, q are cocircular. We can change any configuration where q is inside the circle of u, v, p continuously into the special configuration without going through zero, which implies the correctness of the function for general input points.

Termination and running time. To prove the edge-flip algorithm terminates, we imagine the triangulation lifted to $\mathbb{R}^3$. We do this by projecting the vertices vertically onto the paraboloid, as before, and connecting them with straight edges and triangles in space. Let uv be an edge shared by triangles uvp and uvq that is flipped to pq by the algorithm. It follows the line segments uv and pq cross and their endpoints form a convex quadrilateral, as shown in Figure 104. After lifting the two line segments, we get

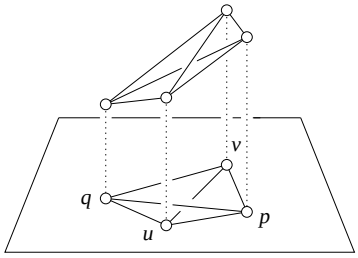

Figure 104: A flip in the plane lifts to a tetrahedron in space in which the lD edge passes below the non-lD edge.

u^+v^+ passing above p^+q^+. We may thus think of the flip as gluing the tetrahedron $u^+v^+p^+q^+$ underneath the surface obtained by lifting the triangulation. The surface is pushed down by each flip and never pushed back up. The removed edge is now above the new surface and can therefore not be reintroduced by a later flip. It follows that the algorithm performs at most $\binom{n}{2}$ flips and thus takes at most time O(n^2) to construct the Delaunay triangulation of S. There are faster algorithms that work in time O($n \log n$) but we prefer the suboptimal method because it is simpler and it reveals more about Delaunay triangulations than the other algorithms.

The lifting of the input points to $\mathbb{R}^3$ leads to an interesting interpretation of the edge-flip algorithm. Starting with a monotone triangulated surface passing through the lifted points, we glue tetrahedra below the surface until we reach the unique convex surface that passes through the points. The projection of this convex surface is the Delaunay triangulation of the points in the plane. This also gives a reinterpretation of the Delaunay Lemma in terms of convex and concave edges of the surface.