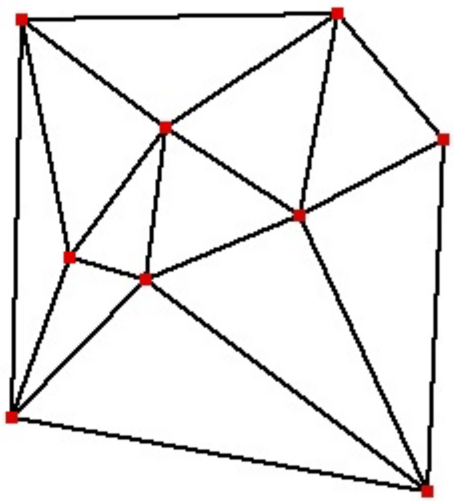
Computational Geometry



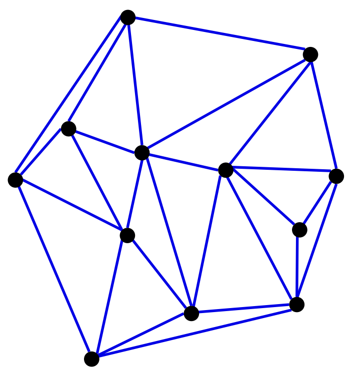
Delaunay Triangulations

Michael Goodrich

with slides from Carola Wenk

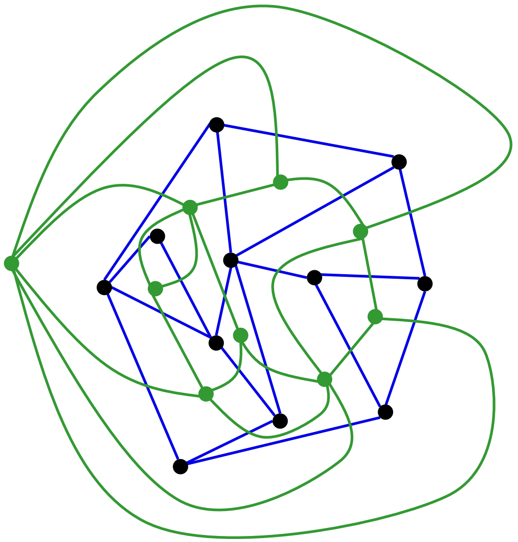
Triangulation

- Let $P = \{p_1, \ldots, p_n\} \subseteq R^2$ be a finite set of points in the plane.
- A **triangulation of *P*** is a simple, plane (i.e., planar embedded), connected graph $T=(P,E)$ such that
  - every edge in E is a line segment,
  - the outer face is bounded by edges of CH(P),
  - all inner faces are triangles.



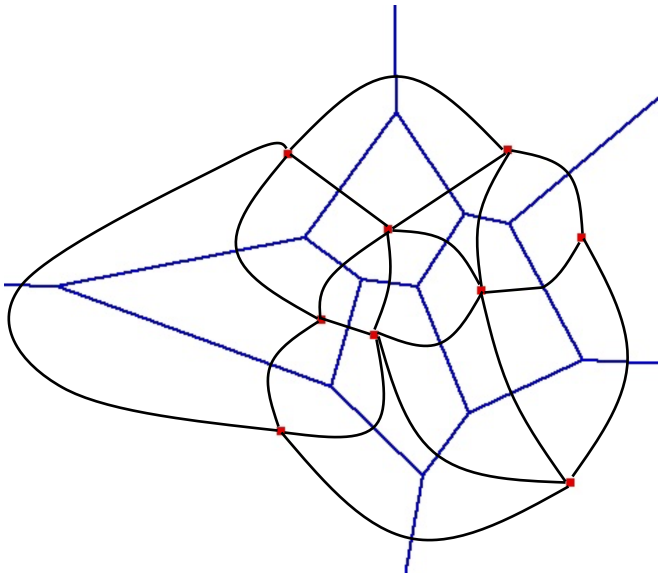
Dual Graph

- Let $G = (V, E)$ be a plane graph. The dual graph G^* has
  - a vertex for every face of G,
  - an edge for every edge of G, between the two faces incident to the original edge

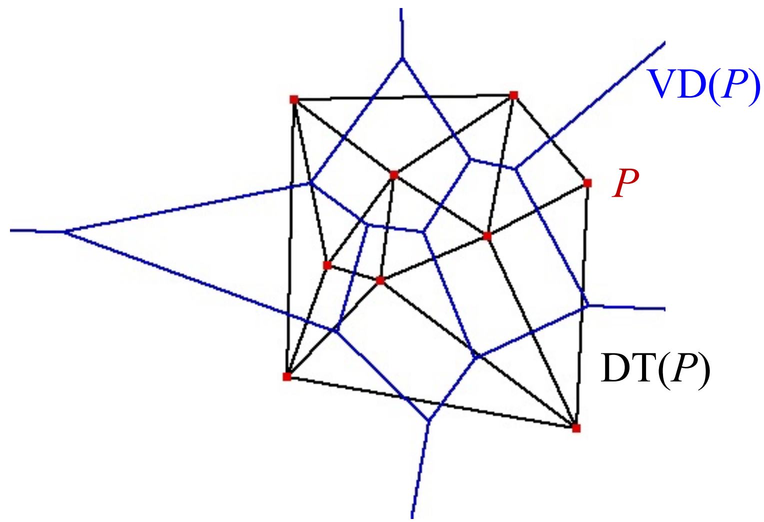


Delaunay Triangulation

- Let *G* be the plane graph for the Voronoi diagram VD(*P*) . Then the dual graph *G** is called the **Delaunay Triangulation DT(*P*)**.



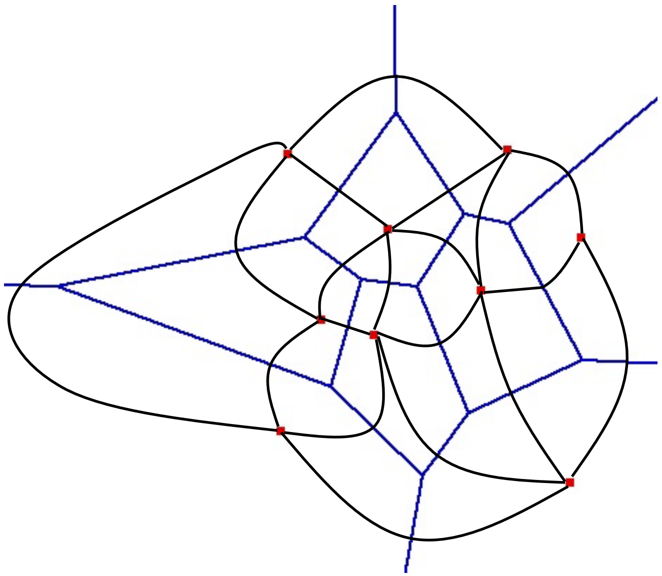
Canonical straight-line embedding for DT(P):



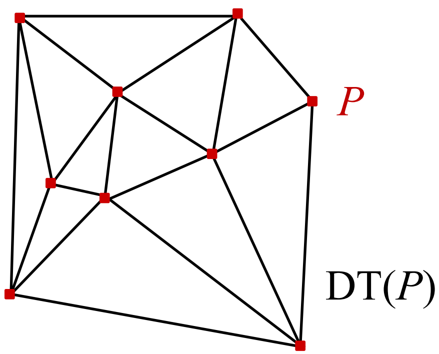
- If *P* is in general position (no three points on a line, no four points on a circle) then every inner face of DT(*P*) is indeed a triangle.
- DT(*P*) can be stored as an abstract graph, without geometric information. (No such obvious storing scheme for VD(*P*).)

Delaunay Triangulation

- Let G be the plane graph for the Voronoi diagram VD(P) . Then the dual graph G^* is called the **Delaunay Triangulation DT(*P*)**.



Canonical straight-line embedding for DT(P):

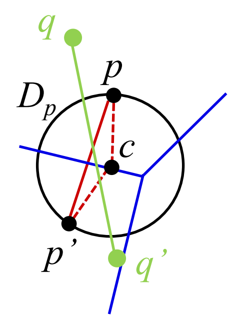


- If P is in general position (no three points on a line, no four points on a circle) then every inner face of DT(P) is indeed a triangle.
- DT(P) can be stored as an abstract graph, without geometric information. (No such obvious storing scheme for VD(P).)

Straight-Line Embedding

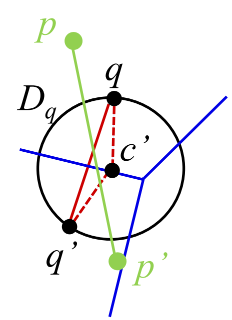
- **Lemma:** DT(P) is a plane graph, i.e., the straight-line edges do not intersect.
- **Proof:**
  - $\overline{pp'}$ is an edge of DT(P) $\Leftrightarrow$ There is an empty closed disk D_p with p and p' on its boundary, and its center c on the bisector.
  - Let $\overline{qq'}$ be another Delaunay edge that intersects $\overline{pp'}$

    $\Rightarrow$ q and q' lie outside of D_p, therefore $\overline{qq'}$ also intersects $\overline{pc}$ or $\overline{p'c}$
  - Symmetrically, $\overline{pp'}$ also intersects $\overline{qc'}$ or $\overline{q'c'}$



$\overline{pc} \subset V(p)$

$\overline{p'c} \subset V(p')$



$\overline{qc'} \subset V(q)$

$\overline{q'c'} \subset V(q')$

$\Rightarrow$ ($\overline{pc}$ or $\overline{p'c'}$) and ($\overline{qc'}$ or $\overline{q'c'}$) intersect

$\Rightarrow$ The edges do not lie in different Voronoi cells.

$\Rightarrow$ Contradiction

$\square$

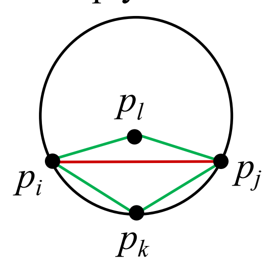
Characterization I of DT(P)

- **Lemma**: Let $p,q,r \in P$ let Δ be the triangle they define. Then the following statements are equivalent:
  a) Δ belongs to DT(P)
  b) The circumcenter c of Δ is a vertex in VD(P)
  c) The circumcircle of Δ is empty (i.e., contains no other point of P)

  Proof sketch: All follow directly from the definition of DT(P) in VD(P). By definition of VD(P), we know that p,q,r are c's nearest neighbors.

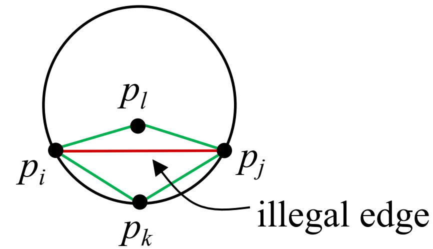
- **Characterization I**: Let T be a triangulation of P.
  Then T=DT(P) $\Leftrightarrow$ The circumcircle of any triangle in T is empty.

non-empty circumcircle

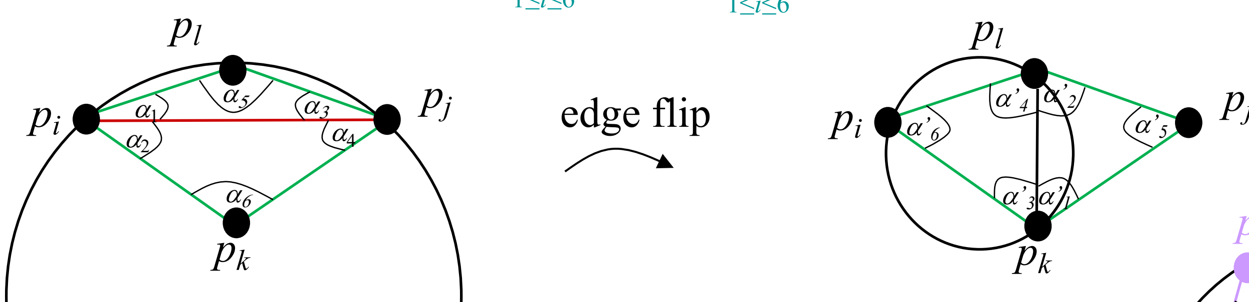


Illegal Edges

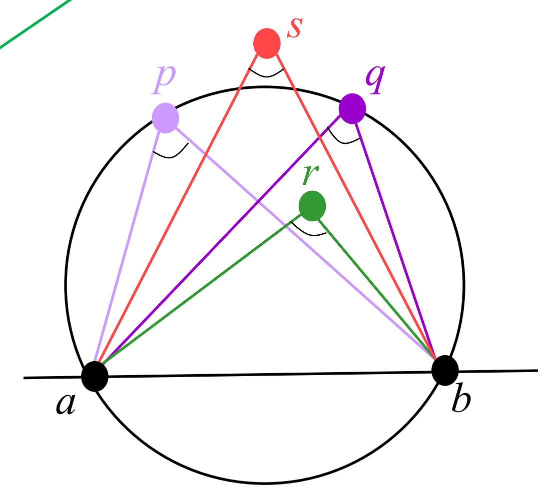
- **Definition:** Let $p_i, p_j, p_k, p_l \in P$.
Then $\overline{p_i p_j}$ is an **illegal edge** $\Leftrightarrow$ p_l lies in the interior of the circle through p_i, p_j, p_k .



- **Lemma:** Let $p_i, p_j, p_k, p_l \in P$.
Then $p_i p_j$ is **illegal** $\Leftrightarrow$ $\min_{1 \le i \le 6} \alpha_i < \min_{1 \le i \le 6} \alpha'_i$



- **Theorem (Thales):** Let a, b, p, q be four points on a circle, and let r be inside and let s be outside of the circle, such that p,q,r,s lie on the same side of the line through a, b.
Then $\angle a,s,b < \angle a,q,b = \angle a,p,b < \angle a,r,b$



So, $\alpha_1 = \angle p_j, p_i, p_l < \angle p_j, p_k, p_l = \alpha'_1$ and $\alpha_3 = \angle p_l, p_j, p_i < \angle p_l, p_k, p_i = \alpha'_3$, etc.

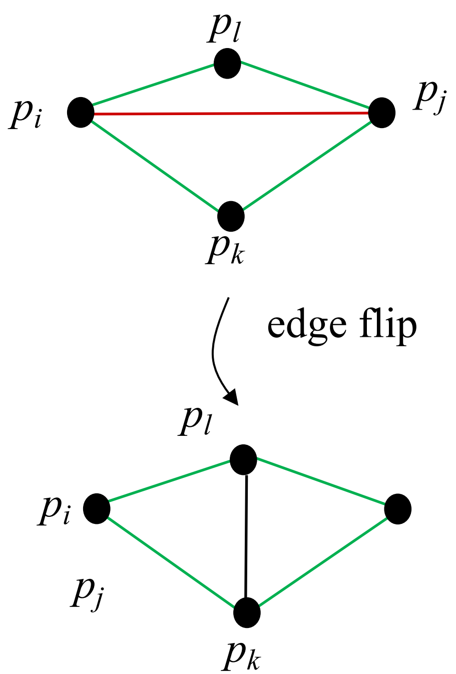
Characterization II of DT(P)

- **Definition:** A triangulation is called legal if it does not contain any illegal edges.
- **Characterization II**: Let T be a triangulation of P. Then T=DT(P) $\Leftrightarrow$ T is legal.
- **Algorithm Legal_Triangulation(T):**

  Input: A triangulation T of a point set P

  Output: A legal triangulation of P

  while T contains an illegal edge $\overline{p_i p_j}$ do
  //Flip $\overline{p_i p_j}$
  Let p_i, p_j, p_k, p_l be the quadrilateral containing $\overline{p_i p_j}$
  Remove $\overline{p_i p_j}$ and add $\overline{p_k p_l}$
  return T



Runtime analysis:

– In every iteration of the loop the angle vector of T (all angles in T sorted by increasing value) increases

– With this one can show that a flipped edge never appears again

– There are $O(n^2)$ edges, therefore the runtime is $O(n^2)$

Characterization III of DT(P)

- **Definition:** Let T be a triangulation of P and let $\alpha_1, \alpha_2, \ldots, \alpha_{3m}$ be the angles of the m triangles in T sorted by increasing value. Then $A(T)=(\alpha_1,\alpha_2,\ldots,\alpha_{3m})$ is called the angle vector of T.
- **Definition:** A triangulation T is called **angle optimal** $\Leftrightarrow A(T) > A(T')$ for any other triangulation T' of the same point set P.

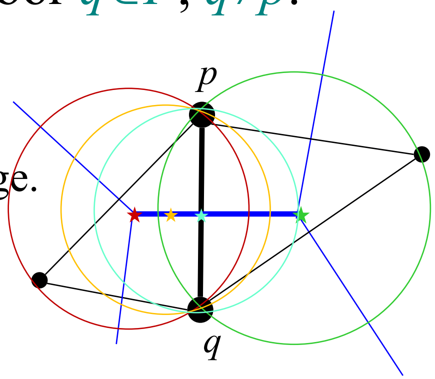
- Let T' be a triangulation that contains an illegal edge, and let T'' be the resulting triangulation after flipping this edge. Then $A(T'') > A(T')$.
- T is angle optimal $\Rightarrow$ T is legal $\Rightarrow$ T=DT(P)

- **Characterization III**: Let T be a triangulation of P.
  Then T=DT(P) $\Leftrightarrow$ T is angle optimal.

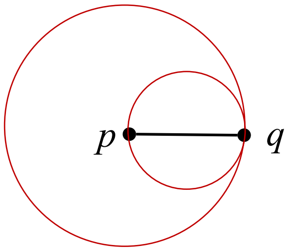
  (If P is not in general position, then any triangulation obtained by triangulating the faces maximizes the minimum angle.)

Applications of DT

- All nearest neighbors: Find for each $p \in P$ its nearest neighbor $q \in P$; $q \neq p$.
  - **Empty circle property:** $p,q \in P$ are connected by an edge in DT(P) $\Leftrightarrow$ there exists an empty circle passing through p and q.
    Proof: "$\Rightarrow$": For the Delaunay edge pq there must be a Voronoi edge. Center a circle through p and q at any point on the Voronoi edge, this circle must be empty.
    "$\Leftarrow$": If there is an empty circle through p and q, then its center c has to lie on the Voronoi edge because it is equidistant to p and q and there is no site closer to c.



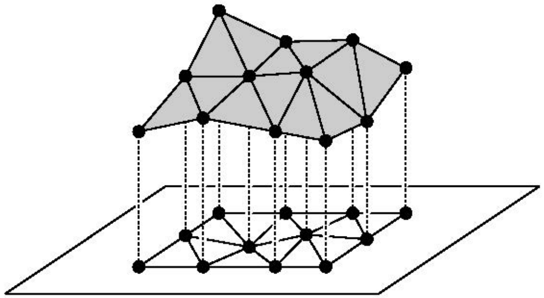
  - **Claim:** In DT(P), every $p \in P$ is adjacent to its nearest neighbors.
    Proof: Let $q \in P$ be a nearest neighbor adjacent to p in DT(P). Then the circle centered at p with q on its boundary has to be empty, so the circle with diameter pq is empty and pq is a Delaunay edge.



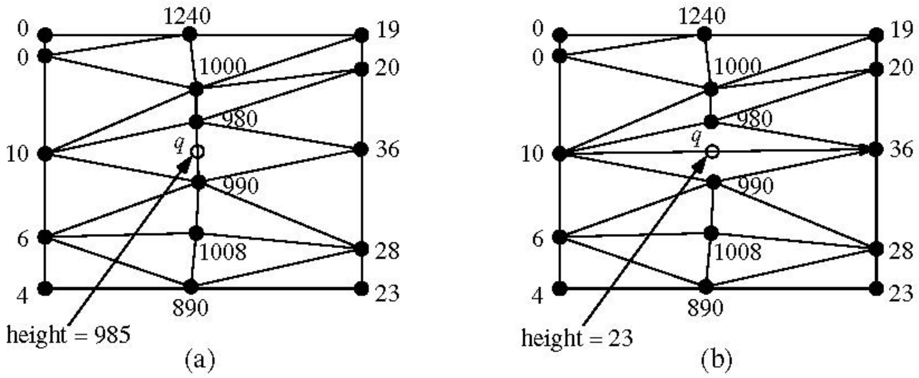
  - **Algorithm:** Find all nearest neighbors in O(n) time: Check for each $p \in P$ all points connected to p with a Delaunay edge.
- Minimum spanning tree: The edges of every Euclidean minimum spanning tree of P are a subset of the edges of DT(P).

Applications of DT

- Terrain modeling:
  - Model a scanned terrain surface by interpolating the height using a piecewise linear function over R^2.

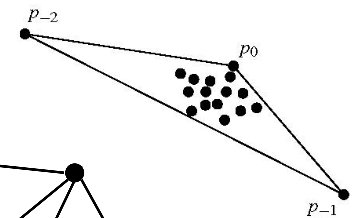


  - Angle-optimal triangulations give better approximations / interpolations since they avoid skinny triangles

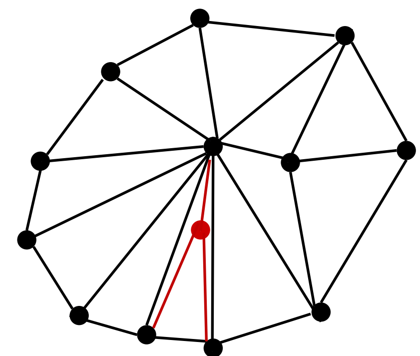


Randomized Incremental Construction of DT(P)

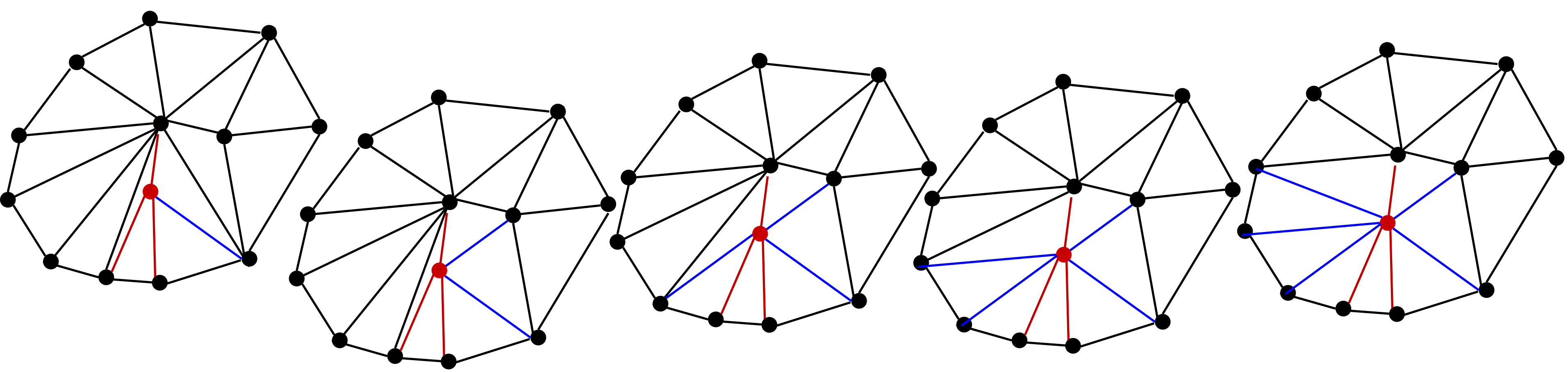
- Start with a large triangle containing *P*.



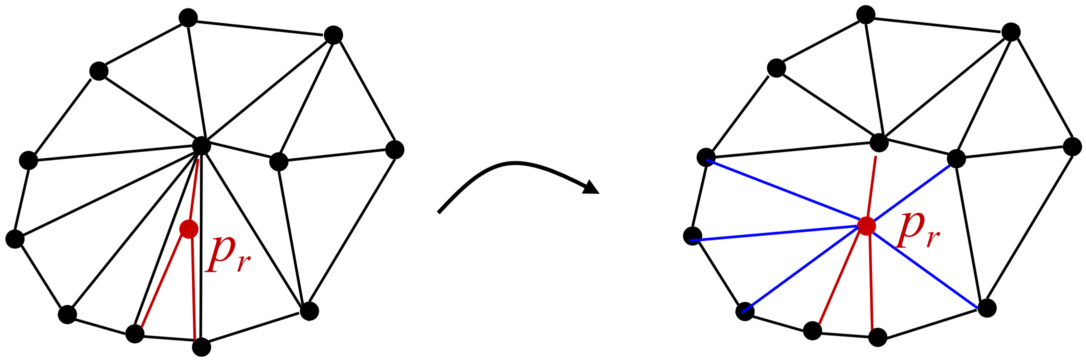
- Insert points of *P* incrementally:
  - Find the containing triangle
  - Add new edges



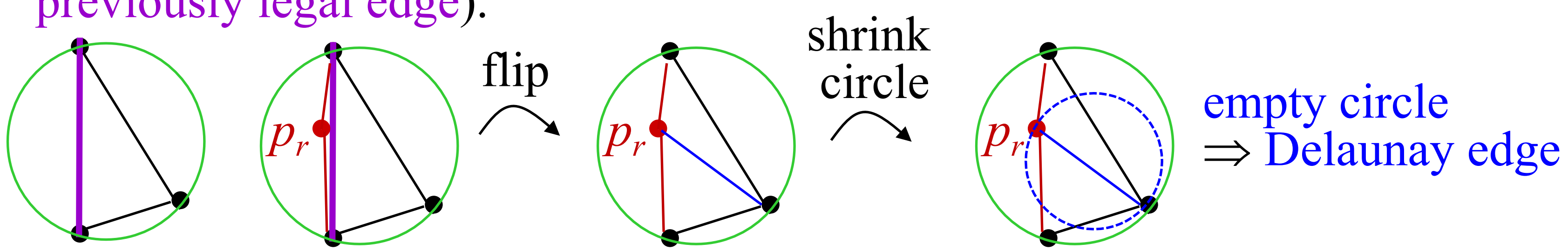
  - Flip all illegal edges until every edge is legal.



Randomized Incremental Construction of DT(P)



- An edge can become illegal only if one of its incident triangles changes.
- Check only edges of new triangles.
- Every new edge created is incident to p_r.
- Every old edge is legal (if p_r is on one of the incident triangles, the edge would have been flipped if it were illegal).
- Every new edge is legal (since it has been created from flipping a previously legal edge).



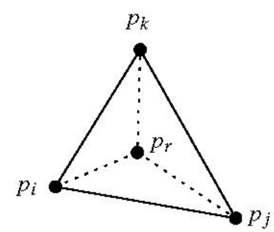
Pseudo Code

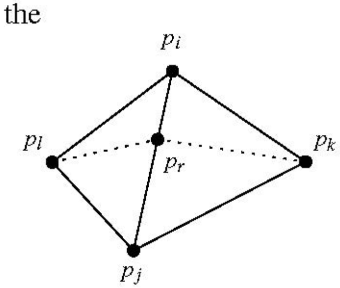
Algorithm DELAUNAYTRIANGULATION(P)
Input. A set P of $n+1$ points in the plane.
Output. A Delaunay triangulation of P.

1. Let p_0 be the lexicographically highest point of P, that is, the rightmost among the points with largest y-coordinate.
2. Let p_{-1} and p_{-2} be two points in $\mathbb{R}^2$ sufficiently far away and such that P is contained in the triangle $p_0p_{-1}p_{-2}$.
3. Initialize $\mathcal{T}$ as the triangulation consisting of the single triangle $p_0p_{-1}p_{-2}$.
4. Compute a random permutation $p_1, p_2, \ldots, p_n$ of $P \setminus \{p_0\}$.
5. **for** $r \leftarrow 1$ **to** n
6. **do** ($*$ Insert p_r into $\mathcal{T}$: $*$)
7. Find a triangle $p_ip_jp_k \in \mathcal{T}$ containing p_r.
8. **if** p_r lies in the interior of the triangle $p_ip_jp_k$
9. **then** Add edges from p_r to the three vertices of $p_ip_jp_k$, thereby splitting $p_ip_jp_k$ into three triangles.
10. LEGALIZEEDGE($p_r, \overline{p_ip_j}, \mathcal{T}$)
11. LEGALIZEEDGE($p_r, \overline{p_jp_k}, \mathcal{T}$)
12. LEGALIZEEDGE($p_r, \overline{p_kp_i}, \mathcal{T}$)
13. **else** ($*$ p_r lies on an edge of $p_ip_jp_k$, say the edge $\overline{p_ip_j}$ $*$)
14. Add edges from p_r to p_k and to the third vertex p_l of the other triangle that is incident to $\overline{p_ip_j}$, thereby splitting the two triangles incident to $\overline{p_ip_j}$ into four triangles.
15. LEGALIZEEDGE($p_r, \overline{p_ip_l}, \mathcal{T}$)
16. LEGALIZEEDGE($p_r, \overline{p_lp_j}, \mathcal{T}$)
17. LEGALIZEEDGE($p_r, \overline{p_jp_k}, \mathcal{T}$)
18. LEGALIZEEDGE($p_r, \overline{p_kp_i}, \mathcal{T}$)
19. Discard p_{-1} and p_{-2} with all their incident edges from $\mathcal{T}$.
20. **return** $\mathcal{T}$

LEGALIZEEDGE($p_r, \overline{p_ip_j}, \mathcal{T}$)

1. ($*$ The point being inserted is p_r, and $\overline{p_ip_j}$ is the edge of $\mathcal{T}$ that may need to be flipped. $*$)
2. **if** $\overline{p_ip_j}$ is illegal
3. **then** Let $p_ip_jp_k$ be the triangle adjacent to $p_rp_ip_j$ along $\overline{p_ip_j}$.
4. ($*$ Flip $\overline{p_ip_j}$: $*$) Replace $\overline{p_ip_j}$ with $\overline{p_rp_k}$.
5. LEGALIZEEDGE($p_r, \overline{p_ip_k}, \mathcal{T}$)
6. LEGALIZEEDGE($p_r, \overline{p_kp_j}, \mathcal{T}$)

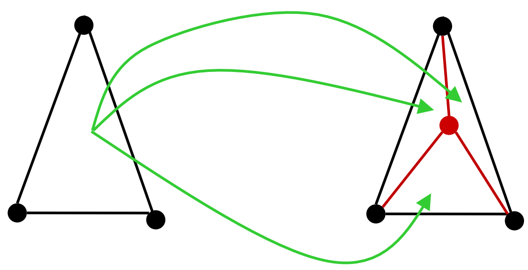




History

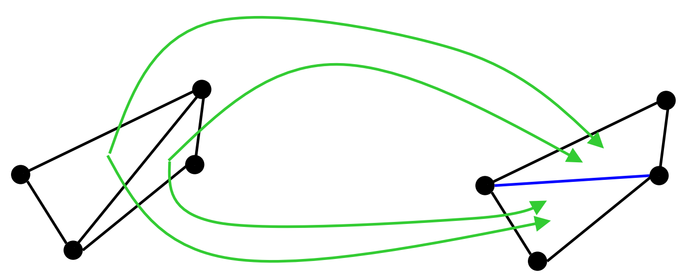
The algorithm stores the history of the constructed triangles. This allows to easily locate the triangle containing a new point by following pointers.

- Division of a triangle:



Store pointers from the old triangle to the three new triangles.

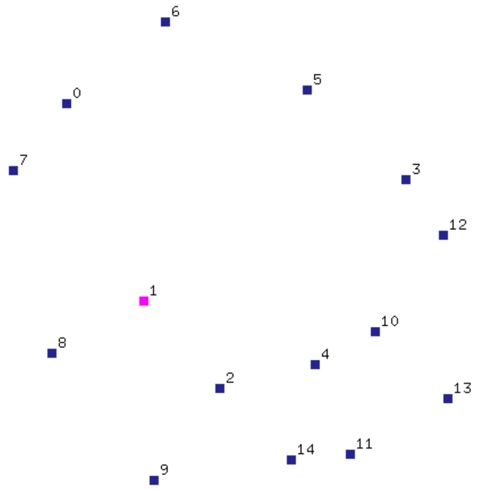
- Flip:

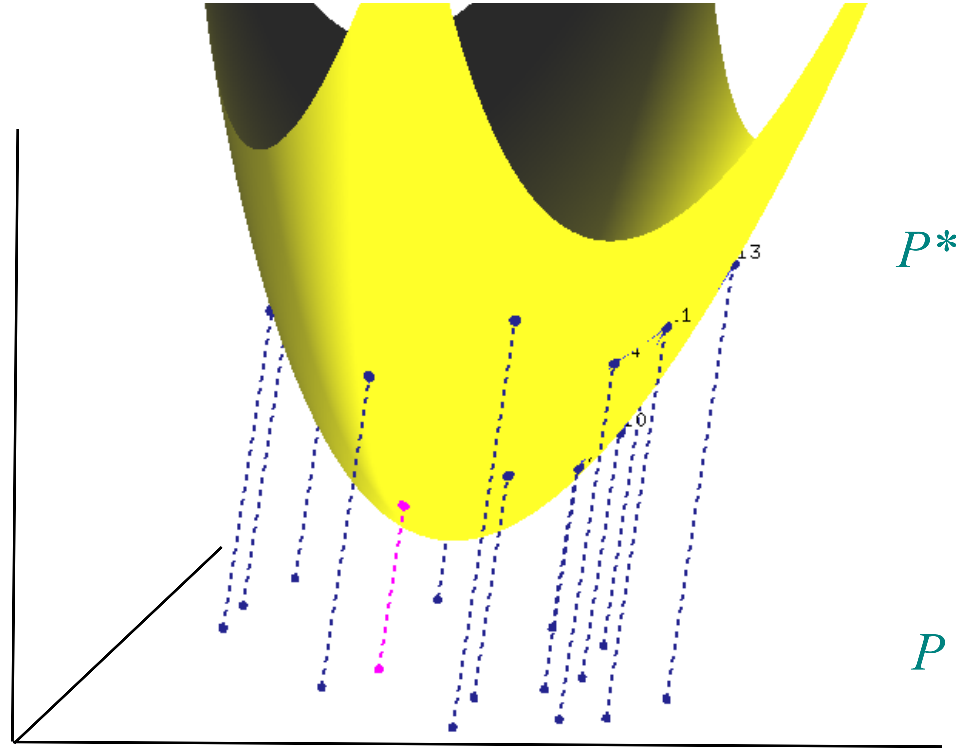


Store pointers from both old triangles to both new triangles.

DT and 3D CH

Theorem: Let $P=\{p_1,\ldots,p_n\}$ with $p_i=(a_i, b_i,0)$. Let $p^*_i=(a_i, b_i, a^2_i+ b^2_i)$ be the vertical projection of each point p_i onto the paraboloid $z=x^2+ y^2$. Then DT(P) is the orthogonal projection onto the plane $z=0$ of the lower convex hull of $P^*=\{p^*_1,\ldots,p^*_n\}$.

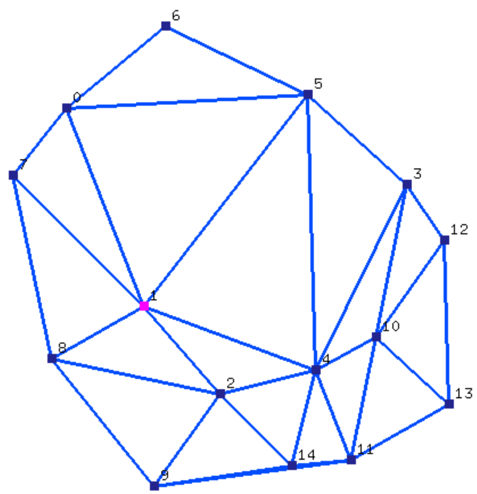


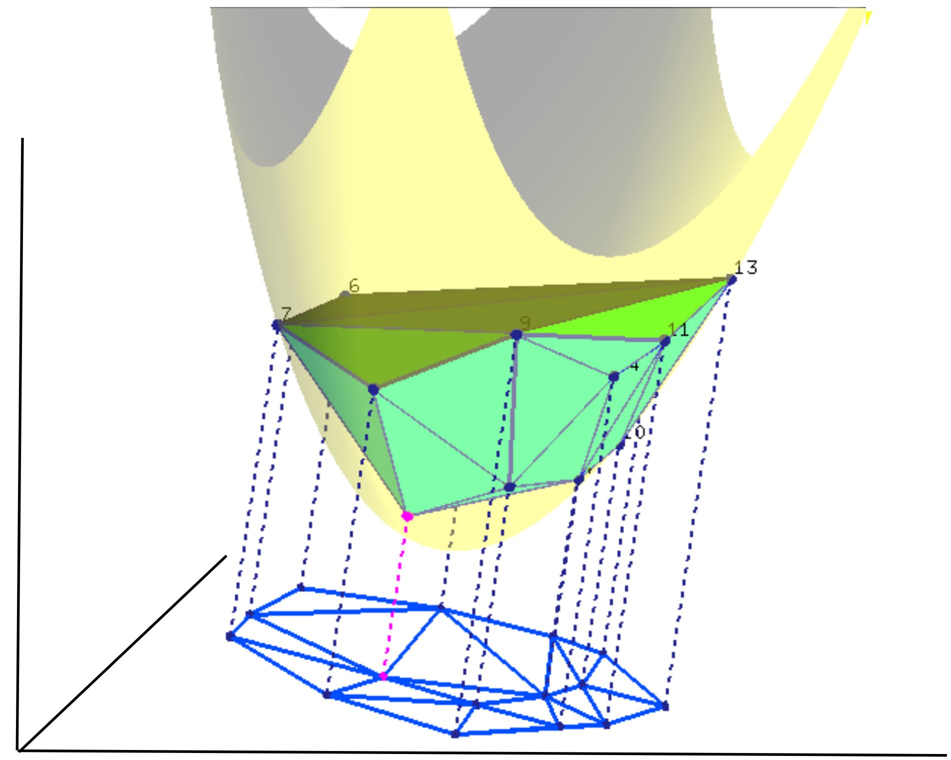


Pictures generated with Hull2VD tool available at http://www.cs.mtu.edu/~shene/NSF-2/DM2-BETA

DT and 3D CH

Theorem: Let $P=\{p_1,\ldots,p_n\}$ with $p_i=(a_i, b_i,0)$. Let $p'_i=(a_i, b_i, a^2_i+b^2_i)$ be the vertical projection of each point p_i onto the paraboloid $z=x^2+y^2$. Then $\mathrm{DT}(P)$ is the orthogonal projection onto the plane $z=0$ of the lower convex hull of $P'=\{p'_1,\ldots,p'_n\}$.

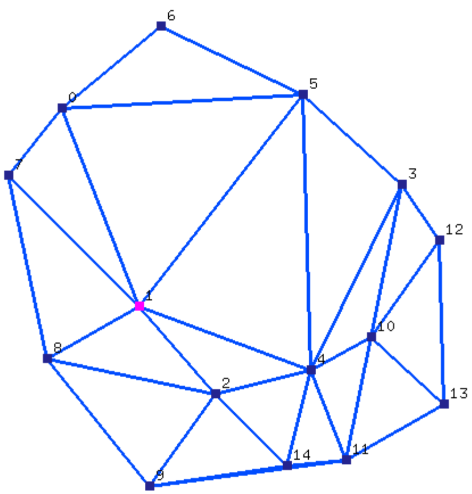


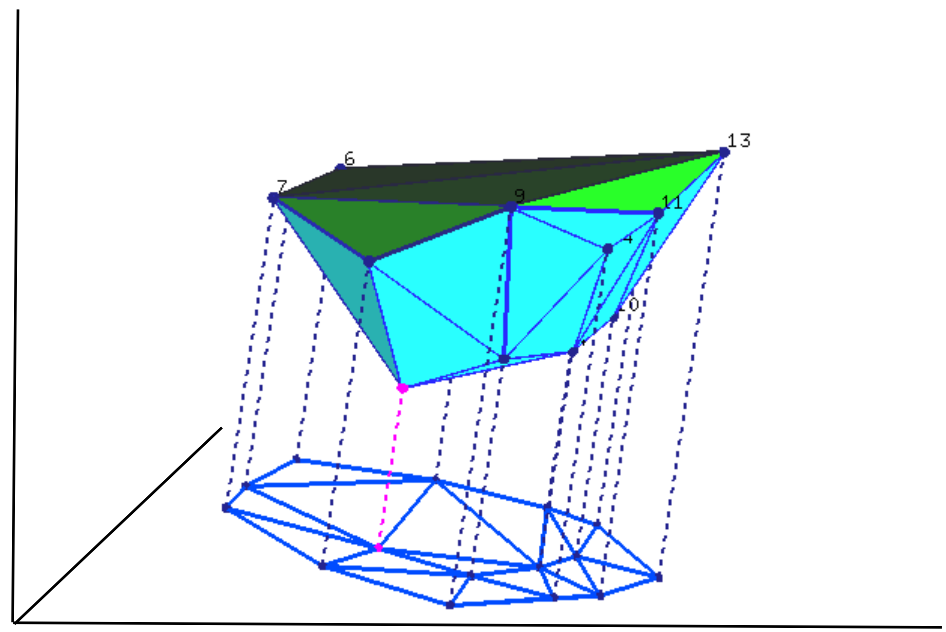


Pictures generated with Hull2VD tool available at http://www.cs.mtu.edu/~shene/NSF-2/DM2-BETA

DT and 3D CH

Theorem: Let $P=\{p_1,\ldots,p_n\}$ with $p_i=(a_i, b_i, 0)$. Let $p'_i=(a_i, b_i, a^2_i+b^2_i)$ be the vertical projection of each point p_i onto the paraboloid $z=x^2+y^2$. Then DT(P) is the orthogonal projection onto the plane $z=0$ of the lower convex hull of $P'=\{p'_1,\ldots,p'_n\}$.





Pictures generated with Hull2VD tool available at http://www.cs.mtu.edu/~shene/NSF-2/DM2-BETA

DT and 3D CH

Theorem: Let $P=\{p_1,\dots,p_n\}$ with $p_i=(a_i, b_i, 0)$. Let $p'_i=(a_i, b_i, a^2_i + b^2_i)$ be the vertical projection of each point p_i onto the paraboloid $z=x^2+y^2$. Then DT(P) is the orthogonal projection onto the plane $z=0$ of the lower convex hull of $P'=\{p'_1,\dots,p'_n\}$.

p'_i, p'_j, p'_k form a (triangular) face of LCH(P')

$\Leftrightarrow$

The plane through p'_i, p'_j, p'_k leaves all remaining points of P above it

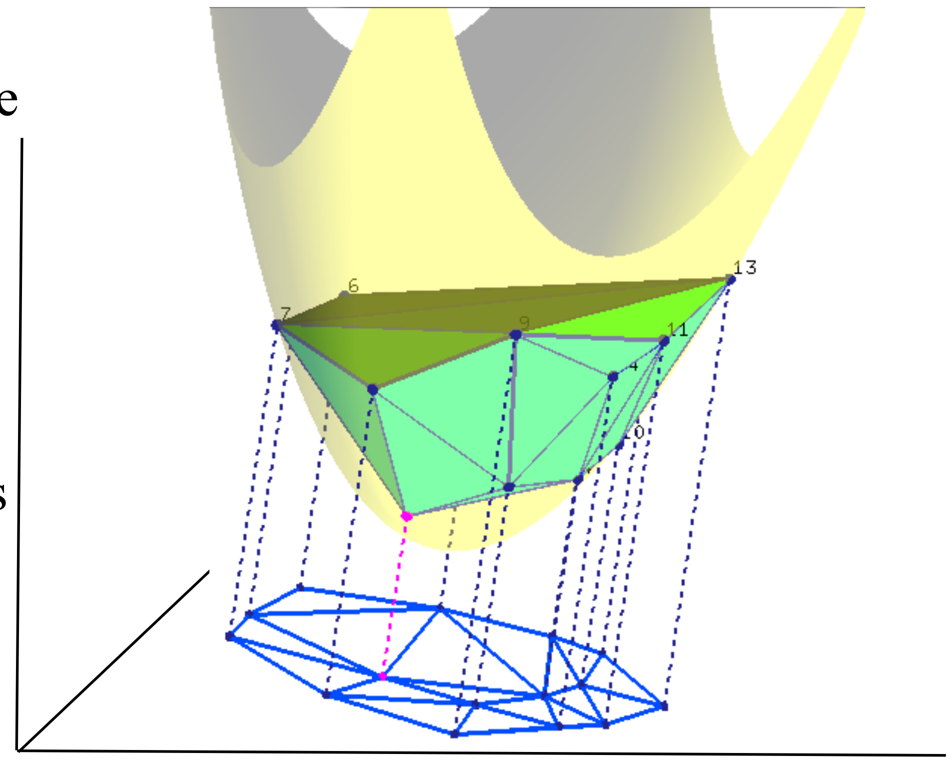
property of unit paraboloid

$\Leftrightarrow$

The circle through p_i, p_j, p_k leaves all remaining points of P in its exterior

$\Leftrightarrow$

p_i, p_j, p_k form a triangle of DT(P)



Slide adapted from slides by Vera Sacristan.