

# Delaunay Triangulations

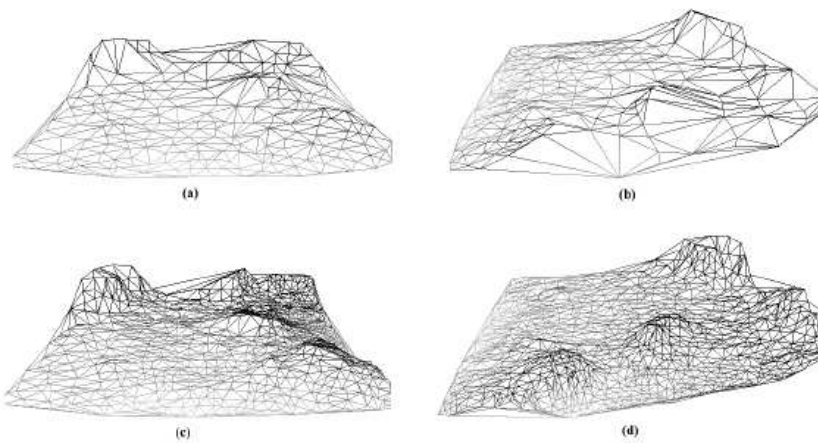
(Triangulations of a set of points.)

And other proximity graphs

---

Delaunay Triangulations - Motivational Example

Terrain Modelling

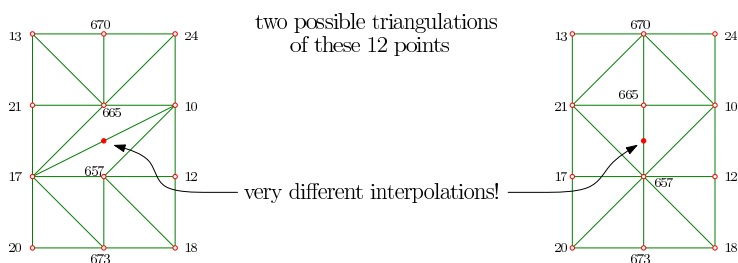
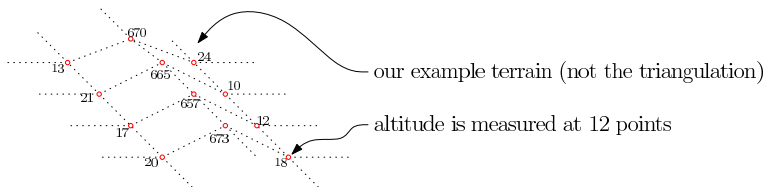


Intuition: edges are linear interpolations between vertices.

---

Delaunay Triangulations - Motivational Example

Terrain Modelling



Intuition: edges are linear interpolations between vertices.

---

What is a good triangulation?

- minimum length?
- minimum number of edges?
- avoiding skinny triangles?
- avoiding obtuse angles?
- ...

## Definition and Basic Properties of Triangulations

**Definition:**

Let  $P = \{p_1, \dots, p_n\}$  be a set of  $n$  points in the plane. A maximal planar subdivision is a subdivision such that no edge can be added between two vertices without destroying its planarity. A *triangulation* of  $P$  is a maximal planar subdivision whose vertex set is  $P$ .

---

- every face (except the unbounded face) must be a triangle
- the convex hull edges belong to the triangulation
- number of triangles:

Let  $k$  be the number of points on the convex hull of  $P$ .

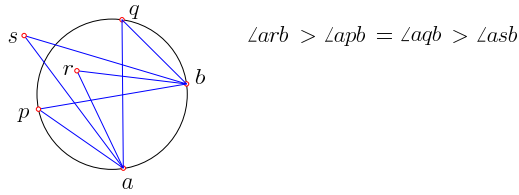
**Theorem:**

Any triangulation of  $P$  has  $2 \cdot n - 2 - k$  triangles and  $3 \cdot n - 3 - k$  edges.

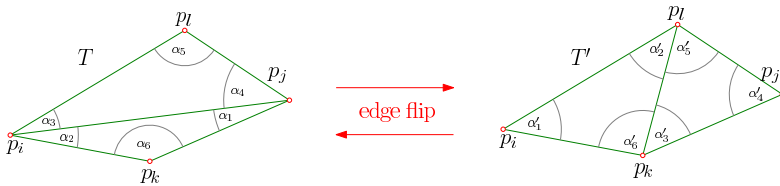
## Triangulations that Maximise the Smallest Angle

- let  $m$  be the number of triangles in a triangulation  $T$  of  $P$
  - we have  $3 \cdot m$  interior angles  $\alpha_i$
  - sort all angles such that  $\alpha_1 \leq \alpha_2 \leq \alpha_3 \leq \dots \leq \alpha_{3m}$
  - the *angle vector* of  $T$  is the vector  $A(T) = (\alpha_1, \dots, \alpha_{3m})$
  - for two triangulations  $T$  and  $T'$ , we define:  
$$A(T) > A(T') \text{ iff } \exists i, 1 \leq i \leq 3m : \alpha_i > \alpha'_i \wedge (\forall j < i : \alpha_j = \alpha'_j)$$
  - if for all  $T'$ ,  $A(T) \geq A(T')$ , then  $T$  is called *angle-optimal*.
-

**Theorem:** (Thales)



- consider an edge  $p_i p_j$  of a triangulation  $T$  (not on the convex hull)
- if  $p_i, p_j, p_k$  and  $p_l$  form a convex quadrilateral then we can perform an edge flip:



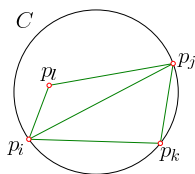
- the only difference are the six angles
- the edge  $p_i p_j$  is *illegal* if:  $\min_{1 \leq i \leq 6} \alpha_i < \min_{1 \leq i \leq 6} \alpha'_i$

**Observation:**

Let  $T$  be a triangulation with an illegal edge  $e$  and  $T'$  be obtained from  $T$  by flipping  $e$ . Then:

$$A(T') > A(T)$$

**Observation:**



let  $p_i p_j$  be an edge  
 let  $C$  be circle through  $p_i, p_j$  and  $p_k$   
 $p_i p_j$  is illegal  $\iff p_l$  lies in the interior of  $C$   
 (this is symmetric for  $p_l$  vs.  $p_k$ )

**Definition:**

A *legal triangulation* is one without illegal edges.

**Algorithm: LegalTriangulation( $T$ )**

- Input: a triangulation  $T$  of  $P$
  - Output: a legal triangulation of  $P$
- ```

1  while  $T$  contains an illegal edge  $e$  do
2      flip  $e$  in  $T$ 
3  return  $T$ 
    
```

According to our observations,  $A(T)$  increases with every edge flip.

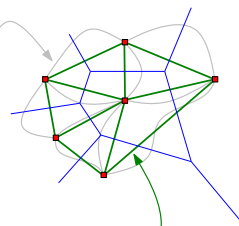
Since there are finitely many triangulations, the algorithm must terminate.

## Delaunay Triangulation

Recall the Voronoi Diagram  $Vor(P)$  of  $P$ .

The dual of  $Vor(P)$  is a graph  $G$ :

- $G$  has a node for every Voronoi cell/site
- $G$  has an arc between two nodes  
if the corresponding Voronoi cells share an edge
- number of edges in  $Vor(P)$  = number of arcs in  $G$



**Definition:**

The *Delaunay Graph*  $DG(P)$  of  $P$  is the straight-line embedding of  $G$ , such that nodes correspond to Voronoi cell sites and arcs connecting nodes are straight line segments.

**Theorem:**

$DG(P)$  is planar.

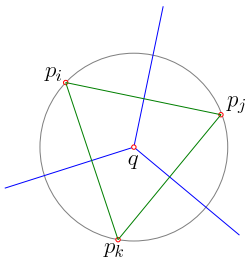
If  $P$  is in general position (i.e. no four points are on a circle), then  $DG(P)$  is a unique triangulation, called  $DT(P)$  – Delaunay Triangulation.

If  $P$  is not in general position, we can add edges to create a Delaunay Triangulation  $DT(P)$ .

- three points  $p_i, p_j$  and  $p_k$  are vertices of the same face of  $DG(P)$

$\iff$

the circle through  $p_i, p_j$  and  $p_k$  contains no point of  $P$  in its interior



**Recall:**

point  $q$  is a vertex of  $Vor(P)$

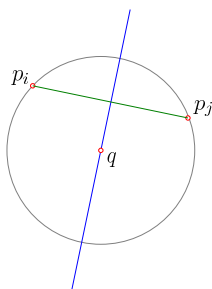
$\iff$

its largest empty circle contains at least three points on its boundary

- $p_i p_j$  is an edge of  $DG(P)$

$\iff$

there exists a closed disc containing  $p_i$  and  $p_j$  on its boundary and no other point of  $P$



**Recall:**

the bisector of  $p_i$  and  $p_j$  defines an edge of  $Vor(P)$

$\iff$

there exists a point  $q$  on the bisector, such that  $C_P(q)$  contains only  $p_i$  and  $p_j$

**Theorem:**

A triangulation  $T$  is a Delaunay Triangulation

$\iff$

the circumcircle of any triangle of  $T$  does not contain points in its interior.

**Theorem:**

A triangulation  $T$  is a Delaunay Triangulation

$\iff$

$T$  is a legal triangulation.

Any angle optimal triangulation is legal.

Hence, any angle optimal triangulation is a Delaunay Triangulation.

Any  $DT(P)$  maximises the minimum angle over all triangulations of  $P$ .

## Computing Delaunay Triangulations

**Recall:**

$DG(P)$  is the dual of  $Vor(P)$ .

$Vor(P)$  can be computed in  $O(n \log n)$  time.

$DG(P)$  and  $DT(P)$  can be computed from  $Vor(P)$  in  $O(n)$  time,  
hence  $O(n \log n)$  time in total.

## Other Proximity Graphs

(Proximity Graphs of a Set of Points.)

### Minimum Spanning Tree $MST(P)$ :

First idea:

- construct complete graph  $O(n^2)$  edges
- sort edges  $O(n^2 \log n)$  time
- process edges  $O(\log n)$  time per edge

New idea:

- construct  $DT(P)$   $O(n)$  edges  $O(n \log n)$  time
- sort edges  $O(n \log n)$  time
- process edges  $O(\log n)$  time per edge

**Theorem:**

$$MST(P) \subseteq DT(P)$$

### Mutual Nearest Neighbour $MNN(P)$ :

- edge  $pq$  iff  $p$  is nearest neighbour of  $q$  and  $q$  is nearest neighbour of  $p$
- $\leq \lfloor \frac{n}{2} \rfloor$  edges
- not always connected

### Nearest Neighbour $NN(P)$ :

- edge  $pq$  iff  $p$  is nearest neighbour of  $q$  or  $q$  is nearest neighbour of  $p$
  - $\leq n - 1$  edges
  - not always connected
-

**Theorem:**

$$MNN(P) \subseteq NN(P)$$

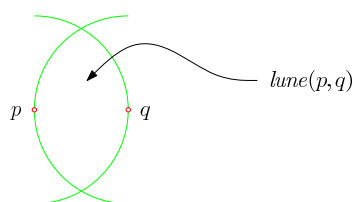
**Theorem:**

$$NN(P) \subseteq MST(P)$$

Recall: Prim's algorithm:

if  $q$  is nearest neighbour of  $p$ , then  $pq \in MST(P)$

**Relative Neighbourhood Graph  $RN(P)$ :**



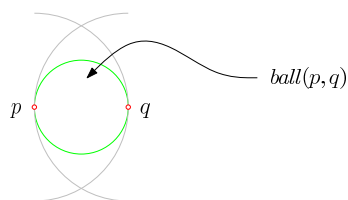
$$\begin{aligned} \text{edge } pq \text{ in } RN(P) &\iff \text{lune}(p, q) = \emptyset \\ &\iff |pq| \leq \min_{r \in P, r \neq p, q} \{\max(|pr|, |rq|)\} \end{aligned}$$

**Theorem:**

$$MST(P) \subseteq RN(P)$$

Note:  $pq \in MST(P) \implies \text{lune}(p, q) = \emptyset$

**Gabriel Graph  $GG(P)$ :**



$$\begin{aligned} \text{edge } pq \text{ in } GG(P) &\iff \text{the circle with diameter } |pq| \\ &\quad \text{and with } p \text{ and } q \text{ on its boundary is empty} \\ &\iff \text{ball}(p, q) = \emptyset \end{aligned}$$

**Theorem:**

$$RN(P) \subseteq GG(P)$$

Note:  $\text{ball}(p, q) \subseteq \text{lune}(p, q)$

**Delaunay Graph  $DG(P)$ :**

edge  $pq$  in  $DG(P)$   $\iff$  there exists an empty disc,  
 e.g.  $ball(p, q)$ ,  
 with  $p$  and  $q$  on its boundary

**Theorem:**

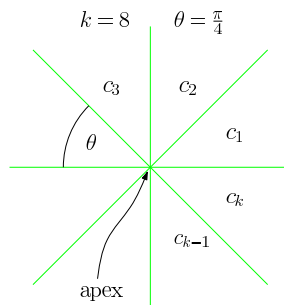
$$GG(P) \subseteq DG(P)$$

**Summary:**

$$MNN(P) \subseteq NN(P) \subseteq MST(P) \subseteq RN(P) \subseteq GG(P) \subseteq DG(P)$$

**Yao Graph  $YG_k(P)$ :**

- let  $k \geq 2$  be an integer
- let  $\theta = \frac{2\pi}{k}$
- let  $C = \{c_1, c_2, \dots, c_k\}$  be a set of cones with the same apex and with apex angle  $\theta$

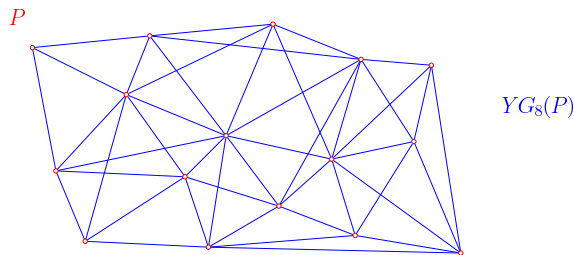


The Yao Graph  $YG_k(P)$  of a set  $P$  of points is obtained in the following way:

- the vertices of  $YG_k(P)$  are the points in  $P$
- for each point  $p \in P$ , we translate the cones in  $C$ , such that their apex is at  $p$ , and for each cone  $c_i \in C$ , we add an edge from  $p$  to the closest point in  $c_i \cap P \setminus p$

**Yao Graph  $YG_k(P)$ :**

$$k = 8 \quad \theta = \frac{\pi}{4}$$



- the vertices of  $YG_k(P)$  are the points in  $P$
- for each point  $p \in P$ , we translate the cones in  $C$ , such that their apex is at  $p$ , and for each cone  $c_i \in C$ , we add an edge from  $p$  to the closest point in  $c_i \cap P \setminus p$

**Yao Graph  $YG_k(P)$ :**

**Theorem:**

$RN(P) \subseteq YG_k(P)$  for  $k \geq 6$ .

Note: for  $k \geq 6$  we have  $\theta \leq 60^\circ$

**Theorem:**

For all  $k$ , there exist point sets, such that  $YG_k(P) \not\subseteq GG(P)$ .

**$t$ -spanner:**

- let  $P$  be a set of points
- let  $t \geq 1$  be a real number
- let  $G$  be a graph with vertex set  $P$
- $G$  is a  $t$ -spanner if for any two points  $p, q \in P$  the length of the shortest path in  $G$  from  $p$  to  $q$  is at most  $t \cdot |p, q|$ .

$t = 5$

