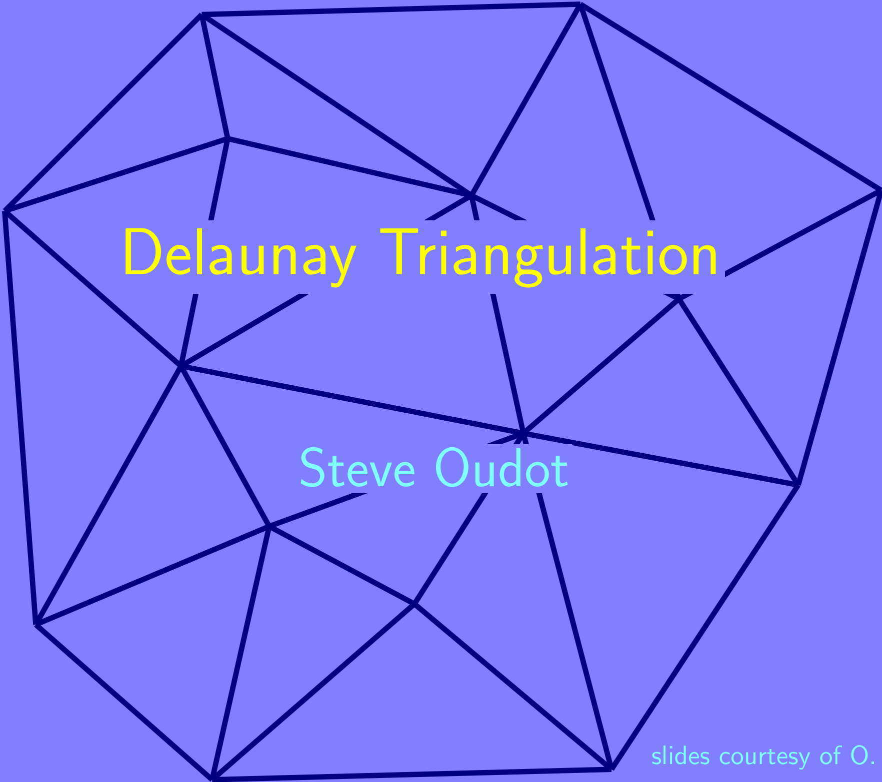


Delaunay Triangulation

Steve Oudot

slides courtesy of O. Devillers

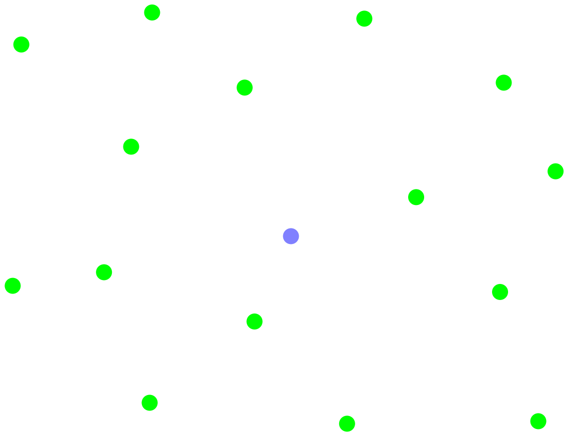
Outline

1. Definition and Examples
2. Applications
3. Basic properties
4. Construction

Definition

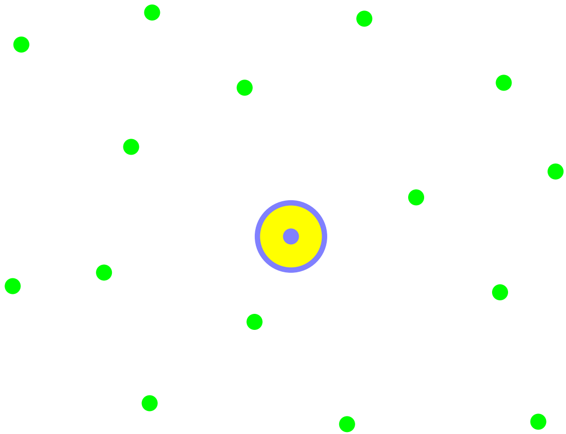
Classical example

looking for nearest neighbor



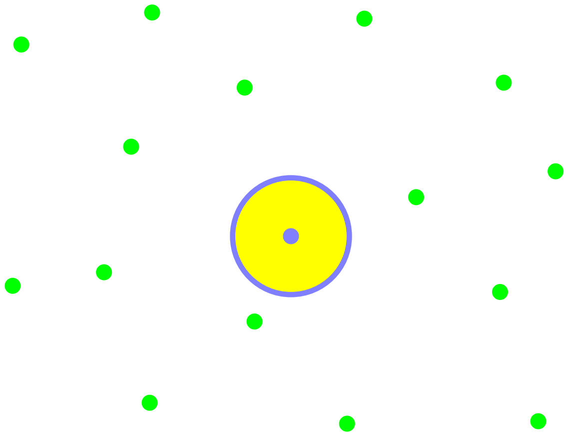
Classical example

looking for nearest neighbor



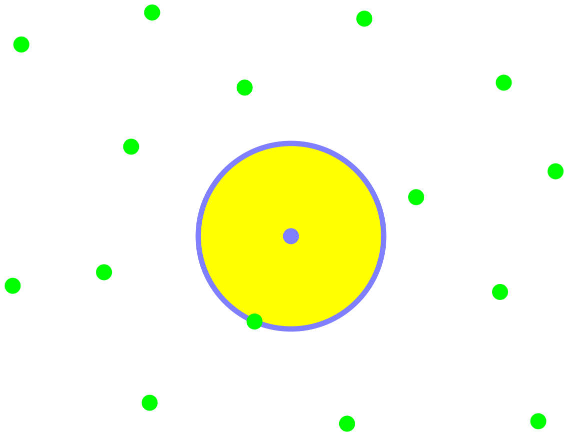
Classical example

looking for nearest neighbor



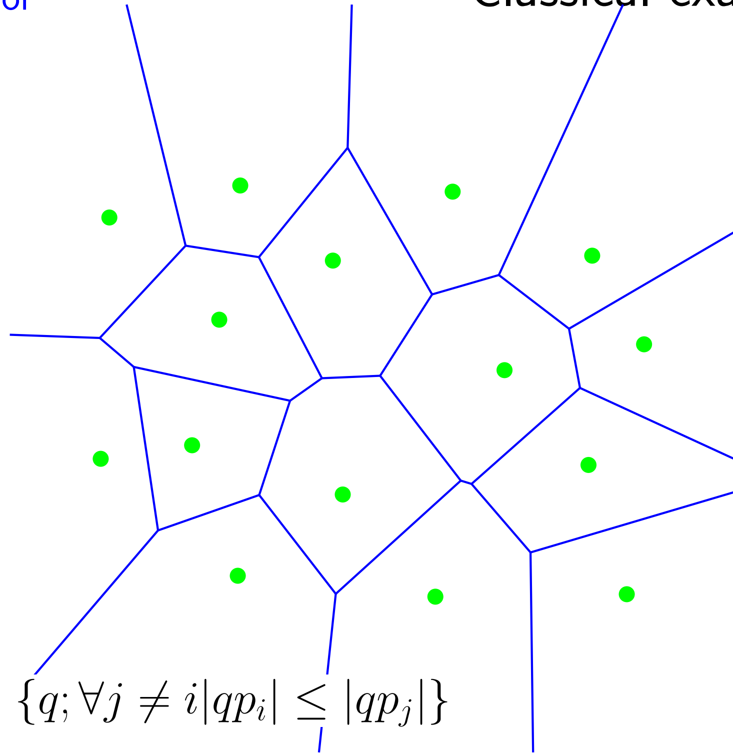
Classical example

looking for nearest neighbor



Voronoi

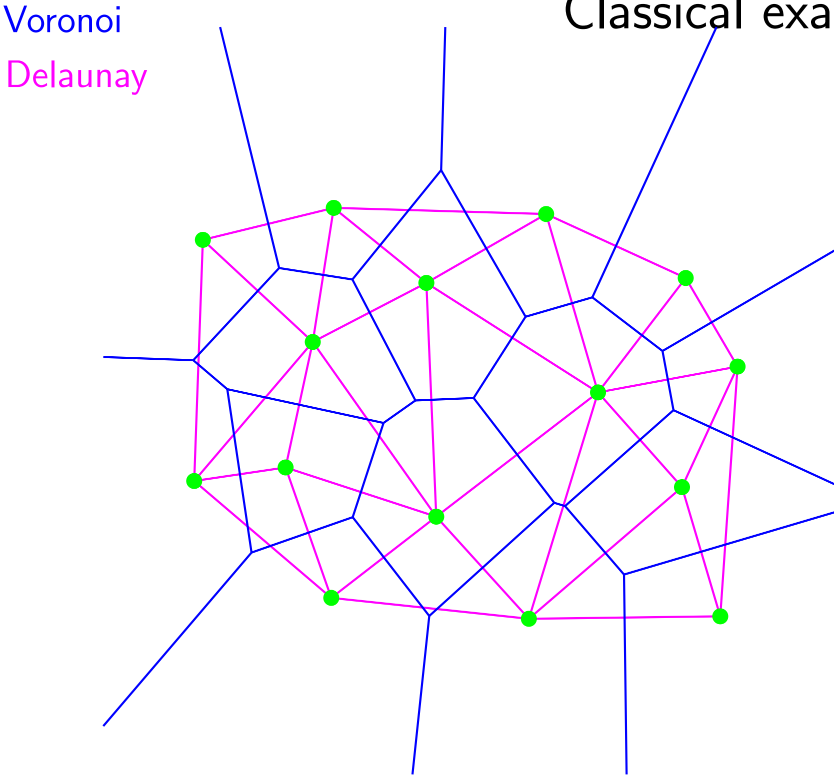
Classical example



Georgy F. Voronoi
(1868–1908)

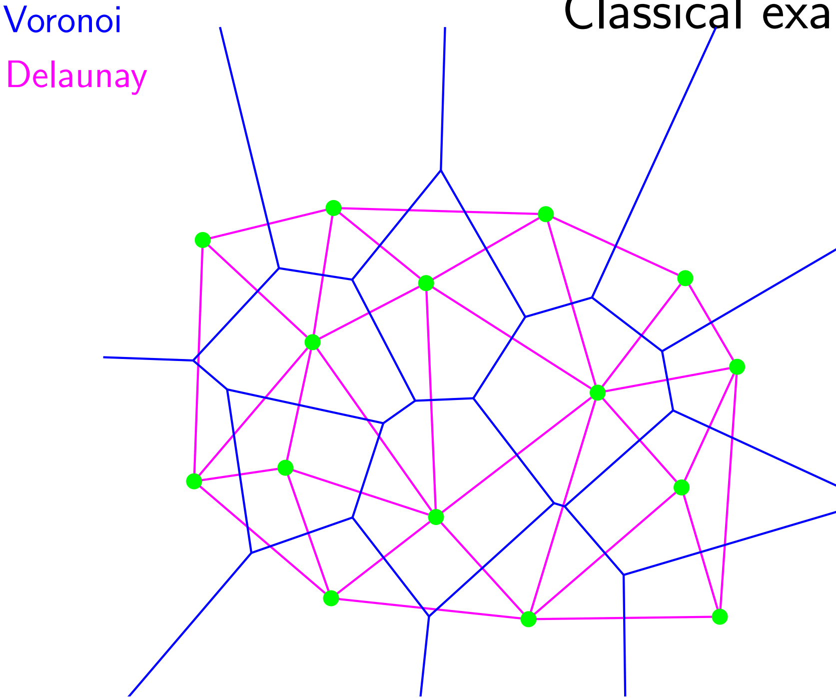
$$V_i = \{q; \forall j \neq i |qp_i| \leq |qp_j|\}$$

Classical example



Boris N. Delaunay
(1890–1980)

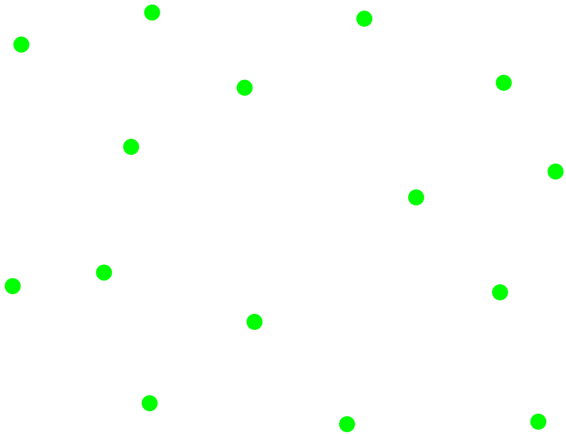
Classical example



Boris N. Delaunay
(1890–1980)

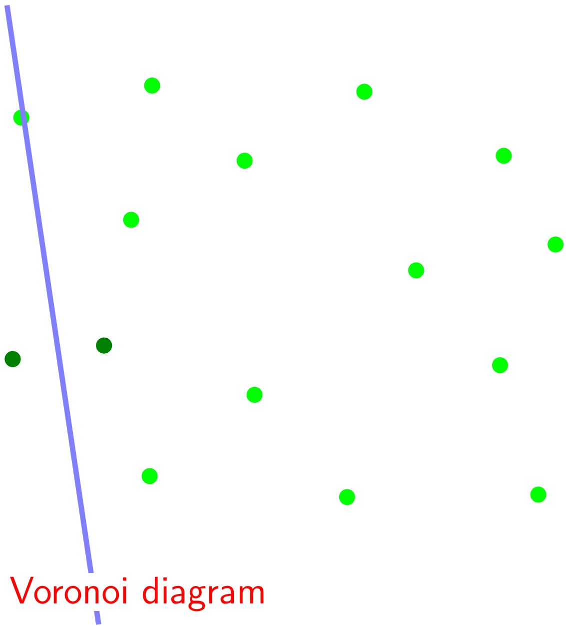
Voronoi $\leftrightarrow$ geometry Delaunay $\leftrightarrow$ topology

Voronoi



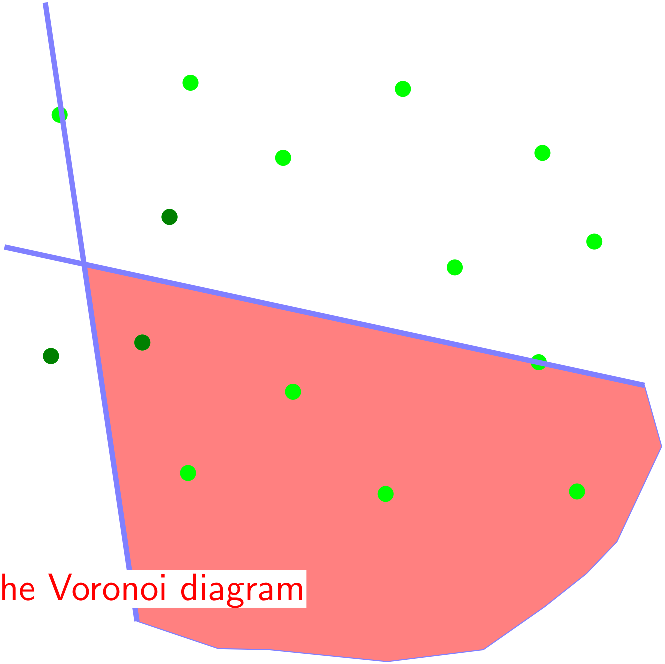
faces of the Voronoi diagram

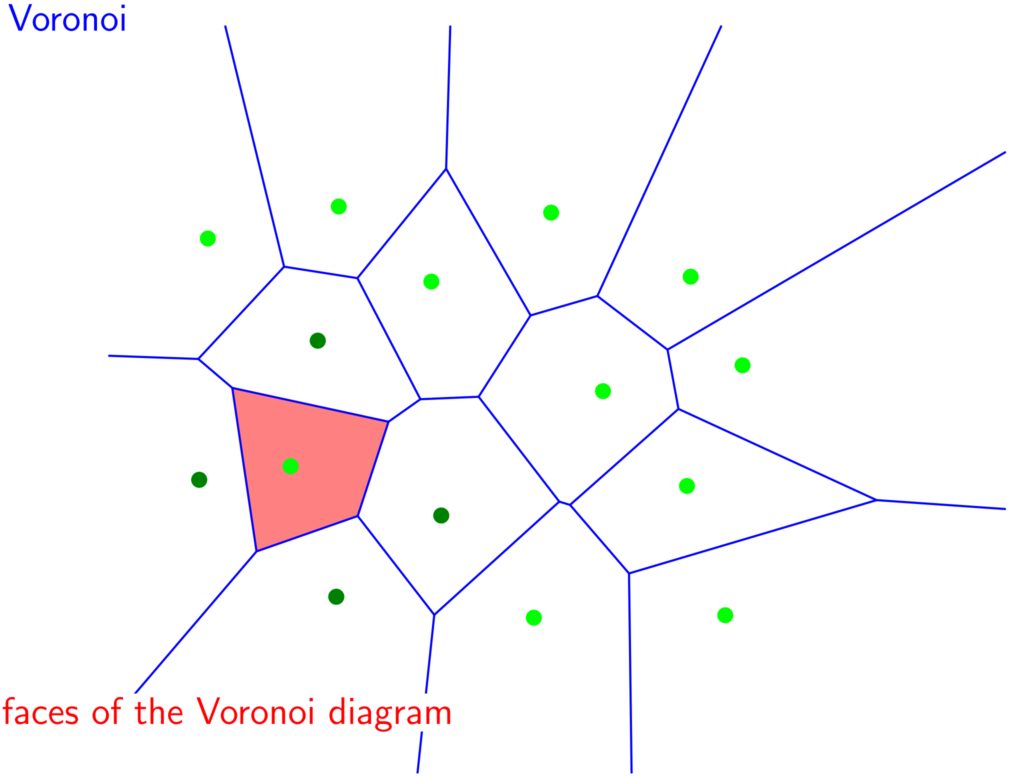
Voronoi



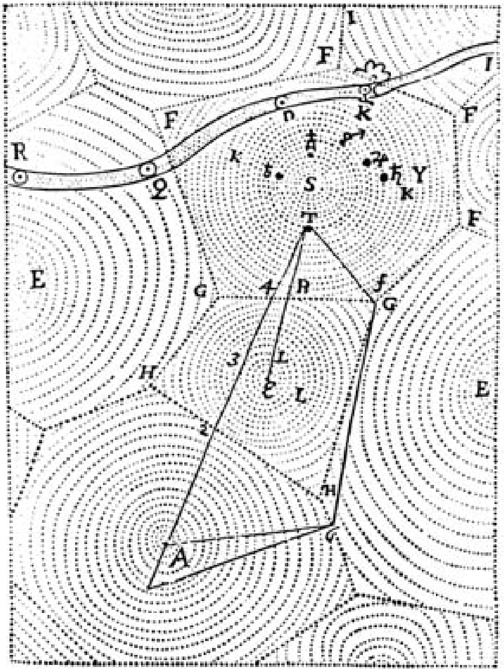
faces of the Voronoi diagram

Voronoi

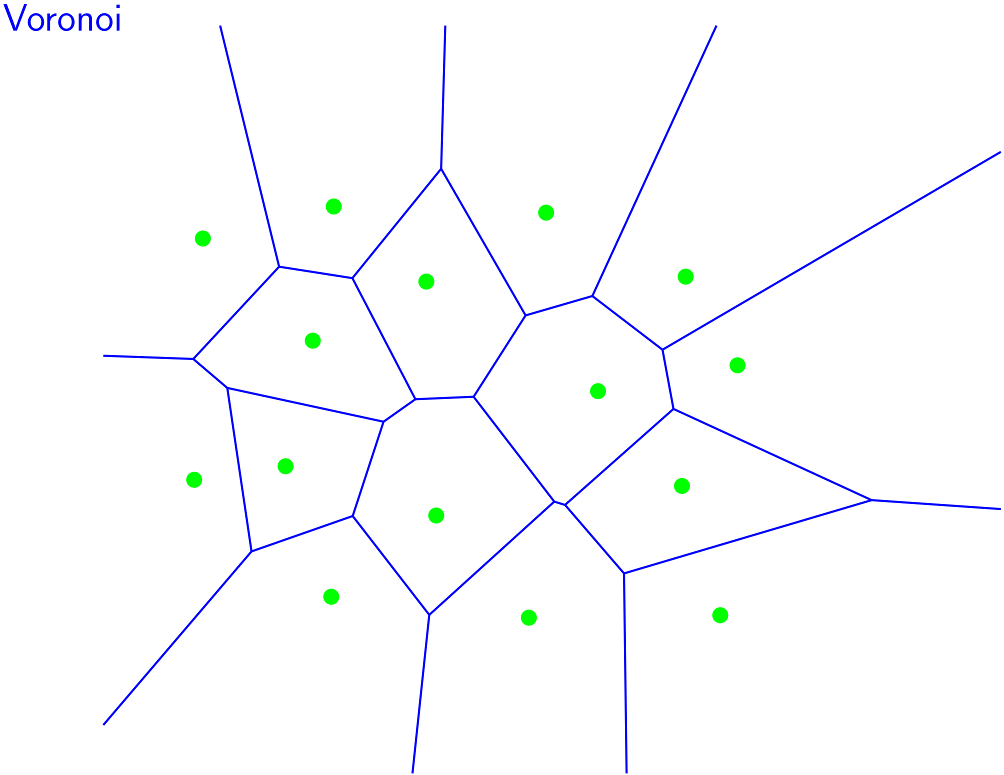


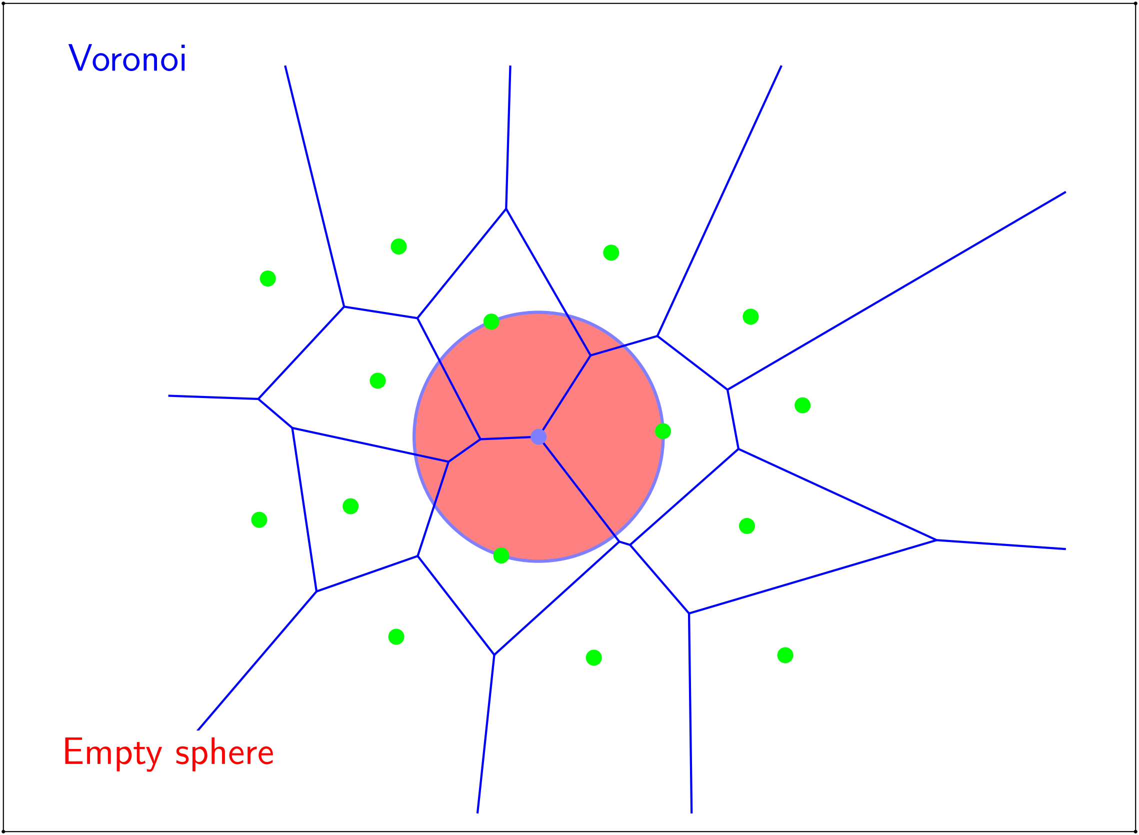

Voronoi
faces of the Voronoi diagram

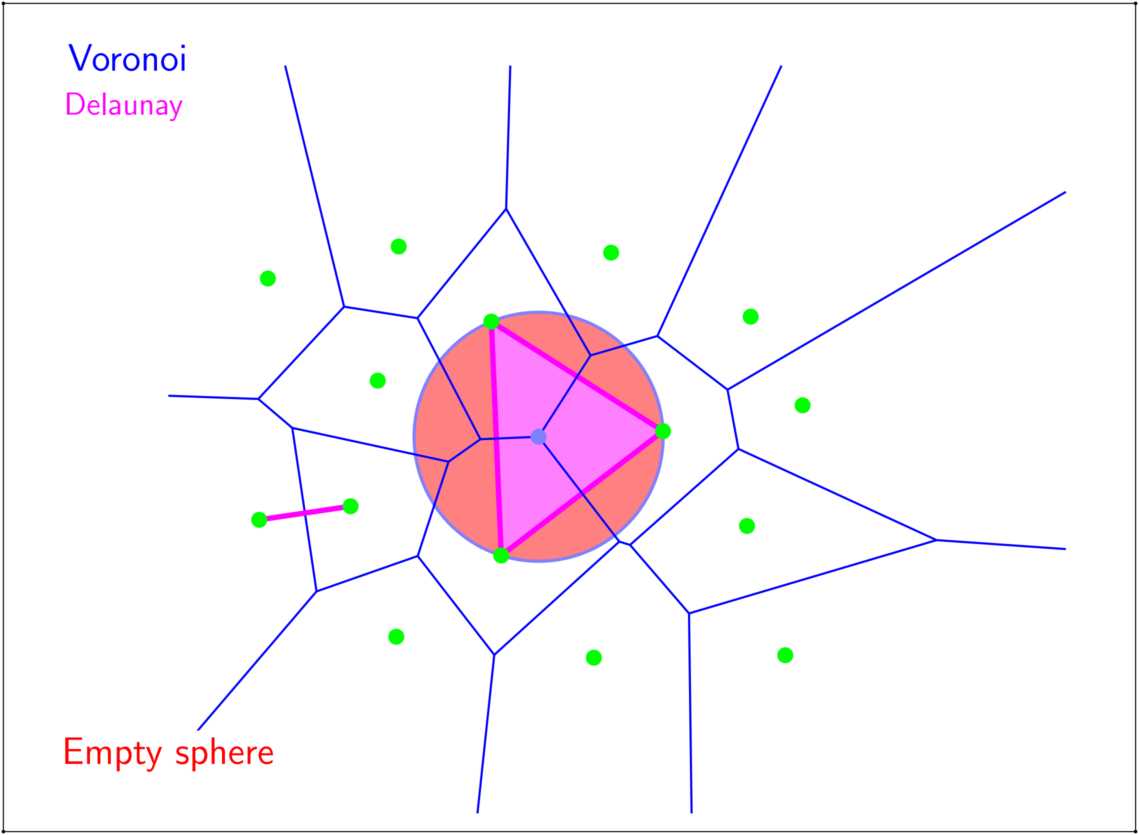
Voronoi is everywhere

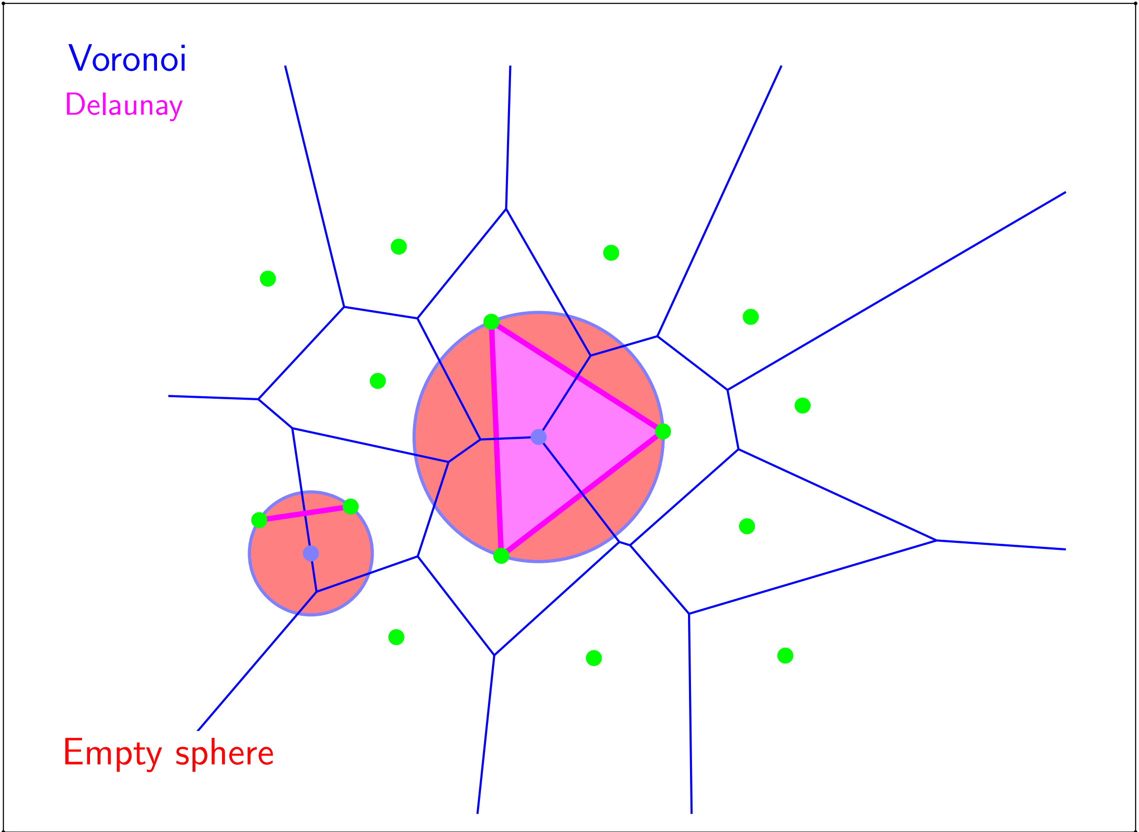


THE Delaunay property


Voronoi

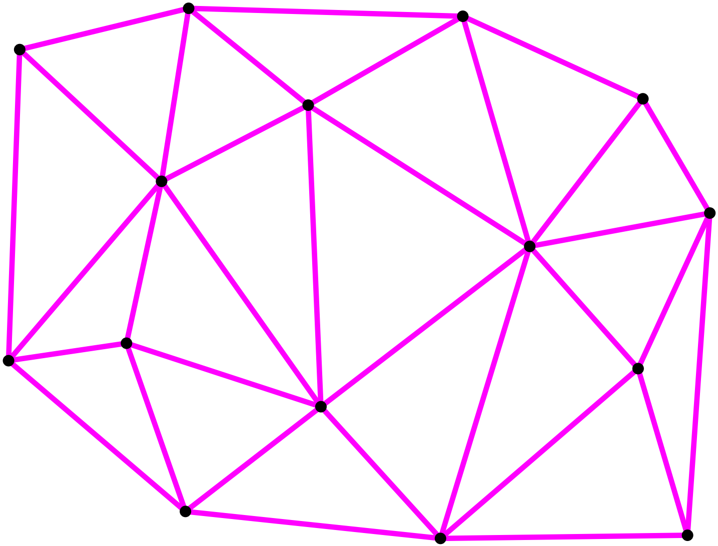

Voronoi
Empty sphere


Voronoi
Delaunay
Empty sphere

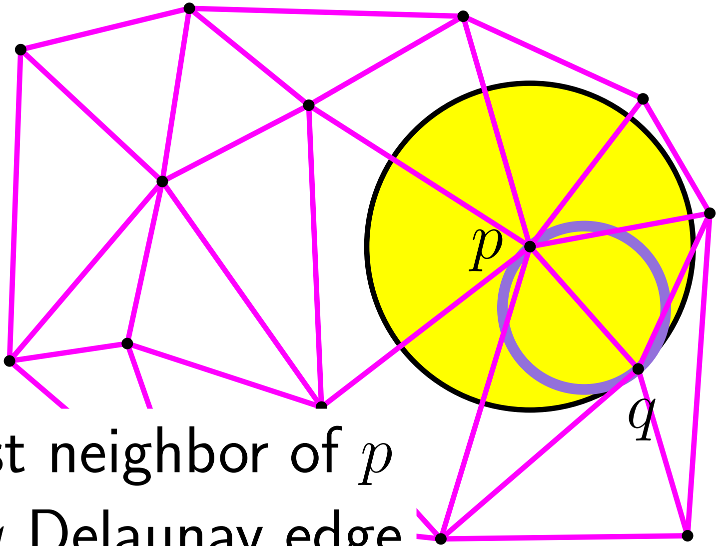

Voronoi
Delaunay
Empty sphere

Several applications

nearest neighbor graph



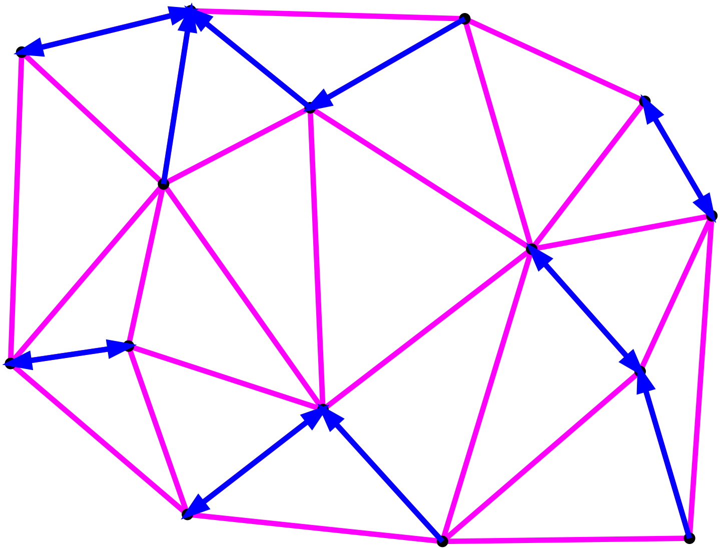
nearest neighbor graph



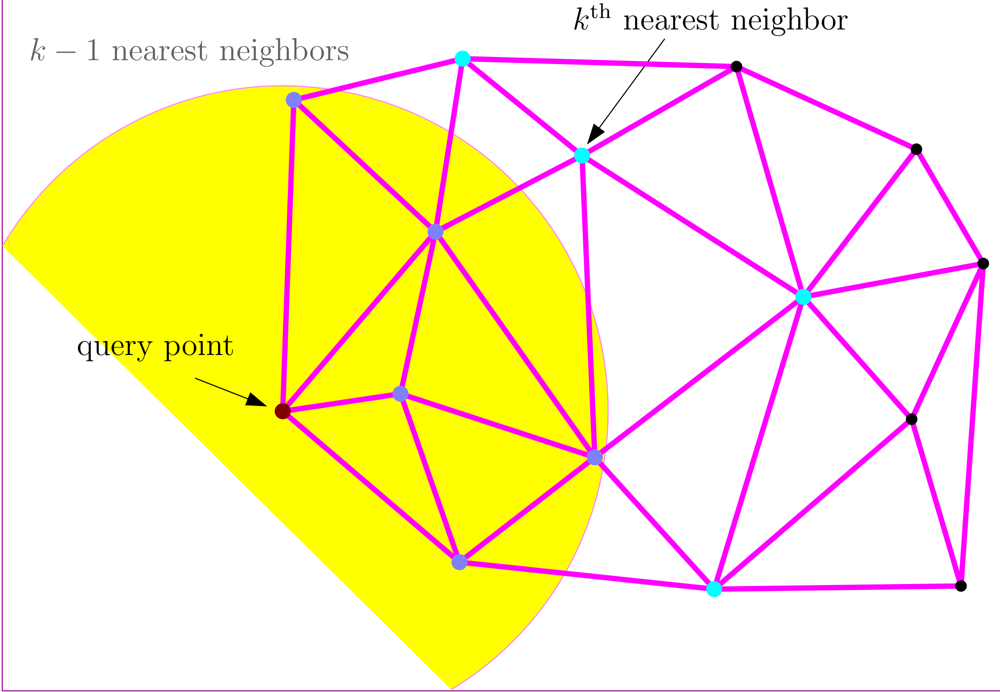
q nearest neighbor of p

$\Rightarrow pq$ Delaunay edge

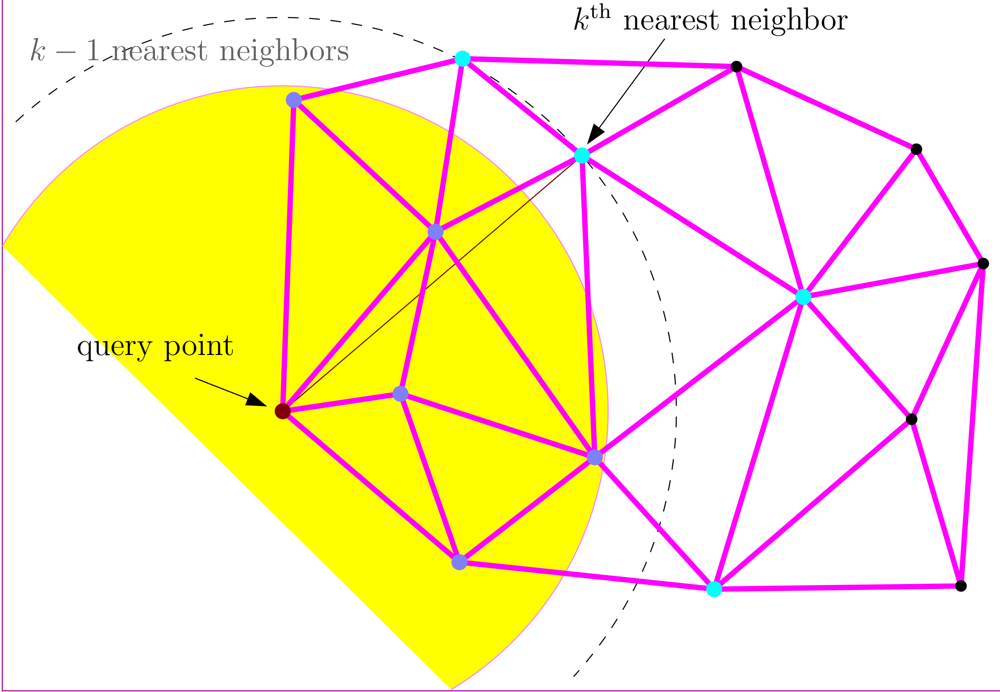
nearest neighbor graph



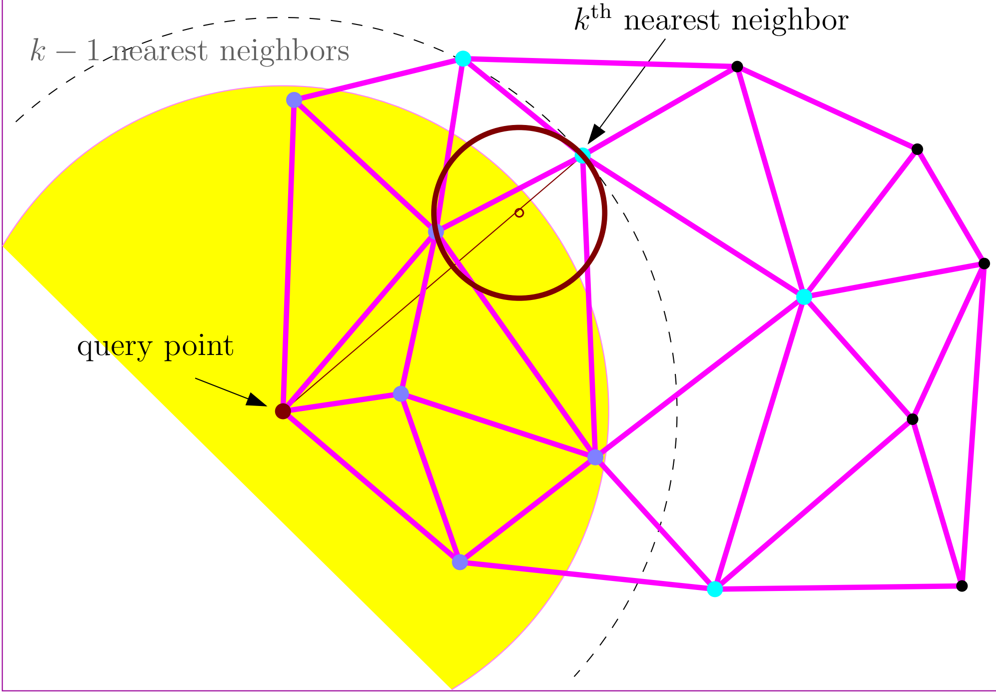
k nearest neighbors



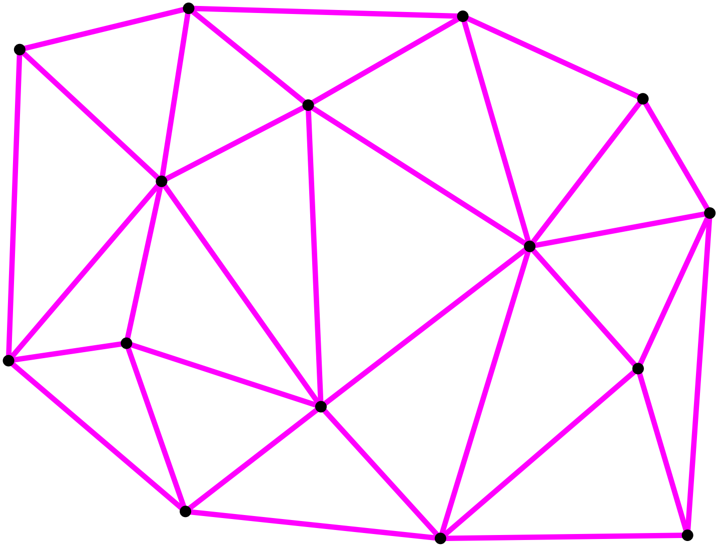
k nearest neighbors



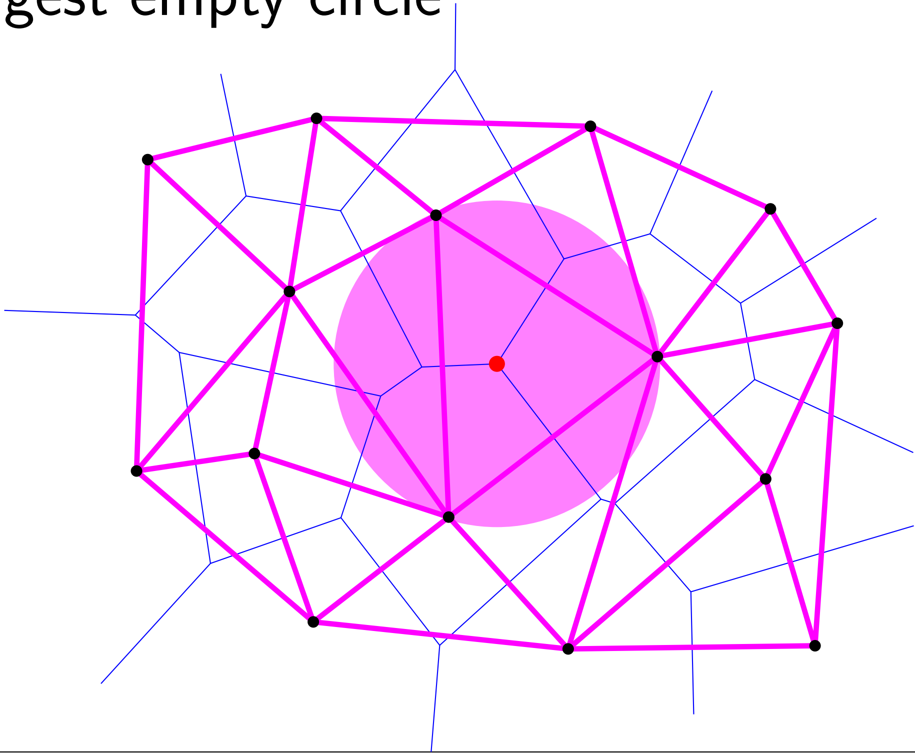
k nearest neighbors



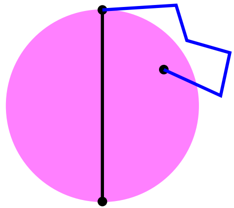
Largest empty circle

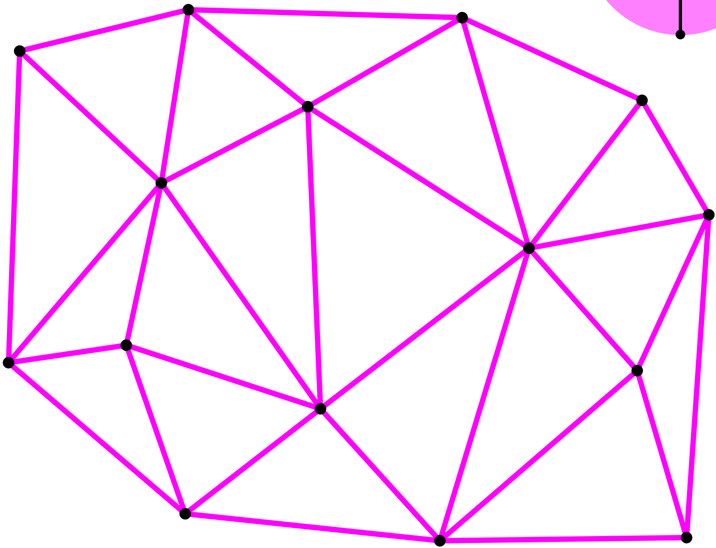


Largest empty circle

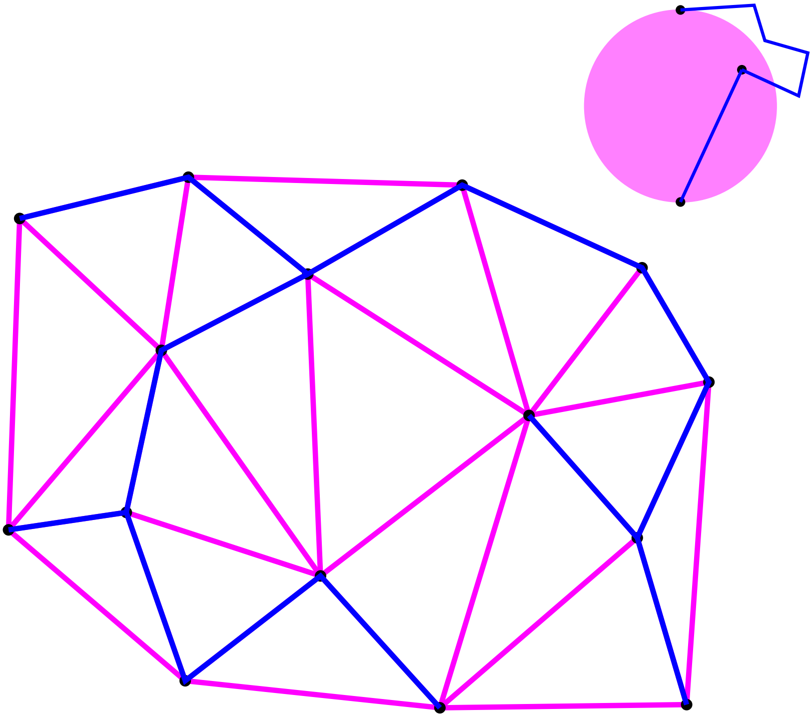


MST



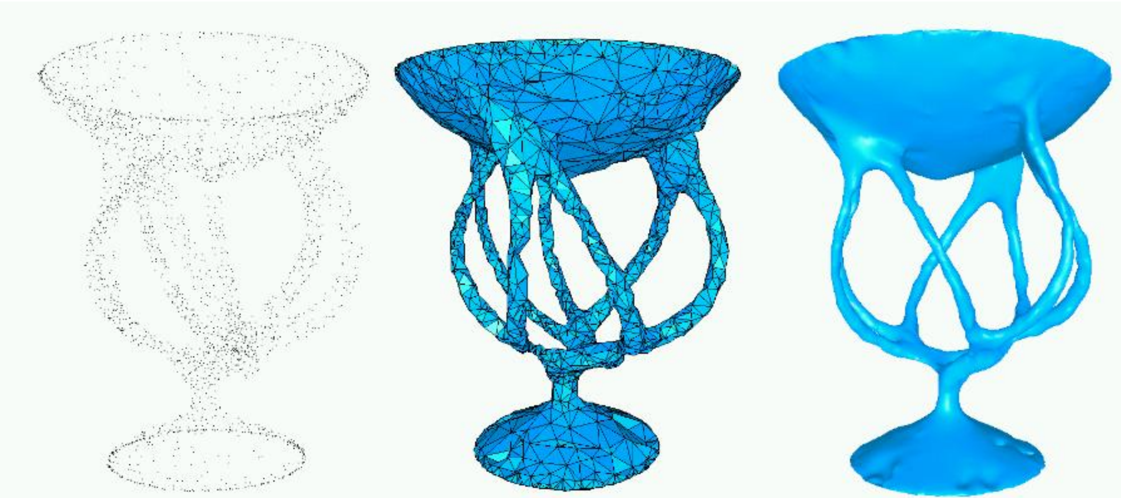


MST



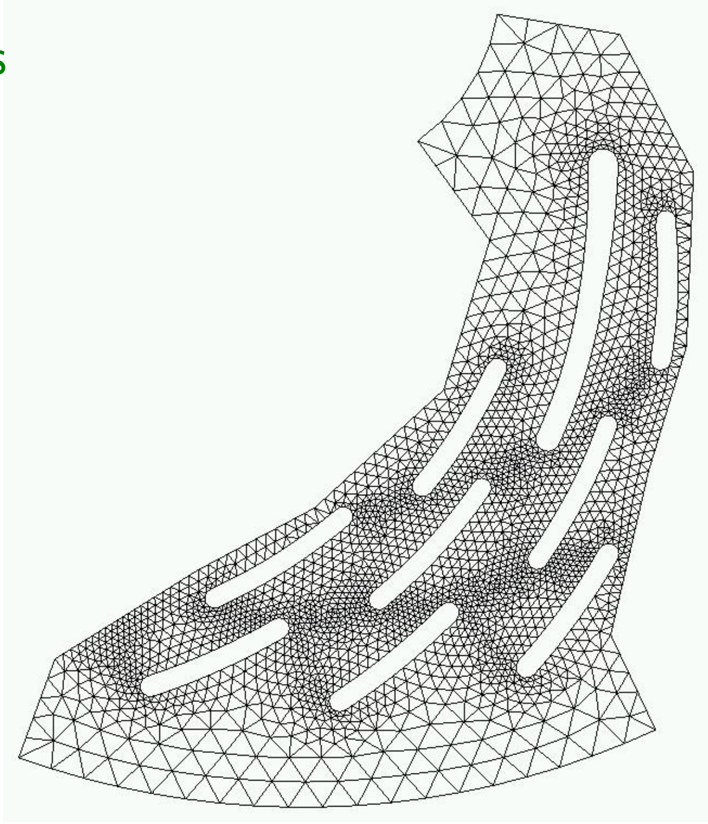
Other applications

Reconstruction



Other applications

- Reconstruction
- Meshing



Other applications

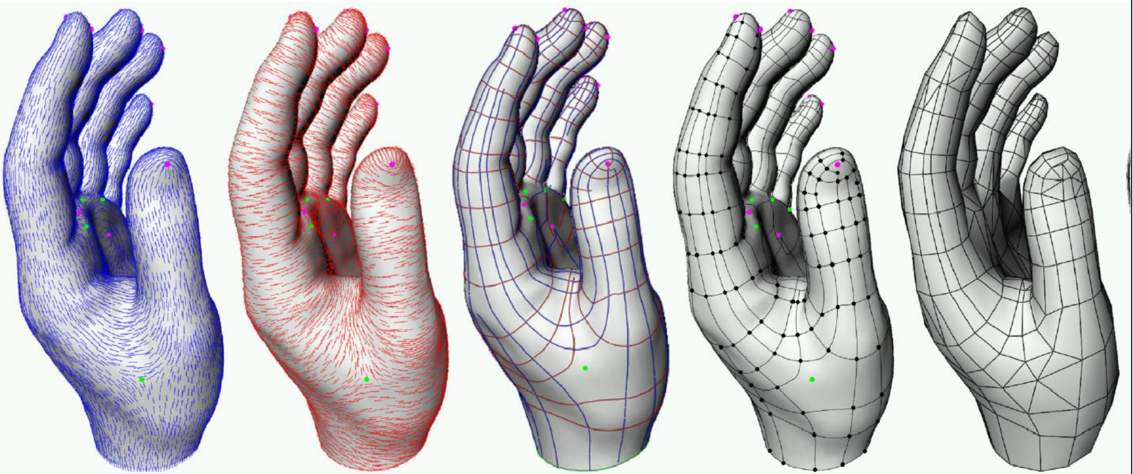
Reconstruction

Meshing / Remeshing

Other applications

Reconstruction

Meshing / Remeshing

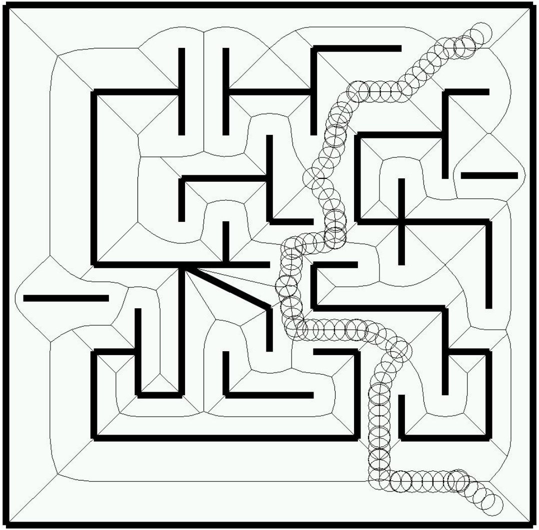


Other applications

Reconstruction

Meshing / Remeshing

Path planning



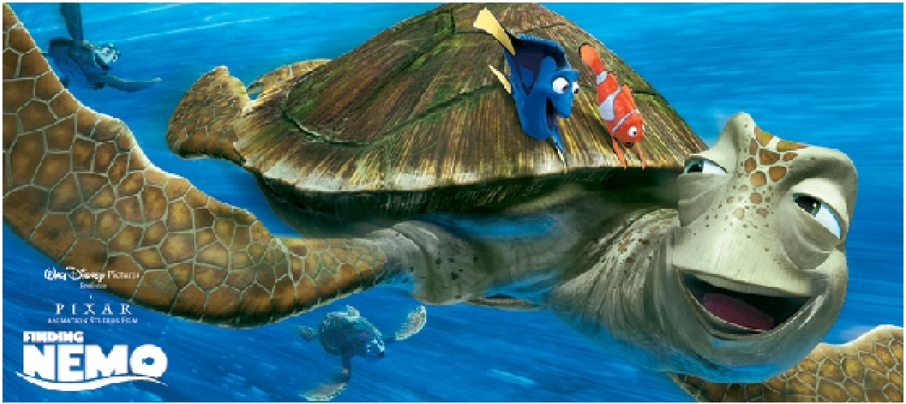
Other applications

Reconstruction

Meshing / Remeshing

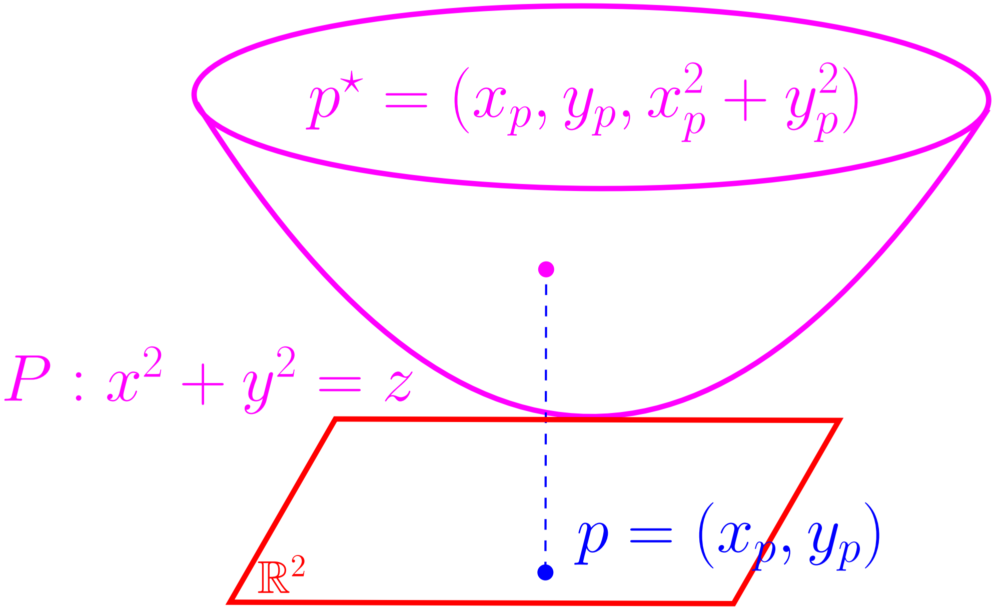
Path planning

and others...

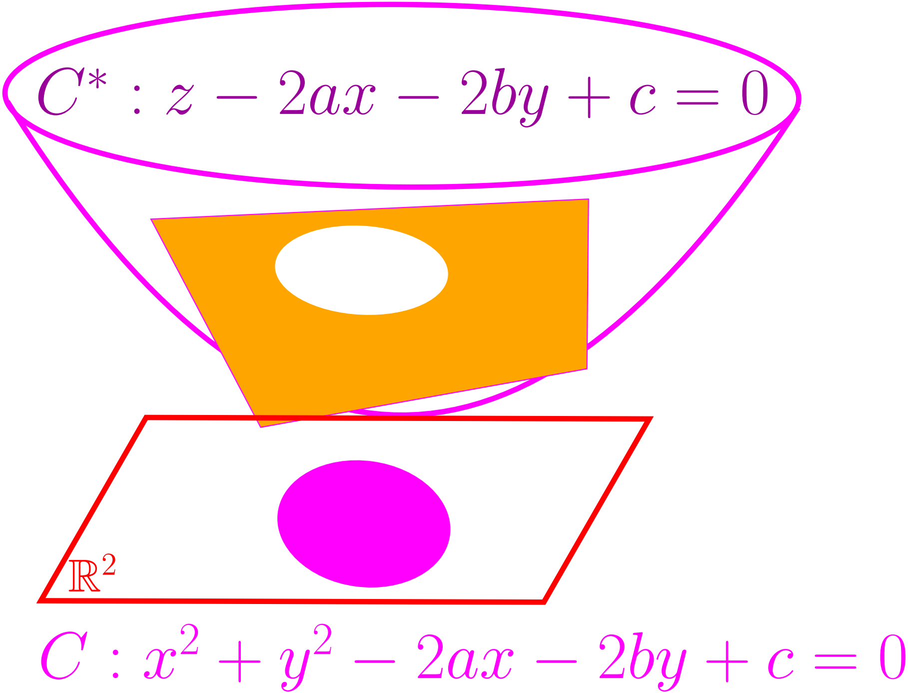


Main properties of Delaunay

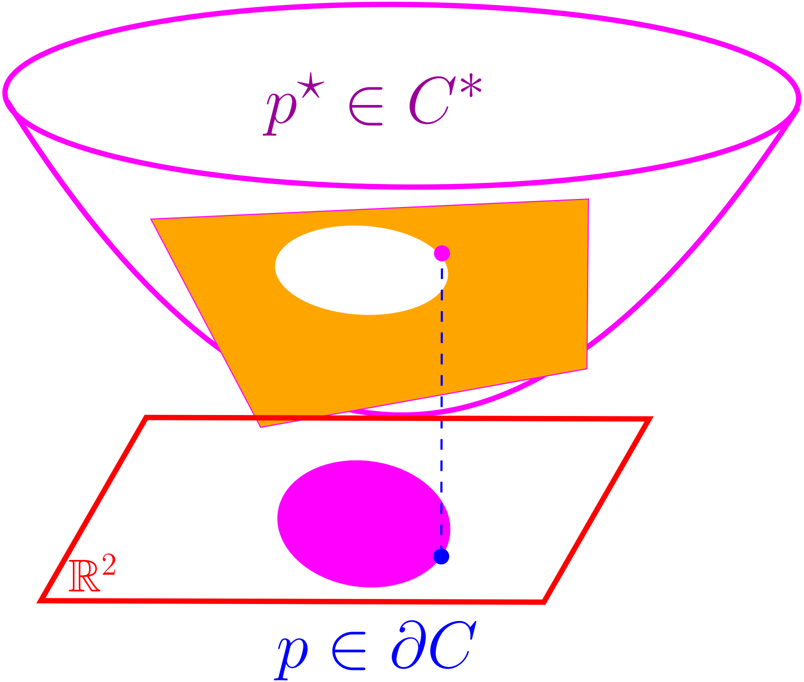
point / sphere duality



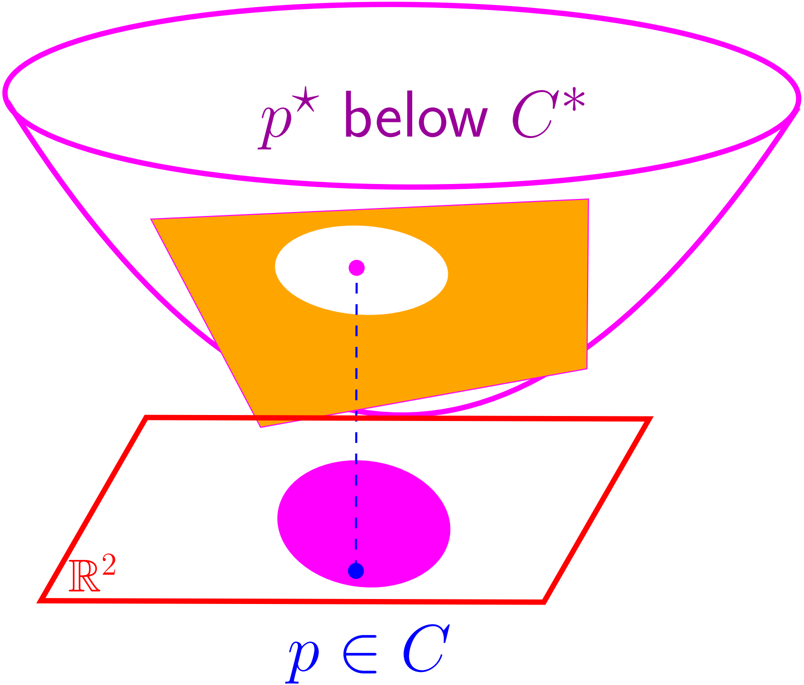
point / sphere duality



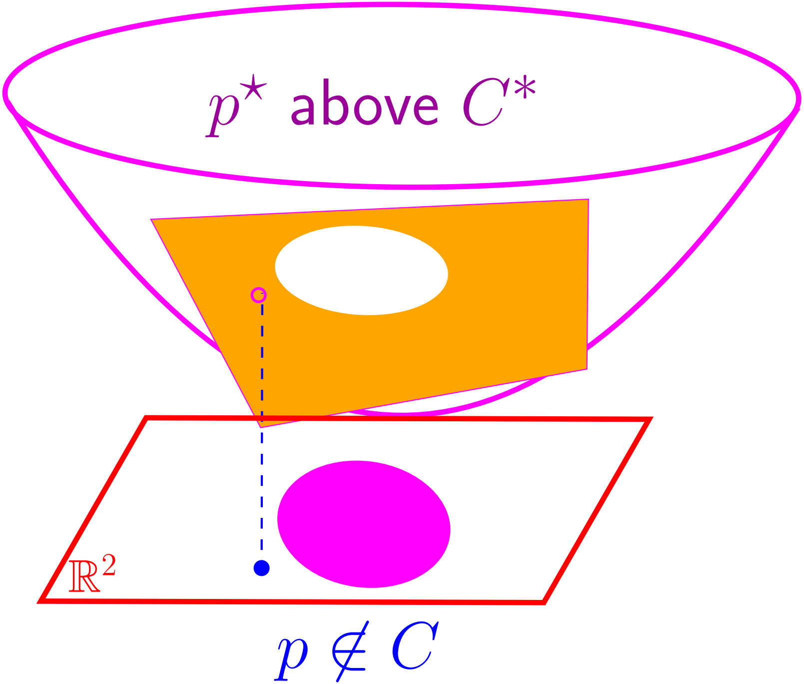
point / sphere duality



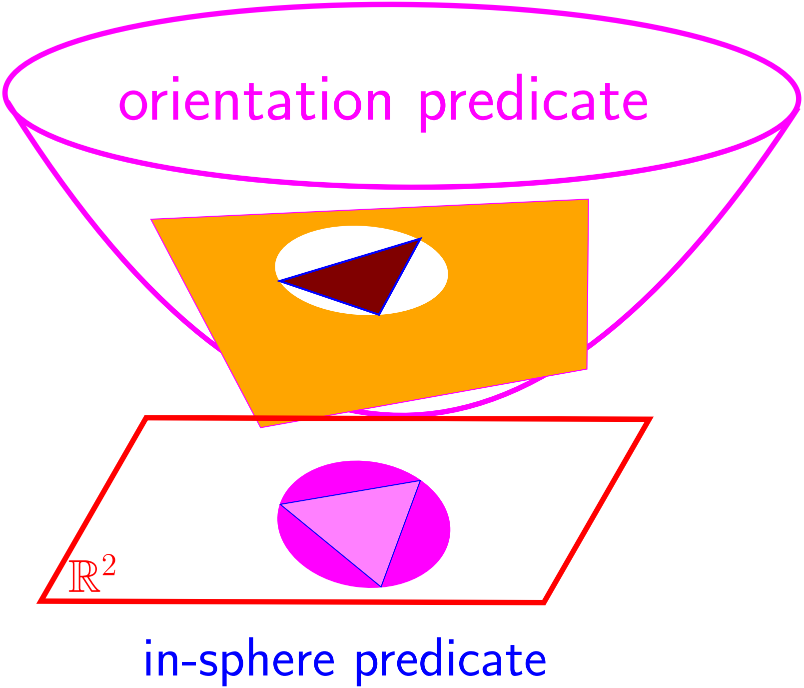
point / sphere duality



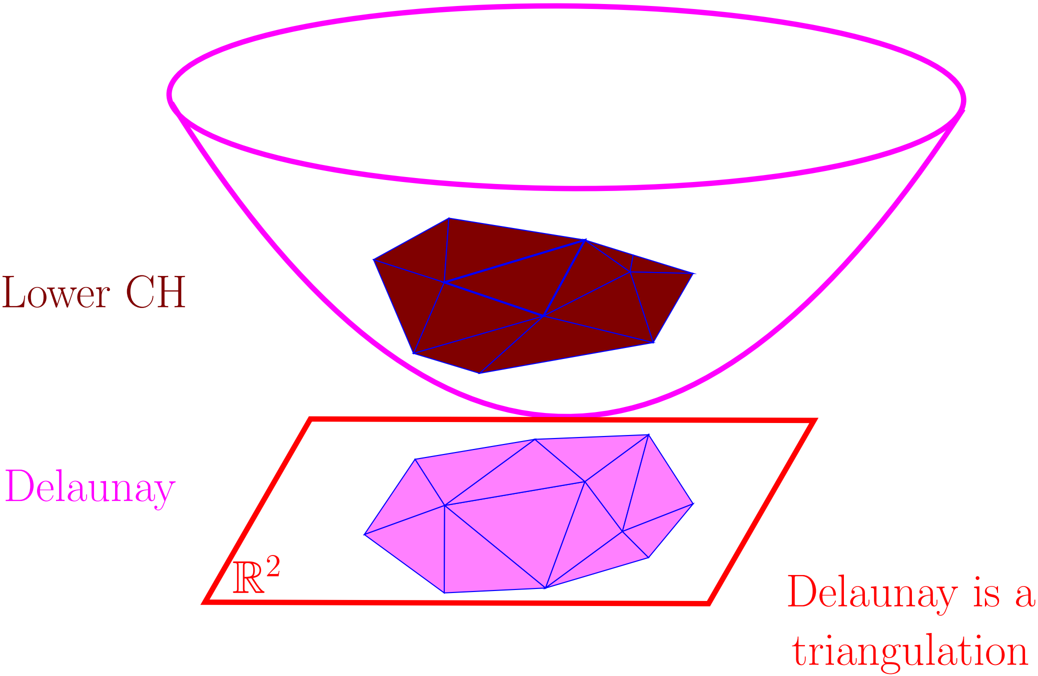
point / sphere duality



point / sphere duality



point / sphere duality



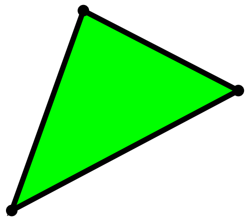
Euler formula

f: number of facets (except ∞)

e: number of edges

v: number of vertices

$$f - e + v = 1$$



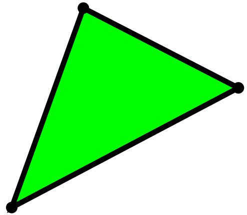
Euler formula

f: number of facets (except ∞)

e: number of edges

v: number of vertices

$$f - e + v = 1$$



$$1 - 3 + 3 = 1$$

Euler formula

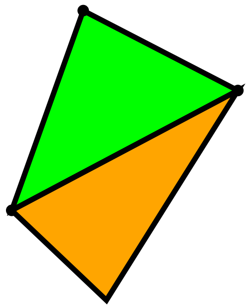
f: number of facets (except ∞)

e: number of edges

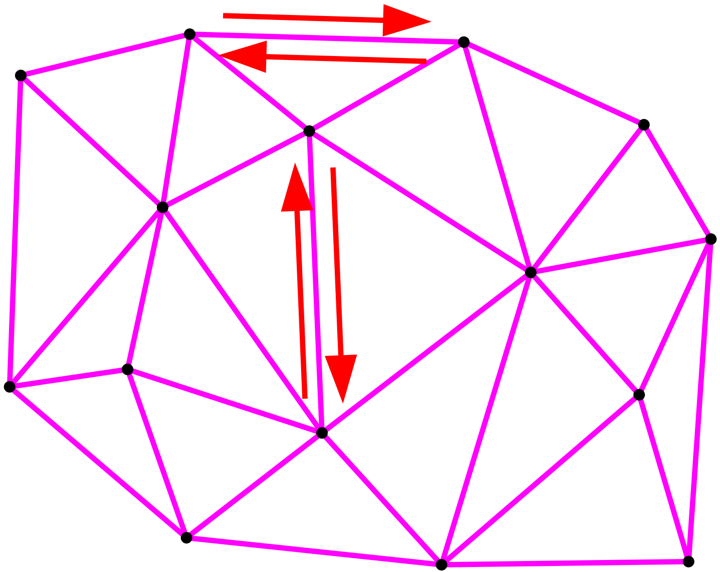
v: number of vertices

$$f - e + v = 1$$

$$+1 - 2 + 1 = +0$$



k: size of ∞ facet



number of oriented edges

in a triangulation: $2e = 3f + k$

Euler formula

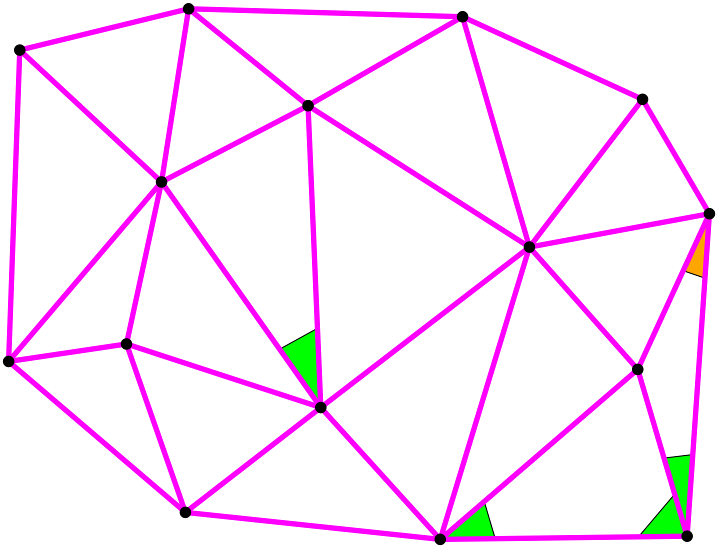
$$f - e + v = 1$$

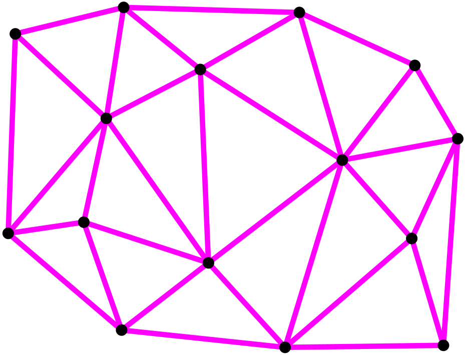
Triangulation

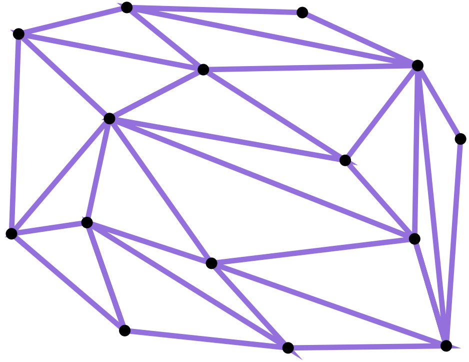
$$2e = 3f + k$$

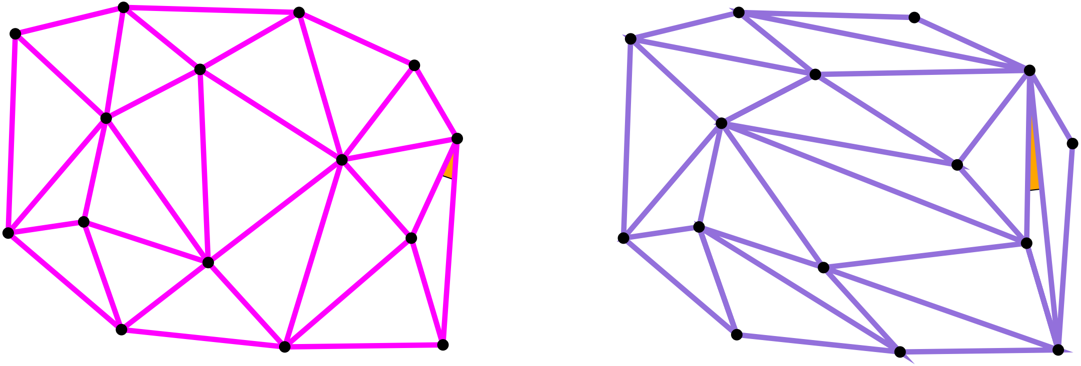
$$f = 2v - 2 - k = O(v)$$
$$e = 3v - 3 - k = O(v)$$

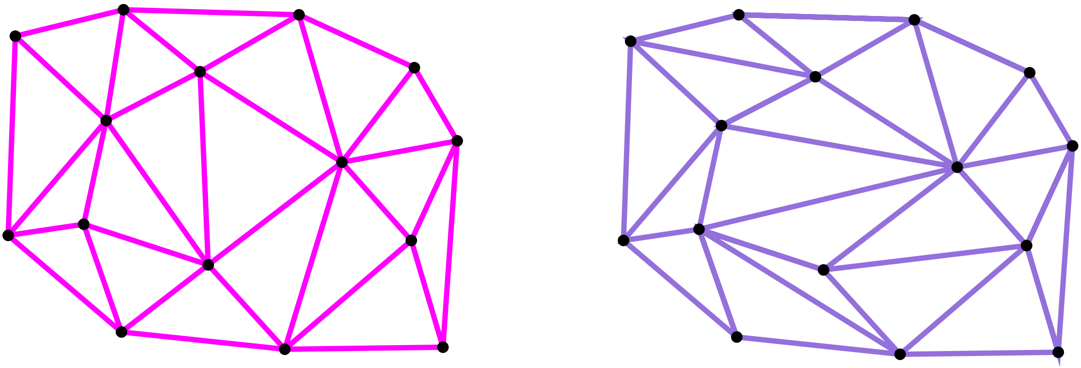
Delaunay maximizes the smallest angle

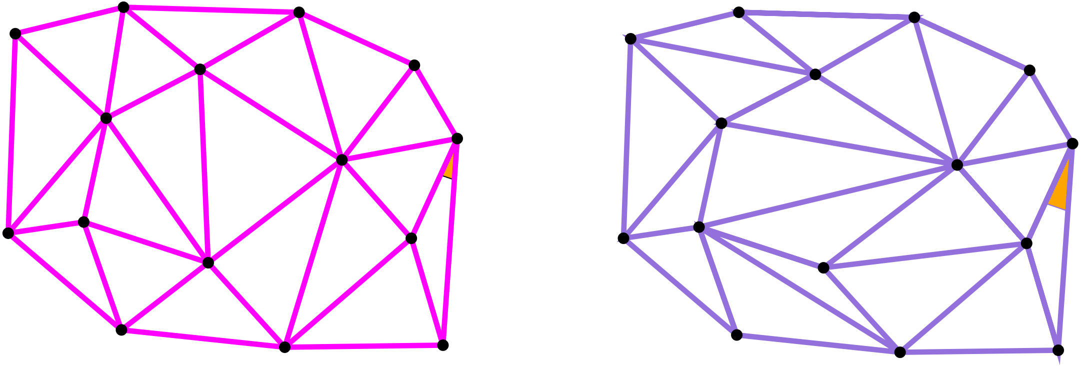


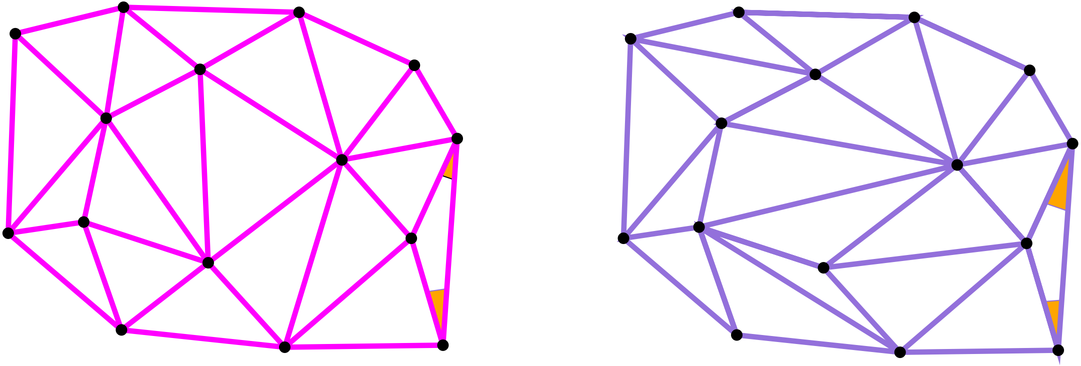


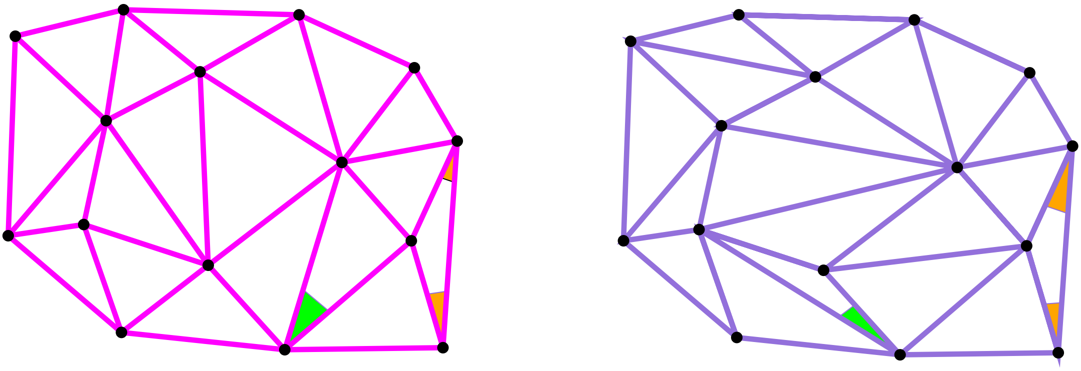






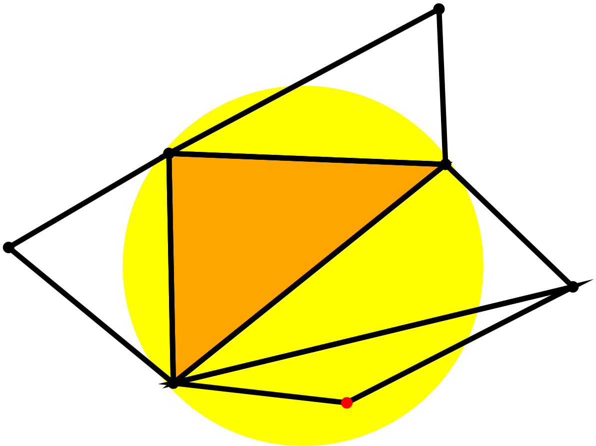






→ Delaunay maximizes the sequence of angles in lexicographical order

Local optimality vs global optimality



locally Delaunay... but not globally Delaunay

Theorem

Locally Delaunay everywhere

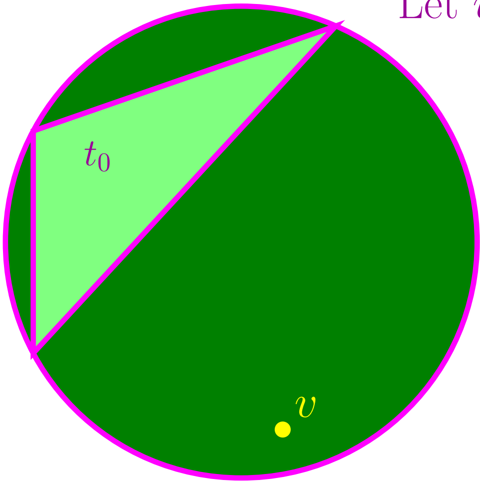
$$\Longleftrightarrow$$

Globally Delaunay

Proof:

Let t_0 be locally Delaunay, but not globally Delaunay

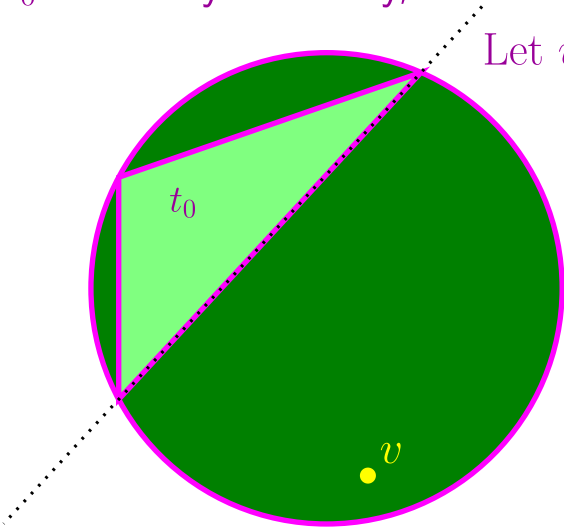
Let $v \in \text{circle}(t)$ $(v \notin t)$



Proof:

Let t_0 be locally Delaunay, but not globally Delaunay

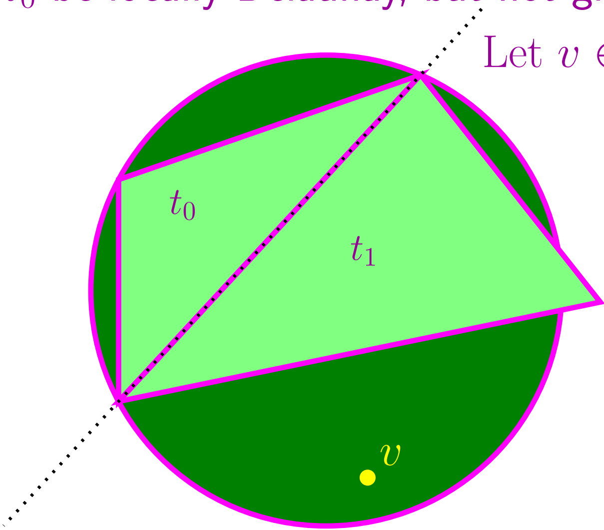
Let $v \in \text{circle}(t)$ $(v \notin t)$



Proof:

Let t_0 be locally Delaunay, but not globally Delaunay

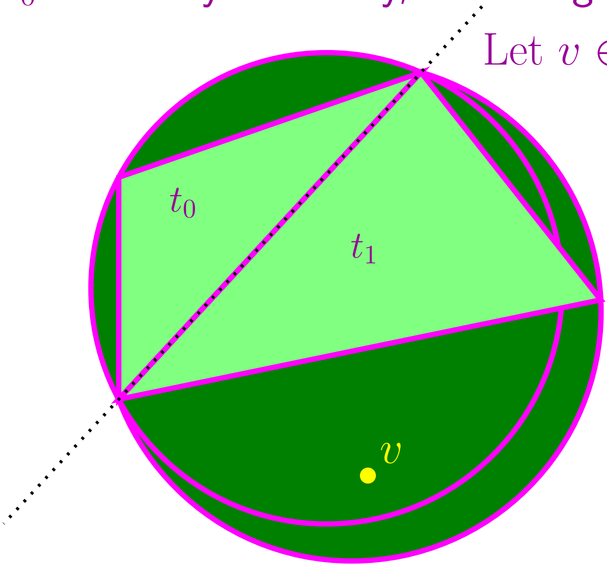
Let $v \in \text{circle}(t)$ $(v \notin t)$



Proof:

Let t_0 be locally Delaunay, but not globally Delaunay

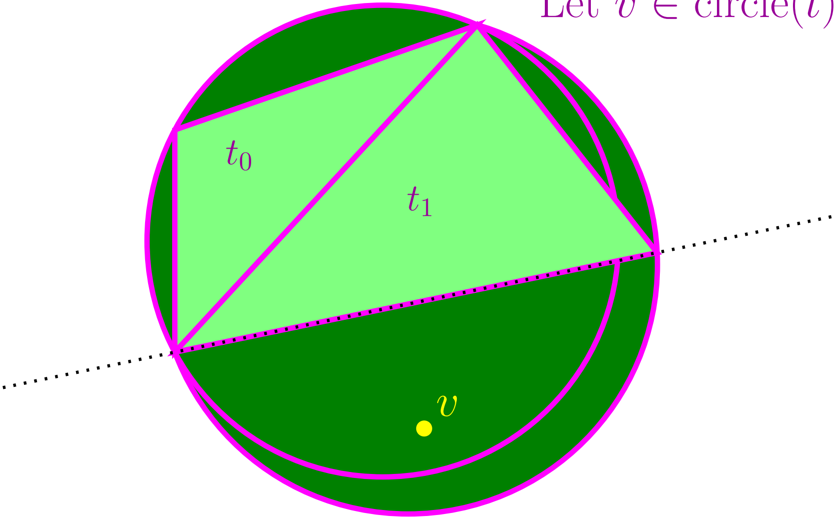
Let $v \in \text{circle}(t)$ $(v \notin t)$



Proof:

Let t_0 be locally Delaunay, but not globally Delaunay

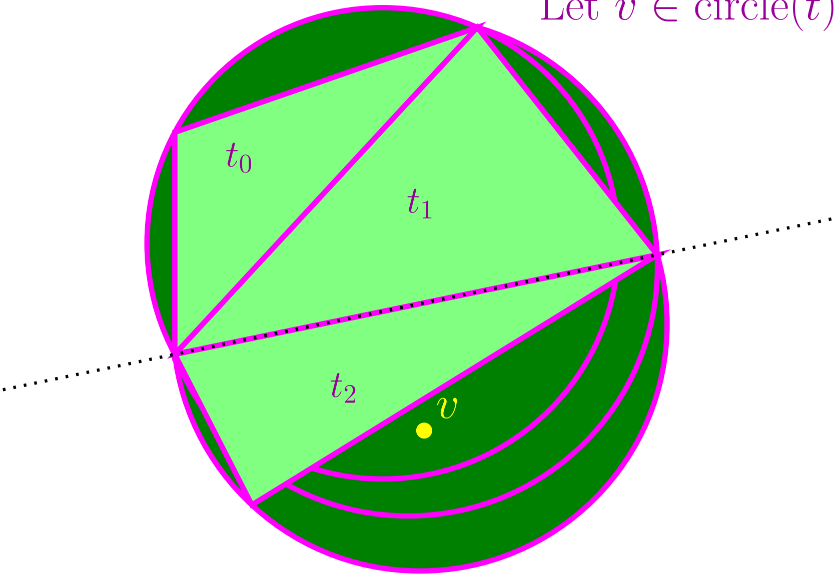
Let $v \in \mathrm{circle}(t)$ $(v \notin t)$



Proof:

Let t_0 be locally Delaunay, but not globally Delaunay

Let $v \in \text{circle}(t)$ $(v \notin t)$



Since $\exists$ finitely many triangles, at some point v is a vertex of t_i

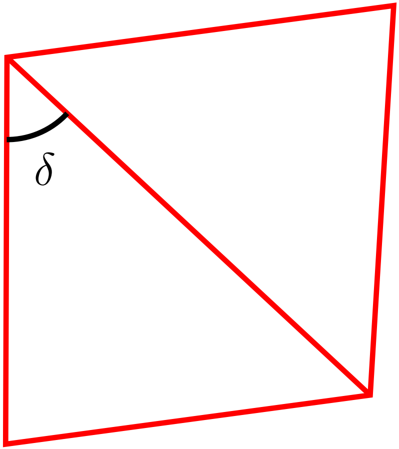
Local optimality and smallest angle

Case of 4 points

Lemma:
For any 4 points in convex position,
Delaunay $\iff$ smallest angle maximized

Local optimality and smallest angle

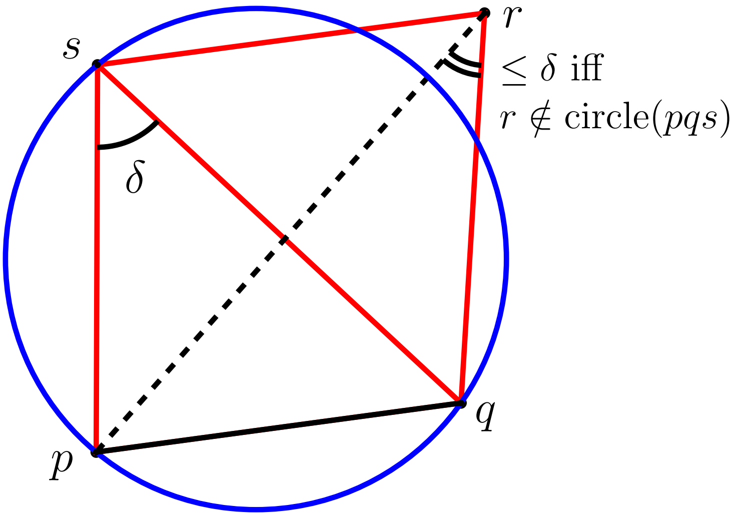
Case of 4 points



Let δ be the smallest angle

Local optimality and smallest angle

Case of 4 points



Let δ be the smallest angle

Local optimality and smallest angle

Theorem

Delaunay $\Longleftrightarrow$ maximum smallest angle

Proof:

Local optimality and smallest angle

Theorem

Delaunay $\iff$ maximum smallest angle

Proof:

T triangulation w/ max. smallest angle

Local optimality and smallest angle

Theorem

Delaunay $\Longleftrightarrow$ maximum smallest angle

Proof:

T triangulation w/ max. smallest angle

$\Longrightarrow$ max. in each quadrilateral

Local optimality and smallest angle

Theorem

Delaunay $\iff$ maximum smallest angle

Proof:

T triangulation w/ max. smallest angle

$\Longrightarrow$ max. in each quadrilateral

$\Longrightarrow$ locally Delaunay

Local optimality and smallest angle

Theorem

Delaunay $\Longleftrightarrow$ maximum smallest angle

Proof:

T triangulation w/ max. smallest angle

$\Longrightarrow$ max. in each quadrilateral

$\Longrightarrow$ locally Delaunay

$\Longrightarrow$ globally Delaunay

Computing Delaunay

Lower bound

Lower bound for Delaunay

Delaunay can be used to sort numbers

Lower bound for Delaunay

Delaunay can be used to sort numbers

Take an instance of sort

Assume one can compute Delaunay in $\mathbb{R}^2$

Use Delaunay to solve this instance of sort

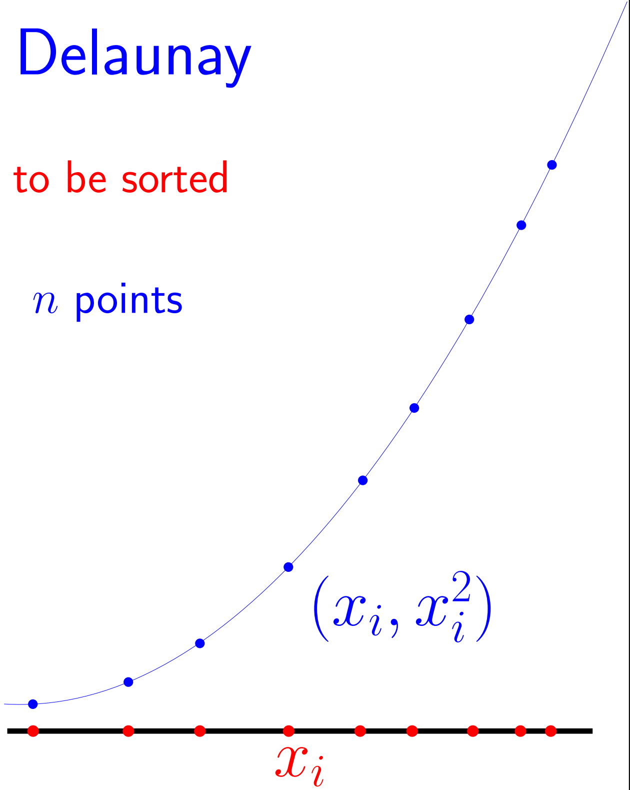
Lower bound for Delaunay

Let $x_1, x_2, \ldots, x_n \in \mathbb{R}$, to be sorted

Lower bound for Delaunay

Let $x_1, x_2, \ldots, x_n \in \mathbb{R}$, to be sorted

$(x_1, x_1^2), \ldots, (x_n, x_n^2)$ n points



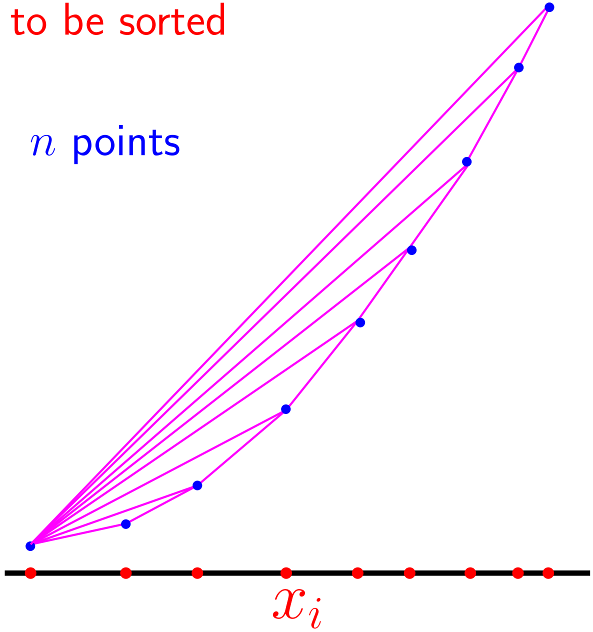
Lower bound for Delaunay

Let $x_1, x_2, \ldots, x_n \in \mathbb{R}$, to be sorted

$(x_1, x_1^2), \ldots, (x_n, x_n^2)$ n points

Delaunay

$\rightarrow$ order in x



Lower bound for Delaunay

Let $x_1, x_2, \ldots, x_n \in \mathbb{R}$, to be sorted

$\downarrow O(n)$

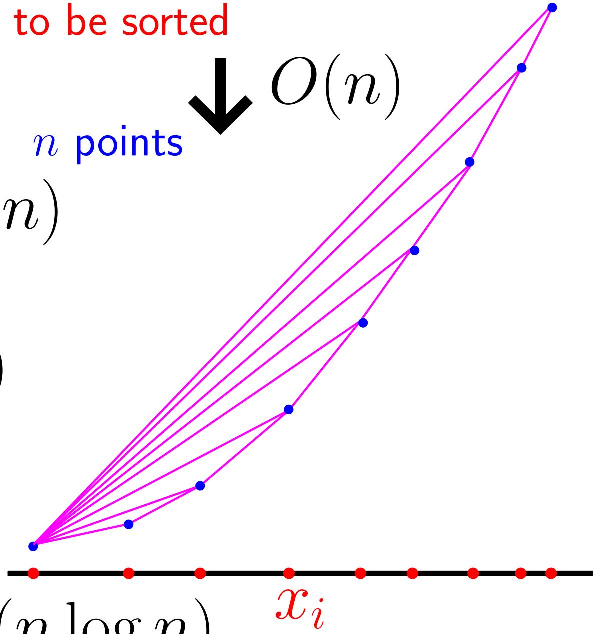
$(x_1, x_1^2), \ldots, (x_n, x_n^2)$ $\quad$ n points

$\downarrow f(n)$

Delaunay

$\downarrow O(n)$

$\rightarrow$ order in x



$$O(n) + f(n) \in \Omega(n \log n)$$

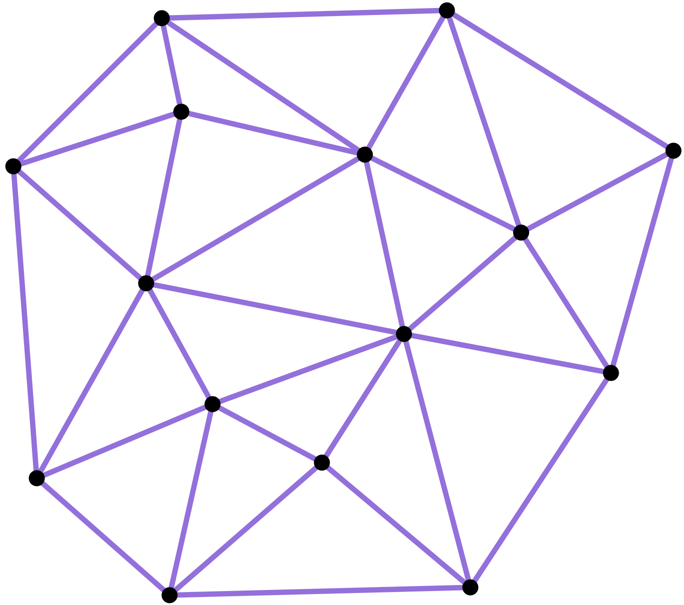
Lower bound for Delaunay

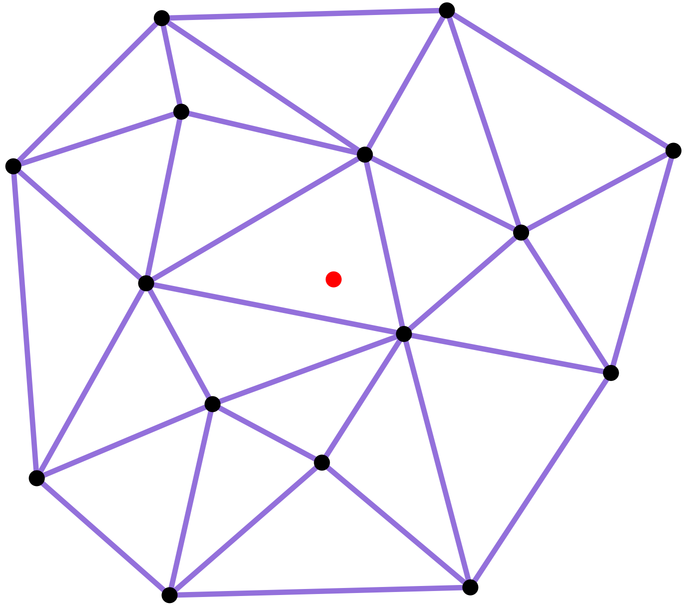
$$\Omega(n \log n)$$

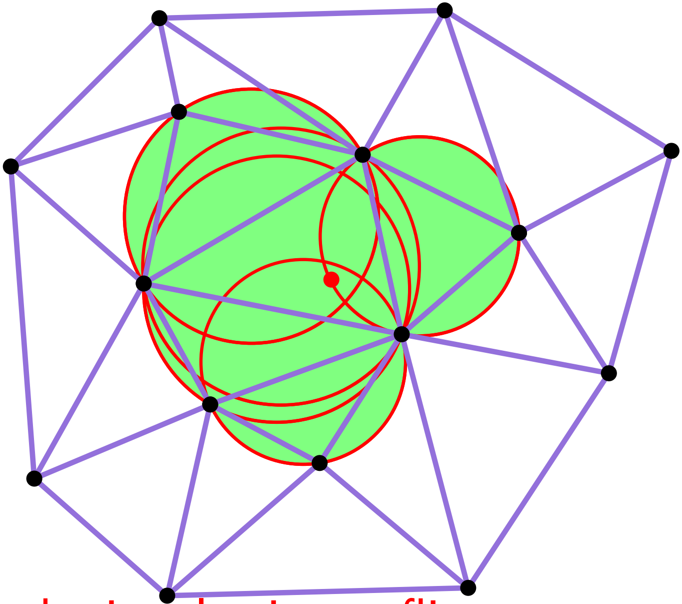
Computing Delaunay

Incremental algorithm

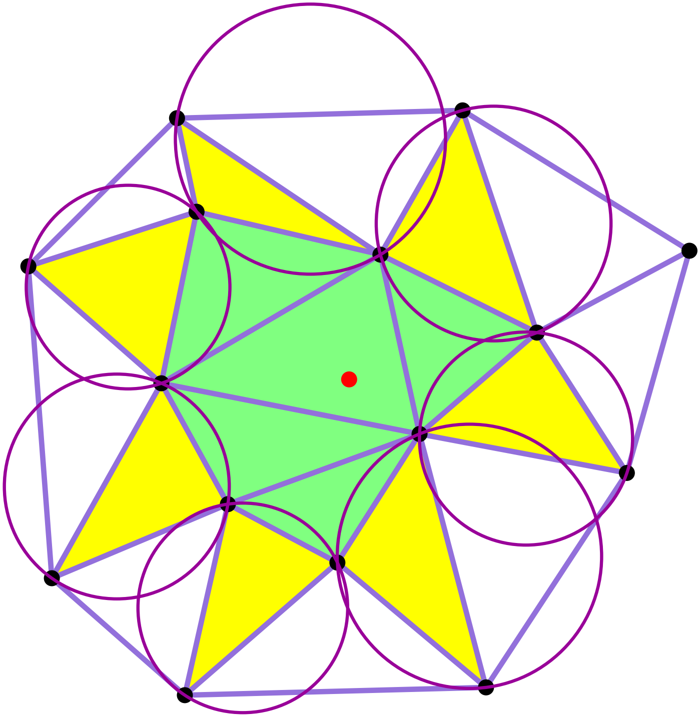
(SHORT OVERVIEW)

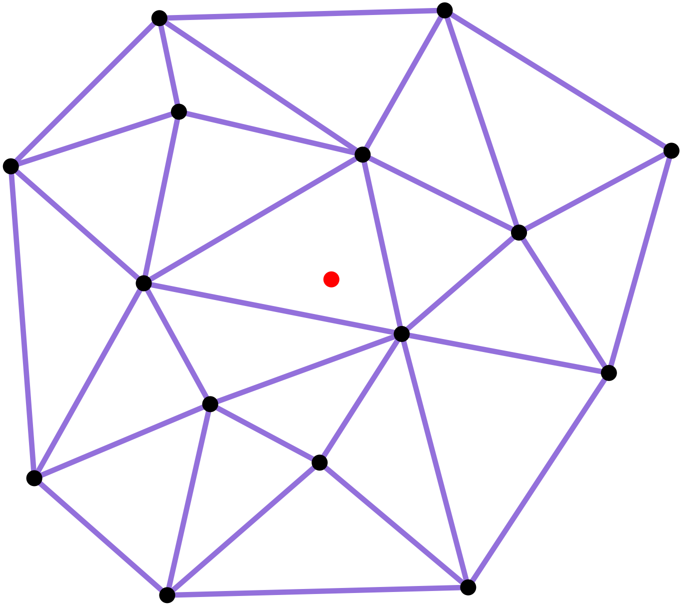


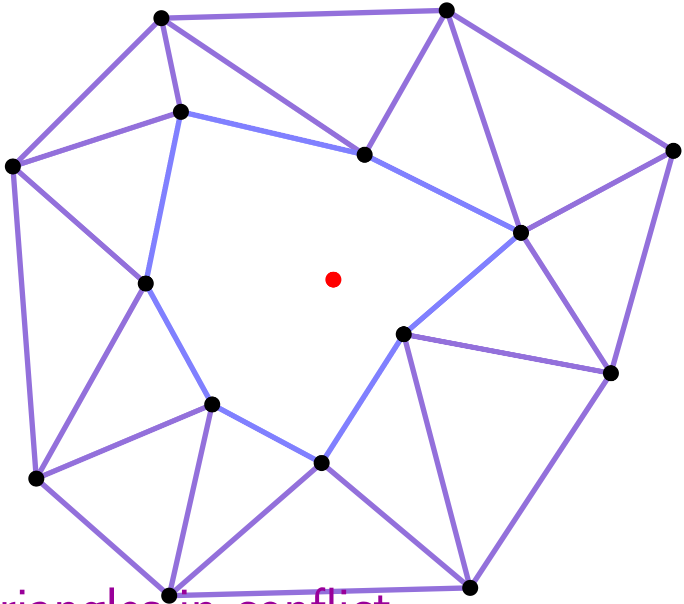




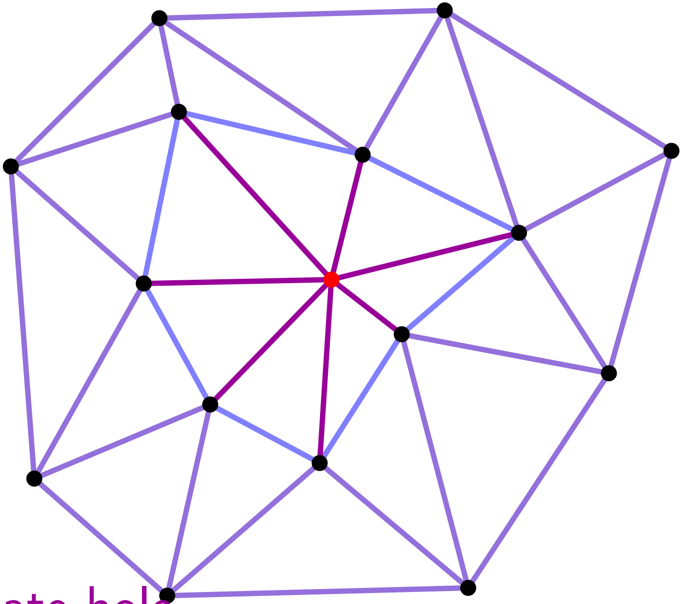
Find triangles in conflict







Delete triangles in conflict



Triangulate hole

That's all for today