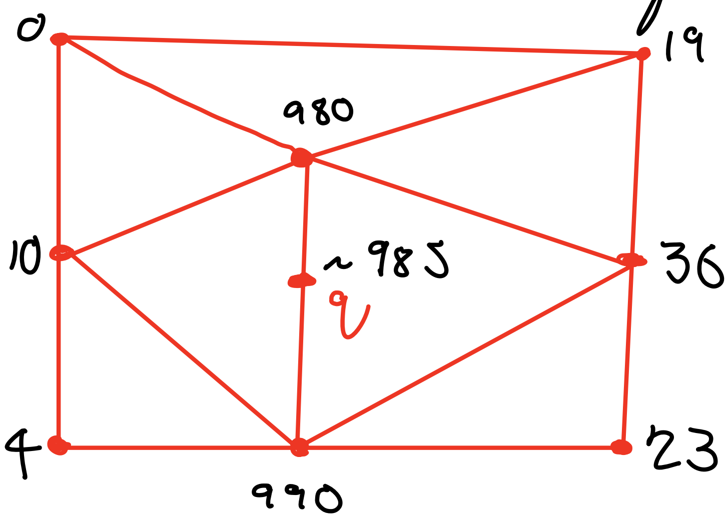
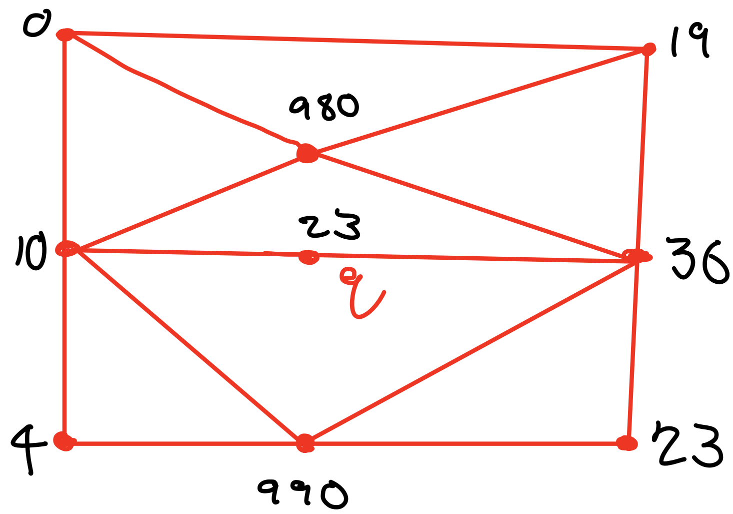
L12 - Delaunay Triangulation

- Consider $f: \mathbb{R}^2 \longrightarrow \mathbb{R}$ whose values we know for finitely many points $P \subseteq \mathbb{R}^2$.
- How can we approximate f on whole plane $\mathbb{R}^2$?
- One way: find triangulation of convex hull of P, & define f linearly on each triangle.





- Picture these as representing mountaineous regions:

Case ① - mountain ridge at q

Case ② - valley at q

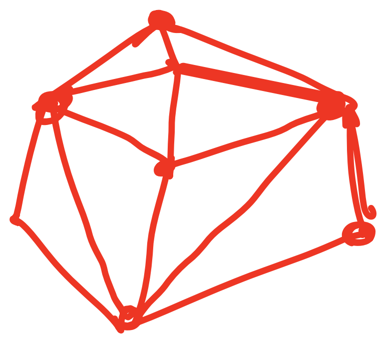
- Which triangulation is better? (No one answer)

Aesthetically, might say ① is better as q determined by nearby points ~ no long thin triangles ~ no small angles

- Will construct triangulations with (lexic.) larger angles.
- In order to compare the angles in two triangulations, we need to know that they have same no. of triangles.

<u>Theorem</u>

Let P be set of n points in plane (not all collinear) & suppose the convex hull of P has k vertices. Then any triang of P has $2n-2-k$ triangles & $3n-3-k$ edges.



<u>Proof</u> m = no of triangles

E = no of edges

= no of edges on 1 triangle (<u>k</u> since one for each outer edge)

+ no on 2 triangles (ℓ)

So $\quad 3m - \ell = E = k + \ell$

$\Rightarrow$ $\circledast$ $2E = 3m - \ell + k + \ell = 3m + k$

Now use Euler $V - E + F = 2$

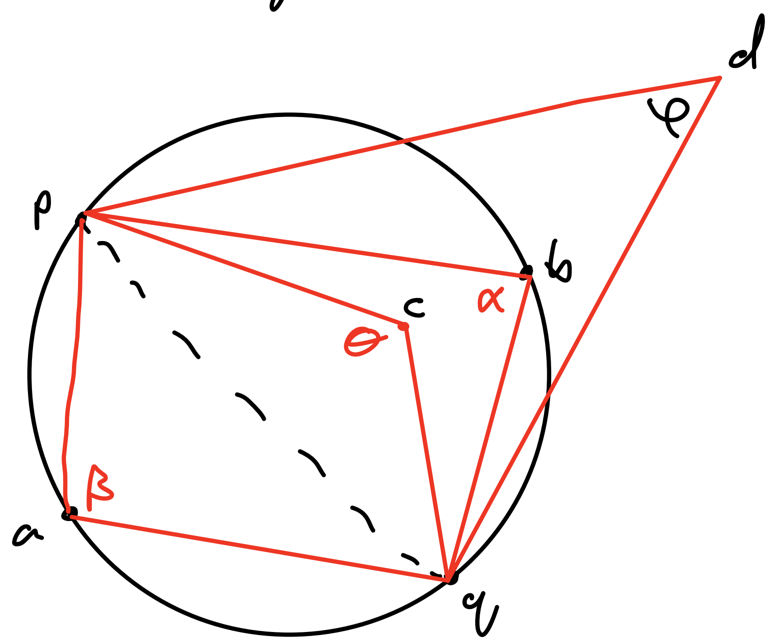
$n - E + (m+1) = 2$.

Subbing $\circledast$ into Euler gives answer. $\square$

- Therefore any triangulation T of P has m triangles & so $3m$ angles.
- Order these in a sequence $\alpha(T) = (\alpha_1 \leq \alpha_2 \leq \ldots \leq \alpha_{3m})$
- Define $\alpha(T) < \alpha(T')$ (lexicogr.) if $\exists i$ s.t. $\forall j < i$ $\alpha_j = \alpha'_j$ but $\alpha_i < \alpha'_i$.

- Angle-optimal triangulation is one which is maximal wrt to this ordering.
- We will not quite find angle-opt. triangulations but slightly weaker notion of legal / Delaunay triangulations.

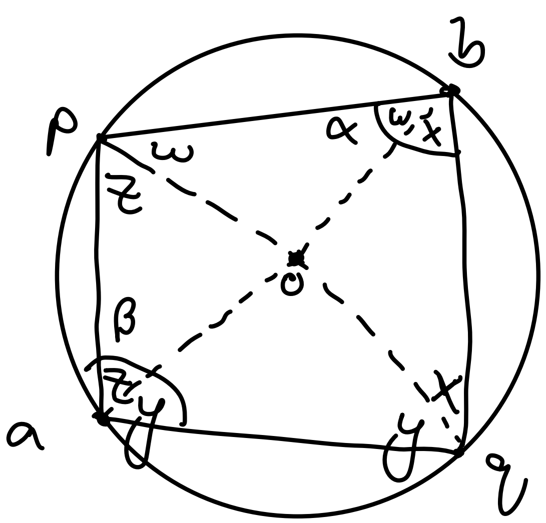
Legal Triangulations



- In circle, $\alpha + \beta = 180°$.
- If c lies inside circle, $\theta > \alpha$.
- For d outside, $\alpha > \varphi$.

Proof that $\alpha + \beta = 180°$

Triangles connected to centre o are isosceles so get angles as drawn.

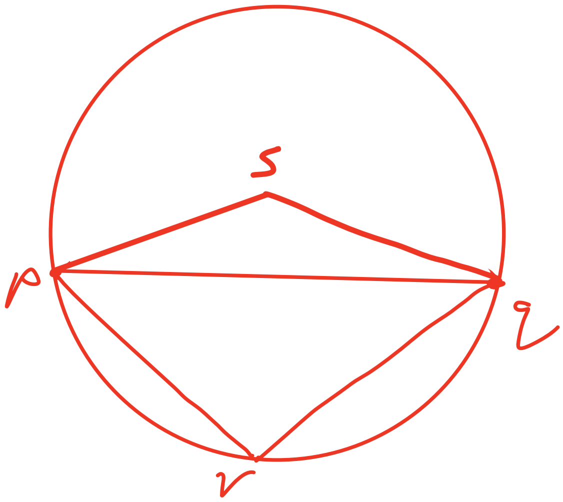


Then $2(w+x+y+z) = 360$.

So $w+x+y+z$

$$\underbrace{w+x}_{\alpha+\beta} + \underbrace{y+z}_{180°}$$

- Consider edge $\vec{pq}$ in triangulation.
- If $\vec{pq}$ not on boundary, it lies on 2 triangles pqr & pqs.



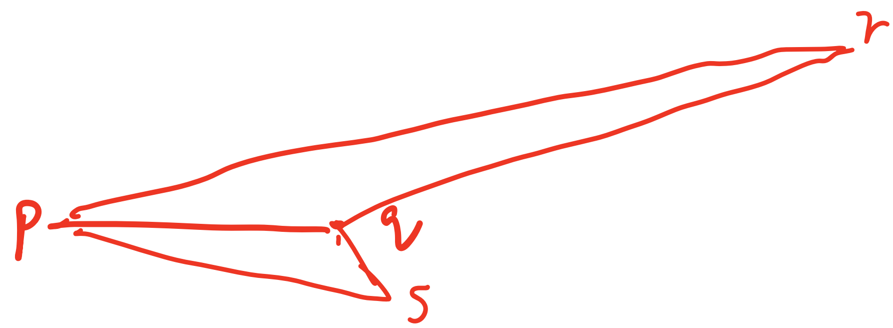
- Say pq is illegal if s lies strictly inside circle circumscribing pqr (i.e. $\angle psq > 180 - \angle prq$)

(Equiv. r lies inside circle containing pqs.)

- Otherwise pq is legal.
- A legal triangulation is one in which all edges are legal.

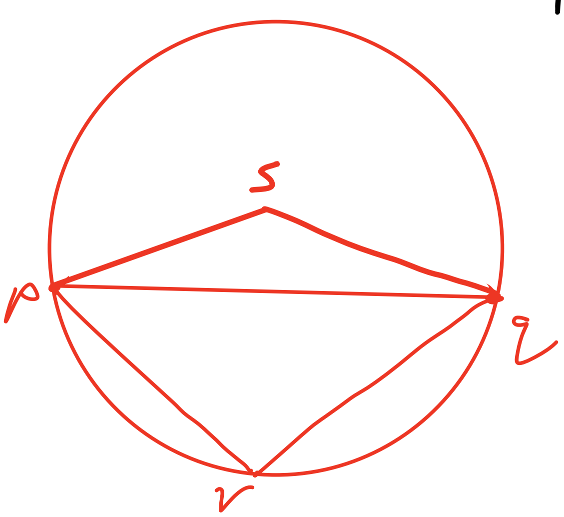
Note (added after lecture)

- In the alg. may encounter edges like pq below

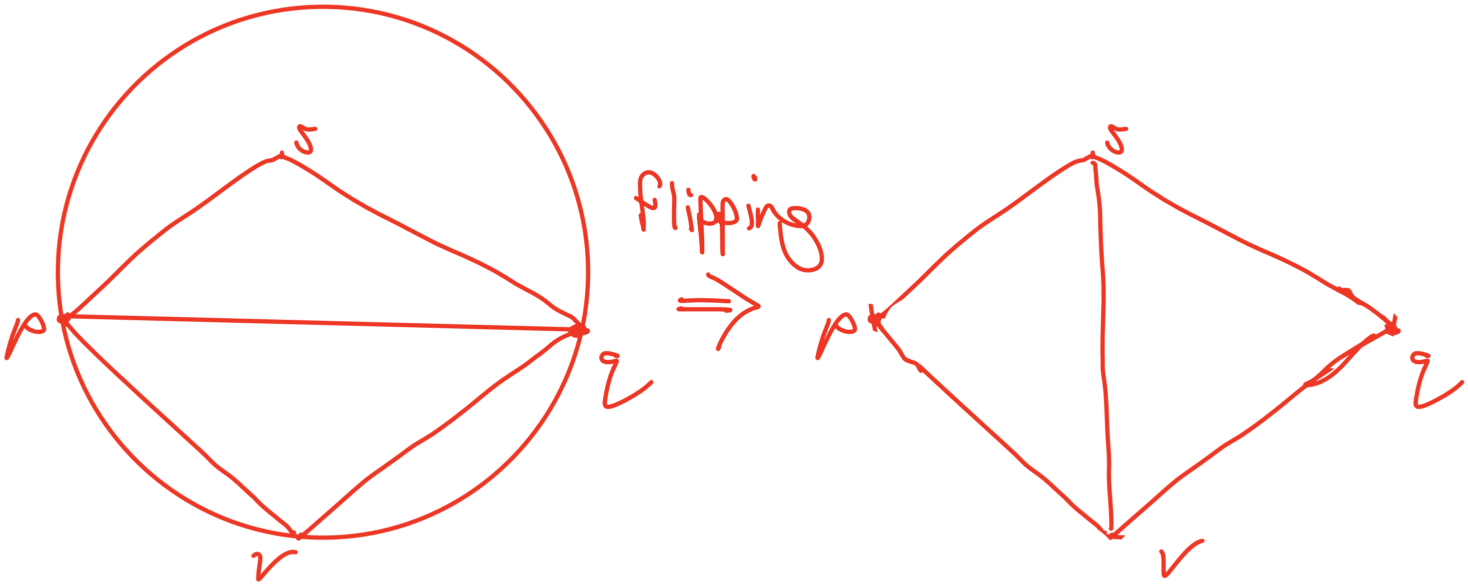


where the quadrilateral pqrs is not convex.

- Such a pq is legal since if s lies inside pqr as below, pqrs must be convex.



- Given an illegal edge $\vec{pq}$, we can <u>flip it</u> to an edge $\vec{rs}$ giving a new triangulation



lemma) Let T have illegal edge $\vec{pq}$. Then the flipped edge $\vec{rs}$ is legal in the new triangulation T' and $\alpha(T) < \alpha(T')$.

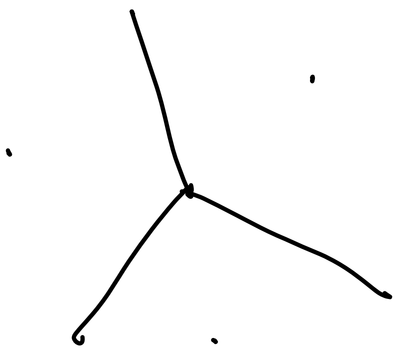
- See lemma 10.3 & proof in E-Learning.

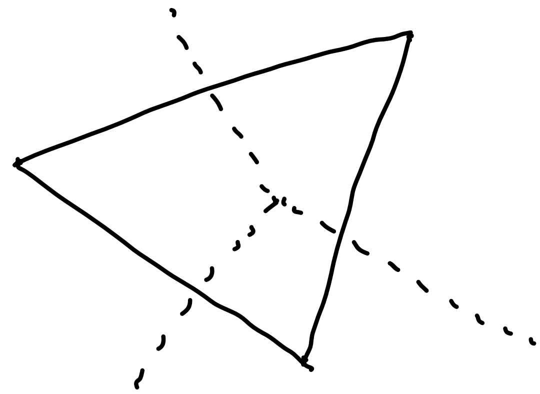
- Hence one can <u>legalise triangulations</u> by flipping illegal edges, & this is what our algorithm will do.

Alternative approach - Delauney Triangulation

- From P form Voronoi diag $V(P)$ last wk

$V(P)$





$D(P)$

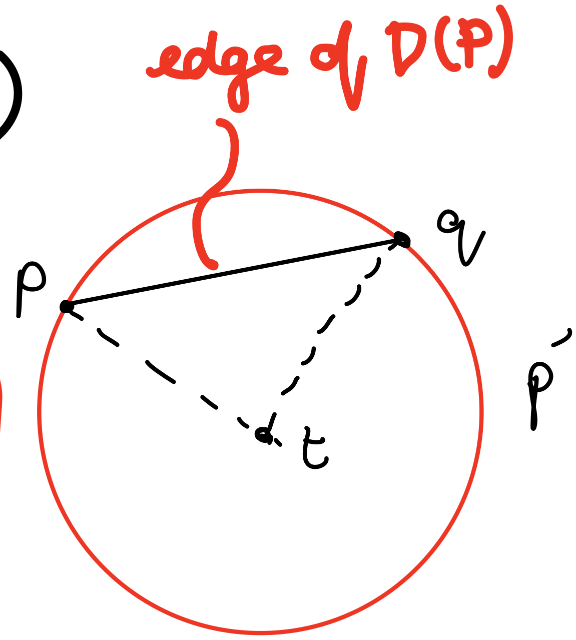
Delauney Graph $D(P)$ is dual graph to $V(P)$: it has

- same vertices as P (one for each face of V(P))
- an edge p to q $\Leftrightarrow$ V(p) & V(q) share a common edge.
- Faces of D(P) correspond to vertices of V(P).

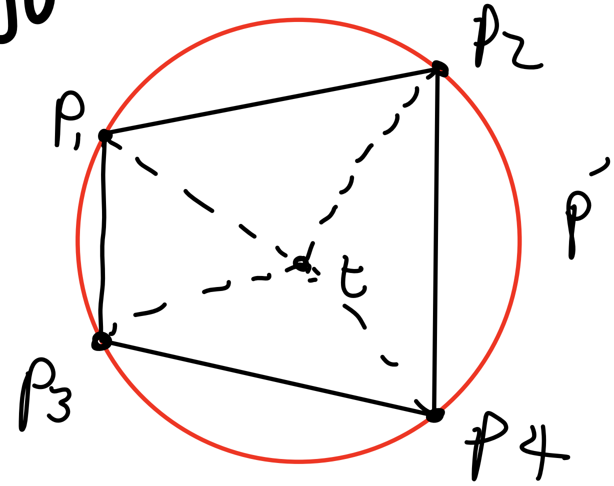
(See Fig 10.7 for example)

Can describe $D(P)$ in elementary terms

- From last week, $U(p)$ & $U(q)$ share an edge $\iff \exists t$ st $d(t,p) = d(t,q) < d(t,p')$ all other $p' \in P$.
(ie. p, q lie on boundary of circle with no other points of P inside)



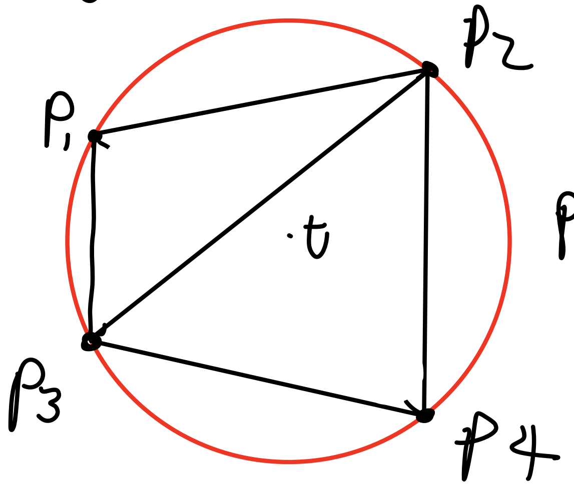
- Faces of $D(P)$ are polygons transcribed on circles with centre t, having same distance to each vertex (at least 3) & no points of P in interior.



Def) A De Launey Triangulation is any triangulation of Delauney graph $D(P)$.

- So De Launey Triang. is obtained by triangulating these polygons.

- By its construction, it has the following property:



(*) Let $p_i p_j p_k$ be a triangle in a De Launey Triangulation.
Then the circle transcribing triangle contains no points of P in its interior.

- This implies that each edge in De Launey Triangulation is legal.

Th 10.7) De Launey Triangulations $\equiv$ Legal Triangulations

from E-Learning

Towards Algorithm

- Could calculate V. diagram of P, calculate its dual & triangulate it.

- We will use legalisation.

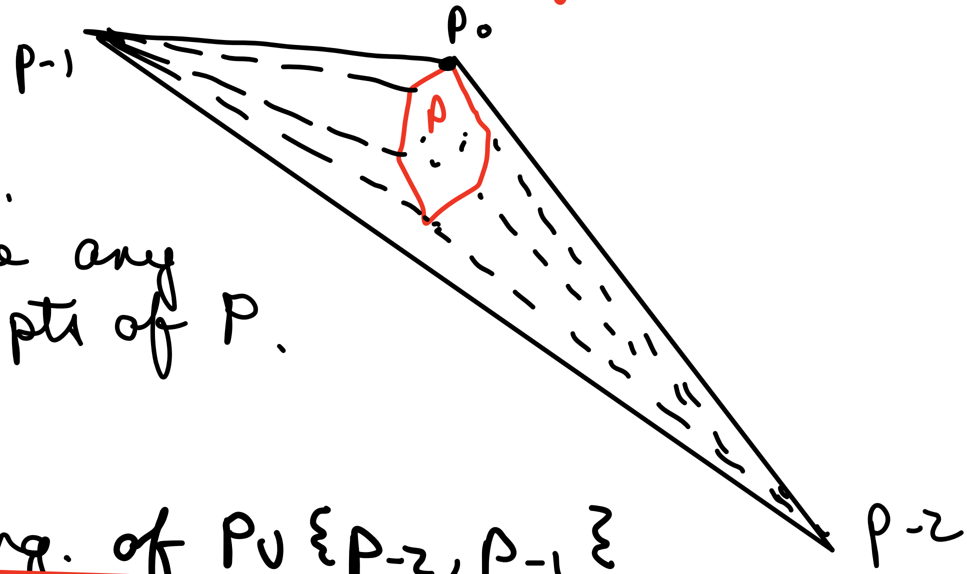
Naive version

- Find any triangulation of convex hull of P.
- Go through edges, flipping them if illegal.
- Process terminates since flipping illegal edges increases position of triang. wrt. lex ordering & only finitely many triangulations of P.

Randomised incremental algorithm

Step 1

- p_0 lex. max pt of P.
- Find pt p_{-1} (above left) & p_{-2} (below right) such that:



- all pts of P live in triangle $p_{-1} p_0 p_{-2}$.
- p_{-1}, p_{-2} don't lie inside any circle defined by 3 pts of P.

Then a legal triang. of $P \cup \{p_{-2}, p_{-1}\}$ consists of a legal triangulation of P

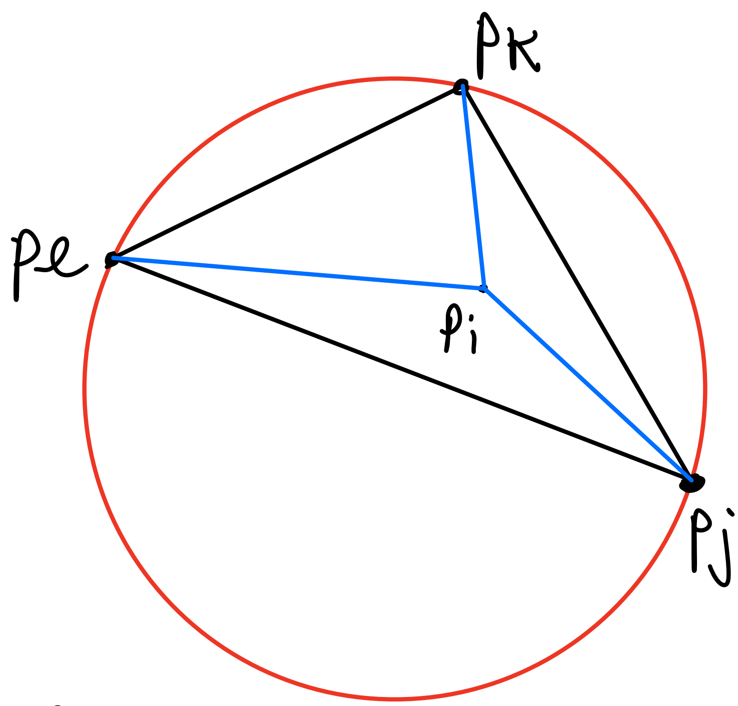
+

- an edge from p_{-1} to each pt on left boundary
- — — — — — p_{-2} — — — — — . right — — —

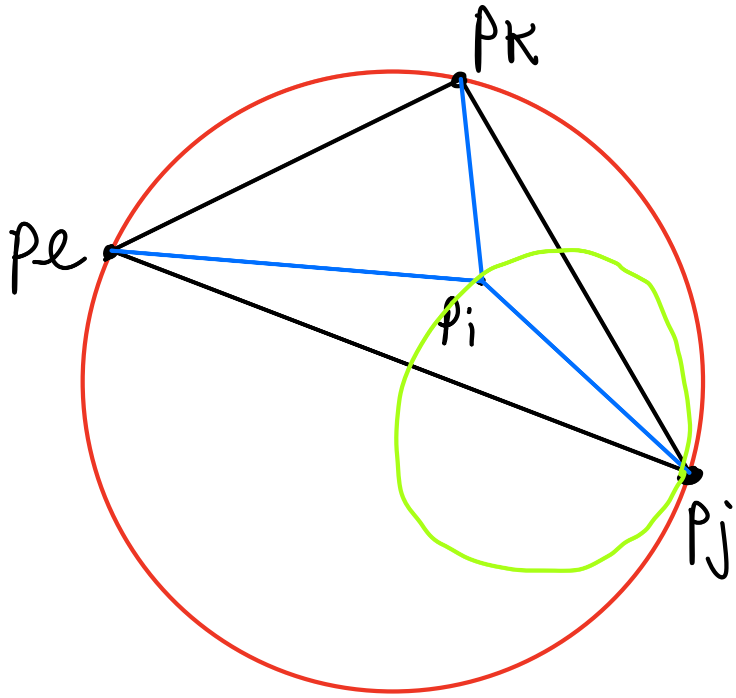
Step 2 Suppose we have legal triangulation T_{i-1} of
$P_{i-1} = \{p_{-2}, p_{-1}, p_0, ----, p_{i-1}\}$ for $i \geq 1$.

- Order of p_i is randomised.
- Use search structure D_{i-1} to find a triangle or edge in T_{i-1} where p_i lies.

- Create new triangles as depicted.



- Now, each edge $p_i p_e, p_i p_j, p_i p_k$ is legal : eg. $p_i p_j$.



- Draw circle homothetic to larger one with chord $p_i p_j$.
- Its centre is equidistant to p_i, p_j & contains no pts of P_i except these
  $\Rightarrow p_i p_j$ is edge of $D(P_i) \Rightarrow$ legal !

- It may happen some old edges become illegal - we have to repeat & legalise these by flipping them.

- See animation in E-learning.

Step 3

Remove p_{-2}, p_{-1} & all edges connected to them.

Search structure

Oriented graph - leaves are triangles of triangulation.

- inner nodes are triangles of prev. stages of triangulations.

(See Fig 10.17, 10.18)

Complexity: expected time $O(n \log n)$.