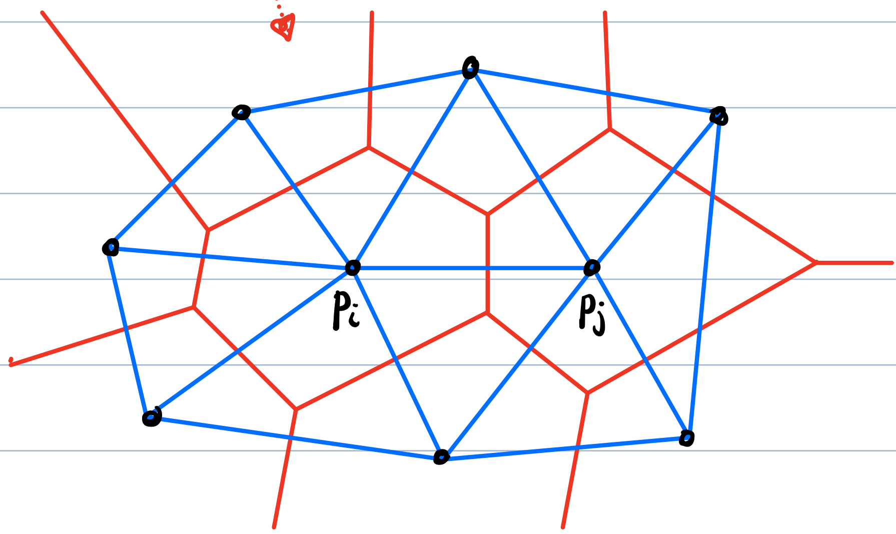
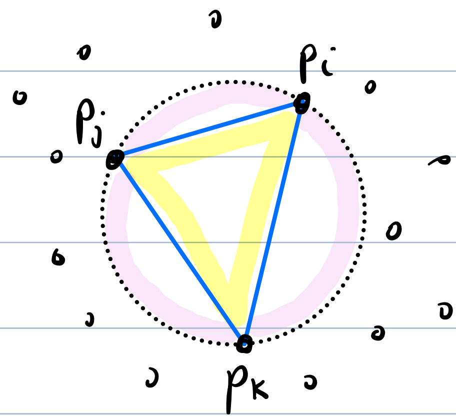
CMSC 754 - Computational Geometry
Lecture 12: Delaunay Triangulations (Construction)

Last lecture: - Delaunay triangulation + properties
- Given a set $P = \{p_1, \ldots, p_n\}$ of sites, DT(P) is the dual of Vor(P)



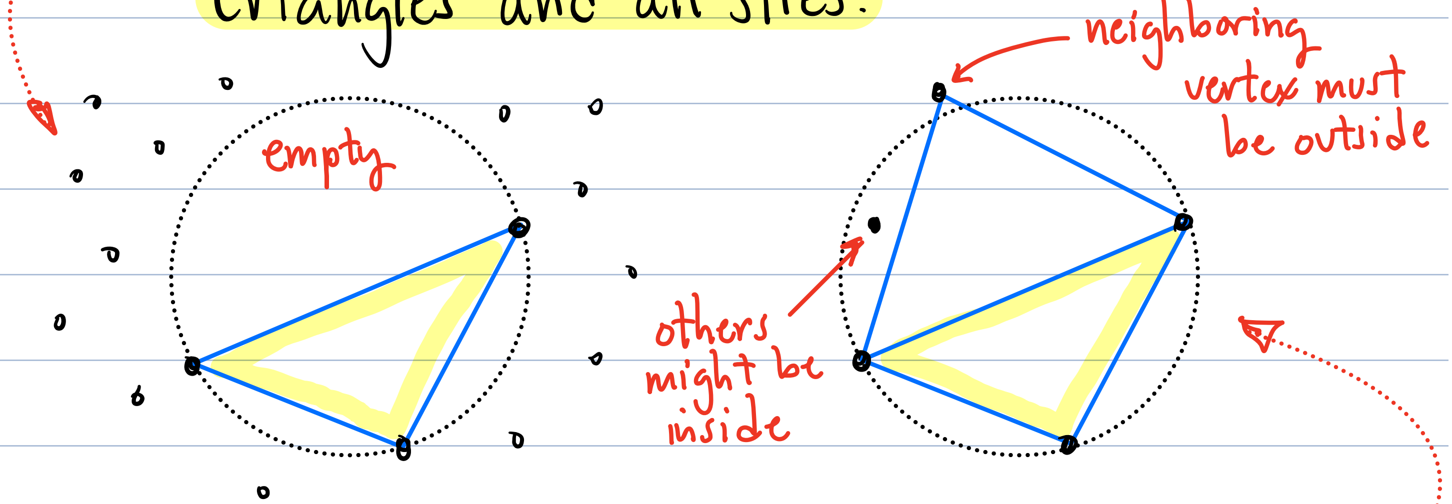


Circumcircle Property:

$\Delta p_i p_j p_k \in DT(P)$ iff circumcirle of p_i, p_j, p_k contains no sites

Local/Global Delaunay:

- A triangulation is globally Delaunay if circumcircle property holds for all triangles and all sites.



- A triangulation is locally Delaunay if circumcle property holds the vertices of every pair of adjacent triangles.

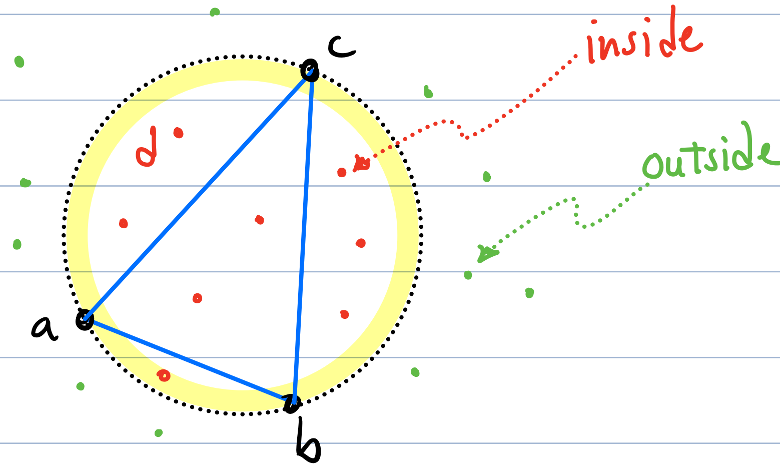
Does it matter? No.

Thm (Delaunay): A triangulation is globally Delaunay iff it is locally Delaunay.

(See lecture notes/text for proof)

Incircle Test: Given points a, b, c + $d \in \mathbb{R}^2$, does d lie in circumcircle of Δabc?

(Assume a, b, c given in CCW order)



inCircle(a, b, c; d): d is inside if

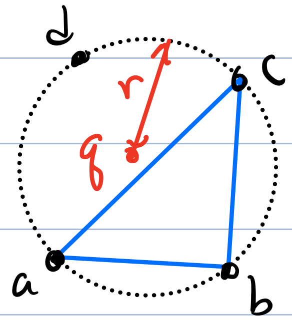
$$\det \begin{pmatrix} a_x & a_y & a_x^2 + a_y^2 & 1 \\ b_x & b_y & b_x^2 + b_y^2 & 1 \\ c_x & c_y & c_x^2 + c_y^2 & 1 \\ d_x & d_y & d_x^2 + d_y^2 & 1 \end{pmatrix} > 0$$

Obs:

- This is an orientation test in $\mathbb{R}^3$
- Generalizes to any dimension
- Computable in $O(1)$ time in any fixed dimension.

Why? Consider boundary case $\rightarrow$ cocircular

Center $q = (q_x, q_y)$ radius $= r$



$$\Rightarrow (a_x - q_x)^2 + (a_y - q_y)^2 = r^2$$

$$\Rightarrow \boxed{-2q_x}\, a_x \boxed{-2q_y} \cdot a_y + \boxed{1} \cdot (a_x^2 + a_y^2)$$

$$+ \boxed{(q_x^2 + q_y^2 - r^2)} = 0$$

Same applies to b, c, d $\Rightarrow$

$$\begin{pmatrix} a_x & a_y & a_x^2 + a_y^2 & 1 \\ b_x & b_y & b_x^2 + b_y^2 & 1 \\ c_x & c_y & c_x^2 + c_y^2 & 1 \\ d_x & d_y & d_x^2 + d_y^2 & 1 \end{pmatrix} \begin{pmatrix} -2q_x \\ -2q_y \\ 1 \\ q_x^2 + q_y^2 - r^2 \end{pmatrix} = 0$$

A linear combination of columns is identically 0

$\Rightarrow$ column vectors are lin. dependent

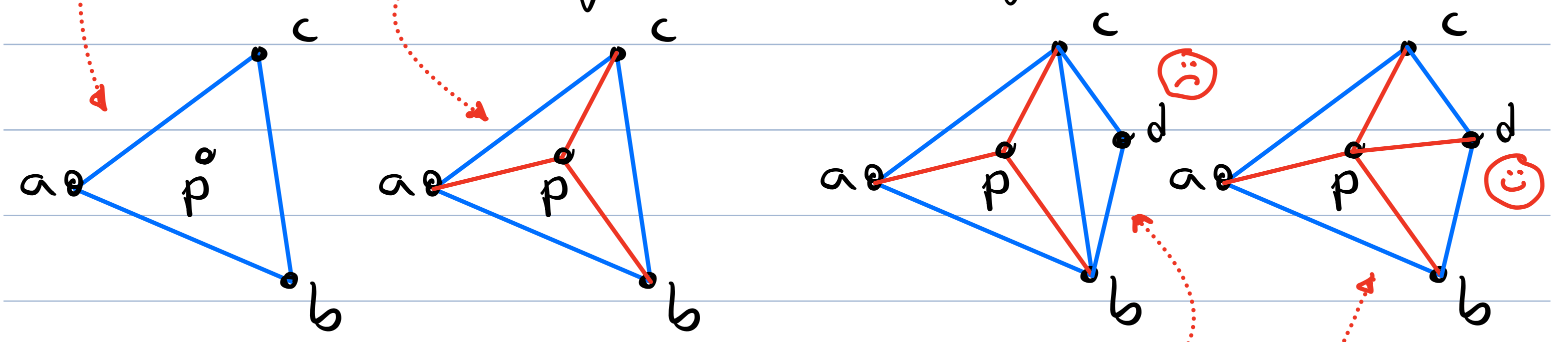
$\Rightarrow$ det of matrix is 0

(Randomized) Incremental Construction:

- Add sites one-by-one in random order + update the triangulation after each.

① Find triangle Δabc containing the new site p.

② Add edges connecting p to a, b, c



③ Check neighboring triangles for violations of local Delaunay

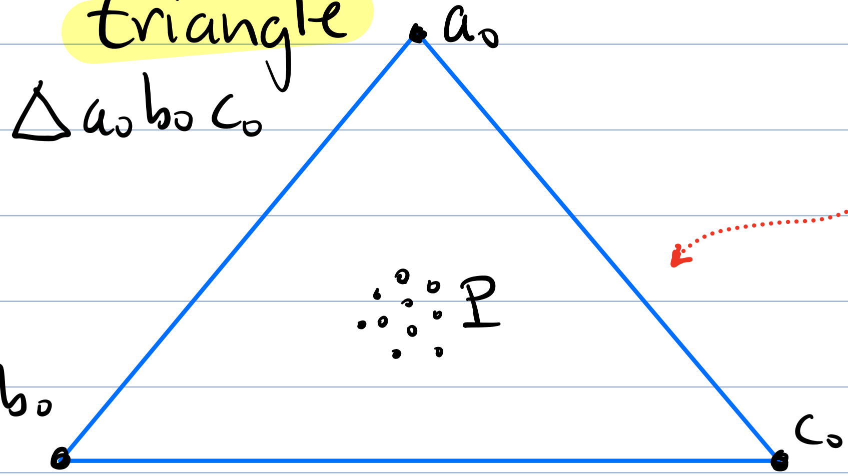
④ Apply edge flips to correct these

⑤ Repeat until local Delaunay for all neighbors

Sentinel sites:

- If new site is not in convex hull – it's not in any triangle!

- Fix: Enclose points in a HUGE triangle



How huge? No circumcircle from P should contain a_0, b_0 or c_0

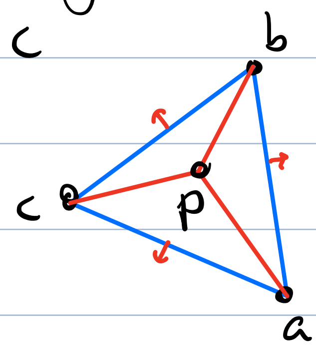
(see text for how)

Build-Delaunay($P = \{p_1, \ldots, p_n\}$)

- Create sentinel triangle $\Delta a_0 b_0 c_0$ containing P
- Randomly permute P
- for $i = 1$ to n Insert(p_i)

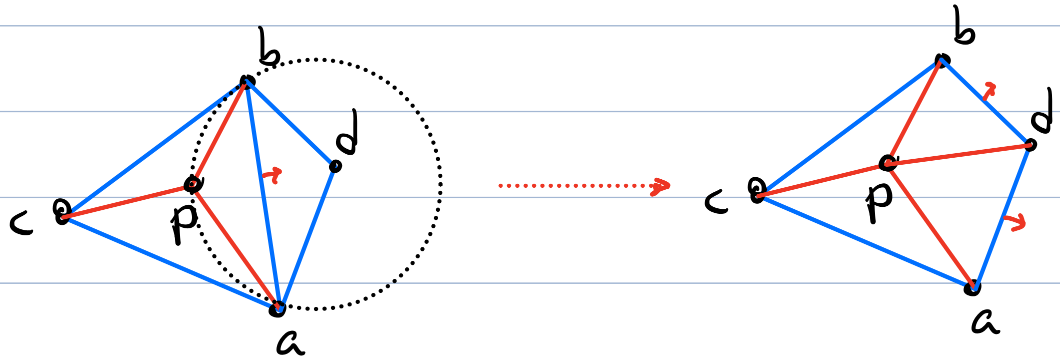
Insert (p):

- Find $\triangle abc$ containing p
- Add edges pa, pb, pc
- SwapTest (ab)
- " (bc)
- " (ca)

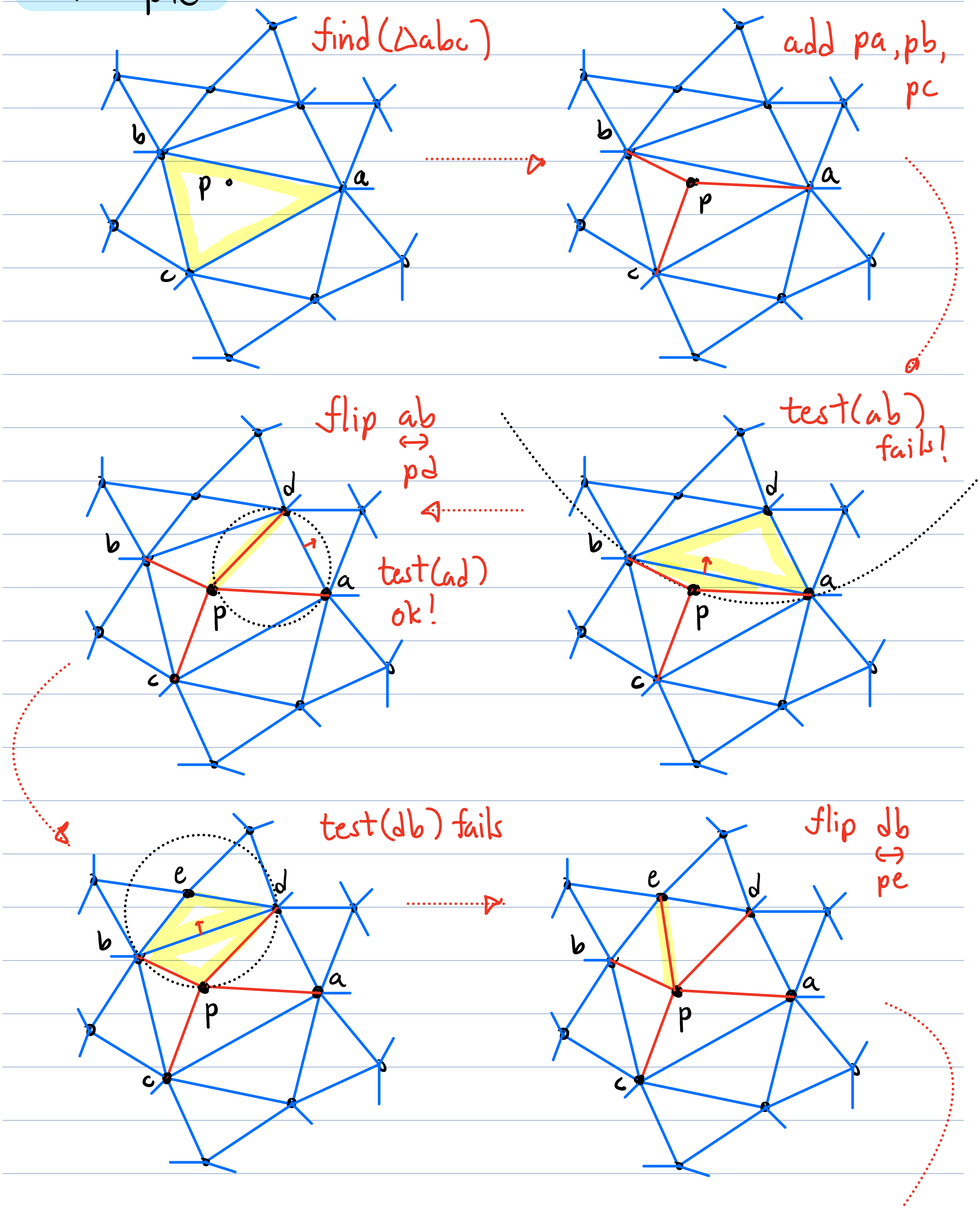


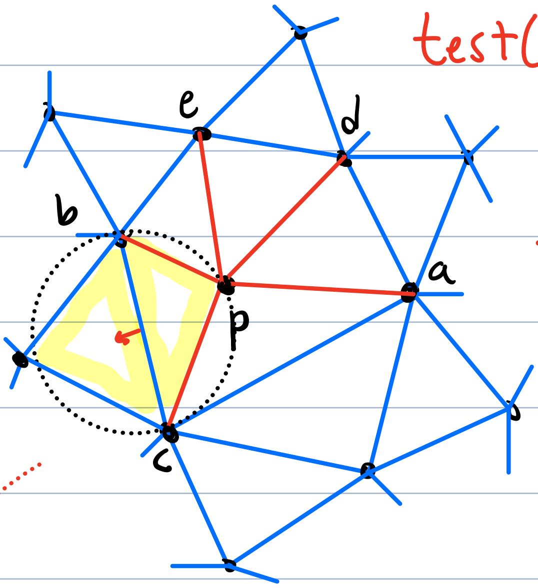
SwapTest (ab):

- if (ab is edge of external face)
  return
- d ← vertex opposite p on ab
- if (inCircle(p, a, b, d))
  - flip edge ab (for pd)
  - SwapTest (ad)
  - SwapTest (db)

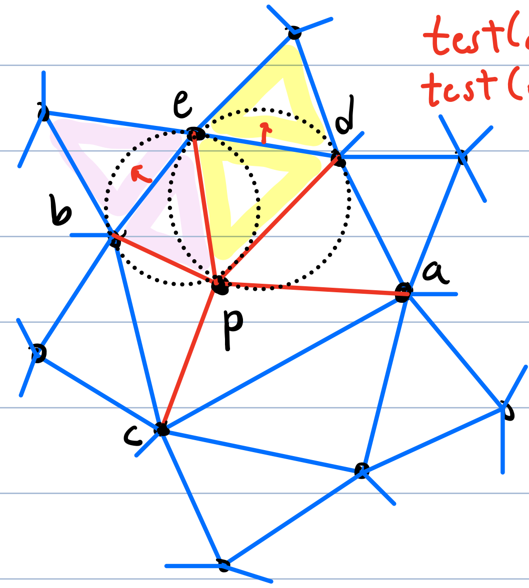


Example:

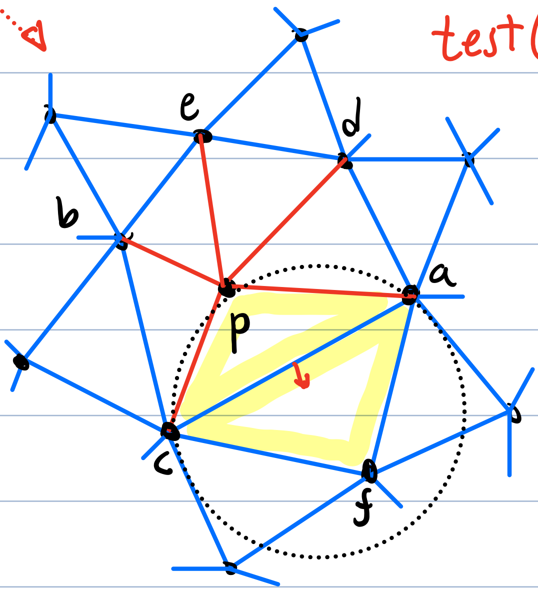

find(Δabc)
add pa, pb, pc
flip ab ↔ pd
test(ad) ok!
test(ab) fails!
test(db) fails
flip db ↔ pe



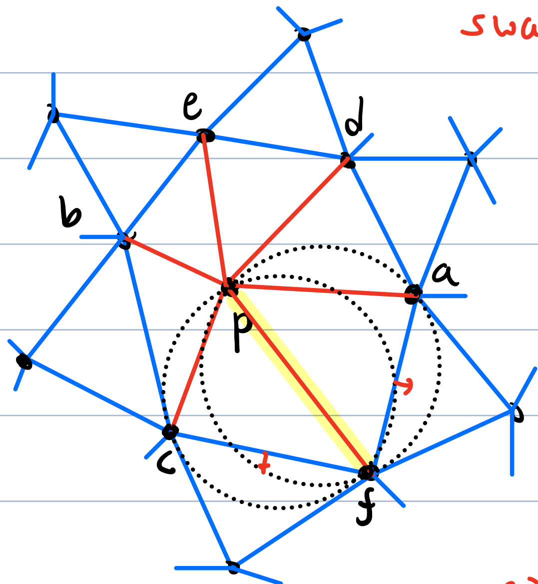
test(bc) ok!



test(de) ok!
test(cb) ok!



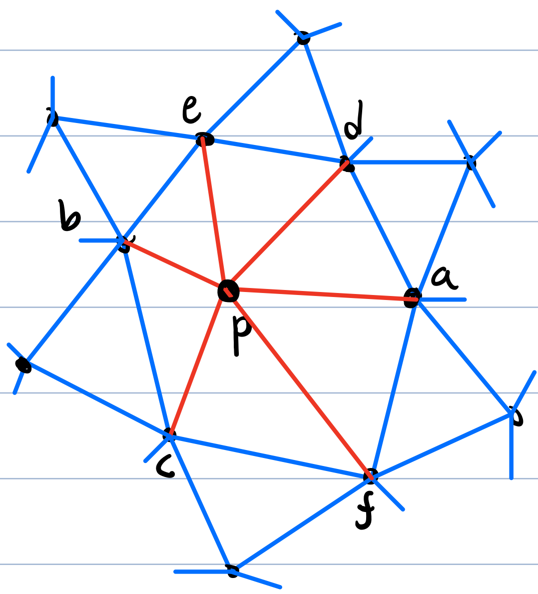
test(ca) fails



swap ca $\leftrightarrow$ pf

test(cf) ok!
test(fa) ok!

Done!



Final

Note: All new edges are incident to p

Correctness:

- Only triangles that could violate local Delaunay are incident to p, + we check all
- By Delaunay's Thm, local $\Rightarrow$ global Delaunay

Running time:

- for each insertion $p_1 \ldots p_n$
- find triangle containing $p_i \longrightarrow O(\log n)$
- swaptests + edge flips $\longrightarrow O(1)$

in expectation

- Total: $O(n \log n)$

Lemma: The expected update time (swaptests + edgeflips) is $O(1)$

Proof: (Backwards analysis)

- Update time $\sim$ degree of p in final Δ-tion
- Every pt is equally likely to be last
- $\sum_i \deg(p_i) = 2(\#\text{edges}) \leq 2(3n) = 6n$
- Expected time $\sim$ Average degree $\leq \frac{1}{n} \cdot 6n = 6$

$\square$

Point-Location:

Bucketing:

called a "bucket"

- Each triangle maintains future sites in this triangle
- To locate a point, just get its bucket id.
- When an edge flip is performed rebucket the affected sites

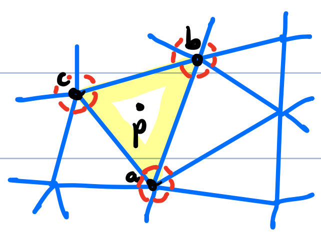
Lemma: Let p be any site. The probability that p is rebucket as result of i^{th} insertion is $\leq 3/i$

Proof: (Backwards Analysis)

- Sites are rebucketed only if in new triangle
- All new triangles are incident to last site
- Each site is equally likely to be last
- Prob(p is rebucketed) [let $p \in \Delta abc$]

$\leq$ Prob(a, b, or c was last inserted)

$\leq 3/i$ $\square$



Lemma: Total time for rebucketing is $O(n \log n)$ in expectation

Proof: Rebucket time (expected)

$$= \sum_{p \in P} \sum_{i=1}^{n} 1 \cdot \text{Prob}(p \text{ was rebucketed in } i^{th} \text{ insertion})$$

$$\leq \sum_{p \in P} \sum_{i=1}^{n} 3/i \approx \sum_{p \in P} 3 \cdot \ln n \quad \left(\text{Harmonic series}\right)$$

$$= 3 n \ln n$$

$$= O(n \log n) \quad \square$$