

Set $m \approx n \Rightarrow$ $O(1)$ expected query time
 $O(n)$ space, preproc time
 $(O(1)$ expected insert/delete time)

Q: can we get good worst case query time?

assume static

Alternative query analysis:

$$\begin{aligned} & E \left[\text{total \# colliding pairs } (x, y) \in S \times S \right] \\ &= E \left[\sum_{y \in S} (\text{\# collisions with } y) \right] \\ &= \sum_{y \in S} E[\text{\# collisions with } y] \\ &= O\left(n \cdot \frac{n}{m}\right) \end{aligned}$$

$$\text{Set } m = cn^2 \Rightarrow E(\text{total \# colliding pairs}) \leq O(1/c)$$

$$\begin{aligned} \left(\Pr[X \geq B] \leq \frac{E[X]}{B} \right) &\Rightarrow \Pr(\text{total \# colliding pairs} \geq 1) \\ &\leq \frac{1/c}{1} = \frac{1}{c} \\ &\Pr[\text{perfect hash}] \geq 1 - \frac{1}{c} \end{aligned}$$

$$\Rightarrow \begin{cases} O(n^2) \text{ space} \\ O(1) \text{ query time worst-case} \\ \cdot O(n^2) \text{ preproc time expected (repeat w. success)} \end{cases}$$

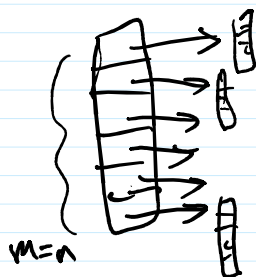
can we reduce space back to $O(n)$?

Final Method: (Fredman, Komlós, Szemerédi '84)

idea - bootstrap! ("2-level hash table")

idea - bootstrap!

("2-level hash table")



perfect hashing
for each bin $A[i]$, store it in structure
with $O(1)$ worst-case query time
 $O(|A[i]|^2)$ space

Analysis:

query time $O(1) + O(1) = O(1)$ worst-case.

expected space $O\left(m + \underbrace{\sum_i |A[i]|^2}_{\text{total \# of colliding pairs}}\right)$

$= E(\text{total \# of colliding pairs})$

$= O\left(m + \frac{n^2}{m}\right)$

Set $m \approx n$

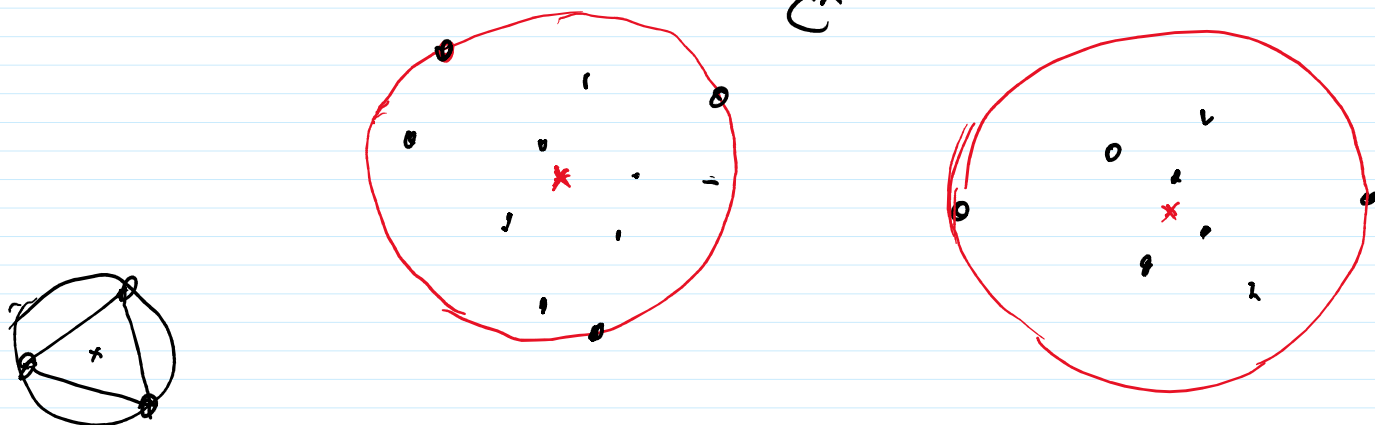
$= \boxed{O(n)}$
 $\boxed{O(1)}$
 $\boxed{O(n)}$

~~expected space~~
worst-case
query time

expected preproc
time

Problem Given set S of n points in 2D,

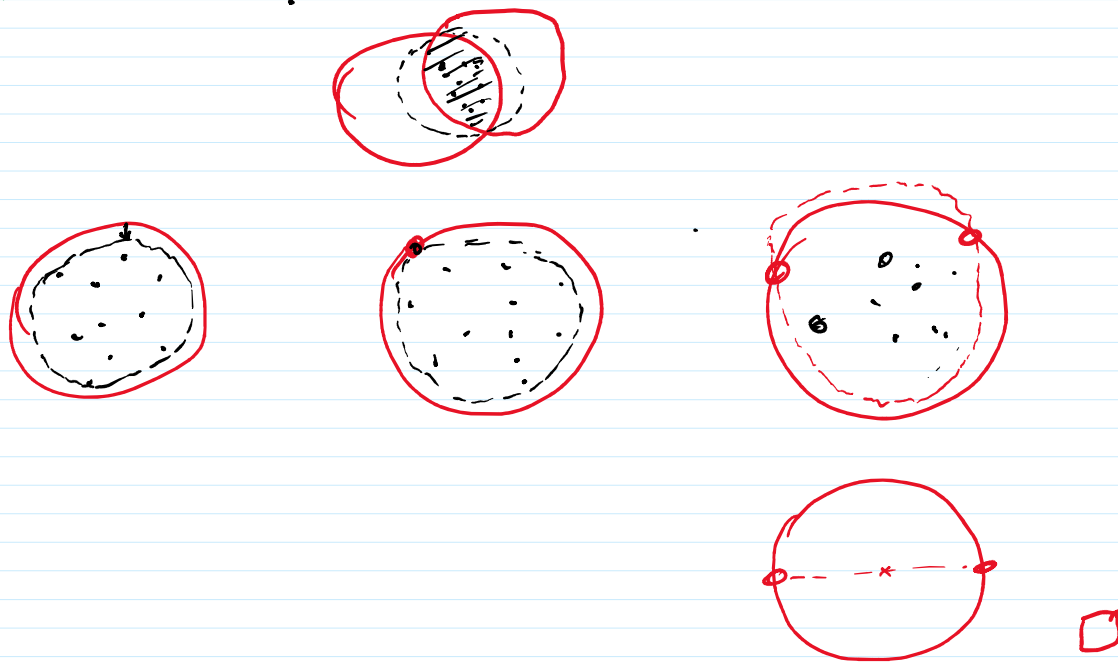
Problem Given set S of n points in $\mathbb{L}^2$,
compute the min circle enclosing S .



Obs C^* is unique & must have 3 pts on boundary
& 2 pts as diameter

$\Rightarrow O(n^4)$ time alg'm

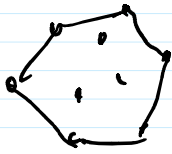
Pf Sketch: By contradiction.



ideas - brute force $\Rightarrow O(n^4)$ time

- compute convex hull $\leftarrow O(n \log n)$ time
but then what?

but then what?



- compute a "Voronoi diagram" $\leftarrow O(n \log n)$
 $\Rightarrow O(n \log n)$ time

Megiddo & Dyer '83: $O(n)$ time
but complicated!

$O(n)$ randomized alg'm that is simpler...

Digression...

Subproblem (in 1D) Given n numbers $S = \{x_1, \dots, x_n\}$,
find the min

"incremental"
Standard alg'm:

0. randomly permute $x_1, \dots, x_n$
1. $ans = \infty$
2. for $i = 1$ to n do
3. if $x_i < ans$
4. $ans = x_i$ (*)

$O(n)$ time

Question: how many times line 4 (*) is done?

worst case analysis: n times e.g. 10, 9, 8, 7, 6, 5, ...

rand. analysis:

rewrite alg'm:

$\min(S)$: // backwards version

$\text{min}(S)$: // backwards version

0. if $S = \emptyset$ return ∞
1. pick $x \in S$ randomly
2. $\text{ans} = \text{min}(S - \{x\})$
3. if $x < \text{ans}$ then
4. $\text{ans} = x$ (*)
5. return ans

for any fixed S ,

$$\begin{aligned}\Pr[(*) \text{ is done}] &= \Pr[x = \text{min}(S)] \\ &= \frac{1}{n}\end{aligned}$$

5, 3, 1, 2, 4
 ↑
 no

5, 2, 3, 4, 1
 ↑
 yes

Expected total # of times (*) is done

$$T(n) = T(n-1) + \underbrace{\frac{1}{n} \cdot 1 + \left(1 - \frac{1}{n}\right) \cdot 0}$$

$$\Rightarrow T(n) = T(n-1) + \frac{1}{n}$$

$$\begin{aligned}\Rightarrow T(n) &= \frac{1}{n} + \frac{1}{n-1} + \frac{1}{n-2} + \dots + \frac{1}{2} + \frac{1}{1} \\ &= \boxed{O(\log n)} \quad \text{harmonic number}\end{aligned}$$