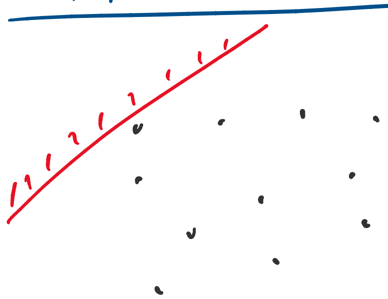
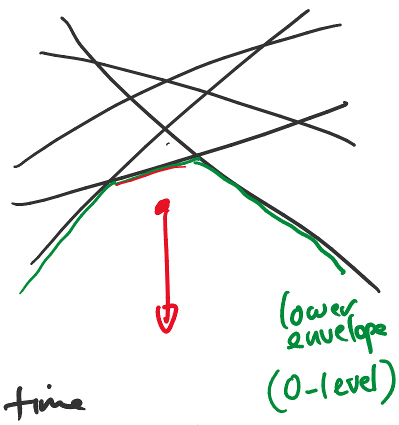
October 12, 2021 10:21 AM

Halfspace Range Reporting in 2D or 3D



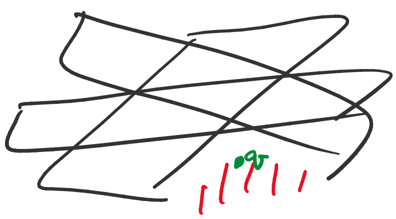
dual $\Longleftrightarrow$



$k=0$, emptiness:

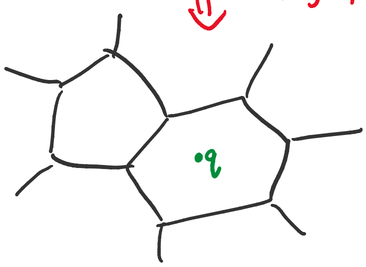
2D: $O(n)$ space, $O(\log n)$ time
by binary search

3D: $O(n)$ space, $O(\log n)$ time
by 2D point location



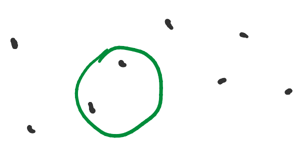
lower envelope or convex polyhedron

$\Downarrow$ xy-projection



appl: 2D circular range report.

$\Downarrow$ 3D halfspace range report.



Reporting?

History:

		space	time
2D	Chazelle-Guibas-Lee '85	n	$\log n + k$
	...-Preparata '86	$n \log^2 n \log\log n$	$\log n + k$

2D	· Chazelle-Guibas ...		
3D	Chazelle-Preparata '86	$n \log^2 n \log\log n$	$\log n + k$
	Aggarwal-Hansen-Leighton '90 ← use separators, large preproc.	$n \log n$	$\log n + k$
		$n \log n$	$\log n + k$
	→ C. '99	$n \log\log n$	$\log n + k$
	Afshani-C. '09	n	$\log n + k$

Shallow Cutting Lemma (Matoušek '92)

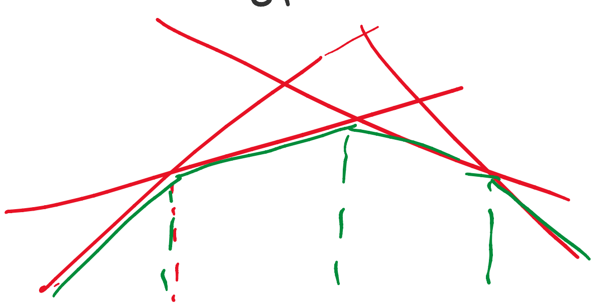
Given n lines in $\mathbb{R}^2$ (or n planes in $\mathbb{R}^3$),
can form $O(r)$ cells
each intersecting $O(\frac{n}{r})$ lines
s.t. all (n/r)-shallow pts are covered

$(\frac{1}{r})$-shallow cutting

pts with $\le \frac{n}{r}$ lines below

Pf idea: (ignore log factors)

1. take random sample $R \subseteq L$ w. prob. $\frac{r}{2n}$
2. return "vertical decomposition" of the lower envelope of R



By prev analysis, each cell intersects $\lesssim \frac{n}{r}$ lines w. good prob.

Fix an $(\frac{n}{r})$-shallow pt q.

$\Pr[\, q$ is not covered$\,]$
$\le \Pr[$ some line below q is in $R\,]$

q $\}\le\frac{n}{r}$ lines

$= \Pr[$ some line below q is in R $]$

$\le \frac{n}{r} \cdot \frac{r}{2n} = \frac{1}{2}$.

Repeat t times $\Rightarrow$ err prob $\le \frac{1}{2^t}$

set $t = 3\log n$ $\Rightarrow$ err prob $\le \frac{1}{n^3}$ for fixed q

$\Rightarrow$ overall err prob $\le \frac{1}{n}$.

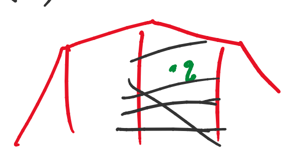
(work harder to avoid extra log ...). $\square$

Cutting Tree Method:

recurse

$$S(n) = O(r)\, S\left(\frac{n}{r}\right) + O(r)$$

$$Q(n) = \begin{cases} Q\left(\frac{n}{r}\right) + O(r) & \text{if shallow} \\ \cancel{O(n)} \leftarrow \text{by brute force} \quad O(rk) & \text{if not shallow } \Rightarrow k > \frac{n}{r} \end{cases}$$



r large const

$\Rightarrow$ $Q(n) = \boxed{O(\log n + k)}$.

$S(n) = \boxed{O(n^{1+\varepsilon})}$ $\leftarrow$ $O(\underbrace{C \cdot C \cdots C}_{\log_r n}\, n)$

$C^{\log_r n} = n^{\log_r C} = n^{\frac{\log C}{\log r}}$

Better Method (C.'99)

idea - hierarchy of cuttings (no recursion)

for $r = 1, 2, 4, 8, \ldots$

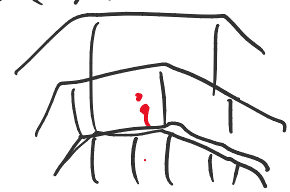
compute $(1/r)$-shallow cutting Γ_r

for each cell Δ,

store L_Δ = "conflict" list of lines/planes intersecting Δ

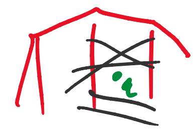
Space: for each r,

$$O\left(\sum_{\Delta \in \Gamma_r} |L_\Delta|\right) = O\left(r \cdot \frac{n}{r}\right) = O(n).$$



$\Rightarrow$ total $\boxed{O(n \log n)}$

Query alg'm, given pt q:



1. find r with $k \le \frac{n}{r} < 2k$
2. find $\Delta \in \Gamma_r$ containing q

   by 2D point location in $O(\log n)$ time

3. Search L_Δ by brute force

   time $O(|L_\Delta|) = O(\frac{n}{r}) = O(k)$

$\Rightarrow O(\log n + k)$

but don't know k in advance

idea - guess k!

try $k = k_0, k_1, k_2, \ldots$ with $k_i := 2^i \log n$

total query time $O\left(\sum_{i:\, k_{i-1} \le k} (\log n + k_i)\right)$

$$= O\left(\sum_{k_{i-1} \le k} k_i\right)$$

$$= \begin{cases} O(k) & \text{if } k > \log n \\ O(k_0) & \text{else} \end{cases}$$

$O(k_0) = O(\log n)$

$$= \boxed{O(\log n + k)}$$

An Improvement:

2/3

An Improvement:

for $r = r_1, r_2, \ldots$ where $\frac{n}{r_{i+1}} = \left(\frac{n}{r_i}\right)^{2/3}$

$\Rightarrow$ loglogn terms

...

store L_Δ in a linear-space DS with $O(|L_\Delta|^{2/3} + k)$ query time

$\Rightarrow$ space $\boxed{O(n \log\log n)}$

Query time:

line 1: find r_i s.t. $k \le \frac{n}{r_i}$, $k > \frac{n}{r_{i+1}}$

$k > \left(\frac{n}{r_i}\right)^{2/3}$

line 3: $O(|L_\Delta|^{2/3} + k)$

$= O\left(\left(\frac{n}{r_i}\right)^{2/3} + k\right)$

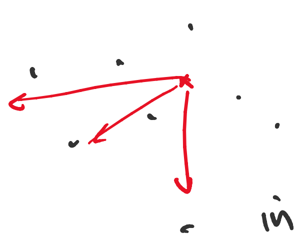
$= O(k)$

$\Rightarrow \boxed{O(\log n + k)}$

Rmks - space improvable to $O(n)$ (Afshani-C. '09)

- works for 3D orthogonal 3-sided (dominance)

$O(n)$ space, $O(\log\log U + k)$ time by 2D orth. pt loc.



in higher-D, size of lower envelope

shallow cutting $\rightarrow$ $O^*(n^{\lfloor d/2 \rfloor})$ space, $O(\log n + k)$ time

shallow partition $\rightarrow$ $O(n)$ space, $O^*(n^{1 - \frac{1}{\lfloor d/2 \rfloor}} + k)$ time