The Convex Hull Prob

15-750
2/27/19

(The sorting prob of CG)

Def $A \subseteq \mathbb{R}^d$ is convex if

Def1 $x, y \in A$ $\text{segment}(x,y) \subseteq A$

Def2 A is closed under convex combinations.

Def Convex Closure$(A) \equiv CC(A)$

= small convex set $\supseteq A$.

2 defs of Convex Hull

Def1 $CH(A) = \partial CC(A)$ (Boundry)

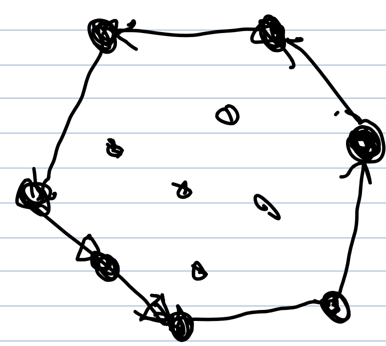
Def2 $CH(A) = CC(A)$

we will use Def1

For rest of lecture A is a finite set.

Thus in 2D

CH(A) = a simple closed polygon of a subset of A.
(say CCW order)



We will return the only the corners.

<u>Claim</u> $[a,b]$ is a side of CH(A) iff

1) $a, b \in A$, $a \neq b$
2) $\forall a' \in A$ either a' left of $[a,b]$ or $a' \in [a,b]$

Lower Bound

Claim: Sorting is reducible to CH.

Input: $x_1, \dots, x_n \in \mathbb{R}$

Return $CH((x_1, x_1^2), \dots (x_n, x_n^2))$

Note the points $P_i = (x_i, x_i^2)$ will be in sorted order.

This construction works in higher Dim

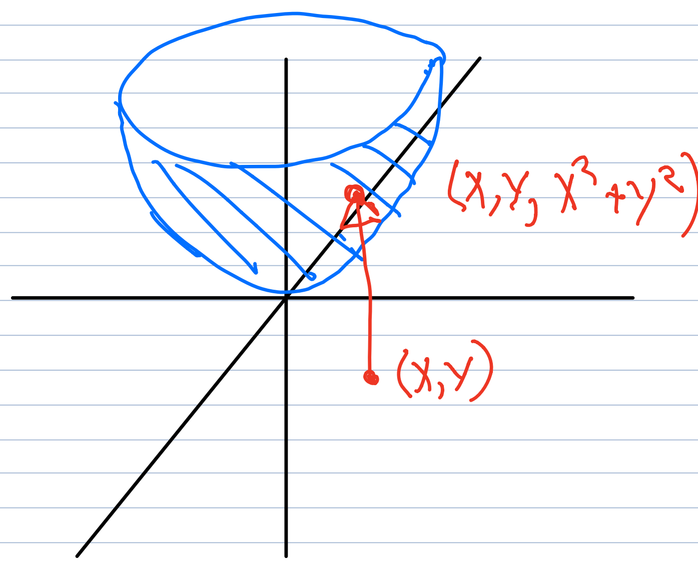
Let $P_1, \dots, P_n \in \mathbb{R}^2$

Let $\bar{P}_i = (P_x, P_y, P_x^2 + P_y^2) \in \mathbb{R}^3$

Then the lower $CH(\bar{P}_1, \dots, \bar{P}_n)$

is a Triangulated surface.

Called the Delaunay Triangulation



Consider two sorting alg

1) Merge Sort
2) Quick Sort $\equiv$ Treaps

We make these into CH algs.

$A = \{P_1, \dots, P_n\}$ $P_i = (x_i, y_i)$

Preprocess: Sort A by x-coordinate

Proc: MergeHull(A)

<u>if</u> $|A| = 1$ return P_1

<u>else</u> $CH_L = \text{MergeHull}(P_1, \dots, P_{n/2})$

$CH_R = \text{MergeHull}(P_{n/2+1}, \dots, P_n)$

STITCH(CH_L, CH_R)

Proc: STITCH(L, R)

Lowerbridge (L, R)

Let $\underline{a} \to a \to \bar{a}$

$\underline{b} \to b \to \bar{b}$

a = rightmost(L)

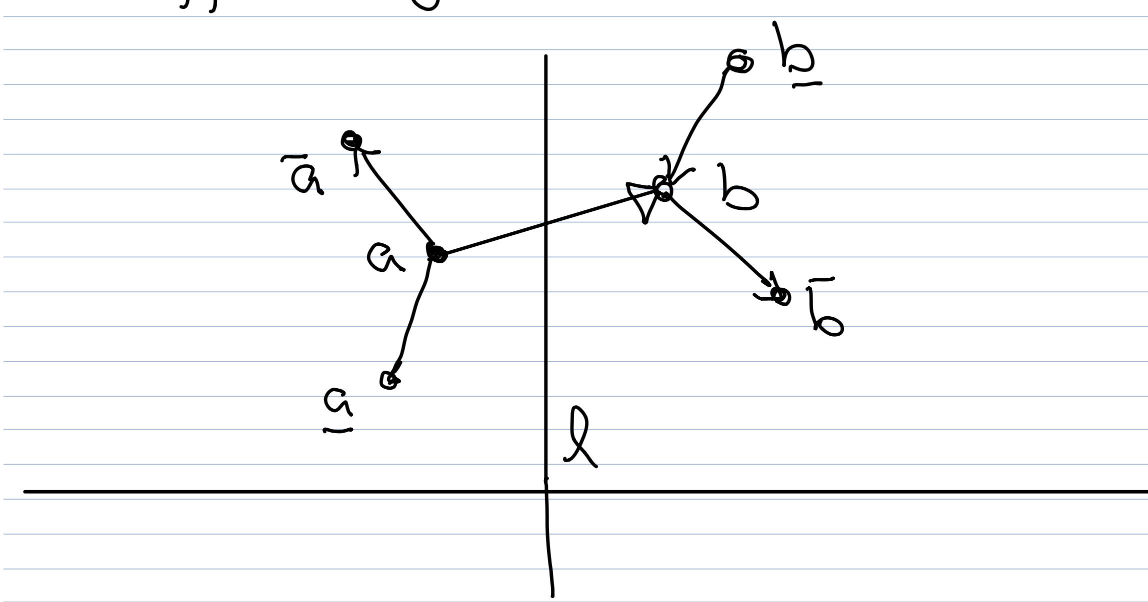
b = leftmost(R)

Repeat *) **) while possible

*) If $\underline{a}$ is right(a,b) set $a \leftarrow \underline{a}$

**) If $\bar{b}$ is right(a,b) set $b \leftarrow \bar{b}$

Upperbridge (L, R)



Termination

let ℓ be a vertical line between L & R.

*) Generates triangles $(\underline{a}, b, a)$

**) Generates triangles $(a, \bar{b}, b)$

1) The Δ_s have disjoint interiors
2) they are ordered by intersection with ℓ.
3) After a rule *) the a cannot reappear.

Thus STITCH terminates in $O(|L|+|R|)$

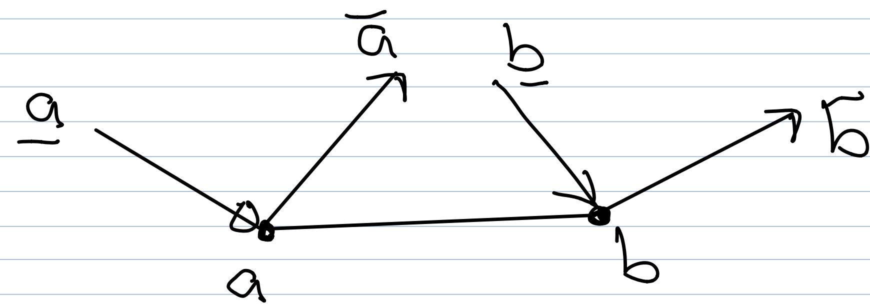
<u>Note</u> STITCH moves on L in a CW fashion
" " R in a CCW " "

STITCH cannot move
1) passed left most point of L.
2) passed right most point of R.

Thus at most n updates.

Correctness

At termination we have



Note Points in the Left are in cone

$\underline{a} \to a \to \bar{a}$

Points in Right are in cone $\underline{b} \to b \to \bar{b}$

Thus L & R are right of (a, b).

Timing: Preprocessing $O(n \log n)$

STITCH $O(n)$

$T(n) = 2T(n/2) + c \cdot n$

$\Rightarrow T(n) = O(n \log n)$

Random Incremental CH

Proc: RandomIncrementalCH(P)

0) Make $\Delta = (P_1, P_2, P_3)$; pick c∈interior(Δ)
1) Contruct a ray from C to each P_i.
2) Paritition P_i's by edge of Δ ray crosses.
3) Randomly permute $P_4, \ldots, P_n$.

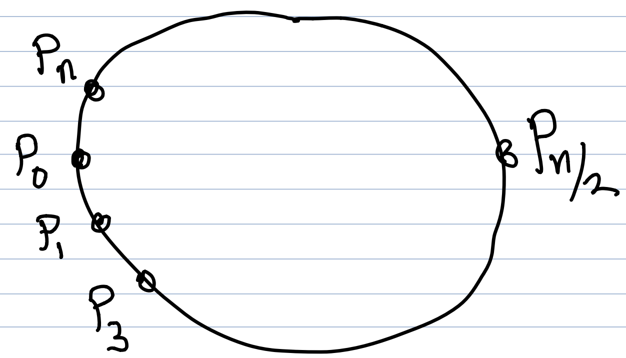
For $i = 4$ to n.

Let e be edge crossed by
ray $C \longrightarrow P_i$.
BuildTent(P_i, e)

Proc: BuildTent(P, e)

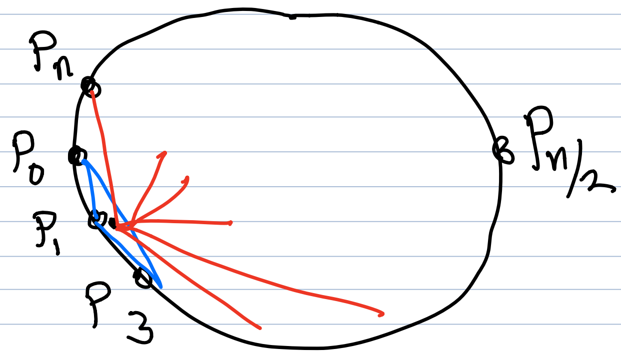
1) Find edges of CH "visible" to P by searching out from e.
2) Replace visible edge with 2 new edges.
3) Assign rays to the new edges.

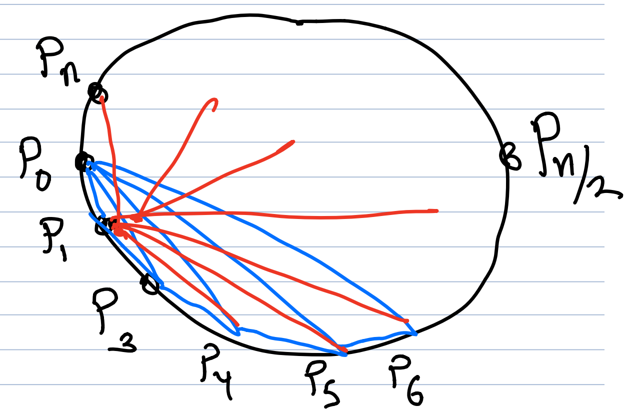
An Example of n-points on a Circle



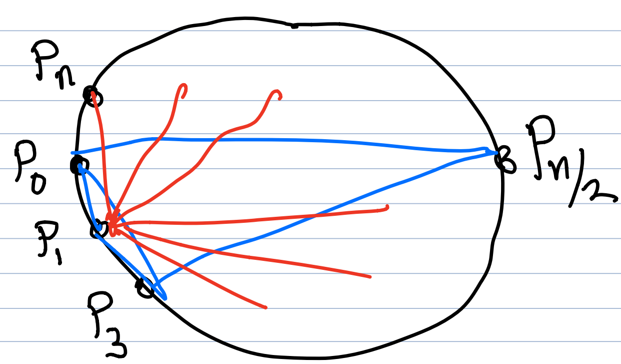
Worst Case: Incremental Order

$P_1, \ldots, P_n$





"Best" Case $P_1 P_2 P_3, P_{n/2}, P_{n/4}, P_{3n/4}, \ldots$



Correctness ?

Timing

Claim: $O(n)$ work other than BuildTent.

Case 1 Steps 1&2 over life of alg

1) At most $2n$ edges generated.

2) Charging rule for "visiblity" test.

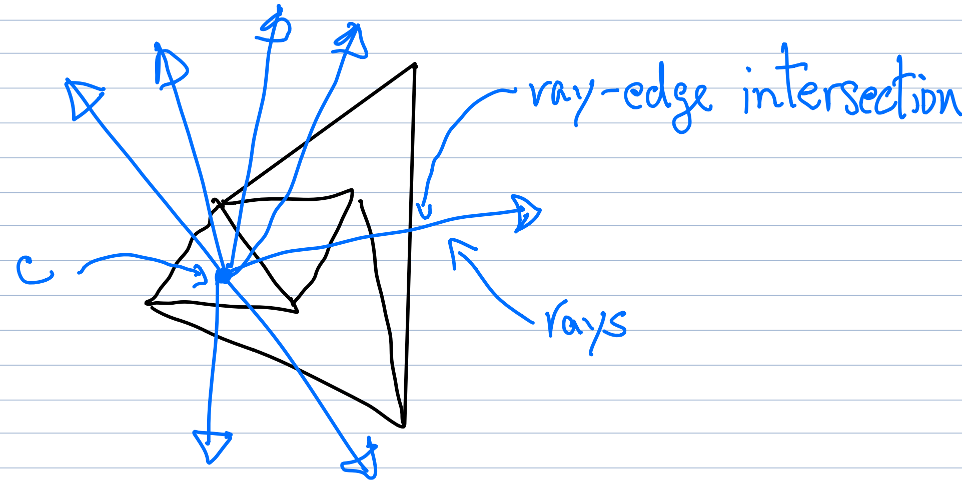
a) At most 2 "not visible" charge P_i.

b) each "visible" charge the visible edge.

Total $\underbrace{2n}_{\text{charge to points}} + \underbrace{2n}_{\text{charge to edges.}} = 4n$ tests.

Consider step 3 in BuildTent

i.e Cost to move point to new tent-edge.



Claim: Step 3 work $\approx$ # ray-edge intersection

pf. Backwards Analysis

i.e, We remove $P_4, \ldots, P_n$ is random order.

Let

$$V_i^j = \begin{cases} 1 & \text{if at time with } i \text{ points remaining } Ray_j \text{ crosses a removed edge.} \\ 0 & \text{o.w.} \end{cases}$$

Claim $P_r[V_i^j = 1] \le 2/i$

a) If Ray_j does not intersect any bdry edge then $V_i^j = 0$

b) Let $(v,w) = e$ be bdry edge that Ray_j intersects

In this case $V_i^j = 1$ iff v or w is removed.

Thus $P_r[V_i^j = 1 \mid Ray_j \cap Bdry] \le 2/i$

Def $V_i = \sum_{j=1}^{n} V_i^j$

$E(V_i) = \sum E(V) = \sum P_r[V_i^j = 1]$

$\le 2n/i$

Total Expected work $2nH_n$.