Convex Hull

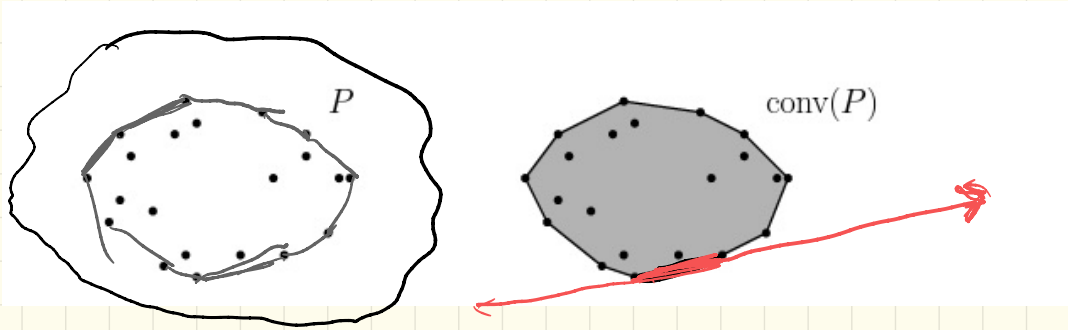
1/19

Convexity

Given a set of points P in the plane

convex hull of P, $\text{conv}(P)$

intuitively area bounded by a rubberband snapped around nails

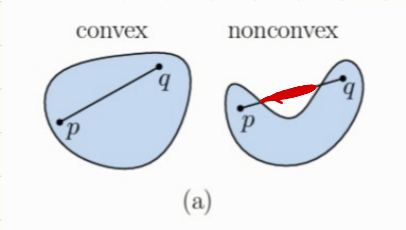


Importance:

- simple shape to represent the pointset
- convex approx to complex shapes
- 1st steps in algos (in practice)

Defs:

convex: a set K is convex if for $p, q \in K$ seg $\overline{pq} \in K$



bounded: if it can be enclosed
in a sphere w/ fixed radius

example of convex bounded set

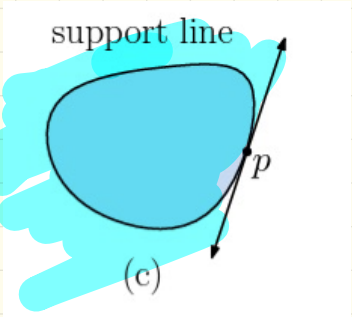
example of a convex unbounded set

Halfplane: given a line ℓ
the halfplane is the set of points on one side of ℓ
(possible include ℓ)

closed- half plane

open halfplane

Supporting line / halfplane: K convex
$\forall\ p \in K$ on the boundary
$\exists$ at least 1 line ℓ
$\ni\ p \in \ell$
and K is completly in the halfplane defined by ℓ

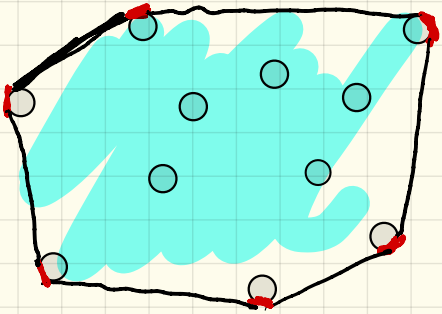


Convex hull:

for a set P

the intersection of all convex sets containing P

... it is the smallest convex set containing P ($\text{conv}(P)$ notated)



Note:

rep of the convex hull – convex polygon

actual convex hull = a convex set

Convex hull problem:

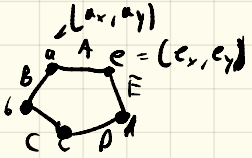
Given a set of n points P in the plane

output a representation of the convex hull of P

Rep: closed convex polygon

- set of line segments

(A, B, C, D, E)



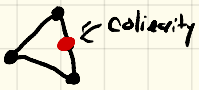
- pair of points

$A = (c, a)$

$B = (a, b)$

- CCW ordering of points (points are cartesian coords)

(a, b, c, d, e)



Assure points are in general position.

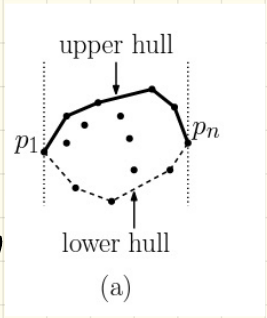
- degenerate configurations do not occur in the input

1. no 3 points are collinear 2. no 2 points share same x- or y-coord

Graham's Scan (1972)

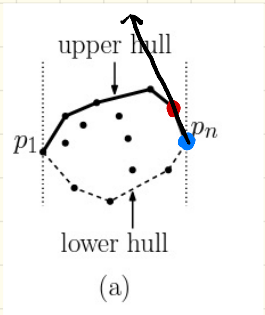
- incremental construction: add 1 point at at a time and update

Represent the bdd of the hull w/ 2 polygonal chains
- * upper hull
- ¥ lower hull



Observe that leftmost and rightmost points of the point set are in the lower hull and the upper hull

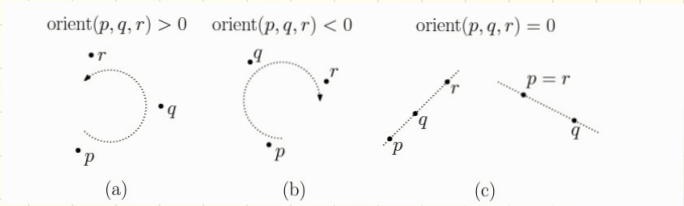
Idea:
- Compute lower hull
- Compute upper hull
- Concat and return



Upper hull computation

How to compute "left turn"

Orient 3 points $\langle p, q, r \rangle$



Computed as:

$$\text{orient}(p,q,r) = \det \begin{pmatrix} 1 & r_x & q_x \\ 1 & p_y & q_y \\ 1 & r_y & r_y \end{pmatrix} = \det \begin{pmatrix} p_x - r_x & p_y - r_y \\ q_x - r_x & q_y - r_y \end{pmatrix}$$

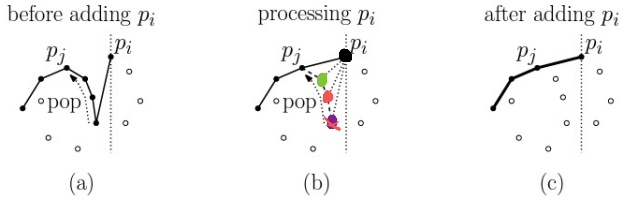


Fig. 14: Graham's scan.

Graham's Scan

(1) Sort the points according to increasing order of their x-coordinates, denoted $\langle p_1, p_2, \ldots, p_n \rangle$.
(2) push p_1 and then p_2 onto S ← a stack
(3) for $i \leftarrow 3, \ldots, n$ do:
  (a) while ($|S| \geq 2$ and Orient$(p_i, S[\text{top}], S[\text{top}-1]) \leq 0$) pop S.
  (b) push p_i onto S.

Correctness:

prove loop invariant is true after each iteration

Claim: after inserting point p_i
S contains the upper hull of $P_i = \{p_1, \ldots, p_i\}$ (in order sorted by x-coord)

Induction:

p_1 & p_2 are the upper hull of P_2

1) as p_i is the right most point in P_i
⇒ p_i is in the upper hull P_i
let p_j be the vertex preceding p_i in upper hull of P_i

Show:

a) entering (3) p_j is on S
b) every point above p_j on the stack is popped off
c) Popping stops at p_j

prove (a)

p_j is on the stack

$\Leftrightarrow$ on the upper hull of P_{i-1}

$\Leftrightarrow$ supporting line passing through p_j

$\overrightarrow{p_i p_j}$ is such a line

prove b) p_k w/ $j < k < i$ in the upper hull of P_{i-1}

orth of p_i, p_k, p_k's pred is neg

$\Rightarrow$ pop p_k

prove c) at a tagney p_j orthcon becoms positive $\Rightarrow$

p_j is at "top" of the stack

Runtime:

(1) sort. $O(n \log n)$

(2) let d_i be the number of pts poped at step i — $O(d_i + 1)$
- each orientation takes $O(1)$
- add p_i to stack

runtime is proportional to

$$\sum_{i=1}^{n} (d_i + 1) = n + \sum_{i=1}^{n} d_i$$

$\sum_{i=1}^{n} d_i \leq n \Rightarrow$ adding n points takes $O(n)$ time for upper hull

- $O(n \log n)$ time to sort
- $O(n)$ time for upper hull
- $O(n)$ time for lower hull

$\Rightarrow$ total time for algo $O(n \log n)$

space $O(n)$