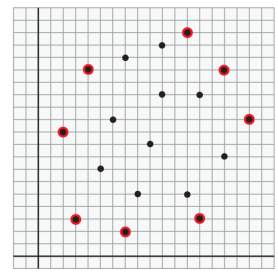
Computational Geometry

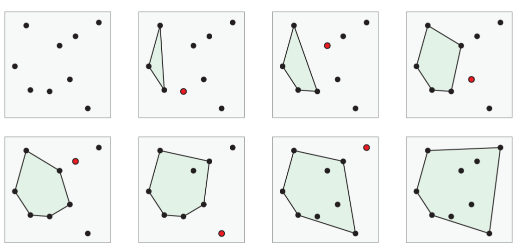
Convex Hull

Hull Points

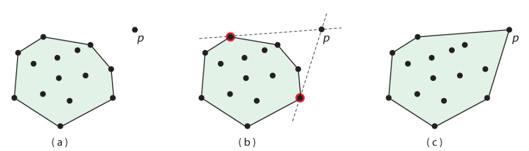
(2, 10)	(10, 17)
(3, 3)	(10, 13)
(4, 15)	(12, 5)
(5, 7)	(12, 18)
(6, 11)	(13, 3)
(7, 2)	(13, 13)
(7, 16)	(15, 8)
(8, 5)	(15, 15)
(9, 9)	(17, 11)



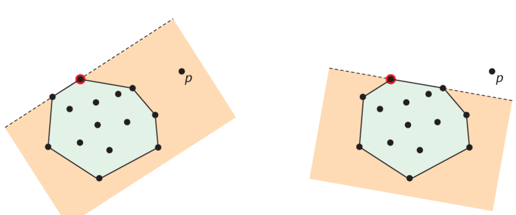
The Incremental Algorithm



$P \notin \mathrm{conv}(H_k)$

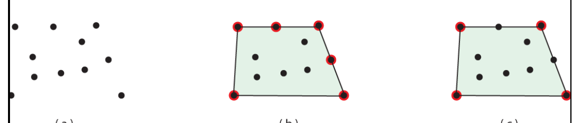


Tangent Test



Corner Cases

- Degeneracies:
  - points with same x-coordinates
  - collinear points



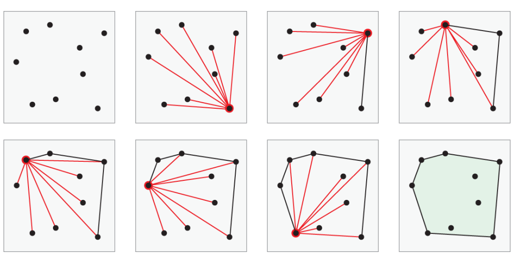
Big-O

- $O(f(n))$ means $cf(n)$ is an upper bound on the running time of the algorithm for some constant $c > 0$ and sufficiently large n.
- Anything that takes one step is $O(1)$: i.e. whether or not an edge is visible to p.
- Sorting takes $n\log(n)$ time.
- Loop steps are multiplied and sequential steps are added.

Complexity

- Sort by x
- For each p, test each edge of the current hull for visibility to p
  - in the worst case, we may have to consider all edges in current hull
- $O(n^2)$

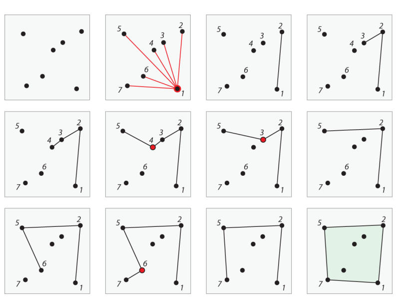
Gift Wrapping



Complexity

- Sort and find bottom (rightmost) point
- For each point on the hull, calculate angles to all other points
- $O(nh)$
- Output sensitive

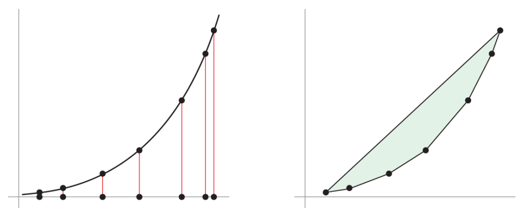
Graham Scan



Lower Bound

- What is the best we can do?
- Denoted by Ω
- Can we do better than sorting?

Reduction to Sorting



Theorem

- A lower bound for any algorithm that identifies the hull points of a point set in the plane is $\Omega(n \log n)$