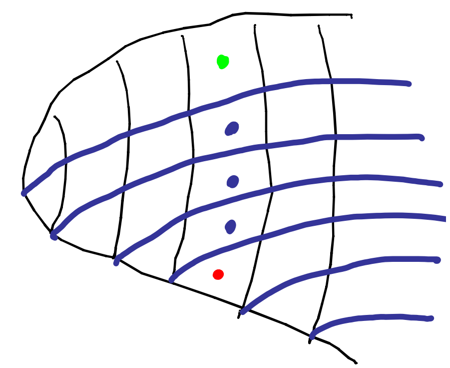
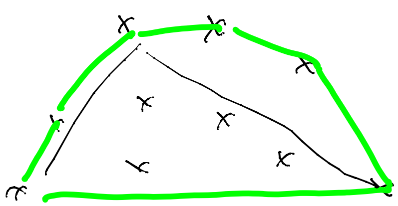
From Output Sensitivity to Instance Optimality (planar) Convex Hull

Peyman Afshani – Jérémy Barbay – Timothy Chan

Plan

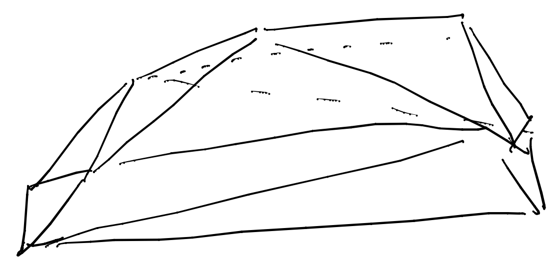
I Beyond Worst Case Analysis





II Planar Convex Hull: the Odyssey

III Summary & Open Problems



I.1 Worst Case Analysis: What and Why ?

What:
- fix a Problem
- reduce to finite set S of Instances (e.g. $|I| = n$)
- Given algorithm A: $\boxed{\max_{I \in S} C(A, I)}$
- Given a family of algorithms $\mathfrak{A}$ $\boxed{\min_{A \in \mathfrak{A}} \max_{I \in S} C(A, I)}$

Why: Yields a Partial Order on $\left\{ \begin{array}{l} \text{Algorithms and} \\ \text{Problems} \end{array} \right.$

Example:

$$\begin{tikzcd} \text{Gift Wrap} \arrow[r] \arrow[d] & \text{Gift Revisited} \arrow[d] \arrow[dl, leftrightarrow] \\ \text{Graham} \arrow[r] & \text{KS} \end{tikzcd}$$

I.2 The limits of Worst Case Analysis

It is too Pessimistic !!!

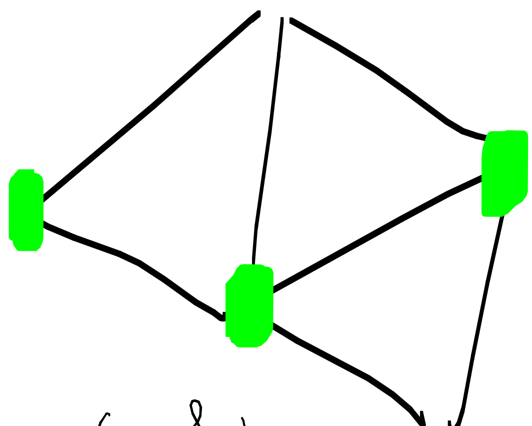
Solutions

- Experimentation (Future $\stackrel{?}{=}$ Past)
- Average Case (Future $\stackrel{?}{=}$ Random)
- Smooth Analysis (Input has random errors)
- Competitive Analysis
- Parameterized Complexity
- Adaptive Analysis
  - Output Sensitivity
  - Instance Optimality

refine the analysis (by restricting S further)

I.3 Quick Examples

Vertex Cover:

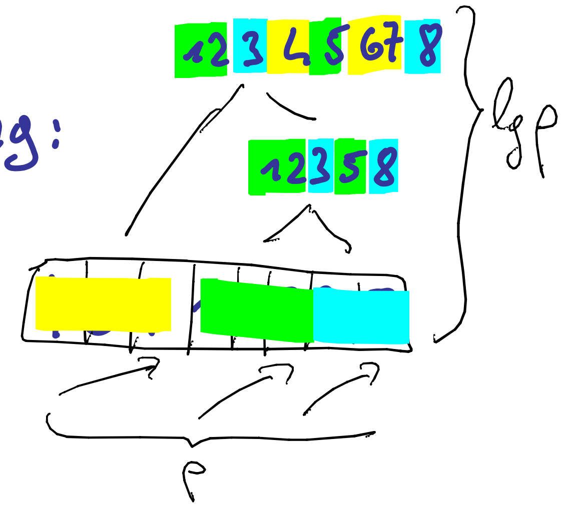


$O(m^k) \subseteq O(n^{2k})$

(- - -)

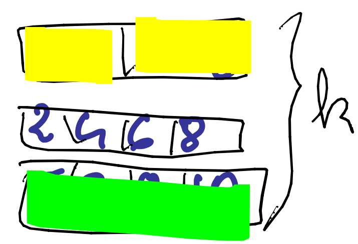
Sorting:

$O(n \lg p)$

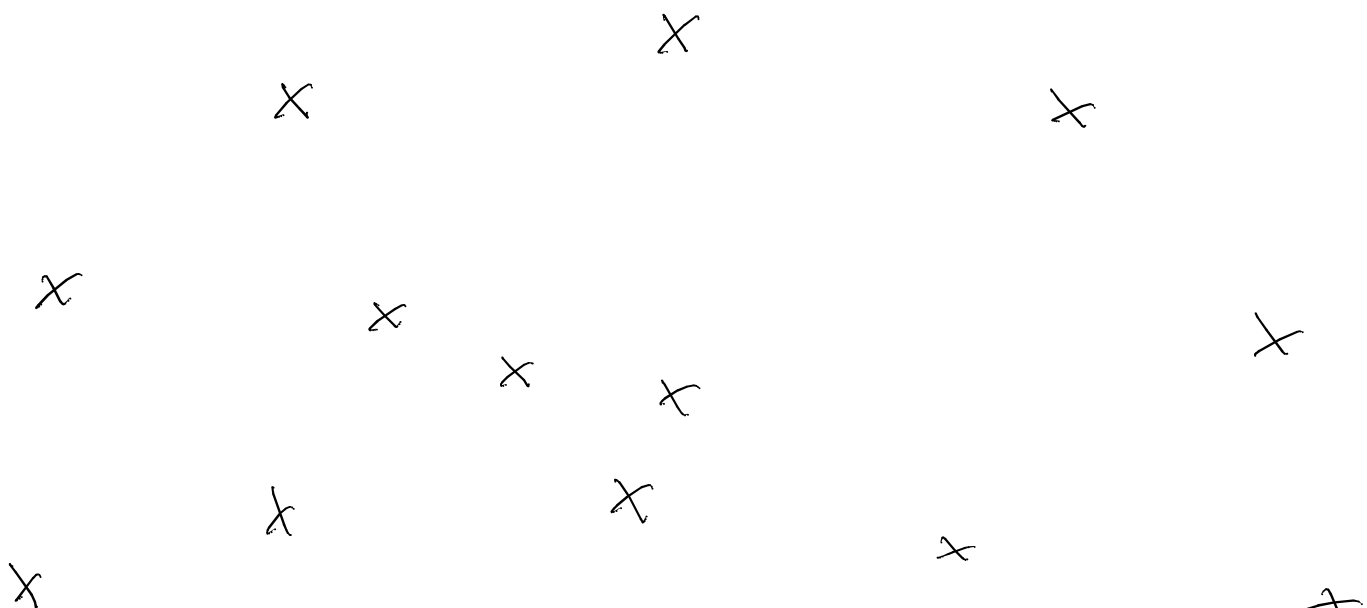


Intersection:

$O(\delta k \lg n/\delta k)$



II Planar Convex Hull



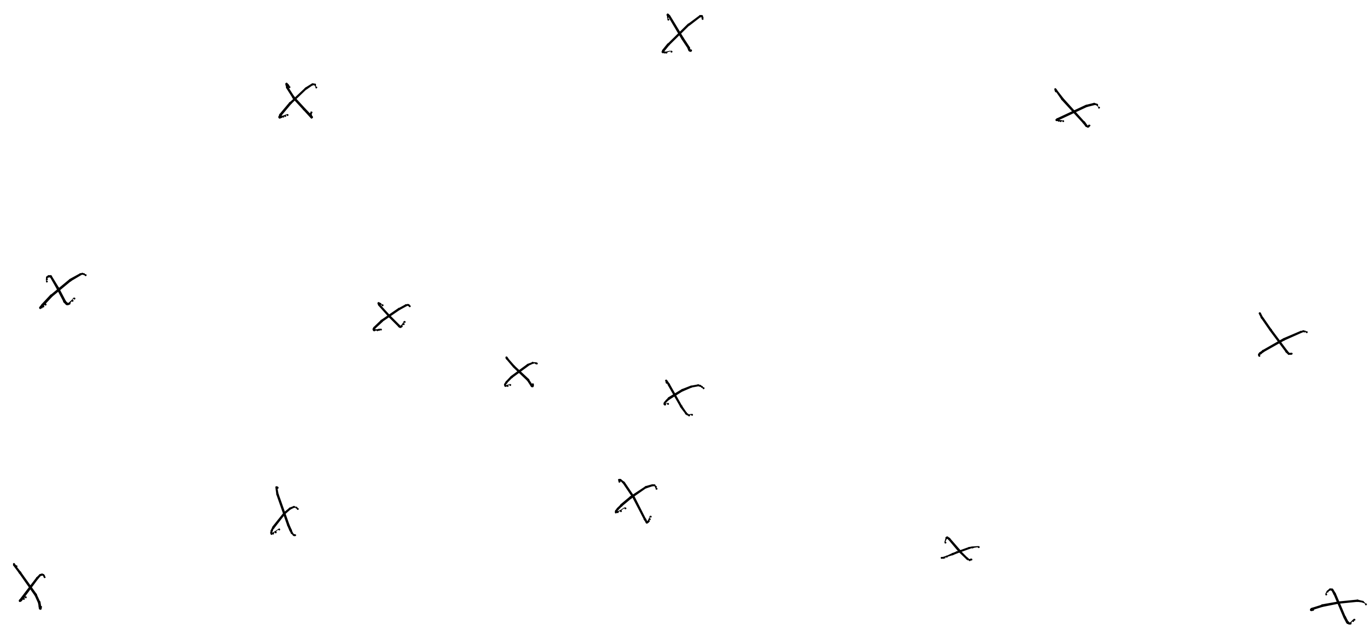
Input n points in the plane

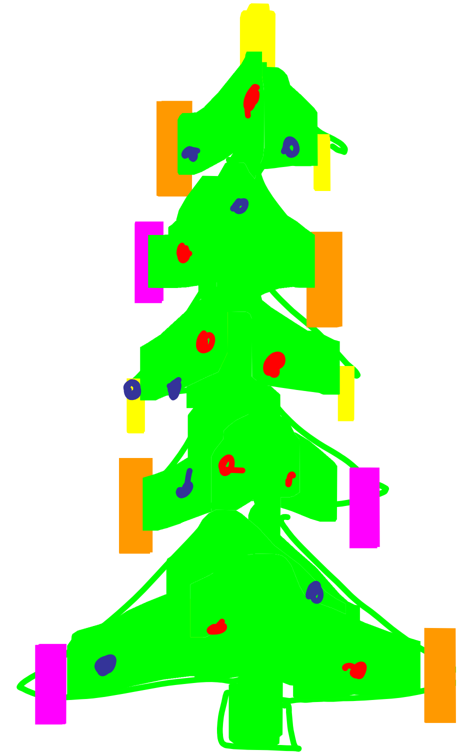
Output h points

$\approx$ UPPER HULL

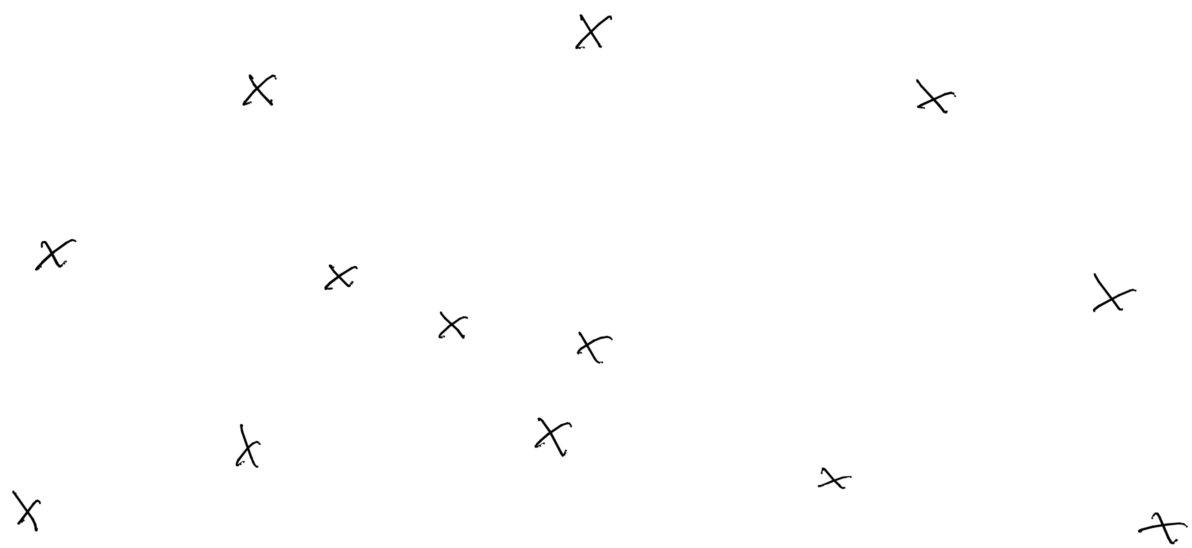
II.1 Gift Wrapping $\mathcal{O}(nh) \subseteq \mathcal{O}(n^2)$

(min angle)





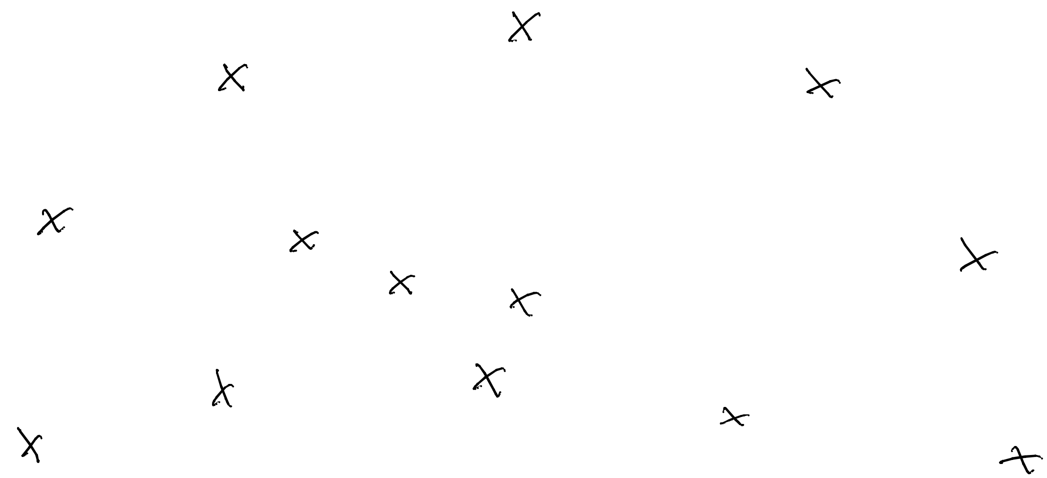
II.2 Graham's Scan $O(\text{Sorting}(n) + n) \subseteq O(n \lg n)$



$$\begin{tikzcd} O(n^2) \arrow[r, equal] \arrow[d] & O(nh) \arrow[dl, leftrightarrow] \arrow[r] & ? \\ \Theta(n \lg n) \arrow[urr] & & \end{tikzcd}$$

II.3 "The ultimate planar convex hull algorithm?"

[Kirkpatrick Seidel 1986] $\Theta(n \lg h) \subseteq \Theta(n \lg n)$



1. Divide by median in L, R
2. Find highest edge crossing L/R
   linear programming
3. Prune
4. Recurse

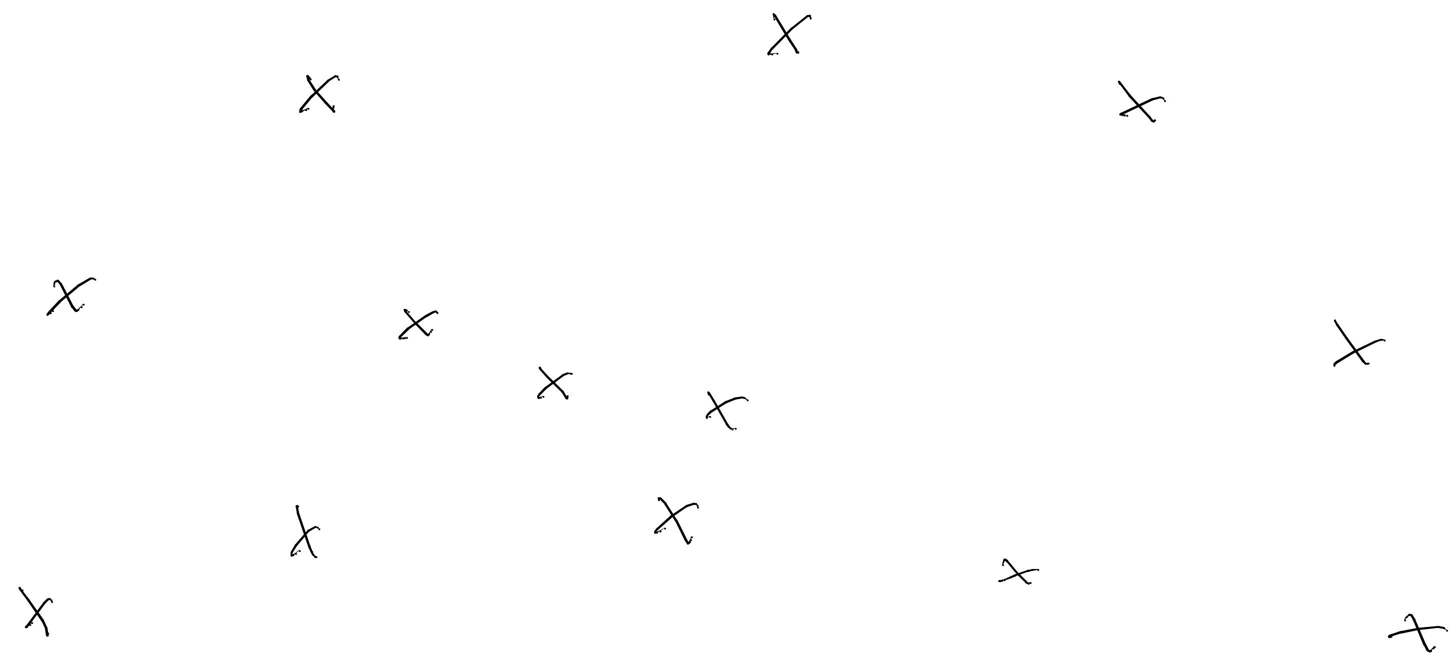
$$\begin{tikzcd}
O(n^2) \arrow[rr, equal] \arrow[dr] & & O(nh) \arrow[dr] & \\
& \Theta(n \lg n) \arrow[ur, leftrightarrow] \arrow[rr] & & \Theta(n \lg h)
\end{tikzcd}$$

II.4 Vertical Partitioning

$O(n \; H_v) \subseteq \Theta(n \lg h)$

[Sen Gupta 1999, Distribution Sensitive algorithms]

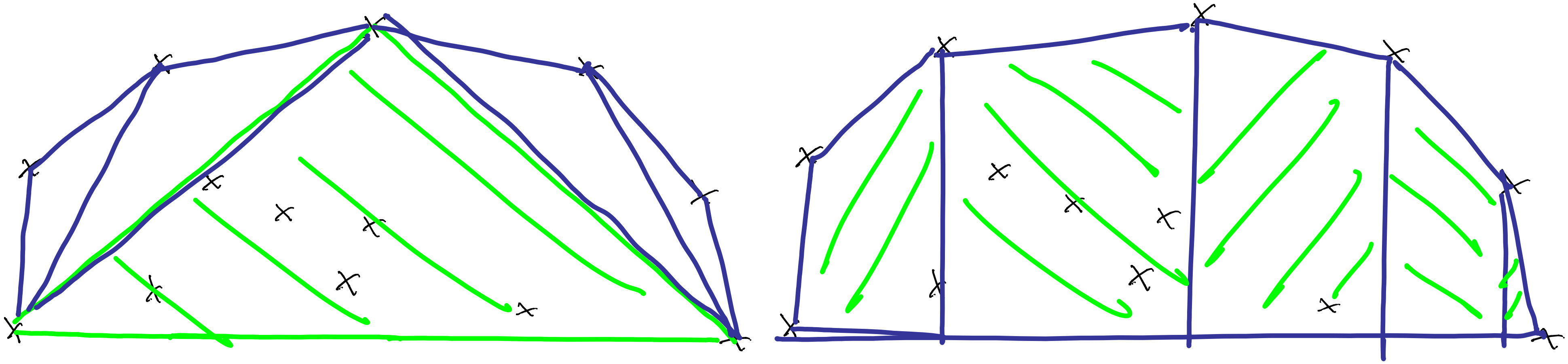
Refined Analysis of KS. $H_v = \sum_{i=1}^{h} \frac{n_i}{n} \lg \frac{n_i}{n}$ Orientation Dependent!



$$\begin{tikzcd}
O(n^2) \arrow[rr, equal] \arrow[dr] & & O(nh) \arrow[dr] & & O(nH_v) \\
& \Theta(n \lg n) \arrow[ur, leftrightarrow] \arrow[rr] & & \Theta(n \lg h) \arrow[ur] &
\end{tikzcd}$$

II.5 Instance Optimality $\Theta(nH) \subset \Theta(nH_v)$

Define a **Certificate** Π as a **Partition** covering the convex Hull



Define **Entropy(Π)** as the distribution of the points it eliminates.

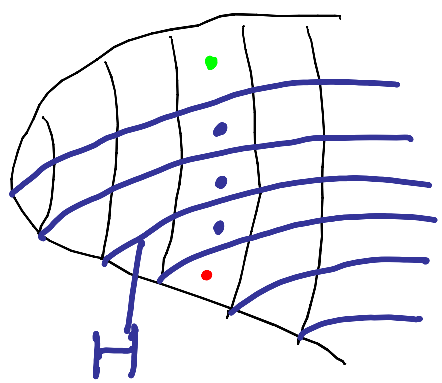
$$H = \text{Entropy}(I) = \min_{\Pi} \text{Entropy}(\Pi)$$

II.5.a (Order Oblivious) Instance Optimality

Ignore Input Order: $\tilde{C}(A,I) = \max_{\pi} C(A,\pi(I))$

An algorithm $A \in \mathcal{A}$ is Instance Optimal if

$\forall I\ \exists c\ \forall A' \in \mathcal{A} \quad \tilde{C}(A,I) \leq c\,\tilde{C}(A',I)$

i.e, in 

the cells contain only permutations of the same instance.

Π.S.b $KS \in O(nH) = \min_{\Pi} O(n\,H(\Pi))$

Proof

- How many time does a point "survives" KS?

Let n_j the total number of points surviving level j.

- Fix a partition $\Pi = (S_1, \dots, S_k)$

  ∟ There are at most $\frac{n}{2^j}$ survivors
  - at level j
  - between 2 points.

$$\Rightarrow \sum_{j=0}^{\lg n} n_j \le \sum_k \sum_{j=0}^{\lg n} \min\left\{|S_k|, \frac{n}{2^j}\right\} \le \sum |S_k| \lg \frac{n}{S_k} + \underbrace{|S_k| + \frac{|S_k|}{2} + \cdots}_{2S_k}$$

$$\le \sum |S_k| \left(\lg \frac{n}{S_k} + 2\right) = O\left(n\left(H(\Pi) + 1\right)\right) \quad \square$$

II.5.C Lower Bound(s) $\Omega(n H)$

- In simplest model:
  nH is the space required to specify the class of each point, i.e. to certify that it is or not in the Convex Hull
- For more general models (e.g Multilinear Constant), there is an adversary strategy forcing any decision tree using multilinear functions of constant degree,

III.1 Summary

Planar Convex Hull: Algorithms and Analysis

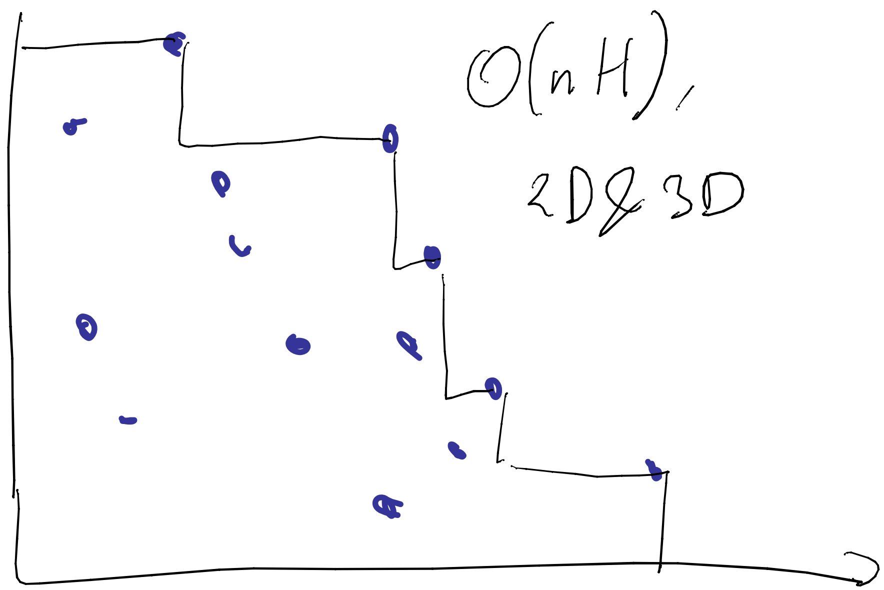
$$\begin{tikzcd}
O(n^2) \arrow[rr, equal] \arrow[dr] & & O(nh) \arrow[dr] \arrow[dl, leftrightarrow] & & & O(nH_v) \arrow[dr] & \\
& \Theta(n \lg n) \arrow[rr] & & \Theta(n \lg h) \arrow[urr] & & & \Theta(nH)
\end{tikzcd}$$

Kirkpatrick and Seidel "Ultimate" algorithm!

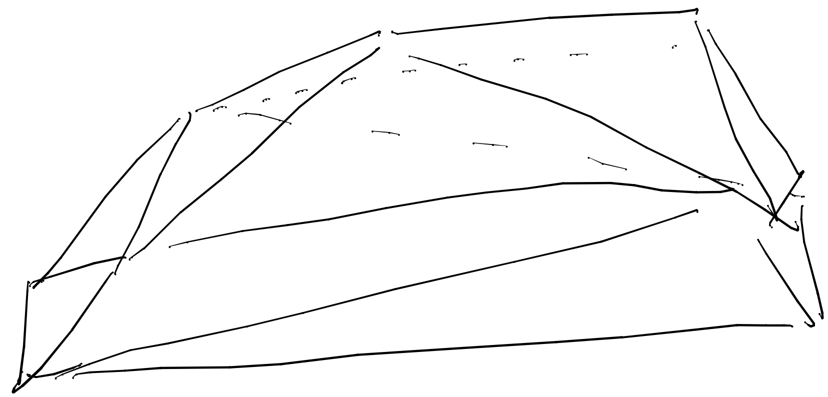
III.2 Similar Results

Levels of Maxima (2d, 3d)

O(nH),
2D & 3D



Convex Hull 3d O(nH)



offline orthogonal searching 2D

(– – –)

Reporting/Counting Pbs in 2d

- orthogonal intersections
- pairs of L_∞ distance at most 1 between red and blue.

III.3.a Open Problems : More general $\mathfrak{A}$.

In our results,

$\mathfrak{A}$ = decision trees where tests involve only Multilinear functions with a constant number of arguments.

$\Longrightarrow$ Extend to larger family of Algorithms

III.3.b Open Problems: Input Order 2D

$O(n^2)$ $=$ $O(nh)$ $\quad$ $O(nH_v)$

$\Theta(n\lg n)$ $\rightarrow$ $\Theta(n\lg h)$ $\quad$ $\Theta(nH)$

$O(n\lg\chi)$ $\quad$ $O(n\lg\rho)$ $\quad$ $O(n\lg Inv)$ ...

Levcopoulos et al

χ = nb of crossing points (for this input order)

Adaptive Sorting Algorithms

$\rightsquigarrow$ Instance Optimality / Input Order Aware ?

III.3.c Open Problems: Input Order 3D

Less results, but (almost) all dominated by "(Input Order Oblivious) Instance Optimality":

$$
\begin{tikzcd}
\Theta(n \lg n) \arrow[r] \arrow[dr, leftrightarrow] & \Theta(n \lg h) \arrow[r] \arrow[d, leftrightarrow] & \Theta(nH) \arrow[dl] \\
& O^{*}(n..n^2) &
\end{tikzcd}
$$

$\Longrightarrow$ Input Order Aware Instance Optimality in 3D?