Chan's Algorithm 1996

Let \( m = \frac{n}{h^2} \)

1. Partition points into \( \left\lceil \frac{n}{m} \right\rceil \) subsets of \( \leq m \) points
2. Use Graham Scan to find \( \text{CH} \) of each subset
   
   \[
   \text{runtime} = \frac{n}{m} \cdot m \log m = O(n \log m)
   \]
3. Use Jarvis March to find \( \text{CH} \) of \( \text{CH}'s \) of subsets.

\( \text{P}_c \) = last point known to be on \( \text{CH}(P) \)

3.1 Use binary search on each subset \( \text{CH} \) to find
   “minimum” (i.e. clockwise-most w.r.t. \( \text{P}_i \)) in each
   subset in time \( O(\log m) \)

Find all minima in \( O(\frac{n}{m} \log m) \) time
[and take the smallest]

How many times do we need to perform such a wrap step?

After \( h \) wrap steps we get the convex hull

Jarvis March time = \( O(h \frac{n}{m} \log m) \)
Total time \( O(n \log m) + O(\frac{h}{m} \log m) = O(n \log m) \) if \( m = h \)

But we don't know \( h \) so we'll guess different values for \( m \).

Try small values of \( m \), first.

If after \( m \) wrap steps we do not find CH

then STOP [which takes \( O(n \log m) + O(\frac{m}{m} \log m) = O(n \log m) \) time]

and increase \( m \).

Repeat

Double \( m \)

\( m = 2, 4, 8, 16, \ldots \)

\( \log h \)

\( 2^i \approx h \Rightarrow i \geq \log h \)

Total time

\[ \sum_{i=1}^{\log h} n \log 2^i = n \sum_{i=1}^{\log h} i = n (\log h)^2 \]

Too Big

Square \( m \)

\( m = 2, 4, 16, 256, \ldots \)

\( \log \log h \)

\( \frac{\log \log h}{\log h} \)

Total time

\[ \sum_{i=0}^{\log \log h} n \log (2^i) = n \sum_{i=0}^{\log \log h} 2^i = n 2^{\log \log h} \]

\( 2^i \approx h \Rightarrow i \geq \log h \)

Just Right
Algebraic decision tree Lower Bound for CH

**Decision Tree** is a model of computation that focuses on decision points in an algorithm.

Every path is a possible execution trace of an algorithm on some input of size \( n \).

Leaves are output.

Algebraic decision trees permit \( a, b, c \), to be algebraic expressions of degree \( d \) (for fixed \( d \)).

- The variables in the expressions are real numbers (like the coordinates of input points for CH).
- The set of inputs that lead to a leaf is a set of vectors (in \( \mathbb{R}^{2n} \) for CH of \( n \) points in 2D).

**Theorem [Ben-Or]** Let \( W \subset \mathbb{R}^n \) be any set and let \( T \) be any \( d \)th order algebraic decision tree that decides membership in \( W. \) If \( W \) has \( m \) disjoint connected components then \( T \) has height \( \Omega(\log m - n) \).

**Problem** Multiset Size Verification: Given multiset \( Z = \{z_1, z_2, \ldots, z_n\} \), where \( z_i \in \mathbb{R} \) and integer \( k \), does \( Z \) have exactly \( k \) distinct elements?

\[ M_k = \{ (z_1, \ldots, z_n) \in \mathbb{R}^n \mid \sum_{i=1}^{n} z_i = k \} \] has \( \geq k! \) \( k \)-\text{disjoint components}.

Thus \( \text{MSV} \) requires \( d \)th order alg. decision tree height

\[ \Omega(\log(k! k^{n-k})) = \Omega(n \log k) \]
Thm: CH size verification requires $\Omega (n \log n)$ steps in the worst case using any $d^{th}$ order decision tree alg.

Proof: Construct points from instance $Z = \{ z_1, z_2, \ldots, z_n \}$ of MSV problem.

$P_i = (z_i, \overline{z_i})$

Then $\overline{P_1, P_2, \ldots, P_n}$ has $k$ hull points iff $Z$ has $k$ distinct elements.

Kirkpatrick & Seidel

Go on to show that even if we assume that all points are distinct, CH size verification still requires $\Omega (n \log n)$ steps.