Method

1. Divide recursively into halves
   - Can fix $x$-direction or alternate $x$ and $y$
2. Triangulate base sets (of size 2 or 3)
3. Combine two triangulations by stitching
Next candidate vertex?

How to find it?
ccw next r-edge starts
$r_2 \in \text{circle: continue}$

$r_3 \in \text{circle: continue}$
$r_4 \in \text{circle: continue}$

$r_5 \notin \text{circle: stop}$
valid edge: Delaunay
$I_3 \not\in \text{circle: stop}$

$r_{next}$ wins over $l_{next}$...
complete stitching…
lower convex hull edge: done
Stitching

- Initially find the connecting upper hull edge
- For edge \((l, r)\) determine \(l_{next}\) and \(r_{next}\)
  - for \(r\), start at ccw next vertex \(r_k\)
  - if \(r_{k+1}\) is in \(\text{circle}(l, r, r_k)\)
    - delete edge \((r, r_k)\) and continue
  - otherwise set \(r_{next} = r_k\) and propose \(\Delta(l, r_k, r)\)

\[ \begin{array}{c}
  \text{l} & \text{r} \\
  \text{r}_{k-1} & \text{r}_k \\
  \text{r}_{k+1}
\end{array} \]

\[ \begin{array}{c}
  \text{l} & \text{r} \\
  \text{r}_k & \text{r}
\end{array} \]

Analysis

- Recursive division is \(O(n \log(n))\) including initial sorting
- Finding upper hull edge is \(O(n)\)
- Interior stitching proportional to number of edges deleted and added (why?)
  - Total number of red edges is \(3n/2\)
  - Total number of blue edges is \(3n/2\)
  - Maximum of black edges that can be created is \(3n\)
- So, stitching is \(O(n)\)
- So the entire algorithm is \(O(n \log(n))\)