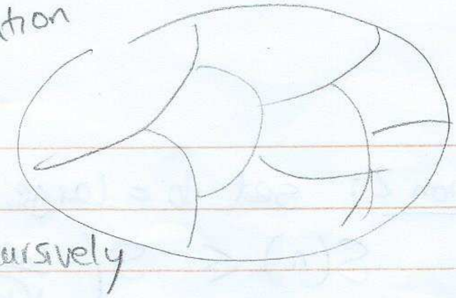
after retriangulation



Build DS for R

Build DS inside each region recursively

Query: find region of R containing q

recurse inside region

$$\begin{cases} S(n) \leq \sum_i S(n_i) + S_0\left(\frac{c_0 n}{\sqrt{b}}\right) + O(n) \\ \qquad \text{with } \sum_i n_i \leq n, \quad \max_i n_i \leq b \\ Q(n) \leq Q(b) + Q_0\left(\frac{c_0 n}{\sqrt{b}}\right) + O(1) \end{cases}$$

<u>Option 1</u>: start with $S_0(n) = O(n^2)$, $Q_0(n) = O(\log n)$ (Method 0)

set $b = n^{1-\delta}$: $S(n) \leq \sum_i S(n_i) + O\left(\left(n^{\frac{1+\delta}{2}}\right)^2\right)$

$$\Rightarrow S(n) = O(n^{1+\delta})$$

$$Q(n) \leq Q(n^{1-\delta}) + O(\log n)$$

$$\Rightarrow Q(n) = O(\log n + (1-\delta)\log n + \dots) = O(\log n)$$

bootstrap with $S_0(n) = O(n^{1+\delta})$, $Q_0(n) = O(\log n)$

set $b = \sqrt{n}$: $S(n) \leq \sum_i S(n_i) + O(n)$

$$\Rightarrow S(n) = O(n \log\log n)$$

$$Q(n) = O(\log n)$$

bootstrap again with $S_0(n) = O(n \log\log n)$, $Q_0(n) = O(\log n)$

set $b = (\log\log n)^2$: $S(n) \leq \sum_i n_i + O(n)$

& don't recurse

$$\Rightarrow S(n) = \boxed{O(n)}$$

$$Q(n) \leq O(b + \log n)$$

$$\Rightarrow Q(n) = \boxed{O(\log n)}$$

Rmk: this gives general space reduction that preserves query time

Option 2: set b = large const, recurse in R instead

$$S(n) \le S\left(\frac{c_0 n}{\sqrt{b}}\right) + O(n) \le S(n/2) + O(n)$$
$$\Rightarrow O(n)$$

$$Q(n) \le Q\left(\frac{cn}{\sqrt{b}}\right) + O(1) \le Q(n/2) + O(1)$$
$$\Rightarrow O(\log n)$$

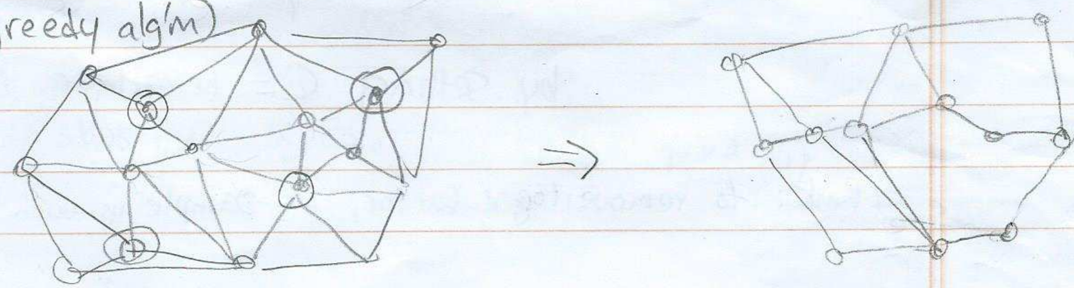
Method 6: Kirkpatrick's Hierarchy ('83)

simplification of option 2 without separators!

Fact In any planar graph with n vertices, $\exists$ indep set I of size $\geq \frac{n}{24}$.

Furthermore, each vertex in I has deg ≤ 11.

(Pf: by greedy alg'm)



remove $I \Rightarrow$ subdivision R with $\le \frac{23}{24} n$ vertices

each region has size ≤ 11.

Rmk: Dobkin-Kirkpatrick extends to 3D convex polyhedron

$$S(n) \le S\left(\tfrac{23}{24} n\right) + O(n) \Rightarrow O(n)$$
$$Q(n) \le Q\left(\tfrac{23}{24} n\right) + O(1) \Rightarrow O(\log n).$$