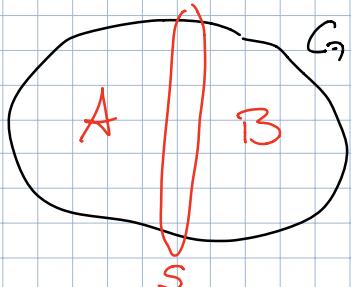


HWS out

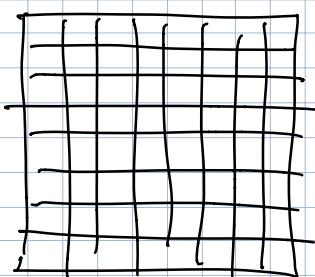
Planar separator theorem.



$$|S| = O(\sqrt{n})$$

$$|A| < \frac{2}{3}n$$

$$|B| < \frac{2}{3}n + O(1)$$



Application: Shortest cycle in dir. graph.

① Separate $\rightarrow (A, S, B)$

② Recurse in $G[A]$, in $G[B]$

③ Look for cycles containing $x \in S$

for all $x \in S$

build BFS(x)

look at all edges $v \rightarrow x$ not in tree

$$T(n) = O(n) + T(n_1) + T(n_2) + O(|S| \cdot n)$$

$$n_1 + n_2 < n \quad n_1 > \frac{n}{3} \quad n_2 > \frac{n}{3}$$

$$\leq T\left(\frac{n}{3}\right) + T\left(\frac{n}{3}\right) + O(n^{3/2}) = \boxed{O(n^{3/2})}$$

[Lipton Tarjan 79]

① Level separator — L_i = all vertices at distance i from some fixed vertex r .

Median level is bal. separator. but maybe big

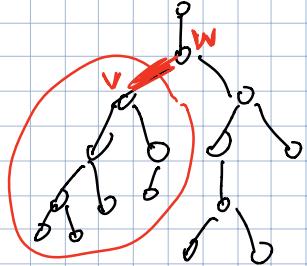
② Fundamental Cycle separator

Spanning tree T $\text{cycle}(T, e) = \text{unique cycle in } T \setminus e$

Assume G is a simple triangulation

$T = \text{any span. tree}$ $C = \text{dual spanning tree}$
 every vertex has degree ≤ 3

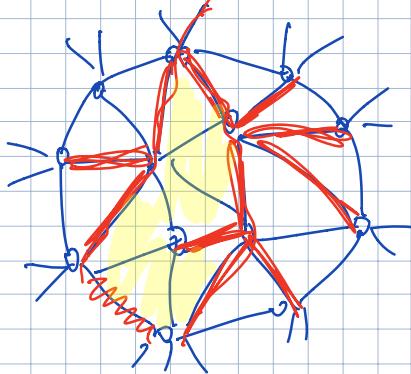
root C at any leaf. \rightarrow now C is a binary tree



Walk down from the root
always into larger subtree

v = first node subtree has $< \frac{2n}{3}$
vertices

$\Rightarrow v$ has $\geq \frac{1}{3}$ descendants ✓

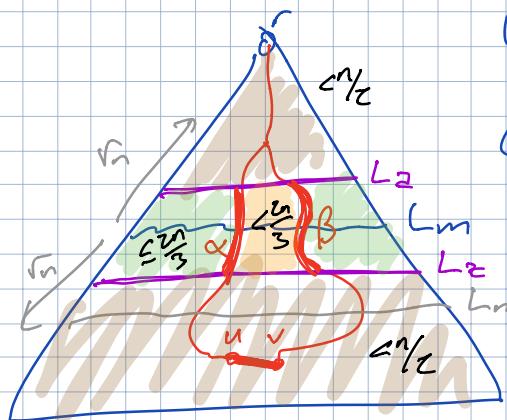


$\text{cycle}(T, (w)^*)$ is bal. separator.

• $T = \text{ISFS tree} \Leftarrow \boxed{\text{practise}}$

• $C = \text{BFS tree}$

① Build ISFS tree rooted at v



② Find median level L_m
Assume $|L_m| > \sqrt{n}$

③ Find balanced fund. cycle (T, uv)

Assume length $\gg \sqrt{n}$
depth $h(T) \gg \sqrt{n}$

④ Find levels L_2, L_z

$$|L_2| \leq \sqrt{n} \quad |L_z| \leq \sqrt{n}$$

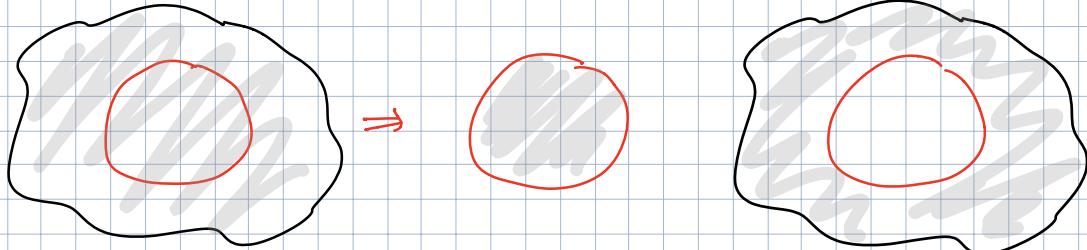
$$m - \sqrt{n} \leq z < m < z < m + \sqrt{n}$$

Claim: $L_2 \cup L_z \cup \alpha \cup \beta$ is a bal. seg of size $\leq 6\sqrt{n}$

Easy to find $O(n)$ time

Cycle separator [Miller 85ish] [Nayyeri Har-Peled '17]

Euler \Rightarrow suffices to balance Faces



$$T(n) = F(n) + T(n_1) + T(n_2)$$

$$n_1 + n_2 < n + O(\Delta)$$

$$n_2, n_1 < \frac{2n}{3} + O(\Delta)$$

T' = BFS rooted at r'

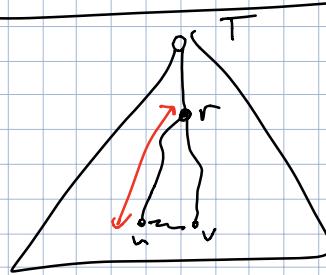
Find $\text{cycle}(T', uv)$ balanced sep

$r = \text{lca}(u, v)$

T = BFS rooted at r

$$\text{cycle}(T, uv) = \text{cycle}(T', uv)$$

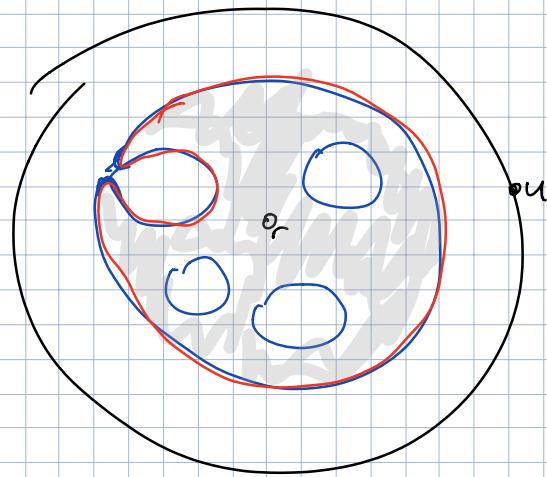
(choose canonical BFS tree)



Assume G is a triangulation

$$h = \text{dist}(r, u) > \text{dist}(r, v)$$

Assume u is on outer face



Define level of a Face xyz
 $= \max\{\text{dist}(x), \text{dist}(y), \text{dist}(z)\}$

R_i = union of all faces level $\leq i$

= disk with holes

C_i = outer boundary of R_i

— every vertex dist = i

— simple cycle

— disjoint from C_j for all $j < i$

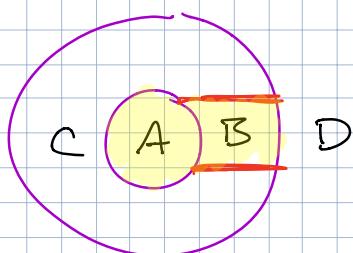
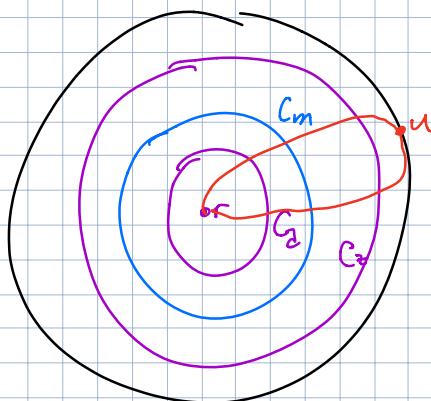
C_m — outermost C_i with $< \frac{E}{2}$ int faces

C_{m+1} has $< \frac{E}{2}$ ext faces

Find α, β

$$m - \sqrt{n} \leq \alpha C_m < \beta C_m + \sqrt{n}$$

$$|C_{\text{al}}| < \sqrt{n} \quad |C_{\text{el}}| < \sqrt{n}$$



- $|A| < \frac{E}{2} \Rightarrow$ if $|A| > \frac{E}{3}$, ∂A is separator

- $|D| < \frac{E}{2} \Rightarrow$ if $|D| > \frac{E}{3}$ done

- $|B| < \frac{2E}{3} \Rightarrow$ if $|B| > \frac{E}{3}$ done

- $|C| < \frac{2E}{3} \Rightarrow$ if $|C| > \frac{E}{3}$ done

- Otherwise $|A \cup B| \leq \frac{2E}{3} \quad |A \cup B| \geq \frac{E}{3}$

Easy find $O(n)$ time

r-division [Frederickson 83]

- Partition G into pieces

- each piece has $O(r)$ verts

- each piece has $O(\sqrt{r})$ bdry verts — shared w/ other pieces

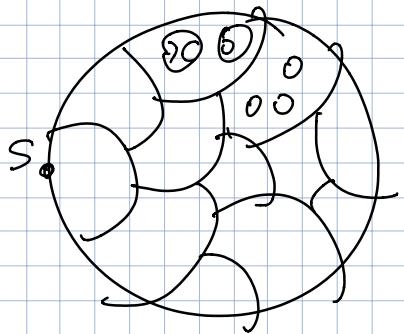
Alternate between split verts
split bdry verts
split holes

More generally, we can partition weight on V, E, F .

— each piece is a disk with $O(1)$ holes

Naively: $O(n \log \frac{n}{r})$ time $\rightarrow O(n)$ time! [KMS 15]

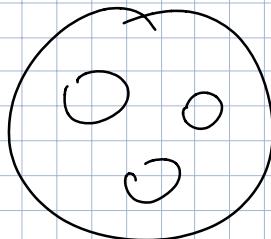
Shortest Paths!



① Build r -division $\rightarrow O(n)$ time

② For each piece:

Compute all bdry to bdry distances



$O(1) \times \text{MSSP}$

$$= O(r \log r)$$

Total: $O(n \log r)$ time

③ Replace each piece with clique with $O(\sqrt{r})$ verts.

Dense distance graph $O\left(\frac{n}{\sqrt{r}}\right)$ verts $O(n)$ edges

wlog S is a bdry vertex

④ Dijkstra on DDG $O(V \log V + E) = O\left(\frac{n}{\sqrt{r}} \log n + n\right)$

⑤ Run Dijkstra in each piece $O(n \log r)$

$$\text{Total time} = O(n \log r + \frac{n}{\sqrt{r}} \log n) \neq O(n \log \log n)$$

$$r = O\left(\frac{(\log n)^2}{\log \log n}\right)$$

$O(n)$ [ugh]