MINIMUM CUT LINEAR ARRANGEMENT

7/11/2025

Given $G=(V,E)$, find an ordering $v_1, \dots, v_n$ of V minimizing $\max_{1 \le i < n} E(R_i, V \setminus R_i)$ where $R_i = \{v_1, \dots, v_i\}$.

💡1 $OPT_{MCLA} \geq$ min bisection.

Proof: for every ordering $v_1, \dots, v_n$,

$$E(R_{n/2}, V \setminus R_{n/2}) \geq \text{min bisection}. \quad \Rightarrow 💡$$

for an OPT ordering: $OPT_{MCLA} \geq E(S_{n/2}, V \setminus S_{n/2})$ ☒

Algorithm for MCLA on $G=(V,E)$:

if $|V| > 2$
  find $\frac{1}{3}$-balanced cut $(S, V \setminus S)$ by the pseudo-approx. alg.
  recursively solve MCLA for S, and for $V \setminus S$
  output MCLA(S), MCLA($V \setminus S$)
else
  output V (in any order)

Theorem: The approximation ratio of the alg. is $O(\log^2 n)$.

Proof: 💡2 The depth of the recursion is $O(\log n)$.
obvious – balanced cuts

💡3 On the first level of recursion,
$$|E(S, V \setminus S)| = O(\log n) \cdot B \leq O(\log n) \cdot OPT$$
(min. bisection; 💡1)

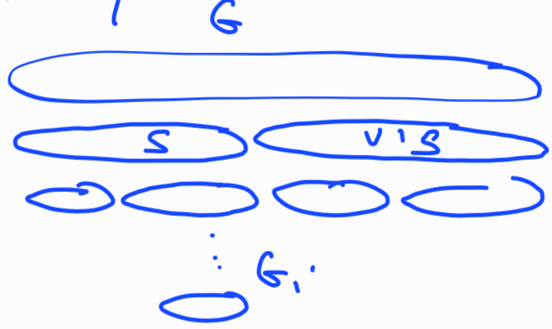
obvious from the properties of the pseudoapprox. alg. for $\frac{1}{3}$-balanced cuts.

Flows, …

6/1

We would like similar bound in any recursion level

Consider a subgraph G_i on which the algorithm was called in the recursion.



Let B_i be the min. bisection of G_i, and let OPT_i be the cost of the optimal LA in G_i.

Then $B_i \overset{\text{by ※1}}{\le} OPT_i \le OPT$ (every solution for G gives a solution for G_i of the same or better cost.)

For every G_i, the cost of the $\frac{1}{3}$-balanced cut $(S_i, V \setminus S_i)$ is at most $|E(S_i, V \setminus S_i)| = O(\log n) \cdot B_i = O(\log n) \cdot OPT$

Thus, for the final ordering $v_1, \dots, v_n$ constructed by the algorithm, and for every i,

$$E(\{v_1, \dots, v_i\}, \{v_{i+1}, \dots, v_n\}) \le \underset{\substack{\uparrow \\ \text{recursion depth}}}{O(\log n)} \cdot \underset{\substack{\uparrow \\ \text{cost of a single recur. call}}}{O(\log n)} \cdot OPT$$

$$= O(\log^2 n) \cdot OPT.$$

<u>Note</u>: 1. <u>Real</u> approximation algorithm, though based on a <u>pseudoapproximate</u> alg.

2. Works for weighted version (capacities) as well.

MINIMUM CROSSING NUMBER

Def: The crossing number $cr(G)$ of a graph G is the min. number of edge crossings in a planar drawing of G (• edges "through vertices" forbidden • no three edges share an interior point)

Problem: find a drawing with min. # of crossings.

For $c \in (0, \frac{1}{2}]$, let $b_c(G)$ denote the size of min c-balanced cut.

Theorem 1: Let $G=(V,E)$ be a $O(1)$-degree graph. Then

$$b_{1/3}^2(G) = O(cr(G)+n), \text{ i.e., } \sqrt{cr(G)+n} = \Omega(b_{1/3}(G)), \quad n=|V|.$$

Proof: Consider an optimal drawing (unknown to us).

every crossing $\rightarrow$ a new vertex

every edge $\leadsto$ several edges between the segments

$\Rightarrow$ we get a planar graph H with $n+cr(G)$ vertices

Assign weight 1 to vertices of G and 0 to other.

By Weighted Planar Separation Theorem, there is $\frac{1}{3}$-balanced cut (w.r.t. the weights) (H_1,H_2) of H of size $O(\sqrt{n+cr(G)})$.

Both sides of H contain $\geq \frac{n}{3}$ vertices of G, thus

$$b_{1/3}(G) \leq E(H_1,H_2) = O(\sqrt{n+cr(G)}).$$

Corollary: $hb^2(G) = O(cr(G)+n)$, where $hb(G) = \max\limits_{H \subseteq G} b_{1/3}(H)$ (subgraphs).

Proof: Consider subgraph H of G s.t. $hb(G) = b_{1/3}(H)$. Then ($n' = |V(H)|$)

$$hb^2(G) = b_{1/3}^2(H) = O(cr(H)+n') = O(cr(G)+n)$$

Drawing algorithm $DC(G)$ $\quad G=(V,E)$

- recursively partition V by $\frac{1}{4}$-balanced cuts (pseudo approx w.r.t $\frac{1}{3}$-balan.) $\dashrightarrow$ ordering of vertices
- put the vertices on a circle arc by this order
- draw edges as straight line segments.

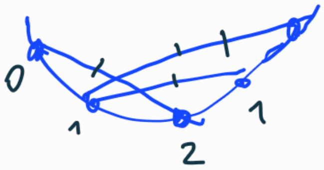
Let $DC(G)$ denote also the number of crossings in the corresponding drawing.

Theorem: $DC(G) = \underset{C_2}{O}(\log^4 n) \cdot (cr(G) + n)$.

Proof: by induction on number of vertices.

induction base: $n = O(1)$ ... obvious

induction step: For a drawing of G constructed by the $DC(G)$ algorithm, let $\ell(G)$ denote the max. number of edges going "over" a vertex of G.



For an appropriate constant C_1

$\ell(G) \le C \cdot hb(G) \cdot \log^2 n$ where $hb(G)$ is the largest $\frac{1}{3}$-balanced cut over all subgraphs of G ("hereditary" bisection)

Proof (idea): as in MinCutLin.Arr. $\square$

by construction, the following holds:

$$DC(G) \le \underset{\substack{\uparrow \\ \text{every 2 edges in the cut can cross}}}{C_1^2 \cdot \log^2 n \cdot b^2(G)} \underbrace{+ C_1 \cdot \log n \cdot b(G)}_{\substack{\text{every edge in the cut can cross at most} \\ \ell(G_1) \text{ edges from } G_1 \text{ and } \ell(G_2) \text{ from } G_2}} \cdot (\ell(G_1) + \ell(G_2)) + DC(G_1) + DC(G_2)$$

by 💡 and ind. ass.

$$\le C_1^2 \cdot \log^2 n \; hb^2(G) + C_1 \cdot \log n \cdot hb(G) \cdot 2\left(C \cdot hb(G) \cdot \log^2 \frac{2n}{3}\right)$$

$$C_2 \cdot \left(\log^4 \frac{2n}{3}\right)\left(cr(G_1) + cr(G_2) + n\right)$$

by Corollary

$$= C_1^2 \cdot \log^2 n \cdot C_3 \, (cr(G) + n) +$$

$$C_1 \cdot \log n \cdot C_3 \cdot (cr(G) + n) \cdot 2 \cdot \log^2 \frac{2n}{3} +$$

$$C_2 \cdot \log^3 n \cdot \left(\log n - \log \frac{3}{2}\right)(cr(G) + n)$$

$$\le C_2 \cdot \log^4 n \cdot (cr(G) + n)$$ for C_2 sufficiently large $\square$ 6/4