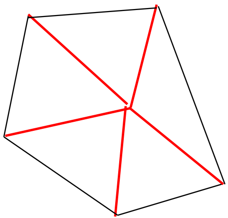
Computing an approximate minimum degree spanning tree

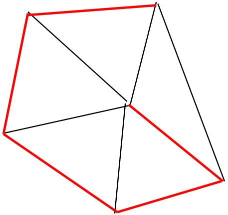
R. Inkulu
http://www.iitg.ac.in/rinkulu/

Minimum degree spanning tree (MDST): definition

Given an undirected graph $G(V, E)$ with $|V| = n$, find a spanning tree whose maximal degree is the smallest among all spanning trees in G.



a spanning tree



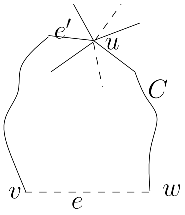
a minimum degree spanning tree

Hardness

It is NP-hard to decide whether a given graph has a minimum-degree spanning tree of maximum degree d:

$$\text{HAM-PATH} \leq_m \text{MDST}$$

Algorithm based on local search



(i) start with an arbitrary spanning tree T

(ii) while (true)

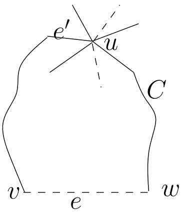
for some edge $e(v,w) \in G - T$, if the degree of some node u located on the unique cycle C in $T \cup \{e\}$ can be reduced

then $T \leftarrow T \cup \{e\} - \{e'\}$ where e' incident to u and located on C; continue

else break

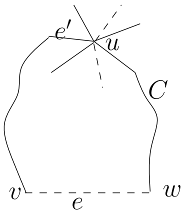
(iii) return a *locally optimal tree LOT* (i.e., when no *local improvements* are possible)

Few constraints in choosing u



Always choose a u so that it satisfies:

Few constraints in choosing u

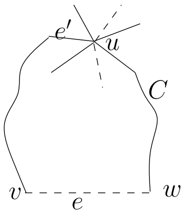


Always choose a u so that it satisfies:

- $\max(d_T(v), d_T(w)) \leq d_T(u) - 2$

  — otherwise, $\max(d_T(u), d_T(v), d_T(w))$ does not improve

Few constraints in choosing u



Always choose a u so that it satisfies:

- $\max(d_T(v), d_T(w)) \leq d_T(u) - 2$

  — otherwise, $\max(d_T(u), d_T(v), d_T(w))$ does not improve

- $\Delta_T \geq d(u) \geq \Delta_T - (\lg n) + 1$

  — helps in analysis

Key idea in upper bounding the approximation

For an optimal tree with maximum degree Δ^*, $\Delta^* \leq \Delta_{LOT}$

Key idea in upper bounding the approximation

For an optimal tree with maximum degree Δ^*, $\Delta^* \leq \Delta_{LOT} \leq ?$

Key idea in upper bounding the approximation

For an optimal tree with maximum degree Δ^*, $\Delta^* \leq \Delta_{LOT} \leq ?$

From LOT, choose a specific set S of nodes so that

Key idea in upper bounding the approximation

For an optimal tree with maximum degree Δ^*, $\Delta^* \leq \Delta_{LOT} \leq ?$

From LOT, choose a specific set S of nodes so that

$$\begin{aligned}
& l \\
& \leq \text{average degree of nodes in } S \text{ in any spanning tree of } G \\
& \leq \text{maximum degree of nodes in } S \text{ in any spanning tree of } G \\
& \leq \Delta^*
\end{aligned}$$

Key idea in upper bounding the approximation

For an optimal tree with maximum degree Δ^*, $\Delta^* \leq \Delta_{LOT} \leq ?$

From LOT, choose a specific set S of nodes so that

l

$\leq$ average degree of nodes in S in any spanning tree of G

$\leq$ maximum degree of nodes in S in any spanning tree of G

$\leq \Delta^*$

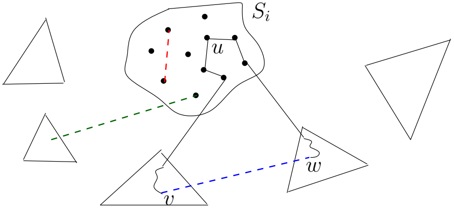
After defining such a set S, we express l as a function of Δ_{LOT}, so that to upper bound Δ_{LOT} in terms of Δ^*.

Bounding the apprx factor: choosing the set S

For every r, let S_r be the set of vertices with degree $\geq r$ in LOT. We show that there exists an i such that the average degree of vertices in S_{i-1} in any spanning tree of G is lower bounded by $\frac{\Delta_{LOT} - \lg n}{2}$. Hence, $\frac{\Delta_{LOT} - \lg n}{2} \leq \Delta^*$.

Bounding the apprx factor: choosing the set S

For every r, let S_r be the set of vertices with degree $\geq r$ in LOT. We show that there exists an i such that the average degree of vertices in S_{i-1} in any spanning tree of G is lower bounded by $\frac{\Delta_{LOT} - \lg n}{2}$. Hence, $\frac{\Delta_{LOT} - \lg n}{2} \leq \Delta^*$.

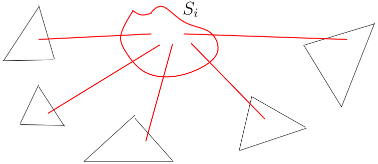


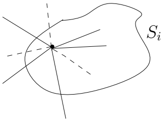
due to local optimality of LOT, $\max(d(v), d(w)) \geq i - 1$

- Let F be the set of components in LOT when vertices in S_i are removed. Also, let $|F| = t$.

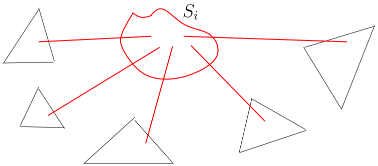
  Every edge in $G - LOT$ except those ones that are between the vertices of F incident to some vertex in S_{i-1}.

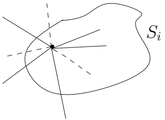
Bounding the apprx factor (cont)





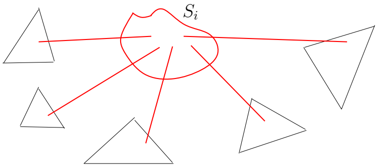
Bounding the apprx factor (cont)

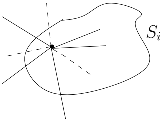




- $\Delta^* \geq$ average degree of vertices in S_{i-1} in any spanning tree of G is $\geq \frac{t+|S_i|-1}{|S_{i-1}|}$

Bounding the apprx factor (cont)

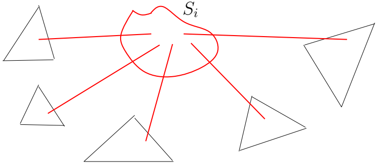


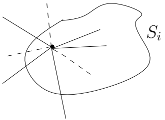


- $\Delta^* \geq$ average degree of vertices in S_{i-1} in any spanning tree of G is $\geq \frac{t+|S_i|-1}{|S_{i-1}|}$

* $t \geq i|S_i| - 2(|S_i| - 1)$ (since LOT is a tree, at most $|S_i| - 1$ edges are between vertices of S_i in LOT)

Bounding the apprx factor (cont)



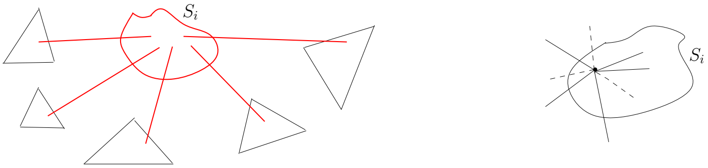


- $\Delta^* \geq$ average degree of vertices in S_{i-1} in any spanning tree of G is $\geq \frac{t+|S_i|-1}{|S_{i-1}|}$

* $t \geq i|S_i| - 2(|S_i| - 1)$ (since LOT is a tree, at most $|S_i| - 1$ edges are between vertices of S_i in LOT)

* There exists a $i \in [\Delta_{LOT} - \lg_2 n + 1,\ \Delta_{LOT}]$ such that $\frac{|S_{i-1}|}{|S_i|} \leq 2$. (suppose $\nexists$ such a i; then $|S_{\Delta_{LOT}-\lg n}| > n|S_{\Delta_{LOT}}|$; since $\forall_i\ |S_i| \geq 1$, $|S_{\Delta_{LOT}-\lg n}| > n$)

Bounding the apprx factor (cont)



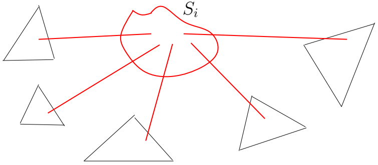
- $\Delta^* \geq$ average degree of vertices in S_{i-1} in any spanning tree of G is $\geq \frac{t+|S_i|-1}{|S_{i-1}|}$

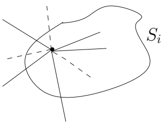
* $t \geq i|S_i| - 2(|S_i| - 1)$ (since LOT is a tree, at most $|S_i| - 1$ edges are between vertices of S_i in LOT)

* There exists a $i \in [\Delta_{LOT} - \lg_2 n + 1,\ \Delta_{LOT}]$ such that $\frac{|S_{i-1}|}{|S_i|} \leq 2$. (suppose $\nexists$ such a i; then $|S_{\Delta_{LOT}-\lg n}| > n|S_{\Delta_{LOT}}|$; since $\forall_i\ |S_i| \geq 1$, $|S_{\Delta_{LOT}-\lg n}| > n$)

Hence, $\Delta^* \geq \frac{t+|S_i|-1}{|S_{i-1}|} \geq \frac{(i-1)|S_i|+1}{2|S_i|} > \frac{(i-1)}{2}$

Bounding the apprx factor (cont)





- $\Delta^* \geq$ average degree of vertices in S_{i-1} in any spanning tree of G is $\geq \frac{t+|S_i|-1}{|S_{i-1}|}$

* $t \geq i|S_i| - 2(|S_i| - 1)$ (since LOT is a tree, at most $|S_i| - 1$ edges are between vertices of S_i in LOT)

* There exists a $i \in [\Delta_{LOT} - \lg_2 n + 1,\ \Delta_{LOT}]$ such that $\frac{|S_{i-1}|}{|S_i|} \leq 2$. (suppose $\nexists$ such a i; then $|S_{\Delta_{LOT}-\lg n}| > n|S_{\Delta_{LOT}}|$; since $\forall_i\ |S_i| \geq 1$, $|S_{\Delta_{LOT}-\lg n}| > n$)

Hence, $\Delta^* \geq \frac{t+|S_i|-1}{|S_{i-1}|} \geq \frac{(i-1)|S_i|+1}{2|S_i|} > \frac{(i-1)}{2}$

For any $i \geq \Delta_{LOT} - \lg n + 1$, $\Delta_{LOT} \leq 2\Delta^* + \lg_2 n$.

Algo terminates: polynomial time algo

Let $\phi(T) = \sum_{v \in V} 3^{d_T(v)}$ be the potential of any tree T, for some constant $c > 2$.

Algo terminates: polynomial time algo

Let $\phi(T) = \sum_{v \in V} 3^{d_T(v)}$ be the potential of any tree T, for some constant $c > 2$.

- note that $n \leq 2.3 + (n-2)3^2 \leq \phi(T) \leq nc^{\Delta_T} \leq n3^n$.

Algo terminates: polynomial time algo

Let $\phi(T) = \sum_{v \in V} 3^{d_T(v)}$ be the potential of any tree T, for some constant $c > 2$.

- note that $n \leq 2.3 + (n-2)3^2 \leq \phi(T) \leq nc^{\Delta_T} \leq n3^n$.
- after an iteration of improvement to any tree T (resulting in a tree T'), $\phi(T') \leq (1 - \frac{2}{27n^3})\phi(T)$

  decrease in potential $= (3^i - 3^{i-1}) - (2(3^{i-1} - 3^{i-2})) = 2.3^{i-1} - 4.3^{i-2} > \frac{2}{9}3^i \geq \frac{2}{9.3^{\ln n - 1}}3^{\Delta_T} \geq \frac{2}{27n^2}3^{\Delta_T} \geq \frac{2}{27n^3}\phi(T)$

Algo terminates: polynomial time algo

Let $\phi(T) = \sum_{v \in V} 3^{d_T(v)}$ be the potential of any tree T, for some constant $c > 2$.

- note that $n \leq 2.3 + (n-2)3^2 \leq \phi(T) \leq nc^{\Delta_T} \leq n3^n$.

- after an iteration of improvement to any tree T (resulting in a tree T'), $\phi(T') \leq (1 - \frac{2}{27n^3})\phi(T)$

  decrease in potential $= (3^i - 3^{i-1}) - (2(3^{i-1} - 3^{i-2})) = 2.3^{i-1} - 4.3^{i-2} > \frac{2}{9}3^i \geq \frac{2}{9.3^{\ln n - 1}}3^{\Delta_T} \geq \frac{2}{27n^2}3^{\Delta_T} \geq \frac{2}{27n^3}\phi(T)$

- after $\frac{27}{2}n^4 \ln 3$ iterations of improvements, potential is $\leq (1 - \frac{2}{27n^3})^{\frac{27}{2}n^4 \ln 3}(n3^n) \leq (e^{-n \ln 3})(n3^n) = n$

Algo terminates: polynomial time algo

Let $\phi(T) = \sum_{v \in V} 3^{d_T(v)}$ be the potential of any tree T, for some constant $c > 2$.

- note that $n \leq 2.3 + (n-2)3^2 \leq \phi(T) \leq nc^{\Delta_T} \leq n3^n$.

- after an iteration of improvement to any tree T (resulting in a tree T'), $\phi(T') \leq (1 - \frac{2}{27n^3})\phi(T)$

  decrease in potential $= (3^i - 3^{i-1}) - (2(3^{i-1} - 3^{i-2})) = 2.3^{i-1} - 4.3^{i-2} > \frac{2}{9}3^i \geq \frac{2}{9.3^{\ln n - 1}}3^{\Delta_T} \geq \frac{2}{27n^2}3^{\Delta_T} \geq \frac{2}{27n^3}\phi(T)$

- after $\frac{27}{2}n^4 \ln 3$ iterations of improvements, potential is $\leq (1 - \frac{2}{27n^3})^{\frac{27}{2}n^4 \ln 3}(n3^n) \leq (e^{-n \ln 3})(n3^n) = n$

  hence, the number of local moves is $O(n^4)$

polynomial time $(OPT + 1)$-apprx algo is known for this problem

— not presented

Open problems

- improving apprx factor of directed version of the problem
- d-vertex/edge-connected subgraph with min-degree
- min-degree spanners
- multicriteria optimization problems: max-leaf min-degree
- relation among problems that have no multiplicative error but have small additive error

References

M. Furer and B. Raghavachari. Approximating the minimum degree spanning tree to within one from the optimal degree. SODA, 1992.

David P. Williamson and David B. Shmoys. The Design of Approximation Algorithms. Cambridge University Press, 2011.

M. Furer and B. Raghavachari. Approximating the minimum degree spanning tree to within one from the optimal degree. Journal of Algorithms, 1994.

Thanks!