Query Complexity of Metric Steiner Tree

Zihan Tan

Based on joint work with Yu Chen and Sanjeev Khanna

Finding a Solution vs Estimating the Cost

(all in the Pair Query model)

Objective	Solution	Cost
Matching	$\Omega(n^2)$ for $(O(1),\varepsilon n)$-approx [Assadi-Chen-Khanna 19]	$\tilde{O}(n/\varepsilon^{O(1)})$ for $(2,\varepsilon n)$-approx [Behnezhad 21]
Metric MST	$\Omega(n^2)$ for $O(1)$-approx [Czumaj-Sohler 09]	$\tilde{O}(n/\varepsilon^{O(1)})$ for $(1+\varepsilon)$-approx [Czumaj-Sohler 09]
Metric TSP	$\Omega(n^2)$ for $O(1)$-approx (even for (1,2) or graphic) [Chen-Khanna-T 22]	$\tilde{O}(n^{1.5})$ for 1.99999-approx (in some special cases) [Chen-Khanna-T 22]

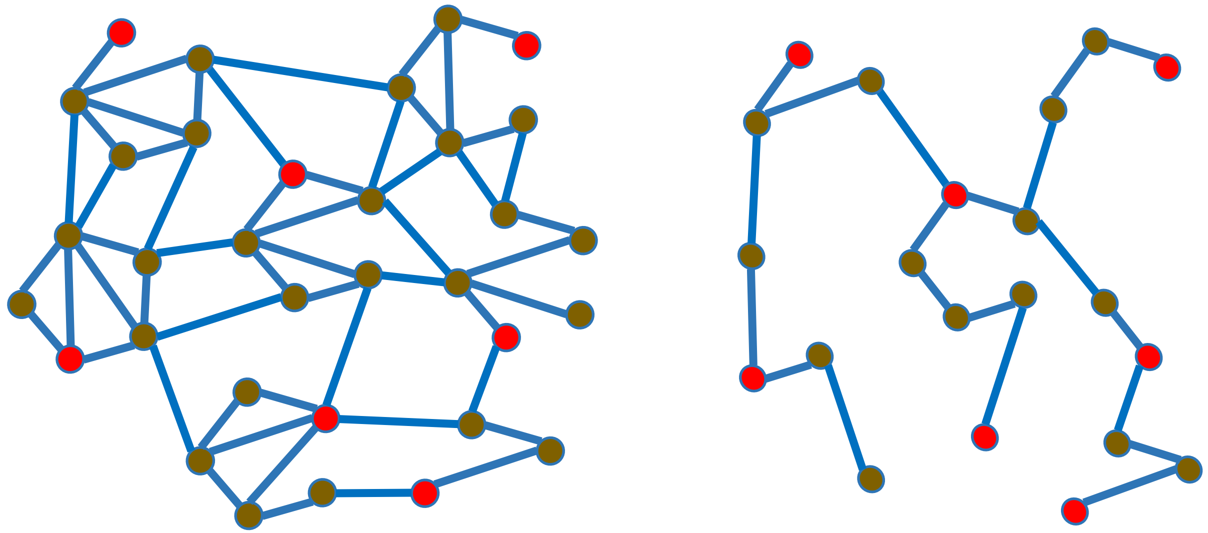
Others: Hierarchical Clustering, Earth Mover Distance, etc.

(Metric) Steiner Tree

Input: an edge-weighted graph G=(V,E), a subset of its vertices (terminals).
Output: a connected subgraph of G with minimum weight that spans all terminals.

w.l.o.g: a tree

Equivalent: a metric d on V (given by the shortest-path distance in G)

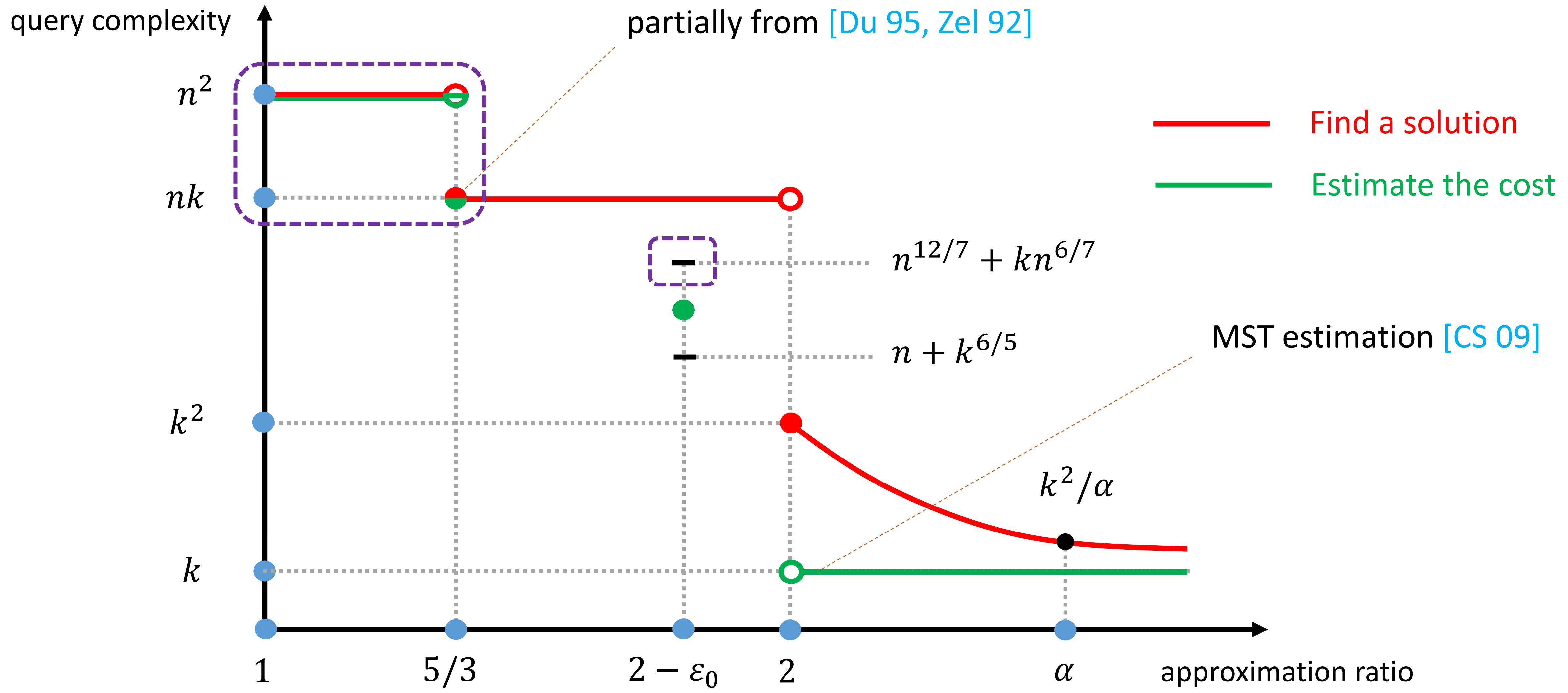


One of the most fundamental NP-hard problems

Best approximation: (ln 4 + ε)
[Byrka-Grandoni-Rothvos-Sanita 10]
[Goemans-Olver-Rothvos-Zenklusen 12]
[Traub-Zenklusen 22]

APX-hard (96/95)
[Chlebik-Chlebikova 08]

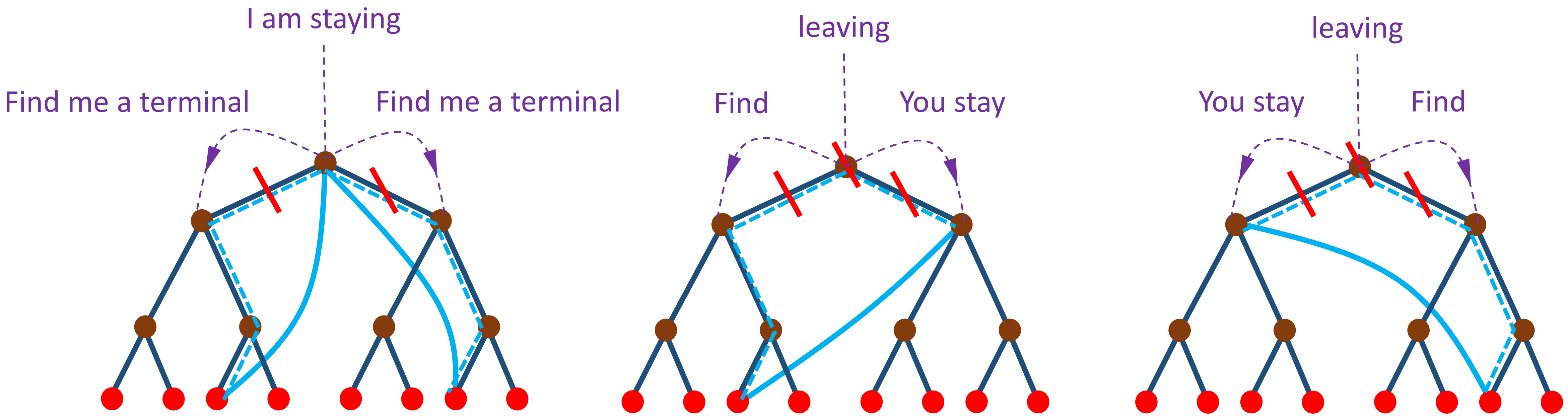
Our Results (Pair Query: input (u,v), returns d(u,v))



5/3-approximation with O(nk) queries

Thm [Du 95, Zelikovsky 92]. There exists a (5/3)-approximate Steiner Tree that only uses terminal-terminal edges and terminal-Steiner edges.

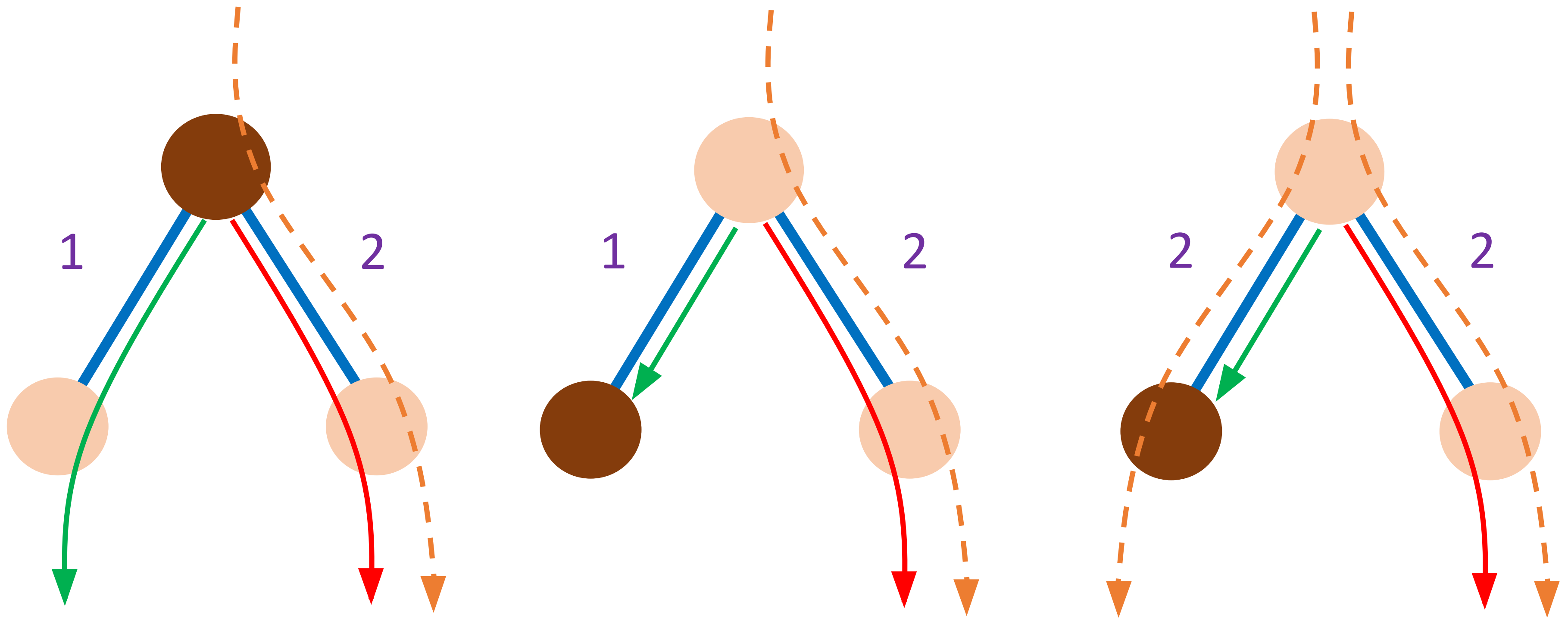
top-down randomization s.t. each tree-edge is covered 5/3 times in expectation



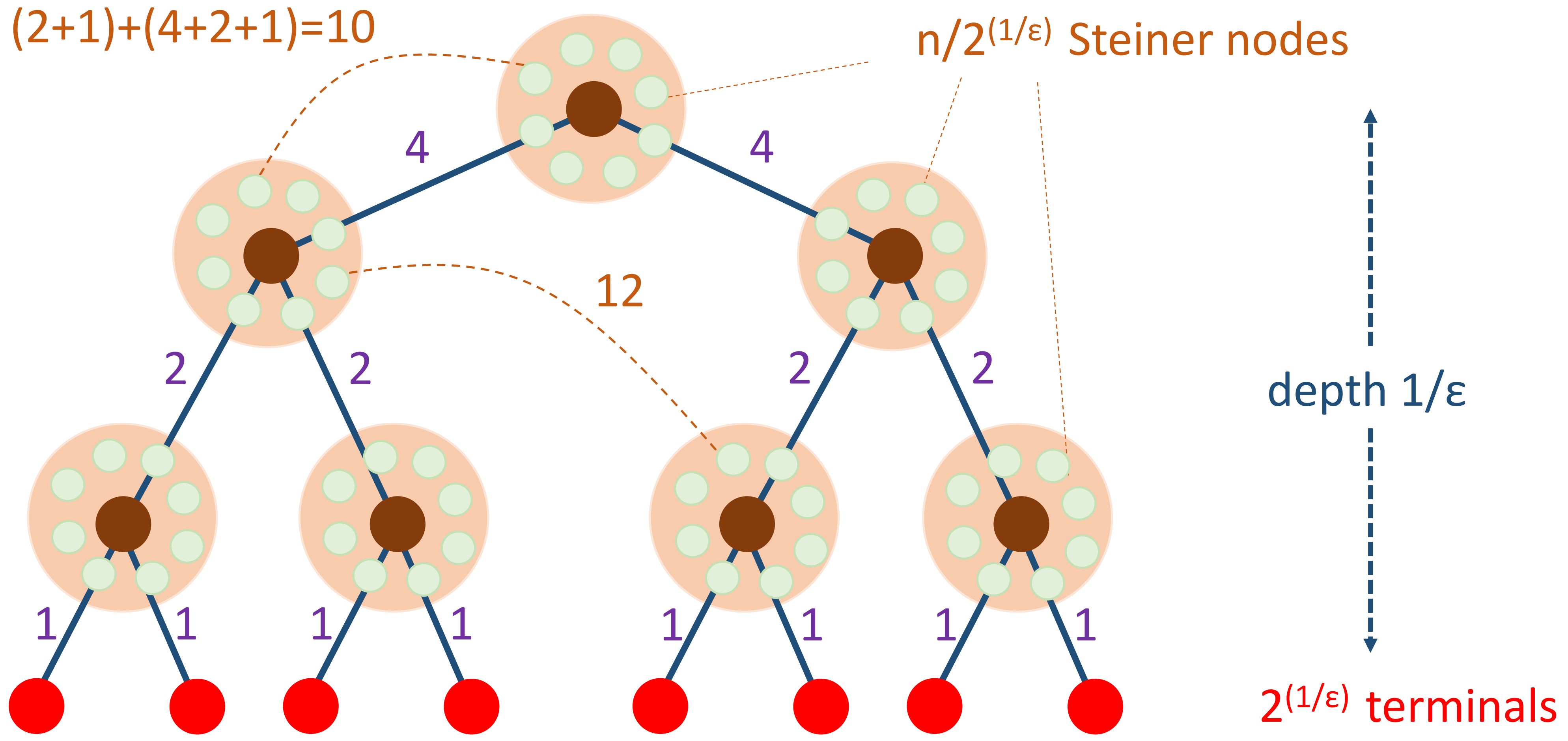
5/3-approximation with O(nk) queries

Thm [Du 95, Zelikovsky 92]. There exists a (5/3)-approximate Steiner Tree that only uses terminal-terminal edges and terminal-Steiner edges.

top-down randomization s.t. each tree-edge is covered 5/3 times in expectation



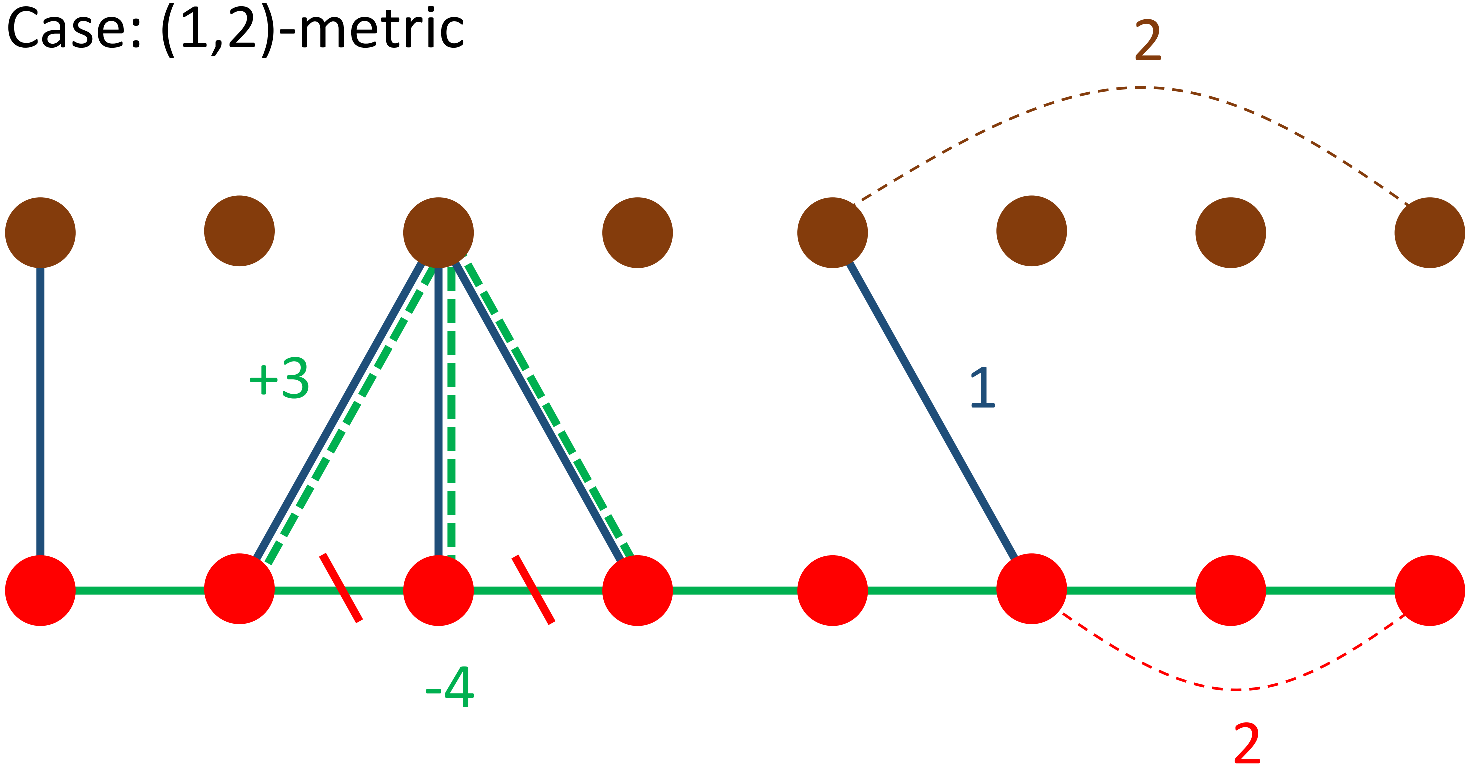
(5/3-ε)-estimation requires Ω(n^2) queries



$(2-\varepsilon_0)$-estimation with $\tilde{O}(n^{12/7}+kn^{6/7})$ queries

Idea: Start from terminal MST (2-approx), try "saving some cost" via Steiner nodes.

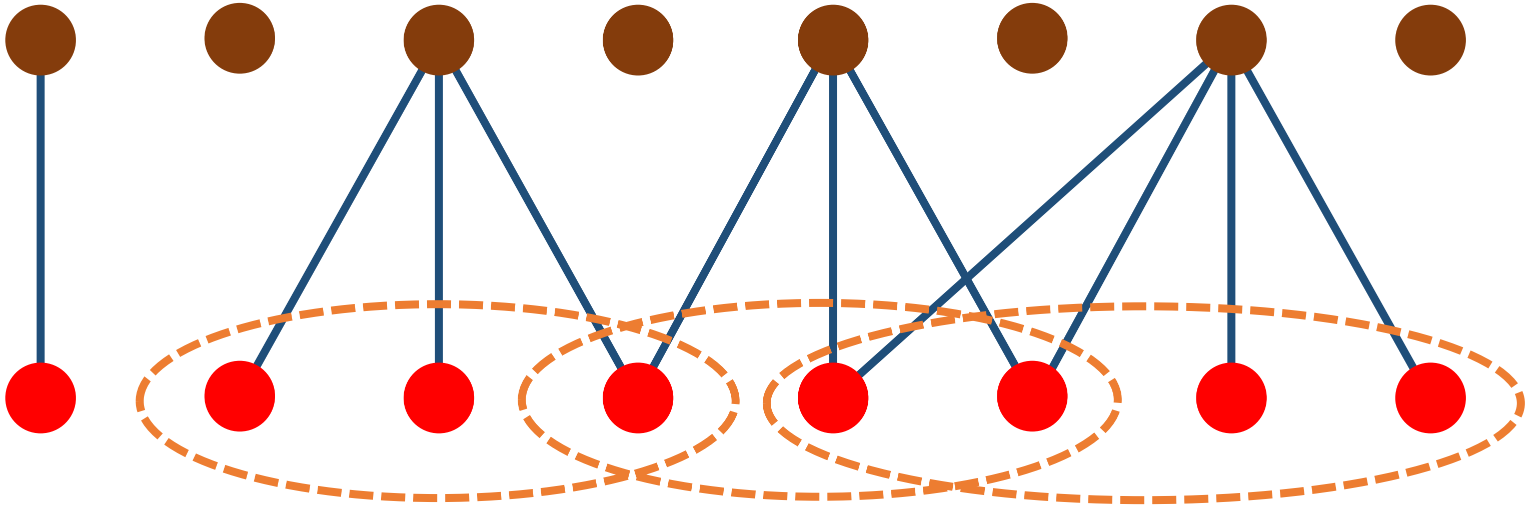
Special Case: (1,2)-metric



$(2\text{-}\varepsilon_0)$-estimation with $\tilde{O}(n^{12/7}+kn^{6/7})$ queries

Idea: Start from terminal MST (2-approx), try "saving some cost" via Steiner nodes.

Special Case: (1,2)-metric



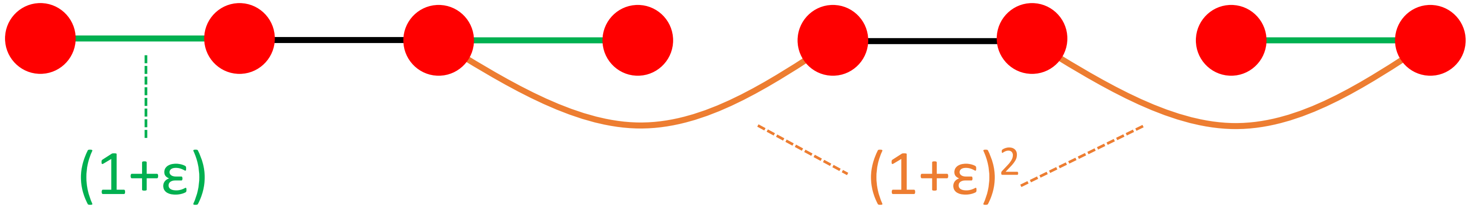
$\tilde{O}(n^{1.5})$ query for O(1)-estimation

select a collection S of sets, get saving cov(S)-|S| ⟶ equivalently: k - (set-cover size)

$(2-\varepsilon_0)$-estimation with $\tilde{O}(n^{12/7}+kn^{6/7})$ queries

Idea: Start from terminal MST (2-approx), try “saving some cost” via Steiner nodes.

General Case: Try to find saving “layer-by-layer”

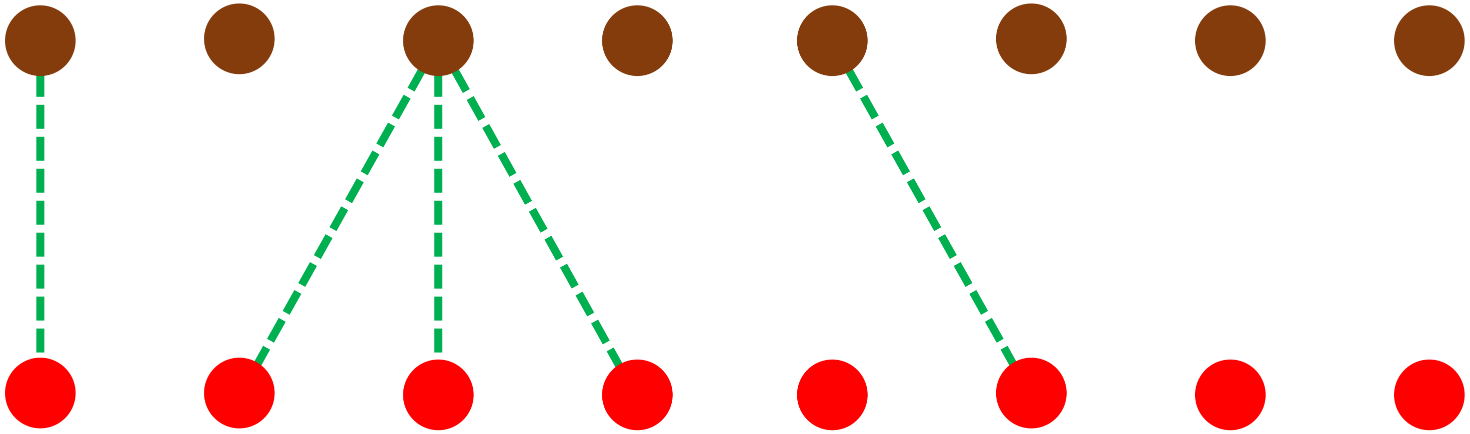


Compute terminal MST and its hierarchical structure (partition its edges into layers)

$(2\text{-}\varepsilon_0)$-estimation with $\tilde{O}(n^{12/7}+kn^{6/7})$ queries

Idea: Start from terminal MST (2-approx), try "saving some cost" via Steiner nodes.

General Case: Try to find saving "layer-by-layer"



Green Layer Saving:
only $(1+\varepsilon)/2$ edges

Compute terminal MST and its hierarchical structure (partition its edges into layers)

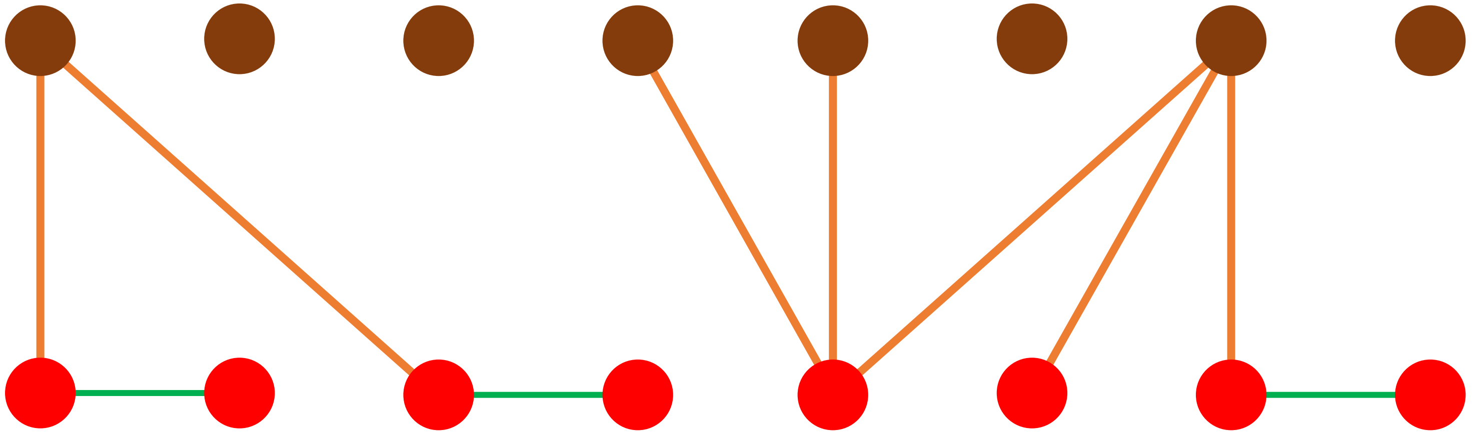
$(2-\varepsilon_0)$-estimation with $\tilde{O}(n^{12/7}+kn^{6/7})$ queries

Idea: Start from terminal MST (2-approx), try "saving some cost" via Steiner nodes.

General Case: Try to find saving "layer-by-layer"

If total saving = Ω(MST) ⟶ Steiner Tree Cost < $(1-\varepsilon_0)$ MST

If total saving = o(MST) ⟶ Steiner Tree Cost > $(0.5+\varepsilon_0)$ MST



⋮

Orange Layer Saving: only $(1+\varepsilon)^2/2$ edges (green-layer components as elements)

Green Layer Saving: only $(1+\varepsilon)/2$ edges

$\tilde{O}(n^{1.5})$

Compute terminal MST and its hierarchical structure (partition its edges into layers)

$O(k^2)$

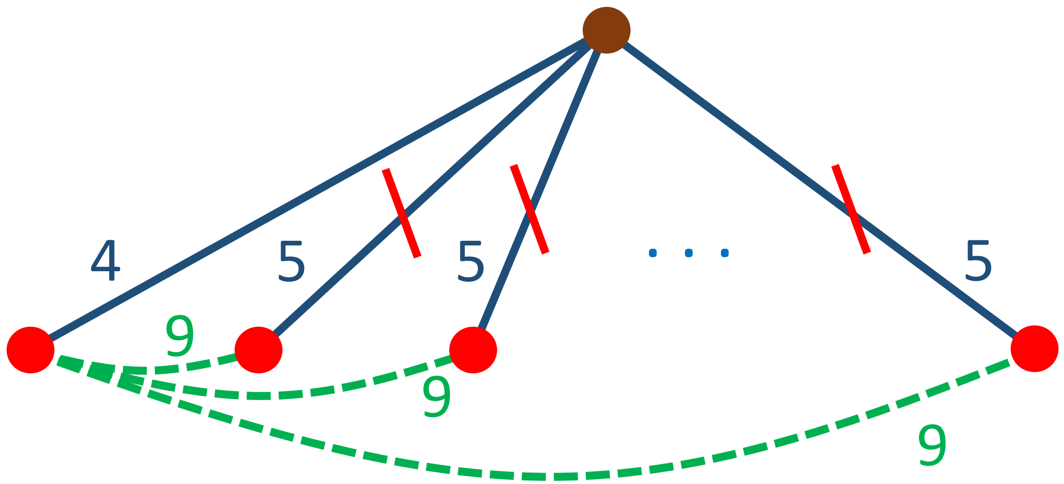
$(2\text{-}\varepsilon_0)$-estimation with $\tilde{O}(n^{12/7}+kn^{6/7})$ queries

Idea: Start from terminal MST (2-approx), try "saving some cost" via Steiner nodes.

General Case: Try to find saving "layer-by-layer"

If total saving = o(MST) ⟶ Steiner Tree Cost > $(0.5+\varepsilon_0)$ MST

Proof strategy: iteratively modify OPT Steiner Tree to a terminal spanning tree



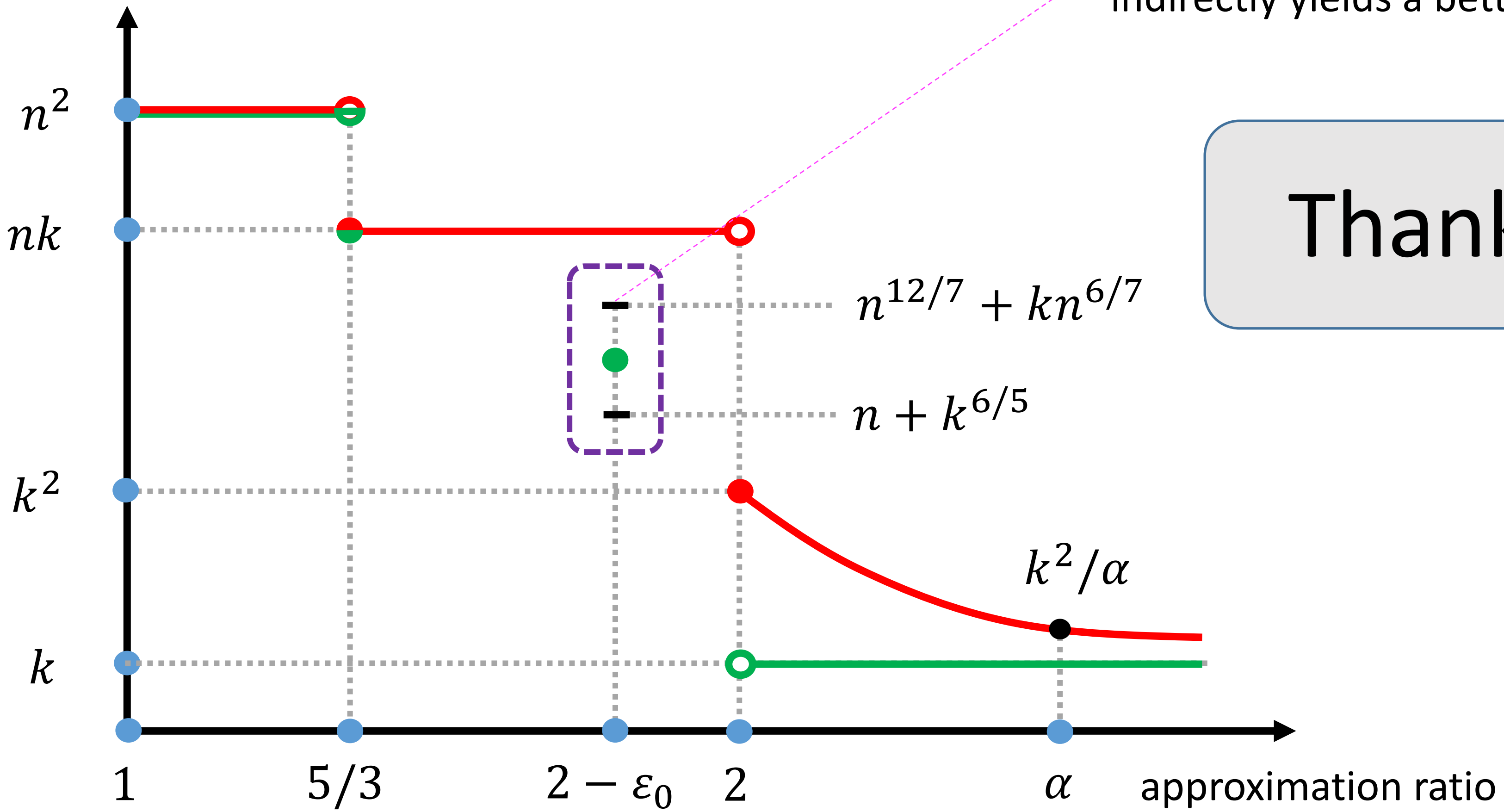
same length ⟶ discovered in saving-finding

different length ⟶ paying $(2\text{-}\varepsilon_0)$ in modification

$(2\text{-}\varepsilon_0)(1+o(1))\text{OPT} > \text{MST}$

Summary and Future Directions

Improving the query algorithm for "Set Cover" indirectly yields a better upper bound



Thanks for listening!