Efficient algorithms for Steiner Tree Problem

Jie Meng

- The Steiner Problem in Graphs,
  - by S.E.Dreyfus and R.A. Wagner, *Networks*, 1972
- A Faster Algorithm for the Steiner Tree Problem,
  - By D. Molle, S. Richter and P.Rossmanith, *STACS06*, 2006

Definitions and Backgroud

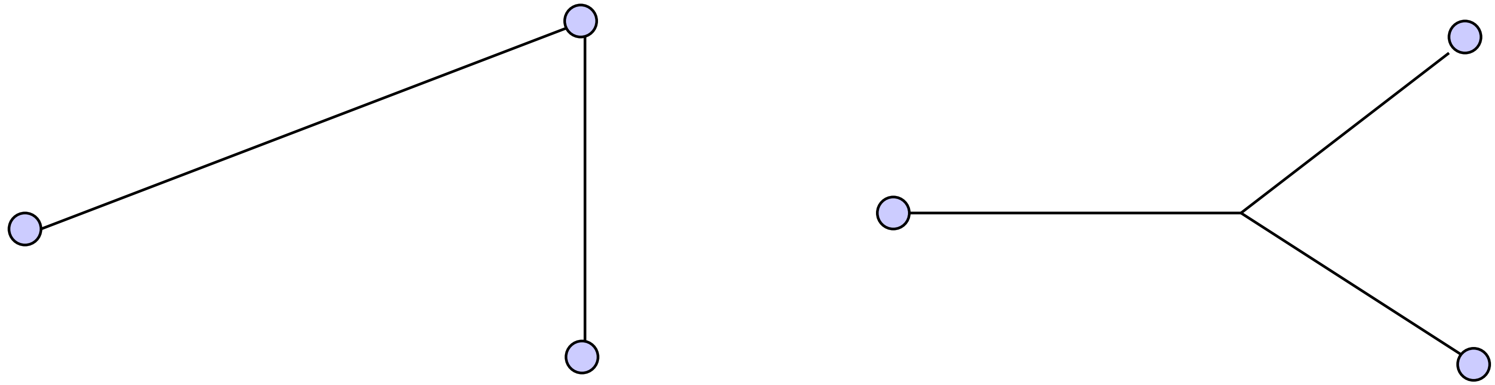
- Given a weighted graph G=(V, E, ω), weight on edges, and a steiner set $S \subseteq V$;
- A *steiner graph* of S is a connected subgraph of G which contains all nodes in S;
- The Steiner Tree problem is to find a steiner graph of S with minimum weight.
- This problem is quite important in industrial applications.

Definitions and Backgroud

- The Steiner Tree problem is NP-Complete(By Karp, 1971).
- Approximatable within 1.55 ratio, but APX-complete.
- Solvable in $O(3^k n^2)$
- Improved up to $(2+\delta)^k n^{O(\frac{1}{\delta}\ln\frac{1}{\delta})}$

Definitions and Background

- Why Steiner Tree problem is Hard?

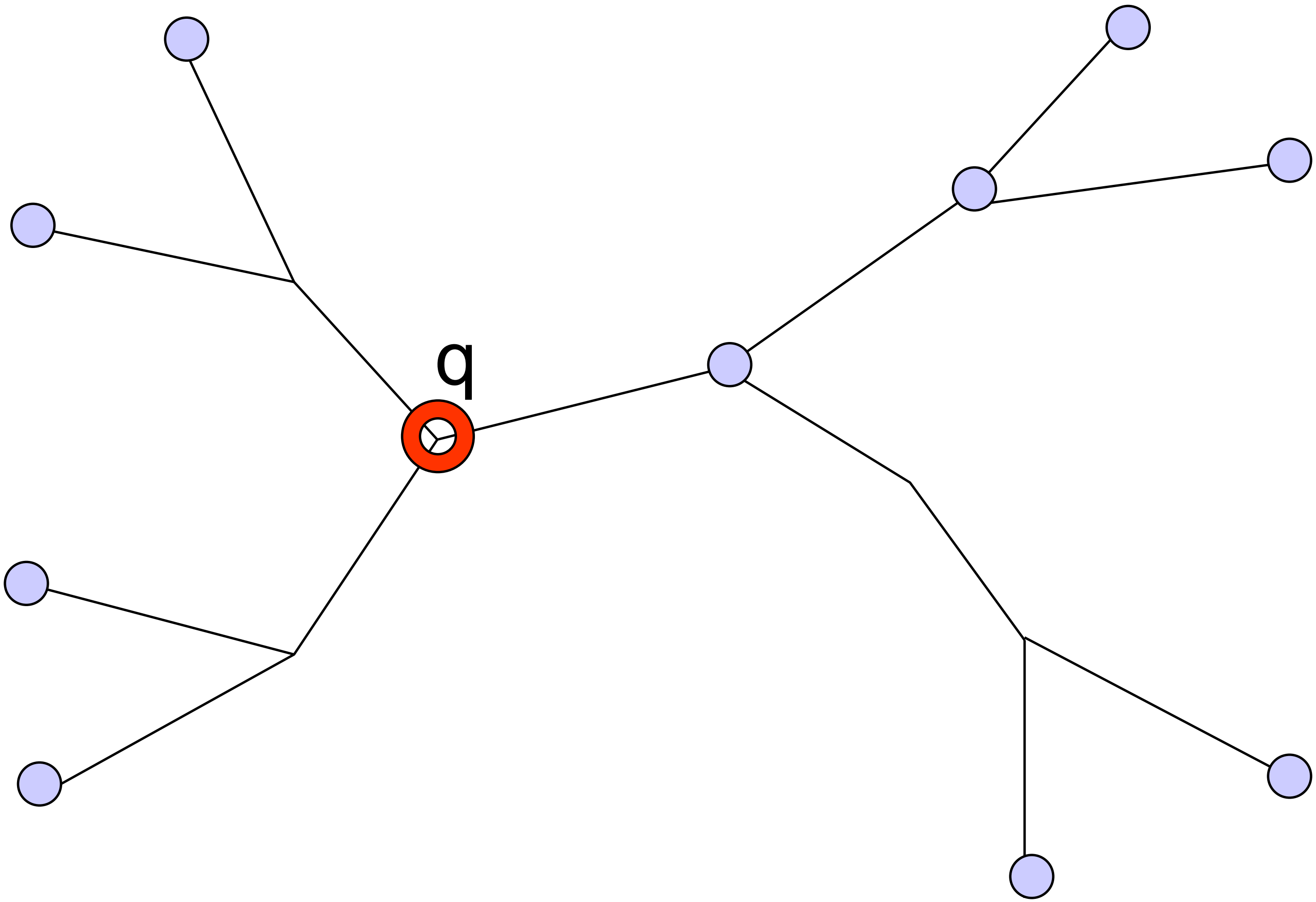


Spanning Tree VS. Steiner Tree

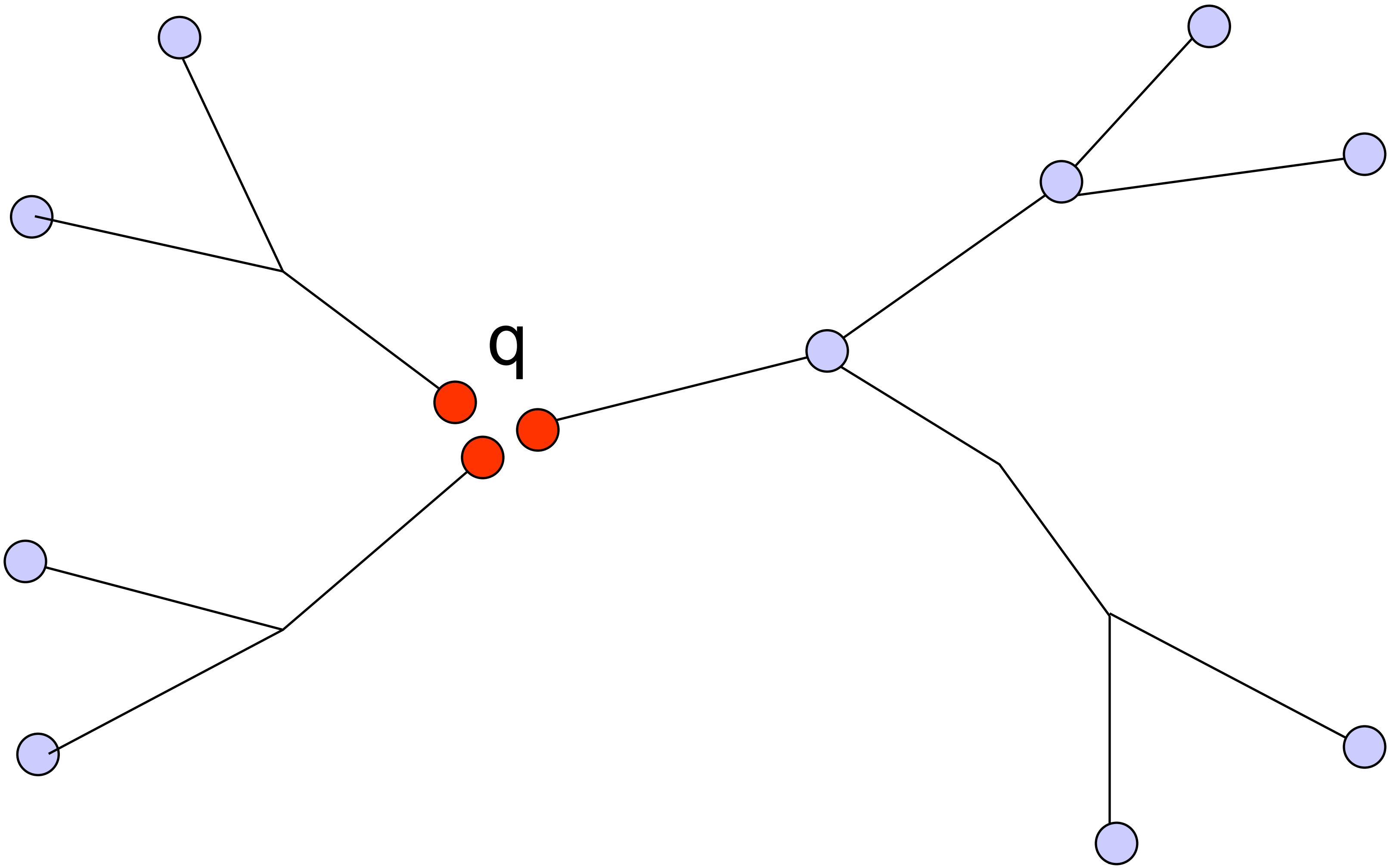
Definitions and Background

- Theorem: Let T be an optimal steiner tree for S, q be an inner node in T; If we break down T into several subtrees T_1, T_2 … in q, T_i are optimal for $V(T_i) \bigcap S + q$
- This can be proved by contradiction.

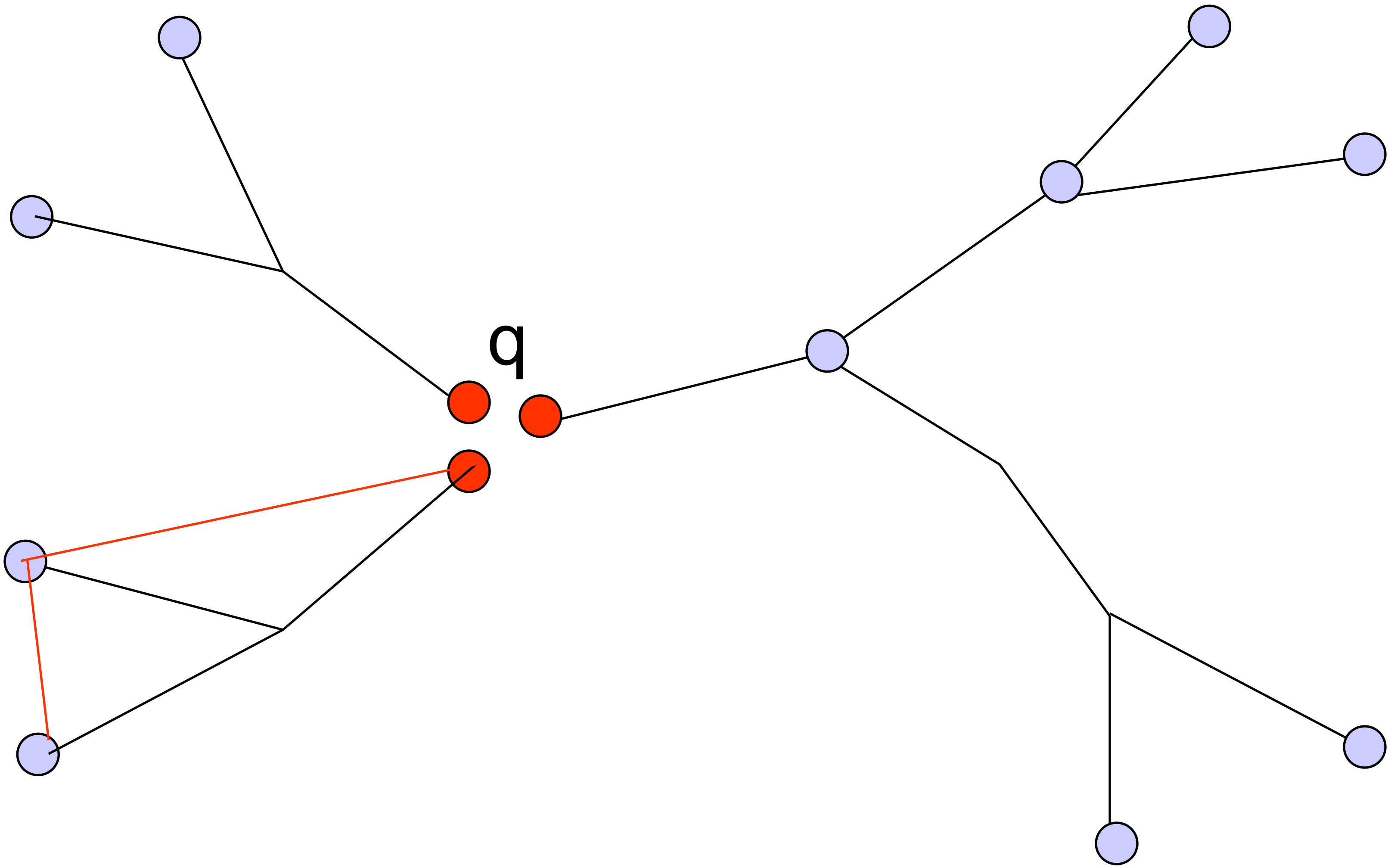
Definition and Background



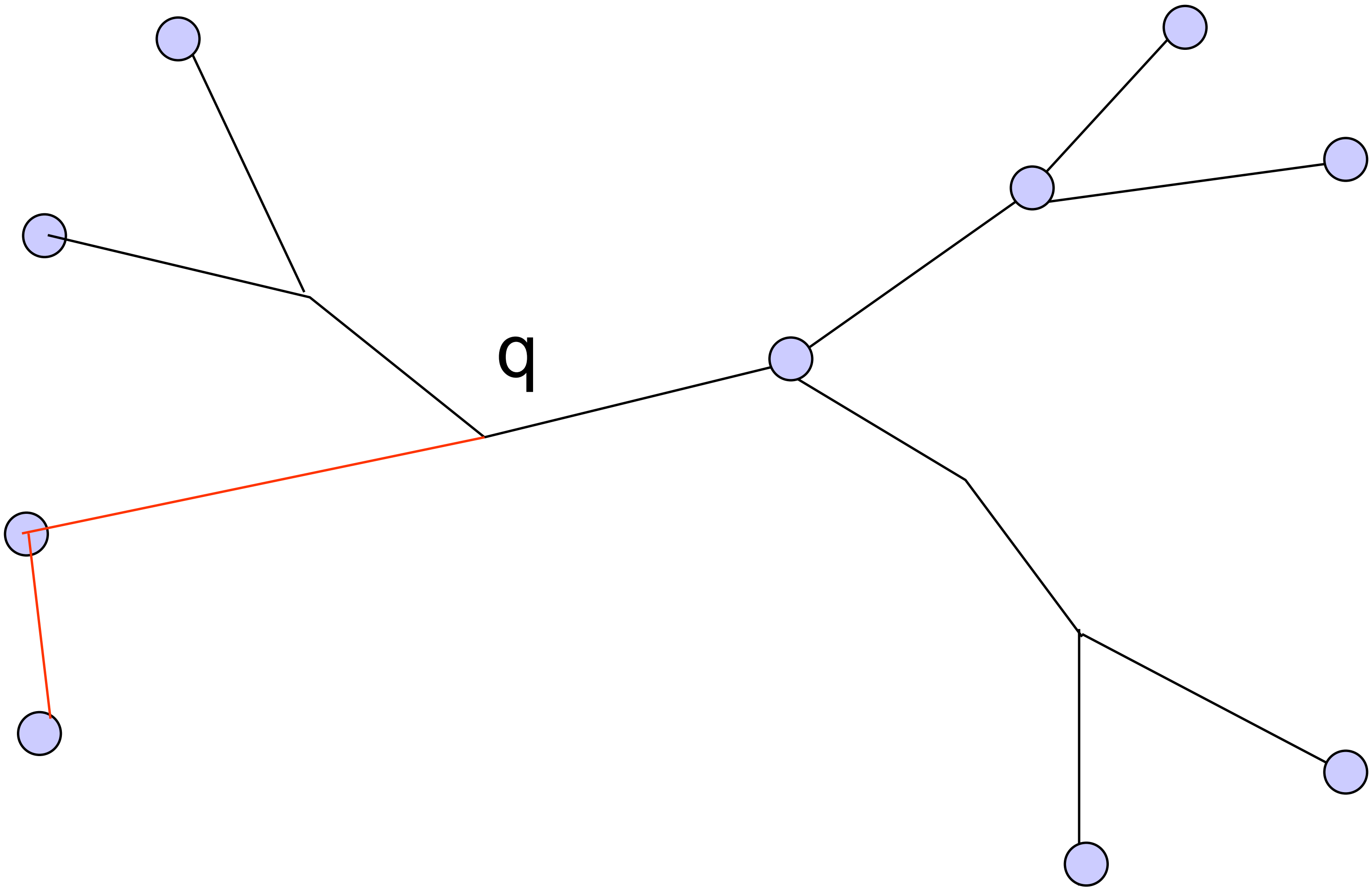
Definition and Background



Definition and Background



Definition and Background



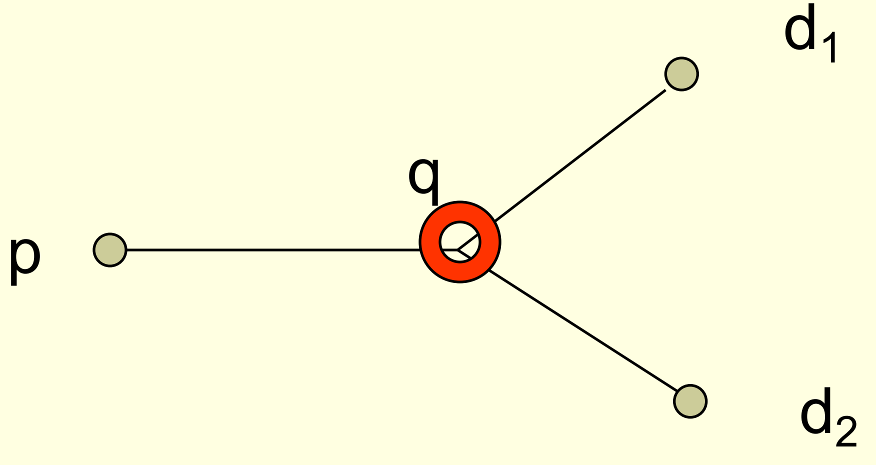
Dreyfus-Wagner's Algorithm

- Two Phrases:
  - Computing the optimal steiner trees containing D and p, where $D \subseteq S, p \in V$;
  - Computing the optimal steiner tree for S;
- Let *OPT(D, p)* denote the optimal steiner tree for D and p, *SP(u, v)* denote the shortest path between u and v in G;

Dreyfus-Wagner's Algorithm

- Computing the optimal steiner trees for D and p;
  - If |D|<3, w.l.o.g. let D={d_1, d_2}, so

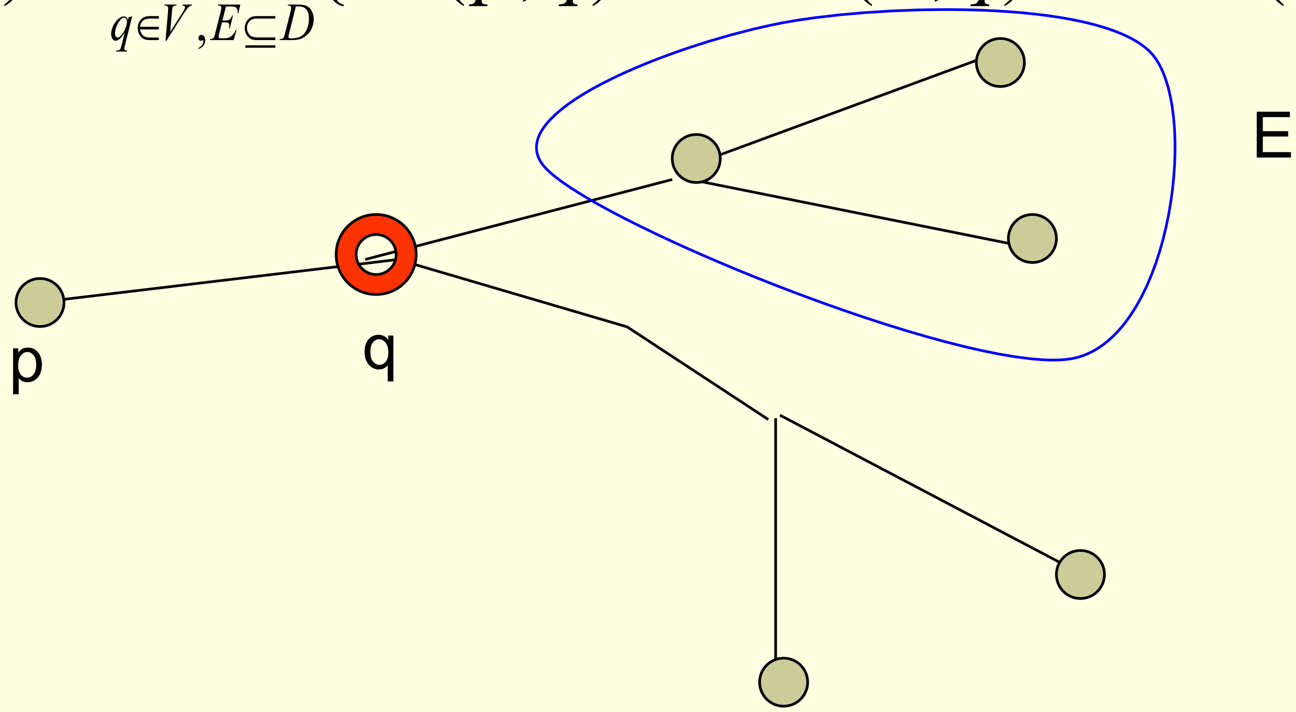
$$OPT(D,p) = \min_{q \in V}\{SP(d_1,q) + SP(d_2,q) + SP(p,q)\}$$



Dreyfus-Wagner's Algorithm

- Computing the optimal steiner trees for D and p;
  - If |D| $\geq$ 3, then

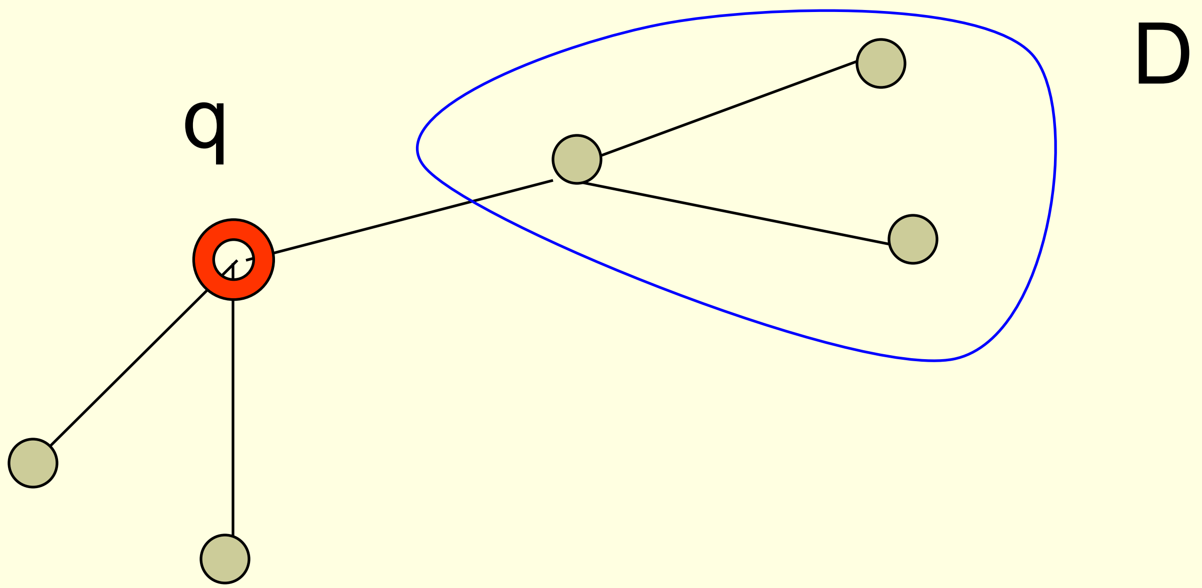
$$OPT(D,p) = \min_{q \in V, E \subseteq D} \{SP(p,q) + OPT(E,q) + OPT(D-E,q)\}$$



Dreyfus-Wagner's Algorithm

- Computing the optimal steiner tree for S;

$$OPT(S) = \min_{q \in V, D \subseteq S} \{OPT(D, q) + OPT(S - D, q)\}$$



Dreyfus-Wagner's Algorithm

- Time Complexity:
  - Phrase 1, time= $\sum_{i=3}^{k} \binom{k}{i} 2^i n^2 \le 3^k n^2$
  - Phrase 2, time= $2^k n^2$
- Total Dreyfus-Wagner's algorithm takes time $O(3^k n^2)$.

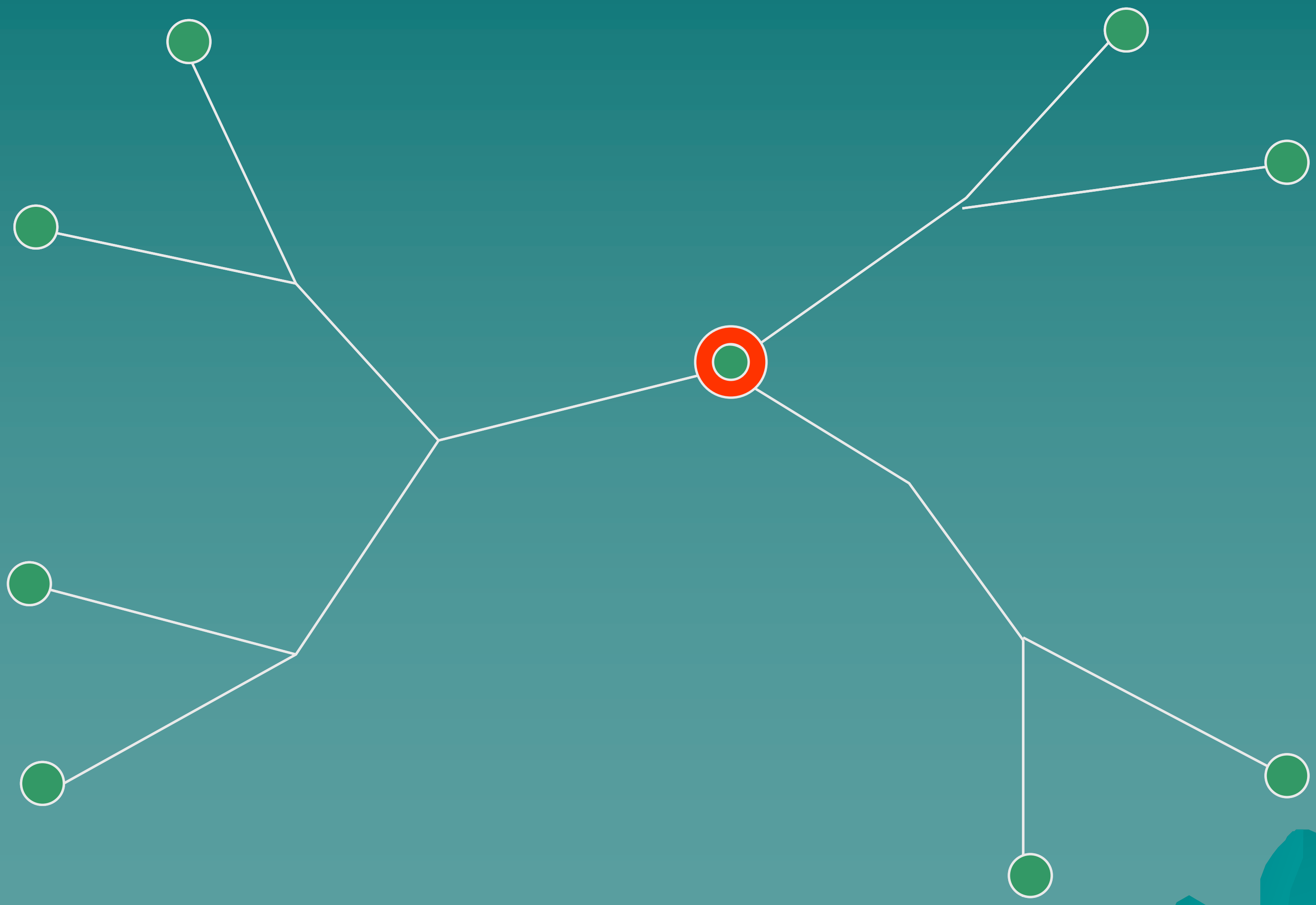
MRR’s algorithm

- The most time-consume part of Dreyfus-Wagner algorithm is phrase 1, we have to enumerate all possible subsets of every subset of S;
- Can we reduce the cost of this part?

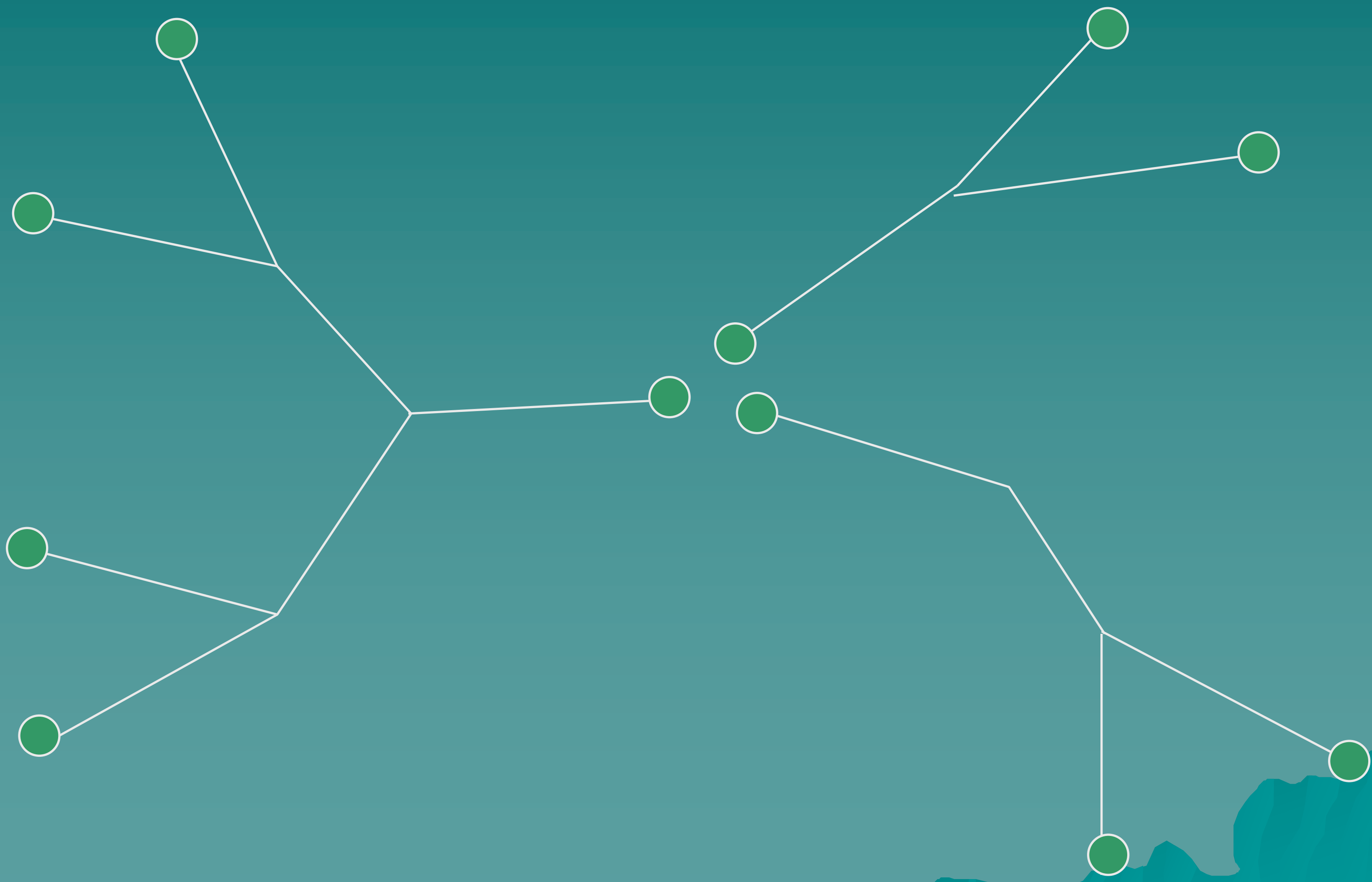
MRR's algorithm

- Break down the optimal steiner tree only in steiner node;
- Enumerate subset only of size <t;

MRR's Algorithm



MRR's Algorithm



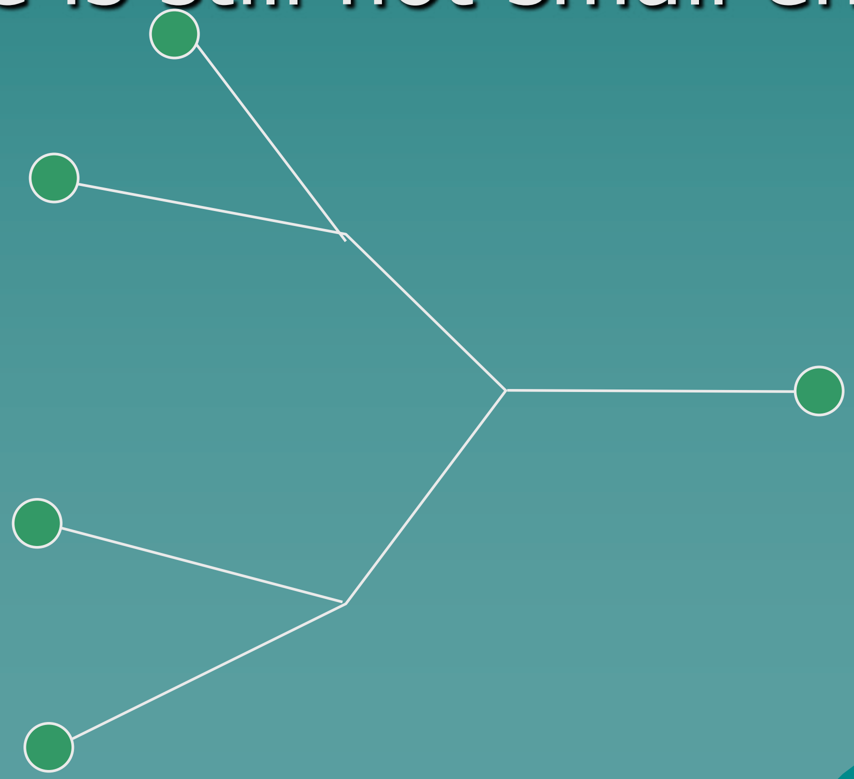
MRR's algorithm

For all subset D of S:

$$OPT(D) = \min_{\substack{E \subseteq D, |E| \leq t \\ q \in D}} \{OPT(E) + OPT(D - E + q)\}$$

MRR’s algorithm

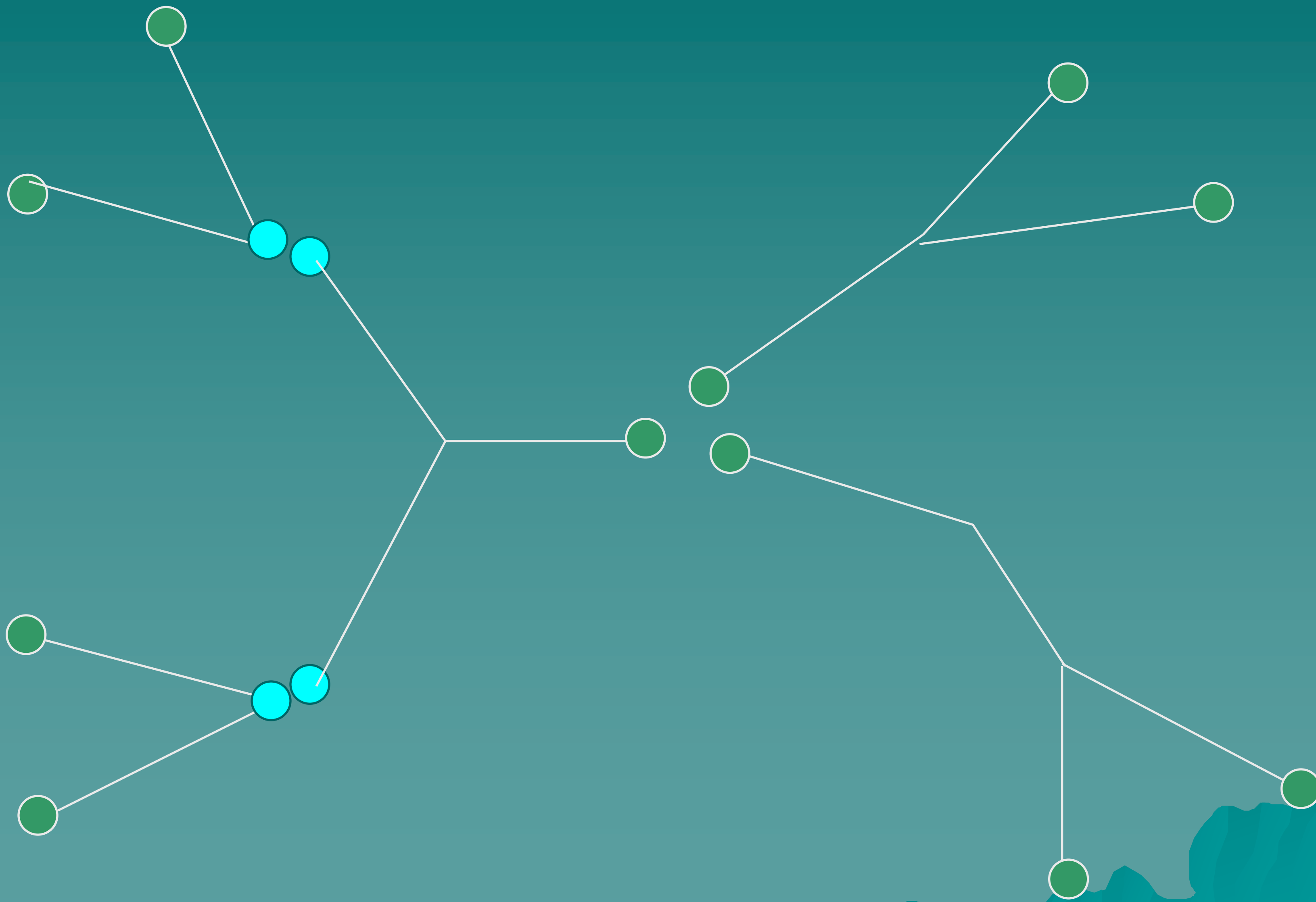
In some cases, this procedure may fail, simply because all steiner nodes have already become leaves, and the tree is still not small enough.



MRR's algorithm

Extend the steiner set

MRR’s algorithm



MRR's algorithm

Lemma Let T be an optimal steiner tree for S, t is an integer, we can add at most |S|/(t-1) many non-steiner nodes into S such that T can be divided into several subtrees of size no larger than t, and all nodes in S are leaves in those subtree.

MRR's algorithm

- Let t= c |S|;
- Enumerate all possible subset of V of size |S|/(t-1), add this set into S;
- For all subsets D of S of size less than t, call Dreyfus-Wagner algorithm to compute the optimal steiner tree for D;
- For all subsets D of S of size no less than t,

$$OPT(D) = \min_{\substack{E \subseteq D, |E| \leq t \\ q \in D}} \{OPT(E) + OPT(D - E + q)\}$$

MRR's algorithm

- The running time of this algorithm is

$$(2+\delta)^k n^{O(\frac{1}{\delta}\ln\frac{1}{\delta})}$$

Question?