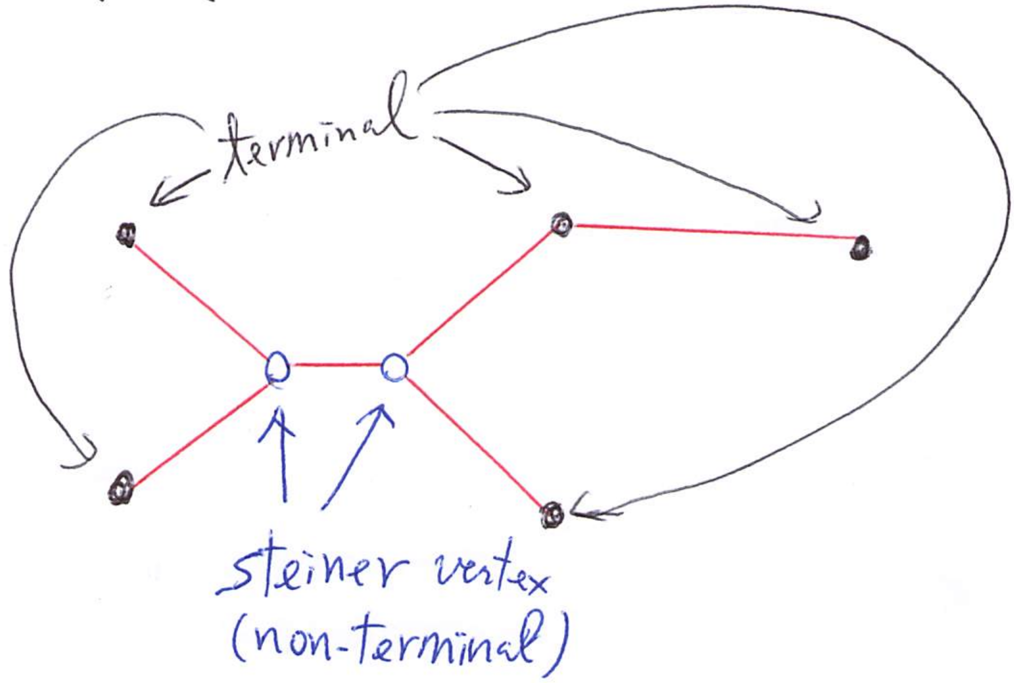
Steiner Minimal Trees (SMT)

Kun-Mao Chao @2019

- Euclidean SMT

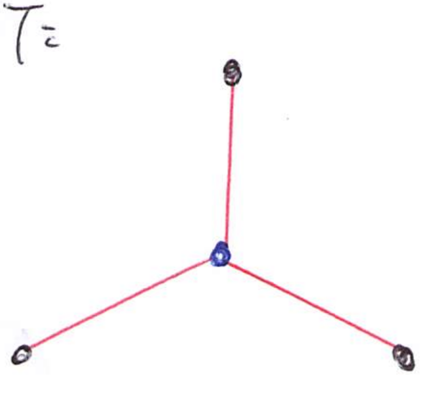


The objective is to find a tree spanning all the terminals with the minimum total cost.

Gilbert-Pollak conjecture:

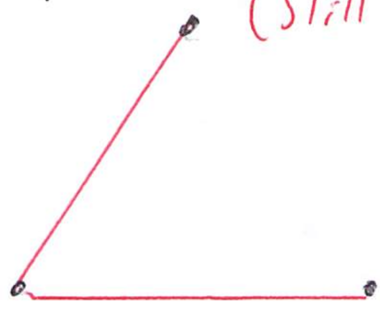
$MST/SMT \leq \frac{2}{\sqrt{3}}$ (still open!)

SMT:



$\frac{\sqrt{3}}{2} \times \frac{2}{3} \times 3 = \sqrt{3}$

MST:

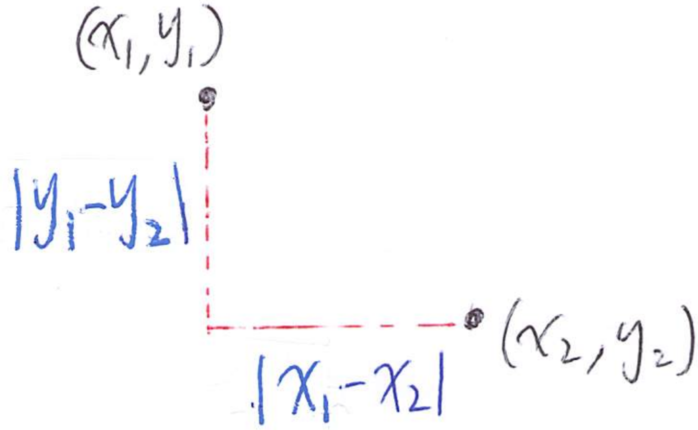


2

Kun-Mao Chao @2019

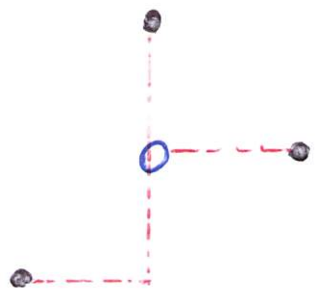
- Rectilinear SMT

L_1-norm (rectilinear distance)
Manhattan distance



$|x_1 - x_2| + |y_1 - y_2|$

SMT:



MST:

MST/SMT $\le \frac{3}{2}$ (Hwang, 1976)

Kun-Mao Chao @2019

Graph SMT

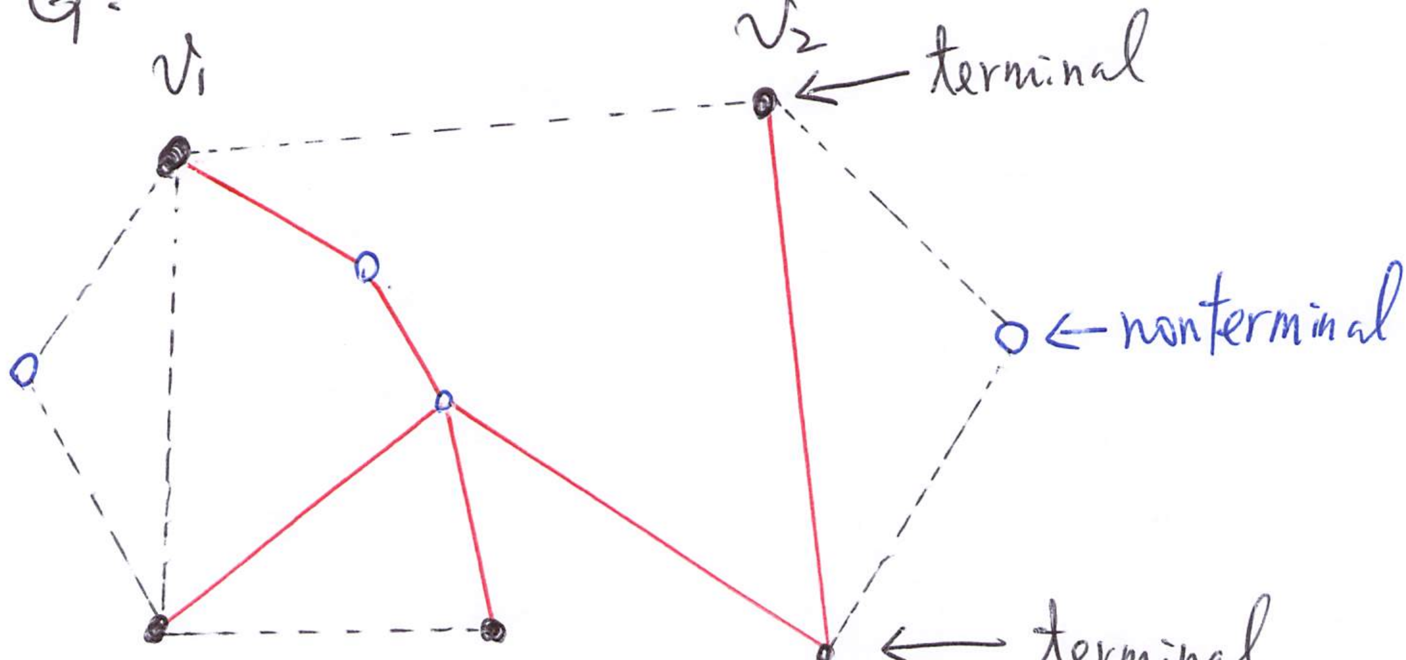
$G = (V, E, w)$

$L \subseteq V$: terminals

w: nonnegative

Find a tree T with L & minimize $w(T)$.

G:



$L = \{v_1, v_2, v_3, v_4, v_5\}$

(4)

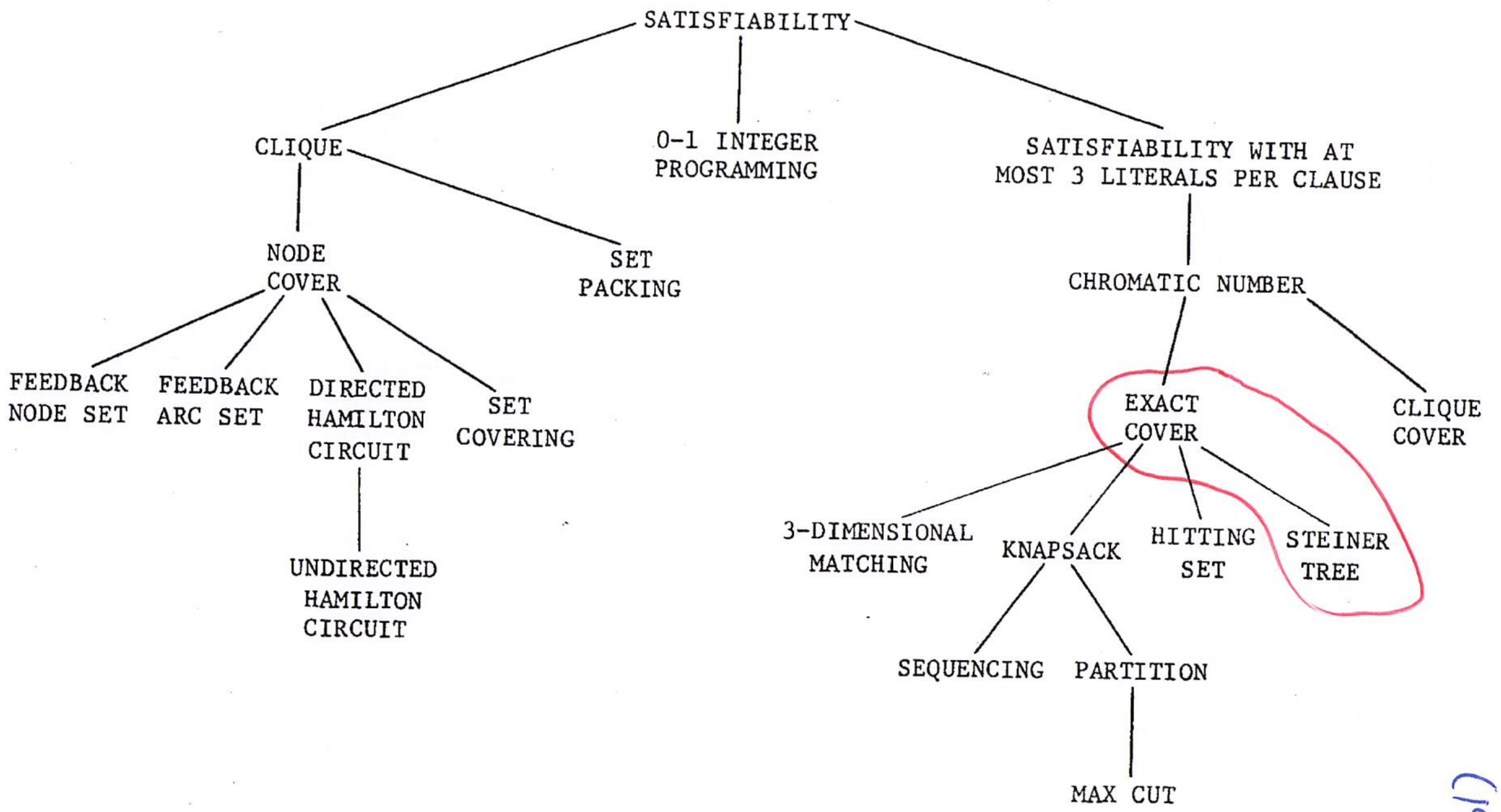


FIGURE 1 - Complete Problems

RICHARD M. KARP

(1972)

Kun-Mao Chao ©2019

Kun-Mao Chao
@2019

Reducibility among Combinatorial Problems -- Richard M. Karp (1972)

14. EXACT COVER
INPUT: family $\{S_j\}$ of subsets of a set $\{u_i,\ i = 1,2,\ldots,t\}$
PROPERTY: There is a subfamily $\{T_h\} \subseteq \{S_j\}$ such that the sets T_h are disjoint and $\cup T_h = \cup S_j = \{u_i,\ i = 1,2,\ldots,t\}$.

In short, an exact cover is "exact" in the sense that each element in $\{u_i\}$ is contained in exactly one subset in $\{T_h\}$.

16. STEINER TREE
INPUT: graph G, $R \subseteq N$, weighting function $w: A \to Z$, positive integer k
PROPERTY: G has a subtree of weight $\leq k$ containing the set of nodes in R.

EXACT COVER $\propto$ STEINER TREE

$N = \{n_o\} \cup \{S_j\} \cup \{u_i\}$

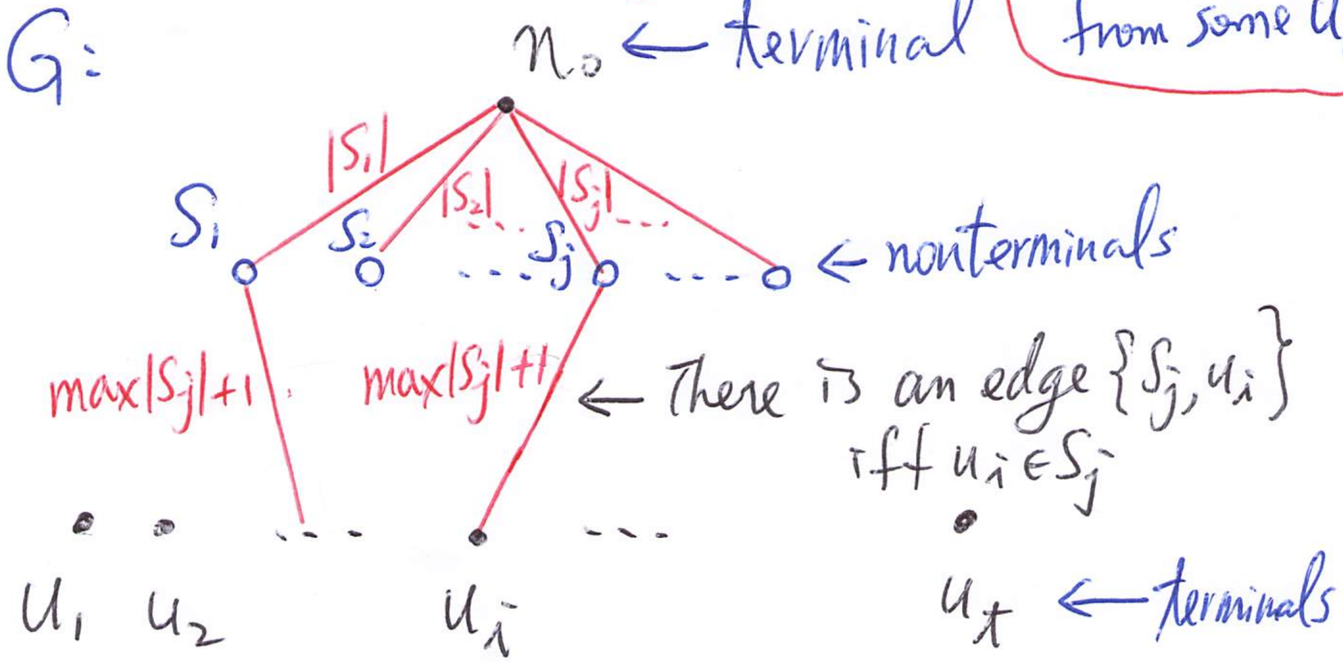
$R = \{n_o\} \cup \{u_i\}$

$A = \{\{n_o, S_j\}\} \cup \{\{S_j, u_i\} \mid u_i \in S_j\}$

$w(\{n_o, S_j\}) = |S_j|$

$w(\{S_j, u_i\}) = \cancel{0} \rightarrow \max|S_j| + 1$ ← to avoid a tree connection to some S_j back from some u_i

$k = |\{u_i\}| \times (\max|S_j| + 2)$

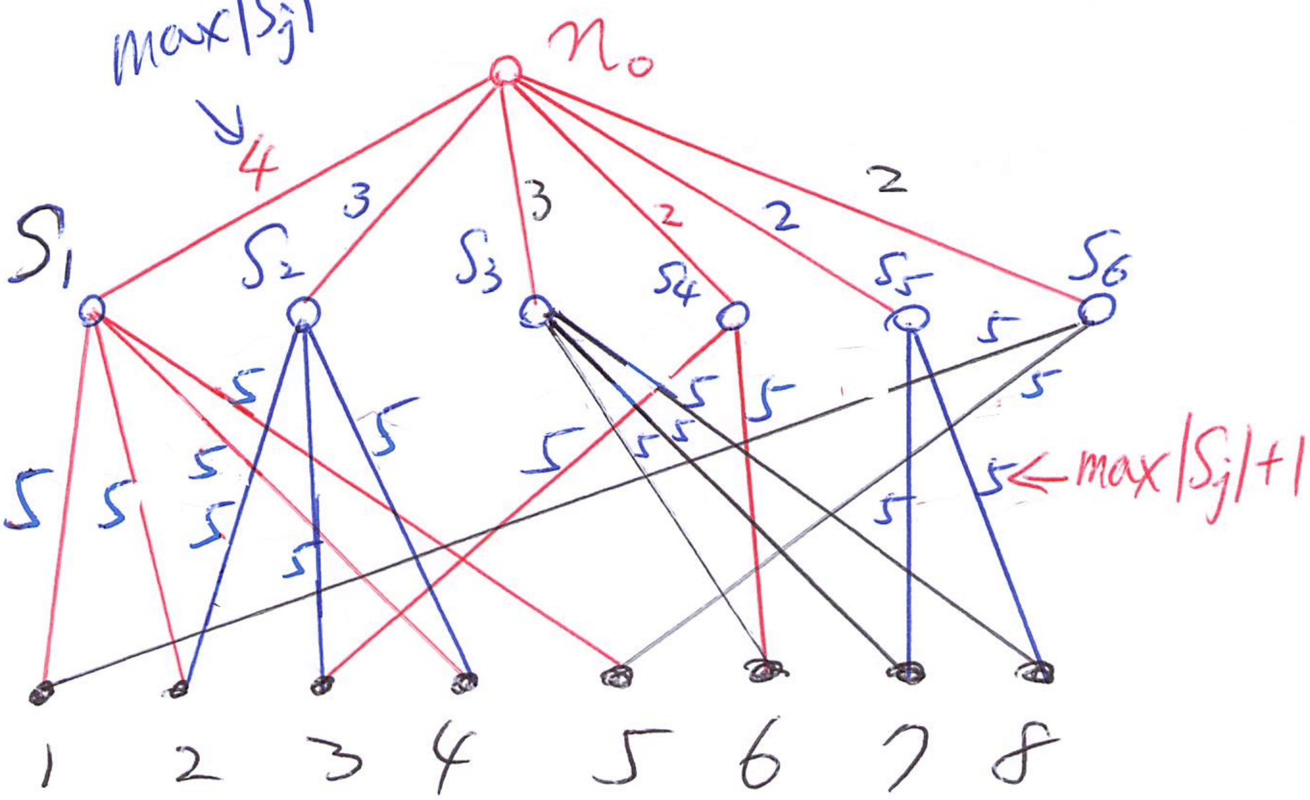


There is an exact cover $\{T_h\} \subseteq \{S_j\}$ if and only if G has an SMT of weight $|\{u_i\}| \times (\max|S_j| + 2)$.

Kun-Mao Chao
@2019

$$\{u_i\} = \{1, 2, 3, 4, 5, 6, 7, 8\}$$

$$\{S_j\} = \{\overset{S_1}{\{1, 2, 4, 5\}}, \overset{S_2}{\{2, 3, 4\}}, \overset{S_3}{\{6, 7, 8\}}, \overset{S_4}{\{3, 6\}}, \overset{S_5}{\{7, 8\}}, \overset{S_6}{\{1, 5\}}\}$$



Exact Cover: $\{S_1, S_4, S_5\}$ or $\{S_2, S_3, S_6\}$.

SMT: weight $= (4+2+2) \times (4+2) = (3+3+2) \times (4+2) = 48.$

Algorithm: MST-STEINER
Input: A graph $G = (V, E, w)$ and a terminal set $L \subset V$.
Output: A Steiner tree T.

1: Construct the metric closure G_L on the terminal set L.
2: Find an MST T_L on G_L.
3: $T \leftarrow \emptyset$.
4: **for** each edge $e = (u, v) \in E(T_L)$ in a depth-first-search order of T_L **do**
4.1: Find a shortest path P from u to v on G.
4.2: **if** P contains less than two vertices in T **then**
Add P to T;
else
Let p_i and p_j be the first and the last vertices already in T;
Add subpaths from u to p_i and from p_j to v to T.
5: Output T.

Basically we replace each edge in T_L with the corresponding shortest path at Step 4. But if there are two vertices already in the tree, adding the path will result in cycles. In this case we only insert the subpaths from the terminals to the vertices already in the tree. It avoids any cycle and ensures that the terminals are included. As a result, we can see that the algorithm returns a Steiner tree.

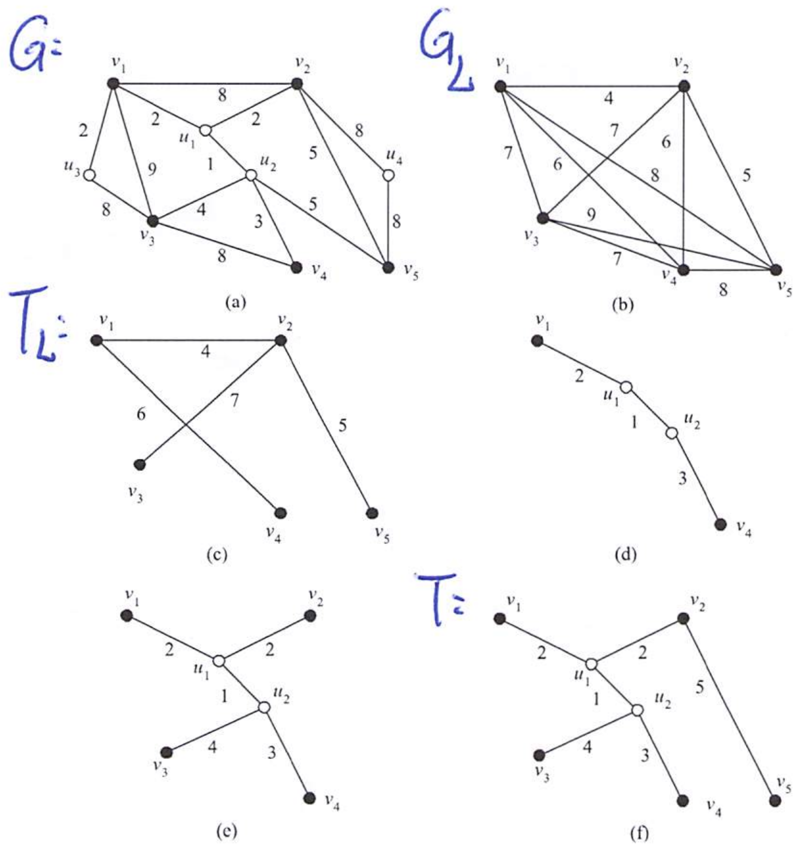


Figure 2: A sample execution of Algorithm MST-STEINER.

7

Kun-Mao Chao@2019

G_L : the metric closure on the terminal set L.

$smt(G, L)$: the Steiner minimal tree.

T_L : an MST on G_L

T : output of MST-Steiner.

* $w(T) \le \bar{w}(T_L)$ ← At most all the shortest paths are inserted.

X : an Eulerian tour on $smt(G, L)$

* $w(X) = 2w(smt(G, L))$

visit all terminals

* $\bar{w}(T_L) \le \bar{w}(tsp(G_L)) \le w(X)$

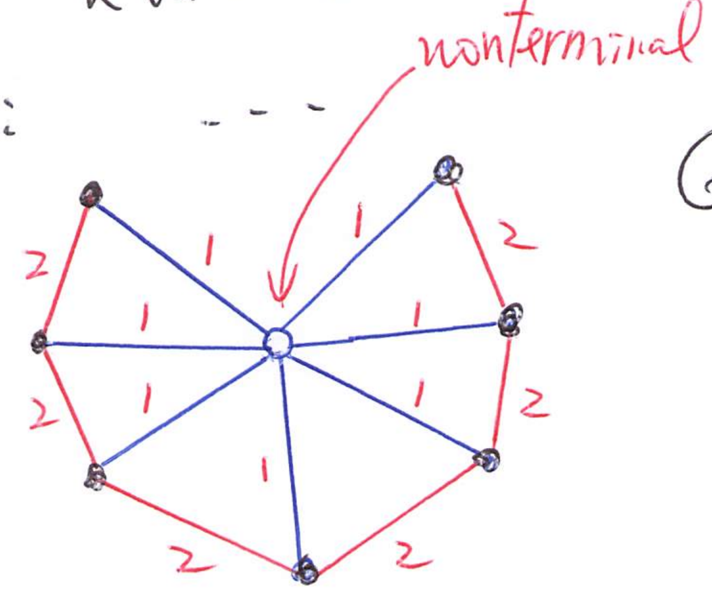
Traveling Salesperson Problem (Removing one edge forms a spanning tree.)

$\Rightarrow w(T) \le 2w(smt(G, L))$

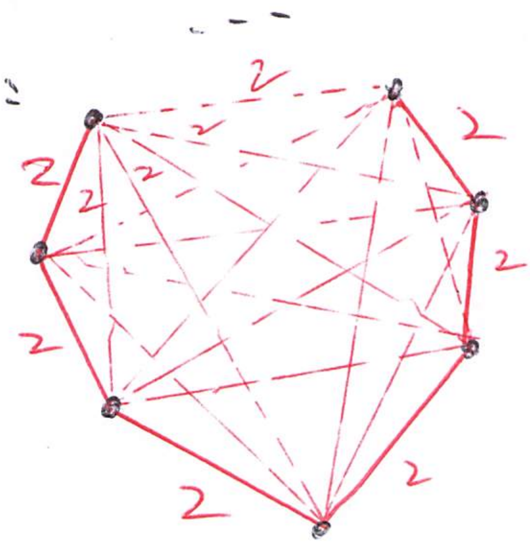
k terminals

Kun-Mao Chao @2019

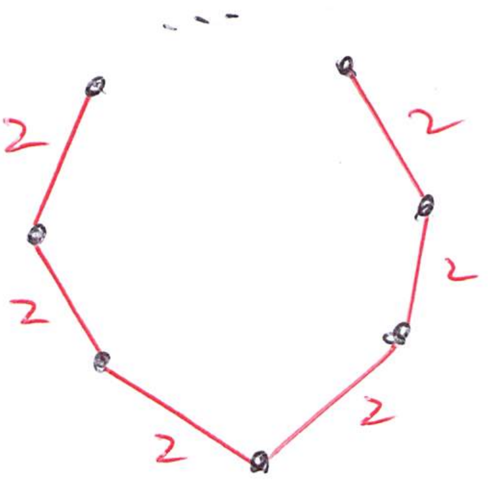
G:



G_L:



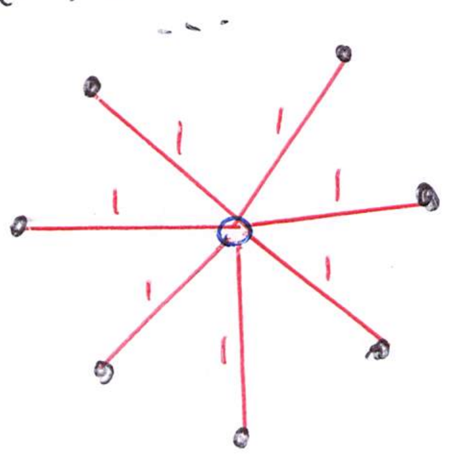
MST on G_L (T_L)



$\leftarrow T$

$w(T) = 2(k-1)$

smt(G, L)



$w(\text{smt}(G,L)) = k$

$$\frac{w(T)}{w(\text{smt}(G,L))} = \frac{2(k-1)}{k} = 2 - \frac{2}{k}$$