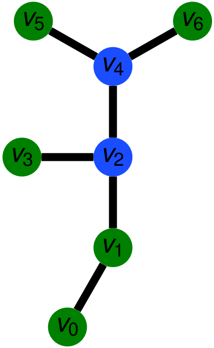
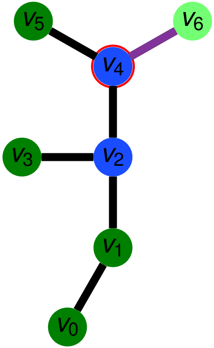
Faster exact algorithm for Steiner trees in weighted graphs

DFS order



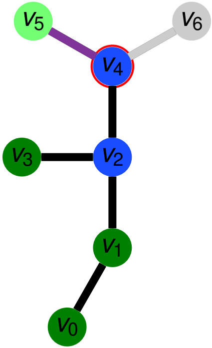
Faster exact algorithm for Steiner trees in weighted graphs

$$\sigma(\{v_0, v_1, v_3, v_5, v_6\}, \emptyset, 1) = c(\{v_6, v_4\}) + \sigma(\{v_0, v_1, v_3, v_5\}, \{v_4\}, 1)$$



Recursion formula: $\sigma(X, B, i) = \min\{c(e) + \sigma(X', B', i') : \ldots\}$.

Faster exact algorithm for Steiner trees in weighted graphs

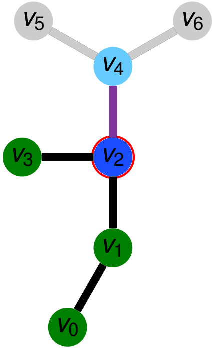


$\sigma(\{v_0, v_1, v_3, v_5, v_6\}, \emptyset, 1) = c(\{v_6, v_4\}) +$
$\sigma(\{v_0, v_1, v_3, v_5\}, \{v_4\}, 1)$

$\sigma(\{v_0, v_1, v_3, v_5\}, \{v_4\}, 1) = c(\{v_5, v_4\}) +$
$\sigma(\{v_0, v_1, v_3\}, \{v_4\}, 2)$

Recursion formula: $\sigma(X, B, i) = \min\{c(e) + \sigma(X', B', i') : \ldots\}$.

Faster exact algorithm for Steiner trees in weighted graphs



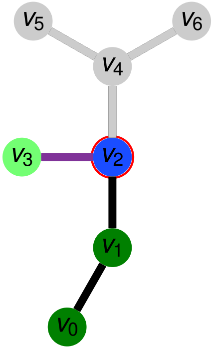
$$\sigma(\{v_0, v_1, v_3, v_5, v_6\}, \emptyset, 1) = c(\{v_6, v_4\}) + \sigma(\{v_0, v_1, v_3, v_5\}, \{v_4\}, 1)$$

$$\sigma(\{v_0, v_1, v_3, v_5\}, \{v_4\}, 1) = c(\{v_5, v_4\}) + \sigma(\{v_0, v_1, v_3\}, \{v_4\}, 2)$$

$$\sigma(\{v_0, v_1, v_3\}, \{v_4\}, 2) = c(\{v_4, v_2\}) + \sigma(\{v_0, v_1, v_3\}, \{v_2\}, 1)$$

Recursion formula: $\sigma(X, B, i) = \min\{c(e) + \sigma(X', B', i') : \ldots\}$.

Faster exact algorithm for Steiner trees in weighted graphs



$$\sigma(\{v_0, v_1, v_3, v_5, v_6\}, \emptyset, 1) = c(\{v_6, v_4\}) + \sigma(\{v_0, v_1, v_3, v_5\}, \{v_4\}, 1)$$

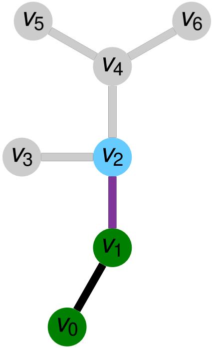
$$\sigma(\{v_0, v_1, v_3, v_5\}, \{v_4\}, 1) = c(\{v_5, v_4\}) + \sigma(\{v_0, v_1, v_3\}, \{v_4\}, 2)$$

$$\sigma(\{v_0, v_1, v_3\}, \{v_4\}, 2) = c(\{v_4, v_2\}) + \sigma(\{v_0, v_1, v_3\}, \{v_2\}, 1)$$

$$\sigma(\{v_0, v_1, v_3\}, \{v_2\}, 1) = c(\{v_3, v_2\}) + \sigma(\{v_0, v_1\}, \{v_2\}, 2)$$

Recursion formula: $\sigma(X, B, i) = \min\{c(e) + \sigma(X', B', i') : \ldots\}$.

Faster exact algorithm for Steiner trees in weighted graphs



$$\sigma(\{v_0, v_1, v_3, v_5, v_6\}, \emptyset, 1) = c(\{v_6, v_4\}) + \sigma(\{v_0, v_1, v_3, v_5\}, \{v_4\}, 1)$$

$$\sigma(\{v_0, v_1, v_3, v_5\}, \{v_4\}, 1) = c(\{v_5, v_4\}) + \sigma(\{v_0, v_1, v_3\}, \{v_4\}, 2)$$

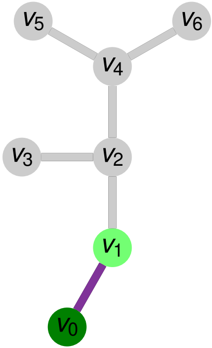
$$\sigma(\{v_0, v_1, v_3\}, \{v_4\}, 2) = c(\{v_4, v_2\}) + \sigma(\{v_0, v_1, v_3\}, \{v_2\}, 1)$$

$$\sigma(\{v_0, v_1, v_3\}, \{v_2\}, 1) = c(\{v_3, v_2\}) + \sigma(\{v_0, v_1\}, \{v_2\}, 2)$$

$$\sigma(\{v_0, v_1\}, \{v_2\}, 2) = c(\{v_2, v_1\}) + \sigma(\{v_0, v_1\}, \emptyset, 1)$$

Recursion formula: $\sigma(X, B, i) = \min\{c(e) + \sigma(X', B', i') : \ldots\}$.

Faster exact algorithm for Steiner trees in weighted graphs



$$\sigma(\{v_0, v_1, v_3, v_5, v_6\}, \emptyset, 1) = c(\{v_6, v_4\}) + \sigma(\{v_0, v_1, v_3, v_5\}, \{v_4\}, 1)$$

$$\sigma(\{v_0, v_1, v_3, v_5\}, \{v_4\}, 1) = c(\{v_5, v_4\}) + \sigma(\{v_0, v_1, v_3\}, \{v_4\}, 2)$$

$$\sigma(\{v_0, v_1, v_3\}, \{v_4\}, 2) = c(\{v_4, v_2\}) + \sigma(\{v_0, v_1, v_3\}, \{v_2\}, 1)$$

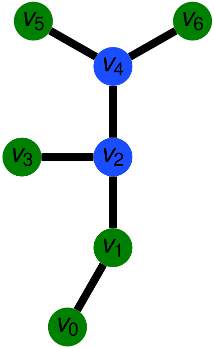
$$\sigma(\{v_0, v_1, v_3\}, \{v_2\}, 1) = c(\{v_3, v_2\}) + \sigma(\{v_0, v_1\}, \{v_2\}, 2)$$

$$\sigma(\{v_0, v_1\}, \{v_2\}, 2) = c(\{v_2, v_1\}) + \sigma(\{v_0, v_1\}, \emptyset, 1)$$

$$\sigma(\{v_0, v_1\}, \emptyset, 1) = c(\{v_1, v_0\}) + \sigma(\{v_0\}, \emptyset, 1)$$

Recursion formula: $\sigma(X, B, i) = \min\{c(e) + \sigma(X', B', i') : \ldots\}$.

Faster exact algorithm for Steiner trees in weighted graphs



$$\sigma(\{v_0, v_1, v_3, v_5, v_6\}, \emptyset, 1) = c(\{v_6, v_4\}) + \sigma(\{v_0, v_1, v_3, v_5\}, \{v_4\}, 1)$$

$$\sigma(\{v_0, v_1, v_3, v_5\}, \{v_4\}, 1) = c(\{v_5, v_4\}) + \sigma(\{v_0, v_1, v_3\}, \{v_4\}, 2)$$

$$\sigma(\{v_0, v_1, v_3\}, \{v_4\}, 2) = c(\{v_4, v_2\}) + \sigma(\{v_0, v_1, v_3\}, \{v_2\}, 1)$$

$$\sigma(\{v_0, v_1, v_3\}, \{v_2\}, 1) = c(\{v_3, v_2\}) + \sigma(\{v_0, v_1\}, \{v_2\}, 2)$$

$$\sigma(\{v_0, v_1\}, \{v_2\}, 2) = c(\{v_2, v_1\}) + \sigma(\{v_0, v_1\}, \emptyset, 1)$$

$$\sigma(\{v_0, v_1\}, \emptyset, 1) = c(\{v_1, v_0\}) + \sigma(\{v_0\}, \emptyset, 1)$$

Recursion formula: $\sigma(X, B, i) = \min\{c(e) + \sigma(X', B', i') : \ldots\}$.
For every metric instance there is an ordering of the vertices such that always $|B| \leq 1 + \lfloor \log_2(\#\text{terminals}/3) \rfloor$

State of the art

- Erickson, Monma, Veinott [1987]: $O\left(3^k n + 2^k(m + n \log n)\right)$
  fastest if $k < 4 \log n$
- Fuchs, Kern, Mölle, Richter, Rossmanith, Wang [2007]: $O\left(2^{k+(k/2)^{1/3}(\ln n)^{2/3}}\right)$
  fastest if
  $4 \log n < k < 2 \log n(\log \log n)^3$
- new: $O\left(nk2^{k+(\log_2 k)(\log_2 n)}\right)$
  fastest if
  $2 \log n(\log \log n)^3 < k < (n - \log^2 n)/2$
- enumeration: $O\left(m\alpha(m, n)2^{n-k}\right)$
  fastest if $k > (n - \log^2 n)/2$