Steiner Minimal Trees*

Bang Ye Wu Kun-Mao Chao

1 Steiner Minimal Trees

While a spanning tree spans all vertices of a given graph, a *Steiner tree* spans a given subset of vertices. In the Steiner minimal tree problem, the vertices are divided into two parts: *terminals* and *nonterminal vertices*. The terminals are the given vertices which must be included in the solution. The cost of a Steiner tree is defined as the total edge weight. A Steiner tree may contain some nonterminal vertices to reduce the cost. Let V be a set of vertices. In general, we are given a set $L \subset V$ of terminals and a metric defining the distance between any two vertices in V. The objective is to find a connected subgraph spanning all the terminals of minimal total cost. Since the distances are all nonnegative in a metric, the solution is a tree structure. Depending on the given metric, two versions of the Steiner tree problem have been studied.

- **(Graph) Steiner minimal trees (SMT)**: In this version, the vertex set and metric is given by a finite graph.
- **Euclidean Steiner minimal trees (Euclidean SMT)**: In this version, V is the entire Euclidean space and thus infinite. Usually the metric is given by the Euclidean distance (L^2-norm). That is, for two points with coordinates (x_1, y_2) and (x_2, y_2), the distance is

  $$\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}.$$

  In some applications such as VLSI routing, L^1-norm, also known as *rectilinear distance*, is used, in which the distance is defined as

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

Figure 1 illustrates a Euclidean Steiner minimal tree and a graph Steiner minimal tree.

1.1 Approximation by MST

Let $G = (V, E, w)$ be an undirected graph with nonnegative edge weights. Given a set $L \subset V$ of terminals, a Steiner minimal tree is the tree $T \subset G$ of minimum total edge weight such that T includes all vertices in L.

Problem: Graph Steiner Minimal Tree
Instance: A graph $G = (V, E, w)$ and a set $L \subset V$ of terminals
Goal: Find a tree T with $L \subset V(T)$ so as to minimize $w(T)$.

[*] An excerpt from the book "Spanning Trees and Optimization Problems," by Bang Ye Wu and Kun-Mao Chao (2004), Chapman & Hall/CRC Press, USA.

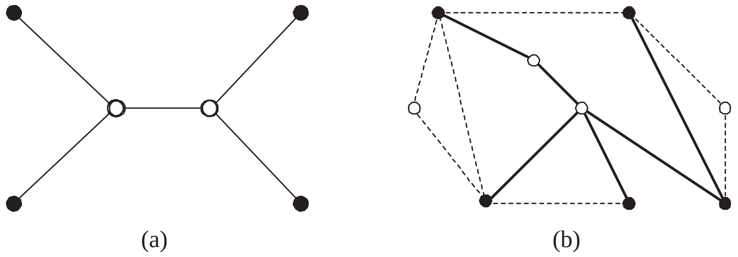

Figure 1: Examples of Steiner minimal trees. (a) A Euclidean Steiner minimal tree; and (b) a graph Steiner minimal tree. The black points are terminals and the white points are nonterminals.

The decision version of the SMT problem was shown to be NP-complete by a transformation from the Exact Cover by 3-Sets problem [8]. We shall focus on some approximation results.

A well-known method to approximate an SMT is to use a minimal spanning tree (MST). First we construct the metric closure on L, i.e., a complete graph with vertices L and edge weights equal to the shortest path lengths. Then we find an MST on the closure, in which each edge corresponds to one shortest path on the original graph. Finally the MST is transformed back to a Steiner tree by replacing each edge with the shortest path and some straightforward postprocessing to remove any possible cycle.

```
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 $e = (u, v) \in E(T_L)$ 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.

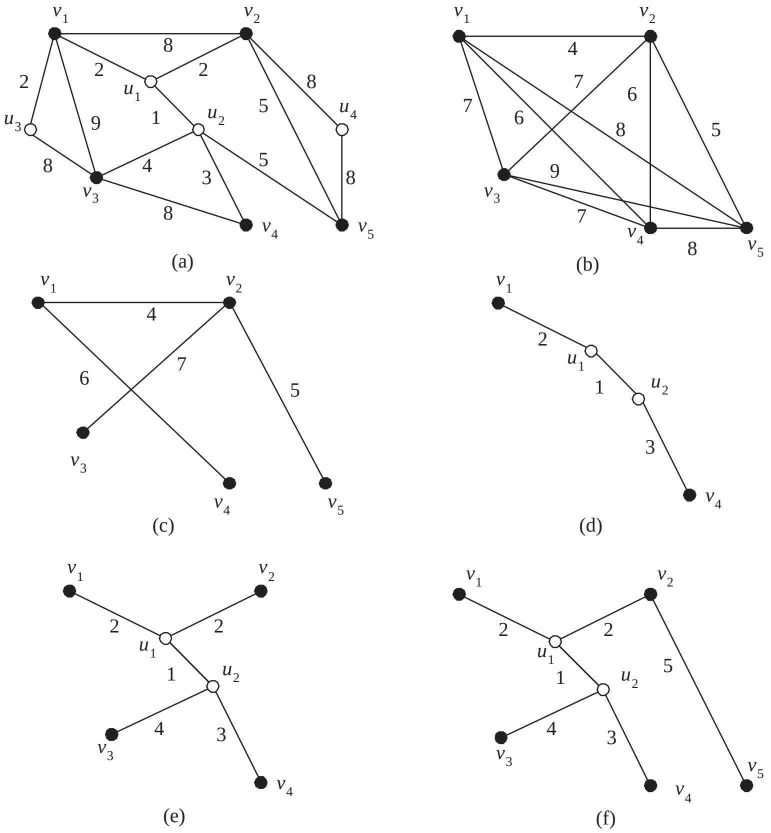

Figure 2: A sample execution of Algorithm MST-Steiner.

Example 1: Figure 2 shows a sample execution of the algorithm. (a) is the given graph G, in which $L = \{v_i | 1 \leq i \leq 5\}$ is the terminal set and u_i, $1 \leq i \leq 4$ are nonterminal vertices. (b) and (c) are the metric closure G_L and a minimum spanning tree T_L on G_L respectively. Initially T is empty. Suppose that edge $(v_1, v_4) \in E(T_L)$ is first chosen edge. The corresponding shortest path in G is (v_1, u_1, u_2, v_4). Since all vertices on the path are not in T, the whole path is inserted into T (Frame(d)). At the second iteration, edge $(v_2, v_3) \in E(T_L)$ is chosen. The corresponding shortest path is (v_2, u_1, u_2, v_3). However, u_1 and u_2 are already in T. Therefore, only (v_2, u_1) and (u_2, v_3) are inserted (Frame (e)). At the third iteration, edge (v_2, v_5) is chosen, and edge (v_2, v_5) is inserted (Frame (f)). Finally, the last edge in T_L is (v_1, v_2). Since both v_1 and v_2 are already in T, no edge is inserted, and the tree in Frame (f) is the output tree.

We are interested in how good the output tree is. Let $G_L = (L, E(G_L), \bar{w})$. First we observe that $w(T) \leq \bar{w}(T_L)$ since at most all the shortest paths are inserted. We want to compare T_L with a Steiner minimal tree. Let smt(G, L) be the Steiner minimal tree. Consider a Eulerian tour X on the doubling tree (see Figure **??** for the definition of a Eulerian tour on a doubling tree). Since the tour traverses each edge exactly twice, $w(X) = 2w(\text{smt}(G, L))$. On the other hand, since the tour visits all the terminals, we have

$$w(X) \geq w(\text{tsp}(G_L)) \geq w(T_L),$$

in which tsp(G_L) is a minimal Hamiltonian cycle on G_L— an optimal solution of the *Traveling Salesperson problem*. Consequently we have

$$w(T) \leq w(T_L) \leq 2w(\text{smt}(G, L)).$$

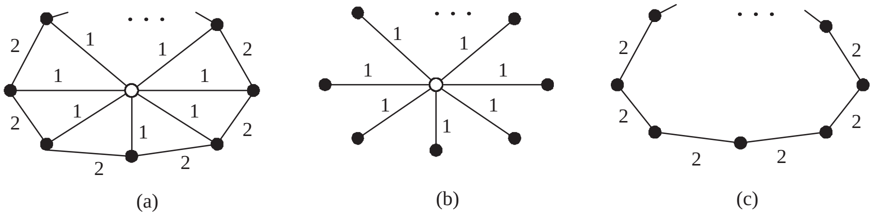

Figure 3: (a) A graph in which the white vertex is a nonterminal vertex and others are terminals. (b) A Steiner minimal tree. (c) A minimum spanning tree of the subgraph induced on the terminals.

To see the bound is asymptotically tight, we show an extreme example in Figure 3. Suppose that there are k terminals and one nonterminal vertex. The terminals form a cycle with $w(e) = 2$ for each edge e in the cycle. The nonterminal vertex is connected to each terminal with an edge of unit length. The Steiner minimal tree is the star centered at the nonterminal vertex and has weight k. Meanwhile an MST is the path consisting of the terminals and has weight $2(k-1)$. Therefore the ratio is $(2 - 2/k)$.

The time complexity of the algorithm is $O(|V||L|^2)$, dominated by the construction of the metric closure. In summary we have the following theorem:

Theorem 1: The MST-Steiner algorithm finds a 2-approximation of an SMT of a general graph in $O(|V||L|^2)$ time, where V is the vertex set and L is the terminal set. Furthermore, the ratio is asymptotically tight.

Define the *Steiner ratio* as

$$\rho = \frac{w(\mathrm{mst}(G_L))}{w(\mathrm{smt}(G, L))}.$$

The above theorem shows that the Steiner ratio in general metric is 2.

Bibliographic Notes and Further Reading

The study of the Euclidean Steiner minimal tree problem has a long history. Frank Kwang-Ming Hwang and D.S. Richards gave a nice survey up to 1989 [7]. Here we only mention some of the results. The readers can find more details in their survey paper. The MST-based approximation algorithm was proposed by several authors independently, such as [10, 9]. The algorithm we introduced in this chapter is a straightforward implementation. There are more efficient algorithms with the same idea. Basically these algorithms are based on Prim's or Kruskal's MST algorithms without explicitly constructing the metric closure. For example, Y.F. Wu, Peter Widmayer, and C. K. Wong [12] gave a Kruskal-based approach, and H. Takahashi and A. Mastsuyama [11] gave a Prim-based algorithm.

The 11/6-approximation algorithm is due to Alexander Zelikovsky [13], who made the breakthrough of the simple 2-approximation. The result was extended by Piotr Berman and Viswanathan Ramaiyer [1], in which they used 4-tuples to derive a 16/9-approximation algorithm.

The Euclidean Steiner ratio is another challenge attracting lots of researchers. Since the Euclidean space is a restricted version of the general metric, it is not surprising that the Euclidean Steiner ratio is smaller than that in general metric. In fact, it was shown that the ratio is $2/\sqrt{3}$. But the proof is beyond the scope of this note (please refer to a presented paper in the same course in Spring 2004.)! It was once a famous open problem. E.N. Gilbert and H.O. Pollak [5] conjectured that the ratio is $2/\sqrt{3}$ in the Euclidean plane. Lots of works had been devoted to the improvement of the ratio, and the conjecture was finally proven by Ding-Zhu Du and Frank Kwang-Ming Hwang [3]. For rectilinear distances, Hwang showed that 3/2 is an upper bound of the Steiner ratio [6]. By Zelikovsky's algorithm, the approximation ratio was improved to 11/8, and further to 97/72 by using 4-tuples [1].

The NP-completeness of the decision version of the graph Steiner minimal tree problem was shown by Richard M. Karp [8]. Some restricted versions are also shown to be NP-complete, such as unweighted graphs and bipartite graphs [4]. It was shown [2] that the problem is MAX SNP-hard in the metric that each edge has length either 1 or 2. As a result, it is impossible to derive a polynomial time approximation scheme unless NP=P.

References

[1] P. Berman and V. Ramaiyer. Improved approximations for the Steiner tree problem. *J. Algorithms*, 17(3):381–408, 1994.

[2] M. Bern and P. Plassmann. The Steiner problem with edge lengths 1 and 2. *Inf. Process. Lett.*, 32(4):171–176, 1989.

[3] D-.Z Du and F.K. Hwang. A proof of the Gilbert-Pollak conjecture on the Steiner ratio. *Algorithmica*, 7:121–135, 1992.

[4] M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W.H. Freeman and Company, San Francisco, 1979.

[5] E.N. Gilbert and H.O. Pollak. Steiner minimal trees. *SIAM J. Appl. Math.*, 16(1):1–29, 1968.

[6] F.K. Hwang. On Steiner minimal trees with rectilinear distance. *SIAM J. Appl. Math.*, 30:104–114, 1976.

[7] F.K. Hwang and D.S. Richards. Steiner tree problems. *Networks*, 22:55–89, 1992.

[8] R.M. Karp. Reducibility among combinatorial problems. In R.E. Miller and J.W. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press, New York, 1975.

[9] L. Kou, G. Markowsky, and L. Berman. A fast algorithm for Steiner trees. *Acta Inform.*, 15(2):141–145, 1981.

[10] J. Plesnik. The complexity of designing a network with minimum diameter. *Networks*, 11:77–85, 1981.

[11] H. Takahashi and A. Mastsuyama. An approximate solution for the Steiner problem in graphs. *Math. Jap.*, 24:573–577, 1980.

[12] Y.F. Wu, P. Widmayer, and C.K. Wong. A faster approximation algorithm for the Steiner problem in graphs. *Acta Inform.*, 23(2):223–229, 1986.

[13] A. Zelikovsky. An 11/6-approximation algorithm for the network Steiner problem. *Algorithmica*, 9:463–470, 1993.