Practitioner's Commentary: The Outstanding Steiner Tree Papers

Marshall Bern
Xerox Palo Alto Research Center
3333 Coyote Hill Road
Palo Alto, CA 94304

Introduction

I'm more of a theoretician than a practitioner, but I have worked on Steiner tree algorithms and talked to a number of practitioners designing communications networks and computer chips. The rectilinear Steiner tree problem arises in VLSI design. An electrical network connecting a set of points on a chip should be as short as possible (more precisely, should have minimum capacitance) in order to reduce its charging and discharging time and increase its speed of operation.

A closely related version of the Steiner tree problem requires a network connecting a given subset of the stations within a much larger communications network. This version of the problem arises in pricing hookups for telephone customers with offices in a number of different cities. Thus the MCM Steiner tree problem posed a fairly realistic challenge.

Exact Algorithms

We might first attempt to solve the problem by brute force. We initially assume that all phantom stations, or *Steiner points*, lie on integer lattice points in the rectangle $[0, 35] \times [0, 24]$. There are 891 (that is, $36 \times 25 - 9$) possible Steiner-point locations in the rectangle. We cannot, however, test locations one at a time, because the best tree may not include the single best Steiner point. To be sure to include an optimal solution among our possibilities, we must test each subset of locations by computing the minimum spanning tree for that subset along with all the given stations. There are over 2×10^{20} subsets of locations with at most seven points. This is an appallingly large number, far too big to search.

A theorem of Hanan [1966], cited by all three Outstanding teams, states that we need only consider Steiner points that share their x-coordinate with one given station and their y-coordinate with another given station. In other words, we need only consider a grid of points formed by passing vertical and horizontal lines through the given stations. This reduces the number of possible locations to 63, and the number of subsets to $\sum_{k=1}^{7} \binom{63}{k}$, which is about 630,000,000.

A further reduction results from removing points near the corners of the grid. We can safely remove a corner point (that is, one at which the outer angle measures $270°$) if it is not a given station; this rule may be repeated to produce a reduced grid with 31 possible locations and about 3,600,000 subsets. This reduction was used first by Yang and Wing [1972] and has been rediscovered several times, most recently by the team from the University of Western Ontario. We are now down to a manageable search, several hours on a fast workstation. The Western Ontario team used this approach (as well as a heuristic approach), thereby proving that the best solution to Part 1 of the problem has cost 94.

The brute-force method is not the only exact algorithm for the Steiner tree problem; there is a somewhat faster algorithm that uses dynamic programming [Dreyfus and Wagner 1972; Levin 1971]. This algorithm avoids testing all possible subsets of Steiner points, although it does have to compute optimal Steiner trees for all subsets of given stations.

Using some variation of this algorithm, it should be possible to find the exact solution to rectilinear Steiner tree problems with about 50 points in a few hours; but I do not know of anyone actually doing this. Computer-chip designers usually are content with approximate solutions, because the total length of a network is not always the limiting factor in the speed of operation. Furthermore, the Steiner tree problems arising in chip design are not very pure—other considerations, such as wiring around obstacles or leaving room for later wirings, may be important.

Approximate Algorithms

Because the computer time to solve the rectilinear Steiner tree problem exactly grows explosively, practitioners really use algorithms that find approximate solutions. One approximate algorithm is to use the minimum spanning tree, that is, to include no phantom stations at all. Hwang [1976] proved that this tree is never more than 50% longer than the optimal Steiner tree. This might be good enough in some practical situations, but it would not suffice as a contest solution.

All three teams programmed approximate algorithms. The team from Beloit College looked for the most advantageous single Steiner point and then added that point to the set of given stations. They repeated this "greedy" process until either the number of added stations reached seven or no further improvement was possible. This method found an optimal solution, though it will not always do so. (In their text, this team implies incorrectly that their algorithm is an exact algorithm.)

The team from Mount St. Mary's College programmed a number of different heuristics and admirably tested their methods on four problem instances besides the given one. They made another good decision in evaluating their algorithms; they report the relative improvement over the minimum spanning tree, as they cannot in general compute the exact Steiner tree. All of their heuristics made greedy decisions, that is, at each step they chose the cheapest alternative. They devised a "modified Kruskal" method that imitates a well-known algorithm for the minimum spanning tree problem but connects subtrees using Steiner points when advantageous. This heuristic found an optimal solution.

Though the Mount St. Mary team's heuristics have a speed advantage over the heuristic of the Beloit team, they probably do not produce solutions quite as good for larger problems. I am impressed that both of these teams invented quite reasonable approximate algorithms, very similar to those actually used in practice.

The Western Ontario team programmed a very different heuristic called "simulated annealing." Simulated annealing is a general scheme that randomly moves from solution to solution according to local rearrangement rules. The chance of moving to another solution depends on the costs of the two solutions and on a control variable called "temperature." With an appropriate "cooling schedule," the heuristic ultimately sticks at a nearly optimal solution. The team's simulated annealing program found an optimal solution 100 times out of 100 trials. I find this result surprising, and I wonder how well simulated annealing would do on larger problems.

Generalizations

Part 2 of the problem adds a new twist: each station has a cost that depends on its *degree*, that is, the number of lines meeting at the station. This assumption models situations such as telephone networks, in which stations as well as lines are expensive.

All three teams used heuristic algorithms for this problem, finding solutions of cost about 134.24 (Beloit), 134.85 (Mount St. Mary's), and 135.89 (Western Ontario). The Western Ontario team also performed an exhaustive search over all solutions with at most four Steiner points. There must have been a bug in their program, however, as the other teams beat them, and careful inspection reveals that their first solution (their **Figure 3**, p. 144) can be improved to 134.89 with a simple rearrangement. I do not know the optimal cost for Part 2, but I am sure all three teams came quite close.

Part 3 asked for further generalizations. Altogether, the Outstanding teams mentioned obstacles, alternative distance metrics, higher dimensions, and additional cost criteria. All of these are natural and useful generalizations [Hwang et al. 1992].

I shall elaborate a little on two of the more famous variants of the Steiner tree problem. The *Euclidean Steiner tree problem* uses the ordinary Euclidean metric. It is not hard to prove that, in an optimal solution, all Steiner points are incident to exactly three lines, and that these lines subtend angles of $120°$. Even armed with this theorem, however, it is a hard problem to find a finite set of points which includes all possible locations of Steiner points. Z.A. Melzak [1961] gave a geometric construction that accomplishes this task; his paper really initiated modern research on the Steiner tree problem.

Another variant of the Steiner tree problem is the *phylogenetic Steiner problem*, proposed by Cavalli-Sforza and Edwards [1967]. In this variant, the given points represent organisms, and the optimal solution represents the most likely evolutionary tree relating these organisms. The minimum-cost criterion corresponds to a "parsimony principle": the most likely tree is the one requiring the fewest mutations.

Recent Research on Steiner Trees

Ronald Graham and I wrote an article for *Scientific American* on the Steiner tree problem less than three years ago [Bern and Graham 1989]. Already that article could stand an update.

The article's teaser asks, "What is the shortest network of line segments interconnecting an arbitrary set of, say, 100 points? The solution to this problem has eluded the fastest computers and sharpest mathematical minds." Later in 1989, Ernie Cockayne and Denton Hewgill [1992] of the University of Victoria sent us a picture of an optimal solution to a Euclidean Steiner tree problem on 100 random points. More-regular arrangements of points, such as points in a grid, tend to be harder, and a 100-station grid is probably (careful here!) still beyond the state of the art.

Later in the article, we mention the long-standing open problem of proving that for any set of points, a minimum spanning tree is never longer than $2/\sqrt{3}$ times the length of an optimal Euclidean Steiner tree. This conjecture was finally proved (very elegantly!) in 1990 by D.Z. Du and F.K. Hwang [1992].

The most recent big theoretical results on Steiner trees are contained in a sequence of papers starting with Zelikovsky [1993]. These papers give fast approximate algorithms with performance guarantees better than the guarantees offered by the minimum spanning tree. For example, P. Berman and V. Ramaiye [1994] have shown that although a minimum spanning tree may be $3/2$ times as long as a optimal rectilinear Steiner tree, a certain greedy solution will never be more than $97/72$ times as long.

For a survey of the latest results on the Steiner tree problem and many other hard geometric problems, see Bern and Eppstein [1995].

References

Berman, P., and V. Ramaiyer. 1994. Improved approximation algorithms for the Steiner tree problem. *Journal of Algorithms* 17: 381–408.

Bern, M.W., and R.L. Graham. 1989. The shortest-network problem. *Scientific American* 260(1) (January 1989): 84–89.

Bern, M.W., and D. Eppstein. 1995. Approximation algorithms for geometric problems. To appear in *Approximation Algorithms for NP-Complete Problems*, edited by D. Hochbaum. PWS Publications.

Cavalli-Sforza, L.L., and A.W.F. Edwards. 1967. Phylogenetic analysis: models and estimation procedures. *American Journal of Human Genetics* 19: 233–257.

Cockayne, E.J., and D.E. Hewgill. 1992. Improved computation of plane Steiner minimal trees. *Algorithmica* 7: 219–229.

Dreyfus, S.E., and R.A. Wagner. 1972. The Steiner problem in graphs. *Networks* 1: 195–208.

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

Hanan, M. 1966. On Steiner's problem with rectilinear distance. *SIAM Journal of Applied Mathematics* 14: 255–265.

Hwang, F.K. 1976. On Steiner minimal trees with rectilinear distance. *SIAM Journal of Applied Mathematics* 30: 104–114.

Hwang, F.K., D.S. Richards, and P. Winter. 1992. *The Steiner Tree Problem.* New York: North-Holland.

Levin, A.J. 1971. Algorithm for the shortest connection of a group of graph vertices. *Soviet Mathematics Doklady* 12: 1477–1481.

Melzak, Z.A. 1961. On the problem of Steiner. *Canadian Mathematical Bulletin* 4: 143–148.

Yang, Y.Y., and O. Wing. 1972. Optimal and suboptimal solution algorithms for the wiring problem. *Proceedings of the IEEE Symposium on Integrated Circuit Theory* (1972): 154–158.

Zelikovsky, A.Z. 1993. An 11/6-approximation algorithm for the network Steiner problem. *Algorithmica* 9 (5): 463–470.

About the Author

Marshall Bern is a member of the research staff at Xerox Palo Alto Research Center. He received a master's degree in mathematics from the University of Texas at Austin in 1980 and a Ph.D. in computer science from the

University of California at Berkeley in 1987. In between graduate schools, he worked in the area of signal processing. His current research is on algorithms for computer graphics and finite-element mesh generation—in fact, on any problems for which he can scribble pictures.