Steiner Tree Problems

F. K. Hwang
Bell Laboratories, AT&T
Murray Hill, New Jersey 07974-2070

Dana S. Richards
University of Virginia

We give a survey up to 1989 on the Steiner tree problems which include the four important cases of euclidean, rectilinear, graphic, phylogenetic and some of their generalizations. We also provide a rather comprehensive and up-to-date bibliography which covers more than three hundred items.

1. INTRODUCTION

The Steiner minimal tree problem was first studied for the Euclidean distance metric. Given a set N of n points in the Euclidean plane, the shortest network interconnecting N is called a *Steiner minimal tree* (denoted ESMT or simply SMT) for N. Junctions of edges in the network that are not points in N are called *Steiner points* or *S-points;* the original points are *N-points*. The study of the Euclidean SMT problem for the special case $n = 3$ has a long history dating back to the 17th century with names like Cavalieri, Fermat, Simpson, Torricelli, Heinen, Viviani, and Steiner among others (e.g., [186, 192, 294]). However, the $n = 3$ Steiner problem is also a special case of what is called "generalized Fermat problem" by Kuhn [170], which asks for a point to minimize the sum of the distances between that point and all the points of N. Since only for $n \geq 4$ does the SMT problem assert its own character distinct from the generalized Fermat problem, we will consider the 1934 Jarnik and Kössler paper [152] as the first paper on the "genuine" SMT problem. However, the real awakening of interest in the SMT problem is due to its inclusion in the popular book *What Is Mathematics?* by Courant and Robbins [65] in 1941, who also attached the name "Steiner" to the problem.

There have been many generalizations of the SMT problem. Some have changed the metric and costs but have preserved the geometric basis. The rectilinear metric has received much attention due to its use in many applications, such as wiring in VLSI and utilities in buildings. Some generalizations

NETWORKS, Vol. 22 (1992) 55–89

CCC 0028-3045/92/01055-35$04.00

have abstracted the problem as finding the minimum connecting network for a subset of all possible points; the graphical formulation is the primary example. Many other cases are treated below.

2. THE EUCLIDEAN SMT PROBLEM

Mechanical solutions of the SMT problem by using pegs and elastic bands or soap films were nicely covered by Miehle [194]; see also Courant and Robbins [65, p. 392], Steinhaus [256, p. 119], Gallawa [110], Clark [51], Polya [207], Palermo [201], and Gilbert and Pollak [118]. Soukup [252] provided a solution using a rubber membrane. These techniques are representative of the class of "analog" techniques that use natural systems' tendencies to find minimum energy configurations to solve optimization problems. However, our survey will concentrate on mathematical analyses and solutions to the SMT problem. It should be mentioned that the paper by Gilbert and Pollak [118] gave an excellent survey for works up to 1968. A more recent introduction to the SMT problem was given by Bern and Graham [20].

An interconnecting network T is called a *Steiner tree,* ST, if it satisfies the following three conditions:

(i) T is a tree.
(ii) Any two edges of T meet at an angle of at least 120°.
(iii) An S-point cannot be of degree 1 or 2.

Clearly, (i) and (iii) are necessary for a shortest interconnecting network. Furthermore, it was shown [118] that (ii) is also necessary by an argument of minimizing the potential of forces (edges). Therefore, a shortest interconnecting network must be an ST. Note that (ii) implies that any point of T can have degree at most three. (ii) and (iii) together imply that each S-point is of degree exactly three. (i) and (iii) together imply that there are at most $n - 2$ S-points. An ST is *full* (denoted FST) if it contains $n - 2$ S-points. It is easily verified that each N-point is of degree 1 in an FST. The graph structure, P, of an ST is called a *topology*. A *full topology* (FP) is the topology for some FST. By a convexity argument (see [118] and [144]), it is easily proved that a topology P can have at most one ST; when the ST exists, it is the shortest interconnecting network for P. We can use standard minimization techniques to solve for the optimal locations of S-points. But as Miehle [194] pointed out, these equations are nonlinear so that an iteration method is indicated. Recently, Smith [249] gave such a method based on a "real RAM" model allowing infinite precision, which has a geometric convergence rate.

Call a topology *oriented* if at each point the clockwise ordering of its edges are specified. Given a full oriented topology P, Cockayne [54] defined a cyclic order of the N-points by always making a left turn, after leaving an N-point, to reach the next N-point (note that Werner [290] defined a different cyclic order by alternating left and right turns). While it is well known [118] that an SMT must lie in the convex hull of N, Cockayne extended its validity to the "Steiner hull," obtained from a convex hull by sequentially removing wide triangles.

Cockayne showed that the clockwise order of N-points on the Steiner polygon, which bounds the Steiner hull, is consistent with the cyclic order that he defined. Let $v_1, \cdots, v_n$ be the N-points ordered in this way. Booth [30] gave a simple formula for the length of the ST in terms of the edges $[v_i, v_{i+1}]$ and P. Recently, Smith [248] improved on the Steiner hull by proving that there always exists a crossing-free Eulerian circuit on the N-points such that the SMT lies on or inside it. Provan [212] has put these results into a more general setting.

One may also study the expected length $E(n)$ and the maximum length $M(n)$ of an SMT for n points distributed in a unit area A. Few [98] proved by construction that, when A is a square, then

$$(3/4)^{1/4}\sqrt{n} + O(1) \leq M(n) \leq \sqrt{n} + 7/4.$$

Chung and Graham [46] refined Few's argument to improve the upper bound to $.995\sqrt{n}$ for n sufficiently large (note that Few's bounds hold for all n). Smith [248] improved the coefficient of $\sqrt{n}$ in the lower bound to $\frac{1}{2}12^{1/4} \approx .9306$ for n large.

Megiddo [189] proved that a game between the N-points to split the cost of their SMT can have an empty "core" (a solution space), contrary to the minimal spanning tree case. Arlinghaus has given an "atlas" of Steiner trees [6].

2.1. Exact Algorithms

Melzak [191] first gave a finite algorithm for constructing SMT for general set N; also see Cockayne [53]. The general approach is to partition each topology P into full topologies $\{FP_i\}$ and apply what Melzak called a "Euclidean construction" to obtain an FST on each FP_i. This partitioning is accomplished by splitting each N-point with degree 2 or more. If each FP_i has an FST, then the union of these FSTs yields the ST for P. If any of the FP_i does not have an FST, then neither does P have an ST. Compare the lengths of the STs of all possible topologies and select a shortest one to be the SMT for N.

We give a brief description of Melzak's Euclidean construction. The construction consists of two stages: a reduction stage and a recovery stage. For a given P, two N-points are called *brothers* if they are adjacent to the same S-point. At each step of the reduction stage, we delete a pair of brothers and add a *substitution point* that forms an equilateral triangle with the two N-points. The substitution point is now considered as an N-point and the process continues until only two N-points are left, thus ending the reduction stage. The recovery stage starts by connecting these two N-points by an edge. We reverse the order of the reduction steps and bring back each pair of brothers at each recovery step. Let c be the circle circumscribing the pair of brothers A and B and their substitution point C. If arc AB of c intersects the edge of C at the point s, then s is the S-point joining A and B. We connect $[A, s]$ and $[B, s]$, discard C, and proceed to the next step. If they do not intersect, then an ST does not exist for this topology.

Melzak's algorithm takes exponential time. There are two sources of exponentiality: one minor and major. The minor one is due to the Euclidean construction that, at each step, chooses one of two possible substitution points (as there are two equilateral triangles for a given side). Since the correctness of the choice is not known until an ST is either constructed or demonstrated to be nonexistent, backtracking is necessary, which requires a worst-time complexity of $O(2^s)$, where s is the number of S-points in the given full topology. Hwang [140] recently removed this minor source by showing that when proceeding with a specified sequence of S-points, each substitution point can be chosen correctly at each step. Thus, no backtracking is necessary and the time complexity is linear.

The major source of exponentiality is due to the large number of topologies. If no N-point is allowed to have three edges (a degenerate case), the number is $2^{-n}\binom{n}{s+2}[(n-s-2)!]/s!$ for n N-points and s S-points. The number is even bigger when this restriction is dropped. For example, this number is 5625 for $n = 6$, 110,880 for $n = 7$, and 2,643,795 for $n = 8$. Many pruning techniques for quickly eliminating the FPs have been proposed (see [118] for a good collection), but they do not affect the exponentiality. The two most effective techniques are

(i) *Reduction of topologies.* By Cockayne's result, we need be concerned only with full oriented topologies whose cyclic orderings of N-points are consistent with the clockwise ordering of N-points on the Steiner polygon. Smith [248] showed that by using his crossing-free circuit idea the number of topologies needed to be checked is reduced from $O(n^n)$ to $O(c^n)$, where c is a constant.

(ii) *Restriction on S-points.* It is well known [118] that an S-point s adjacent to two points a and b must lie on the 120° arc of the circle $C(a, b)$ that circumscribes an equilateral triangle with $[a, b]$ as one side. If a (b) is itself a substitution point corresponding to an S-point confined to an arc, the arc confining s can be made smaller than 120° since it can cross neither the arc of a nor the arc of b. Winter [295] showed that for most full topologies a few rounds of nested confinement will reduce the arc of an S-point to an empty arc, thus eliminating the topology.

Winter also proposed an inverse approach (as versus Melzak's algorithm) to construct an SMT. He first constructs a list of all FSTs on subsets of N-points and then inspects the unions of these STs to see if any such union yields an ST on the set N. He argued that the list of FSTs is never too long in practice due to the powerful restriction (ii) and some optimality tests.

Recently, a new approach for constructing an SMT was proposed by Trietsch and Hwang [268] and also by Hwang and Weng [144]. For each full topology FP, let $D(FP)$ denote the set of topologies that can be obtained from FP by sequentially collapsing S-points into adjacent N-points [FP is also in $D(FP)$]. They proved that each P belongs to a $D(FP)$ for some FP and that there exists at most one ST in each $D(FP)$. Thus, we need only enumerate the

set of FPs, instead of the much larger set of Ps, find the ST in each $D(FP)$, and select the shortest. This approach will compare favorably to the classical approach if there exists an efficient algorithm to find the ST in each $D(FP)$. While the algorithm given by Trietsch and Hwang takes exponential time, the algorithm given by Hwang and Weng takes only $O(n^2)$ time. Triestch and Hwang also showed that the ST in $D(FP)$ is never shorter than the edge constructed in the first step of the recovery stage in the Euclidean construction.

Melzak's algorithm has provided the core of many computer programs. The first published version was by Cockayne and Schiller [61], called STEINER, which can solve cases as large as a 7-point problem. Boyce and Seery [33] and Boyce [34] improved it to allow solutions of up to 10 points, mostly by using a complex multiplication to compute the substitution points. Cockayne and Peters' [60] MFST82 contained a few more geometric tests to eliminate certain cyclic orders. Winter [295] implemented his inverse approach into what he called the GEOSTEINER program and was able to solve problems of up to 15 points. He noted that for $n \geq 15$ the computation time needed to form the unions of "surviving" FSTs dominates the computation time needed for the construction of FSTs. Cockayne and Hewgill [55–57] improved the first aspect and raised the "solvable" range to 30 points. An algebraic approach for the Euclidean construction, which should ease the calculations in these programs, was proposed by Clark [51] and by Hwang and Weng [143].

The exponentiality is more a problem inherent with the SMT, rather than just with Melzak's algorithm. Garey et al. [112] have shown that the SMT decision problem is NP-hard. It is not known to be in NP (i.e., it is not known how to guess a good answer in polynomial time). However, the discretized version (with all distances rounded up) is NP-complete. For NP-hard problems, "decomposability" is often an important consideration. For the SMT problem, *decomposability* means the partitioning of N into subsets $N_1, \cdots, N_m$ such that $N = \cup_{i=1}^{m} N_i$, $|N_i \cap N_j| \leq 1$ for all i, j, and the union of the SMTs for each N_i is an SMT on N. There are three decomposition theorems so far. The first is due to the "double wedge" property as given by Gilbert and Pollak [118]. The second is a consequence of the "Steiner polygon" intersecting itself as given by Cockayne [54]. The third is a recent result of Hwang et al. [142] who gave conditions under which a quadrilateral $ABCD$, where (A, b) and (C, D) are two pairs of adjacent points on the Steiner polygon, can be removed to decompose the Steiner polygon into two smaller polygons linked by an edge.

2.2 Heuristic Algorithms and Bounds

An important consideration for NP-hard problems is the existence of efficient heuristics with provably good performance. A minimal spanning tree (MST) algorithm can be thought of as a heuristic for SMT and can be implemented in $O(n \log n)$ time as first demonstrated by Shamos and Hoey [237]. Define the *Steiner ratio* as

$$\rho = \min_{N} \frac{\text{length of an SMT on } N}{\text{length of an MST on } N}.$$

Then, ρ is a measure of how good the minimal spanning tree is. Gilbert and Pollak [118] conjectured $\pi = \frac{1}{2}\sqrt{3}$ and proved it for $n = 3$. Later, Pollak [206] proved it for $n = 4$ (see also Du et al. [86]), and Du et al. [85] proved it for $n = 5$; see also Friedel and Widmayer [108] and Booth [31]. Recently, Yu and Xu [309] and Rubenstein and Thomas [225] independently announced proofs for $n = 6$. The variational approach proposed by Rubinstein and Thomas [224] seems to be particularly suitable for studying ρ and may push the proof to $n = 7$ or 8. They [226] also studied all the critical points for ρ for $n = 3$ and 4 and showed that the Poincaré–Hopf formula for indices of a vector field is applicable for $n = 3$. For general n, the lower bound for ρ has been steadily pushed up from .5 (Beardwood et al. [11]) to .57 (Graham and Hwang [121]), to .74 (Chung and Hwang [49]), and to .8 (Du and Hwang [78]) through geometrical arguments. Chung and Graham [47] obtained $\rho \geq .824$ by computer-aided calculations.

Chang [38, 39] and Thompson [265] (also see Korhonen [164] and Beasley [14]) first had the idea of converting a minimal spanning tree to an ST through a sequence of "local Steinerizations," i.e., add an S-point to connect three points "close" to each other and delete an edge from any loops formed. This idea has been the backbone of many heuristics for both the Euclidean and the rectilinear distance metrics, though the exact definition of local Steinerization differ. Chang and Thompson did not tailor their algorithms from the computational complexity viewpoint since it was not fashionable then. A literal implementation of Chang's algorithm requires $O(n^4)$ time. Smith and Liebman [245] gave a similar but more complex $O(n^4)$ time algorithm. Smith et al. [243] gave an $O(n \log n)$ algorithm that used Voronoi diagrams and Delauney triangulations. Smith [248] gave an $O(n^2)$ algorithm that always linked a new point to the growing tree by a straight line. Recently, Beasley [14] gave an algorithm that constructs an MST on $N \cup S$, where S is a subset of the S-points contained in the SMTs for all 4-point-connected components of the MST. The improvement using these heuristics over a minimal spanning tree is typically 3–4% in expected length; but no improvement in the worst case has been demonstrated (see also [259]).

A completely different heuristic using "simulated annealing" was proposed by Lundy [181, 182]. Such an approach will successively perturb candidate SMTs searching for an (locally) optimal solution. The perturbation scheme employed replaces three edges [s,1], [s,2], [3,4] by the three edges [s,3], [s,4], [1,2], where the third point adjacent to the S-point s is an N-point. Note that such a scheme not only improves the embedding of a topology but also explores many different topologies. Some experimental results were given. Hesser et al. [131] gave another simulated annealing algorithm and compared it to an analogous "genetic algorithm."

Karp [161] gave a partitioning algorithm for the traveling salesman problem but commented that the algorithm can be modified in a straightforward way to the SMT problem, Euclidean or rectilinear. Define a *probabilistic ε-approximation scheme* to be an algorithm that constructs a tree T in time polynomial in $1/\varepsilon$ such that the ratio $r(T)$ of the lengths of T and of an SMT is bounded by $1 + \varepsilon$. Karp showed that his algorithm is a probabilistic ε-approximation scheme. A

particular implementation yields a tree T in $O(n \log n)$ time such that $r(T) \to 1$ with probability one.

Finally, Soukup proposed a scheme motivated by a physical analog approach [252]. It involves a root-finding step for a derived polynomial. Experimental results were given by Soukup and Chow [253].

2.3. Special Cases

Another important avenue of research with NP-hard problems is the study of special cases. By using the special geometry of the given set N, one may obtain an explicit construction or a fast algorithm for an SMT. The first special set N is the set of vertices of a regular n-gon as studied in the first genuine SMT paper of Jarnik an Kössler [152]. They obtained the SMT for $3 \leq n \leq 5$ and also proved that for $n = 6$ or $n \geq 13$ the SMT is the perimeter of the n-gon minus a side. Du et al. [84] (see also Kotzig [165] and Weng [288]) finished the problem by proving the same for $7 \leq n \leq 12$. When N is a set of vertices on a circle with radius r, Du et al. [81] proved that the SMT is the perimeter of the n-gon minus a longest side when at most one side can be longer than $.557r$. Curiously, the result depends crucially on the value of a lower bound of ρ. Provan [210] gave an approximation algorithm for the case where the N-points were on their convex hull.

Other special sets N whose SMTs can be efficiently constructed are ladders (Chung and Graham [45]) and their generalizations (Du and Hwang [80]), vertices on a zigzag line (Du et al. [83]), and vertices on a splitting tree (Hwang et al. [145]). Gardner [111] and Chung et al. [43] constructed STs for $n \times n$ checkerboards and conjectured that the construction is optimal for n a power of 2. Hwang [141] constructed STs for a Chinese checkerboard and proved that the construction is optimal for all order $n > 2$.

We can also restrict N by fixing n. As mentioned in the Introduction, the $n = 3$ SMT was well studied by 17th century mathematicians. The $n = 4$ case was studied by Pollak [206], Ollerenshaw [200], Du et al. [86], and Du et al. [82]. It should be noted that a good understanding of the $n = 3, 4$ cases are crucial in obtaining two of the decomposition results, one by removing a triangle, and the other, a quadrilateral. One can also restrict the number of S-points to be k regardless of n (called a k-Steiner problem). Georgakopoulos and Papadimitriou [115] gave an $O(n^2)$ algorithm for the 1-Steiner problem.

2.4. Generalizations

So far we have only discussed the Euclidean SMT problem on the plane. For higher-dimensional Euclidean space, Gilbert and Pollak [118] showed that most of the basic properties of SMT, including the 120° and degree-3 conditions on S-points, still hold. The extension of the Euclidean construction of an FST to d dimensions is nontrivial and has not been attempted. Even the SMT for the corners of a unit cube is unknown [254]. In fact, Smith [249] argued that, in general, for $d = 3$ and $n = 4$, the coordinates of the two S-points of a full Steiner

tree can be expressed as the roots of a generally irreducible 8th degree polynomial; hence, no geometric construction is possible. Dolan et al. [75] gave an $O(n \log n)$ time heuristic for finding an SMT on the unit sphere.

Let $E(n)$ be the expected length and $M(n)$ be the maximum length of an SMT for n points distributed in a unit region of d-space. Beardwood et al. [11], using a partitioning argument that later led to the partitioning algorithm of Karp [161], showed that

$$E(n)n^{-d-1/d} \to \beta_d$$

with probability 1 for any bounded probability distribution and N lying in a Lebesque measurable region. Steele [255] showed that the convergence to β_d follows from a result on subadditive Euclidean functionals (also true for the rectilinear metric). Few [98] proved an upper bound of $M(n)$ for N in a unit square.

Let $(r_d)^d$ denote the ratio of the best packing density and the volume of the unit sphere. Moran [197] showed that $\beta_d' \leq (2d)/(d-1)r_d$ for N with unit diameter, and Smith [248] combined it with the result

$$.12098\sqrt{d} \leq r_d \leq .15975\sqrt{d}$$

to obtain $\beta_d' \leq .3197\sqrt{d}$. He also strengthened Few's argument to obtain $\beta_d' \leq .2420\sqrt{d}$.

Let ρ_d denote the Steiner ratio in d-space. Graham and Hwang [121] proved that $\rho_d \geq \sqrt{3} \approx .57$. Combined with Smith's result, we obtain $\beta_d' \geq .13970\sqrt{d}$, as $d \to \infty$. Moran [197] bounded $M(n)$ for sets in d-space with bounded diameter. Recently, Du [77] improved the .57 bound to .615.

The Steiner ratio was conjectured [44, 118] to be achieved by the corners of the d-dimensional simplex. However, SMTs for d-dimensional simplices are not known, in general. Chung and Gilbert [44] gave an upper bound for their lengths that approaches .669842 for large d. Beardwood et al. [11] showed that β and $\beta' \geq \frac{1}{2}$ in any metric space. Recently, Smith [249] disproved the conjecture for $3 \leq d \leq 9$ by showing that using the corners of regular octahedron achieves a smaller ratio.

Other types of generalizations have been attempted for the Euclidean SMT problem. The first is a generalization of N from a point-set to other objects, as well as adding other objects in the space as obstacles. The second is a generalization of the Euclidean space to some other metric space. The third is Gilbert's flow-dependent networks.

For the first type, Cockayne and Melzak [58] extended Melzak's algorithm to the case that N is a set of point-sets. Weng [289] used the hexagonal coordinate system to treat the case that N is a set of regions. Trietsch [267] gave a finite-time algorithm for adding points to an existing network. The SMT problem with obstacles is a practical problem in pipeline layout or routing heating mains (see Smith and Liebman [245], Liestman [179], and Smith [240]). Provan [211] has given effective approximation algorithms for obstacles where the obstacles are

polygons and there is only a constant number of N-points not on the sides of polygons. Smith and Winter [247] considered a single convex obstacle with k corners and gave an $O(k)$ time algorithm for $n = 3$ and an $O(k^2)$ time algorithm for $n = 4$; a polynomial heuristic was also outlined.

For the second type, Soukup [251] and Cheung and Soukup [41] studied nonlinear cost and proved that if the cost function is increasing in a certain sense then the degree of each vertex is bounded. He also proved that if the cost function is convex, then an SMT is full. Chang [39] extended his algorithm to the case where the cost function is piecewise linear. Lee [174] discussed how such cost functions (and other generalizations) arise from industrial problems. Howard [134] discussed how these costs arise in modeling drainage networks. Sankoff and Rousseau [233] gave a dynamic programming solution for locating S-points of an FST in an arbitrary metric space in their study of phylogeny. It should be noted that the .5 bound for the Steiner ratio holds for any metric space satisfying the triangle inequality.

For the third type, Gilbert [117] first studied the shortest interconnecting network that must satisfy a given matrix of two-way flows, $f(i, j)$ where the cost of an edge with capacity c is $g(c)$ per mile. When $g(c)$ is a constant, the shortest network is an SMT. But, in general, the shortest network loses many of the properties of an SMT and need not even be a tree. This network was also studied by van de Heyden [271], Trietsch and Handler [269], and Du and Hwang [79]. Werner [310] considered a flow matrix that affects only the usage, but not the size, of an edge. Under a linear cost model, he showed that if the flow demand decreases at least linearly with the distance the minimum cost network is an SMT. In the special case that $f(i, j)$ is positive only for i = source (or j = sink), then we have a *rooted* SMT problem [23, 24, 172]. Some variations are what Smith termed the *directed* SMT problem and the *Steiner minimal circuit* problem [228, 239, 241, 242, 246].

As mentioned in the Introduction, the $n = 3$ case has been alternately generalized as the "Fermat problem," where there is only one additional point and all N-points connect to that point. (This is different from the 1-Steiner problem in which the S-point need be connected to only a subset of the N-points.) If each N-point has a weight and the lengths of the edges to the central point are biased by the weights, then the minimization problem is the "Weber problem." This is the cornerstone of a vast literature for Location Theory. Some interesting work has been done on this problem, often misattributing the problem as a Steiner problem, e.g., [59, 158, 162, 193, 201, 266].

3. THE GRAPHICAL SMT PROBLEM

The graphical Steiner minimal tree (GSMT) problem asks for a minimum tree spanning some designated vertices of a graph. Formally, a graph $G = (V, E)$ is given, with weights on the edges, and a set $N \subseteq V$ is specified. A solution to the GSMT problem is the minimum weight tree subgraph of G that spans N. The vertices not in N with degree ≥ 3 in the tree are called *Steiner vertices*. (A more general formulation asks for a minimum weight subgraph, perhaps cyclic. If the

edge weights are nonnegative, the two formulations are essentially the same. Otherwise, a simple preprocessing step can be used [123] to handle the negative edges.) Hakimi [123] and Levin [178] independently posed this problem. While it is possible to motivate the graphical case from the Euclidean case, the two problems have little in common. Recall, however, that some heuristics for the Euclidean case have graphical origins.

Research on the GSMT problem has taken a familiar route. First, some exponential time exact algorithms for finding the optimal configuration were posed. Then, the GSMT problem and its variants were shown to be NP-complete. Finally, many heuristics and special cases have been considered. Winter has presented an excellent survey [298]. Maculan [186] gave a survey that emphasizes the various exact algorithms.

3.1. Exact Algorithms

Hakimi [123] proposed an exact algorithm that ran in $O(2^{\nu-n}\nu^2 + \nu^3)$ time, where $\omega = |V|$, and $n = |N|$. This algorithm tries all possible choices for the Steiner vertices and for each choice computes the corresponding minimum tree. Lawler [171] observed that it is easily implemented in $O(2^{\nu-n}n^2 + \nu^3)$ time. Levin [177, 178] proposed a dynamic programming approach (i.e., precomputing possible subtrees) that ran in $O(3^n\nu + 2^n\nu^2 + n^2\nu)$ time. Essentially, the same algorithm was given by Dreyfus and Wagner [74] with a time complexity of $O(3^n\nu + 2^n\nu^2 + \nu^3)$; however, this algorithm does additional work at each stage. A branch-and-bound approach has been proposed [104, 238] that uses heuristics to provide good lower bounds and to choose the next edge for consideration in the backtracking process. Of course, it is difficult to analyze this algorithm, but empirical evidence [105] suggests that it compares favorably to the other two when n is about $\frac{1}{2}\nu$; Hakimi's algorithm is favored when n is about ν and the dynamic programming approach is favored when n is small. Johnston [156] empirically studied the above algorithms for the unit-weight case. He found that the branch-and-bound approach was the best (though in rare cases it took an extraordinary amount of time) and that Levin outperformed Dreyfus and Wagner.

Other exponential time exact algorithms have also appeared without analyses. Hakimi [123] proposed a recursive approach to enumerate the various tree topologies and gave a useful theorem to prune the search. Balakrishnan and Patel [8, 9] gave an algorithm that cleverly generates spanning trees for a derived problem in order of increasing weight until a solution to the original problem can be inferred. Aneja [3] reformulated the problem as a set-covering problem, presented as a 0-1 linear programming problem. Several relaxation techniques and improvements were used. Beasley [12] gave a branch-and-bound algorithm where the bounds used Lagrangean relaxations of several 0-1 linear programming reformulations (see also [87, 184, 185]). Further constraints were added [13], leading to improved performance. In fact, the latter algorithm has the best reported performance on large random sparse graphs; with $\nu = 500$ and $n = 125$, less than 6 min were required on some sparse graphs. Another 0-1

linear programming algorithm was given by Wong [302] that combined two known approaches from cognate problems. This formulation admits directions on the edges and, hence, applies to directed graphs, discussed below. Segev [235] used similar techniques for a weighted variant of the GSMT problem. Jain [150] used probabilistic arguments to show that a linear programming relaxation (of an aggregate formulation) can give poor bounds for the integer solution. Goemans and Bertsimas [119] showed that the LP relaxation of the cut formulation is at most half the value of the optimal Steiner tree.

Algorithms for the GMST problem can benefit from preprocessing the graph, so that constraints on possible trees can be discovered. This is true especially for those algorithms that incrementally build the SMT; as a tree is built, new constraints evolve. There are several obvious reductions that can be made (e.g., handling vertices of degree 1) as well as some nontrivial reductions (e.g., using triangle inequalities when possible). These are discussed by Beasley [12], Balakrishnan and Patel [9], Duin and Volgenant [87–90], Voss [275, 276], and Iwainsky et al. [148]. In general, there are no known theorems bounding the length of a GSMT, beyond the results discussed in the heuristics section below.

An early result, due to Karp, was the GSMT decision problem was NP-complete [159, 160]. Many restricted versions remain hard. If G is planar, the problem is still NP-complete [113]. Further, Berlekamp claims that the GSMT remains NP-complete if all edges have unit weights, even if G is bipartite with all edges from N to $V - N$ [114, p. 209]. The problem remains NP-complete for chordal graphs and split graphs [291]. (Hakimi [123] gave several explicit reductions to other problems; today, these reductions are implicit from the theory of NP-completeness.) Provan and Ball [213] showed that the variant of the rooted Steiner arborescence of a directed graph is NP-complete. Provan [209] showed this remains true even for planar acyclic digraphs with in-degree and out-degree at most 2. Hence, there has been much interest in heuristics and special cases.

3.2. Heuristic Algorithms

Many heuristics have been proposed for finding approximate solutions to the GSMT problem. Those in the first group are based on known graph algorithms, usually for the minimum spanning tree (MST) problem. A second group of algorithms use suboptimal variants of exact algorithms. The latter group are not necessary polynomial time.

A straightforward MST-based algorithm has been proposed by El-Arbi [92] and independently by Plesnik [204], Kou et al. [168], Bharath-Kumar and Jaffe [22], and Iwainsky et al. [148]. The algorithm begins by constructing a complete graph G' on the vertex set N with edge weights equal to the lengths of the corresponding shortest paths in the original graph G. Then, an MST is found for G' and each edge of that MST corresponds to a shortest path in G. Let G'' be the subgraph of G that is the union of these paths. The approximate GSMT is the MST of G'', after some obvious postprocessing. Let L_{MST} and L_{OPT} be the length of the approximate GSMT found and the optimal GSMT, respectively. It is known [168] that $L_{\text{MST}}/L_{\text{OPT}} \leq 2[1 - (1/t)]$, where an optimal tree has t leaves

(see also [22, 92, 148, 204]). All the proofs of this bound are based on an observation by Moore (as communicated in [118]) about the length of an Eulerian tour based on an optimal GSMT. These algorithms run in $O(nv^2)$ time, where the construction of G' dominates. Variants are discussed in [204].

Wu et al. [303] proposed an implementation, based on Kruskal's MST algorithm, that does not explicitly construct G'. Recall that Kruskal's algorithm begins with v singleton trees and iteratively connects the two nearest current trees, until one tree remains. To avoid constructing G', the algorithm simultaneously grows "shortest path trees" from each vertex (always adding the least unused edge to one of these trees). When two trees meet, a path between their roots is formed, corresponding to the next edge that Kruskal's algorithm on G' would have used. The algorithm runs in $O(e \log v)$ time, where $e = |E|$ and has the same performance bounds as above. A similar algorithm was independently given by Wang [283]. Widmayer [292] compared all the above algorithms and showed that no one is always superior to the others (see also [205]).

Widmayer [293] refined the above algorithm (to better handle difficult cases) and used an efficient priority queue data structure: the Fibonacci heap. The resulting algorithm has the same performance bounds and runs in $O(e + (v + \min\{e, n^2\}) \log v)$. This algorithm is never (asymptotically) slower than the above algorithms and can be faster. Recently, Mehlhorn [190] has improved on the above variations by giving a simple implementation that runs in $O(e + v \log v)$ time. Instead of G', this algorithm uses another graph that is easier to find (using a single shortest-path computation) and that will give the same answer that G' would have produced. Kou [166] and Kou and Makki [167] presented a similar algorithm with the same time and performance bounds; they also analyzed the improved performance on sparse problems. It has been shown [169] that the Kruskal-based approach is (essentially) optimal for random unweighted graphs, i.e., where each edge is present with probability p. In that case, the expected size of the GSMT is approximately $n(\log v)/(\log vp)$. (The trivial algorithm of using a shortest-path spanning tree is also nearly optimal for random graphs!)

Takahashi and Matsuyama [260] gave a heuristic based on Prim's MST algorithm. Recall that Prim's algorithm begins with one singleton tree and iteratively connects the nearest vertex to it, until there is one tree. Their algorithm begins with one vertex of N and connects, with a path, the nearest other point in N. At each step, there is some tree and the nearest unused vertex in N is connected, with a path, to some vertex in the tree. The performance bound is $L_{\text{PRIM}}/L_{\text{OPT}} < 2[1 - (1/n)]$, where L_{PRIM} is the length of the approximate GSMT. The algorithm runs in $O(n\, v^2)$ time. A variant has been proposed that has the same performance bounds and time complexity but uses less space [147]. The algorithm was independently given by Wang [283] and it is related to the heuristics above [218]. Segev [235] and Ma [183] have given simple Prim-based algorithms. A related algorithm was given by Bharath-Kumar and Jaffe [22].

Rayward–Smith's heuristic [107, 217, 218] explicitly selects the Steiner vertices, while the above approaches implicitly choose them. It is Kruskal-based

but does not connect the two nearest subtrees. Instead, two or more trees may be connected to a Steiner vertex, chosen because it minimizes an "average distance" to the set of all current subtrees. In particular, $T_1, T_2, \cdots, T_k$ are connected to p if $\sum_{i=1}^{k} d(T_i, p)/(k-1)$ is minimal over all p and all such sets of trees, where d is the appropriate distance function. It runs in $O(v^3)$ time and has good empirical performance [218, 285]. The worst-case bound on the performance ratio is claimed to be $2[1 - (1/n)]$ [287]; Plesnik [205] gave a counterexample to a key lemma. Its average case analysis has also been studied [286]. Bern and Plassmann [21] showed that this algorithm has a 4/3 performance bound on complete graphs with edge lengths of 1 and 2. While the latter case is specialized, it is the only known example of a constant bound less than 2; further, they show the problem is "SNP-hard."

Minoux [195] suggested a greedy algorithm that begins with an MST of G', as defined above. At each step, a new vertex from $V - N$ is added to the graph (with edge lengths inferred from G) as long as the MST length decreases. It is argued that the $O(n^3)$ worst-case time is not encountered in practice due to a supermodularity property.

Chen's two heuristics [40] are based on a subroutine for the case of $n = 3$. (We note that other special cases of $n = 2$ and $n = v$ are trivially handled by shortest-path and MST routines, respectively.) An exact $O(e \log v)$ time algorithm is given for this case; if the edges have unit weights, then $O(e)$ time suffices. The first heuristic uses the routine to introduce "local Steinerizations" into a Prim-based approach. The second heuristic generalizes Kruskal's algorithm by linking together three subtrees at each step. The time complexities are $O(\Delta\, n\, e \log v)$ and $O(n^2\, e \log v)$, respectively, where Δ is the maximum degree.

Plesnik [204] presented another heuristic based on the identification of neighborhoods that can be contracted to a point. Before contraction, the GSMT problem is solved within each neighborhood, and then the problem is recursively solved on the contracted graph. That algorithm and improved variant [205] have a performance ratio of 2. Two trivial heuristics have been proposed [22, 260] and shown to have poor worst-case performance: simply pruning an MST of G and pruning a shortest-path spanning tree of G.

Several suboptimal variations of exact algorithms have been proposed. Aneja [3] gave a greedy variation of his set-covering approach. Wong [302] used an MST routine as a shortcut in his exact algorithm. Segev [235], in his work on a generalized GSMT problem, used an MST routine, as well as proposing a subgradient heuristic. Computational experience with all of these has been encouraging. For related results, see [149, 257, 262, 263, 272].

3.3. Special Cases

Polynomial time algorithms can be found for classes of graphs where enough structure ensures restrictions on possible GSMTs. It is discouraging that the problem remains hard for planar graphs and, in fact, for grid graphs (see the

next section). However, the problem can be solved efficiently for some classes of graphs where there are strong decomposition theorems. Related results are discussed in the next section.

The class of partial k-trees and its various subclasses [154, 155] have had many efficient algorithms proposed for the Steiner problem, as well as for other NP-complete graph problems that use decomposition results. Perhaps the first result was by Wald and Colbourn [279] for outerplanar graphs. They generalized the result to the related classes of 2-trees, partial 2-trees, minimum IFI networks, and their applications [278, 280–282] (see also [208]). Takamizawa et al. [261] indicated an algorithm for series-parallel graphs (see also [4, 32, 64, 216]). Winter gave similar results for Halin networks [299]. The general case of partial k-trees, for a constant k, was treated in a very general way by Bodlaender [28]; a parallel formulation was also given [29]. All these algorithms run in linear time.

The class of perfect graphs and its various subclasses [154] have been studied. Polynomial time algorithms have been given for permutation graphs (and cographs) [63], distance–hereditary graphs [67], homogeneous graphs [68], and (6,2)-chordal bipartite graphs [7]. White et al. [291] gave an $O(\nu^3)$ time algorithm for strongly chordal graphs (and, hence, interval graphs) with unit edge weights. The problem remains NP-complete for strictly chordal graphs with strictly triangular, or more general, weight functions [154]. Recall that the problem remains NP-complete for bipartite, comparability, split, and chordal graphs, as well as for perfect graphs in general.

Planar graphs can be solved in polynomial time if the vertices of N are found on just a constant k faces of some given embedding of the graph, i.e., it is *k-planar*. (Bienstock and Monma [25] have polynomial-time algorithms for finding an embedding for a fixed k, if it exists.) The *nearly* k-planar case (where one vertex of N may not be on the k faces) was solved by Erickson et al. [93] and Bern [17] using the dynamic programming techniques of Dreyfus and Wagner. The time complexities are $O(\nu\, n^{3k} + (\nu \log \nu)n^{2k})$ and $O(\nu\, n^{2k+1} + (\nu \log \nu)n^{2k})$, respectively. Provan [210] generalized these results by casting k-planarity as a specific "convexity" property. (Beinstock and Monma [26] presented polynomial time algorithms for recognizing some of these convexity properties.) Bern extended this to n' interior points and gave an $O(\nu(n - n')^3 3^{n'} + (\nu \log \nu)(n - n')^2 2^{n'})$ time result. Bern and Bienstock [18] showed that if all the N-points were within a bounded number of "layers" of the infinite face of a planar embedding then the time bound is $O(n^4)$. (Beinstock and Monma [27] showed how to recognize graphs with a small number of layers.)

3.4. Generalizations

There have been three principal avenues of generalization: changing the definition of the Steiner network, adding vertex weights, and using directed graphs.

Krarup (in an unpublished 1978 memo) suggested finding the minimum-weight 2-connected (or 2-edge-connected) subgraph that spans N in G, instead

of a tree. Winter has given polynomial-time exact algorithms for outerplanar graphs [296], series-parallel graphs [297], and Halin networks [300]. In the latter paper, the 3-connected case was solved. Voss and Mehr [277] gave two heuristics for the 2-edge-connected case. Duin and Volgenant [87] consider Steiner forests of at most m components. Duin and Volgenant [91] also consider multiweighted Steiner trees, where each edge has a primary and secondary weight. The N-points are connected with a tree using primary weights and the points in $V - N$ are connected to this tree with subtrees using the secondary weights. Cornuejols et al. [64] considered the "Steiner traveling salesman problem." Chung et al. discussed "Steiner caterpillars" [48] in connection with the analysis of a search problem. Voss [273] gave a branch-and-bound algorithm for the degree-constrained case (see also [274]). Bharath-Kumar and Jaffe [22] considered optimizing both total length and maximum path length from a source vertex. They adapted known heuristics and got weak $\sqrt{n}$ performance bounds on both quantities simultaneously. Chiang et al. [42] and Sarrafzadeh and Wong [234] gave polynomial time algorithms for the case when the length of the longest edge (but not necessarily the total length) is to be minimized.

Segev [235] proposed using vertex weights in addition to edge weights. He concentrated on the single-point weighted Steiner tree problem, of which the GSMT problem is a special case. All vertex weights are negative (or nonpositive) and there is a distinguished vertex s. The desired tree includes s and is of minimum total weight. The formulation is motivated by having edge weights be the cost of establishing connections (to s) and the vertex weights are the payoffs. Exact and heuristic approaches are given. The formulation is actually in terms of directed graphs. Duin and Volgenant [87] discussed the general problem, where an s is not specified.

Smith [239] worked on a "directed Steiner minimal tree" problem as described in Section 2.4 (see also the survey paper [241]). In related work, Smith and Liebman [246] gave a heuristic based on local Steinerizations of a rooted spanning arborescence, where the local updates use a Weber point instead of a Steiner point.

Wong [302] used a directed graph model with a distinguished vertex s and asked for a minimum-weight directed tree, rooted at s, that spans N. It was cast as a multicommodity network flow problem. He gave exact and heuristic algorithms, both discussed above (see also [76, 87, 180]). Voss gave several problem reduction techniques [275] that were used with Wong's algorithm. Maculan and Candia-Vesar [187] gave upper and lower bounding procedures for an exact algorithm, based on [12]. Other integer programming have been presented [5, 52, 202], which are summarized in [186]. Suurballe [258] briefly mentioned similar techniques. Nastansky et al. [198] used a similar model but restricted their attention to rooted directed acyclic graphs, with root s. They gave an exact enumerative scheme that can be time-limited to give suboptimal answers. Their model is largely motivated by an m-dimensional grid graph with all edges directed away from the origin s and is discussed further below. Jain [151] used probabilistic arguments to show that a linear programming relaxation (of a disaggregate formulation) is within a constant of optimal in the directed case.

Provan [209] treated the special rooted directed planar case where all the vertices but the root are on the boundary of an embedding. Duin and Volgenant [91] extended their analysis of the multiweighted Steiner tree problem to directed graphs. Balas [10] discussed the three-point-directed problem in terms of convex hulls of certain polyhedra. Wald and Sorenson [282] considered symmetric-directed graphs and do not specify the root of the Steiner tree. Further, they may require some edges be used. They only present an exact algorithm for 2-trees. The problem is motivated by query inference applications for databases.

4. THE RECTILINEAR SMT PROBLEM

Rectilinear Steiner minimal trees (RSMT) are related to both the Euclidean and the graphical cases. As with the Euclidean case, the basic problem has the n points of N located in the plane, and additional Steiner points may be used in constructing the minimal length tree. The distance metric is the rectilinear or Manhattan metric, i.e., the distance from a to b is $|x_a - x_b| + |y_a - y_b|$.

The expected length of an RSMT can be bounded. An early result is for rectilinear minimum spanning trees (RMSTs) from which a similar bound follows for RSMTs. Let RMST(n) be the random variable for the length of the RMST for n points drawn uniformly from the unit square. Gilbert [116] established that $E[\text{RMST}(n)] = O(\sqrt{n})$. Few [98] showed that $F(n) \leq \sqrt{n} + O(1)$, which was sharpened by Chung and Graham [46] to

$$\sqrt{n} + O(1) \leq F(n) \leq \sqrt{n} + 1 + o(1),$$

where $F(n)$ is the length of the longest possible RSMT on any n points drawn from the unit square. Komlos and Shing [163] showed that $\text{RSMT}(n) \geq \sqrt{n}/5$ with probability $1 - o(1)$, where RSMT(n) is the random variable for the length of the RSMT for n points drawn uniformly from the unit square.

The first investigation of the RSMT problem was by Hanan [126]. He established the following important "dimension reduction" result: Extend horizontal and vertical lines through each of the points. Define the graph $G_N(V, E)$ by letting V be the set of intersections of these lines and there is an edge between two vertices if they are directly connected by a line, horizontally or vertically. Hanan showed that an RSMT is contained in G_N, i.e., the segments of the tree are composed of edges of G_N. An approach for generating candidate RSMTs has been given [199].

4.1. Exact Algorithms

While any exact algorithm for the GSMT problem could be used, only one exact algorithm for the RSMT has been proposed. Yang and Wing gave a straightforward branch-and-bound algorithm [305] and improved it somewhat [307]. It appears to be applicable only for $n < 10$. Sankoff and Rousseau [233] gave an $O(n)$ time dynamic programming algorithm for the case when the

topology is fixed. Thomborson et al. [264] gave an $n^{O(\sqrt{n})}$ time algorithm that computes the optimal RSMT with high probability, when the points are uniformly distributed in the unit square.

The RSMT decision problem is NP-complete [113]. Therefore, there has been much interest in heuristics and special cases.

4.2. Heuristic Algorithms

As with the Euclidean SMT problem, a minimum spanning tree algorithm is a simple heuristic for the RSMT. The rectilinear minimum spanning tree can be found with any MST algorithm. (An embedding of the edges of an RMST may have edges that "overlap.")

Hwang [136] established many characterizations of RSMTs that allow their form to be sharply limited without loss of generality (see also [220]). For example, we may assume all tree edges consist of at most one horizontal and one vertical segment. He was able to establish that

$$\frac{L_{\mathrm{RMST}}(N)}{L_{\mathrm{OPT}}(N)} \leq \frac{3}{2},$$

where $L_{\mathrm{RMST}}(N)$ is defined to be the length of the RMST and $L_{\mathrm{OPT}}(N)$ is the length of the RSMT. Empirically, evidence for small N for random data (no details given) suggests the expected value of the ratio is 1.13 [306]. No heuristics discussed below improve on the $\frac{3}{2}$ worst-case bound. While an RMST can serve as a good approximation to the RST, Chung and Hwang [50] showed that the semiperimeter of the least bounding rectangle, an often-used approximation, can be worse by a factor about $\frac{1}{2}\sqrt{n}$. (Such an approximation is optimal if $n = 3$ [126].)

The first nontrivial heuristic was proposed by Hanan in an unpublished report [125]. It found a tree using a Prim-based line-sweep algorithm, i.e., it built the tree in a greedy fashion from left to right. Four orthogonal sweeps were suggested. The best implementation seemed to require $O(n^2)$ time (e.g., [137]). Servit [236] suggested a simplified approach, which did not optimize each greedy step, that admitted an $O(n \log n)$ time implementation. Richards [219], using data structures used in computational geometry, was able to implement Hanan's full algorithm in $O(n \log n)$ time. The trees produced are slightly inferior to those of later heuristics. These approximate RSMT's are 4% better than the RMST, while later heuristics are usually 7% to 9% better.

Fu [109] gave an unanalyzed manual technique to iteratively improve a spanning tree by creating and breaking cycles. Hanan [127] disproved Fu's claim of optimality. An $O(n^4)$ time solution that uses several ad hoc stages was proposed [245]. It began by selecting a linear-sized subset of the n^2 vertices of G_N as candidates for Steiner points. Then, it evaluated each in isolation and entered them into a priority queue. The queue determined the order the candidates were tried. Beasley's [14] $O(n^2 \log n)$ time algorithm also begins by selecting

candidate Steiner points; the choice is based on optimal solutions to 4-point connected subproblems.

Yang and Wing proposed a suboptimal branch-and-bound algorithm, which, while still exponential time, appears to be applicable for $n < 30$. When compared with known exact results, the answers were remarkably close to optimal [306–308]. It simply uses Prim's algorithm, but instead of choosing one of the two possible one-bend orientations of the new wire, it explores both.

Several heuristics begin with an RMST. Since the underlying complete graph has $\Omega(n^2)$ edges, to improve on the $O(n^2)$ time bound requires preprocessing to reduce the size of the underlying graph. The rectilinear Voronoi diagram (defined analogously to the Euclidean case) has a Delauney triangulation that contains enough edges to find the RMST. Since the triangulation corresponds to a planar graph, the RMST can be found in $O(n)$ additional time. Hwang [139] showed, with standard divide-and-conquer techniques, how to find the Delauney triangulation in $O(n \log n)$ time (see also [173]). It should be noted that the details for the rectilinear Voronoi diagram are more complicated than for the Euclidean case.

Hwang gave a heuristic for the RST problem, using the above RMST solution, that was based on earlier work by Lee et al. [175]. That work built an RSMT in Prim fashion. It used a "3-point connection scheme" instead of simply connecting the nearest unused vertex. This involved a constant time search around the intended 2-point connection for three points that could be connected in a Steiner fashion, perhaps introducing a new Steiner point. It ran in $O(n^2)$ time with most of the time spent on deciding which point to connect next. Hwang [138] proposed an $O(n \log n)$ implementation that began with an RMST and from that inferred an ordering of the vertices that the above algorithm could use to decide which node to connect next and where to try to connect it.

Smith et al. [244] proposed an $O(n \log n)$ time approach based on iteratively improving the RMST found over the Delauney triangulation (see also Beasley [14]). The technique is complex and used 4-point Steinerizations as well. A simplified approach was proposed by Richards [219]. Ho et al. [132] gave a linear time algorithm for finding the optimal RSMT derived from an RSMT where all the edges are L-shaped. They also solve the case where the edges are "staircases" [133].

Bern and de Carvalho [19] investigated Kruskal-based approaches attributed to Thompson. Variations attributed to Ng and themselves were proposed that were supposed to be faster than, but usually inferior to, the original. Thompson began with n singleton trees, and at each stage, a new wire connects two trees and the shortest such wire is chosen. This wire may not necessarily connect terminals, but it can be "slid" so that it at least has one endpoint at a terminal, Steiner point, or corner of a previous wire. There are only $O(n)$ such positions. They assume the points are grid points of an $m \times m$ grid. While m is theoretically unrelated to n (except $m > \sqrt{n}$), they assume a data structure in size $O(m^2)$; in their application, domain m had a reasonable bound. Thompson's algorithm is implemented in $O(m\, n^2 \log n)$ time and their variation takes $O(m$

n^2) time. Their algorithms can, by a small alteration, be made to run in $O(n^2 \log n)$ time [219].

Bern [16] has shown that the expected savings of their approximate RSMT relative to the optimal RMST is bounded away from 0 in the limit. The proven savings, $.0014n$, is much less than empirical studies indicate it should be. It is also shown that a linear number of Steiner points are expected. The results are for points drawn at random from the unit square distributed according to a Poisson process with intensity n. Similar results for a uniform distribution of n points were conjectured in [16]. Recently, Hwang and Yao [146] proved this conjecture and improved the savings to $.0056n$.

Komlos and Shing [163] applied Karp's partitioning algorithm to the rectilinear case. They start with n points, assumed to be uniformly distributed in the unit square, and a parameter t. Next, iteratively partition, using medians, the square into small rectangles until each rectangle contains approximately t points. Find the optimal RSMT for each rectangle and combine these trees to give the final tree, after some clean-up steps. If each optimal subproblem is solved in $f(t)$ steps, then the algorithm runs in $O(f(t)n + n \log n)$ time, which is $O(n \log n)$ for $t = O((\log n)^{1/4})$ if, say, Levin's algorithm is used on the grid graph. They show that their approximation is within a factor of $1 + O(1/\sqrt{t})$ of the length of an optimal RSMT, with probability $1 - o(1)$. Another slightly faster algorithm was presented that depends heavily on the uniformity of the point distribution. Hence, even though the same probabilistic bound above holds, this algorithm has a worst-case performance that is at least a factor of t of the length of an optimal RSMT. No implementation was attempted.

Jiang and Tang [153] began by collecting statistics for all subproblems using 3 vertices from N. Then, after some additional preprocessing, a Kruskal-based algorithm was given that favors Steiner points and edges suggested by the statistics.

Several recent heuristics are presented as automatic wire routing algorithms, for multiterminal nets. Xiong [304] presented an interesting two-phase approach. It is based on the underlying grid graph; to find a nearest neighbor, a breadth-first search is done. The first phase builds an RMST using Prim's algorithm and tries both embeddings of each edge and marks commonly used grid edges. The second phase is much the same as the first except embeddings are chosen that favor marked commonly used edges; some backtracking is done. Hsu et al. [135] used straightforward versions of Prim's and Kruskal's algorithms, but he tried different schemes when a "nearest-neighbor" calculation was done. The schemes cleverly take into account the "magnetic attractions" of the other points in N. Lee and Sechen [176] gave a detailed treatment of the 3-point steinerization of an MST approach; their details were influenced by constraints of the row-routing problem.

Another wire-routing algorithm was proposed for implementation on a (specific) parallel architecture. Watanabe and Sugiyama [284] suggested using a "wavefront" approach that has a running time proportional to the size of the tree, not just the number of points. The number of processors in the proposed implementation is proportional to the length of the tree. It is based on succes-

sive 3-point subproblems. Hambrusch and TeWinkel [124] gave a parallel algorithm that is ultimately used to compute the connected components in an $n \times n$ mesh of processors.

4.3. Special Cases

The principal special cases studied restrict the placement of the points in N. Aho et al. [2] studied the case where all the points lie on the boundary of the grid graph G_N, which has p rows and q columns. They gave an $O(p^2q + p\,q^2)$ or $O(n^3)$ time dynamic programming algorithm for this case. It was improved to $O(n)$ time if $p = 2$. Agarwal and Shing [1] gave an $O(p + q)$ time algorithm, which was simplified by Cohoon et al. [62]. Recently, Pulleybank and Steger [214] gave another linear-time algorithm using arguments based on cut-sets. An alternative approach has been given that has an exponential worst-case time but quadratic performance for certain wiring applications [157].

An extension to the above approach is to restrict the points of N to lie on the boundary of their rectilinear convex hull; i.e., each point has one empty open quadrant. Independently, Bern [15] and Provan [210] gave $O(n^6)$ time algorithms for this case, based on their algorithms for 1-planar graphs. Bern improved the result to $O(n^5)$ time, by using advanced data structures. Recently, Bern [17] generalized this to an $O(n^2\,h^3\,3^{n-h})$ time algorithm where h of the N-points lie on the rectilinear convex hull. Richards and Salowe [221] gave a linear time algorithm for the case where the convex hull has a constant k number of sides; in particular, it runs in $O(k^5\,n)$ time. Bern and Bienstock [18] gave an $O(n^{6l-1})$ time algorithm for the case were all the N-points would be removed if l consecutive rectilinear convex hulls were removed.

Farley et al. [94] looked at minimal distance rectangle trees (MDRT). Basically, a rectangle tree is a subgraph of G_N where every point is on the outer boundary and is adjacent to two other boundary points. An MDRT is a rectangle tree that spans N and preserves the length of shortest paths between N-points in G_N. They showed that given an instance restricted to an MDRT then the RSMT can be found in linear time. This is a restricted form of the convex case above. Winter [301] gave a polynomial time algorithm for deciding if an MDRT exists and exhibiting one if it does.

4.4. Generalizations

There are few generalizations of the RSMT per se. Smith and Gross [242] consider generalizing the metric from the rectilinear, or L_1, metric to the L_p metric, i.e., $(|x_a - x_b|^p + |x_a - x_b|^p)^{1/p}$. They argue that the only interesting range is $1 \leq p \leq 2$, with respect to applications. They begin with an MST constructed using a generalized Voronoi diagram and do local Steinerizations.

It is obvious how to generalize the RSMT to d-dimensions. Gilbert and Pollak [118] and Foulds [102] have shown that the upper bound on $L_{\text{RMST}}/L_{\text{OPT}}$ goes to $\frac{1}{2}$ as d increases (see also [227]). Sankoff and Rousseau's [233] algorithm is for the d-dimensional rectilinear metric, but only for fixed topologies. It runs in

linear time for fixed d, but its performance is exponential in d. Recall that Steele's result on $E(n)$ for the Euclidean metric is also good for the rectilinear metric $M(n)$. Snyder [250], using a similar argument, proved $M(n)n^{-d-1/d} \to \beta_d'$ with probability 1 and also showed that $\beta_\delta' \leq d\ 4^{d-1/d}$. Smith improved the bound to

$$\frac{d}{4e} < \beta_d' \leq \frac{d}{2e}$$

by a randomizing method first given in [249] for a similar problem.

As mentioned in the Section 3.4, Nastansky et al. [198] discussed rooted directed versions of a d-dimensional rectilinear metric. In particular, the origin is an N-point and all edges of the corresponding grid graph are directed away from the origin. The desired subgraph is a rooted Steiner arborescence. Trubin [270] claimed a polynomial time algorithm for this problem in two dimensions. Rao et al. [215] found an error in Trubin's algorithm and gave an $O(n \log n)$ time heuristic algorithm that is never worse than twice optimal (see also [188]).

5. PHYLOGENETIC TREES

The construction of a phylogenetic or evolutionary tree is a basic problem in biological systematics. Such a tree has vertices identified with extant species (operational taxonomic units) and other vertices associated with hypothetical ancestral species. Typically, the tree is rooted and the root is regarded as the progenitor. The fundamental conjecture is that the "maximum parsimony principle," that is, "distance" between two species is (nearly) equal to the minimum number of evolutionary steps (mutations, etc.) needed to transform one into the other. This is an area where Steiner trees are the appropriate mathematical model, but it is difficult to agree on the point space and what metric should be used.

The literature is, of course, quite large and only relevant pointers into it are given below. Felsenstein presents an extensive review of the terminology and algorithms [96]. Explicit biological motivation is given only rarely in the cited literature [36, 37, 97]. Often, the easier problem of finding the phylogenetic tree given its topology is solved, i.e., finding the best positions for the given Steiner points. This is justified since an educated guess of the topology can be made from other biological evidence. The literature can be grouped into three major areas.

The first major area presumes that experts have been able to identify *characters* of the species. Further, each species is labeled with m states corresponding to m different characters. Each character has an associated set of character states, which can be strictly ordered, partially ordered, or unordered. A problem is *cladistic* if they are strictly ordered and *qualitative* if they are unordered. A character with two states is *binary*. Species are points in the space of the m-dimensional cross-product of the character states. Suppose that the "distance" between the states of a given character are given. The distance between

two points is the "rectilinear" distance, i.e., the sum of the m distances across the various dimensions. Usually, character states are identified with integer values and the distance is the rectilinear metric itself. The lowest integer is the ancestral state in the cladistic case; in the "qualitative" case, the ancestor is to be chosen but it is cladistic relative to that choice (since the integers imply an ordering). (For additional discussion, see [35, 95, 223].)

Four subcases are identified [69], each with its variants. A *Wagner* tree is a Steiner tree where the character states are integers; typically it is unrooted. A *Camin–Sokal* tree is a Steiner tree, where all edges are between points that reflect a change from an ancestral state to a derived state [35, 130]. This is equivalent to the directed rectilinear Steiner problem where all edges are directed away from the origin (progenitor); the algorithms from the previous section can be applied here. There are two subcases: The root is specified or it is chosen. A *Dollo* tree is a Steiner tree, with binary characters, such that no transition between the two states of a character occurs twice in the tree (this assumes $m \geq n - 1$). The *chromosone inversion* tree is a Steiner tree with ternary characters with biologically motivated restrictions on the tree edges. All the problems have been rigorously defined and their decision problems shown to be NP-complete [69, 72, 73, 106, 120]. Perhaps the most interesting algorithmic work has been on the Wagner problem for fixed topologies [95, 233]. Heuristics have been published [35, 95].

The second major area starts with genetic or molecular strings for each of the species. There are three subcases: the Hamming metric [99, 101, 196], the Euclidean metric [36], and the minimum mutation metric [37, 232].

The Hamming metric assumes that the characters (or nucleotides or alleles or codons) of two sequences can be compared position by position. Hence, formally, the distance between two strings of m characters is the number of positions where characters mismatch. In a *binary* problem, each position can be one of two characters. (Note that the binary Hamming metric problem is equivalent to the binary Wagner tree problem, which remains NP-complete [106].) Heuristics [103] and branch-and-bound algorithms [129, 203] have been proposed. Optimal algorithms for a given topology are known [99, 100, 128, 230, 233]. Note that when the topology is fixed each dimension can be solved independently.

The Euclidean metric approach also assumes that sequences can be compared position by position. Cavalli-Sforza and Edwards [36] map each original sequence into a point in m-dimensional Euclidean space and give local Steinerization heuristics. Thompson [265] continued this work and Rogers [222] gave related results.

The minimal mutation approach assumes, more realistically, that mutations do not just change the value at a position, but that positions are deleted and added. This makes "alignment" difficult and sophisticated definitions of the mutation distance (or "edit distance") between strings are needed. In fact, so much evolutionary change has occurred that it is hard to find sufficiently long subsequences of genetic material to allow comparisons. The work in this area is based on Sankoff's algorithm [229] for fixed topologies. This is remarkable due

to the complexity of the metric. It is efficient for long strings but is exponential in n. Various costs can be assigned to the different operations (mutate, delete, insert) depending on their naturally occurring frequencies [231]. Of course, by varying these costs, different optimal trees are discovered.

Several algorithms have been used. For a fixed topology, there is Sankoff's algorithm [229, 230, 233], as well as an iterative improvement approach [231]. For large n, Sankoff's algorithm must be used with iterative improvement. Two approaches have been used to generate topologies. One approach uses an algorithm that does iterative improvement by subtree swapping [37]. Another approach is to generate all topologies satisfying restrictions, usually biological constraints. (To bolster other independent data for imposing restrictions, a biological study [122] used smaller unrestricted runs on troublesome sets of species.)

The third major area in the study of phylogenetic trees is the so-called *additive tree* approach. This is a Steiner tree problem only in that new vertices are introduced. A *dissimiliarity matrix* is given that provides lower bounds on the distances between pairs of species, using a common scale. The goal is to construct a tree, possibly with additional nodes, over the n species such that the path distances in the tree for every pair satisfy the given bounds. The tree may be arbitrary or it may be constrained to be a subgraph of a given weighted graph. For complexity results and additional references, see [69, 70]. A fast algorithm is known for the case when the bounds must be met exactly [66]. This is considered to be an inferior approach.

Other methods have been used. One method uses a variant of the GSMT using a *lower evaluation* metric, defined in terms of a finite poset [69]. Another, the *cladistic compatibility method,* is not essentially a Steiner problem [71]. Many others have been proposed but they lack sufficient mathematical rigor to be considered here.

REFERENCES

Many of the recent preprints below will appear in forthcoming special issues of *Algorithmica* and the *Annals of Operation Research.*

[1] P. K. Agarwal and M. T. Shing, Algorithms for the special cases of rectilinear Steiner trees: I. Points on the boundary of a rectilinear rectangle. *Networks* **20** (1990) 453–485.

[2] A. V. Aho, M. R. Garey, and F. K. Hwang, Rectilinear Steiner trees: Efficient special-case algorithms. *Networks* **7** (1977) 37–58.

[3] Y. P. Aneja, An integer linear programming approach to the Steiner problem in graphs. *Networks* **10** (1980) 167–178.

[4] S. Arnborg, J. Lagergren, and D. Seese, Problems easy for tree-decomposable graphs. *Automata Languages and Programming [LNCS 317]* (T. Lepisto and A. Salomaa, Eds.) Springer-Verlag, Berlin (1988) 38–51.

[5] D. Arpin, N. Maculan, and S. Nguyen, Le Probleme de Steiner sur un Graphe Oriente: Formulations and Relaxations. Publication 315. Centre de Recherche sur les Transports, University of Montreal (1983).

[6] S. L. Arlinghaus, *An Atlas of Steiner Networks*. Monograph 9. Institute of Math. Geography, Ann Arbor, MI (1989).
[7] G. Ausiello, A. D'Atri, and M. Moscarini, Chordality properties on graphs and minimal conceptual connections in semantic data models. *J. Comput. Syst. Sci.* **33** (1986) 179–202.
[8] A. Balakrishnan, Formulations and algorithms for the Steiner network problem. Unpublished manuscript, Sloan School Management, MIT (1982).
[9] A. Balakrishnan and N. R. Patel, Problem reduction methods and a tree generation algorithm for the Steiner network problem. *Networks* **17** (1987) 65–85.
[10] E. Balas, On the convex hull of the union of certain polyhedra. *Operations Res. Lett.* **7** (1988) 279–283.
[11] J. Beardwood, J. H. Halton, and J. M. Hammersley, The shortest path through many points. *Proc. Cambr. Philos. Soc.* **55** (1959) 299–327.
[12] J. E. Beasley, An algorithm for the Steiner problem in graphs. *Networks* **14** (1984) 147–159.
[13] J. E. Beasley, An SST-based algorithm for the Steiner problem on graphs. *Networks* **19** (1989) 1–16.
[14] J. E. Beasley, A heuristic for the Euclidean and rectilinear Steiner problems. Working Paper, Management School, Imperial College, London (1989).
[15] M. W. Bern, Network design problems: Steiner trees and spanning k-trees. PhD Dissertation, Computer Science Division, University of California at Berkeley (1987).
[16] M. W. Bern, Two probabilistic results on rectilinear Steiner trees. *Algorithmica* (1988) 191–204.
[17] M. W. Bern, Faster exact algorithms for Steiner trees in planar networks. *Networks* **20** (1990) 109–120.
[18] M. Bern and D. Bienstock, Further polynomially solvable special cases of the Steiner problem in planar graphs. Preprint (1989).
[19] M. W. Bern and M. de Carvalho, A greedy heuristic for the rectilinear Steiner tree problem. Technical Report, Computer Science Division, University of California at Berkeley (1985).
[20] M. W. Bern and R. L. Graham, The shortest-network problem. *Sci. Am.* **January** (1989) 84–89.
[21] M. W. Bern and P. Plassmann, The Steiner problem with edge lengths 1 and 2. *Info. Process. Lett.* **32** (1989) 171–176.
[22] K. Bharath-Kumar and J. M. Jaffe, Routing to multiple destinations in computer networks. *IEEE Trans. Commun.* **COM-31** (1983) 343–351.
[23] S. Bhaskaran, Optimal design of gas pipeline networks. PhD Dissertation, University of Adelaide, Australia (1978).
[24] S. Bhaskaran and F. J. M. Salzborn, Optimal design of gas pipeline networks. *J. Operational Res. Soc.* **30** (1979) 1047–1060.
[25] D. Bienstock and C. L. Monma, On the complexity of covering vertices by faces in a planar graph. *SIAM J. Comput.* **17** (1988) 53–76.
[26] D. Bienstock and C. L. Monma, Optimal enclosing regions in planar graphs. *Networks* **19** (1989) 79–84.
[27] D. Bienstock and C. L. Monma, On the complexity of embedding planar graphs to minimize certain distance measures. *Algorithmica,* to appear.
[28] H. L. Bodlaender, Dynamic programming on graphs with bounded treewidth. *Automata, Languages and Programming [LNCS 317]* (T. Lepisto and A. Salomaa, Ed.). Springer-Verlag (1988) 103–118.
[29] H. L. Bodlaender, NC-algorithms for graphs with small treewidth. *Graph-Theoretic Concepts in Computer Science [LNCS 344]*. (J. van Leeuwen, Ed.). Springer-Verlag, Berlin (1988) 1–10.
[30] R. S. Booth, Analytic formulas for full Steiner trees. *Disc. Comput. Geom.*, to appear.
[31] R. S. Booth, The Steiner ratio for five points. *Ann. Oper. Res.*, to appear.

[32] R. Borie, R. G. Parker, and C. A. Tovey, Recursively constructed graphs: decomposition and linear-time algorithm generation. Preprint (1989).
[33] W. M. Boyce and J. B. Seery, STEINER72: An improved version of the minimal network problem. Technical Report 35, Comp. Sci. Res. Ctr., Bell Labs (1975).
[34] W. M. Boyce, An improved program for the full Steiner tree problem. *ACM Trans. Math. Software* **3** (1977) 339–385.
[35] J. H. Camin and R. R. Sokal, A method for deducing branching sequences in phylogeny. *Evolution* **19** (1965) 311–326.
[36] L. L. Cavalli-Sforza and A. W. F. Edwards, Phylogenetic analysis: Models and estimation procedures. *Am. J. Hum. Genet.* **19** (1967) 233–257.
[37] R. J. Cedergren, D. Sankoff, B. LaRue, and H. Grosjean, The evolving tRNA molecule. *Crit. Rev. Biochem.* **11** (1981) 35–104.
[38] S.-K. Chang, The generation of minimal trees with a Steiner topology. *J. ACM* **19** (1972) 699–711.
[39] S.-K. Chang, The design of network configurations with linear or piecewise linear cost functions. *Symp. Comput. Commun., Networks, Teletraffic* (1972) 363–369.
[40] N.-P. Chen, New algorithm for Steiner tree on graphs. *IEEE Symp Circuits Syst.* (1983) 1217–1219.
[41] T.-Y. Cheung and J. Soukup, A note on minimum cost networks with nonlinear costs. *SIAM J. Appl. Math.* **34** (1978) 225–226.
[42] C. Chiang, M. Sarrafzadeh, and C. K. Wong, A powerful global router: Based on Steiner min-max trees. Preprint (1989).
[43] F. R. K. Chung, M. Gardner, and R. L. Graham, Steiner trees on a checkerboard. *Math. Mag.* **62** (1989) 83–96.
[44] F. R. K. Chung and E. N. Gilbert, Steiner trees for the regular simplex. *Bull. Inst. Math. Acad. Sinica* **4** (1976) 313–325.
[45] F. R. K. Chung and R. L. Graham, Steiner trees for ladders. *Ann. Disc. Math.* **2** (1978) 173–200.
[46] F. R. K. Chung and R. L. Graham, On Steiner trees for bounded point sets. *Geometriae Dedicata* **11** (1981) 353–361.
[47] F. R. K. Chung and R. L. Graham, A new bound for Euclidean Steiner minimal trees. *Ann. N.Y. Acad. Sci.* **440** (1985) 328–346.
[48] F. R. K. Chung, R. L. Graham, and M. E. Saks, Dynamic search in graphs. *Discrete Algorithms and Complexity* (D. S. Johnson et al., Eds.). Academic, New York (1987) 351–387.
[49] F. R. K. Chung and F. K. Hwang, A lower bound for the Steiner tree problem. *SIAM J. Appl. Math.* **34** (1978) 27–36.
[50] F. R. K. Chung and F. K. Hwang, The largest minimal rectilinear Steiner trees for a set of n points enclosed in a rectangle with given perimeter. *Networks* **9** (1979) 19–36.
[51] R. C. Clark, Communication networks, soap films, and vectors. *Phys. Ed.* **16** (1981) 32–37.
[52] A. Claus and N. Maculan, Une nouvelle formulation du probleme de Steiner sur un graphe. Publication 280, Centre de Recherche sur les Transports, University of Montreal (1983).
[53] E. J. Cockayne, On the Steiner problem. *Can. J. Math.* **10** (1967) 431–450.
[54] E. J. Cockayne, On the efficiency of the algorithm for Steiner minimal trees. *SIAM J. Appl. Math.* **18** (1970) 150–159.
[55] E. J. Cockayne and D. E. Hewgill, Exact computation of Steiner minimal trees in the plane. *Info. Proc. Lett.* **22** (1986) 151–156.
[56] E. J. Cockayne and D. E. Hewgill, Leaf paths of full Steiner trees and computation of planar Steiner minimal trees. *Algorithmica,* to appear.
[57] E. J. Cockayne and D. E. Hewgill, Improved computation of plane Steiner minimal trees. Preprint (1989).

[58] E. J. Cockayne and Z. A. Melzak, Steiner's problem for set-terminals. *Q. Appl. Math.* **26** (1967) 213–218.
[59] E. J. Cockayne and Z. A. Melzak, Euclidean constructability in graph-minimization problems. *Math. Mag.* **42** (1969) 206–208.
[60] E. J. Cockayne and D. B. Peters, MEST82, A program for minimal full Steiner trees in the plane. *Congressus Numeratium (Proc. 14th S. E. Conf. on Combinatorics, Graph Theory and Computing)* **39** (1983) 203–224.
[61] E. J. Cockayne and D. G. Schiller, Computation of a Steiner minimal tree. *Combinatorics* (D. J. A. Welsh and D. R. Woodall, Eds.). *Inst. Math. Appl.* (1972) 53–71.
[62] J. P. Cohoon, D. S. Richards, and J. S. Salowe, An optimal Steiner tree routing algorithm for a net whose terminals lie on the perimeter of a rectangle. *IEEE Trans. Comput.-Aided Design* **9** (1990) 398–407.
[63] C. J. Colbourn and L. K. Stewart, Permutation graphs: Connected domination and Steiner trees. *Discrete Math.* **86** (1990) 179–189.
[64] G. Cornuejols, J. Fonlupt, and D. Naddef, The graphical travelling salesman problem and some related integer polyhedra. Research Report 378, Appliquees de Grenoble (1983).
[65] R. Courant and H. Robbins, *What Is Mathematics?* Oxford University Press, New York (1941).
[66] J. C. Culberson and P. Rudnicki, A fast algorithm for constructing trees from distance matrices. *Info. Proc. Lett.* **30** (1989) 215–220.
[67] A. D'Atri and M. Moscarini, Distance-hereditary graphs, Steiner trees, and connected domination. *SIAM J. Comput.* **17** (1988) 521–538.
[68] A. D'Atri, M. Moscarini, and A. Sassano, The Steiner tree problem and homogeneous sets. *Math. Foundations Comput. Sci. [LNCS 324]* (1988) 249–261.
[69] W. H. E. Day, Computationally difficult parsimony problems in phylogenetic systematics. *J. Theor. Biol.* **103** (1983) 429–438.
[70] W. H. E. Day, Computational complexity of inferring phylogenies from dissimilarity matrices. *Bull. Math. Biol.* **49** (1987) 461–467.
[71] W. H. E. Day and D. Sankoff, Computational complexity of inferring phylogenies by compatibility. *Syst. Zool.* **35** (1986) 224–229.
[72] W. H. E. Day, D. S. Johnson, and D. Sankoff, The computational complexity of inferring rooted phylogenies. *Math. Biosci.* **81** (1986) 33–42.
[73] W. H. E. Day and D. Sankoff, Computational complexity of inferring phylogenies from chromosone inversion data. *J. Theor. Biol.* **124** (1987) 213–218.
[74] S. E. Dreyfus and R. A. Wagner, The Steiner problem in graphs. *Networks* **1** (1972) 195–207.
[75] J. Dolan, R. Weiss, and J. M. Smith, Minimal length trees on the unit sphere. Preprint (1989).
[76] M. Dror, B. Gavish, and J. Choquette, Directed Steiner tree problem on a graph: Models, relaxations and algorithms. *INFOR* **28** (1990) 266–281.
[77] D. Z. Du, On Steiner ratio conjectures. *Ann. Oper. Res.*, to appear.
[78] D. Z. Du and F. K. Hwang, A new bound for the Steiner ratio. *Trans. Am. Math. Soc.* **278** (1983) 137–148.
[79] D. Z. Du and F. K. Hwang, On a conjecture of Trietsch and Handler on the flow-dependent Steiner ratio. *Networks* **16** (1986) 47–50.
[80] D. Z. Du and F. K. Hwang, Steiner minimal trees for bar waves. *Acta Math. Appl. Sinica* **3** (1987) 246–256.
[81] D. Z. Du, F. K. Hwang, and S. C. Chao, Steiner minimal tree for points on a circle. *Proc. Am. Math. Soc.* **95** (1985) 613–618.
[82] D. Z. Du, F. K. Hwang, G. D. Song, and G. Y. Ting, Steiner minimal trees on sets of four points. *Disc. Comput. Geom.* **2** (1987) 401–414.
[83] D. Z. Du, F. K. Hwang, and J. F. Weng, Steiner minimal trees on zig-zag lines. *Trans. Am. Math. Soc.* **278** (1983) 149–156.

[84] D. Z. Du, F. K. Hwang, and J. F. Weng, Steiner minimal trees for regular polygons. *Disc. Comput. Geom.* **2** (1987) 65–87.
[85] D. Z. Du, F. K. Hwang, and E. N. Yao, The Steiner ratio conjecture is true for five points. *J. Combinatorial Theory Ser. A* **38** (1985) 230–240.
[86] D. Z. Du, E. Y. Yao, and F. K. Hwang, A short proof of a result of Pollak on Steiner minimal trees. *J. Combinatorial Theory Ser. A* **32** (1982) 396–400.
[87] C. W. Duin and A. Volgenant, Some generalizations of the Steiner problem in graphs. *Networks* **17** (1987) 353–364.
[88] C. W. Duin and A. Volgenant, An edge elimination test for the Steiner problem in graphs. *Operations Res. Lett.* **8** (1989) 79–83.
[89] C. W. Duin and A. Volgenant, Reducing the hierarchical network design problem. *Eur. J. Operational Res.* **39** (1989) 332–344.
[90] C. W. Duin and A. Volgenant, Reduction tests for the Steiner problem in graphs. *Networks* **19** (1989) 549–567.
[91] C. W. Duin and A. Volgenant, The multiweighted Steiner tree problem. *Ann. Oper. Res.*, to appear.
[92] C. El-Arbi, Une heuristique pour le probleme de l'Arbre de Steiner (in French). *RAIRO (Operations Res.)* **12** (1978) 207–212.
[93] R. E. Erickson, C. L. Monma, and A. F. Veinott, Send-and-split method for minimum-concave-cost network flows. *Math. Operations Res.* **12** (1987) 634–664.
[94] A. M. Farley, S. T. Hedetniemi, and S. L. Mitchell, Rectilinear Steiner trees in rectangle trees. *SIAM J. Alg. Disc. Methods* **1** (1980) 70–81.
[95] J. S. Farris, Methods for computing Wagner trees. *Syst. Zool.* **19** (1970) 83–92.
[96] J. Felsenstein, Numerical methods for inferring evolutionary trees. *Q. Rev. Biol.* **57** (1982) 379–404.
[97] J. Felsenstein, Phylogenies from molecular sequences: Inference and reliability. *Annu. Rev. Genet.* **22** (1988) 521–565.
[98] L. Few, The shortest path and shortest road through n points. *Mathematika* **2** (1955) 141–144.
[99] W. M. Fitch, Towards defining a course of evolution: Minimum change for a specific tree topology. *Syst. Zool.* **20** (1971) 406–416.
[100] W. M. Fitch, On the problem of discovering the most parsimonious tree. *Am. Naturalist* **111** (1977) 223–257.
[101] L. R. Foulds, Testing the theory of evolution. *Optima* **10** (1983) 1–3.
[102] L. R. Foulds, Maximum savings in the Steiner problem in phylogeny. *J. Theor. Biol.* **107** (1984) 471–474.
[103] L. R. Foulds, Testing the theory of evolution: A novel application of combinatorial optimization. *Discrete Appl. Math.* **15** (1986) 271–282.
[104] L. R. Foulds and P. B. Gibbons, A branch and bound approach to the Steiner problem in graphs. *Proceedings of the 14th Annual Conference of the Operational Research Society of New Zealand* (1978), Vol. 1, 61–70.
[105] L. R. Foulds, P. B. Gibbons, and M. L. Shore, Algorithms for the Steiner problem in graphs. *J. Combin. Inf. Syst. Sci.* **6** (1981) 215–219.
[106] L. R. Foulds and R. L. Graham, The Steiner problem in phylogeny is NP-complete. *Adv. Appl. Math.* **3** (1982) 43–49.
[107] L. R. Foulds and V. J. Rayward-Smith, Steiner problems in graphs: Algorithms and applications. *Eng. Optimization* **7** (1983) 7–16.
[108] J. Friedel and P. Widmayer, A simple proof of the Steiner ratio conjecture for five points. *SIAM J Appl. Math.* **49** (1989) 960–967.
[109] Y. Fu, Application of linear graph theory to printed circuits. *Proc. Asilomar Conf. Syst. Circuits* (1967) 721–728.
[110] R. L. Gallawa, Application of the Steiner street problem to a more economical data bus. *Fiber Integrated Opt.* **1** (1978) 289–311.
[111] M. Gardner, Mathematical games. *Sci. Am.* **June** (1986) 16–22.

[112] M. R. Garey, L. R. Graham, and D. S. Johnson, The complexity of computing Steiner minimal trees. *SIAM J. Appl. Math.* **32** (1977) 835–859.
[113] M. R. Garey and D. S. Johnson, The rectilinear Steiner tree problem is NP-complete. *SIAM J. Appl. Math.* **32** (1977) 826–834.
[114] M. R. Garey and D. S. Johnson, *Computers and Intractability*. W. H. Freeman, San Francisco (1979).
[115] G. Georgakopoulos and C. H. Papadimitriou, A 1-Steiner tree problem. *J Algorithms* **8** (1987) 122–130.
[116] E. N. Gilbert, Random minimal trees. *J. SIAM* **13** (1965) 376–387.
[117] E. N. Gilbert, A minimum cost communication networks. *Bell Syst. Tech. J.* **46** (1967) 2209–2227.
[118] E. N. Gilbert and H. O. Pollak, Steiner minimal trees. *SIAM J. Appl. Math.* **16** (1968) 1–29.
[119] M. X. Goemans and D. J. Bertsimas, On the parsimonious property of connectivity problems. Working Paper, Sloan School, MIT (1989).
[120] R. L. Graham and L. R. Foulds, Unlikelihood that minimal phylogenies for a realistic biological study can be constructed in reasonable computation time. *Math. Biosci.* **60** (1982) 133–142.
[121] R. L. Graham and F. K. Hwang, Remarks on Steiner minimal trees I. *Bull. Inst. Math. Acad. Sinica* **4** (1976) 177–182.
[122] M. W. Gray, D. Sankoff, and R. J. Cedergren, On the evolutionary descent of organisms and organelles: A global phylogeny based on a highly conserved structural core in small subunit ribosomal RNA. *Nucleic Acids Res.* **12** (1984) 5837–5852.
[123] S. L. Hakimi, Steiner's problem in graphs and its implications. *Networks* **1** (1971) 113–133.
[124] S. Hambrusch and L. TeWinkel, Parallel heuristics for the Steiner tree problem in images without sorting or routing. Technical Report 89-048, Computer Science Dept., Purdue University (1989).
[125] M. Hanan, Net wiring for large scale integrated circuits. RC 1375. IBM Research Report (1965).
[126] M. Hanan, On Steiner's problem with rectilinear distance. *J. SIAM Appl. Math.* **14** (1966) 255–265.
[127] M. Hanan, A counterexample to a theorem of Fu on Steiner's problem. *IEEE Trans. Circuit Theory* **CT-19** (1972) 74.
[128] J. A. Hartigan, Minimum mutations fits to a given tree. *Biometrics* **29** (1973) 53–65.
[129] M. D. Hendy and D. Penny, Branch and bound algorithms to determine minimal evolutionary trees. *Math. Biosci.* **59** (1982) 277–290.
[130] J. A. Hendrickson, Clustering in numerical cladistics: A minimum-length directed tree problem. *Math. Biosci.* **3** (1968) 371–381.
[131] J. Hesser, R. Manner, and O. Stucky, Optimization of Steiner trees using genetic algorithms. *Proceedings of the Third International Conference on Genetic Algorithms* (1989) 231–236.
[132] J.-M. Ho, G. Vijayan, and C. K. Wong, A new approach to the rectilinear Steiner tree problem, *Proceedings of the 26th Design Automation Conference* (1989) 161–166.
[133] J.-M. Ho, G. Vijayan, and C. K. Wong, Constructing the optimal rectilinear Steiner tree derived from a minimum spanning tree. *Digest of Tech. Papers* ICCAD-89 (1989) 6–9.
[134] A. D. Howard, Theoretical model to optimal drainage. To appear.
[135] Y. C. Hsu, Y. Pan, and W. J. Kubitz, A path selection global router. *24th ACM/IEEE Design Automation Conference* (1987) 641–644.
[136] F. K. Hwang, On Steiner minimal trees with rectilinear distance. *SIAM J. Appl. Math.* **30** (1976) 104–114.

[137] F. K. Hwang, The rectilinear Steiner problem. *Design Automation Fault Tolerant Comput.* **3** (1978) 303–310.
[138] F. K. Hwang, An O(n log n) algorithm for suboptimal rectilinear Steiner trees. *IEEE Trans. Circuits Syst.* **CAS-26** (1979) 75–77.
[139] F. K. Hwang, An O(n log n) algorithm for rectilinear minimal spanning trees. *J. ACM* **26** (1979) 177–182.
[140] F. K. Hwang, A linear time algorithm for full Steiner trees. *Operations Res. Lett.* **5** (1986) 235–237.
[141] F. K. Hwang, Steiner minimal trees on the Chinese checkerboard. *Math. Mag.*, to appear.
[142] F. K. Hwang, G. D. Song, G. Y. Ting, and D. Z. Du, A decomposition theorem on Euclidean Steiner minimal trees. *Disc. Comput. Geom.* **3** (1988) 367–382.
[143] F. K. Hwang and J. F. Weng, Hexagonal coordinate systems and Steiner minimal trees. *Disc. Math.* **62** (1986) 49–57.
[144] F. K. Hwang and J. F. Weng, Shortest networks with a given topology. Preprint.
[145] F. K. Hwang, J. F. Weng, and D. Z. Du, A class of full Steiner minimal trees. *Disc. Math.* **45** (1983) 107–112.
[146] F. K. Hwang and Y. C. Yao, Comments on Bern's probabilistic results on rectilinear Steiner trees. *Algorithmica* **5** (1990) 591–598.
[147] A. Iwainsky, Some notes on the Steiner tree problem in graphs. *Optimization of Connection Structures in Graphs* (A. Iwainsky, Ed.). Centr. Inst. of Cyb. a. Inform. Processes, Berlin (1985) 57–73.
[148] A. Iwainsky, E. Canuto, O. Taraszow, and A. Villa, Network decomposition for the optimization of connection structures. *Networks* **16** (1986) 205–235.
[149] A. Iwainsky and A. Nuemann, Some variants of a heuristic approach to the solution of the Steiner tree problem in graphs, to appear.
[150] A. Jain, Probabilistic analysis of LP relaxation bounds for the Steiner problem in networks. *Networks* **19** (1989) 793–801.
[151] A. Jain, The Steiner problem on directed graphs: A probabilistic analysis of the duality gap. Preprint (1989).
[152] V. Jarnik and O. Kössler, O minimálnich gratech obsahujicich n daných bodu [in Czech]. *Casopis Pesk. Mat. Fyr.* **63** (1934) 223–235.
[153] J. W. Jiang and P. S. Tang, An algorithm for generating rectilinear Steiner trees [in Chinese]. *J. Fudan Univ. (Natural Sci.)* **25** (1986) 343–349.
[154] D. S. Johnson, The NP-completeness column: An ongoing guide. *J. Algorithms* **6** (1985) 434–451.
[155] D. S. Johnson, The NP-completeness column: An ongoing guide. *J. Algorithms* **8** (1987) 438–448.
[156] B. Johnston, An empirical study of exact algorithms for the cardinality Steiner problem. Master's Thesis, Simon Fraser University, Burnaby, B.C. Canada (1987).
[157] R. Joobbani, *An Artificial Intelligence Approach to VLSI Routing.* Kluwer Academic, Boston (1986).
[158] R. R. Kallman, A conjecture of Gilbert and Pollak on minimal trees. *Stud. Appl. Math.* **22** (1973) 141–151.
[159] R. M. Karp, Reducibility among combinatorial problems. *Complexity of Computer Computations.* (R. E. Miller and J. W. Thatcher, Ed.). Plenum Press, New York (1972) 85–103.
[160] R. M. Karp, The probabilistic analysis of some combinatorial search algorithms. *Algorithms and Complexity: New Directions and Recent Results.* (J. Traub, Ed.). Academic Press, New York (1976) 1–19.
[161] R. M. Karp, Probabilistic analysis of partitioning algorithms for the traveling-salesman problem in the plane. *Math. Operations Res.* **2** (1977) 209–224.
[162] I. N. Katz and L. Cooper, Facility location in the presence of forbidden regions. I. *Eur. J. Operational Res.* **6** (1981) 166–173.

[163] J. Komlos and M. T. Shing, Probabilistic partitioning algorithms for the rectilinear Steiner problem. *Networks* **15** (1985) 413–423.
[164] P. Korhonen, An algorithm for transforming a spanning tree into a Steiner tree. *Survey of Math Programming* (*Proc 9th Int. Math. Prog. Symp.*), Vol. 2, North-Holland, Amsterdam (1979) 349–357.
[165] A. Kotzig, Optimálne spojovacie systemy (in Czech). *Mathematické Metódy v. Hospodársky Praxi* (V. Kapitoly, Ed.). Vydavatelstvo Slovenskej Akadémie Vied, Bratislava (1961).
[166] L. T. Kou, On efficient implementation of an approximation algorithm for the Steiner tree problem. *Acta Info.* **27** (1990) 369–380.
[167] L. T. Kou and K. Makki, An even faster approximation algorithm for the Steiner tree problem in graphs. *Congressus Numerantium* **59** (1987) 147–154.
[168] L. Kou, G. Markowsky, and L. Berman, A fast algorithm for Steiner trees. *Acta Info.* **15** (1981) 141–145.
[169] L. Kucera, A. Marchetti-Spaccamela, M. Protasi, and M. Talamo, Near optimal algorithms for finding minimum Steiner trees on random graphs. *Mathematical Foundations of Computer Science 1986* [*LNCS 233*] (J. Gruska et al., Eds.). Springer-Verlag, Berlin (1986) 501–511.
[170] H. W. Kuhn, Steiner's problem revisited. *Studies in Optimization,* Vol. 10 (G. B. Dantzig and B. C. Eaves, Eds.). Math Association of America, Wash. D.C. (1975) 98–107.
[171] E. L. Lawler, *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart, and Winston, New York (1976).
[172] D. H. Lee, Low cost drainage networks. *Networks* **6** (1976) 351–371.
[173] D. T. Lee and C. K. Wong, Voronoi diagrams in L_1 (L_∞) metrics with 2-dimensional storage applications. *SIAM J. Comput.* **9** (1980) 200–211.
[174] D. Lee, Some industrial case studies of Steiner trees. Preprint (1989).
[175] J. H. Lee, N. K. Bose, and F. K. Hwang, Use of Steiner's problem in suboptimal routing in rectilinear metric. *IEEE Trans. Circuits Syst.* **CAS-23** (1976) 470–476.
[176] K.-W. Lee and C. Sechen, A new global router for row-based layout. *IEEE Int. Conf. Computer-Aided Design* (1988) 180–183.
[177] A. Ju. Levin, The shortest connection of a group of graph vertices [in Russian]. *Proc. of the Inst. of Math., Voronez State University* (1970) 6–17.
[178] A. Ju. Levin, Algorithm for the shortest connection of a group of graph vertices. *Soviet Math. Doklady* **12** (1971) 1477–1481.
[179] A. Liestman, Construction of Steiner trees with obstacles in the plane. Masters Thesis, Dept. Computer Science, University of Illinois (1978).
[180] H. P. L. Luna, N. Ziviani, and R. H. B. Cabral, The telephone switching centre network problem: Formulation and computational experience. *Discrete Appl. Math.* **18** (1987) 199–210.
[181] M. Lundy, Applications of the annealing algorithm to combinatorial problems in statics. *Biometrika* **72** (1985) 191–198.
[182] M. Lundy and A. Mees, Convergence of an annealing algorithm. *Math. Prog.* **34** (1986) 111–124.
[183] S. Ma, The Steiner tree problem on graph and its heuristic algorithm (in Chinese). *Chin. J. Comput.* **8** (1985) 237–239.
[184] G. P. McKeown, V. J. Rayward-Smith, and H. J. Turpin, Branch-and-bound as a higher-order function. Preprint (1989).
[185] N. Maculan, O problema de Steiner em grafos orientados. *II Congresso Latino-americano de Pesquina Operacional e Engenharia de Sistemas*, Buenos Aires (1984) 206–213.
[186] N. Maculan, The Steiner problem in graphs. *Ann. Discrete Math.* **31** (1987) 185–222.
[187] N. Maculan and A. Candia-Vejar, An approach for the Steiner problem in directed graphs. Preprint (1989).

[188] R. R. L. Matos, Rectilinear arborescence and rectilinear Steiner tree problems. PhD Dissertation, University of Birmingham (1980).
[189] N. Megiddo, Cost allocation for Steiner trees. *Networks* **8** (1978) 1–6.
[190] K. Mehlhorn, A faster approximation algorithm for the Steiner problem in graphs. *Info. Process. Lett.* **27** (1988) 125–128.
[191] Z. A. Melzak, On the problem of Steiner. *Can. Math. Bull.* **4** (1961) 143–148.
[192] Z. A. Melzak, *Companion to Concrete Mathematics*. Wiley, New York (1973).
[193] Z. A. Melzak and J. S. Forsyth, Polyconics 1: Polyellipses and optimization. *Q. J. Math.* (1977) 239–255.
[194] W. Miehle, Link-length minimization in networks. *Oper. Res.* **6** (1958) 232–243.
[195] M. Minoux, Efficient approximate solution of Steiner tree problems using reoptimization, supermodularity, and the accelerated greedy algorithm. *INFOR* **28** (1990) 221–233.
[196] G. W. Moore, J. Barnabas, and M. Goodman, A method for constructing maximum parsimony ancestral amino acid sequences on a given network. *J. Theor. Biol.* **38** (1973) 459–485.
[197] S. Moran, On the length of optimal TSP circuits in sets of bounded diameter. *J. Combinatorial Theory Ser. B*, **37** (1984) 113–114.
[198] L. Nastansky, S. M. Selkow, and N. F. Stewart, Cost-minimal trees in directed acyclic graphs. *Z. Operations Res.* **18** (1974) 59–67.
[199] A. P.-C. Ng, P. Raghavan, and C. D. Thompson, A language for describing rectilinear Steiner tree configurations. *23rd ACM/IEEE Design Automation Conference* (1986) 659–662.
[200] K. Ollerenshaw, Minimum networks linking four points in a plane. *Inst. Math. Appl.* **15** (1978) 208–211.
[201] F. P. Palermo, A network minimization problem. *IBM J. Res. Dev.* **5** (1961) 335–337.
[202] O. I. Palma-Pacheco, Contribuicao para a Resolucao do Problema de Steiner num Grafo Direcionado: un Metodo Heuristico. PhD Dissertation, Systems Engineering Computer Science, COPPE, Federal University of Rio de Janeiro (1985).
[203] D. Penny and M. D. Hendy, TurboTree: A fast algorithm for minimal trees. *CABIOS* **3** (1987) 183–187.
[204] J. Plesnik, A bound for the Steiner tree problem in graphs. *Math. Slovaca* **31** (1981) 155–163.
[205] J. Plesnik, On heuristics for the Steiner problem in graphs. Preprint (1989).
[206] H. O. Pollak, Some remarks on the Steiner problem. *J. Combinatorial Theory, Ser. A* **24** (1978) 278–295.
[207] G. Polya, *Induction and Analogy in Mathematics*. Princeton University Press, Princeton, NJ (1954).
[208] A. Prodon, T. M. Liebling, and H. Groflin, Steiner's problem on two-trees. Technical Report RO850315, Ecole Polytechnique Federale de Lausanne (1985).
[209] J. S. Provan, A polynomial algorithm for the Steiner tree problem on terminal-planar graphs. Technical Report UNC/ORSA/Tech. Rep.-83/10, Dept. Operations Research, University of North Carolina, Chapel Hill (1983).
[210] J. S. Provan, Convexity and the Steiner tree problem. *Networks* **18** (1988) 55–72.
[211] J. S. Provan, An approximation scheme for finding Steiner trees with obstacles. *SIAM J Comput.* **17** (1988) 920–934.
[212] J. S. Provan, The role of Steiner hulls in the solution to Steiner tree problems. *Ann. Oper. Res.*, to appear.
[213] J. S. Provan and M. O. Ball, Computing network reliability. *Operations Res.* **32** (1984) 516–526.
[214] W. R. Pulleybank and A. E. Steger, A min-max equation for Steiner trees on rectangular grids. Preprint (1989).
[215] S. K. Rao, P. Sadayappan, F. K. Hwang, and P. W. Shor, The rectilinear Steiner arborescence problem. *Algorithmica,* to appear.

[216] R. L. Rardin, R. G. Parker, and M. B. Richey, A polynomial algorithm for a class of Steiner tree problems on graphs. Indus. and Syst. Engr. Rep. Series J-82-5, Georgia Institute of Technology (1982).
[217] V. J. Rayward-Smith, The computation of nearly minimal Steiner trees in graphs. *Int. J. Math. Ed. Sci. Technol.* **14** (1983) 15–23.
[218] V. J. Rayward-Smith and A. Clare, On finding Steiner vertices. *Networks* **16** (1986) 283–294.
[219] D. Richards, Fast heuristic algorithms for rectilinear Steiner trees. *Algorithmica* **4** (1989) 191–207.
[220] D. Richards and J. S. Salowe, A simple proof of Hwang's theorem for rectilinear Steiner minimal trees, *Ann. Oper. Res.*, to appear.
[221] D. Richards and J. S. Salowe, A linear-time algorithm to construct a rectilinear Steiner minimal tree for k-extremal point sets. *Algorithmica*, to appear.
[222] J. S. Rogers, Deriving phylogenetic trees from allele frequencies. *Syst. Zool.* **33** (1984) 52–63.
[223] F. J. Rohlf, A note on minimum length trees. *Syst. Zool.* **33** (1984) 341–343.
[224] J. H. Rubenstein and D. A. Thomas, The calculus of variations and the Steiner problem. *Ann. Oper. Res.*, to appear.
[225] J. H. Rubenstein and D. A. Thomas, A variational approach to the Steiner ratio conjecture for six points. *J. Combinatorial Theory Ser. A.*
[226] J. H. Rubenstein and D. A. Thomas, Critical points for the Steiner ratio conjecture. Preprint (1989).
[227] M. H. Rudowski, Hypothesis on the length of minimal Steiner tree in multidimensional spaces with a rectilinear metric (in Polish). *Arch. Autom. Telemech.* **29** (1984) 353–358.
[228] T. M. Sandifer, Algorithms for solving the directed minimal Steiner tree problem. PhD Dissertation, University of Massachusetts, Amherst (1981).
[229] D. Sankoff, Minimal mutation trees of sequences. *SIAM J. Appl. Math.* **28** (1975) 35–42.
[230] D. Sankoff and R. J. Cedergren, Simultaneous comparison of three or more sequences related by a tree. *Time Warps, String Edits and Macromolecules: The Theory and Practice of Sequence Comparison.* (D. Sankoff and J. B. Kruskal, Eds.). Addison-Wesley, Reading, MA (1983) 253–263.
[231] D. Sankoff, R. J. Cedergren, and G. Lapalme, Frequency of insertion-deletion, transversion, and transition in the evolution of 5S ribosomal RNA. *J. Mol. Evol.* **7** (1976) 133–149.
[232] D. Sankoff, R. J. Cedergren, and W. McKay, A strategy for sequence phylogeny research. *Nucleic Acids Res.* **10** (1982) 421–431.
[233] D. Sankoff and P. Rousseau, Locating the vertices of a Steiner tree in an arbitrary metric space. *Math. Program.* **9** (1975) 240–246.
[234] M. Sarrafzadeh and C. K. Wong, Geometric Steiner min-max trees. Preprint (1989).
[235] A. Segev, The node-weighted Steiner tree problem. *Networks* **17** (1987) 1–17.
[236] M. Servit, Heuristic algorithms for rectilinear Steiner trees. *Digital Processes* **7** (1981) 21–32.
[237] M. I. Shamos and D. Hoey, Closest-point problems. *16th Annual Symp. on Foundations of Computer Science* (1975) 151–162.
[238] M. L. Shore, L. R. Foulds, and P. B. Gibbons, An algorithm for the Steiner problem in graphs. *Networks* **12** (1982) 323–333.
[239] J. M. Smith, Generalized Steiner network problems. PhD Dissertation, University of Illinois, Urbana (1978).
[240] J. M. Smith, Steiner minimal trees with obstacles. Technical Report, Dept. Indust. Engr. and Oper. Res., University of Massachusetts (1982).
[241] J. M. Smith, Generalized Steiner network problems in engineering design. *Engineering Design* (J. F. Gero, Ed.). Academic Press, New York (1985) 119–161.

[242] J. M. Smith and M. Gross, Steiner minimal trees and urban service networks. *J. Socio. Econ. Planning* **16** (1982) 21–38.
[243] J. M. Smith, D. T. Lee, and J. S. Liebman, An O(N log N) heuristic algorithm for the rectilinear Steiner minimal tree problem. *Eng. Optimization* **4** (1980) 179–192.
[244] J. M. Smith, D. T. Lee, and J. S. Liebman, An O(N log N) heuristic for Steiner minimal tree problems on the Euclidean metric. *Networks* **11** (1981) 23–29.
[245] J. M. Smith and J. S. Liebman, Steiner trees, Steiner circuits and the interference problem in building design. *Eng. Optimization* **4** (1979) 15–36.
[246] J. M. Smith and J. S. Liebman, An $O(n^2)$ heuristic algorithm for the directed Steiner minimal tree problem. *Appl. Math. Modelling* **4** (1980) 369–375.
[247] J. M. Smith and P. Winter, Steiner minimal trees for three points and one convex polygonal obstacle. *Ann. Oper. Res.*, to appear.
[248] W. D. Smith, Studies in computational geometry motivated by mesh generation. PhD dissertation, Princeton University (1988).
[249] W. D. Smith, How to find Steiner minimal trees in Euclidean d-Space. *Algorithmica*, to appear.
[250] T. Snyder, On minimal rectilinear Steiner trees in all dimensions. *Proc. 6th ACM Symp. Comput. Geom.* (1990) 311–320.
[251] J. Soukup, On minimum cost networks with non-linear costs. *SIAM J. Appl. Math.* **29** (1975) 571–581.
[252] J. Soukup, Minimum Steiner trees, roots of a polynomial and other magic. *ACM/SIGMAP Newsletter* **22** (1977) 37–51.
[253] J. Soukup and W. F. Chow, Set of test problems of the minimum length connection networks. *SIGMAP Newsletter* **15** (1973) 48–51.
[254] R. Spira, Am. math. mon. problem. Preprint.
[255] J. M. Steele, Subadditive Euclidean functionals and nonlinear growth in geometric probability. *Ann. Probability* **9** (1982) 365–376.
[256] H. Steinhaus, *Mathematical Snapshots*. Oxford University Press, New York (1960).
[257] G. F. Sullivan, Approximation algorithms for Steiner tree problems. Technical Report 249, Dept. Computer Science, Yale University (1982).
[258] J. W. Suurballe, Algorithms for minimal trees and semi-Steiner trees based on the simplex method (Abstract). *Proc. Princeton Symp on Math Program.* (1970) 614–615.
[259] A. Suzuki and M. Iri, A heuristic method for Euclidean Steiner problem as a geographical optimization problem. *Asia-Pacific J. Operations Res.* **3** (1986) 106–122.
[260] H. Takahashi and A. Matsuyama, An approximate solution for the Steiner problem in graphs. *Math. Jpn.* **24** (1980) 573–577.
[261] K. Takamizawa, T. Nishizeki, and N. Saito, Linear-time computability of combinatorial problems on series-parallel graphs. *J. ACM* **29** (1982) 623–641.
[262] O. Taraszow and A. Iwainsky, Trassierung und Plazierung auf Graphenmodellen. *Vortrags-Sammelband der 4. Fachtagung Numerische Realisierung Math. Modelle* (1981) 215–231.
[263] O. Taraszow and P. Richter, New heuristic algorithms for solving the Steiner tree problem in graphs. To appear.
[264] C. D. Thomborson (a.k.a. Thompson), L. L. Deneen, and G. M. Shute, Computing a rectilinear Steiner minimal tree in $n^{O(\sqrt{n})}$ time. *Parallel Algorithms and Architectures [LNCS 269]* (Albrecht et al., Eds.). Akademie-Verlag, Berlin, 1987, 176–183.
[265] E. A. Thompson, The method of minimum evolution. *Ann. Hum. Genet. Lond.* **36** (1973) 333–340.
[266] M. Tideman, Comment on "a network minimization problem." *IBM J. Res. Dev.* **6** (1962) 259.

[267] D. Trietsch, Augmenting Euclidean networks—The Steiner case. *SIAM J. Appl. Math.* **45** (1985) 855–860.
[268] D. Trietsch and F. K. Hwang, An improved algorithm for Steiner trees. *SIAM J. Appl. Math.* **50** (1990) 244–263.
[269] D. Trietsch and G. Y. Handler, The Gilbert and Pollak conjecture—A generalization. *Networks* **15** (1983) 365–380.
[270] V. A. Trubin, Subclass of the Steiner problems on a plane with rectilinear metric. *Cybernetics* **21** (1985) 320–324.
[271] P. A. van de Heyden, Some experiments with Steiner trees. *Int. Ser. Numer. Math.* **36** (1977) 79–109.
[272] L. P. P. P. van Ginneken and R. H. J. M. Otten, Global wiring for custom layout design. *Proc. Intl. Symp. Circuits and Systems* (1985) 207–208.
[273] S. Voss, Degree-constrained Steiner problem. *Proc. EuroVIII* (1986).
[274] S. Voss, A survey of some generalizations of Steiner's problem in graphs. *Proc. First Balkan Conf. Operations Res.* (1988).
[275] S. Voss, Steiner's problem in directed graphs: Logical tests in a branch and bound algorithm. Preprint (1989).
[276] S. Voss, A reduction-based algorithm for the Steiner problem in graphs. *Methods Operations Res.* **58** (1989) 239–252.
[277] S. Voss and K. D. Mehr, On a generalized Steiner problem with 2-edge-connectivity. *Methods Operations Res.* **57** (1987) 161–172.
[278] J. A. Wald, A linear algorithm for a class of problems on partial 2-trees. Manuscript, Schlumberger Tech. Corp. (1987).
[279] J. A. Wald and C. J. Colbourn, Steiner trees in outerplanar graphs. *Proc. 13th S.E. Conf. Combinatorics, Graph Theory, and Computing* (1982) 15–22.
[280] J. A. Wald and C. J. Colbourn, Steiner trees, partial 2-trees, and minimal IFI networks. *Networks* **13** (1983) 159–167.
[281] J. A. Wald and C. J. Colbourn, Steiner trees in probabilistic networks. *Microelectron. Reliab.* **23** (1983) 837–840.
[282] J. A. Wald and P. G. Sorenson, Resolving the query inference problem using Steiner trees. *ACM Trans. Database Sys.* **9** (1984) 348–368.
[283] S.-M. Wang, A multiple source algorithm for suboptimum Steiner trees in graphs. *Proc WG'85* (1985) 387–396.
[284] T. Watanabe and Y. Sugiyama, A new routing algorithm and its hardware implementation. *Proc. 23rd ACM/IEEE Design Automation Conf.* (1986) 574–580.
[285] B. M. Waxman, Routing of multipoint connections. *IEEE J. Selected Areas Comm.* **6** (1988) 1617–1622.
[286] B. M. Waxman, Probable performance of Steiner tree algorithms. Technical report WUCS-88-4, Dept. Computer Science, Washington University, St. Louis (1988).
[287] B. M. Waxman and M. Imase, Worst-case performance of Rayward-Smith's Steiner tree heuristics. *Info. Proc. Lett.* **29** (1988) 283–287.
[288] J. F. Weng, Steiner minimal trees on vertices of regular polygons. *Acta Math. Appl. Sinica* **8** (1983) 129–141.
[289] J. F. Weng, Generalized Steiner problems and hexagonal coordinate system (in Chinese). *Acta Math. Appl. Sinica* **8** (1985) 383–397.
[290] C. Werner, Networks of minimal length. *Can. Geographer* **13** (1969) 47–69.
[291] K. White, M. Farber, and W. Pulleybank, Steiner trees, connected domination and strongly chordal graphs. *Networks* **15** (1985) 109–124.
[292] P. Widmayer, Fast approximation algorithms for Steiner's problem in graphs. Technical report, Inst. fur Angew. Info., University of Karlsruhe (1985).
[293] P. Widmayer, On approximation algorithms for Steiner's problem in graphs. *Graph-Theoretic Concepts in Computer Science [LNCS 246]* (G. Tinhofer and G. Schmidt, Eds.). Springer-Verlag, Berlin (1986) 17–28.

[294] P. Winter, The Steiner problem. M.Sc. Thesis, Inst. of Datalogy, University of Copenhagen, Denmark (1981).
[295] P. Winter, An algorithm for the Steiner problem in the Euclidean plane. *Networks* **15** (1985) 323–345.
[296] P. Winter, Generalized Steiner problem in outerplanar networks. *BIT* **25** (1985) 485–496.
[297] P. Winter, Generalized Steiner problem in series-parallel networks. *J. Algorithms* **7** (1986) 549–566.
[298] P. Winter, Steiner problem in networks: A survey. *Networks* **17** (1987) 129–167.
[299] P. Winter, Steiner problem in Halin networks. *Discrete Appl. Math.* **17** (1987) 281–294.
[300] P. Winter, Generalized Steiner problem in Halin networks. To appear.
[301] P. Winter, Construction of minimum distance rectangle trees. To appear.
[302] R. T. Wong, A dual ascent approach for Steiner tree problems on a directed graphs. *Math. Programming* **28** (1984) 271–287.
[303] Y. F. Wu, P. Widmayer, and C. K. Wong, A faster approximation algorithm for the Steiner problem in graphs. *Acta Info.* **23** (1986) 223–229.
[304] J. G. Xiong, Algorithms for global routing. *23rd ACM/IEEE Design Automation Conference* (1986) 824–830.
[305] Y. Y. Yang and O. Wing, An algorithm for the wiring problem. *Digest IEEE Int. Symp. Electrical Networks* (1971) 14–15.
[306] Y. Y. Yang and O. Wing, Supoptimal algorithm for a wire routing problem. *IEEE Trans. Circuit Theory* **CT-19** (1972) 508–510.
[307] Y. Y. Yang and O. Wing, Optimal and suboptimal solution algorithms for the wiring problem. *Proc. IEEE Int. Symp. Circuit Theory* (1972) 154–158.
[308] Y. Y. Yang and O. Wing, On a multinet wiring problem. *IEEE Trans. Circuit Theory* **CT-20** (1973) 250–252.
[309] W. C. Yu and S. B. Xu, The Steiner ratio conjecture for six points. *Tienjing Conf. Combinatorial Optimization* (1988).
[310] C. Werner, The role of topology and geometry in optimal network design. *Papers of Regional Sci. Assoc.* XXI (1968) 173–189.

Received February 1988
Accepted November 1989