Computation of Minimal Length Full Steiner Trees on the Vertices of a Convex Polygon

By E. J. Cockayne*

Abstract. A Steiner minimal tree is a tree of minimal length whose vertices are a given set of points $a_1, \cdots, a_n$ in E^2 and any set of additional points $s_1, \cdots, s_k$ ($k \geqq 0$). In general, the introduction of extra points makes possible shorter trees than the minimal length tree whose vertices are precisely $a_1, \cdots, a_n$ and for which practical algorithms are known. A Steiner minimal tree is the union of special subtrees, known as full Steiner trees. This paper demonstrates the use of the computer in generating minimal length full Steiner trees on sets of points in E^2 which are the vertices of convex polygons. The procedure given is a basis from which further research might proceed towards an ultimate practical algorithm for the construction of Steiner minimal trees.

1. Introduction. There are simple algorithms available for constructing the minimal length tree(s) whose vertices $a_1, \cdots, a_n$ are a given set of points in E^2 [7]. However, it is often possible to construct even shorter trees connecting $a_1, \cdots, a_n$ by introducing a number of extra vertices $s_1, \cdots, s_k$. Hence the following problem which is known as the Steiner problem: Given n points $a_1, \cdots, a_n$ in E^2, to construct the tree(s) with vertices $a_1, \cdots, a_n$ and any k additional vertices $s_1, \cdots, s_k$ ($k \geqq 0$) whose total length is minimum. Solutions are called Steiner minimal trees. The problem and its generalisations have received much attention in the literature recently due to the wide variety of applications. Principally Z. A. Melzak [5] provided an algorithm for solution and in fact proved that a finite number of classical ruler-compass constructions would solve the problem. E. J. Cockayne [1] developed this procedure and discussed certain generalisations. Melzak and Cockayne [2] have extended their algorithm to cover the case where the a_i are sets rather than points. The principal problem remaining at this time is that of increasing the efficiency of the algorithm so that one can implement it on a digital computer and construct Steiner minimal trees, say for $n = 20$ or 30. As it stands, the number of steps required renders the algorithm completely impractical, except for a very small number of points. A few suggestions on efficiency were given in [1]. E. N. Gilbert and H. O. Pollak [4] have also attacked this aspect of the problem, giving a variety of geometric criteria which might be useful in reducing the algorithm. Their paper also contains work on generalisations, an excellent review of known results and an extensive bibliography. In the present work, we have adopted some of their terminology. This paper, also, is concerned with implementing the algorithm and its efficiency. In order to be more specific, we shall need a little mathematical background.

If a_1, a_2, a_3 are points in E^2 such that no angle of the triangle is $\geqq 120°$, then the point S which minimises the sum of distances $a_1S + a_2S + a_3S$ is the unique point (called the Steiner point of the triangle) at which each side subtends 120°. From this

---

* University of Victoria, British Columbia, Canada and University of Auckland, New Zealand.

521

fact one deduces (see [3]) that Steiner minimal trees have the five properties (P1) − (P5) listed below, i.e. Steiner minimal trees are Steiner trees.

Definition. A tree U with vertices $a_1, \cdots, a_n, s_1, \cdots, s_k$ ($k \geqq 0$) is a Steiner tree on $a_1, \cdots, a_n$ if and only if it has the following properties:

(P1) U is non-self-intersecting.

(P2) $w(s_i) = 3, 1 \leqq i \leqq k$. ($w(x)$ is the valency of vertex x.)

(P3) $w(a_j) \leqq 3, 1 \leqq j \leqq n$.

(P4) Each s_i is the Steiner point of the triangle formed by the points which directly connect s_i in U.

(P5) $0 \leqq k \leqq n - 2$.

The original method of [5] consists of a proof that there are finitely many Steiner trees on any given set $a_1, \cdots, a_n$ and a geometric method of constructing these, which must include all Steiner minimal trees.

A *full* Steiner tree on $a_1, \cdots, a_n$ is a tree which satisfies (P1) − (P4) and also has $k = n - 2$. The construction of minimal length full Steiner trees on a set is the essential step in the author's algorithm for the Steiner problem; the reason being that any Steiner minimal tree is a union of full Steiner subtrees. The present work is devoted to computing minimal length full Steiner trees (i.e. implementing the geometric procedure given in [1] on a machine) for vertex sets of convex polygons. The mathematical structures used give further insight into the Steiner problem and indicate the way that further research might proceed towards an ultimate solution. Sample results are supplied and the FORTRAN IV programs appear in the microfiche section of this issue.

Finally we note that in our previous paper [1] Steiner trees, full Steiner trees and the Steiner point of a triangle were termed S-trees, $\bar{S}$-trees and the S-point of a triangle respectively.

2. Associations of Full Steiner Trees. Denote by (pq) and (qp) the third vertices of the equilateral triangles on pq as base, (pq) being the point to the left looking from p towards q. Below, we shall have cause to consider "higher order equilateral points" e.g. $((p_1p_2)p_3)$ which is the third vertex of an equilateral triangle whose base is the line joining (p_1p_2) and p_3 (see Fig. 1).
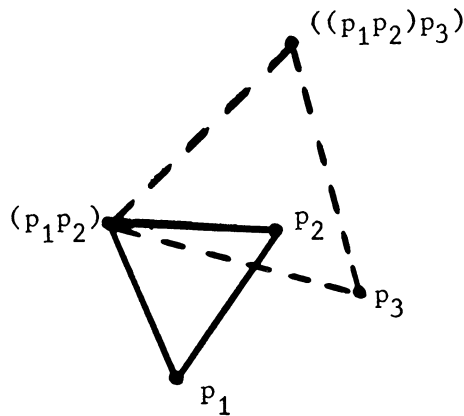

FIGURE 1

If a_1, a_2, a_3 is a clockwise order of the vertices of a triangle having a Steiner point S, then the line segments $(a_1a_2)a_3$, $(a_2a_3)a_1$, $(a_3a_1)a_2$ are concurrent at S and further each of these line segments has length equal to $a_1S + a_2S + a_3S$. This provides a simple construction for S (we term it the "equilateral construction") which plays a fundamental role in the Steiner algorithm.

In [1] we defined the term "association of a full Steiner tree." This definition is repeated here for completeness. Let U^1 be such a tree on $A^1 = \{a_1, \cdots, a_n\}$ with extra vertices $s_1, \cdots, s_{n-2}$, (these will be called a-points and s-points respectively). U^1 has at least one pair of a-points (say a_1, a_2) which are directly joined to the same s-point (say s_1) in U^1. In fact a portion of the tree appears as in Fig. 2.
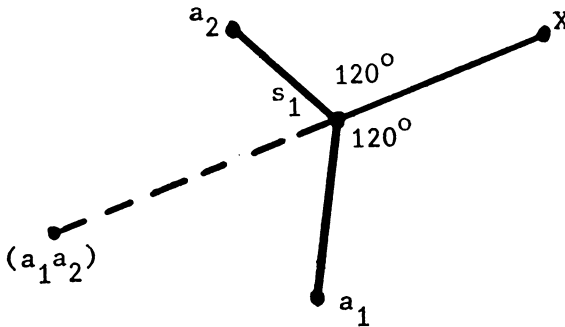

FIGURE 2

X is the third point which directly joins s_1 in U^1. s_1 is the Steiner point of the triangle a_1a_2X, hence by the equilateral construction the line Xs_1 produced passes through (a_1a_2). Therefore the tree U^2 on $A^2 = \{(a_1a_2), a_3, \cdots, a_n\}$ with s-points $s_2, \cdots, s_{n-2}$, formed from U^1 by replacing the segments a_1s_1, a_2s_1, s_1X by the single segment $(a_1a_2)X$, has all the Steiner tree properties except perhaps (P1). a_1, a_2 are said to be "paired" in the formation of U^2 from U^1. This process may be repeated forming trees U^3 on A^3, $\cdots$, U^{n-1} on A^{n-1}. At each stage two points of A^i are paired to form A^{i+1} and U^{i+1} has one less a-point and s-point than U^i. The process terminates since U^{n-1} has two a-points and no s-points. The two points of A^{n-1} can be expressed in terms of the original a-points of U^1 and the equilateral point bracketing notation defined above. This representation of A^{n-1} is called an association of U^1 and the segment joining the two points of A^{n-1} is called an "axis" of U'. There are many associations and axes of a given full Steiner tree.

Any formal combination of the symbols $a_1, \cdots, a_n$ by bracketing as above which forms precisely two points is called an association of $a_1, \cdots, a_n$ and defines a possible structure for a full Steiner tree. Whether or not there exists a full Steiner tree, having this structure, on a particular set of points $a_1, \cdots, a_n$ in E^2, depends on their geometric locations. Associations which define the same full Steiner tree structure are said to be equivalent (written $\equiv$).

The equilateral construction shows that length is preserved at each stage of the above process, i.e. for all $i = 1, \cdots, n - 1$ the length of U^i is constant. Thus the length of a full Steiner tree is equal to the length of any of its axes. This is an important fact which will be used in the computation.

Example. In Fig. 3. the continuous line is U^1, a full Steiner tree on $A^1 = \{1, 2, 3, 4, 5\}$.
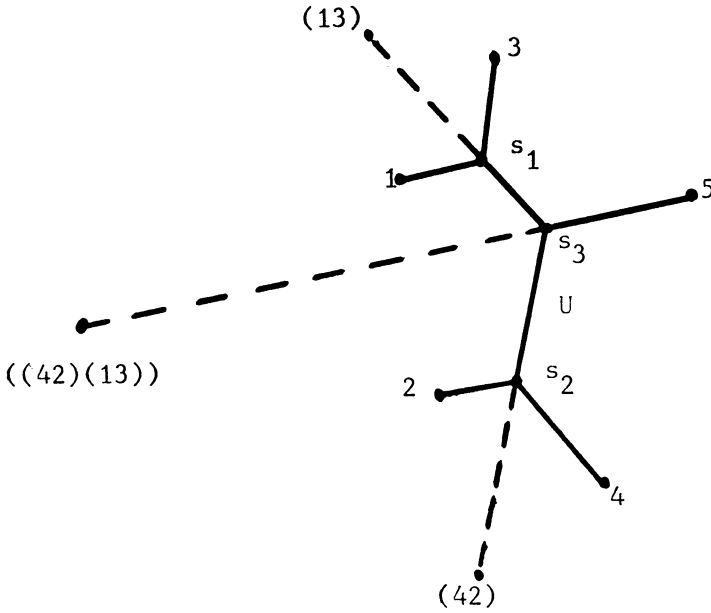

FIGURE 3

We "pair" the points 1 and 3 giving

$$A^2 = \{(13), 2, 4, 5\} \text{ and } U^2 \text{ with branches } (13)s_3, s_35, s_2s_3, 2s_2, 4s_2.$$

Next we pair 4 and 2

$$A^3 = \{(13), (42), 5\}, U^3 \text{ has branches } (42)s_3, (13)s_3, s_35.$$

Finally we pair (13) and (42)

$$A^4 = \{((42)(13)), 5\}, U^4 \text{ has branch } ((42)(13))5.$$

$((42)(13))$, 5 is an association of U^1 and the segment joining $((42)(13))$ and 5 is an axis of U^1.

Let (ab), c be an association of a full Steiner tree on $a_1, \cdots, a_n$ (i.e. a, b, c are themselves combinations of $a_1, \cdots, a_n$ with equilateral point bracketings). Then by inspection of the unique full Steiner tree on the three points a, b, c it follows that

(1) $\qquad (ab), c \equiv a, (bc) \equiv (bc), a \equiv b, (ca) \equiv (ca), b \equiv c, (ab)$

i.e. we get equivalent associations by interchanging the comma and parentheses or by permuting the letters provided that the cyclic order of the letters is left invariant. As a consequence, any full Steiner tree has an association in which any specified point appears last and is immediately preceded by the comma. Geometrically, this means that any a-point of a full Steiner tree may be used as one of the vertices of an axis. Further this association is unique. As an example, using two applications of (1) we determine the association of the tree U^1 of Fig. 3 terminating in " , 1"

$$((42)(13)), 5 \equiv (5(42)), (13) \equiv (3(5(42))), 1.$$

We notice that repeated applications of (1) do not disturb the cyclic order of the points. Summarising, we have proved:

THEOREM 1. *Any full Steiner tree has a unique association terminating in the comma and a specified point. All associations of a full Steiner tree have the same cyclic order of points.*

Theorem 1 is extremely useful in the struggle to improve the efficiency of our Steiner algorithm. We strongly suspect that for any set of points $\{a_1, \cdots, a_n\}$, geometric considerations (perhaps of the type discussed in [4]) will allow only very few cyclic orders of the a_i in associations of minimal length full Steiner trees. The following theorem shows the truth of this conjecture when convexity is assumed.

THEOREM 2. *Let H be the set of vertices of the polygon $C(A)$ which bounds the convex hull of $A = \{a_1, \cdots, a_n\}$ and let ω be any association of any full Steiner tree U on A. Then the order in which the points of H appear in ω, is a clockwise order of the vertices of $C(A)$.*

In other words the theorem states that the cyclic order of the a-points in any full Steiner tree association and the clockwise order of the vertices of the convex hull boundary are compatible. We offer an illustration before proceeding with the proof of the theorem. Let $H = \{a_1, a_2, a_3, a_4, a_5\}$ be a clockwise ordering of the vertices of $C(A)$ where $A = \{a_i : i = 1, \cdots, 7\}$. The theorem asserts that associations of full Steiner trees on A do not have orders of A such as $a_5a_1a_4a_2a_6a_3a_7$. For the points of H occur here in the order $a_5a_1a_4a_2a_3$ which is not a clockwise order.

LEMMA. *Let a_i, a_j, a_k be distinct a-points of U and let σ be the s-point of U at which the unique paths a_ia_j, a_ia_k of U split. Suppose that $x_i\sigma$, $x_j\sigma$, $x_k\sigma$ are the last edges of the paths $a_i\sigma, a_j\sigma, a_k\sigma$ respectively (see Fig. 4). For $t = i, j, k$ either x_t is another s-point of U or $x_t = a_t$. If (as in Fig. 4) looking from x_i along $x_i\sigma$, the path σa_j branches to the left and σa_k to the right, then in any association of U, a_i, a_j, a_k occur in this cyclic order.*

Proof. By successive pairings as above, we can reduce U to a tree U^{n-2} having the point σ as its only s-point and 3 a-points $\alpha_i, \alpha_j, \alpha_k$ which are bracketing combinations of the original a-points of U containing a_i, a_j, a_k respectively and which lie on $\sigma x_i, \sigma x_j, \sigma x_k$ (perhaps produced) respectively. The final pairing forming the association $(\alpha_i\alpha_j), \alpha_k$ of U and Theorem 1 complete the proof of the lemma.

Proof of Theorem 2. Let $a_ua_va_w$ be any three consecutive points in clockwise order on C(A) and σ be the s-point of U where the paths a_ua_v and a_ua_w split. Since all s-points of U are within $C(A)$ and U is non-self-intersecting, in the terminology of the lemma, σa_v branches to the left and σa_w to the right. By the lemma $a_ua_va_w$ have this cyclic order in any association of U. Hence Theorem 2.

COROLLARY. *Let $a_1, \cdots, a_n$ be the vertices of a convex polygon. Then the cyclic order of the a_i in any association of a full Steiner tree on $a_1, \cdots, a_n$ is the clockwise order of vertices on the polygon.*

The methods of the subsequent sections successively generate associations of $A = \{a_1, \cdots, a_n\}$ and endeavour to construct the full Steiner tree on A from each association, if it exists. The above theorem shows that *all* full Steiner trees on a set of n points, r of which are on the convex hull boundary, may be constructed from the subclass of associations of the a_i in which the clockwise order of the r vertices of the

convex hull boundary is preserved. An elementary combinatorial argument shows that only $(n - 1)!/(r - 1)!$ cyclic orders of the total $(n - 1)!$ cyclic orders need to be considered. In particular, for the present case where we have a convex polygon, $r = n$ and only 1 cyclic order need be processed (which reduces machine time considerably!).
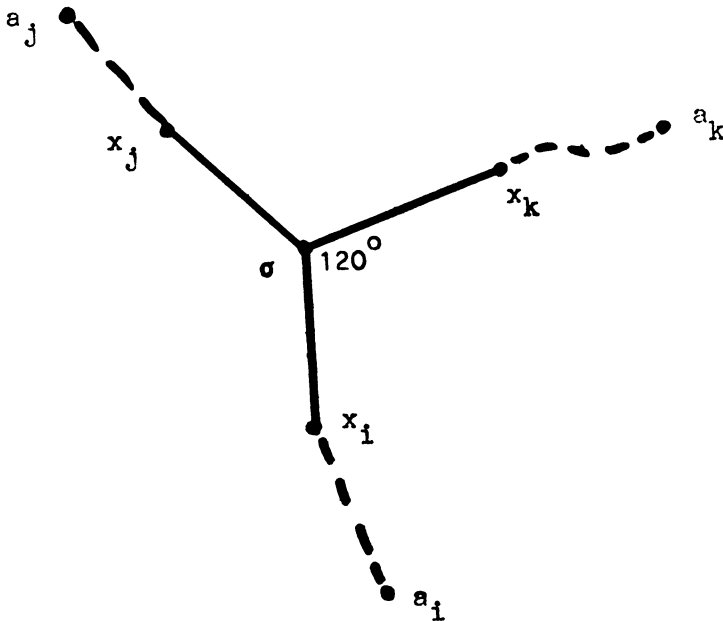

FIGURE 4

3. The A-Matrix. In this and subsequent sections $a_1, \cdots, a_n$ denote, in clockwise order, the vertices of a convex polygon and the a_i are assumed to be complex numbers wherever necessary.

Our algorithm will process each element of Ω, the class of all associations of $a_1, \cdots, a_n$ which exhibit the a_i in this order and in which the comma directly precedes a_n. For each $\omega \in \Omega$, by means of an A-matrix for ω (defined below), the computer will test whether or not $U(\omega)$, a full Steiner tree on $a_1, \cdots, a_n$ with association ω, exists and will isolate a tree of minimum length from the set $\{U(\omega): \omega \in \Omega\}$. Theorems 1 and 2 imply that this will be the required minimal length full Steiner tree on the a_i.

Let $\omega \in \Omega$ and assume, for the moment, the existence of $U(\omega)$. Suppose that $A^1, A^2, \cdots, A^{n-1}$ are successive sets of points encountered in the construction of ω from $U(\omega)$ (cf. Section 2) where for each i the equilateral bracketing representation of A^i exhibits the a_i in their natural order. By an A-matrix of ω, we mean an $n \times n$ matrix of complex numbers in which the first $(n - i + 1)$ elements of the ith row contains the points of A^i $(i = 1, \cdots, n - 1)$ and the remaining elements are zero. Examples will be useful at this point. The matrices M_1, M_2 are both A-matrices for the association $(((12)(34))5), 6$.

1	2	3	4	5	6
(12)	3	4	5	6	0
(12)	(34)	5	6	0	0
((12)(34))	5	6	0	0	0
(((12)(34))5)	6	0	0	0	0
0	0	0	0	0	0

$$\textit{Matrix } M_1$$

1	2	3	4	5	6
1	2	(34)	5	6	0
(12)	(34)	5	6	0	0
((12)(34))	5	6	0	0	0
(((12)(34))5)	6	0	0	0	0
0	0	0	0	0	0

$$\textit{Matrix } M_2$$

4. The Pairing Vector. In this section we show how to ensure that the computer considers all the associations of Ω precisely once.

To each A-matrix there corresponds a vector v, of dimension $n - 1$, with non-negative integral components where $v(1) = 0$ and for each $i = 2, \cdots, n - 1$ the ith row of the A-matrix is formed from the $(i - 1)$th row by pairing the $v(i)$th and $(v(i) + 1)$th elements. This vector we call the pairing vector of the A-matrix. This correspondence is 1-1 between the class of all A-matrices of associations in Ω and the set V of all vectors v, of dimension $n - 1$ with nonnegative integral components satisfying

$$v(1) = 0 \text{ and } 1 \leqq v(i) \leqq n - i, \quad i = 2, \cdots, n - 1.$$

For example, the pairing vectors of the A-matrices M_1 and M_2 given in Section 3 are $\{0, 1, 2, 1, 1\}$ and $\{0, 3, 1, 1, 1\}$ respectively.

Pairing vectors are easily generated on the machine and given a pairing vector and the original $a_1, \cdots, a_n$ it is a simple process for the computer to pair the appropriate points and form the corresponding A-matrix.

The correspondence of A-matrices and associations of Ω and hence the correspondence of pairing vectors and Ω is many-one. Thus, in order to generate precisely one A-matrix for each $\omega \in \Omega$, we (fortunately!) do not have to use all the $(n - 2)!$ vectors of V. In fact it is sufficient to use the subclass V^* of V whose vectors also satisfy

$$v(i) \geqq v(i + 1) \quad \text{for } i = 2, \cdots, n - 2.$$

It is left to the reader to show that the correspondence of Ω and V^* is one-one.

To complete this section we enumerate the associations which our algorithm must process in its search for a minimal length full Steiner tree on n a-points with $k = n - 2$ s-points, i.e. we determine the cardinality $N(\Omega)$ of Ω. Ignoring the convexity assumption, we state that there are $(k + 1)!N(\Omega)$ nonequivalent associations because, by Theorem 1, only those associations terminating in "$, a_n$" need be counted and the remaining $(k + 1)$ a-points may be permuted in $(k + 1)!$ ways. Since an

incidence matrix does not reflect the fact that *ordered* pairs of points are bracketed to form equilateral points at each of the k pairing stages, there are 2^k nonequivalent associations for which the corresponding full Steiner tree structures have the same incidence matrix. Therefore the total number of incidence matrices is $(k + 1)!N(\Omega)/2^k$. We equate this to the expression obtained by Gilbert and Pollak [4] and obtain

$$(k + 1)!N(\Omega)/2^k = (2k)!/k!2^k$$

i.e.

$$N(\Omega) = (2k)!/k!(k + 1)!\,.$$

This formula is also obtained in a book by Ivan Niven [6, Chapter 11].

5. Computational Details. The first part of our algorithm generates the pairing vectors of V^*. For each pairing vector v, the corresponding A-matrix is computed. The ordered n-tuple of complex numbers $a_1, \cdots, a_n$ is read into the first row of the A-matrix, and the remaining rows are generated recursively as follows. For each i from 2 to $n - 1$ the subroutine EQPT calculates the equilateral point for the ordered pair $A(i - 1, v(i))$, $A(i - 1, v(i) + 1)$ and the other elements of the ith row are identical to elements in the $(i - 1)$th row. The axis, corresponding to the vector v is the distance between the two points in the $(n - 1)$th row of the A-matrix. Should this axis be longer than the axis of a tree that has already been constructed by the procedure, then v cannot yield a minimum tree and we proceed to the next pairing vector, otherwise the s-points for the tree are computed. At this stage we may well discover that no full Steiner tree on $a_1, \cdots, a_n$ may be drawn with pairing vector v.

We now give details of the procedure for computation of the s-points from an A-matrix. Each s-point is computed as a Steiner point of a certain triangle by a subroutine known as STEIN. The first s-point is the Steiner point of the three points in the $(n - 2)$th row of the A-matrix. We then work with progressively higher rows of the matrix, considering successively the $(n - i)$th row as i varies from 3 to $n - 1$. For each i, let $x(n - i)$, $y(n - i)$ be the two elements (say α, β) of the $(n - i)$th row which are paired in the formation of the $(n - i + 1)$th row and let $z(n - i)$ be the s-point which has already been computed as the Steiner point of a triangle, one of whose vertices is the equilateral point $(\alpha\beta)$. Then the next s-point is the Steiner point of $x(n - i)$, $y(n - i)$, $z(n - i)$. If at any stage no Steiner point for the triangle exists, there is no tree corresponding to v and we proceed to the next pairing vector. As an illustration we list the steps in the construction of the s-points from the A-matrix M_1 of Section 3.

	X	Y	Z
s_1 is the Steiner point of	$((12)(34))$	5	6
s_2 is the Steiner point of	(12)	(34)	s_1
s_3 is the Steiner point of	3	4	s_2
s_4 is the Steiner point of	1	2	s_2

Assuming the existence of a minimum length full Steiner tree on $a_1, \cdots, a_n$, the output from the program gives the length of such a tree, the pairing vector which

produced it, the s-points of the tree and the triangles from which they were computed. The incidence matrix is simply derived from this information.

On the IBM 360/44 machine at the University of Victoria, approximately 25 minutes of computing time were used for examples with $n = 12$. With the present program and comparable times on a faster machine, we estimate one could run the algorithm up to $n = 17$.

We are presently trying to increase the efficiency of the program by implementing the "deciding region" criterion of [4], a geometric method for rejecting certain pairings of points in associations. We note here that this criterion can be used at any stage in the formation of an association.
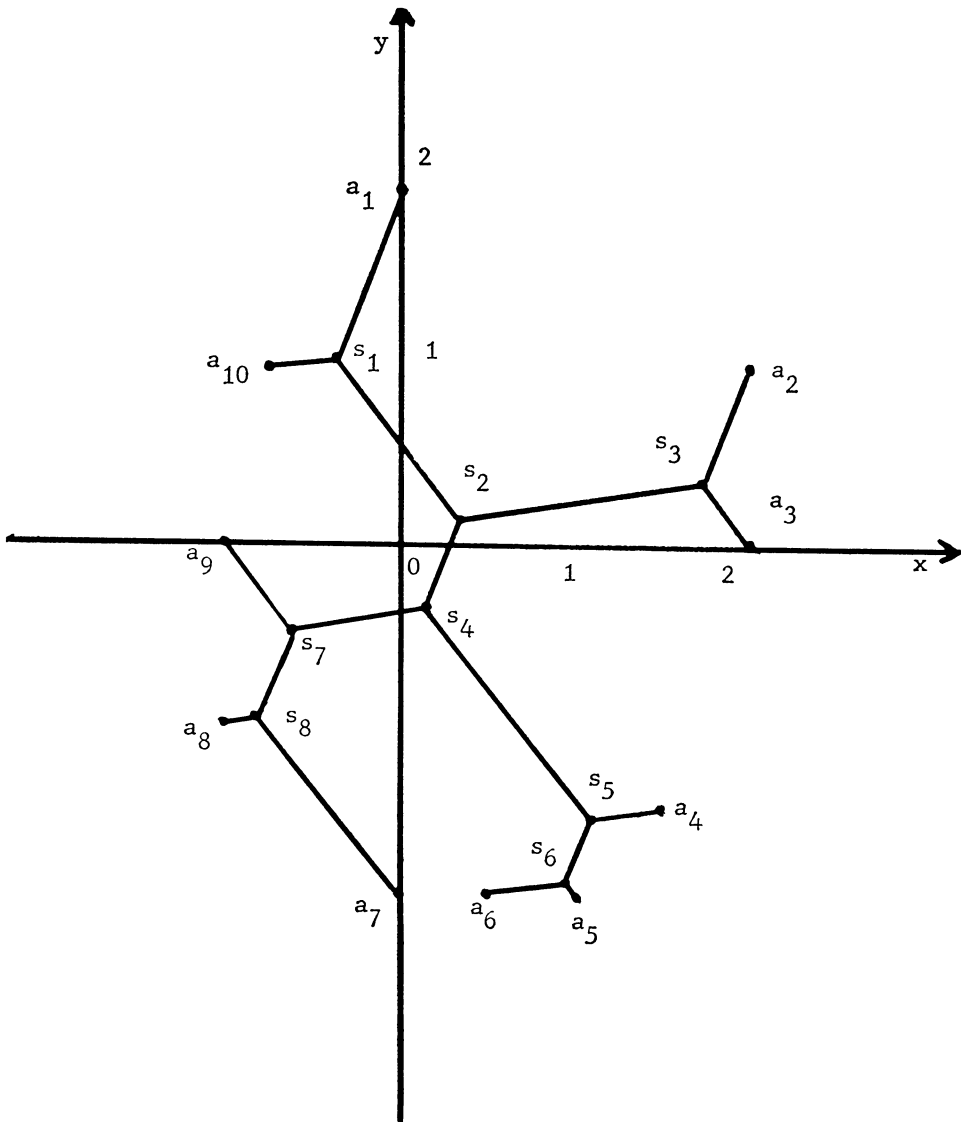

FIGURE 5. *Sketch of the Minimal Length Full Steiner Tree Computed in Example 3.*

APPENDIX I. *Sample Results.*
Example 1.

$$N\text{-}4$$

A-POINTS

$$-3.000, \quad 1.000$$
$$3.000, \quad 1.000$$
$$3.000, \; -1.000$$
$$-3.000, \; -1.000$$

FULL STEINER TREE
LENGTH = 9.464
PAIRING VECTOR IS 0 2 1

	X		Y		Z		S-POINT	
1	$-3.000,$	1.000	$4.732,$	0.0	$-3.000,$	-1.000	$-2.423,$	0.0
2	$3.000,$	1.000	$3.000,$	-1.000	$-2.423,$	0.0	$2.423,$	0.0

Example 2.

$$N = 5$$

A-POINTS

$$-1.000, \quad 2.000$$
$$1.000, \quad 1.000$$
$$1.000, \; -1.000$$
$$-1.000, \; -1.000$$
$$-2.000, \quad 0.0$$

FULL STEINER TREE
LENGTH = 7.433
PAIRING VECTOR IS 0 1 1 1

	X		Y		Z		S-POINT	
1	$4.598,$	1.232	$-1.000,$	-1.000	$-2.000,$	0.0	$-0.974,$	-0.847
2	$0.866,$	3.232	$1.000,$	-1.000	$-0.974,$	-0.847	$0.258,$	-0.387
3	$-1.000,$	2.000	$1.000,$	1.000	$0.258,$	-0.387	$0.457,$	0.797

Example 3.

$$N = 10$$

A-POINTS

$$0.0 \quad , \quad 2.000$$
$$2.000, \quad 1.000$$
$$2.000, \quad 0.0$$
$$1.500, \; -1.500$$
$$1.000, \; -2.000$$
$$0.500, \; -2.000$$
$$0.0 \quad , \; -2.000$$
$$-1.000, \; -1.000$$
$$-1.000, \quad 0.0$$
$$-0.750, \quad 1.000$$

FULL STEINER TREE
LENGTH = 11.938
PAIRING VECTOR IS   0 7 7 5 4 4 2 2 1

	X	Y	Z	S-POINT
1	0.0 , 2.000	6.165, −7.214	−0.750, 1.000	−0.375, 1.055
2	2.866, 0.500	−2.165, −6.214	−0.375, 1.055	0.355, 0.132
3	2.000, 1.000	2.000, 0.0	0.355, 0.132	1.736, 0.334
4	1.933, −2.616	−3.232, −0.866	0.355, 0.132	0.156, −0.369
5	1.500, −1.500	0.750, −2.433	0.156, −0.369	1.097, −1.559
6	1.000, −2.000	0.500, −2.000	1.097, −1.559	0.948, −1.934
7	−1.366, −2.366	−1.000, 0.0	0.156, −0.369	−0.618, −0.483
8	0.0 , −2.000	−1.000, −1.000	−0.618, −0.483	−0.813, −0.973

Department of Mathematics
University of Auckland
Auckland, New Zealand

1. E. J. Cockayne, "On the Steiner problem," *Canad. Math. Bull.*, v. 10, 1967, pp. 431–450. MR **35** #6585.

2. E. J. Cockayne & Z. A. Melzak, "On Steiner's problem for set-terminals," *Quart. Appl. Math.* (To appear.)

3. R. Courant & H. Robbins, *What is Mathematics*, Oxford Univ. Press, New York, 1941. MR **3**, 144.

4. E. N. Gilbert & H. O. Pollak, "Steiner minimal trees," *SIAM J. Appl. Math.*, v. 16, 1968, pp. 1–29. MR **36** #6317.

5. Z. A. Melzak, "On the problem of Steiner," *Canad. Math. Bull.*, v. 4, 1961, pp. 143–148. MR **23** #A2767.

6. Ivan Niven, *Mathematics of Choice, or How to Count Without Counting*, Random House, New York, 1965.

7. R. C. Prim, "Shortest connecting networks," *Bell System Tech. J.*, v. 31, pp. 1398–1401.