Geometric Conditions for Euclidean Steiner Trees in $\Re^d$

Jon W. Van Laarhoven
Department of Applied Mathematics and Computational Sciences
University of Iowa, Iowa City, IA 52242
jon-vanlaarhoven@uiowa.edu

Kurt M. Anstreicher*
Department of Management Sciences
University of Iowa, Iowa City, IA 52242
kurt-anstreicher@uiowa.edu

July 1, 2010

Abstract

We present geometric conditions that can be used to restrict or eliminate candidate topologes for Euclidean Steiner minimal trees in $\Re^d$, $d \geq 2$. Our emphasis is on conditions that are not restricted to the planar case ($d = 2$). For trees with a Steiner topology we give restrictions on terminal-Steiner connections that are based on the Voronoi diagram associated with the set of terminal nodes. We then describe more restrictive conditions for trees with a full Steiner topology and show how these conditions can be used to improve implicit enumeration algorithms for finding Euclidean Steiner minimal trees with $d > 2$.

Keywords: Euclidean Steiner tree, branch and bound, Voronoi diagram, Delaunay triangulation

*Corresponding author

1 Introduction

The objective of the Euclidean Steiner tree problem (ESTP) is to determine the minimal length tree (with respect to the Euclidean metric) spanning a set of *terminal points*, $X \subset \Re^d$, while permitting the introduction of extra *Steiner points* S into the network to reduce its overall length. The ESTP is a difficult combinatorial optimization problem; Garey et al. (1977) shows that the recognition version of ESTP is NP-hard. Arora (1998) shows that ESTP belongs to the class of NP-hard problems which have a polynomial-time approximation scheme (PTAS).

A *topology* is a configuration of terminal points and Steiner points where the connections are specified, but the locations of the Steiner points are not. A topology is said to be a *Steiner topology* if every Steiner point has degree 3 and every terminal node has degree at most 3. A Steiner topology where all terminal points have degree one is a *full Steiner topology* (FST). Any non-full Steiner topology can be identified with a FST where some edges have zero length; such a FST is called *degenerate*. A solution to the ESTP problem is called a *Steiner minimal tree* (SMT). It can be shown that a SMT has a Steiner topology. Other known results regarding SMT's include (Gilbert and Pollak, 1968):

- The *angle condition*: the angles between edges connecting a Steiner point and its 3 neighbors are all 120 degrees.
- A SMT is a concatenation of FST's on subsets of terminal points.
- A SMT for a problem on N terminal points has at most $N - 2$ Steiner points.

The main difficulty with solving the ESTP to optimality is that the number of Steiner topologies grows extremely rapidly with n, the number of terminals. For planar ESTP ($d = 2$), a sequence of advancements have lead to substantial increases in the size of instances that can be solved to optimality by exact algorithms. In particular, the Geosteiner algorithm (Winter and Zachariasen, 1997; Warme et al., 2001) makes intensive use of geometric exclusion criteria to eliminate candidate topologies and is capable of solving instances with thousands of terminal points. For higher dimensions, however, exact algorithms are very limited as most of the known pruning rules are inapplicable when $d > 2$. The most interesting case with $d > 2$ is of course $d = 3$; see for example Smith and Toppur (1996).

For $d > 2$, the scheme of Smith (1992) is the best known method for computing and verifying the optimality of a SMT. Smith's algorithm is based on an implicit enumeration of FSTs, where each level $k \geq 0$ of the enumeration tree has nodes corresponding to all FSTs on a subset of $n = k + 3$ terminal nodes. The descendants ("children") of a given partial FST are obtained by replacing in turn each edge l of the corresponding tree with 3 edges connecting a Steiner point and a new terminal using the "sprout" or "merge" operation illustrated in Figure 1. Smith showed that by performing this merge operation $N - 3$ times, all possible FSTs on the N terminal points will be generated. In addition, it can be shown that the merge operation cannot *decrease* the minimal length of a tree with the given topology, so if the minimal length

tree with a given partial FST is longer than a known Steiner tree on all the terminal points, it and all of its descendants can be removed from consideration (or "fathomed" in the terminology of branch-and-bound algorithms). The Smith+ algorithm of Fampa and Anstreicher (2008) enhances the algorithm of Smith (1992) by using second-order cone programming to locate the Steiner points as well as "strong branching" to accelerate the fathoming process. For $d > 2$, the Smith+ algorithm is capable of solving instances with about $N = 16$ terminal points to optimality.

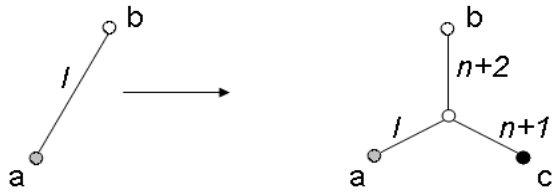

Figure 1: The "merge" operation used to create descendants in Smith's enumeration scheme

The main drawback of Smith's algorithm is that the fathoming criterion (the minimal length of a tree with a given FST on a subset of the terminals) is quite weak, and cannot be expected to remove many topologies from consideration until a substantial number of the terminal points are included. This deficiency, combined with the fact that the number of distinct FSTs grows super-exponentially with the number of terminals, means that the enumeration process can easily get out of hand for even relatively small problems.

A surprising feature of Smith's scheme is that it makes no use of geometry whatsoever to reduce the search space that must be considered. By contrast the GeoSteiner algorithm, which is restriced to $d = 2$, makes very extensive use of geometric exclusion criteria. One difficulty in trying to apply geometric criteria to Smith's algorithm is that the FSTs at intermediate nodes in the enumeration tree only include a subset of terminal points. As a result, even if some property of SMTs is violated at an intermediate node, it is possible that after some number of merge operations the property will hold ("bad" partial FSTs can have "good" descendants). Our goal in this paper is to derive geometric conditions that apply for $d > 2$ and that can be used to eliminate partial FSTs from further consideration.

In the next section we describe geometric conditions that apply to SMTs in $\Re^d$, $d \geq 2$. The conditions that we describe are all related to the Voronoi diagram induced by the terminal nodes, and extend to the nonplanar case. In section 3 we describe stronger conditions that apply to SMTs having a FST. These conditions are of particular interest for $d > 2$ since Smith's enumeration scheme works with FSTs. In section 4 we show how the geometric conditions developed in section 3 can be used to enhance fathoming of candidate partial FSTs in Smith's algorithm, and give computational results on test problems with $d > 2$.

2 Geometric conditions for Euclidean Steiner minimal trees

Let $X \subset \Re^d$ be a set of terminal points $\{x_0, ..., x_{N-1}\}$, and assume without loss of generality that x_0 is the origin in $\Re^d$. Denote a set $S \subset \Re^d$ of Steiner points by $\{s_1, ..., s_k\}$. The *Voronoi diagram* of X is a partitioning of $\Re^d$ into polyhedra with one node x_i in each polyhedra. For every point $x \in X$ the Voronoi region about x, denoted $\text{vor}(x)$ consists of all points no further from x than from any other point in X; that is $\text{vor}(x) = \{u \in \Re^d : \ ||u - x|| \leq ||u - y|| \ \forall \ y \in X\}$. Since the Voronoi regions are polyhedra they have extreme points, which are called *Voronoi points*. Let $V = \{v_1, .., v_m\}$ be the set of Voronoi points. The Voronoi diagram is a natural candidate for developing geometric criteria for SMTs since it exists for $X \subset \Re^d$ for any d, and is itself defined by a condition involving Euclidean norms. The dual of the Voronoi diagram is called the *Delaunay tesselation*. More specifically, to each Voronoi point v there is an associated Delaunay cell consisting of the convex hull of $\{x \in X : \ v \in \text{vor}(x)\}$. When the points $X \subset \Re^d$ are in general position, each Voronoi point is containined in exactly $d + 1$ Voronoi regions, and each Delaunay cell is a simplex. In this case the Delaunay tesselation is a partioning of the convex hull of X into simpleces, and the tesselation is a triangulation. Since Delaunay cells that are not simpleces can always be further partioned to obtain a triangulation, the Delaunay tesselation is often referred to as the *Delaunay triangulation*. For more details on properties of the Voronoi diagram and Delaunay triangulation see for example De Berg et al. (2008).

Let $B_\delta(x)$ denote the closed ball of radius δ centered at x. The *lune* between two points u and v, denoted $l(u, v)$ is defined as $B_{\|u-v\|}(u) \cap B_{\|u-v\|}(v)$. Lunes are the basis for the following well-known geometric criterion for SMTs, the *lune condition*.

Proposition 1. (Gilbert and Pollak, 1968) *If points u and v are connected in a SMT, then the interior of $l(u, v)$ contains no other nodes of the SMT.*

The lune condition is an example of a geometric condition that can be used to exclude connections between points in the SMT. For example, if there are terminals x, y, z such that z is in the interior of $l(x, y)$, then x and y cannot be connected in a SMT. The lune condition is also related to the Delaunay triangulation, as shown in the next lemma. For points X in general position we say that terminals x_i and x_j are *adjacent* in the Delaunay triangulation if x_i and x_j are vertices of a common simplex in the Delaunay triangulation, or equivalently if $\text{vor}(x_i)$ and $\text{vor}(x_j)$ share a common boundary face of dimension $d - 1$.

Lemma 2. *Suppose that the points in X are in general position. If two terminals x_i and x_j are not adjacent in the Delaunay triangulation, then the lune between them contains another terminal node in its interior.*

Proof. Let p denote the midpoint of the line segment between x_i and x_j. First note that p is neither in the interior of $\text{vor}(x_i)$ or $\text{vor}(x_j)$. To see this, assume that p was in the interior of $\text{vor}(x_i)$. Then there would be some q also in the interior of $\text{vor}(x_i)$ such that q was on the line segment between p and x_j. But recall that p was the midpoint of the segment between x_i and x_j, so this means q is closer to x_j than it is to x_i and so could not be in $\text{vor}(x_i)$, a contradiction. Similarly p is not in the interior of $\text{vor}(x_j)$. Therefore there

is another terminal z so that $p \in \text{vor}(z)$, $z \neq x_i$, $z \neq x_j$. (If there were no such z then p would be on a boundary facet of both $\text{vor}(x_i)$ and $\text{vor}(x_j)$, which is impossible since x_i and x_j are nonadjacent.) We can now show $z \in l(x_i, x_j)$. For example,

$$\|x_i - z\| \leq \|x_i - p\| + \|z - p\| \leq 2\|x_i - p\| = \|x_i - x_j\|,$$

where the first inequality follows from the triangle inequality and the second follows from $p \in \text{vor}(z)$. Likewise, we can show $\|x_j - z\| \leq \|x_i - x_j\|$, and these two inequalities combined give $z \in l(x_i, x_j)$. Finally, if the points X are in general position then the midpoint p cannot be on the boundary of either $\text{vor}(x_i)$ or $\text{vor}(x_j)$, which makes the second inequality strict, and therefore z is in the interior of the lune. □

Proposition 1 and Lemma 2 together imply the following well-known property of SMTs.

Corollary 1. *Assume that the points in X are in general position. If a SMT contains an edge between terminal points x_i and x_j, then x_i and x_j are adjacent in the Delaunay triangulation.*

For points not in general position, Corollary 1 remains true as stated so long as "adjacency" of two terminals is taken to mean that there is a cell in the Delaunay tesselation containing them both.

2.1 Clover regions

Corollary 1, which concerns connections between terminal points in an SMT, can also be used to prove a property regarding connections between terminal points and Steiner points.

Lemma 3. *Assume that a Steiner point s is connected to a terminal node x in a SMT. Let* $\text{vor}(x)$ *denote the Voronoi region of x in the Voronoi diagram associated with X. Form the Voronoi diagram of $X \cup \{s\}$ treating s as an additional terminal node, and denote the resulting Voronoi region of s by* $\widehat{\text{vor}(s)}$. *Then* $\widehat{\text{vor}(s)} \cap \text{vor}(x) \neq \emptyset$.

Proof. The proof follows from Corollary 1 and the fact that a SMT is also a SMT for the set of terminal nodes $X \cup \{s\}$ where the location of the Steiner point s is fixed. □

Note that the condition $\widehat{\text{vor}(s)} \cap \text{vor}(x) \neq \emptyset$ in Lemma 3 is equivalent to the hyperplane $\{u : ||u - s|| = ||u - x||\}$ intersecting $\text{vor}(x)$, and since $\text{vor}(x)$ is a polyhedra, this hyperplane intersects $\text{vor}(x)$ if and only if $||v - s|| \leq ||v - x||$ for one of the Voronoi points v of $\text{vor}(x)$. This leads us to the concept of a *clover region* associated with x, denoted $\text{clover}(x)$. For convenience take $x = x_0 = 0$, and let the Voronoi points in $\text{vor}(0)$ be $\{v_1, \ldots, v_k\}$. We consider connecting x_0 to a Steiner point s and ask where s may lie so that Lemma 3 is satisfied. The perpendicular bisector between x_0 and s can be written $\{u \mid s^T u = s^T \left(\frac{s}{2}\right) = \frac{||s||^2}{2}\}$. Note

that v_i and s are on opposite sides of this bisector if $s^T v_i < \frac{||s||^2}{2}$. Then define

$$\begin{aligned} C_i &= \{s \mid ||s||^2 \leq 2s^T v_i\} \\ &= \{s \mid (s^T s - 2s^T v_i + v_i^T v_i) - v_i^T v_i \leq 0\} \\ &= \{s \mid ||s - v_i||^2 \leq ||v_i||^2\} \\ &= B_{\|v_i\|}(v_i). \end{aligned}$$

For each Voronoi point v_i of $\text{vor}(x_0)$, the associated ball C_i describes where a Steiner point can lie so that $\widehat{\text{vor}(s)}$ contains v_i. We define $\text{clover}(x_0)$ to be the union of these balls over the Voronoi points $\{v_1, \ldots, v_k\}$. We similarly construct $\text{clover}(x_j)$ for each terminal node x_j, $j = 1, \ldots, N-1$. Note that by construction each sphere C_i has the property that it intersects the terminal points x_j such that $v_i \in \text{vor}(x_j)$; in other words, each C_i is the circumsphere of a Delaunay simplex. Thus for each x_j, $\text{clover}(x_j)$ is the union of the circumspheres of the Delaunay simpleces that have x_j as a vertex. See Figure 2 for the illustration of a typical clover region for $d = 2$.

The clover region represents a first attempt at using geometry to restrict topologies that involve Steiner points. In particular, note that if x_i and x_j are connected to the same Steiner point, then their clover regions must intersect. Moreover, since $\text{clover}(x_i)$ and $\text{clover}(x_j)$ are both the unions of spheres, the condition that $\text{clover}(x_i) \cap \text{clover}(x_j) \neq \emptyset$ can be efficiently checked. In the sequel we will develop more restricive conditions that must be satisfied if two points x_i and x_j are connected to the same Steiner point.

2.2 Lunar regions

Lemma 3, which leads to the definition of a clover region, itself can be viewed as a consequence of the lune condition. A natural question is then if the lune condition can be directly used to give a more restrictive condition for connections between a terminal node and a Steiner point in a SMT. To answer this question we consider the terminal $x_0 = 0$ connected to a Steiner point s, and determine the restrictions on the location of s implied by the other terminals and the lune property. In particular, s cannot be located so that $l(0, s)$ contains another terminal node x in its interior, meaning that for this particular x we need

$$\begin{aligned} \text{either} \quad & ||x - s|| \geq \|s\| \Leftrightarrow s^T s \leq x^T x - 2s^T x + s^T s \Leftrightarrow s^T x \leq \frac{||x||^2}{2}, \\ \text{or} \quad & \|s\| \leq \|x\|. \end{aligned}$$

The feasible set corresponding to these two conditions is the union of a halfspace and a halfsphere, where the hyperplane bounding the halfspace bisects the sphere. There is such a region for each terminal point $x_i \in X \setminus \{0\}$, and taking the intersection of all such regions for x_i adjacent to $x_0 = 0$ in the Delaunay triangulation, we arrive at the *lunar region* associated with x_0, denoted $\text{lunar}(x_0)$[1]. We can similarly associate

[1]It can be shown by example that there are cases where the lunar region can be further restricted by imposing constraints from terminals that are not adjacent to x_0 in the Delaunay triangulation.

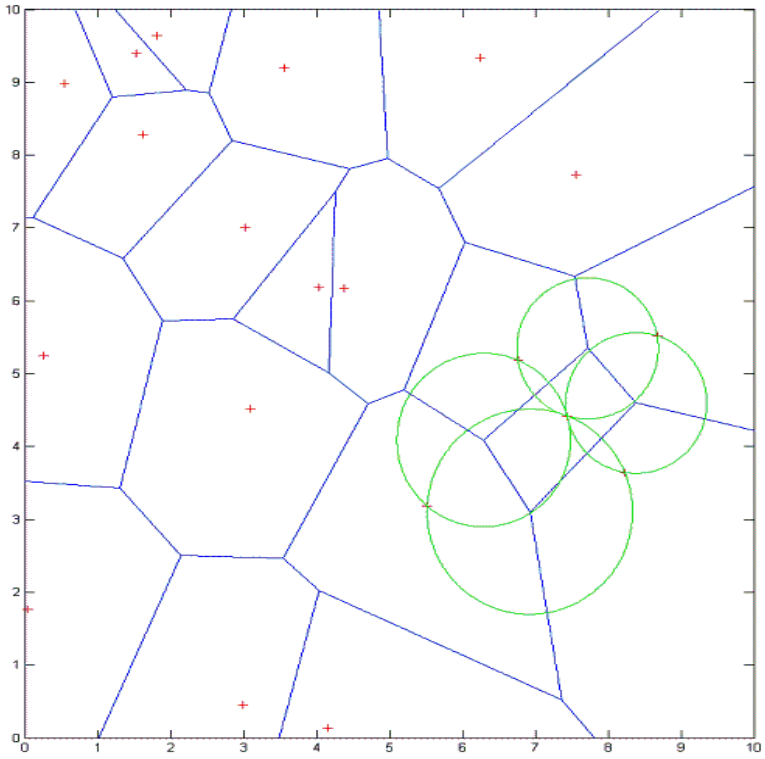

Figure 2: A clover region

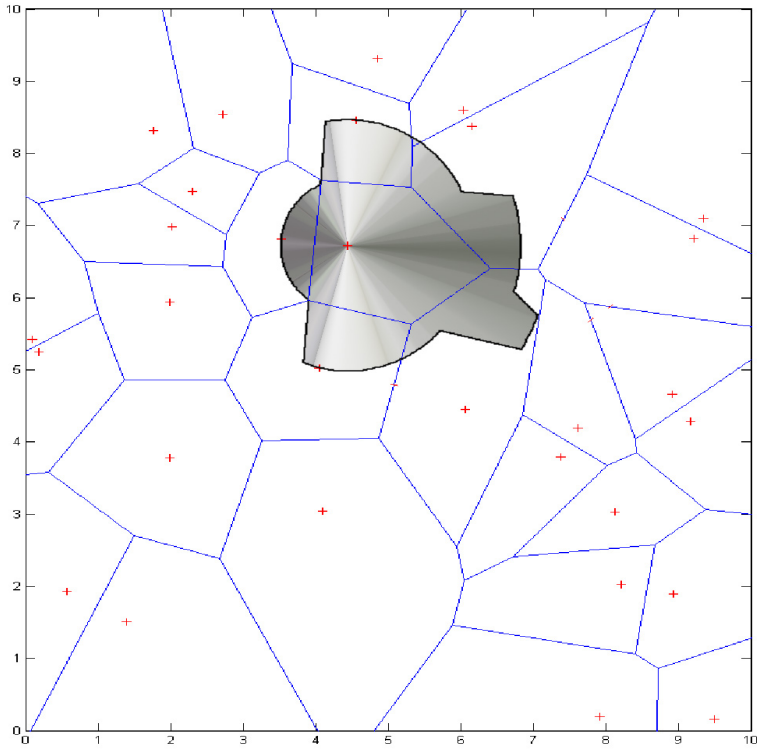

Figure 3: A lunar region

a lunar region lunar(x) to every terminal $x \in X$. A typical lunar region for $d = 2$ is depicted in Figure 3. We will show below that for any terminal x, lunar(x) $\subset$ clover(x), so the lunar region is a sharper estimate for possible locations of a Steiner point connected to x than the clover region. As with clover regions, lunar regions can be used to exclude possible Steiner topologies; if lunar(x_i) $\cap$ lunar(x_j) $= \emptyset$, then one need not consider any topology in which x_i and x_j connect to the same Steiner point. However, because lunar regions are defined by the intersection of nonconvex constraints, determining whether or not two such regions overlap is somewhat awkward. We next show that there is an easily-computed convex relaxation of the lunar region which is itself contained in the clover region.

2.3 Doubled Voronoi cells

The doubled Voronoi cell about x, denoted dvor(x), is simply vor(x) dilated by a factor of two about x; see Figure 4 for a typical illustration with $d = 2$. Note that each semi-spherical portion of the lunar region lunar(x) is tangent to the corresponding bounding hyperplane of the doubled Voronoi cell dvor(x), as shown in Figure 4. As an immediate consequence of this fact we obtain the following relationship between lunar regions and doubled Voronoi cells.

Lemma 4. *For any terminal node x,* vor(x) $\subset$ lunar(x) $\subset$ dvor(x).

Proof. Assume without loss of generality that $x = x_0 = 0$. Let x_i, $i = 1, \ldots, k$ be the adjacent terminal points in the Delaunay triangulation. Then

$$\text{dvor}(x_0) = \{x \,|\, x_i^T x \leq \|x_i\|^2,\ i = 1, \ldots, k\}.$$

The lunar region lunar(x_0) is the set of x such that for $i = 1, \ldots, k$ either $x_i^T x \leq \|x_i\|^2/2$ or $\|x\| \leq \|x_i\|$. Note that for each i, the first constraint defines the face of vor(x_0) induced by x_i, and therefore vor(x_0) $\subset$ lunar(x_0). Moreover if x satisfies the second condition for a given i, then

$$x_i^T x \leq \|x_i\|\|x\| \leq \|x_i\|^2.$$

It follows immediately that $x \in$ dvor(x_0) for any $x \in$ lunar(x_0). □

Since dvor(x) is polyhedral for each $x \in X$, the condition that dvor(x_i) $\cap$ dvor(x_j) $\neq \emptyset$ can be efficiently checked, for example by solving a small linear programming problem. We next show that this condition is stronger than the analogous condition based on clover regions. See Figure 5 for motivation of the following theorem, whose proof is due to Carlos de la Mora (private communication).

Theorem 5. *For any terminal node x,* dvor(x) $\subset$ clover(x).

Proof. For simplicity we take $x = x_0 = 0$. Each Delaunay simplex that has x_0 as a vertex has an additional d vertices, each of which is a terminal node. The cones generated by these simpleces partition $\Re^d$, so any

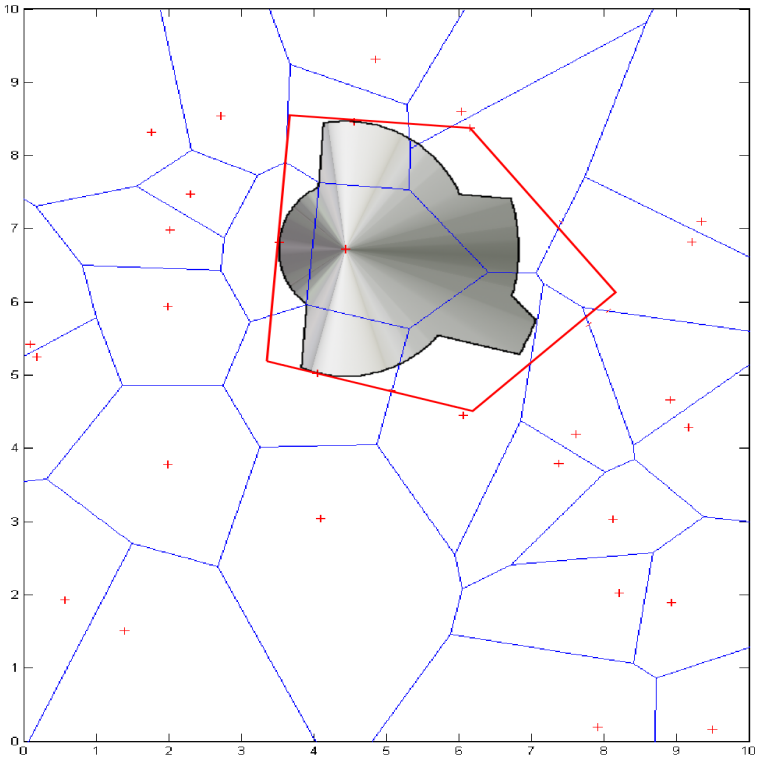

Figure 4: The lunar region is contained in the doubled Voronoi cell.

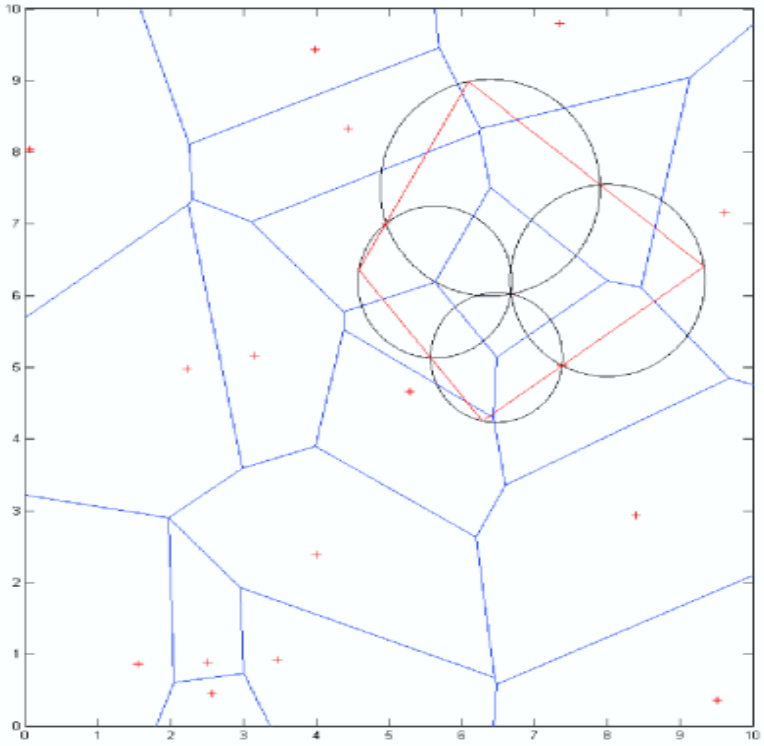

Figure 5: The doubled Voronoi region is contained in the clover region.

$u \in \text{vor}(0)$ must be in at least one such cone. Assume that u is in the cone generated by x_i, $i = 1, \ldots, d$, so there are $\lambda_i \geq 0$, $i = 1, \ldots, d$ such that

$$\sum_{i=1}^{d} \lambda_i x_i = u.$$

Let v be the Voronoi point that is the center of the hypersphere that circumscribes the simplex with extreme points x_i, $i = 0, 1, \ldots, d$. It follows that $\|x_i - v\| = \|v\|$, $i = 1, \ldots, d$, which is equivalent to

$$\|x_i\|^2 = 2x_i^T v, \quad i = 1, \ldots, d. \tag{1}$$

Our goal is to show that $2u \in \text{clover}(0)$. To do this it suffices to show that

$$||2u - v|| \leq ||v||,$$

which is equivalent to $u^T u \leq u^T v$. Since $u \in \text{vor}(0)$ we have $x_i^T u \leq \|x_i\|^2/2$, $i = 1, \ldots, d$, which combined with (1) implies that $x_i^T u \leq x_i^T v$, $i = 1, \ldots, d$. Therefore

$$u^T u = \sum_i \lambda_i x_i^T u \leq \sum_i \lambda_i x_i^T v = u^T v,$$

as required. □

Although the literature concerning Voronoi diagrams and Delaunay triangulations is extensive, to our knowledge Theorem 5 is a new result. Lemma 4 and Theorem 5 together imply the appealing hierarchy

$$\text{vor}(x) \subset \text{lunar}(x) \subset \text{dvor}(x) \subset \text{clover}(x)$$

for any terminal node x. A simple counterexample demonstrates that no inclusion of the form

$$\text{clover}(x) \subset \alpha \cdot \text{vor}(x)$$

holds universally for fixed $\alpha > 2$, where $\alpha \cdot \text{vor}(x)$ denotes a dilation of $\text{vor}(x)$ by a facor of α about x (so $\text{dvor}(x) = 2 \cdot \text{vor}(x)$). The construction of such a counterexample is shown in Figure 6. In the figures, one terminal point q is moved closer and closer to another terminal point p. As q moves closer to p, the face of the Voronoi cell separating p and q moves closer to p, while the circumspheres that make up $\text{clover}(p)$ converge to nonzero radii. It follows that for any $\alpha > 2$ it is possible to move q close enough to p to ensure that $\text{clover}(p)$ is not contained in $\alpha \cdot \text{vor}(p)$. The figures on the right side in Figure 6 include the quadruple Voronoi region $4 \cdot \text{vor}(p)$ for illustrative purposes.

3 Geometric conditions for SMTs with a FST

Clover regions, doubled Voronoi cells, and lunar regions provide increasingly sharp restrictions on the location of a Steiner point connected to a given terminal node in a SMT. When a SMT has a FST, the fact that every terminal is a leaf in the tree allows the feasible locations of Steiner points to be further restricted. A simple edge exchange argument proves the following result.

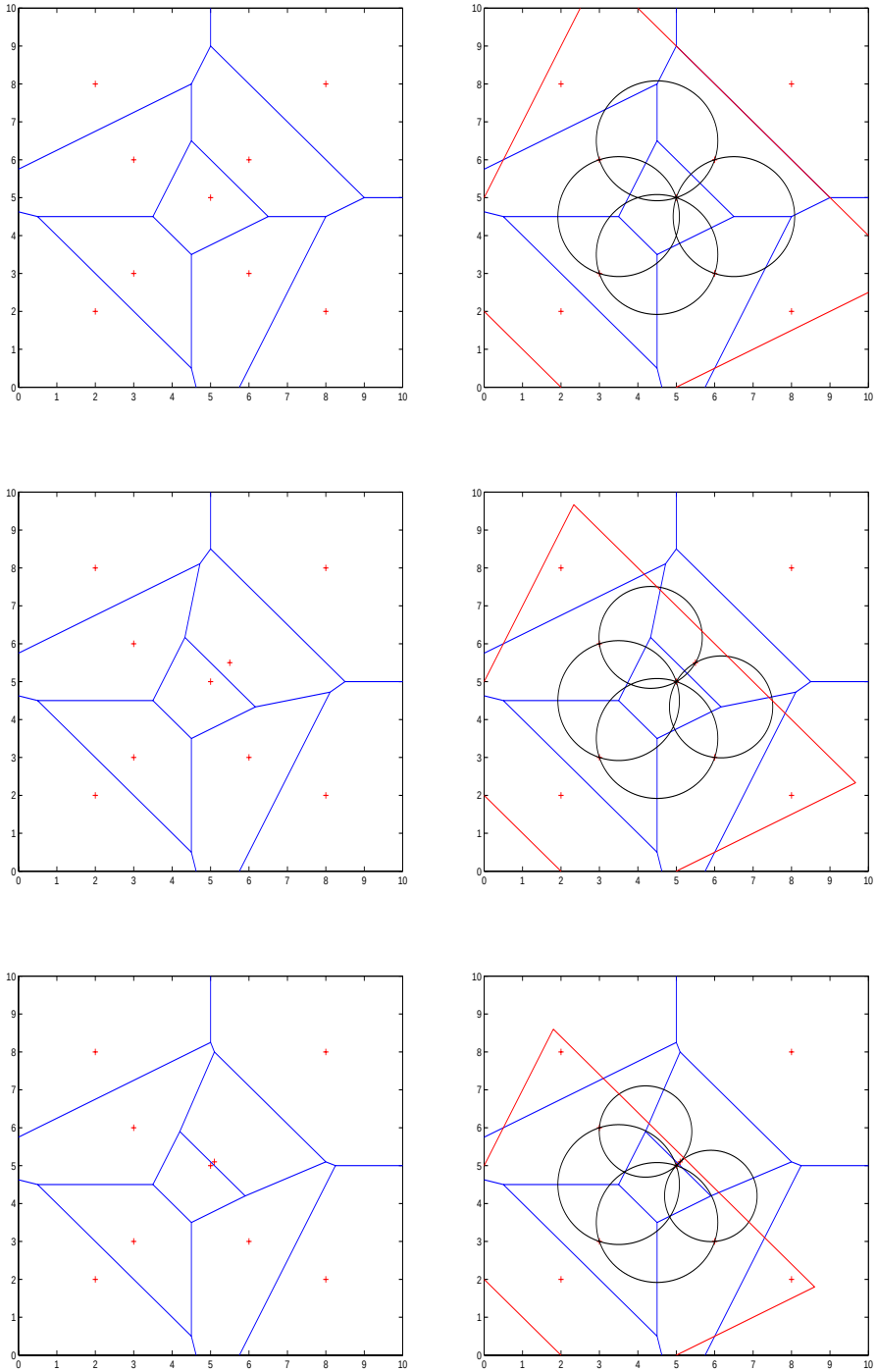

Figure 6: The clover region is not contained in any fixed multiple of the Voronoi region

Lemma 6. *In an SMT with a FST, an edge between a terminal point x_i and a Steiner point in a SMT is of length at most d_i, where d_i is the distance from x_i to the nearest other terminal point.*

For d_i as in Lemma 6, we refer to $B_{d_i}(x_i)$ as the "smallest sphere" about x_i. Note that $B_{d_i}(x_i) \subset \text{lunar}(x_i)$, and Lemma 6 immediately implies that in a SMT with a FST, two terminals x_i and x_j may be connected to a common Steiner point only if

$$||x_i - x_j|| \leq d_i + d_j = \sqrt{d_i^2 + d_j^2 + 2d_i d_j} \ .$$

We next strengthen this condition by incorporating the angle condition.

Lemma 7. *Consider terminals x_i and x_j with smallest spheres of radii d_i and d_j, respectively. Then in a SMT with a FST, x_i and x_j may be connected to a common Steiner point only if*

$$||x_i - x_j|| \leq \sqrt{d_i^2 + d_j^2 + d_i d_j}.$$

Proof. Suppose that x_i and x_j are two terminals connected to a common Steiner point s, and let $\theta = \angle x_i s x_j$. By the law of cosines

$$\begin{aligned} ||x_i - x_j||^2 &= ||x_i - s||^2 + ||s - x_j||^2 - 2||x_i - s|| \cdot ||s - x_j|| \cos\theta \\ &= ||x_i - s||^2 + ||s - x_j||^2 + ||x_i - s|| \cdot ||s - x_j||, \end{aligned}$$

since $\theta = 120^\circ$ for a SMT. The proof is completed using $||x_i - s|| \leq d_i$, $||x_j - s|| \leq d_j$, from Lemma 6. □

To extend Lemma 7 to paths between terminals that include more than one Steiner point, we need an upper bound on the lengths of edges between Steiner points. For terminals x_i and x_j in X, let b_{ij} denote the length of the longest edge on the unique path between x_i and x_j in a minimum length spanning tree on X (the *bottleneck distance*). The proof of the following lemma follows by a simple edge exchange argument.

Lemma 8. *A SMT contains no edge of length greater than b_{ij} on the unique path between x_i and x_j.*

Note that $b_{ij} \geq \max\{d_i, d_j\}$ for any terminals $\{x_i, x_j\}$. Let x_i and x_j be two terminals, with bottleneck distance b_{ij} and smallest spheres of radii d_i and d_j, respectively. We conclude from Lemmas 6 and 8 that x_i and x_j may be connected by two or fewer Steiner points only if

$$||x_i - x_j|| \leq d_i + d_j + b_{ij}.$$

Once again, we strengthen this condition by applying the angle condition.

Lemma 9. *Consider terminals x_i and x_j with bottleneck distance b_{ij} and smallest spheres of radii d_i and d_j, respectively. Then in a SMT with a FST, x_i and x_j may be connected by a path having two or fewer Steiner points only if*

$$||x_i - x_j|| \leq \sqrt{(d_i + d_j)^2 + b_{ij}^2 + (d_i + d_j) b_{ij}}.$$

Proof. Consider a path connecting x_i and x_j of the form (x_i, s_1, s_2, x_j), where the edges obey the angle condition and length restrictions in Lemmas 6 and 8. Note that increasing the length of any edge only increases the inter-terminal distance, so without loss of generality, assume all edge lengths are at their upper bounds.

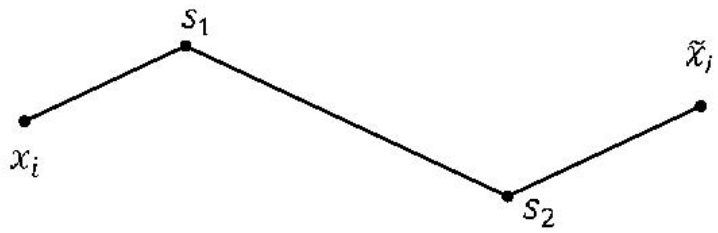

Figure 7: The configuration with two Steiner points that maximizes the inter-terminal distance

Let $(x_i, s_1, s_2, \tilde{x}_j)$ denote the configuration where the last point has been rotated into the plane defined by the first three points. There are two choices for such a point $\tilde{x}_j$ that satisfy the angle condition, and we take $\tilde{x}_j$ to be on the opposite side of the edge connecting s_1 and s_2) from x_i; see Figure 7. Since $b_{ij} \geq d_i$ and $b_{ij} \geq d_j$, the line segment between x_i and $\tilde{x}_j$ must intersect the edge connecting s_1 and s_2. Let m be this point of intersection. Then

$$\begin{aligned} ||x_i - \tilde{x}_j|| &= ||x_i - m|| + ||m - \tilde{x}_j|| \\ &= ||x_i - m|| + ||m - x_j|| \\ &\geq ||x_i - x_j||, \end{aligned}$$

where the second equality follows since $\tilde{x}_j$ and x_j are the same distance from any point on the line between s_1 and s_2. We have therefore shown that moving from the original configuration (x_i, s_1, s_2, x_j) to the planar configuration $(x_i, s_1, s_2, \tilde{x}_j)$ does not decrease the inter-terminal distance. The maximal inter-terminal distance of the planar configuration is then easily found since alternating edges in the path between x_i and $\tilde{x}_j$ are parallel. Translating the edge connecting s_2 and $\tilde{x}_j$ to be co-linear with the edge connecting x_1 and s_1 and applying the law of cosines with $\theta = 120°$, as in the proof of Lemma 7, obtains

$$||x_i - x_j|| \leq ||x_i - \tilde{x}_j|| \leq \sqrt{(d_i + d_j)^2 + b_{ij}^2 + (d_i + d_j)b_{ij}} \ .$$

□

It is natural to conjecture an extension of Lemmas 7 and 9 to paths between terminals involving more than two Steiner points, but we have been unable to prove this conjecture. (The main difficulty is extending the step in the proof of Lemma 9 that obtains a planar configuration without decreasing the inter-termianl distance.) However, for the problem sizes considered in the next section, the extension to paths containing more than 2 Steiner points rarely applies even if correct.

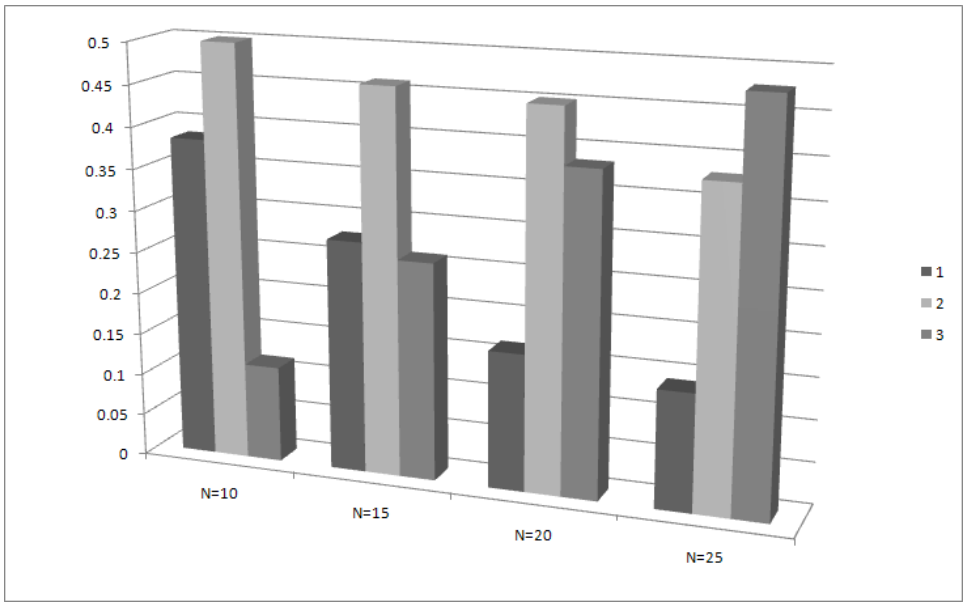

Figure 8: The distribution of the minimum number of Steiner points connecting pairs of terminals

For a set of terminal nodes X with $|X| = N$, Lemmas 6 and 9 can be used to create an $N \times N$ matrix D where each off-diagonal entry d_{ij} gives the minimum number of Steiner points $k \in \{1, 2, 3\}$ on the path from x_i to x_j in a SMT with a FST. In Figure 8 we give the distibutions for the values of the off-diagonal entries in the D matrices corresponding to $N = 10$, 15, 20 and 25 terminals randomly distributed in a unit cube in $\Re^3$. The shift in the distributions as N increases is apparent; note that for $N = 25$ almost 50% of terminal pairs $\{x_i, x_j\}$ require at least 3 Steiner nodes on the path connecting them.

4 Implementing geometric criteria to compute SMTs

In this section we describe an enhancement of Smith's algorithm for solving the Euclidean Steiner tree problem in $\Re^d$, $d \geq 2$ that incorporates the geometric restrictions described in section 3. For a set of terminal nodes X we compute the matrix D described at the end of the section in a pre-processing step.

Recall that Smith's algorithm uses an implicit enumeration scheme where nodes in the enumeration tree correspond to FSTs on a subset of the N terminal nodes. We tabulate the *deficit* for such a partial FST as follows. If two terminals x_i and x_j are connected by a path having s_{ij} Steiner points and $s_{ij} < d_{ij}$, we define the *deficit* for this path to be $m_{ij} = d_{ij} - s_{ij}$, otherwise the deficit is zero. Note that by construction, if a path has a deficit of $m_{ij} > 0$, then at least m_{ij} merges on this path will be required to satisfy the condition for the required number of Steiner points on the path connecting x_i and x_j in a SMT. (Using the restrictions derived in the previous section, if a path has a positive deficit then the possible values of the deficit are 1 and 2.) Note further that the deficit tabulation is additive over disjoint paths: if the paths between pairs of terminals $\{x_i, x_j\}$ and $\{x_k, x_l\}$ are disjoint, and require m_{ij} and m_{kl} additional merges respectively, then

a total of at least $m_{ij} + m_{kl}$ merges are required to satisfy the requirements for the minimum number of Steiner nodes on the two paths.

For a given FST on a subset of $n > 3$ of the terminal nodes, we perform a greedy set-packing of disjoint paths in an attempt to maximize the total deficit. We first choose all paths between pairs $\{x_i, x_j\}$ having $s_{ij} = 1$ and a positive deficit. (Note that for $n > 3$, paths between terminals with $s_{ij} = 1$ in a FST are all disjoint from one another.) We then greedily pack any additional disjoint paths with $s_{ij} = 2$ that have a deficit of 1 (it is not possible for a path with $s_{ij} = 2$ to have a deficit of 2 since the largest possible required number of Steiner points is $d_{ij} = 3$). Let M denote the sum of the deficits over all of the chosen paths. If $M > N - n$, the number of merges remaining, this topology and all of its descendants may be removed from consideration, with no need to compute its minimal length. We incorporate this new *fathoming by geometry* criterion in an implementation of Smith's algorithm in an attempt to improve its performance. When processing a node in the enumeration tree, we check if the partial FST can be fathomed by geometry before computing the minimal length of a tree with the given topology.

In reporting statistics below, we use "Nodes" to refer to the number partial FSTs in the enumeration tree which are not fathomed based on their parent's minimal length or geometric conditions. Each such node requires the solution of a second-order cone optimization problem using MOSEK (Andersen and Andersen, 2010) to optimize the location of Steiner points and determine the minimal length of a tree with the given topology. We use "Time" to denote the CPU seconds taken to solve an instance. All values reported in tables are averages taken over the number of instances solved. We consider two sets of problem instances. The problems from Fampa and Anstreicher (2008) are randomized instances in the hypercube $[0, 10]^d$ with $N = 10$ terminals in dimensions $d = 3$, 4, 5. There are 10 problems with $d = 3$, and 5 each for $d = 4$ and 5. For $d = 3$ we also created additional random instances with $N = 12$, 14, 16. There are 10 problems for $N = 12$ and 14, and 5 for $N = 16$. All computations were performed using a modified version of the C implementation of Smith's algorithm written by Marcia Fampa, running under Linux on a 3.2 GHz dual core Pentium with 2 GB of RAM.

The Smith+ algorithm of Fampa and Anstreicher (2008) utilizes a "strong branching" scheme to vary the order in which terminals are merged to the existing partial FST. For each node in the enumeration tree, the next terminal merged is chosen in such a way that the maximum number of children are eliminated, and/or the child bounds are increased as rapidly as possible. The strong branching scheme attempts to minimize the number of nodes processed in the implicit enumeration, but is expensive to implement due to the additional computations required to decide the next terminal to merge. As an alternative to strong branching, we sorted the terminal nodes in an attempt to accelerate fathoming without the computational effort associated with a sophisticated branching scheme. We sort the terminals via distance from their centroid, with the first terminal being the farthest away and the final terminal being the closest. The hope is that by first adding terminals that are "far apart" the length of the tree will grow more rapidly.

In Table 1 we compare the performance of the Smith+ algorithm to our implementation of Smith's

algorithm with sorting of the terminal nodes, using the instances from Fampa and Anstreicher (2008). For both algorithms, the "Nodes Factor" and "Time Factor" give the improvement in the average Nodes and Time compared to an implementation of Smith's algorithm that simply adds the terminal nodes in the order in which they appear in the input file (the factors for the Smith+ algorithm are taken from the Fampa and Anstreicher (2008)). As expected, the more sophisticated strong branching scheme reduces the nodes processed by a much larger factor, but the factors for time reduction obtained by the two approaches are comparable. We conclude that sorting the terminals is an economical way to reduce both nodes and computational time, and utilize the centroid sorting procedure for all remaining computations.

Table 1: The effect of sorting terminals on instances in $\Re^d$

		Smith		Smith w/sorting				Smith+	
Problems		Nodes	Time	Nodes	Time	Nodes	Time	Nodes	Time
N	d					Factor	Factor	Factor	Factor
10	3	139,971	248.6	39,571	69.9	3.5	3.6	50.7	4.8
10	4	368,763	766.5	60,489	120.1	6.1	6.4	26.9	2.8
10	5	470,322	1,116.0	128,485	296.3	3.7	3.8	50.8	4.4

We next consider the effect of adding fathoming by geometry. As Table 2 demonstrates, geometric conditions have a significant impact on fathoming of nodes and decreasing computation time. The factors for nodes and time in Table 2 are relative to Smith's algorithm with sorting of the terminals. The two improvement factors are approximately equal due to the minor computational effort in checking geometric criteria and performing all other work compared to solving second-order cone optimization problems using MOSEK. (The instances in $\Re^5$, for example, averaged 167.0 CPU seconds per instance with 165.9 seconds consumed by MOSEK.) Information for individual problems is given in Figures 9 and 10. In both figures the x-axis gives the number of nodes required by the Smith algorithm with sorting (on a logarithmic scale) and the y-axis gives the factor improvement in the number of nodes when fathoming by geometry is added[2]. Note that there is one problem with $d = 3$, $N = 10$ where adding fathoming by geometry *increases* the number of nodes required by the algorithm. This is possible due to the effect of obtaining an improved upper bound earlier in the enumeration process: a node that is fathomed by geometry might lead to an improved but nonoptimal Steiner tree that permits fathoming of other nodes encountered before the SMT is found. There is clearly a large variation in the number of nodes required to solve problems with a given d and N. Figure 9 shows that for the problems with $N = 10$, the improvement from adding geometry tends to be larger for more difficult problems. This trend is also apparent for the problems with $d = 3$ in Figure 10, up to those requiring approximately 1 million nodes. Beyond this level of difficulty, however, the marginal effect of adding fathoming by geometry appears to diminish. We conclude that adding geometric criteria to Smith's algorithm provides significant improvements, but stronger geometric restrictions are required to obtain the same degree of improvement on more difficult problems.

[2]Two instances with $N = 12$, $d = 3$ have almost identical coordinates in Figure 10 and appear as a single point.

Table 2: The effect of geometric conditions on instances in $\Re^d$

Problems		Smith w/sorting		Smith w/sorting & geometry			
		Nodes	Time	Nodes	Time	Nodes	Time
N	d					Factor	Factor
10	3	39,571	69.9	22,766	40.2	1.7	1.7
10	4	60,489	120.1	44,858	89.9	1.3	1.3
10	5	128,485	296.3	72,850	167.0	1.8	1.8
12	3	323,829	613.3	191,319	362.7	1.7	1.7
14	3	3,053,659	6,239.0	2,284,117	4,681.0	1.3	1.3
16	3	21,114,698	48,350.0	15,849,345	36,200.0	1.3	1.3

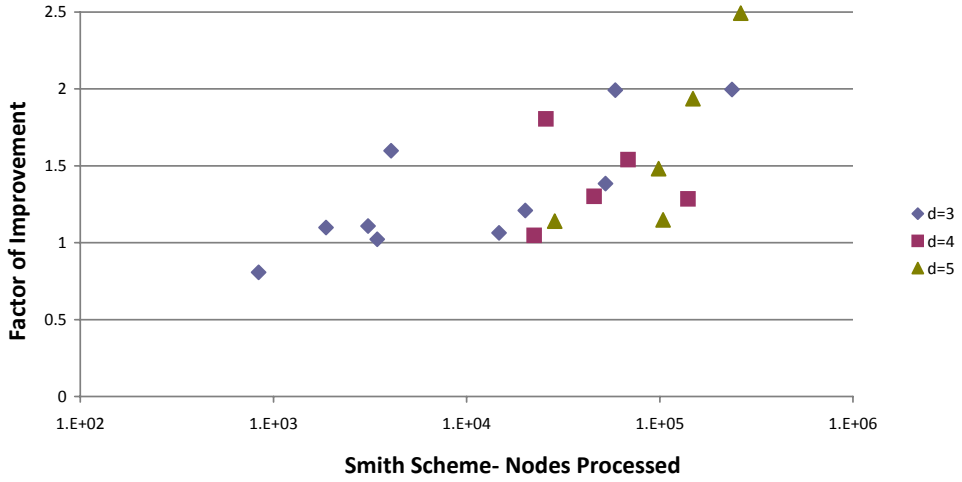

Figure 9: Effect of adding geometry to problems with $N = 10$

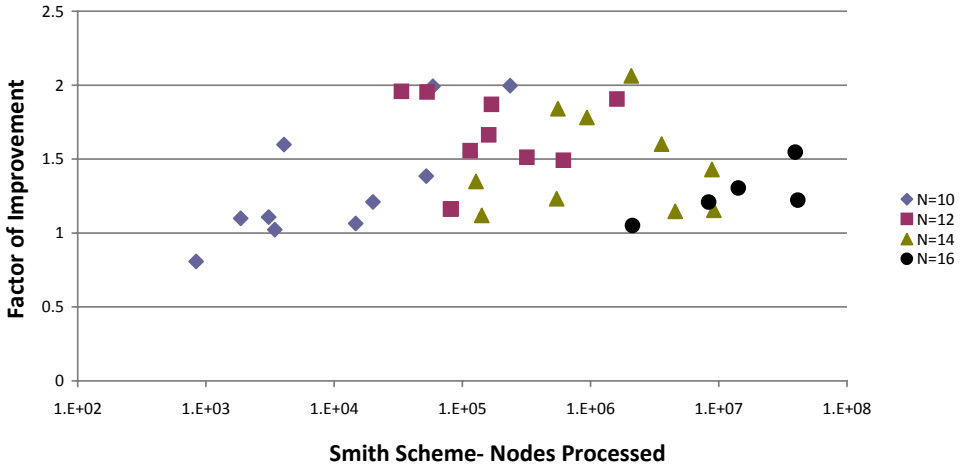

Figure 10: Effect of adding geometry to problems with $d = 3$

There are several ways in which the performance of Smith's algorithm might be further improved. We first describe two possiblities that we have investigated.

- For the computational results reported here, the algorithm is initialized with an upper bound of $+\infty$, so no fathoming is possible until an initial Steiner tree on all terminal nodes is found by the algorithm. The performance of the algorithm can be improved by running a heuristic that obtains an initial, hopefully near-optimal solution. We investigated the effect of having an initial upper bound (IUB) by using the heuristic from Van Laarhoven and Ohlmann (2010) to obtain a good initial solution. Using an IUB improves the performance of the algorithm with and without the use of fathoming by geometry. Compared to the algorithm with sorting of the terminals, but without an IUB, adding both fathoming by geometry and an IUB results in improvement factors for nodes and time in the range $[1.4, 2.2]$, compared to $[1.3, 1.8]$ as reported in Table 2. (The use of an IUB also eliminates the single instance where fathoming by geometry increased the number of nodes.) When using an IUB, the marginal improvement factors obtained by adding fathoming by geometry are very close to those reported without the use of an IUB in Table 2.

- As described in section 3, it is natural to conjecture that the results of Lemmas 7 and 9 extend to paths between terminals involving more than two Steiner points. To examine the effect of such an extension, we assumed the result was true and considered the possibility of obtaining off-diagonal entries of the matrix D great than 3. For the test problems used here, no instances with $N = 10$ or 12 obtain any values $d_{ij} > 3$, and a small number of such entries appear in problems with $N = 14$ and 16. When these latter instances were re-run with the revised D matrices, the change in the number of nodes required was very small.

Additional possibilites for improvements that remain topics for further research include the following.

- In attempting to fathom by geometry, we currently use a greedy procedure to pack disjoint paths having a positive deficit in an attempt to maximize the total deficit for a FST on a subset of the terminal nodes. A more sophisticated packing procedure could obtain a higher total deficit, leading to more fathoming. The use of a more sophisticated procedure could be especially beneficial if there were longer paths with positive deficits, as would result for N sufficiently large from the extension of Lemmas 7 and 9 to paths with more than two Steiner nodes mentioned above.

- In addition to fathoming, the geometric conditions could be used to alter the branching process of Smith's algorithm. For example, consider a node at level k of Smith's enumeration scheme, having a FST on $n = k + 3$ terminals, where some path between a pair of terminals $\{x_i, x_j\}$ connected to a common Steiner point has a positive deficit. A merge must occur on one of the two edges in this path if it is to lead to a SMT, so one could create children of this node by applying the merge operation using each of the remaining terminals merged to each of the two edges, resulting in a total of $2(N - n)$

children. The usual branching process for Smith's enumeration scheme creates one child for each of the $2n-3$ edges in the partial FST. It follows that if $n > \lceil N/2 \rceil$, then fewer children are created by the alternative branching scheme, and the difference between the two schemes increases with depth in the enumeration tree.

Acknowledgement

We are grateful to Marcia Fampa for providing the C implementation of Smith's algorithm on which our computations are based.

References

E.D. Andersen and K.D. Andersen. The MOSEK optimization tools manual Version 6.0. 2010. URL `www.mosek.com`.

S. Arora. Polynomial time approximation schemes for Euclidean traveling salesman and other geometric problems. *JACM*, 45(5):753–782, 1998.

M. De Berg, O. Cheong, M Van Kreveld, and M. Overmars. *Computational Geometry: Algorithms and Applications*. Springer-Verlag, 2008.

M. Fampa and K.M. Anstreicher. An improved algorithm for computing Steiner minimal trees in Euclidean d-space. *Discrete Optimization*, 5:530–540, 2008.

M.R. Garey, R.L. Graham, and D.S. Johnson. The complexity of computing Steiner minimal trees. *SIAM J. Applied Mathematics*, 32(4):835–859, 1977.

E.N. Gilbert and H.O. Pollak. Steiner minimal trees. *SIAM J. Applied Mathematics*, 16(1):1–29, 1968.

J.M. Smith and B. Toppur. Euclidean Steiner minimal trees, minimum energy configurations, and the embedding problem of weighted graphs in E^3. *Discrete Applied Mathematics*, 71:187–215, 1996.

W.D. Smith. How to find Steiner minimal trees in Euclidean d-space. *Algorithmica*, 7(1):137–177, 1992.

J.W. Van Laarhoven and J. Ohlmann. A randomized Delaunay triangulation heuristic for the Euclidean Steiner tree problem in $\Re^d$. *J. Heuristics*, to appear, 2010.

D.M. Warme, P. Winter, and M. Zachariasen. GeoSteiner 3.1., 2001. URL `www.diku.dk/hjemmesider/ansatte/martinz/geosteiner/`.

P. Winter and M. Zachariasen. Euclidean Steiner minimum trees: An improved exact algorithm. *Networks*, 30(3):149–166, 1997.