Minimal Length Tree Networks on the Unit Sphere

J. Dolan
R. Weiss
J. MacGregor Smith

COINS TR 89-105

October 1989

This work is supported by the Defense Advanced Research Projects Agency under contract number DACA76-89-C-0017, and by the National Science Foundation under grant DCR 8500332.

MINIMAL LENGTH TREE NETWORKS ON THE UNIT SPHERE

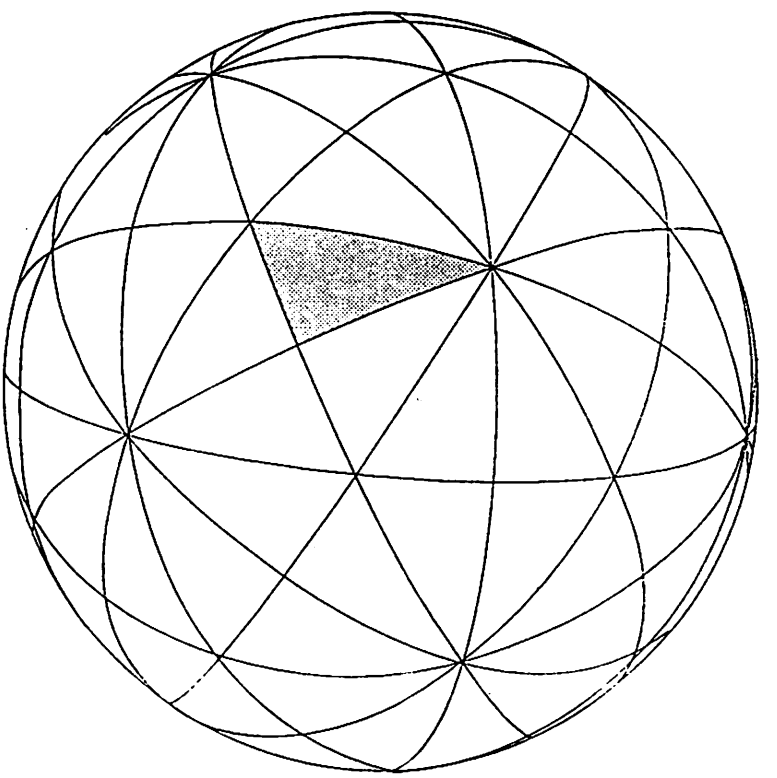
John Dolan
COINS Department

Richard Weiss
COINS Department
University of Massachusetts
Amherst, MA 01003
USA

J. MacGregor Smith
IEOR Department

Abstract

In this paper, we examine the problem of finding minimal length tree networks on the unit sphere Φ of a given point set ($\mathcal{V}$) where distance is measured along arcs of great circle. The related problems of finding a Steiner Minimal Tree $SMT(\mathcal{V})$ and of finding a Minimum Spanning Tree $MST(\mathcal{V})$ are treated through a simplicial decomposition technique based on computing the Delaunay Triangulation $DT(\mathcal{V})$ and the Voronoi Diagram $VD(\mathcal{V})$ of the given point set. The design and analysis of $O(N \log N)$ algorithms for computing $DT(\mathcal{V})$, $VD(\mathcal{V})$, and $MST(\mathcal{V})$ as well as an $O(N \log N)$ heuristic for finding a sub-optimal $SMT(\mathcal{V})$ solution are presented together with computational experience for randomly distributed points on Φ.



A. Introduction

In this paper, we are concerned with the problem of determining minimal length tree networks on the sphere of radius 1, the *unit sphere* Φ. We will treat both the Minimum Spanning Tree (*MST*) problem as well as the Steiner Minimal Tree (*SMT*) problem on the Φ. We are interested in the computational complexity of these problems and we generally examine the development of algorithms and heuristics for these problems from a Computational Geometry point of view [PREP85].

1. Problem Definition

The general problem of determining a minimal length network has its origins in the plane E^2 with the famous Euclidean Steiner Minimal Tree (*ESMT*) problem [COUR41; MELZ61; COCK68; GILB68; COCK69; CHUN76,78; SMIT79;81; DU82; WINT87; HWAN89] in which one wishes to construct the minimal length network interconnecting a given set of points defined by their Cartesian coordinates (x_i, y_i). We shall provide the acronyms *GMST* and *GSMT* for the respective problems on Φ because we will measure distance with geodesics. More formally we have:

- **GSMT Problem:** Given points on Φ from a set $(\mathcal{V}) = \{v_1, v_2, \ldots, v_n\}$ we wish to construct a tree interconnecting $(\mathcal{V})$ with possible additional points from a set $\mathcal{S} = \{s_1, s_2, \ldots, s_m\}$ where necessary in order to achieve the minimal length possible.

If no points from the additional set $\mathcal{S}$ are utilized, then the *GSMT* becomes a Geodesic Minimum Spanning Tree (*GMST*). The counterparts of our problem in E^2 have similar functional relationships.

For our problem on the sphere, locations for the given points are defined by their latitude and longitude *viz.* (ϕ_i, θ_i) for $i = 1, 2, \ldots, n$. In order to discuss the minimal networks on the sphere, one needs a metric for length. The standard metric, which corresponds to the L_2 metric in Euclidean space, is the great circle distance. A *great circle* is any circle having a radius equal to that of the sphere. The distance between a pair of points on the sphere is the length (in the L_2 sense) of the lesser arc of a great circle between the points.

Figure 1 is illustrative of the type of minimization problem we are trying to solve. We would like to find the additional point $s_i(\phi_i, \theta_i)$, if it exists, where the sum of the geodesic distances is minimized [ALY79;LITW80;DREZ81;WESO82;LOVE88]. For three given points on Φ, we have the following unconstrained optimization problem:

$$Minimize\ Z(X) = \sum_{j=1}^{3} D(X, v_j) \tag{1}$$

where $X = (x_1, x_2)$ are the latitude and longitude coordinates of the Steiner point and $D(X, v_j)$ is the shortest distance measured (in radians) on the surface of the sphere between the Steiner point X and an existing point j located at $v_j = (\phi_j, \theta_j)$. In fact, the distance of the shortest arcs are given by [DONN45]:

$$D(X, v_j) = \cos^{-1}\left[\cos x_1 \cos \phi_j \cos(x_2 - \theta_j) + \sin x_1 \phi_j\right] \tag{2}$$

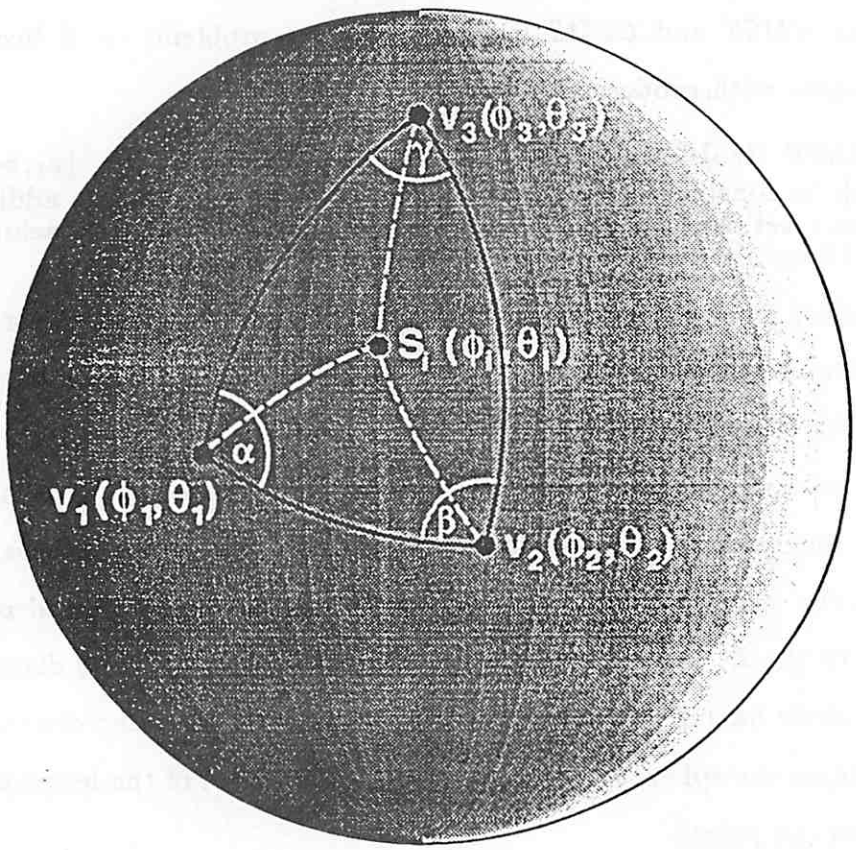


Figure 1: Steiner Minimal Tree in an Euler Triangle

2. Applications

As might be inferred from examination of Figure 1, there are a host of applications for the *GMST* and *GSMT* problems. Originally, the *EMST* and *ESMT* problems were essential in setting of long distance telephone rates [PREP85; BERN89], so it is natural to infer that efficient procedures for the *GMST* and *GSMT* problems would also be useful in terrestrial tele-communications systems. Also, satellite location and geo-communication system should naturally require *GMST* and *GSMT* topologies [MOTW87].

We are interested in developing an efficient algorithm for the *GMST* problem and a corresponding heuristic for the *GSMT* problem. In our approach to the problem, we seek to explore the use of efficient geometric decompositions of our problem so that optimal and sub-optimal solutions to the *GMST* and *GSMT* can be developed respectively. For the *EMST* and *ESMT* problems, the Voronoi Diagram and Delaunay Triangulation [PREP85] were crucial to the development of algorithms and heuristics in E^2 [SMIT81]. Correspondingly, we need the machinery of the Voronoi Diagram and Delaunay Triangulation on the sphere to effectuate our approach.

3. History and Background

The interest in *MST* and *SMT* problems has a rather long and convoluted history [HWAN89]. Polynomial running time algorithms have existed for the *EMST* problem for three decades, yet algorithms for our general network design problems on the sphere have remained elusive. It is interesting to note that the optimal running time algorithm $\Omega(N \log N)$ for the geometric *EMST* problem [SHAM78] forms the foundation for our *GMST* problem on Φ. Efficient polynomial running time algorithms for three and higher dimensional *MST* 's have also proved difficult, although recently $O(N \log N)$ heuristics for the *MST* problem in E^d space have become available [VAID88].

The NP-Completeness of the *ESMT* problem in E^2 was demonstrated by [GARE77;79]. By a simple point set transformation onto Φ, our *GSMT* problem is also NP-Complete.

Since 1972, a number of heuristics have been developed which seek to calculate near-optimal solutions in polynomial time for the *ESMT* problem [CHAN72;SMIT81]. Algorithms for special cases have also been examined. The development of algorithms and heuristics for our *GSMT* problem remain to our knowledge unexplored.

The key to the implementation of efficient algorithms and heuristics for our *GMST* and *GSMT* problems lies in developing geometric data structures which are simpler to compute than directly trying to compute *GMST* and *GSMT* on Φ without the structure, but which correspondingly facilitate the efficient solution to our larger problems. As we shall see, the simplicial decomposition embodied in the Voronoi Diagram and Delaunay Triangulation are critical to our venture.

B. Notation, Definitions, & Assumptions

Before we begin the remaining part of the paper, let us establish our notation, definitions and common assumptions.

- $\Phi := S^2$ the unit sphere in R^3
- $\mathcal{V}$:= our given point set on Φ
- $\mathcal{S}$:= set of Steiner points on Φ
- $\mathcal{P}$:= pointset in R^2
- $(\vec{X}_e, \vec{Y}_e, \vec{N})$:= 3D cartesian coordinate system defined on Φ, where $\vec{N}$ is the north pole and $\vec{X}_e, \vec{Y}_e$ are respectively the x and y axes in the corresponding equatorial plane
- Θ_N := the northern hemisphere of Φ defined by $\vec{N}$
- Θ_S := the corresponding southern hemisphere of Φ defined by $\vec{N}$—by convention, the equator is in Θ_S
- γ_N := the projection plane of Θ_N defined by $(\vec{X}_e, \vec{Y}_e, \vec{N})$
- γ_S := the corresponding projection plane of Θ_S defined by $(\vec{X}_e, \vec{Y}_e, \vec{N})$
- Γ_N := the image of $\mathcal{V} \cap \Theta_N$ under stereographic projection
- Γ_S := the image of $\mathcal{V} \cap \Theta_S$ under stereographic projection
- $L_M(\mathcal{V})$:= the length of the GMST of $\mathcal{V}$
- $L_S(\mathcal{V})$:= the length of the GSMT calculated for $\mathcal{V}$
- $\rho = L_S(\mathcal{V})/L_M(\mathcal{V})$ — the Steiner ratio

Definitions

- *Spherical Triangles* Points A, B, C on sphere, which do not all lie on one great circle and of which no two are diametrically opposite (antipodes), determine three great circles. These circles intersect in A, B, C and the diametrically opposite points A', B', C' thus partitioning the surface of the sphere into eight triangular regions called spherical triangles.
- *Euler Triangles* are spherical triangles in which no side is greater than π. In general, we are concerned only with Euler triangles and for these it is known that the sum of the angles is between π and 3π and the sum of the sides is between 0 and 2π. The sum of any two angles is less than the third plus π. The sum of two sides is greater than the third; the difference of two sides is smaller than the third. The greater angle is opposite the greater side.
- *Convex Set on* Φ*:* Every two points in the set have at least one shortest arc joining them that lies entirely within the set.
- *Spherical Cap:* A plane intersecting a sphere and not tangent to the sphere divides the spherical surface into two regions termed spherical caps. In general, it is the smaller of these two regions that is intended when we use the term spherical cap.
- *Spherical Disk:* The planar surface of intersection of a plane and a solid sphere.
- *Bisector:* The great circle that is the locus of points between two given vertices v_i and v_j on Φ which is equidistant between them:

$$B_D(v_i, v_j) = (z \in \Phi \mid d_D(z, v_i) = d_D(z, v_j)) \tag{3}$$

- *Half-Space:* This is the locus of points closer to v_i than to another v_j:

$$H_D(v_i, v_j) = \{x \in \Phi \mid d(x, v_i) \leq d(x, v_j); j \neq i\} \tag{4}$$

- *Voronoi Polygon* Given a set of points $\mathcal{V}$ on the sphere, the *Voronoi polygon* of a point is the set of points closer to that point than any other point. This set can also be described as an intersection of half-spaces.

$$VP(v_i) = \cap_{i \neq j}^{n} H_D(v_i, v_j) \tag{5}$$

- *Voronoi diagram:* is the collection of the Voronoi polygons. The intersection of two adjacent Voronoi polygons is a *Voronoi edge*, and the Voronoi edges intersect in Voronoi vertices of exact degree 3.
- *Delaunay diagram:* is also a graph on the surface of the sphere and is dual to the Voronoi diagram; i.e. the vertices of the Delaunay diagram are the points of $\mathcal{V}$, the edges are segments of great circles orthogonal to the edges of the Voronoi diagram. In addition, the edges of the Delaunay diagram can be characterized by the property: there is a Delaunay edge through two points if and only if there is a sphere in R^3 through those points not containing any other points in $\mathcal{V}$.

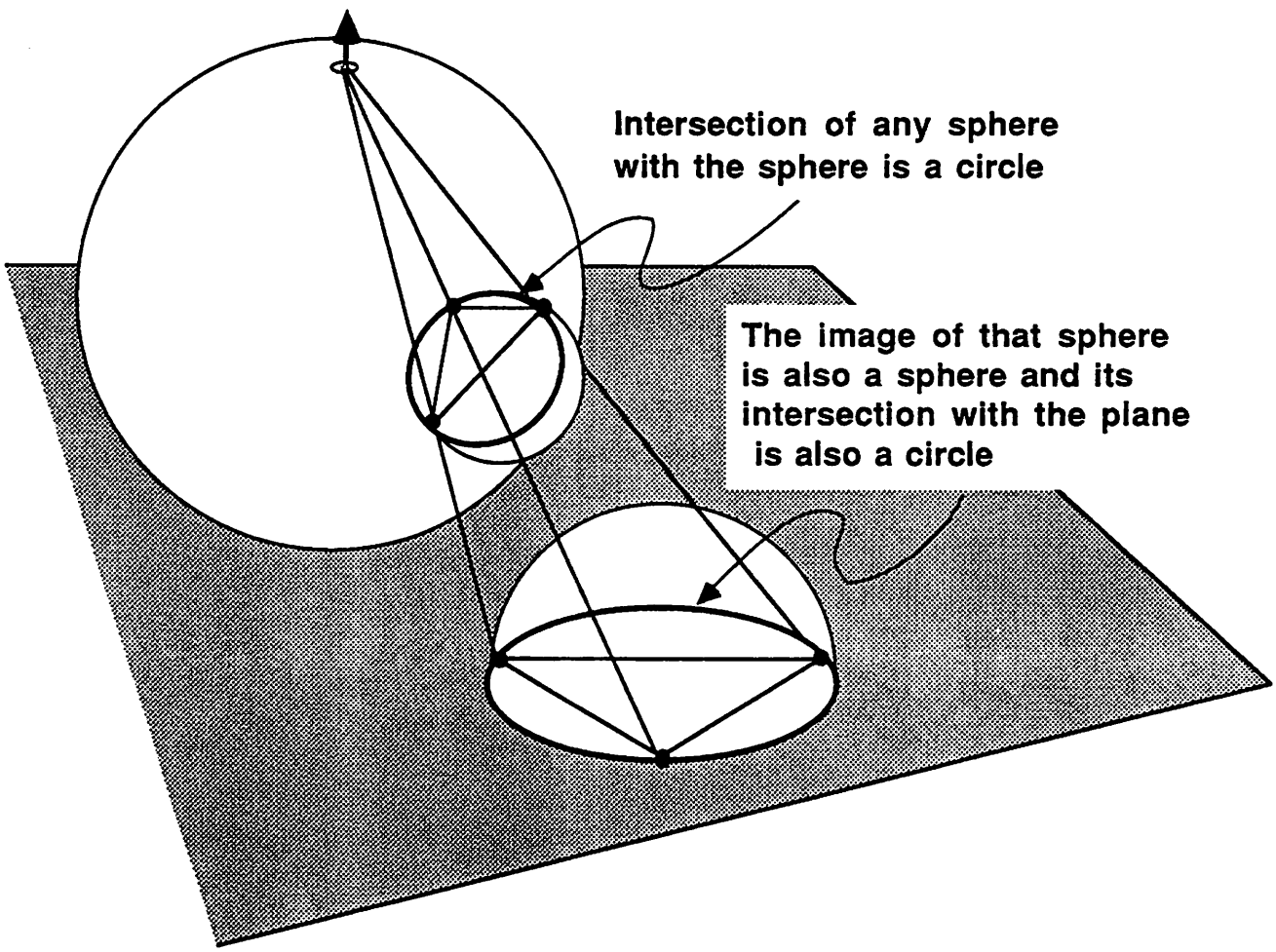


Figure 2 The intersection of any sphere with Φ is a circle whose plane partitions the surface of Φ into two spherical caps. The image of that circle under stereographic projection is a circle *in* the plane and the image of the spherical cap, not containing the pole, is the interior region of the planar circle.

C. Mathematical Properties

We need to array the known properties which are helpful in construction of both the algorithms and heuristics.

1. Proximity Properties

In constructing the Delaunay Triangluation for pointsets on the sphere $DT(\mathcal{V})$ we employ stereographic projection so that much of the triangulation may be accomplished in the plane using known techniques [GUIB85]. The principal feature of stereographic projection, of use for our constructions on the sphere, is its circle-preserving nature — i.e., circles on the sphere, not passing through the pole, map to circles in the plane and vice-versa. Moreover, such a circle on the sphere defines two spherical caps — one containing the pole, the other not — which map respectively to the exterior and interior regions defined by the corresponding planar circle. See Figure 2. For further discussion and proofs see [HILB83.] Thus, testing whether or not a point is interior to a spherical cap is exactly equivalent to testing whether or not its stereographic image is interior to the corresponding circular disk in the projection plane. Thus, we arrive at the following:

Lemma 1. For a pointset in a spherical region not containing the pole, the connectivity of the Delaunay Triangulation constructed in the plane on the stereographic image of the pointset is identical with the connectivity of the Delaunay Triangulation constructed on the points themselves within the spherical region.

Proof. In the plane, an edge e_{ij} between two points $p_i, p_j \in \mathcal{P}$ is admitted to the Delaunay Triangulation of $\mathcal{P}$: $DT(\mathcal{P})$ iff there exists a circle passing through the p_i, p_j, containing no other point of $\mathcal{P}$. Likewise on the sphere Φ, a geodesic edge g_{ij} between two points $v_i, v_j \in \mathcal{V}$ is admitted to the Delaunay Triangulation of $\mathcal{V}$: $DT(\mathcal{V})$ iff there exists a sphere passing through v_i, v_j, containing no other point of $\mathcal{V}$. Since the intersection of such a sphere with Φ is a circle passing through v_i, v_j and defining spherical caps on Φ, g_{ij} is admitted to $DT(\mathcal{V})$ iff there is a point-free spherical cap defined by v_i, v_j. Defining p_i, p_j as the stereographic images of v_i, v_j, it is obvious that $g_{ij} \in DT(\mathcal{V})$ iff $e_{ij} \in DT(\mathcal{P})$.

For our algorithm, we require a second property of the Delaunay Triangulation of pointsets on the sphere—i.e., the equivalence of $DT(\mathcal{V})$ with the 3-dimensional convex hull of $\mathcal{V}$: $CH^3(\mathcal{V})$. If, instead of arcs of great circles, the edges of $DT(\mathcal{V})$ are taken as the spherical chords connecting the points of $\mathcal{V}$ then $DT(\mathcal{V}) \equiv CH^3(\mathcal{V})$. This is easily shown by two facts:

1) There are no points other than those in $\mathcal{V}$.

2) All $v_i \in \mathcal{V}$ are equidistant from the center of the sphere—i.e, every v_i is an extremum and therefore every $v_i \in CH^3(\mathcal{V})$.

A more formal proof is available in [MOTW87]. This equivalence guarantees that the final merge step of the algorithm begins with a Delaunay edge.

2. Geodesic Minimum Spanning Tree

For a set of points in the plane, the Euclidean minimal spanning tree or EMST is a tree consisting of straight lines connecting all of the points, such that the sum of the arc lengths is minimum. This can be found in $O(N \log N)$ time, and the EMST is a subgraph of the Delaunay triangulation associated with the set of points. Similarly, the *GMST* is a subgraph of the Delaunay triangulation on the sphere.

One can use Prim's [PRIM57] algorithm for constructing minimum spanning tree. It starts with a set of points and maintains a forest of trees, merging each tree with another using the lowest-weight edge. This algorithm is valid whether the weights come from the Euclidean distance or the geodesic distance on the sphere. The algorithm is based on the following lemmas [PREP85].

Lemma 2. Let G = (V,E), be a graph with weighted edges and let $\{V_1, V_2\}$ be a partition of the set V. There is a minimum spanning tree of V which contains the shortest edge connecting V_1 to V_2.

Proof: see [PRIM57;PREP85].

Lemma 3. Let S be a set of points on the sphere Φ and let $\{S_1, S_2\}$ be a partition of S. The shortest segment (geodesic) between a point of S_1 and a point of S_2 belongs to the Delaunay triangulation.

Proof (see Preparata and Shamos for details): It follows from the definition of the Voronoi diagram that Voronoi polygons are the regions defined by bisecting

planes through the origin, or equivalently, bisecting great circles. In either case, the Delaunay triangulation is dual to the Voronoi diagram in the sense that the bisector of an arc between two points contains a Voronoi edge of either point if and only if the arc is in the Delaunay triangulation. This fact together with the triangle inequality can be used to give a proof by contradiction.

Note that the properties which were used in the last proof hold in general for any Reimann surface. Combining these two lemmas, it follows that the *GMST* on the sphere is a subgraph of the Delaunay triangulation. In addition, since any graph on the sphere can be projected to a planar graph, the number of edges in the Delaunay triangulation is still only O(N). Cheriton and Tarjan showed that the *EMST* can be computed from the Delaunay triangulation in the plane in linear time, and this proof can be extended to the sphere as well.

Steiner Points in a spherical $\triangle$

Let us present some of the known properties of Steiner points in spherical triangles that have already been developed and form the foundation for our heuristic search for Steiner trees on Φ.

- **Theorem 1 [GREE71;COCK72]:** *Let v_1, v_2, v_3 be three distinct points on the surface of a sphere, not all of which are on the same great circle, then any point X which minimizes the sum of the minor great circle arcs as in* (1)*:*

$$\sum_{j=1}^{3} D(X, v_j) \in \triangle v_1 v_2 v_3$$

is a Steiner point.

- **Theorem 2 [COCK72]:** *Either there is a unique point X which minimizes $\sum_{j=1}^{3} D(X, v_j)$ or X coincides with one of the given points v_1, v_2, v_3 or if*

$$\angle v_1 v_2 v_3 of \triangle v_1 v_2 v_3 < 2\pi/3$$

then the minimum point of $v_1 v_2 v_3 \neq v_2$.

Theorem 1 and 2 provide the analogues for locating Steiner points on Φ as they occur in E^2. Also, under an azimuthal equidistant projection, with a Steiner point as

a pole, an optimum Steiner point for three given points on Φ, is the optimum for the corresponding problem in the plane E^2 [LITW80]. In fact, it is this local projection property for three and four points on Φ that we wish to explore in our development of an heuristic for the general Steiner problem on Φ.

Finally, there are numerical difficulties with directly solving for the Steiner points on Φ as a continuous nonlinear optimization problem as can be surmised from the trigonometric relations of our distance function [LOVE88]. In fact, in problems where the given fixed points cannot be covered by a disc of radius $\frac{\pi}{4}$, local minima, maxima and flat points must be dealt with [DREZ78]. As an alternative to solving for the *GSMT* as a continuous nonlinear optimization problem, we now present our computational geometry approach.

D. Algorithms and Heuristics

Before commencing a detailed description of the algorithms themselves, a brief discussion of our primitive data structures is in order. Spherical points (all $v_i \in \mathcal{V}$) are represented simultaneously in spherical coordinates (ϕ_i, θ_i), 3D cartesian coordinates $(\vec{X}_{e_i}, \vec{Y}_{e_i}, \vec{N}_i)$, and 2D cartesian coordinates of the projected image $(\vec{X}_{\gamma_i}, \vec{Y}_{\gamma_i})$. The first of these is useful for calculations on the sphere, the second for partitioning and projecting pointsets, the third for calculating the planar Delaunay Triangulations. Another important data structure is a variant of the quad-edge, devised by [GUIB85]. This structure permits simultaneous representation of the Delaunay and Voronoi Diagrams, and with appropriate markings the desired networks *GMST* and *GSMT* as well. This representation is edge- rather than point-based; and edges are pairs of directed arcs, a fact which permits diagrams to be oriented.

The general outline for this section is to present our simplicial decomposition of the points on Φ through the Delaunay triangulation, then the algorithm for the *GMST*, and finally the heuristic for the *GSMT*.

$O(N \log N)$ Algorithm for Delaunay Triangulation

The following is a description of our algorithm for the Delaunay Triangulation of a set of points $\mathcal{V}$ on the sphere (Φ) where $|\mathcal{V}| \geq 4$

Step 1.0 *Delaunay Triangulation*

Step 1.1: Determine Poles of Projection $\vec{N}, \vec{S}$, so that the hemispheres defined by the corresponding equatorial plane each contain at least 2 points. Also define equatorial reference vectors $\vec{X_e}, \vec{Y_e}$ so that a new coordinate system (in terms of $\vec{X_e}, \vec{Y_e}, \vec{N}$) is established for the sphere. This partitions the pointset by the equatorial plane . $O(1)$ [See Figure 3]

Step 1.2 Stereographically project points of each hemisphere $\ominus_N, \ominus_S$ to the respective projection plane γ_N, γ_S. $O(N)$ [See Figure 4]

Step 1.3 For each projection plane, sort the projected points Γ_N, Γ_S lexicographically and then perform the planar Delaunay procedure [GUIB85]$O(N \log N)$

Step 1.4 Perform equatorial merge of triangulations of both hemispheres using the following *incap predicate*: [See Figures 5& 6]

$$\begin{vmatrix} A_{x_e} & A_{y_e} & A_n & 1 \\ B_{x_e} & B_{y_e} & B_n & 1 \\ C_{x_e} & C_{y_e} & C_n & 1 \\ D_{x_e} & D_{y_e} & D_n & 1 \end{vmatrix} < 0$$

This predicate returns TRUE iff a point $D \in \mathcal{V}$ is interior to the spherical cap defined by points $A, B, C \in \mathcal{V}$. Thus, spherical triangle $\widehat{\triangle} ABC$ is Delaunay iff the predicate is FALSE for all $D \in \mathcal{V}$. $O(N)$

The partitioning of the $\mathcal{V}$, Step 1.1, is desirable for two reasons. The first has to do with numerical stability. Under stereographic projection, points near the pole project to positions whose distance is near infinity. By projecting each hemisphere independently, the maximum distance of projection is realized by points on the equator and is bounded by the spherical diameter. The second reason has to do with uniformity of treatment. By ensuring that each hemisphere has at least 2 points, the final merge step requires no special case handling. The partitioning of the pointset, Step 1.1, uses any 4 points of $\mathcal{V}$ and thus does not depend on the number of points (N). It is therefore is O(1).

The projection and planar delaunay steps, Steps 1.2 and 1.3, are straight-forward. Step 1.2 is clearly O(N) since each point is projected; and Step 1.3, which uses a known O(N log N) algorithm [GUIB85] on the projected pointsets of each hemisphere, must itself be O(N log N).

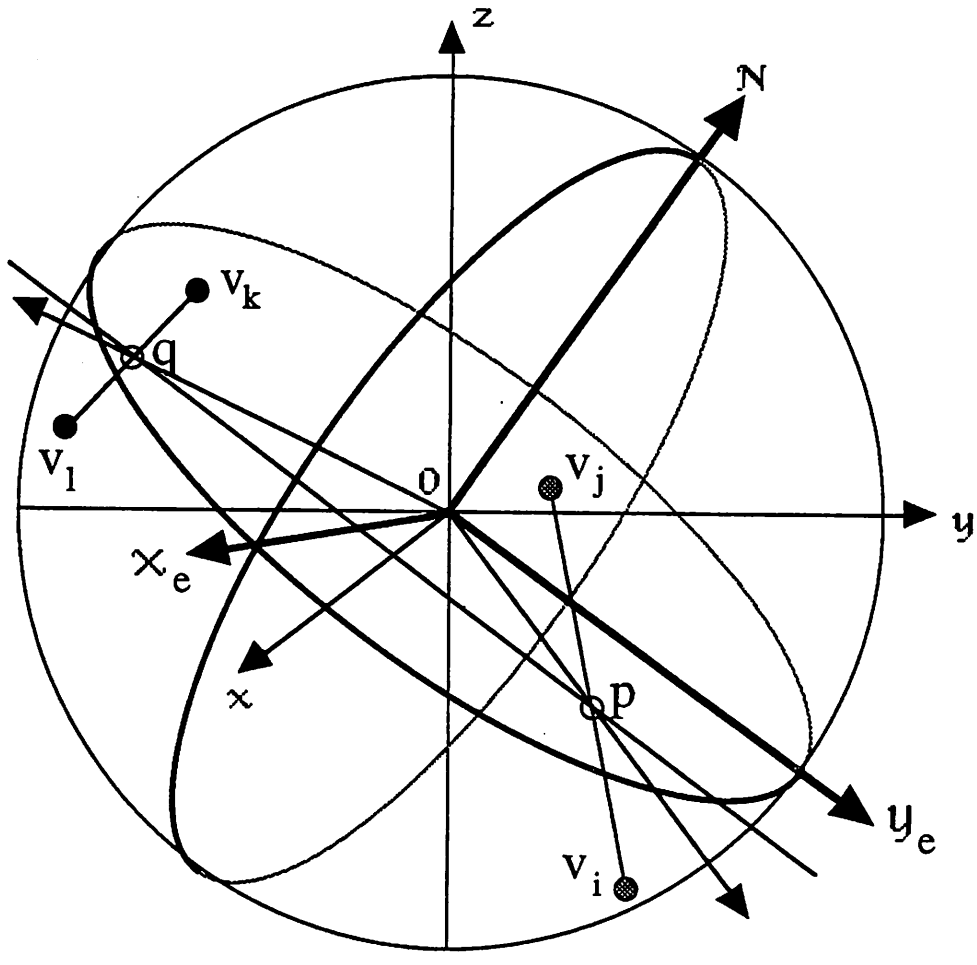


Figure 3 The pointset is partitioned in such a way that each hemisphere contains at least two points. This is accomplished by selecting any 4 points of V and finding an equatorial plane that partitions them. The equatorial plane yields a new coordinate system (X_e, Y_e, N). For example, select point p along line $v_i v_j$ and point q along line $v_k v_l$ such that line pq does not contain the origin. Then N is given by the normalized cross-product of p and q.

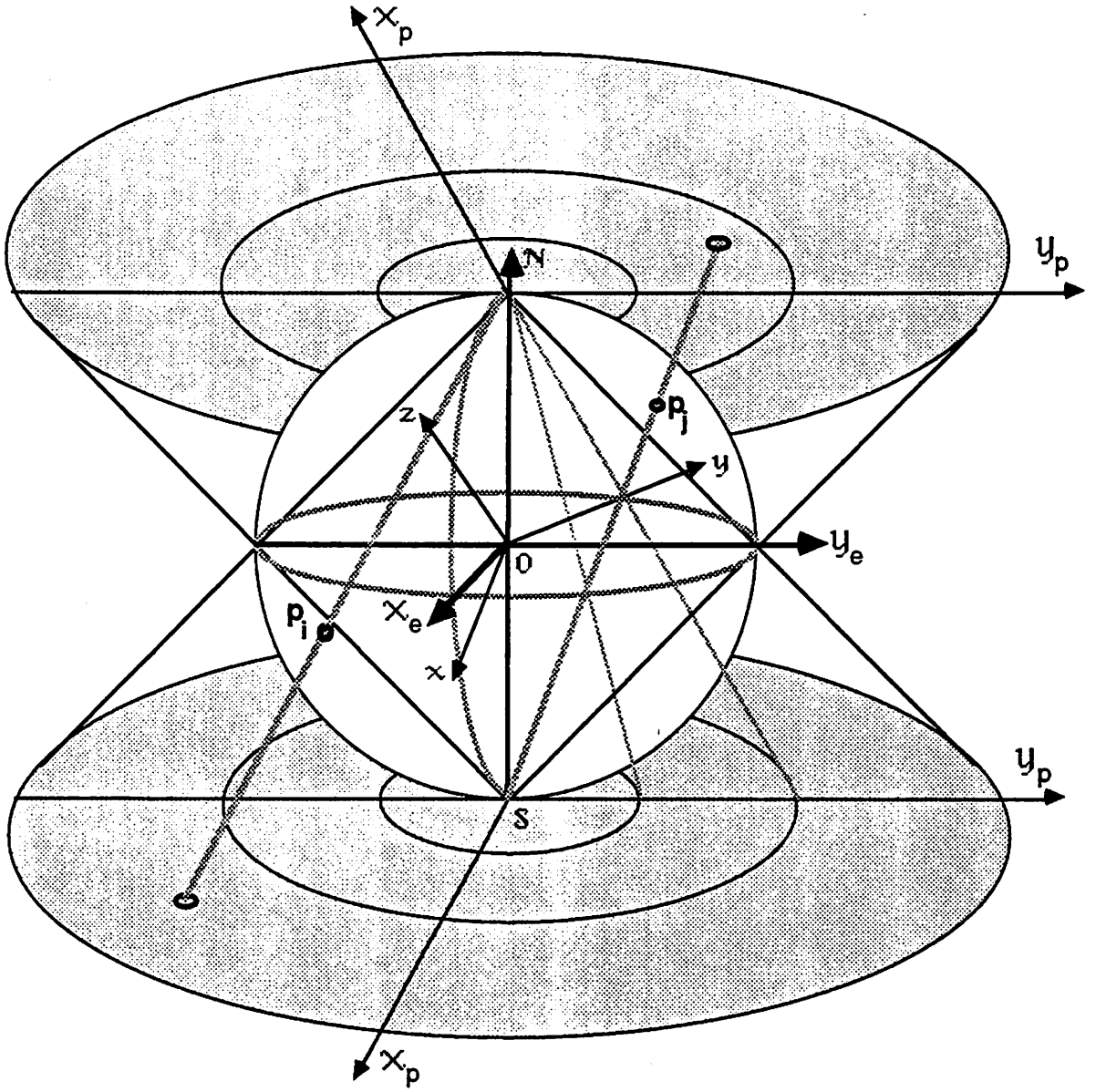
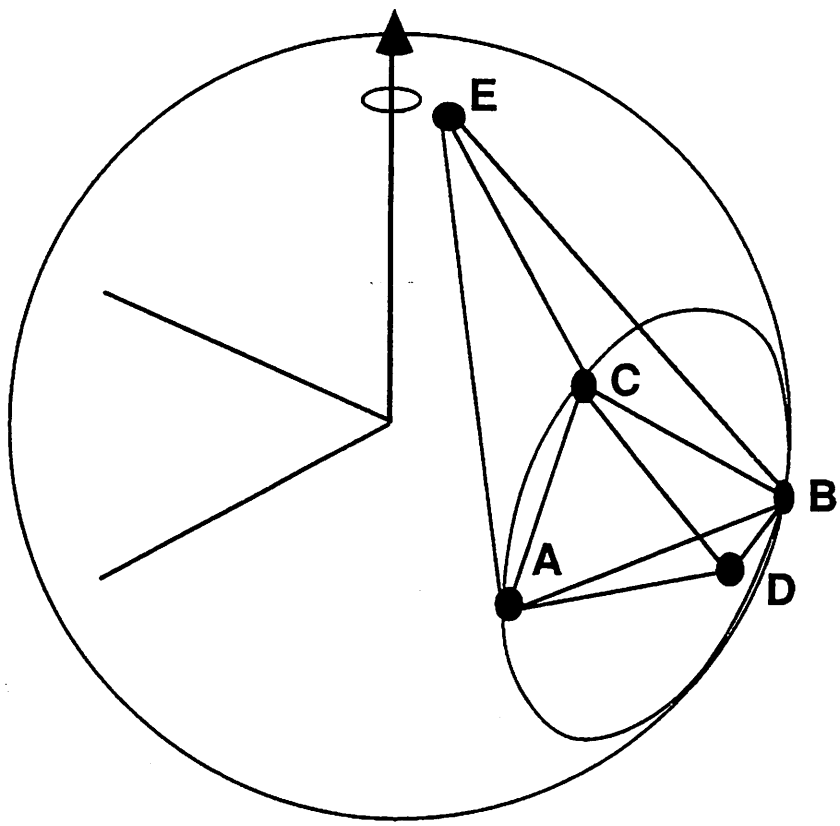


Figure 4 The points are each hemisphere ($\mathcal{V} \cap O_N$ and $\mathcal{V} \cap O_S$) are projected by the opposite pole ($\mathcal{N}$ and $\mathcal{S}$ respectively) into their corresponding planes γ_N and γ_S.



$$\begin{vmatrix} A_{X_e} A_{Y_e} A_N & 1 \\ B_{X_e} B_{Y_e} B_N & 1 \\ C_{X_e} C_{Y_e} C_N & 1 \\ D_{X_e} D_{Y_e} D_N & 1 \end{vmatrix} \overset{?}{>} 0$$

Figure 5 The Incap (Delaunay) predicate is a 4 x 4 determinant that yields the signed volume of the tetrahedron defined by 4 points. In the figure, **ABCD** will have a signed volume < 0, while the volume of **ABCE** is > 0. Triangle **ABC** is Delaunay iff this determinant is > 0 for every other point in $\mathcal{V}$.

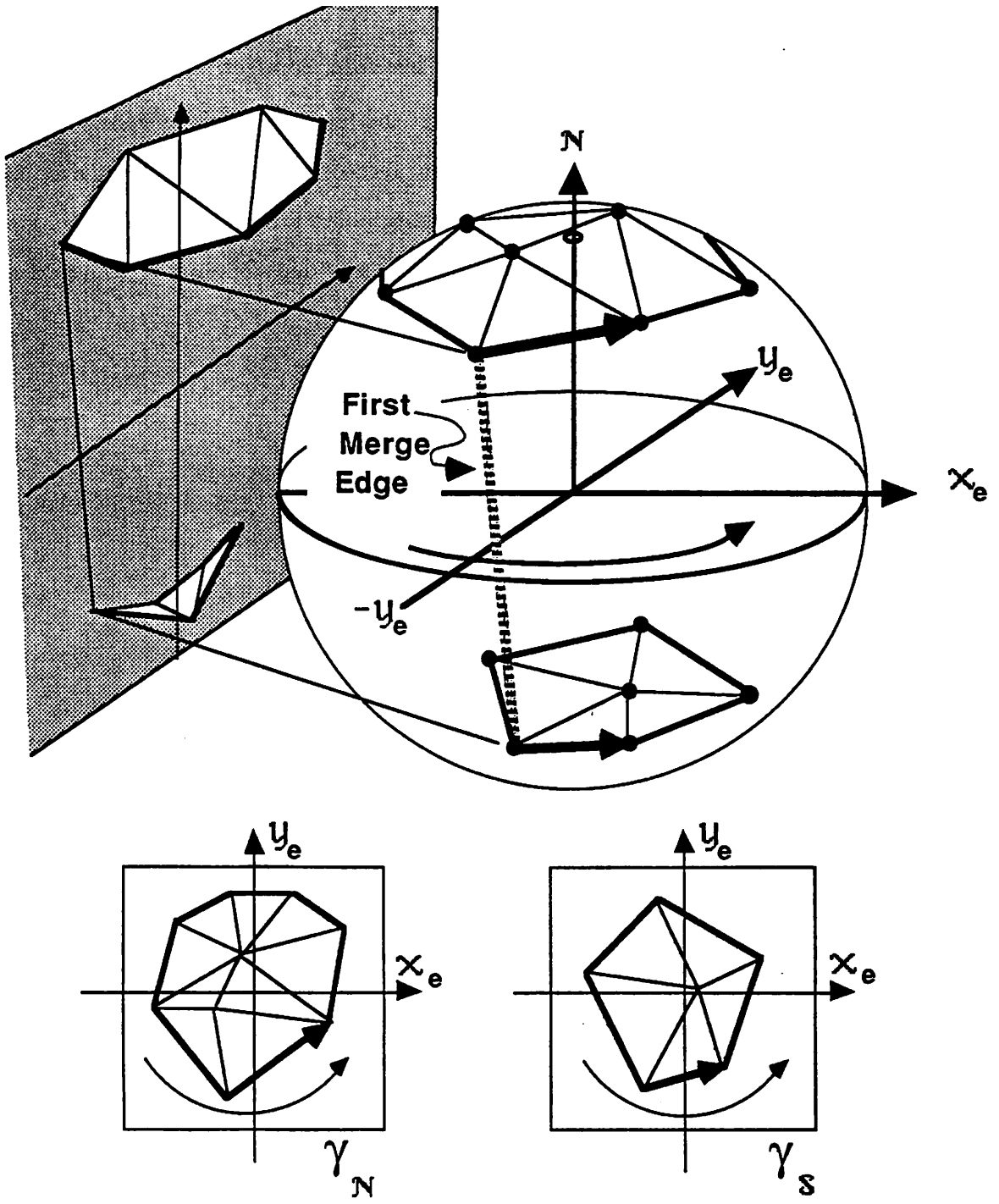


Figure 6 The first merge edge is obtained by connecting the points of each hemisphere with the minimum y_e value. This edge is guaranteed to be Delalunay. The equatorial merge then proceeds in counterclockwise fashion deleting and adding edges as indicated by the Incap predicate until the first edge is reached once more—thus closing the triangulation.

The final merge, Step 1.4, depends for simplicity, on the equivalence between the $DT(\mathcal{V})$ and $CH^3(\mathcal{V})$ (see previous section). Therefore, connecting the points of each hemisphere that are minimum in $\vec{Y}_e$ ensures that the final equatorial merge commences with an edge that is Delaunay. Because it is guaranteed Delaunay, this edge need never be retracted and thus provides a reliable stopping point for the algorithm. This final merge requires total operations on the order of the number of points on the perimeter of both hemispherical triangulations—i.e., O(N).

O(N) Algorithm for Voronoi Diagram

Given the $DT(\mathcal{V})$, the Voronoi Diagram $Vor(\mathcal{V})$ may be constructed in linear time as follows. [See Figures 7 & 8]

Step 2.0 *Voronoi Diagram* $\forall\ de_{ij} \in DT(\mathcal{V})$ **do**

Step 2.1 If the Voronoi Vertex of the Delaunay triangle bounding de_{ij} to the LEFT is NULL

2.1.1 Calculate this vertex as the circumcenter of the 3 points $l_i, l_j, l_k \in \mathcal{V}$ defining this Delaunay triangle

2.1.2 Update the LEFT Voronoi Vertex field of each edge bounding this triangle

Step 2.2 If the Voronoi Vertex of the Delaunay triangle bounding de_{ij} to the RIGHT is NULL

2.2.1 Calculate this vertex as the circumcenter of the 3 points $r_i, r_j, r_k \in \mathcal{V}$ defining this Delaunay triangle

2.2.2 Update the RIGHT Voronoi Vertex field of each edge bounding this triangle

Step 2.3 Connect the LEFT and RIGHT Voronoi vertices by a Voronoi edge ve_{lr} and associate it with de_{ij}

Each Voronoi vertex is easily calculated as the circumcenter of the associated Delaunay triangle v_i, v_j, v_k as follows:

$$vor_{ABC} = \frac{(v_C - v_B) \times (v_B - v_A)}{\|(v_C - v_B) \times (v_B - v_A)\|}$$

A simple dot product $v_{vor} \cdot v_i$ is sufficient to ensure that v_{vor} is properly oriented.

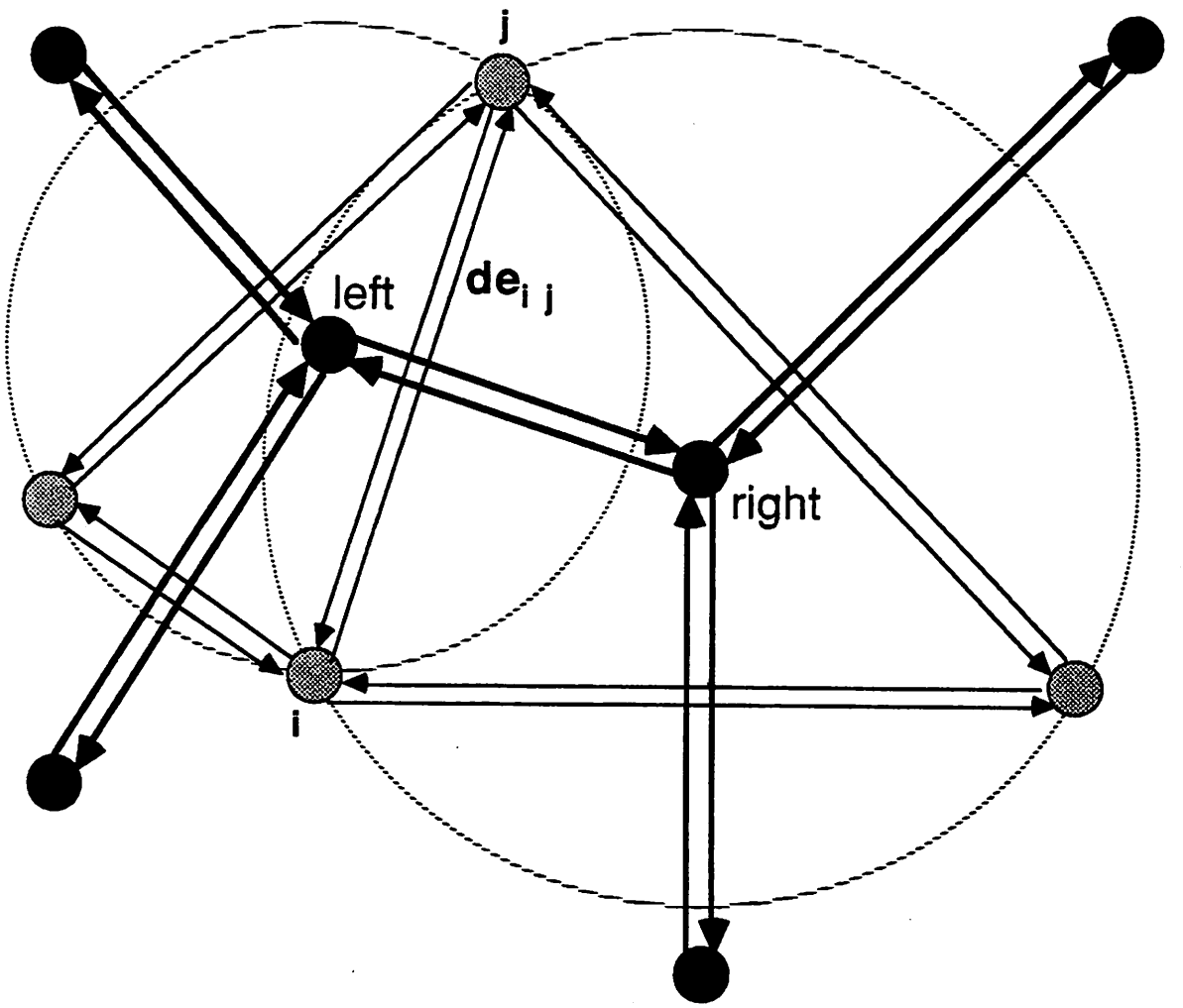


Figure 7 A quad-edge neighborhood. Black dots are voronoi vertices and gray dots are points in $\mathcal{V}$. This shows how edges are pairs of directed arcs. In particular, Delaunay edge $\mathbf{de}_{ij}$ is shown with its corresponding **left** and **right** Voronoi vertices.

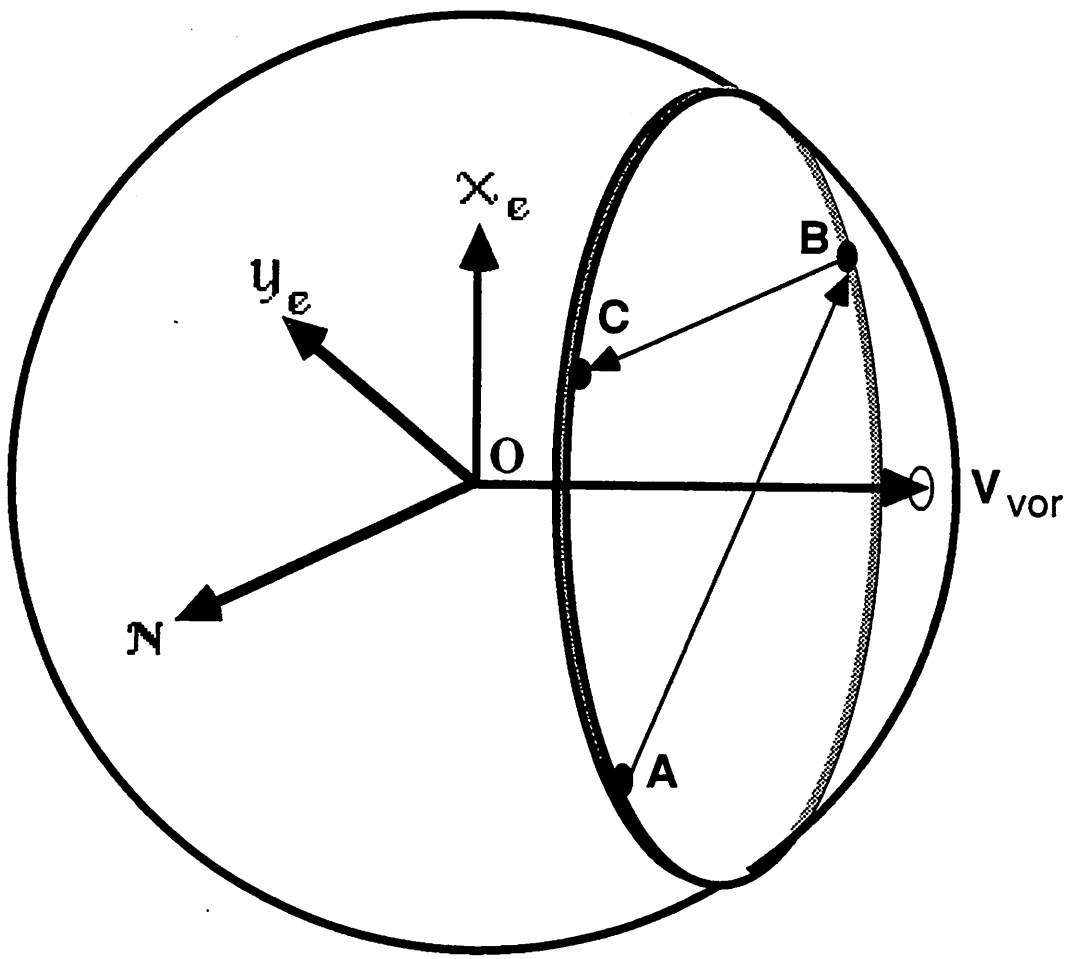


Figure 8 For neighbors in the Delaunay Triangulation the corresponding Voronoi vertex is the center of spherical circle defined by the 3 points. It is obtained as the normalized cross-product of 2 counter-clockwise ordered difference vectors between the points.

Algorithm 2.0 is obviously linear in the number of Delaunay edges, which is easily shown to be order of the number of points O(N). In fact, because of the equivalence of $DT(\mathcal{V})$ with $CH^3(\mathcal{V})$, we know that every vertex of the graph of $DT(\mathcal{V})$ (i.e., every point in $\mathcal{V}$ has degree ≥ 3. Therefore by Euler's formula, we can bound the number of edges E with respect to the number of delaunay vertices N from above and below:

$$3/2N \leq E \leq 3N - 6$$

Thus on Φ, the Voronoi diagram is derived from the Delaunay Triangulation in O(N) time.

$O(N)$ Algorithm for *GMST* on Φ

Given $DT(\mathcal{V})$, we use the following algorithm to extract the GMST in linear time. The algorithm maintains a forest of trees in a queue F. The trees are each assigned numbers indicating the stage of processing, and at any time in the processing these form a non-decreasing sequence in F. A graph structure G^* is also employed. Initially $G^* = DT(\mathcal{V})$, but it is progressively simplified with the completion of each stage by deleting all unselected edges internal to each tree and deleting all but the shortest edges between trees.

Step 3.0 *GMST Construction*

Step 3.1 For every $v_i \in \mathcal{V}$

$stage(v_i) := 0$

$F \Leftarrow v_i$ */* insert into back of queue */*

Step 3.2 current-stage := 0

Step 3.3 while F holds more than 1 tree **do**

3.3.1 $T_0 \Leftarrow F$ */* remove from front of queue */*

3.3.2 if($stage(T_0) >$ current-stage)

simplify G^*

current-stage + = 1

3.3.3 (u, v) := shortest unselected edge of $T_0 (u \in T_0)$

3.3.4 T_1 :=tree in F containing v

3.3.5 delete T_1 from F

3.3.6 $T_2 := merge(T_0, T_1)$

3.3.7 $stage(T_2) := min(stage(T_0), stage(T_1)) + 1$

3.3.8 $F \Leftarrow T_2$

This is the Cheriton/Tarjan Algorithm [CHER76] for linear time extraction of the EMST from the Delaunay Triangulation of a planar pointset. However, the algorithm is equally applicable to pointsets on the sphere, since the major performance requirement is that the embedding graph be planar and $DT(\mathcal{V})$ is obviously planar. The algorithm runs in time proportional to the number of edges, which is O(N). For a detailed analysis see [CHER76;PREP85].

<u>$O(N \log N)$ Heuristic for *GSMT* 's on (Φ)</u>

The heuristic for the *GSMT* works by utilizing the simplicial decomposition of the $DT(V)$ and the *GMST* to decompose the point set into local optimal solutions of the *GSMT* then concatenates these local optimal solution into a sub-optimal solution for $\mathcal{V}$.

The heuristic is largely based on the ideas for solving the *ESMT* in E^2 [SMIT81] although their are some unique problems associated with solving for the three and four point cases on Φ. As in the planar solution, we examine only 3 and 4-point components as experimentally they have the most significant reduction in *GSMT* solutions.

Step 4.0 *GSMT Construction*

Step 4.1 Mark all the triangles during the *GMST* process with two edges of the *GMST* . Place those marked triangles in a priority queue $\mathcal{Q}$ where $\mathcal{Q}$ is prioritized on $\rho_i = L_S(\mathcal{V})/L_M(\mathcal{V})$ for each doubly-marked triangle i.

Step 4.2 Where possible, concatenate pairs of Delaunay triangles, so that 4-point local optimal solution of the *GSMT* are generated.

Step 4.2.1 The adjacent triangles are identified by the spanning tree edges of the marked triangle t_i with Voronoi point vp_i as its root. See Figure 9 below for this four-point concatenation process.

Step 4.2.2 For the four points of each adjacent triangle, construct a solution using arcs of great circles as straight lines and geodesic distances as lengths. [See additional details in the next subsection which follows this formal algorithmic description.]

Step 4.2.3 If this four point concatenation is not possible, add the single Steiner point associated with t_i and resulting edges to the overall *GSMT* solution.

Step 4.3 The process is complete once the priority queue Q is empty.

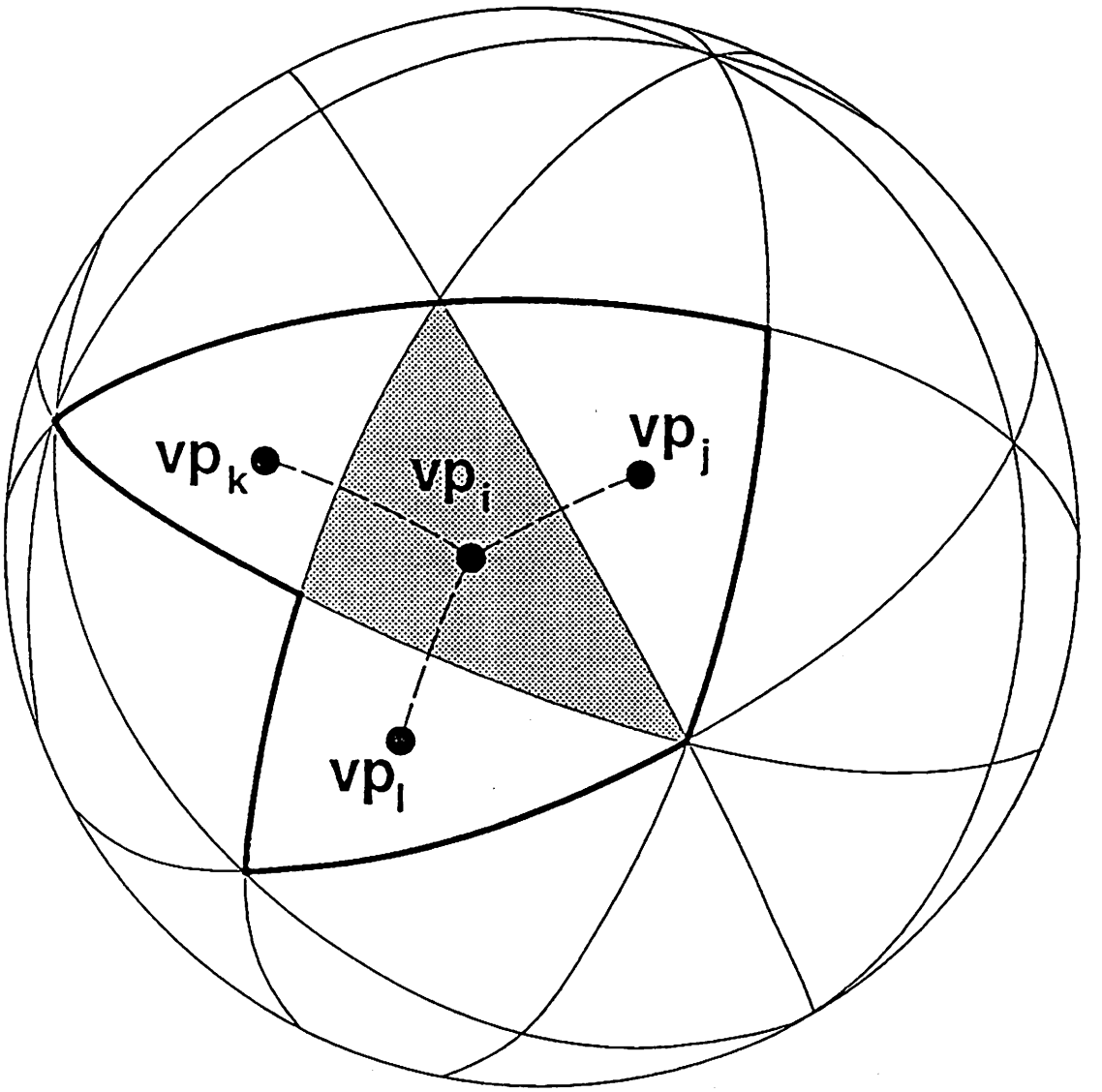


Figure : Nearest Neighbor $DT(vp_i)$ Triangle Pairs

3 and 4 Point Constructions on Φ

Local 3-point and 4-point solutions on Φ are computed in constant time by geometric constructions, similar to those developed for 3 and 4 points in the plane. Figures 10a and 10b show the planar and spherical 3-point constructions respectively. Referring to figure 10b, the geometric construction for points A, B, C on Φ entails first computing points $a1$ and $a2$, which we term *auxiliary points*. These lie on Φ and they each complete outward-pointing equilateral spherical triangles with edges AB and BC respectively. Once these auxiliary points are computed the 3-point Steiner solution *gspt* is calculated as the common intersection of Φ and two planes: the plane of $a1 \times C$; and the plane of $a2 \times A$. Note that this is a direct analog of constructing the 3-point solution in the plane (figure 10a) by intersecting lines connecting each of two auxiliary points to the opposing vertex of the original triangle.

Auxiliary points are computed as the intersection of a plane and two spheres. For point $a1$ these are: the bisector of edge AB, because an equilateral vertex lies in the plane bisecting its opposite side; Φ, because the auxiliary point must lie on the unit sphere; and the sphere of radius $|AB|$ centered at A, by the equilateral property $|Aa2| = |AB|$. Respectively, this yields the following set of simultaneous equations:

$$ax + by + cz = 0,$$

$$x^2 + y^2 + z^2 - 1 = 0,$$

$$(x - x_A)^2 + (y - y_A)^2 + (z - z_A)^2 - |AB|^2 = 0.$$

Note that the constant term d of the first equation is known to be 0 since the bisecting plane of any chord or geodesic of a sphere passes through the center. Note also that the third equation constrains the straight-line/chord length rather than the geodesic length. However, these lengths are monotonically related via the sine function, so that constraining chord length is equivalent to constraining geodesic length. The first two equations can be used to rewrite two of the variables in terms of the third. Substituting into the third equation, yields a quadratic in one unknown. This produces two solution

points, only one of which is associated with the outward pointing triangle that defines the auxiliary point. However, the oriented quad-edge structure of *GMST* indicates the proper choice. Triangle ABC is selected so that edges AB, BC, and CA form a counterclockwise traversal—i.e., they share the same left Voronoi vertex vor_{ABC}. Now $\vec{AB} \times \vec{Ba1}$ must point in a direction opposite to $vor_{ABC} = \vec{AB} \times \vec{BC}$—i.e., $vor_{ABC} \cdot (\vec{AB} \times \vec{Ba1}) < 0$.

The Steiner intersection requires solving the following set of equations:

$$a_1 x + b_1 y + c_1 z = 0,$$

$$a_2 x + b_2 y + c_2 z = 0,$$

$$x^2 + y^2 + z^2 - 1 = 0.$$

The first is the plane equation of the geodesic connecting $a1$ and C. There is no constant term as the plane is through the origin; the coefficients are given by $a1 \times C$. Similarly the second equation has coefficients given by $a2 \times A$. The third equation is that of Φ. Row reducing the the first two equations and substituting into the third yields a quadratic in one unknown. Of the two solution points the desired one *gspt* is oriented positively with respect to the Voronoi point—i.e., $vor_{ABC} \cdot gspt > 0$. Finally, one must check that the intersection is within the perimeter of the given points. As in selecting the appropriate auxiliary point, this is accomplished by checking the dot-product of a cross-product for each perimeter edge. For example, if $vor_{ABC} \cdot (\vec{AB} \times \vec{Bgspt}) < 0$ then intersection is outside. If this is > 0 then check against the next edge—$vor_{ABC} \cdot (\vec{BC} \times \vec{Cgspt}) < 0$, etc. If these dot-products are positive for all sides then the intersection is internal.

The 4-point solution also emulates the corresponding planar construction. It is slightly more complicated than the 3-point method, but it too utilizes the same basic constructions: auxiliary point and Steiner intersection. Figures 11a and 11b show the planar and spherical constructions respectively. The complication arises from the fact that there are two possible topologies for a 4-point Steiner solution as shown in Figure 11a. In general these are of different lengths and the Steiner solution is by definition the

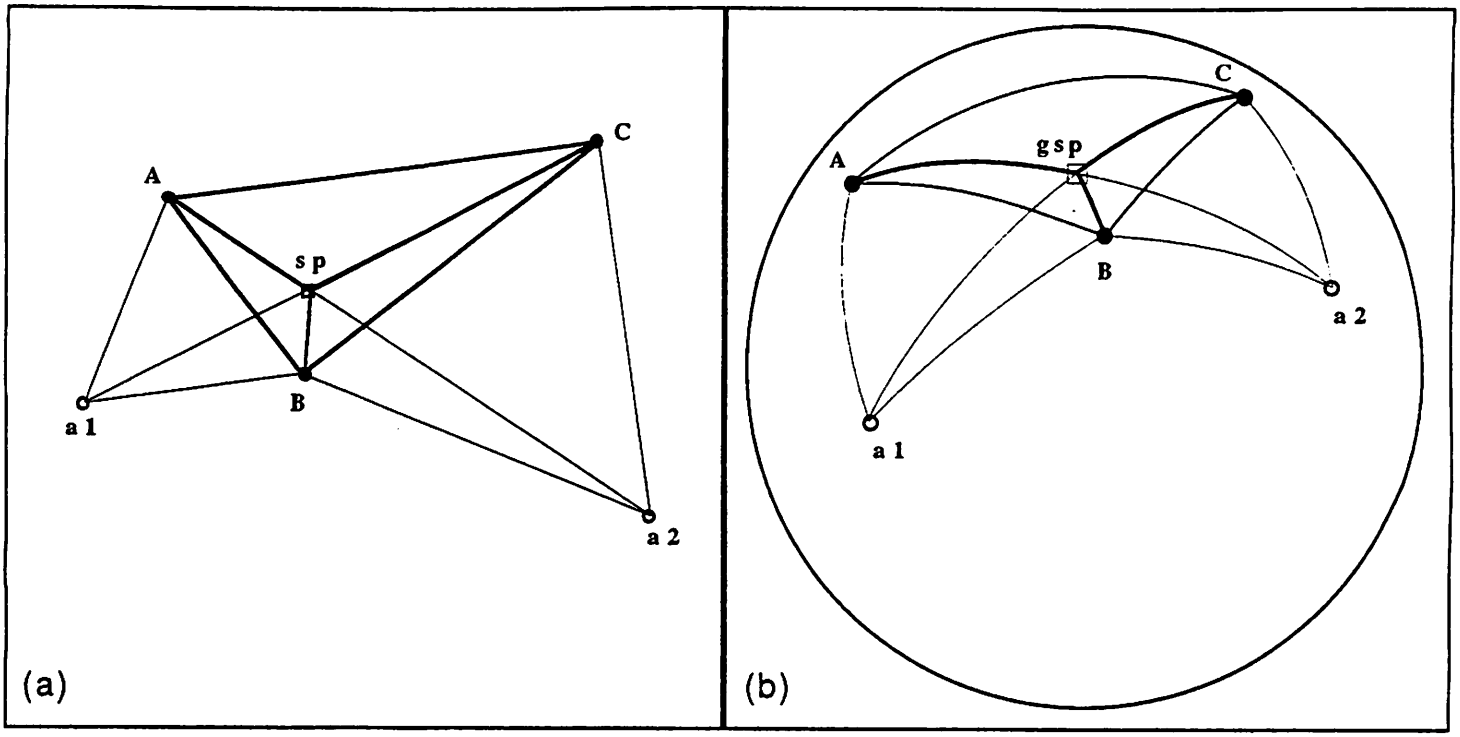


Figure 10: (a) Planar 3-point solution method by intersection of straight-lines in the plane; (b) Spherical 3-point solution method by intersection of great circles on sphere. Points a1 and a2 are equilateral auxiliary points; sp is the planar steiner point in (a); gsp is the geodesic steiner point in (b).

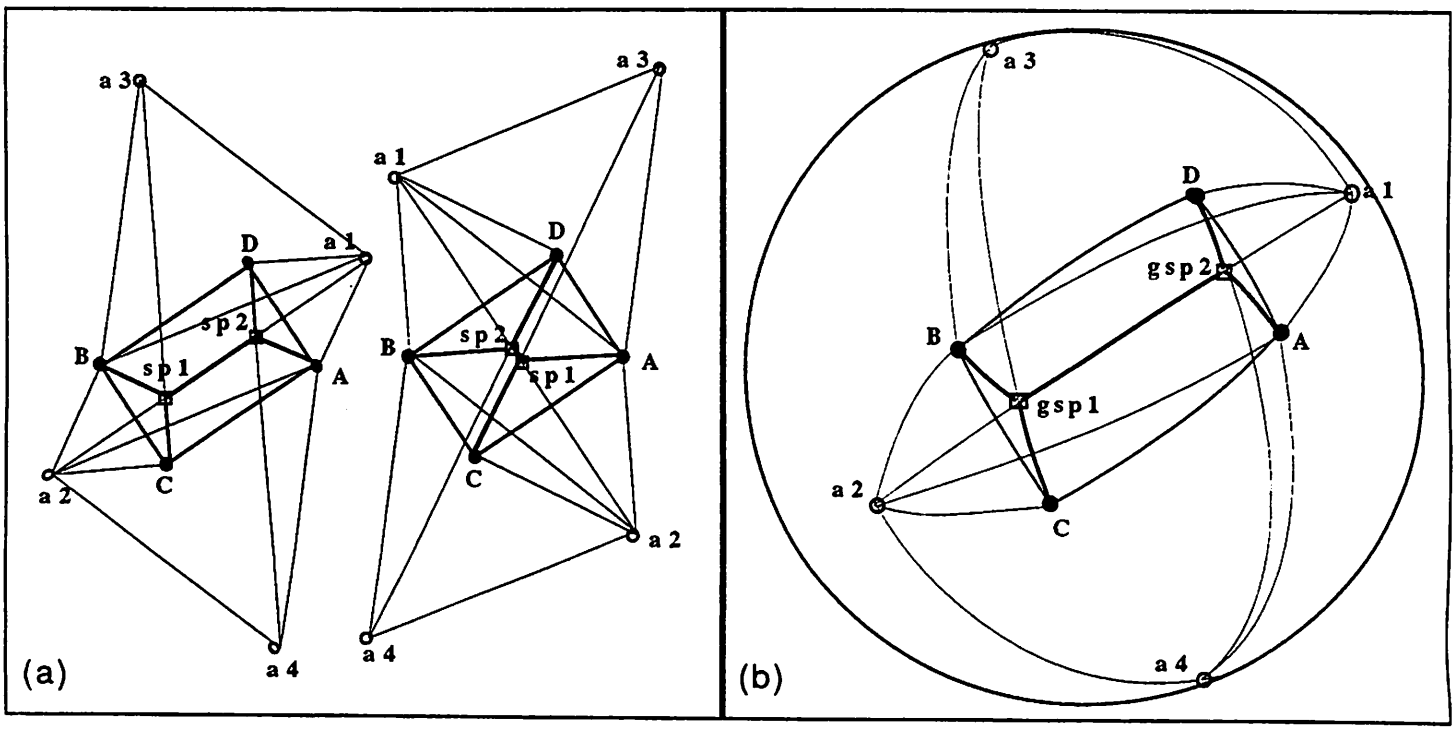


Figure 11: (a) Planar 4-point solution method yields two topologies, one of which is shorter and is the solution; (b) Spherical 4-point solution method showing only shorter topology. Points a1, a2, a3, a4 are equilateral auxiliary points; sp1 and sp2 are the planar steiner points in (a); gsp1 and gsp2 are the geodesic steiner points in (b).

shorter. In the plane, it is known that by measuring the angle between the diagonals of the quadrilateral of points and constructing the initial auxiliary points on the sides opposite the smaller angle, the shorter topology is achieved [POLL78]. It can be shown that the same property holds on Φ, so this provides a reliable method of selecting topologies.

Referring to figure 11b, the 4 points A, B, C, D form a spherical quadrilateral $ADBC$. Measuring the angles between the planes of the diagonals AB and CD indicates on which sides to erect the initial auxiliary points $a1$ and $a2$—in the figure these are on sides AD and BC respectively. Secondary auxiliary points are erected on bases formed by connecting the initial auxiliary points to the opposing vertices of their respective given triangles. In figure 11b, $a1$ is constructed on side AD of triangle ADB and is therefore connected to opposing vertex B to form new base $a1B$. On this base is erected secondary auxiliary point $a3$. Likewise, $a2$ is connected to opposing vertex A to form base $a2A$ on which is erected secondary auxiliary point $A4$. Each of the two Steiner points is computed as the intersection of: the plane containing one of these secondary auxiliary points and its opposing quadrilateral vertex; the plane of the two initial auxiliary points; and Φ. Again in figure 11b, $gsp1$ is the intersection of plane $a3 \times C$, plane $a1 \times a2$, and Φ. Likewise, $gsp2$ is the intersection of plane $a4 \times D$, plane $a1 \times a2$, and Φ.

E. Complexity Analysis

As can be seen from the preliminary analysis of each step, the overall worst case running time of the algorithm is bounded by the time complexity of the construction of the Delaunay triangulation at **Step 1.0**—i.e., $O(NlogN)$. All the remaining major steps and sub-steps of the algorithm for the *GMST* and the sub-optimal solution for the *GSMT* are either $O(1)$ or in the worst case $O(N)$. Storage is also $O(N)$.

In the next section we experimentally compare the heuristic for the *GSMT* with the solution for the *GMST* for randomly distributed points on Φ.

F. Experimental Results

There are a number of issues related to the geometry of the sphere that complicate attempts to directly compute spanning trees on Φ. First and most obviously, the sphere embodies degeneracies usually associated with antipodes—e.g.,the multiplicity of geodesics between antipodal points. Such events require special tests and conventions to handle. Second, the sphere is bounded—i.e., the wrap-around that exists complicates the neighbor relation. For example, a triangle of points on the sphere defines two spherical triangles, usually a major and a minor one. Without information about the rest of the spanning network, it is unclear in which of these to search for a Steiner point. However, the Delaunay triangulation and Voronoi diagrams provide this oriented proximity information; by first computing these structures the task of constructing spanning trees is simplified. Third, the sphere lacks a natural orientation by means of which positional relations like left-of and right-of are defined. These relations help maintain performance bounds for the Delaunay triangulation by providing an ordering of the points. Our use of stereographic projection imparts such an ordering during the Delaunay construction process. To recap, our implementation strategy entails first computing the Delaunay triangulation. A partition on the point set helps avoid degeneracies, and projection provides an ordering. Once the triangluation is computed we may reliably derive our other structures without special handling. The proximity information of the Delaunay and Voronoi diagrams simplifies construction of the *GMST* , and from all of these we obtain oriented proximity information that simplifies construction of local Steiner solutions.

As a conceptual and coding expedient, we did not implement the 3- and 4-point methods as described in the algorithms section. The primary difference is that the geometric constructions are actually performed in an arbitrarily oriented plane and then mapped to Φ, rather than being carried out directly on Φ. Accordingly, the 3-point case entails geometrically constructing a "planar solution" in the arbitrarily oriented plane of the given points. Referring to figure 12a, for points A, B, C on Φ auxiliary points $a1$ and $a2$ are first computed. These still complete outward-pointing equilateral triangles with edges BC and AB respectively, but lie in the common plane of A, B, C instead

of on Φ. This means that in the auxiliary point function the equation of plane ABC—i.e.,$a_1x+b_1y+c_1z-d_1 = 0$—is substituted for the equation of Φ. Since the Voronoi point vor_{ABC} associated with triangle ABC is the unit normal vector of plane ABC, a_1, b_1, c_1 are simply the components of this vector. Further, $d_1 = (a_1, b_1, c_1) \cdot (x_A, y_A, z_A)$.

Once these auxiliary points are computed a "planar solution" *spt* is calculated as the common intersection of three planes: the plane of $a1 \times A$; the plane of $a2 \times C$; and the plane of ABC. Again the equation of plane ABC replaces the equation of Φ in the Steiner intersection function. The point *spt* is then raised to the sphere via central projection. For points on an equilateral triangle it is easy to show that the resulting solution point on the sphere is optimal. For other configurations it is likely sub-optimal, although we currently lack a proof.

The 4-point solution is a bit more complicated in that 4 points lie in two arbitrarily oriented triangles sharing a common edge. In order to exploit the known planar construction, it is necessary to first rotate one triangle about the common edge until it lies in the plane of the other triangle. Referring to figure 12b, triangle ADB is rotated into the plane ABC, giving $AD'B$. Auxiliary points are then erected and Steiner intersections computed just as described in the algorithm, but with the equation of plane ABC replacing the equation of Φ in both calculations. This yields the two "planar solutions" *spt*1 and *spt*2′. Upon completing this planar construction, the triangle $AD'B$ with its corresponding solution point *spt*2′ is rotated back to its original orientation—i.e., into plane ADB yielding *spt*2. Both solution points *spt*1 and *spt*2 are then raised to the sphere by central projection. We do not yet know for which point configurations and indeed if such a construction achieves optimality. Nevertheless, experimental evidence indicates that these "planar" constructions, both 3- and 4-point, achieve respectable Steiner ratios.

One further interesting issue, related to the geometry of the sphere, concerns the Steiner ratio $\rho = L(MST)/L(SMT)$. It came to light as a result of our attempts to implement minimum spanning trees on Φ. For a triangle of points in the plane, ρ is solely a function of the geometric configuration of the points—attaining the well known

limit of $\sqrt{3}/2$ in the equilateral case.[1] By contrast for a triangle of points on the sphere, ρ is not only a function of the configuration but is a function of the central angle as well. The central angle is defined as follows: a triangle of points on Φ defines a plane which intersects Φ in a circle—the circumcircle of the triangle. This circle defines a right circular cone with the origin, and the central angle is the angle between the axis and side of this cone.

Referring to figure 13, we can now consider the particular relationship between ρ and the central angle α for an equilateral triangle of points on Φ. Note that, as in the plane, the circumcenter and Steiner point are coincident in the equilateral case. Denoting the length of a side as s and the associated chord length as τ, yields $\tau = \sqrt{3}\ sin\alpha$ and $s = 2\ sin^{-1}(\tau/2) = 2\ sin^{-1}(\sqrt{3}/2\ sin\alpha)$. Thus

$$\rho = L(GSMT)/L(GMST) = 3\alpha/2s = 3\alpha/4\ sin^{-1}(\sqrt{3}/2\ sin\alpha).$$

Plotting the numerator and denominator simultaneously we see that their graphs are respectively linear and trigonometric and that they cross at $0°$ and $\approx 76.5°$. Thus Steiner points exist only for equilateral triangles whose central angle is $0° \leq \alpha <\approx 76.5°$. Examining the neighborhood around $0°$ reveals that ρ approaches the planar optimal of $\sqrt{3}/2$ as α approaches $0°$. This is entirely consistent when one considers that progressively smaller neighborhoods on the sphere are more and more nearly planar.

This central angle property seems to limit the effectiveness of improvement for sparse point sets. In fact, we would expect to see maximum improvement in dense point clusters and especially for points on an equilateral lattice. With this in mind, we tried three different methods for selecting random points. Method 1: points are selected uniformly from the ϕ, θ rectangle—i.e., ϕ uniformly $\in [0, \pi]$ and θ uniformly $\in [0, 2\pi]$. Because of the compression of this space at the poles when wrapped onto the sphere, this method should produce moderate concentrations of points near the poles. Method 2: points are selected with a polar bias — i.e., θ uniformly $\in [0, 2\pi]$ and α

[1] *Theorems 1 and 2 indicate that similar geometric constraints hold for points on the sphere.*

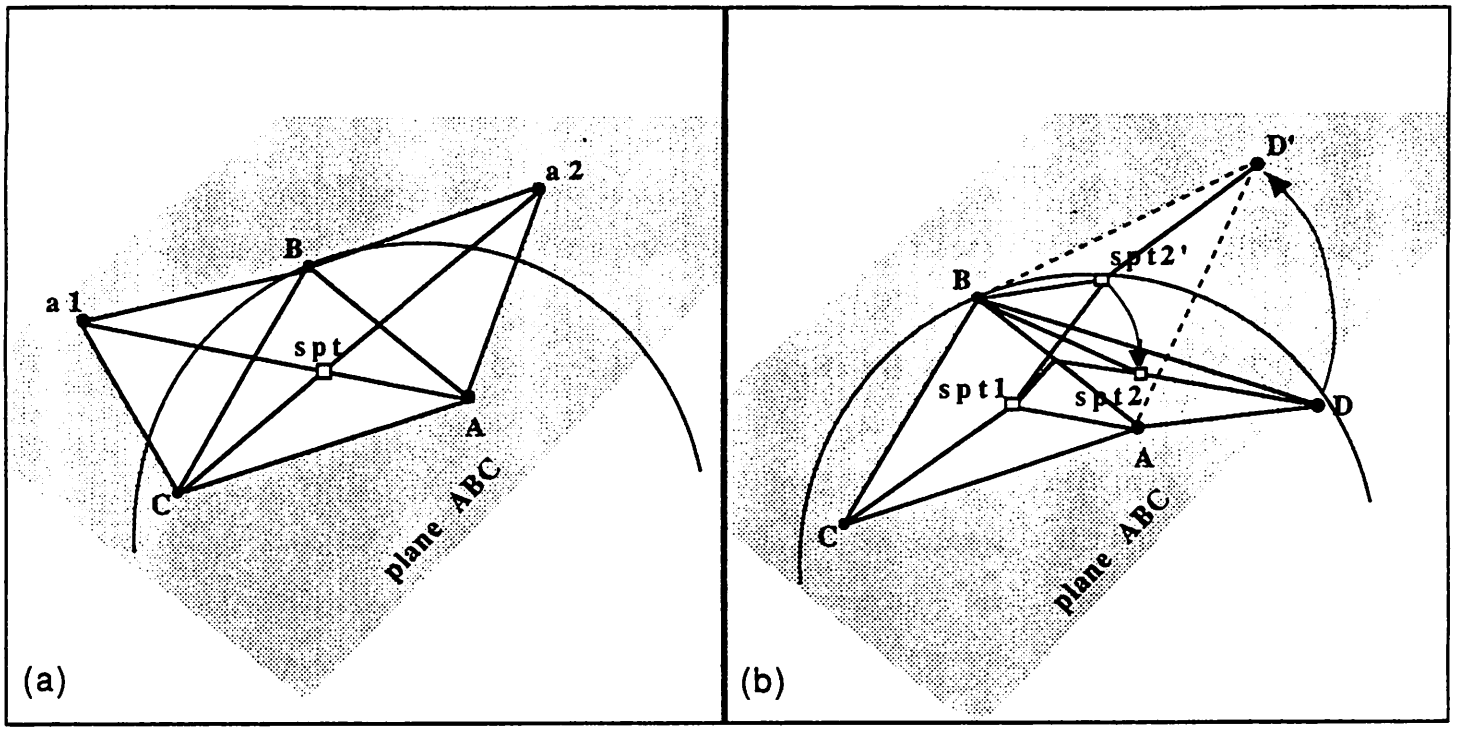


Figure 12 Solution methods as actually implemented: (a) 3-point method solved in arbitrarily oriented plane of ABC; (b) 4-point method first rotates fourth point into common plane, computes solution points in that plane, and then rotates second solution point back into orginal plane. The 3- and 4-point "planar solutions" are then raised to the sphere by central projection.

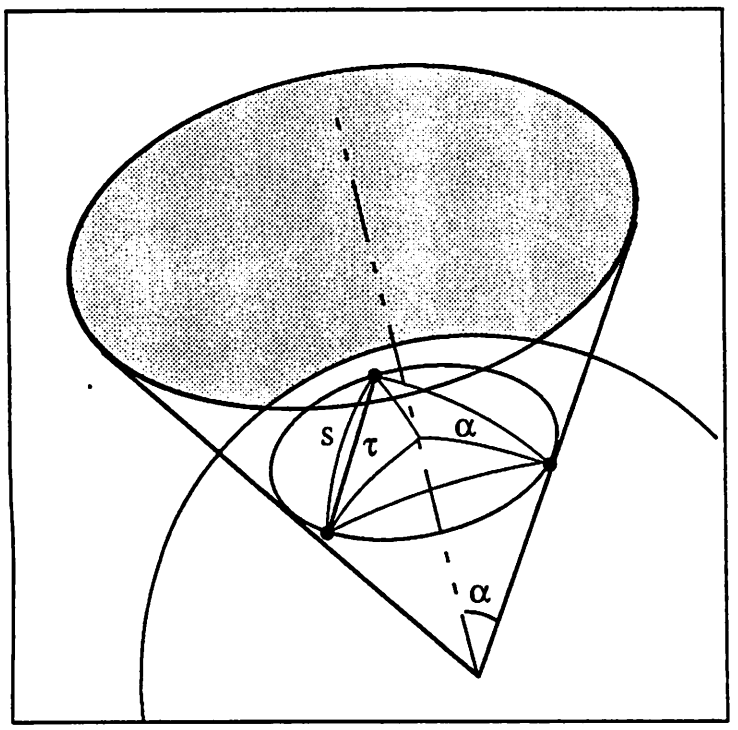


Figure 13 Relationship of central angle α to steiner ratio ρ = L(GSMT)/L(GMST), for the equilateral case. Length of side of spherical triangle = s, length of chord = τ. L(GSMT) = 3α; L(GMST) = $2s$; $s = 2 \sin^{-1}(\tau/2)$; and $\tau = \sqrt{3} \sin \alpha$.
Thus, L(GMST) = $4 \sin^{-1}(\sqrt{3}/2 \sin \alpha)$.

uniformly $\in [0, \pi]$. Then $\phi = (1 + cos(\alpha))\pi/2$. Mapping through the cosine function tends to distributes points away from the equator $\phi = \pi/2$ and concentrate them about the poles—i.e.,$\phi \approx 0$ and $\phi \approx \pi$. Method 3: points are selected by approximate uniform area of the sphere's surface—i.e., θ uniformly $\in [0, 2\pi]$ and x uniformly $\in [-1, 1]$. Then $\phi = cos^{-1}(x)$. We have seen earlier that the distance formula for points on the sphere involves an inverse cosine. The form of this function tends to map a large band of values centered about 0 to a narrow band of values centered about $\pi/2$—the equator. Few points are mapped near 0 or near π—the poles. Being thus responsive to the length of lines of latitude, this method tends to produce point sets that are approximately evenly distributed on the surface of the sphere.

Tables 1 $\sim$ 15 contain preliminary experimental results of the *GSMT* heuristic applied to point sets of 9, 15, 50, 75, and 100 points generated by each of the three random methods. In addition, in order to isolate the effect of increasing point density, we decided to examine performance on regular point sets—sets whose points are distributed on an equilateral lattice. For this purpose, we used the vertices of the three regular polyhedra with equilateral faces: the tetrahedron with 4 vertices, the octahedron with 6 vertices, and the icosahedron with 12 vertices. These are included at the top of Table 16. We must point out that the results that appear in Tables 1 $\sim$ 15 are based on only five runs per random method per point set size; they are therefore not statistically significant. However, they do provide some indication of the relative effects of the different distributions and the effect of the number of points.

We track three local and five global measures. The local measures are: **min3ρ**—the minimum 3-point Steiner ratio over all triangles of points considered; **adj3ρ**—the minimum ratio obtained by comparing the length of a 3-point solution adjoined with a neighboring *GMST* edge versus the length of the 3 *GMST* edges connecting the same 4 points; **min4ρ**—the minimum ratio of the lengths of 4-point Steiner solutions compared with the lengths of the corresponding *GMST* edges. Note, these local measure are the minimums computed regardless of whether these local solutions were actually used in the final network. The global measures are: **#3**—the number of 3-point local solutions

used in the final network; #4—the number of 4-point local solutions used in the final network; **GMST**—the length of the minimum spanning tree (only the given points are nodes); **GSMT**—the length of the corresponding sub-optimal Steiner tree (the length of the final network, with local Steiner solutions incorporated); ρ—the ratio of length of *GSMT* to the length of the *GMST* . Table 16 also illustrates the net measure in the more intuitive form of percentage improvement—i.e., $(1-\rho)*100$.

In the case of regular point sets, our analysis of the central angle predicts monotonically better performance with an increasing number of points, and this is precisely what is illustrated in Table 16. Each measure improves with an increase in the number of points. In the case of randomly selected points, the expectation is that measures both global and local should generally improve with an increase in the number of points. Based on the discussion of the central angle, we would expect that for a given number of points, the polar-bias method would exhibit the best local measures due to clustering. The uniform ϕ, θ method would have slightly worse local measures and the uniform-area method would likely have the poorest local measures. The comparative behavior of global measures are more difficult to anticipate across methods. Table 16 summarizes the experimental results of Table $1 \sim 15$ across all three methods. Finally it should be pointed out that all results on Φ compare with the exepected results in E^2 since we should expect between $2 \sim 4\%$ improvements [GILB68] and the heuristic results found in [SMIT81] also compare favorably with those found in this paper. Additional refinements in the solutions for the 3 and 4 point local optimal solutions on Φ may yield further improvements for the general case of n randomly distributed point sets.

Table #1 METHOD = 0 POINTS = 9

seed	#3 pts	#4 pts	min $3pt.\rho$	adj $3pt.\rho$	min $4pt\rho$	GMST	GSMT	ρ
0	1	0	0.9797	0.9862	1.0755	6.0939	6.0645	0.9952
1	2	0	0.9671	0.9817	0.9878	6.4028	6.3265	0.9881
2	3	0	0.8813	0.9801	1.1218	4.9472	4.7844	0.9671
3	3	0	0.9043	0.9100	0.9101	6.1266	5.8300	0.9516
4	1	0	0.9664	0.9864	1.0210	4.8983	4.8889	0.9981

Table #2 METHOD = 0 POINTS =15

seed	#3 pts	#4 pts	min $3pt.\rho$	adj $3pt.\rho$	min $4pt\rho$	GMST	GSMT	ρ
0	1	1	0.9287	0.9372	0.9343	7.2053	7.0146	0.9735
1	4	0	0.8989	0.9251	0.9568	9.1004	8.8637	0.9740
2	4	0	0.8813	0.9869	1.1259	7.1997	7.0540	0.9798
3	4	0	0.9271	0.9316	1.2441	7.3403	7.1305	0.9714
4	3	0	0.9664	0.9731	1.0436	7.7031	7.6210	0.9893

Table #3 METHOD = 0 POINTS = 50

seed	#3 pts	#4 pts	min $3pt.\rho$	adj $3pt.\rho$	min $4pt\rho$	GMST	GSMT	ρ
0	10	2	0.8762	0.9256	0.9238	13.7910	13.4316	0.9739
1	13	1	0.9192	0.9248	0.9316	16.1125	15.6673	0.9724
2	11	0	0.8946	0.9354	0.9489	14.3199	14.0287	0.9797
3	13	2	0.8905	0.9259	0.9204	16.0652	15.5921	0.9706
4	8	2	0.8957	0.9224	0.9460	15.1209	14.8351	0.9811

Table #4 METHOD = 0 POINTS = 75

seed	#3 pts	#4 pts	min $3pt.\rho$	adj $3pt.\rho$	min $4pt\rho$	GMST	GSMT	ρ
0	12	5	0.8859	0.9171	0.9294	18.6809	18.3255	0.9810
1	12	4	0.8972	0.9135	0.9135	19.0198	18.5483	0.9752
2	19	2	0.9239	0.9375	0.9489	17.6499	17.2620	0.9780
3	22	1	0.8905	0.9075	0.9498	18.7812	18.2468	0.9715
4	18	1	0.8875	0.9233	0.9054	18.0016	17.5922	0.9773

Table #5 METHOD = 0 POINTS = 100

seed	#3 pts	#4 pts	min $3pt.\rho$	adj $3pt.\rho$	min $4pt\rho$	GMST	GSMT	ρ
0	22	6	0.8934	0.9015	0.9214	22.3470	21.8265	0.9767
1	26	3	0.9288	0.9456	0.9305	21.7814	21.2730	0.9767
2	23	3	0.8727	0.9305	0.9310	20.3624	19.8184	0.9733
3	22	5	0.8905	0.9055	0.9387	22.1415	21.5613	0.9738
4	26	2	0.8806	0.9194	0.9444	20.3867	19.9376	0.9780

Table #6 METHOD = 1 POINTS = 9

seed	#3 pts	#4 pts	min $3pt.\rho$	adj $3pt.\rho$	min $4pt\rho$	GMST	GSMT	ρ
0	1	0	0.9588	0.9701	1.0466	5.5984	5.5525	0.9918
1	0	1	0.9643	0.9791	0.9450	5.6957	5.6508	0.9921
2	2	0	0.9461	0.9910	0.9915	4.4479	4.4252	0.9949
3	3	0	0.9417	0.9586	1.0449	5.9765	5.8188	0.9736
4	1	0	0.9942	0.9996	1.0579	4.1611	4.1603	0.9998

Table #7 METHOD = 1 POINTS = 15

seed	#3 pts	#4 pts	min $3pt.\rho$	adj $3pt.\rho$	min $4pt\rho$	GMST	GSMT	ρ
0	2	1	0.9558	0.9610	0.9953	6.0317	5.9833	0.9920
1	2	1	0.9181	0.9532	0.9646	7.9252	7.7817	0.9819
2	4	1	0.9461	0.9963	0.9960	6.1381	6.0402	0.9841
3	1	1	0.9385	0.9442	0.9507	6.4992	6.3408	0.9756
4	4	0	0.9703	0.9876	1.0571	6.2442	6.1864	0.9908

Table #8 METHOD = 1 POINTS = 50

seed	#3 pts	#4 pts	min $3pt.\rho$	adj $3pt.\rho$	min $4pt\rho$	GMST	GSMT	ρ
0	12	0	0.8798	0.9478	1.0380	11.5865	11.3354	0.9783
1	13	1	0.8901	0.9182	0.9305	13.5892	13.1912	0.9707
2	9	1	0.9192	0.9281	0.9567	11.9223	11.7262	0.9836
3	12	2	0.9076	0.9186	0.9179	14.8406	14.5621	0.9812
4	11	1	0.9147	0.9534	0.9528	11.8387	11.7035	0.9886

Table #9 METHOD = 1 POINTS = 75

seed	#3 pts	#4 pts	min $3pt.\rho$	adj $3pt.\rho$	min $4pt\rho$	GMST	GSMT	ρ
0	16	4	0.8771	0.9099	0.9010	15.8609	15.4359	0.9732
1	17	3	0.9081	0.9342	0.9305	16.5530	16.1105	0.9733
2	18	1	0.9171	0.9281	0.9817	14.8449	14.5551	0.9805
3	18	2	0.8939	0.9223	0.9519	17.8439	17.4186	0.9762
4	13	3	0.9188	0.9262	0.9382	14.0643	13.8268	0.9831

Table #10 METHOD = 1 POINTS = 100

seed	#3 pts	#4 pts	min $3pt.\rho$	adj $3pt.\rho$	min $4pt\rho$	GMST	GSMT	ρ
0	19	7	0.8764	0.9194	0.9192	19.1219	18.8380	0.9852
1	22	2	0.8864	0.9179	0.9187	19.3930	18.9558	0.9775
2	23	6	0.8856	0.9045	0.9144	17.0171	16.6214	0.9768
3	20	4	0.8850	0.9046	0.9519	20.8637	20.3188	0.9739
4	24	2	0.8792	0.8896	0.9161	18.0713	17.7745	0.9836

Table #11 METHOD = 2 POINTS = 9

seed	#3 pts	#4 pts	min $3pt.\rho$	adj $3pt.\rho$	min $4pt\rho$	GMST	GSMT	ρ
0	1	0	0.9771	0.9837	1.0482	6.0226	5.9677	0.9909
1	3	0	0.9233	0.9569	1.0144	6.3076	6.1308	0.9720
2	3	0	0.9172	0.9824	1.0643	5.1865	5.0259	0.9690
3	2	0	0.9013	0.9082	1.0160	5.8341	5.6698	0.9718
4	1	0	0.9993	0.9997	1.0804	4.9930	4.9926	0.9999

Table #12 METHOD = 2 POINTS = 15

seed	#3 pts	#4 pts	min $3pt.\rho$	adj $3pt.\rho$	min $4pt\rho$	GMST	GSMT	ρ
0	3	0	0.9805	0.9919	1.0009	7.4833	7.4413	0.9944
1	4	0	0.9099	0.9358	0.9815	9.0013	8.7286	0.9697
2	3	0	0.9172	0.9682	1.0333	8.1139	7.9264	0.9769
3	3	0	0.9013	0.9082	1.0160	8.0161	7.8269	0.9764
4	2	0	0.9293	0.9382	1.0410	7.8912	7.7465	0.9817

Table #13 METHOD = 2 POINTS = 50

Seed	#3 pts	#4 pts	min $3pt.\rho$	adj $3pt.\rho$	min $4pt\rho$	GMST	GSMT	ρ
0	11	1	0.9132	0.9240	0.9768	14.7020	14.3667	0.9772
1	10	3	0.8779	0.9127	0.9392	16.9329	16.4008	0.9686
2	11	3	0.8950	0.9292	0.9286	16.0575	15.7956	0.9837
3	11	4	0.9049	0.9344	0.9575	17.2184	16.6653	0.9679
4	13	1	0.9424	0.9602	0.9769	16.6048	16.3302	0.9835

Table #14 METHOD = 2 POINTS = 75

seed	#3 pts	#4 pts	min $3pt.\rho$	adj $3pt.\rho$	min $4pt\rho$	GMST	GSMT	ρ
0	18	2	0.8861	0.9065	0.9090	19.7319	19.1809	0.9721
1	14	4	0.8889	0.9127	0.9348	20.1224	19.6492	0.9765
2	16	3	0.8922	0.9145	0.9249	19.3560	18.8501	0.9739
3	21	2	0.9222	0.9402	0.9725	20.4322	19.9767	0.9777
4	17	4	0.9104	0.9318	0.9312	20.2120	19.7357	0.9764

Table #15 METHOD = 2 POINTS = 100

seed	#3 pts	#4 pts	min $3pt.\rho$	adj $3pt.\rho$	min $4pt\rho$	GMST	GSMT	ρ
0	19	6	0.8941	0.9335	0.9080	23.7107	23.0353	0.9715
1	20	6	0.8992	0.9303	0.9536	22.2486	21.7762	0.9788
2	24	5	0.8922	0.9373	0.9639	22.4214	21.8171	0.9730
3	25	2	0.8932	0.9211	0.9495	23.5756	22.9700	0.9743
4	22	5	0.8860	0.9140	0.9098	23.0282	22.4547	0.9751

Table 16 Summary of Results

Regular Polyhedra - (tetrahedron, octahedron, icosahedron)

seed	#3 pts	#4 pts	min $3pt.\rho$	adj $3pt.\rho$	min $4pt\rho$	GMST	GSMT	ρ	% net
4	1	0	0.9664	0.9776	1.0225	5.7319	5.6035	0.9776	2.24
8	2	0	0.9123	0.9415	0.9541	7.8540	7.3027	0.9298	7.02
12	2	2	0.8838	0.9226	0.9111	12.1786	11.0737	0.9093	9.07

Random Points - Summary Method 0

seed	#3 pts	#4 pts	min $3pt.\rho$	adj $3pt.\rho$	min $4pt\rho$	GMST	GSMT	ρ	% net
9	2.0	0.0	0.9398	0.9689	1.0232	5.6938	5.5789	0.9800	2.00
15	3.2	0.2	0.9205	0.9508	1.0609	7.7098	7.5368	0.9776	2.24
50	11.0	1.4	0.8952	0.9268	0.9341	15.0819	14.7110	0.9755	2.45
75	16.6	2.6	0.8970	0.9198	0.9294	18.4267	17.9950	0.9766	2.34
100	23.8	3.8	0.8932	0.9205	0.9332	21.4038	20.8834	0.9757	2.43

Random Points - Summary Method 1

seed	#3 pts	#4 pts	min $3pt.\rho$	adj $3pt.\rho$	min $4pt\rho$	GMST	GSMT	ρ	% net
9	1.4	0.2	0.9610	0.9797	1.0172	5.1759	5.1215	0.9904	0.96
15	2.6	0.8	0.9458	0.9685	0.9927	6.5677	6.4665	0.9849	1.51
50	11.4	1.0	0.9023	0.9332	0.9592	12.7555	12.5037	0.9805	1.95
75	16.4	2.6	0.9030	0.9241	0.9407	15.8334	15.4694	0.9773	2.27
100	21.6	4.2	0.8825	0.9072	0.9241	18.8934	18.5017	0.9794	2.06

Random Points - Summary Method 2

seed	#3 pts	#4 pts	min $3pt.\rho$	adj $3pt.\rho$	min $4pt\rho$	GMST	GSMT	ρ	% net
9	2.0	0.0	0.9436	0.9662	1.0447	5.6688	5.5574	0.9807	1.93
15	3.0	0.0	0.9276	0.9485	1.0145	8.1012	7.9339	0.9798	2.02
50	11.2	2.4	0.9067	0.9321	0.9558	16.3031	15.9117	0.9762	2.38
75	17.2	3.0	0.9000	0.9211	0.9345	19.9709	19.4785	0.9753	2.47
100	22.0	4.8	0.8929	0.9272	0.9370	22.9969	22.4107	0.9745	2.55

G. Summary and Conclusions

This paper has demonstrated the development and implementation of an $O(N \log N)$ algorithm and an $O(N \log N)$ heuristic for the *GMST* and the *GSMT* problems respectively. Issues related to the geometry of the sphere, complicate direct solutions to these problems for points on Φ. However, by exploiting the orientation and proximity information of the Delaunay triangulation and the Voronoi diagram, reasonable performance is obtained. In fact for points sets on Φ, the use of simplicial decomposition is crucial to the complexity and effectiveness of the methodology developed in the paper. It is felt that a similiar approach utilizing simplicial decomposition could be both constructive and effective for minimal length network problems on any Reimann surface.

H. References

[ALY79] Aly, A.A., D.C. Kay and D.W. Litwhiler, 1979. "Location Dominance on Spherical Surfaces," Operations Research 27, 972-981.

[BERN89] Bern, M. and R. Graham, 1989. "The Shortest-Network Problem," Scientific American 260 (1), 84-89.

[CHAN72] Chang, S.K., 1972 "The Generation of Minimal Trees with a Steiner Topology," JACM 19 (4) 699-711.

[CHER76] Cheriton, D. and R. E. Tarjan, 1976. "Finding Minimal Spanning Trees," Siam J. of Computing 5 (4), 724-742.

[CHUN76] Chung,F.R.K. and F.K. Hwang, 1976, "A Lower Bound for the Steiner Tree Problem," Siam J. Appl.Math 34(1), 27-36.

[CHUN78] Chung, F.R.K. and R.L. Graham, 1978. "Steiner Trees for Ladders," Annals Of Discrete Mathematics 2, 173-200.

[COCK68] Cockayne, E.J. and Z.A. Melzak, 1968. "Steiner's Problem for Set Terminals," J.Appl.Math., 26 (2), 213-218.

[COCK69] Cockayne, E.J. and Z.A. Melzak, 1969. "Euclidean Constructability in Graph Minimization Problems." Math.Mag.42, 206-208.

[COCK72] Cockayne, E.J., 1972 "On Fermat's Problem on the Surface of a Sphere." Mathematics Magazine Sept-Oct, 216-219.

[COUR41] Courant, D.R. and H. Robbins, 1941. **What Is Mathematics ?** *New York: Oxford University Press.*

[DOBK87] Dobkin, D. and P. Thurston, "The geometry of Circles: Voronoi Diagrams, Moebius Transformations, Convex Hulls, Fortunes Algorithm, the cut locus and parameterization of Shape," Unpublished Notes for a course. Princeton University, Department of Computer Science, 1987.

[DONN45] Donnay, J.D.H. 1945. Spherical Trigonometry, in **Encyclopedia of Mathematics,** *Interscience: New York.*

[DREZ79] Drezner, Z. and G.O. Wesolowsky, 1979. "Facility Location on a Sphere," Journal of the Operational Research Society 29, 997–1004.

[DU 82] Du, D.z., F.K.Hwang, and J.F. Weng, 1982. "Steiner Minimal Trees on Zig-Zag Lines." Trans. Amer. Math Soc. 278 (1), 149–156.

[GARE77] Garey, M.R., R.L. Graham and D.S. Johnson, 1977. "The Complexity of Computing Steiner Minimal Trees," Siam J. Appl. Math 32 (4), 835–859.

[GARE79] Garey, M.R. and D.S. Johnson, 1979. **Computers And Intractability; A Guide To The Theory Of Np-completeness.** *(San Francisco: W.H.Freeman and Company.)*

[GRAH76] Graham, R.L. and F.K. Hwang, 1976, "Remarks on Steiner Minimal Trees." Bull.Inst.Math.Acad.Sinica 4, 177–182.

[GREE71] Greening, M.G., 1971. "Solution to Problem E2233," Amer. Math. Monthly, 4, 303–304.

[GILB68] Gilbert, E.N. and H.O. Pollak, 1968, "Steiner Minimal Trees." Siam J. Appl. Math 16, 1–29.

[GUIB85] Guibas, L. and J. Stolfi, 1985. "Primitives for the Manipulation of General Subdivisions and the Computations of Voronoi Diagrams," ACM Transactions on Graphics 4 (2), 74–123.

[HILB83] Hilbert, D. and S. Cohn-Vossen, 1983. **Geometry and the Imagination,** *Chelsea.*

[HWAN89] Hwang, F. W. and D. Richards, 1989 "Steiner Tree Problems," (unpublished manuscript).

[LITW80] Litwhiler, D.W., 1980, "Steiner's Problem and Fagnano's Result on the Sphere." Mathematical Programming 18, 286–290.

[LOVE88] Love, R.F. J.G. Morris and G.O. Wesolowsky, 1988. Facilities Location, North-Holland.

[MELZ61] Melzak, Z.A., 1961. "On the Problem of Steiner," Can Math Bulletin 4, 143–148.

[MOTW87] Motwhani, R. and P. Raghavan, 1987. "Geometry on the Sphere: Deterministic and Probabilistic Computations," Unpublished Manuscript.

[PREP85] Preparata, F. and M.I. Shamos, 1985. **Computational Geometry,** *Springer-Verlag.*

[PRIM57] Prim, R.C., 1957. "Shortest Connecting Networks and Some Generalizations." BSTJ 36 1389-1401.

[SHAM78] Shamos, M.I., 1978. "Computational Geometry." Ph.D. Thesis. Yale University.

[SMIT79] Smith, J. MacGregor and J.S. Liebman, 1979. "Steiner Trees, Steiner Circuits, and the Interference Problem in Building Design." Eng Opt 4 (1), 15-36.

[SMIT81] Smith, J. MacGregor, Lee, D.T. and J.S. Liebman, 1981. "An O(NlogN) Heuristic for Steiner Minimal Tree Problems on the Euclidean Metric." Networks 11 (1), 23-39.

[VAID88] Vaidya, Pravin M., 1988. "Minimum Spanning Trees in k−Dimensional Space" SIAM J. Comput 17 (3), 572-582.

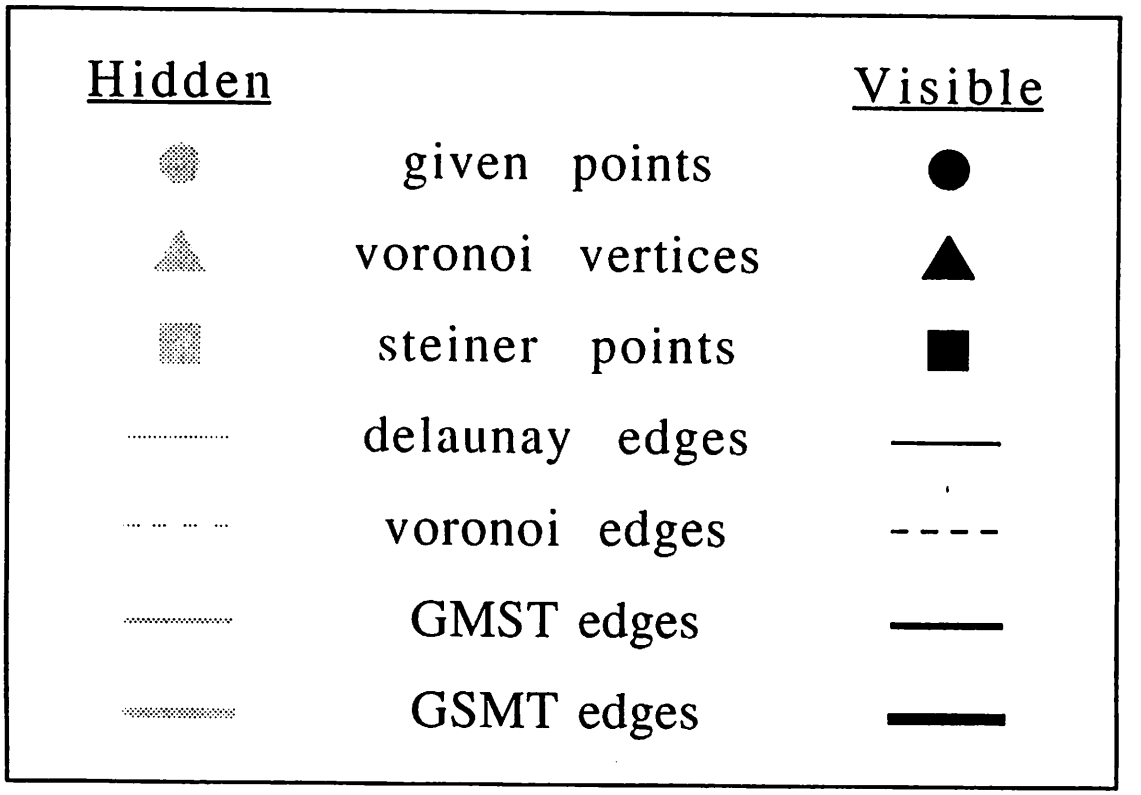
[WESO82] Wesolowsky, G.O., "Location Problems on a Sphere." Regional Science and Urban Economics, *12, 495-508.*

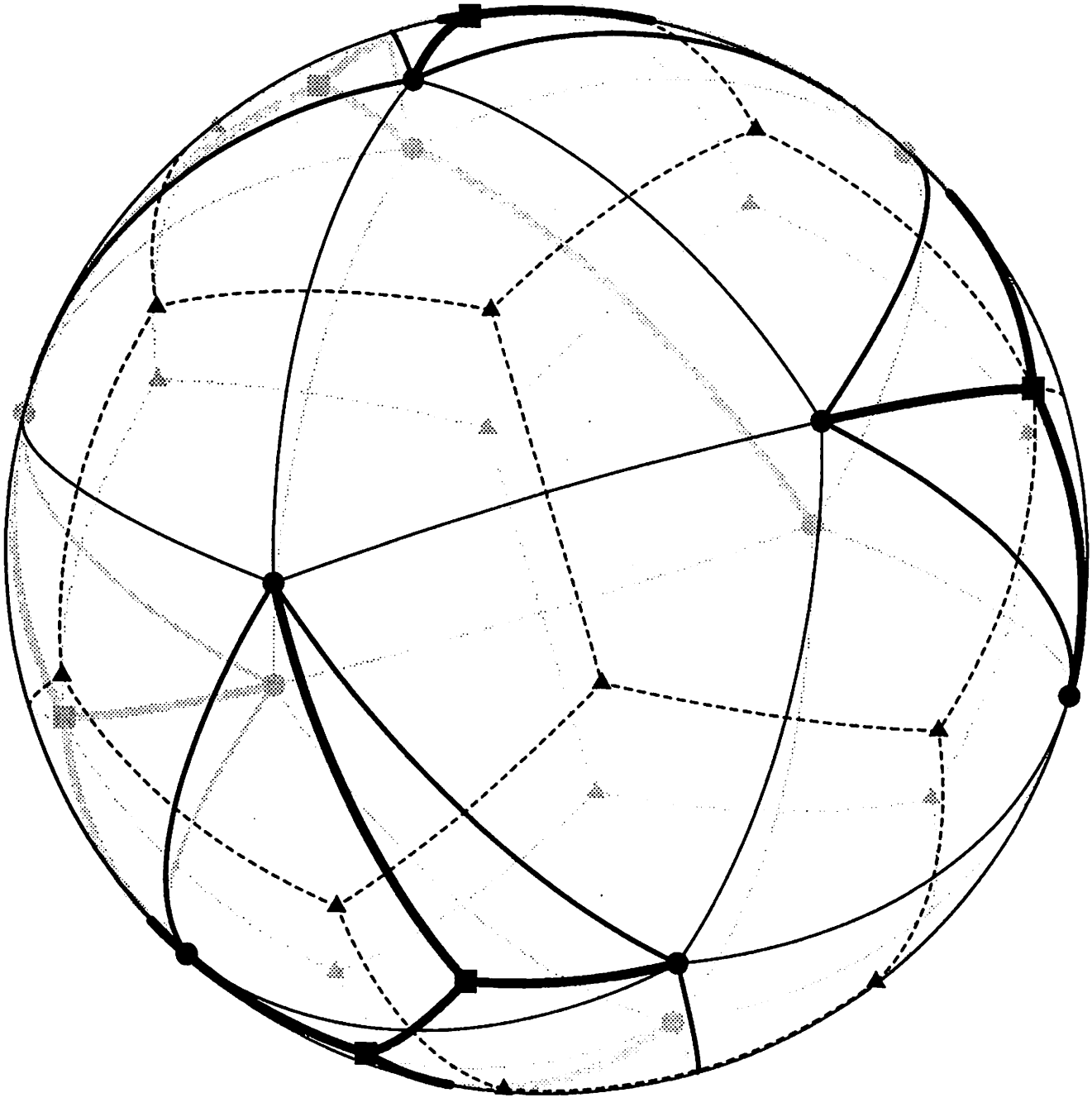
[WINT87] Winter, P., 1987. "Steiner Problem in Networks: A Survey." NETWORKS *17, 129-167.*

Appendix

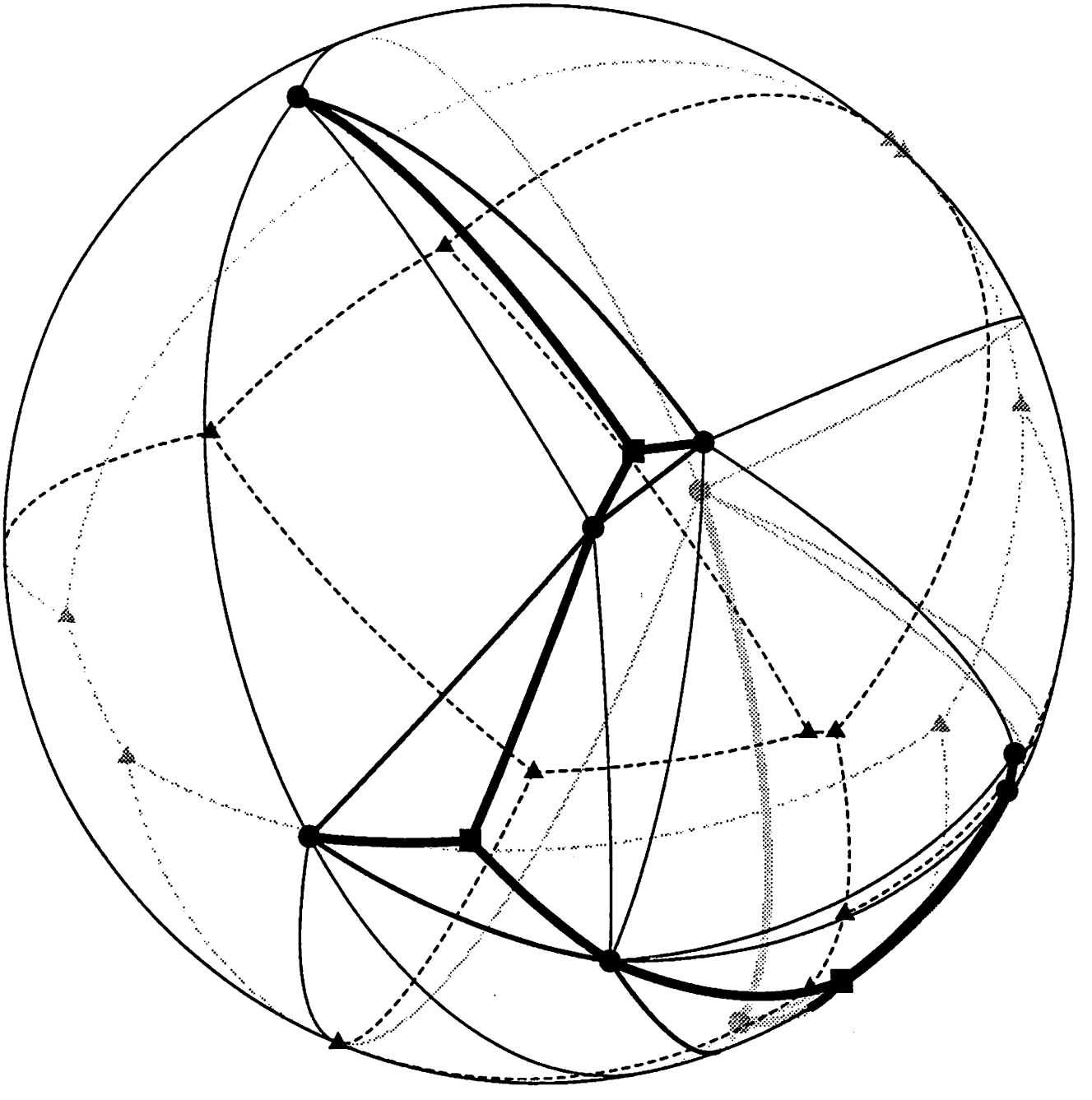
Sample Results

Key to Figures

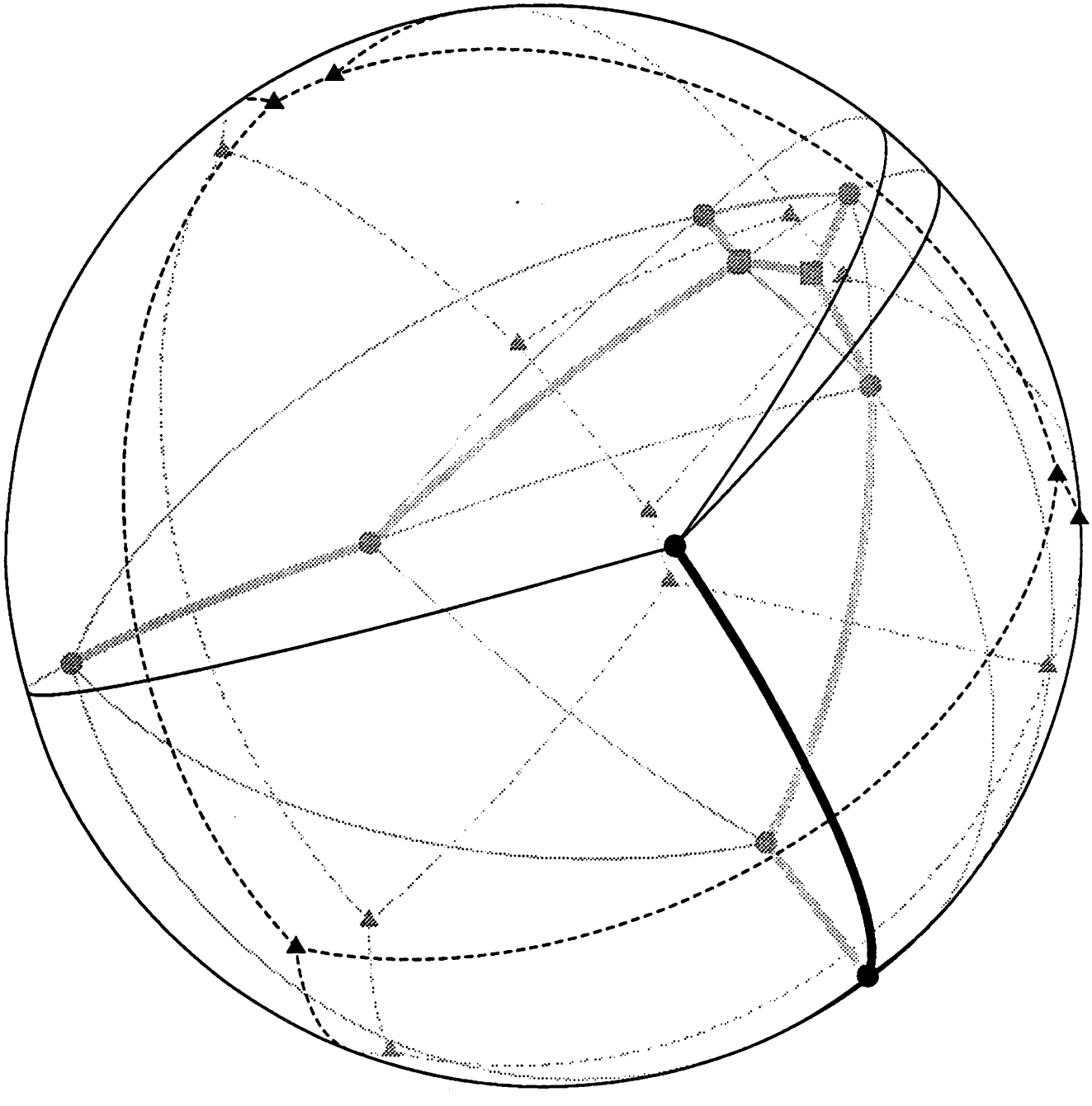




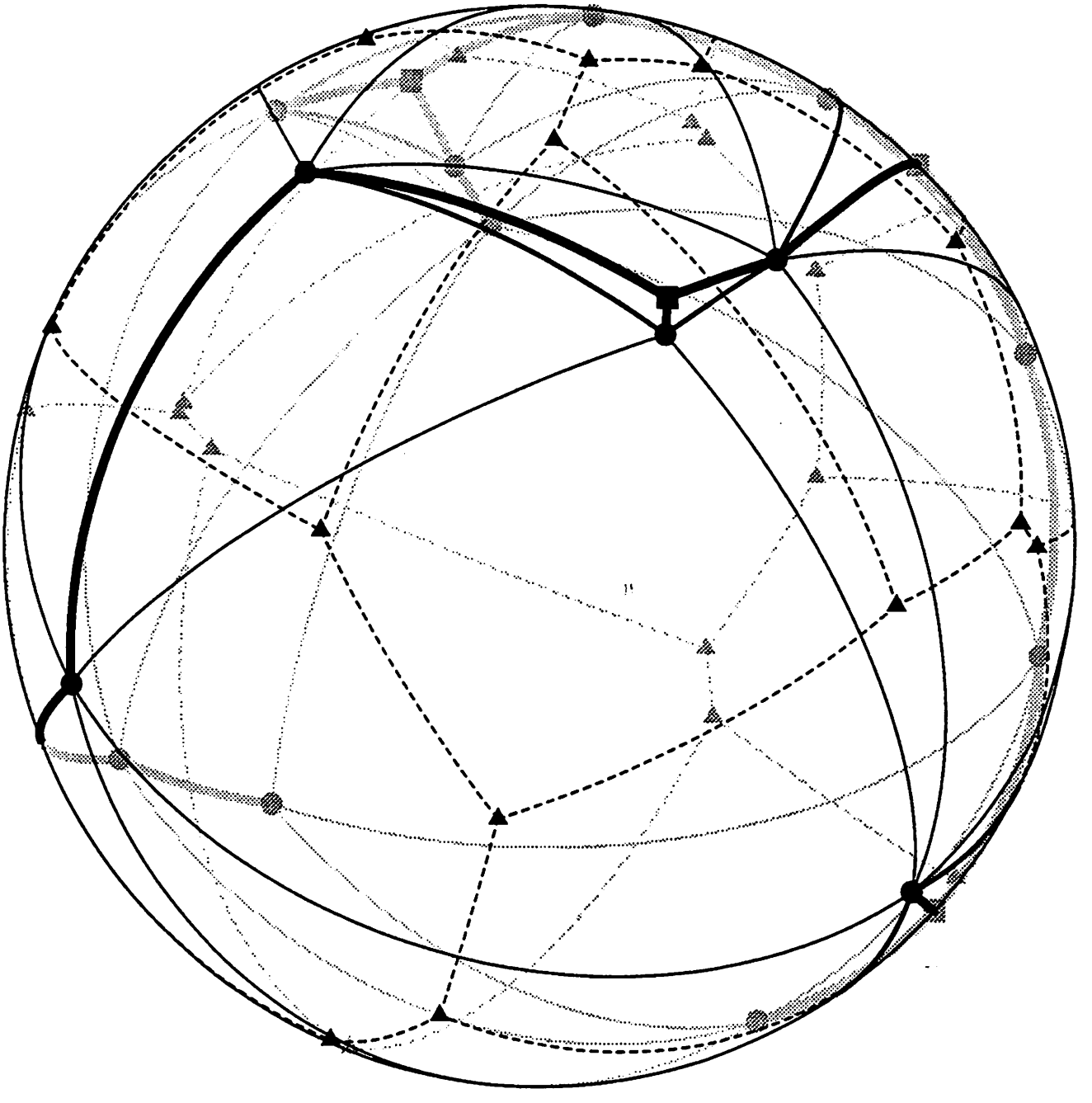
Regular Points — Icosahedron: 12 points (vertices); Statistics: cf. Row 2, Table 16



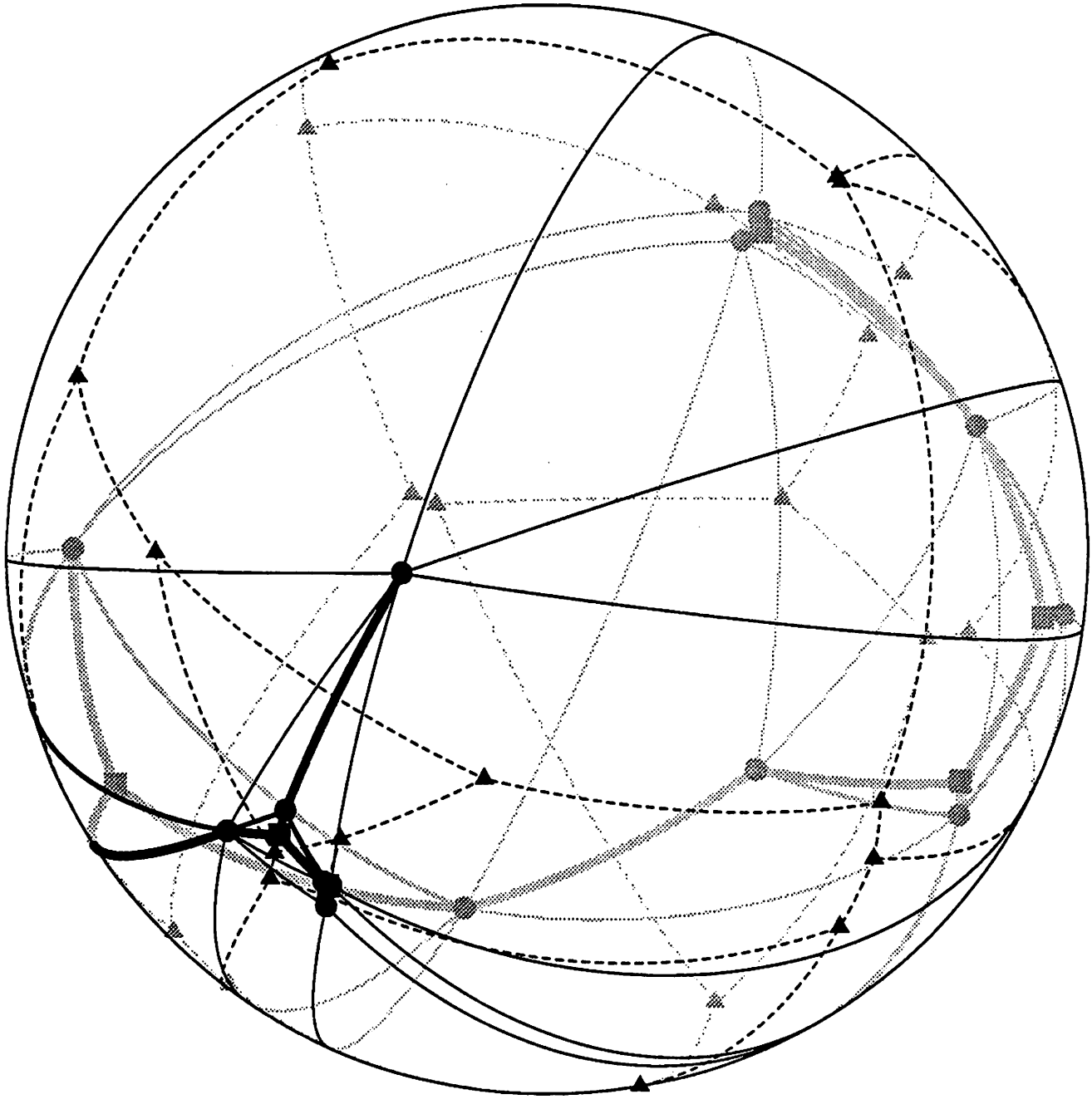
Method — Uniform ϕ, θ: 9 points; Statistics: cf. Row 3, Table 1



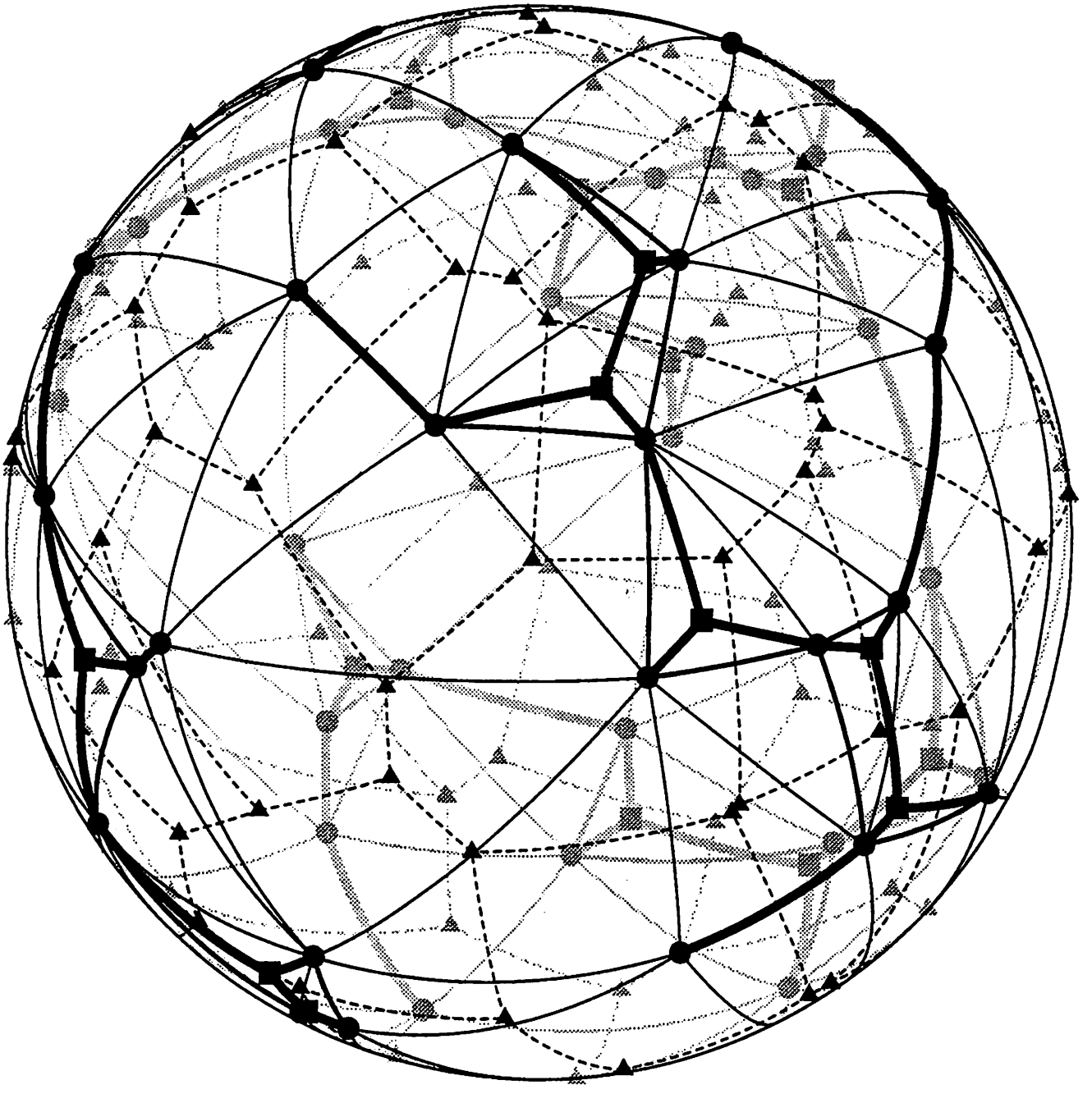
Method — Polar Bias: 9 points; Statistics: cf. Row 1, Table 6



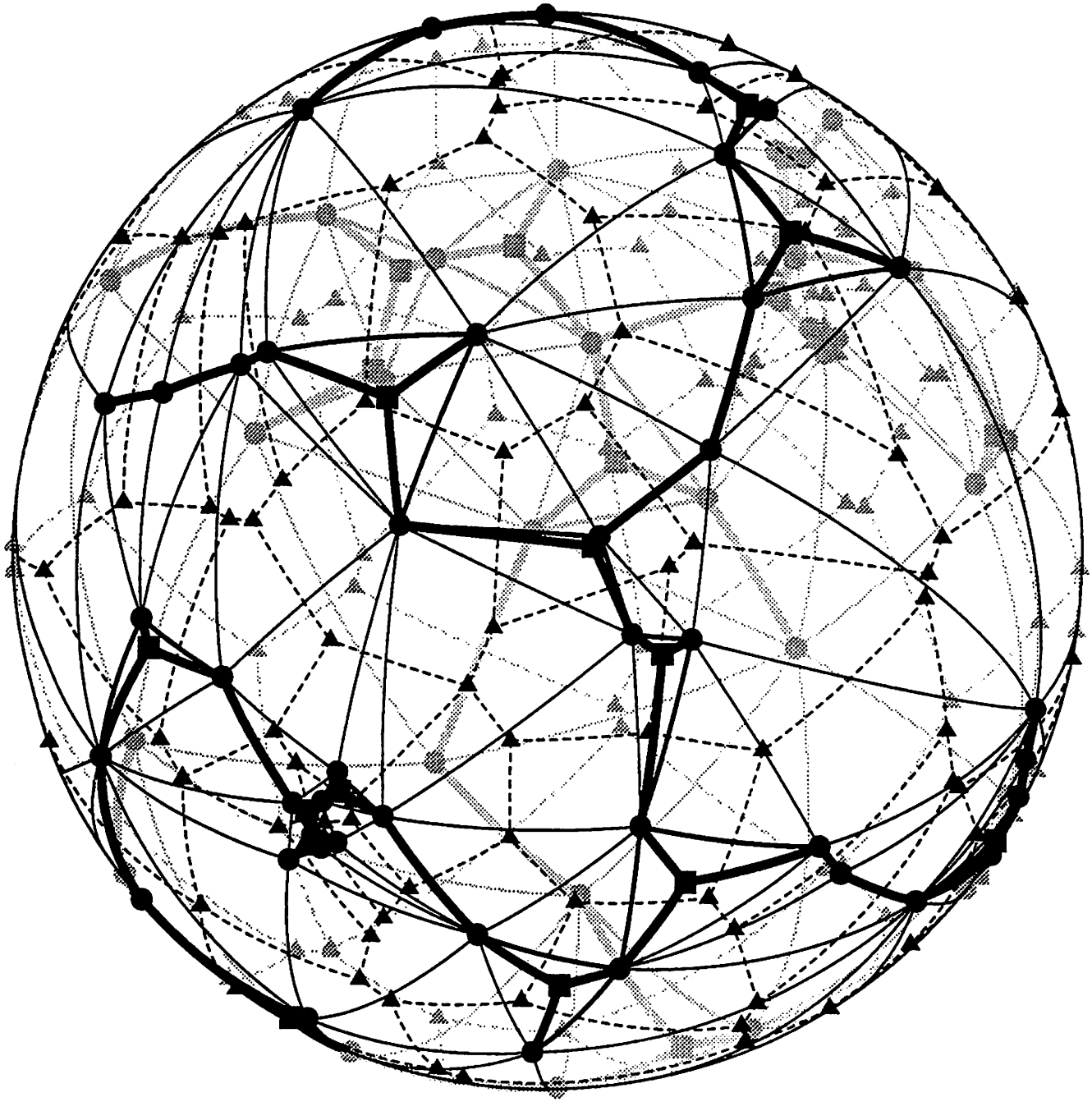
Method — Uniform Area: 15 points; Statistics: cf. Row 1, Table 12



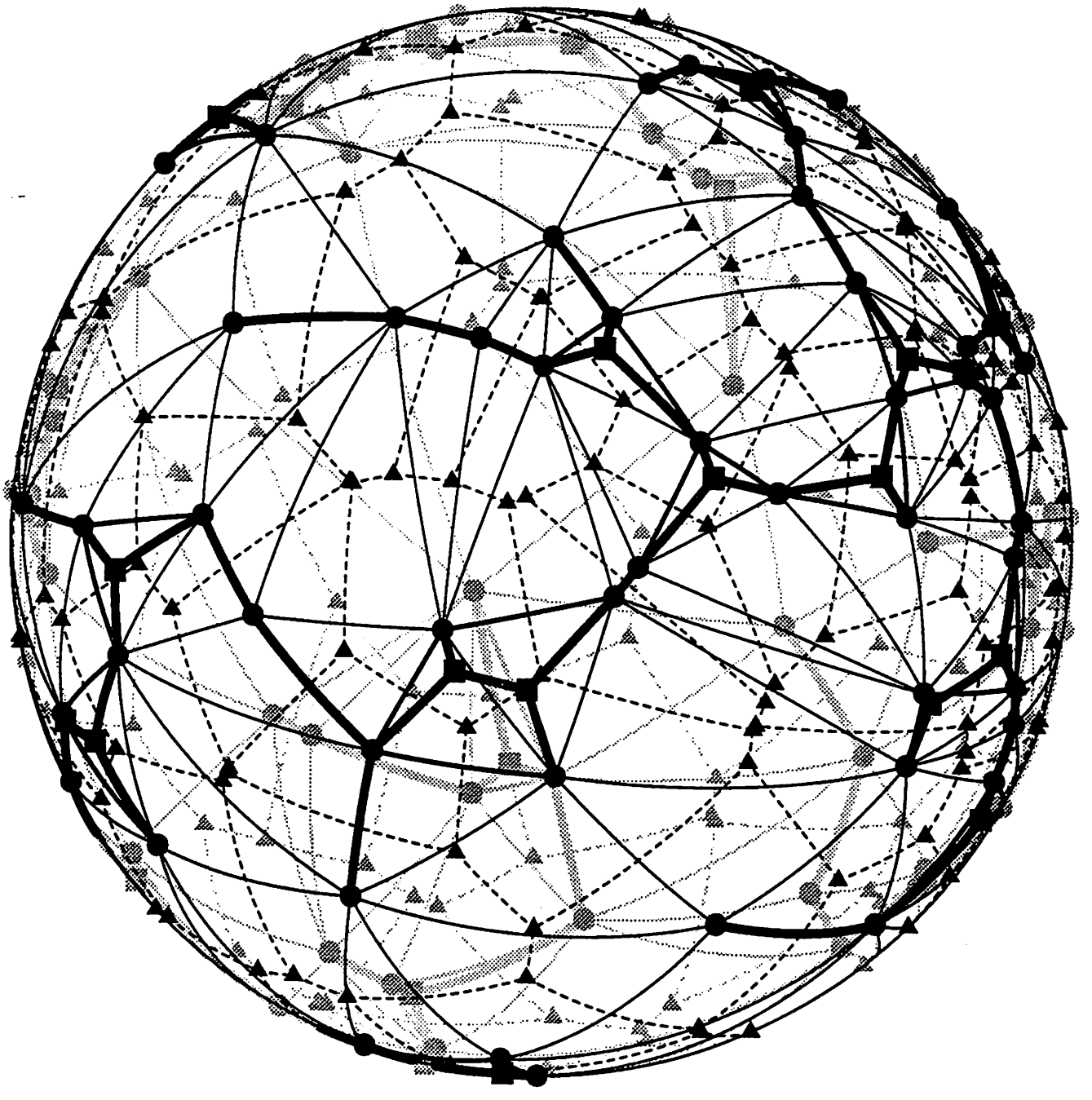
Method — Polar Bias: 15 points; Statistics: cf. Row 2, Table 7



Method — Uniform Area: 50 points; Statistics: cf. Row 3, Table 13



Method — Uniform ϕ, θ: 75 points; Statistics: cf. Row 3, Table 4



Method — Uniform Area: 100 points; Statistics: cf. Row 0, Table 15