Reconstructing convex polygons and polyhedra from edge and face counts in orthogonal projections

(Extended Abstract)

Therese Biedl[1], Masud Hasan[2], and Alejandro López-Ortiz[1]

[1] School of Computer Science, University of Waterloo, Waterloo, Ontario, Canada N2M 3G1
Email: {biedl, alopez-o}@uwaterloo.ca
[2] Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology, Dhaka-1000, Bangladesh
Email: masudhasan@cse.buet.ac.bd

Abstract. We study the problem of constructing convex polygons and convex polyhedra given the number of visible edges and visible faces from some orthogonal projections. In 2D, we find necessary and sufficient conditions for the existence of a feasible polygon of size N and give an algorithm to construct one, if it exists. When N is not known, we give an algorithm to find the maximum and minimum size of a feasible polygon. In 3D, when the directions span a single plane we show that a feasible polyhedron can be constructed from a feasible polygon. We also give an algorithm to construct a feasible polyhedron when the directions are covered by two planes. Finally, we show that the problem becomes NP-complete for three or more planes.

1 Introduction

Reconstructing polyhedra from projection information is an important field of research due to its applications in geometric modeling, computer vision, geometric tomography, and computer graphics. The nature of reconstruction problems and the techniques to solve them depend upon the types of information given, such as line drawings, silhouettes, and area/volume/shape of shadows, among others.

The computational geometry community has studied the problem of reconstructing convex polyhedra from triangulations of the shadow boundary. Marlin and Toussaint [16] gave an $O(n^2)$ algorithm for deciding whether such a polyhedron exists and constructing a polyhedron where possible. In another variation of this problem, where the triangulations are isomorphic to two opposite projections from the z-axis, Bereg [2] showed that the polyhedron can always be reconstructed. See [6] for a collection of similar problems on reconstruction of polyhedra.

Reconstructing polyhedra has also been studied from the point of view of applications, and various types of projection information have been considered. Among them line drawings [14, 15, 18–21, 24, 25] are possibly the most common. Line drawings may be obtained from images, from geometric drawings from the designers [21, Chapter 1], or may be freehand drawings [13, 23]. The reconstruction algorithms differ for a single and multiple drawings. For multiple drawings there are two common approaches based on the representation of the polyhedra to be reconstructed: constructive solid geometry and boundary representation. Both approaches are used in engineering and product design such as designing complex mechanical parts and in CAD [10, 24]. It is more difficult to construct a polyhedron from a single drawing [21, 24].

Reconstruction from the area and shape of projections has been considered in geometric tomography [8]. Usually convex objects are reconstructed here. A related but more application oriented field is computerized tomography, where 3D objects are reconstructed from sectioning information such as the area of a plane section of the objects. Medical CAT scanning is an important application of computerized tomography where an image of the human body is reconstructed from X-ray information [8]. The information achieved through X-rays gives the lengths, widths, volumes and shapes of different parts of an object, which are similar to area and shape of projections.

Instead of whole projections, sometimes only silhouettes are used to reconstruct polyhedra [4, 5, 12, 17]. In volume intersection, which is a well-known technique in computer vision, the only information available is a set of silhouettes [4, 5, 12], sometimes even with unknown view points [4, 5].

Our results Most reconstruction algorithms are based on fairly complex information such as triangulations, line drawings, silhouettes, and geometric measures of the projections, along with some non-geometric surface information such as shading, texture, and reflection of light. In contrast, we consider a very different and very limited type of information, which is also robust: we consider number of visible edges for polygons and number of visible faces for polyhedra in some orthogonal projections. Here we study reconstructing convex polygons and polyhedra from orthogonal projections only; see [9] for results on perspective projections and non-convex polygons and polyhedra.

We only consider non-degenerate orthogonal projections where the view directions are not parallel to the edges (faces) of the polygon (polyhedron). A *direction-integer pair*, or simply a *d-i pair*, $\langle d, n \rangle$ consists of a direction vector d and a positive integer n, and expresses how edges (faces) should be seen from the direction. A *d-i set* $\mathcal{R}$ is a set of d-i pairs where no two directions are the same or opposite to each other. (We assume this because we will ultimately generate and then use the d-i pairs for all opposite directions too. See Page 3.) A convex polygon (polyhedron) P is *feasible* for $\mathcal{R}$ if, for each d-i pair $\langle d, n \rangle$ in $\mathcal{R}$, d is not parallel to the edges (faces) of P and the number of visible edges (faces) from d is n. For a d-i set, a feasible polygon may or may not exist or it may exist for more than one possible number of edges (see Figure 1.)

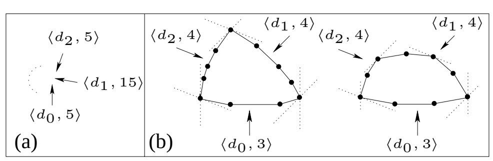

Fig. 1. (a) A d-i set with no feasible polygon. (b) Example of feasible polygons of different size.

In this paper, we consider the problem of given a d-i set $\mathcal{R}$ and an integer N, create a feasible polygon (or polyhedron) of size N for $\mathcal{R}$. Clearly, we must have $N \geq 3$ or 4, respectively, and $N > \max_i\{n_i\}$, and we assume this throughout.

We first give necessary and sufficient conditions for a feasible polygon to exist, which also gives an algorithm to construct the polygon, if it exists. With K directions, our algorithm runs in $O(K + N)$ time if $\mathcal{R}$ is ordered, and in $O(K \log K + N)$-time otherwise. For unknown N, above characterization gives an $O(K + v \log v)$-time algorithm to find the maximum and minimum size of a feasible polygon where $1 \leq v < N$.

In 3D, we consider cases by the minimum number of planes that *cover* the directions, where "covering" means each direction lies in at least one plane. For one plane, 2D results are easily transferred. For two planes, we give an algorithm to construct a feasible polyhedron, whenever it exists, except for one special case. Finally, for three or more planes, we prove that the existence of a feasible polyhedron is NP-complete.

Although from the applications point of view the problem of reconstructing polyhedra is more common than that of reconstructing polygons, surprisingly, the latter are themselves very rich and their solution techniques will serve as foundation for solving the former. Our algorithm to test feasibility of reconstruction can be useful as a preliminary step in applications in which other types of information are used, in addition, for reconstruction purposes—the user can decide quickly the existence of possible resulting polyhedra before starting a rigorous reconstruction process.

Preliminaries We defined our problems in terms of a d-i set $\mathcal{R}$, but to solve it we will use a *proper d-i set* $\mathcal{S}$ which has $2K$ d-i pairs and is derived from $\mathcal{R}$ and N as follows: For each d-i pair $\langle d, n\rangle$ in $\mathcal{R}$, $\mathcal{S}$ has both $\langle d, n\rangle$ and $\langle d', N - n\rangle$, where d' is opposite to d, and $\mathcal{S}$ has no other d-i pair. The d-i pairs $\langle d, n\rangle$ and $\langle d', N - n\rangle$ in $\mathcal{S}$ are called *opposite* to each other. We omit the (straightforward) proof of the following lemma.

Lemma 1. *A convex polygon (polyhedron) P with N edges is feasible for $\mathcal{R}$ if and only if it is feasible for $\mathcal{S}$.*

When directions of $\mathcal{S}$ lie in one plane, $\mathcal{S}$ is represented as $\mathcal{S} = \{\langle d_0, n_0\rangle, \langle d_1, n_1\rangle, \ldots, \langle d_{2K-1}, n_{2K-1}\rangle\}$, where the d-i pairs are ordered counter-clockwise by directions, and from now on indices of the terms related to $\mathcal{S}$ are taken modulo $2K$.

2 Reconstructing polygons

Let P be a feasible polygon of size N for $\mathcal{S}$. Consider the sets of visible edges of P from the directions of $\mathcal{S}$. When we move from a direction d_i to d_{i+1}, there may be some edges of P that become *newly visible* and/or *newly invisible* to d_{i+1}. From n_i and n_{i+1} alone, it cannot be said exactly how many edges become newly visible or invisible to d_{i+1}. However, it is possible to lower bound these quantities. Observe that if an edge e becomes newly visible when going from d_i to d_{i+1}, then it becomes newly invisible when going from d_{i+K} to d_{i+K+1}. This implies that although the change in the visibility of each edge happens twice, the total change in the visibility for all edges can be counted by only considering their change from invisible to visible. (This use of opposite directions is the main motivation to consider the proper d-i set $\mathcal{S}$ instead of the d-i set $\mathcal{R}$.) Moreover, e is newly visible for exactly one direction of $\mathcal{S}$.

We now state the characterization formally. For each i, define $\delta_i = \mathbf{max}\{0, n_{i+1} - n_i\}$. We call δ_i the *i-th view difference*. There must be at least δ_i edges that become newly visible while moving from d_i to d_{i+1}. Therefore if a polygon exists, then $D := \sum_{i=0}^{2K-1} \delta_i \leq N$. Our main result here is that this necessary condition is also sufficient.

Theorem 1. *Given a proper d-i set $\mathcal{S}$ and an integer N, a feasible polygon P of size N exists if and only if $D \leq N$.*

Proof. The proof starts with the following crucial lemma.

Lemma 2. *For any i, $n_i - \sum_{j=i+K}^{i-1} \delta_j = \frac{1}{2}(N - D)$.*

Proof. We have $n_{j+1} - n_j = \delta_j - \delta_{j+K}$, since $\delta_{j+K} = \max\{0, n_{j+K+1} - n_{j+K}\} = \max\{0, (N - n_{j+1}) - (N - n_j)\} = \max\{0, n_j - n_{j+1}\}$. Using this for K times gives $n_{i+K} - n_i = \sum_{j=i}^{i+K-1} \delta_j - \sum_{j=i+K}^{i-1} \delta_j$. Using $n_{i+K} = N - n_i$ and subtracting $D = \sum_{j=0}^{2K-1} \delta_j$, this becomes $N - 2n_i - D = -2 \sum_{j=i+K}^{i-1} \delta_j$ as desired. □

In particular this shows that $N - D$ is even. The idea for the proof is now as follows. For each view direction d_i, choose δ_i edges, if $\delta_i > 0$, such that they are newly visible for d_{i+1}. The remaining $N - D$ edges are chosen in antipodal pairs so that one becomes visible exactly when the other becomes invisible. To avoid constructing an unbounded polygon we have to be careful in how to chose edges.

Placing the normal-points Instead of choosing edges directly, we will choose a normal-point for each edge on a circle c centered at the origin o. Let P be an arbitrary convex polygon. The *normal-point* of an edge e of P is the intersection of c with the outward normal of e, translated to the origin. Conversely, any set of points on c can be converted to a polygon by computing the intersection of the tangent lines at those points.

To explain how the normal-points can be placed for each direction, we need some more notations. For any direction d_i, denote by h_i the *visible half-circle of* d_i, i.e., the (closed) half-circle of c that is visible from d_i. Clearly e is visible from d_i if and only if its normal-point is strictly within h_i. Moreover, a polygon defined by normal-points is bounded if and only if not all normal-points are within a single open half-circle.

The arc $\theta_i = h_{i+1} \backslash h_i$ is called the *i-th d-arc* ("d" for difference). Normal-points in θ_i correspond to edges that are newly visible to d_{i+1}. Normal-points will never be placed on the boundary of θ_i, and hence we will not distinguish as to whether θ_i is open or closed. Observe that θ_i and θ_{i+K} are the reflections of each other with respect to the origin and are called *opposite* to each other. (See Figure 2(a)). Since d_i and d_{i+K} are opposite directions, we have $\bigcup_{j=i-K}^{i-1} \theta_j = h_i$ for all i. (See also Figure 2(b)).

Now place δ_i arbitrary normal-points strictly within each θ_i. If $D < N$, then by Lemma 2, $N - D$ is even. Select $N - D - 2$ additional normal-points in antipodal pairs arbitrarily (but not on end-points of any θ_i) and the remaining two normal-points p_1 and p_2 as follows. Let p be one among $N - 2$ already selected normal-points. Let p' be the opposite of p. Select p_1 at clockwise ε (circular) distance apart from p and p_2 at clockwise $\varepsilon/2$ distance apart from p'. ε is small enough so that p_1 and p_2 are within two opposite d-arcs. See Figure 2(c).

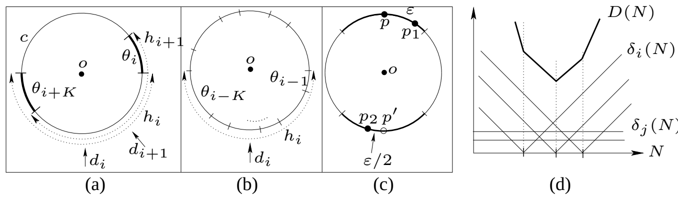

Fig. 2. (a) Visible half-circles, and d-arcs; two opposite d-arcs are in bold. (b) $h_i = \bigcup_{j=i-K}^{i-1} \theta_j$. (c) Selecting the last two normal-points when $D < N$. (d) δ_i and $D(N)$ against unknown N.

Correctness Two things need to be proven: (1) Each d_i sees n_i normal-points, and (2) the polygon is bounded, i.e., every open half-circle gets at least one normal-point. The first holds since each pair among the $N - D$ normal-points chosen last goes into two opposite d-arcs. So of each pair exactly one is strictly within h_i. Since $h_i = \bigcup_{j=i-K}^{i-1} \theta_j$, the number of normal-points that are strictly within h_i is $\sum_{j=i-K}^{i-1} \delta_j + \frac{1}{2}(N-D)$, which by Lemma 2 is n_i. To see (2), consider two cases. If $D < N$, then p_1 and p_2 were chosen such that the minimum circular distance between any two of p, p_1 and p_2 is less than a half-circle, and (2) holds. If $N = D$, then each d-arc θ_i gets exactly δ_i normal-points. Any open half-circle h intersects $K - 1$ d-arcs fully, and we claim that $\delta_j > 0$ for one of them. For if not, then using $\min\{\delta_i, \delta_{i+K}\} = 0$ and adding the adjacent d-arc which achieves 0, we get K consecutive d-arcs without normal-points. Say $\sum_{j=i}^{i+K-1} \delta_i = 0$, then $n_{i+K} = \sum_{j=i}^{i+K-1} \delta_i + \frac{1}{2}(N - D) = 0 + 0 = 0$, a contradiction. □

Corollary 1. *P exists if and only if each θ_i gets at least δ_i normal-points.*

The above proof is algorithmic, and it is straightforward how to implement it in $O(N + K)$ time if $\mathcal{S}$ is ordered, and in $O(N + K \log K)$ otherwise. We summarize,

Theorem 2. *Given a d-i set $\mathcal{R}$ of size K and given an integer N, a feasible polygon P with N edges can be computed, whenever it exists, in $O(N + K)$ time when $\mathcal{R}$ is ordered, or in $O(N + K \log K)$ time otherwise.*

Maximum and minimum polygon Using Theorem 1, we can also find out whether there exists a feasible polygon for a given d-i set $\mathcal{R}$ even if N is unknown. In fact, we find both the maximum and minimum size of a feasible polygon. Observe that if $\mathcal{R}$ contains two opposite d-i pairs, then the sum of the two integers would give the value of N. Hence, once again it is assumed that no opposite d-i pair appears in $\mathcal{R}$.

The overall idea is as follows. We compute as before a proper d-i set $\mathcal{S}(N)$ from $\mathcal{R}$, but this time the d-i pairs of $\mathcal{S}(N)$ will be functions of N—for each pair $\langle d, n \rangle$ in $\mathcal{R}$, the opposite pair $\langle d', N - n \rangle$ in $\mathcal{S}$ contains the unknown N. We call $\langle d, n \rangle$ *original* and $\langle d', N - n \rangle$ *derived*. Then we compute $\delta_i(N)$ and $D(N)$, which also become functions of N. Recall from Theorem 1 that a feasible polygon exists if and only if $D(N) \le N$.

Analyzing cases, one can observe that the function $\delta_i(N)$ is either a constant or a V-shape with slopes ± 1 for which the tip (with $\delta_i(N) = 0$) occurs at a place well-defined in terms of n_i, n_{i+1} and whether d_i and d_{i+1} are original and derived respectively. (See

[9] for detail.) Hence the function $D(N)$, which is the sum of these, is convex and piecewise linear. See also Figure 2(d). So $D(N) = N$ has at most two solutions, and any N between them is feasible as long as $N \geq 3$ and $N \leq \max_i\{n_i\}$. The algorithm to compute this range of N takes $O(K + v \log v)$ time, where v is the number original d-i pairs in $\mathcal{S}(N)$ whose corresponding next d-i pairs are derived. Of course $v \in O(K)$, but v could be as small as one if all directions in $\mathcal{R}$ are spanned within a half-plane.

Theorem 3. *Given an ordered d-i set $\mathcal{R}$ of size K, the maximum and minimum size of a feasible polygon can be computed in $O(K + v \log v)$ time, where v is the number of original d-i pairs in $\mathcal{S}(N)$ whose corresponding next d-i pairs are derived. If $\mathcal{R}$ is not ordered, then the algorithms takes $O(K \log K)$ time.*

Corollary 2. *For any value of N between its maximum and minimum, there exists a feasible polygon of size N.*

3 Reconstructing polyhedra

Similar to 2D, in order to construct a feasible polyhedron P we will compute the proper d-i set $\mathcal{S}$ from the given d-i set $\mathcal{R}$ and instead of choosing faces directly we will choose them implicitly by choosing normal-points of the faces on the surface of an origin-centered sphere s. Then given such normal-points, we can compute a polyhedron from them by computing the intersection of their tangent half-planes in $O(N \log N)$ time [7].

A face f is visible from a direction d_i if and only if its normal-point is strictly within the *visible hemisphere* h_i of d_i. Moreover P is bounded if and only if not all normal-points intersects a single open hemisphere.

3.1 Directions covered by a single plane

It is natural to interpret $\mathcal{S}$ as input to the 2D case. In fact we show that feasibility of the 3D problem is equivalent to that of 2D, and all results from 2D then transfer.

We need some notations first, which will be used both here and in the next section. Given a proper d-i set $\mathcal{S}$ with directions in one plane, we define $\theta_i = h_{i+1} \backslash h_i$ and call it *i-th d-lune* of $\mathcal{S}$. See Figure 3(a). All d-lunes of $\mathcal{S}$ have two common antipodal points which are called *poles* of $\mathcal{S}$. As in 2D, $h_i = \bigcup_{j=i-K}^{i-1} \theta_j$.

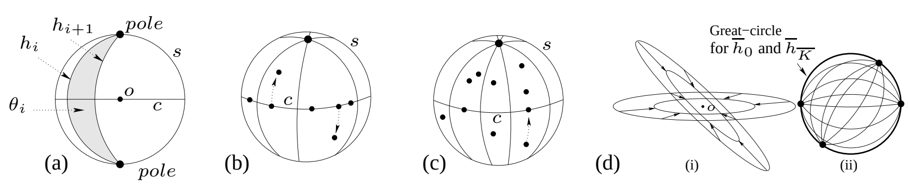

Fig. 3. (a) Visible half-sphere and d-lune. (b) P from p. (c) p from P. (d) (i) Two planes of $\mathcal{S}$ with common directions, and (ii) arrangement of the d-lunes for such $\mathcal{S}$.

Theorem 4. *Given an ordered proper d-i set $\mathcal{S}$ of size $2K$, where all the directions lie in one plane, and given $N \geq 4$, there exists a feasible polyhedron P for $\mathcal{S}$ of size N if and only if there exists a feasible polygon p for $\mathcal{S}$ of size N. Moreover, the time required to construct P from p is $O(N \log N)$ and p from P is $O(N)$.*

Proof (Sketch only). Let c be the great circle of s corresponding to the plane of directions π. Assume first that p exists and we want to construct P. Consider the normal-points of the edges of p on s; all of them are on c. If we create a polyhedron P with them, then it would be a cylinder with two ends unbounded. To make it bounded, we need to incline two of its faces slightly in "opposite" directions. This can be done by moving two normal-points towards the two poles of $\mathcal{S}$, respectively, but within their respective d-lunes. See Figure 3(b).

Now we show how to construct p if P exists. Find the normal-points of the faces of P. Move each of them onto c along the great-circle through the point and the poles, using the shorter arc. See Figure 3(c). If points overlap after the movement, then move them slightly but within their respective d-lunes and keeping them on c. Now all normal-points are within a plane, and we can construct a polygon from them in $O(N)$ time. □

Corollary 3. *Given an ordered proper d-i set $\mathcal{S}$ of size $2K$, where the directions lie in one plane, (i) given N, the necessary and sufficient conditions for the existence of a feasible polyhedron P of size N is $D \leq N$, and (ii) with unknown N, the maximum and minimum size of P can be found by the same algorithm for a feasible polygon.*

3.2 Directions covered by two planes

For a feasible polyhedron to exist, it is of course necessary that it exists for each plane of directions separately. But a little exercise will show that this is not sufficient. We give here an algorithm that constructs a feasible polyhedron, whenever it exists, except for the case when directions may see less than four faces and for each plane of directions the sum of the view differences is N, for which we show why our algorithm fails.

Let $\overline{\mathcal{S}}$ and $\widetilde{\mathcal{S}}$ be the two subsets of $\mathcal{S}$ based on the two planes of directions respectively. (The markings $^-$ and $^\sim$ corresponds to $\overline{\mathcal{S}}$ and $\widetilde{\mathcal{S}}$, respectively). The size of $\overline{\mathcal{S}}$ and $\widetilde{\mathcal{S}}$ are $2\overline{K}$ and $2\widetilde{K}$ and any indices related to $\overline{\mathcal{S}}$ and $\widetilde{\mathcal{S}}$ will be taken modulo $2\overline{K}$ and $2\widetilde{K}$, respectively. The d-lunes of $\overline{\mathcal{S}}$ intersects those of $\widetilde{\mathcal{S}}$ and thus divides s into spherical polygons called *d-polygons*; we denote $\theta_{a,b} = \overline{\theta}_a \cap \widetilde{\theta}_b$. Note that $\overline{\mathcal{S}}$ and $\widetilde{\mathcal{S}}$ can share either two opposite directions or none. See Figure 3(d)(i).

Our algorithm has two phases: finding a *valid assignment* and then finding a *valid selection*. A valid assignment assigns N normal-points to the d-polygons such that for each d-i pair $\langle d, n \rangle$ in $\overline{\mathcal{S}}$ or $\widetilde{\mathcal{S}}$, the number of normal-points assigned to d-polygons in the visible hemisphere of d is n. Note that there may be more than one valid assignment. A valid assignment does not give positions for the normal-points, which may decide whether P is bounded or not. So if the feasible polyhedron is allowed to be unbounded, then the existence of a valid assignment is necessary and also sufficient for the existence of a feasible polyhedron. For a bounded one, the actual positions of the normal-points are important, and for that we need a valid selection—which is to select the normal-points within their respective d-polygons (as assigned by a valid assignment) such that not all normal-points intersect a single hemisphere.

Finding a valid assignment Note that the arrangement of the d-polygons resembles cells of a matrix (see also Figure 3(d)(ii)). Finding a valid assignment hence resembles assigning numbers to matrix positions such that row-sums and column-sums satisfy conditions. This has been studied a lot under the name of *transportation problem* [1, 11]. We will need the following result, whose easy proof we omit:

Lemma 3. *Let $R_1, \ldots, R_m$ and $C_1, \ldots, C_n$ be non-negative integers with $\sum_{i=1}^{m} R_i \leq \sum_{j=1}^{n} C_j$. Then there exist non-negative integers $(M_{i,j})_{1\leq i\leq m, 1\leq j\leq n}$ such that $\sum_{j=1}^{n} M_{i,j} = R_i$ and $\sum_{i=1}^{m} M_{i,j} \leq C_j$. Moreover, we can find the positive integers among them in $O(m+n)$ time.*

We have two cases: either the two planes have common view direction or not. In the former case there always exists a valid assignment and we will find one. In the latter case we will find a valid assignment whenever it exists. In both cases, we will also create two opposite d-polygons with at least one normal-points each if $\mathbf{max}\{\overline{D}, \widetilde{D}\} < N$, which will be needed for the valid selection process. The following lemma summerizes the selection process and its proof is nothing but to adapt Lemma 3 several times.

Lemma 4. *Given $\overline{\mathcal{S}}$ and $\widetilde{\mathcal{S}}$, if they have common directions, then it is always possible in $O(\overline{K} + \widetilde{K})$ time to find a valid assignment. If $\overline{\mathcal{S}}$ and $\widetilde{\mathcal{S}}$ do not have any common directions, then it is possible to find a valid assignment, whenever it exists, in $O(\overline{K} + \widetilde{K})$ time. Moreover, in both cases if $\mathbf{max}\{\overline{D}, \widetilde{D}\} < N$, the resulting valid assignment gives two opposite d-polygons which are positive.*

Finding a valid selection Again we have cases: (i) $\mathbf{max}\{\overline{D}, \widetilde{D}\} < N$ and (ii) all directions see at least four faces. In both cases it is always possible to find a valid selection. If some direction sees less than four faces and if $\max\{\overline{D}, \widetilde{D}\} = N$, then there exists a polyhedron that has valid assignment but no valid selection and we do not know the complexity of testing whether a valid selection exists, but we show an example in Page 9 for that.

Lemma 5. *If $\mathbf{max}\{\overline{D}, \widetilde{D}\} < N$, then we can find a valid selection in $O(N)$ time.*

Now consider the case when $\mathbf{max}\{\overline{D}, \widetilde{D}\} = N$, and each direction sees at least four faces. This case is significantly more complicated. The overall idea is based on the following lemma, which we state without proof:

Lemma 6. *Consider any three great circles of s that do not intersect in a common pair of antipodal points. These three great circles divide s into eight octants. For each octant consider an arbitrary point that is strictly within it. Then these eight points cannot intersect a hemisphere of s.*

We choose these three great circles as follows. One is the great-circle g^* that contains the four poles. Then for each $\overline{\mathcal{S}}$ and $\widetilde{\mathcal{S}}$, we pick another great-circle $\overline{g}$ and $\widetilde{g}$ through the respective poles. The main obstacle here is to choose them such that each octant in fact is allowed to have a normal-point in it, even with changing the valid assignment to another one. In particular, we must choose $\overline{g}$ such that the four lunes defined by g^* and $\overline{g}$ contain at least two normal-points each (and similarly for $\widetilde{g}$.)

Lemma 7. *Given a proper d-i set $\mathcal{S}$ of size $2K$, where the directions are in one plane, $n_i \geq 4$ for any $\langle d_i, n_i \rangle$, and $D = N$. Then for any great circle g^* passing through the poles, we can find in $O(K)$ time another great circle g also passing through the poles such that the four lunes created by g^* and g contain at least two normal-points each, after a suitably distribution of normal-points in the d-lunes intersected by g^* and g.*

Lemma 8. *Given $\overline{\mathcal{S}}$ and $\widetilde{\mathcal{S}}$, where $n \geq 4$ for any d-i pair $\langle d, n \rangle$ in $\overline{\mathcal{S}}$ or $\widetilde{\mathcal{S}}$, and given a valid assignment for $\mathcal{S}$, a valid selection, possibly with a different valid assignment, can be found in $O(N + \overline{K} + \widetilde{K})$ time*

The following theorem summarizes the results, where the term $O(N \log N)$ in the time complexity comes from the intersection of the half-spaces defined by the planes passing tangents to the normal-points.

Theorem 5. *Given a proper d-i set $\mathcal{S}$ and an integer $N \geq 4$, where the directions of $\mathcal{S}$ are covered by two planes. We can construct a feasible polyhedron P, if it exists, in $O(N \log N + |\mathcal{S}|)$ time, in each of the following cases: (i) $\max\{\overline{D}, \widetilde{D}\} < N$, or (ii) $n \geq 4$ for each d-i pair $\langle d, n \rangle$ in $\mathcal{S}$.*

Corollary 4. *If the feasible polyhedron is allowed to be unbounded, a feasible polyhedron can always be constructed, when it exists, in $O(N \log N + |\mathcal{S}|)$ time by finding a valid assignment only.*

Insufficiency of a valid assignment Our example is in Figure 4. Consider the proper d-i set $\mathcal{S}'$ of (a). It has twelve d-i pairs and $N = 4$. The only positive view differences are $\delta_0 = 1$, $\delta_4 = 1$, and $\delta_8 = 2$. So $D = N$, and by Corollary 3 there always exists a feasible polyhedron for $\mathcal{S}'$. The key property of $\mathcal{S}'$ is that the positive d-lunes are very "thin" Moreover, the circular distance between consecutive directions can be adjusted to increase/decrease the circular distance among the positive d-lunes. For example, (b) and (c) show the positive d-lunes for two different versions $\overline{\mathcal{S}}$ and $\widetilde{\mathcal{S}}$ of $\mathcal{S}'$. Now, consider

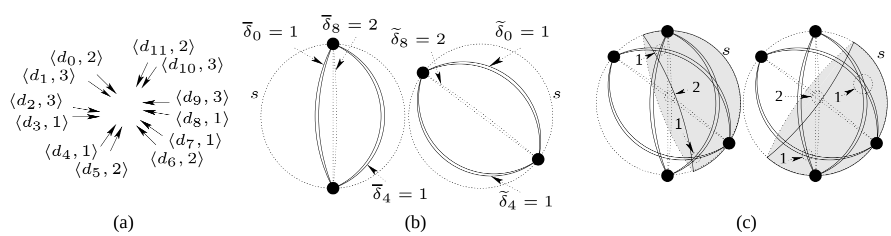

Fig. 4. Example Insufficiency of a valid assignment

the proper d-i set $\mathcal{S} = \overline{\mathcal{S}} \cup \widetilde{\mathcal{S}}$. There are only two possible valid assignments for the d-polygons of $\mathcal{S}$ which are shown in (c). But in either case all three positive d-polygons are strictly within a single hemisphere (shaded area).

4 NP-hardness for arbitrary directions

We will prove that the problem of finding a valid assignment, which is necessary for two or more planes, is NP-complete for three planes.

Theorem 6. *Given a proper d-i set $\mathcal{S}$ of size $2K$ with three panes of directions, it is NP-complete to decide the existence of a feasible polyhedron for $\mathcal{S}$.*

Proof (Sketch only). We will reduce, three consecutive reductions, the problem of finding an independent set (IS) of size k, for an arbitrary k, in a 2-edge connected cubic planar graph G, which is proven to be NP-complete in [3]. An *independent set* I of G is a set of vertices s.t. no two vertices of I are connected by an edge. See Figure 5.

First reduction We convert G to another planar graph G^H by replacing each of its edge by a chain of size three. Now, finding an IS I of size k in G is equivalent to finding an IS I^H of size $k + m$ in G^H, where m is the number of edges in G. Given I, I^H is the vertices of I plus one degree-two vertex from each chain. Conversely, given I^H, if it contains both the degree-three vertices of some chain, then it is possible to change I^H to replace one degree-three vertex with a degree-two vertex of the same chain. Then the set of degree-three vertices in I^H gives I. Since for each chain, at most one degree-two vertex is in I^H, I has size at least k.

Second reduction Given G^H, we create a set $\mathcal{L}$ of lines in three directions. First we draw G^H using three directions as follows. We place all degree three vertices G^H in a horizontal line. Since G is a 2-edge connected cubic planar graph, it is 3-edge colorable (by four color theorem [22].) For each color, use a distinct pair of direction to draw the corresponding chains of G^H such that no three vertices of G^H are collinear. Now, extend all the edges of G^H to lines and for each degree-two vertex add an extra line in the third direction. Adjust one direction, if necessary, such that no three lines intersect a single point except at the vertices of G^H. That ends the construction of $\mathcal{L}$. Observe that this construction takes polynomial time. The only time consuming operation is to find a suitable place for the drawing of a subsequent chain, for which we can always maintain a place far enough from the "so-far-completed" construction of $\mathcal{L}$. Observe also that the number of lines in each direction is the same.

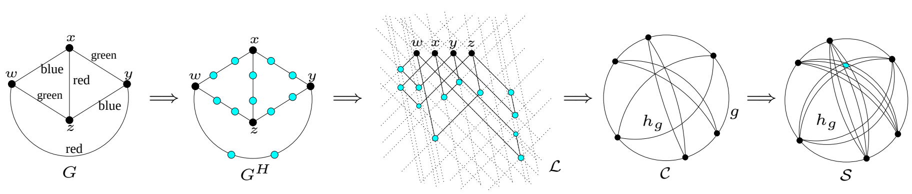

Fig. 5. Reductions for NP-completeness.

Now the problem of finding an IS I^H of size k in G^H becomes equivalent to the following problem: given $\mathcal{L}$, find a set of points $\mathcal{T}$ such that (a) each line of $\mathcal{L}$ intersects exactly one point of $\mathcal{T}$, (b) each point of $\mathcal{T}$ intersects either one or three lines of $\mathcal{L}$, and (c) the number of points in $\mathcal{T}$ that intersect three lines is k. The proof is as follow. Given I^H of size k, pick the points that correspond to the vertices of I^H and put them into $\mathcal{T}$. Since no two vertices of I^H intersect a single line of $\mathcal{L}$, the vertices of I^H intersect exactly k lines from each direction. For each of the remaining lines (who does not have any point picked up yet) pick an arbitrary point that does not intersect any other line and put it into $\mathcal{T}$. That's all in $\mathcal{T}$. Clearly each line has exactly one point in $\mathcal{T}$ and the number of tri-valent points in $\mathcal{T}$ is k. On the other hand, given $\mathcal{T}$, set I^H to be the set of vertices of G^H that correspond to the tri-valent points of $\mathcal{T}$. Since each line of $\mathcal{L}$ intersects exactly one point of $\mathcal{T}$, I^H is an independent set G^H, and since the number of tri-valent vertices in $\mathcal{T}$ is k, the size of I^H is k.

Third reduction We create a set $\mathcal{C}$ of $|\mathcal{L}|+1$ great-circles by two simultaneous mapping of $\mathcal{L}$ onto two opposite hemispheres of s—take $|\mathcal{L}|$ great-circles corresponding to the lines of $\mathcal{L}$ and take one more great-circle g corresponding to the plane of $\mathcal{L}$. Great-circles corresponding to the parallel lines in one direction intersect in two distinct poles. g passes through all six poles, and now $\mathcal{T}$ is to be selected from one hemisphere of s.

Next we find a proper d-i set $\mathcal{S}$ and the integer N from $\mathcal{C}$. Remember that the number of lines in $\mathcal{L}$ in each direction, and so the number of great-circles except g for each pair of poles, is the same. Let this number be k'. We set $N = 6k' - 4k$ (k is the number of trivalent vertices, except the poles, in $\mathcal{C}$). We now create the directions of $\mathcal{S}$. For each great-circle c of $\mathcal{C}$, we create two great-circles *very* close in two sides of c. For each of the newly created great-circles and for g, we create a pair of opposite directions in $\mathcal{S}$ so that their visible hemispheres are those defined by this great-circle. Observe that in $\mathcal{S}$ the total number of directions created is $4k' + 6$, the directions are covered by three planes, and the two directions due to g are common to each plane. Moreover, the d-lunes of $\mathcal{S}$ are of two types: *thin* d-lunes due to the close direction pairs and *non-thin* d-lunes due to the other pairs. Thin d-lunes are further categorized as *boundary* or *non-boundary* if they are due to g or not. We now assign the integers of $\mathcal{S}$. Assume that h_g and h'_g are the two hemispheres of g. Set each of the two integers associated with the two directions due to g as $\frac{1}{2}N = 3k' - 2k$. The remaining integers are such that all non-thin d-lunes have view difference zero, all non-boundary thin d-lunes have view difference one, and the boundary thin d-lunes have view difference such that the sum of the view differences of all d-lunes in h_g is $3k' - 2k$. This assignment will imply that the sum of the view differences of all d-lunes in h'_g is also $3k' - 2k$.

Now, given $\mathcal{T}$, we will only show that each non-boundary thin d-lunes get one points and as whole h_g gets $3k' - 2k$ normal-points, which will imply that the number of normal-points in the boundary thin d-lunes are equal to their respective view difference. Let t be an arbitrary point of $\mathcal{T}$. If t is trivalent, then it corresponds to a d-polygon, which is the intersection of three thin d-lunes of $\mathcal{S}$ and which we call *3-critical*, and put one normal-point in it. When t belongs to only one semi-circle c, put one normal-point in the d-polygon which belongs to the d-lune corresponding to c and also belongs to the boundary thin d-lunes of other two sets of directions. Since each semi-circle in h_g intersects exactly one point of $\mathcal{T}$, the number of points in each non-boundary thin d-lune is exactly one, which is equal to its view difference. Moreover, the total number of normal-points in h_g is thrice the number of non-boundary thin d-lunes in each group minus the twice the number of trivalent vertices, which is $3k' - 2k$.

On the other hand, assume that we have a valid assignment. So h_g has $3k' - 2k$ normal-points. From our construction, each d-polygon can have at most one normal-point since it is one way or the other belongs to a non-boundary thin d-lune which has view difference one. Now for d-polygon that has a normal-point, pick the corresponding intersection point or an arbitrary point from the corresponding semi-circle of $\mathcal{C}$. Observe that only the d-polygons that are 3-critical or belong to boundary thin d-lunes can have a normal-point, since any other d-polygon is a subset of a non-thin d-lune of another set of directions that has view difference zero and since from our construction the sum of all view differences of that set of directions is $3k' - 2k$. It implies that $\mathcal{T}$ has no bi-valent point. Moreover, since for each set of directions the boundary thin d-lunes have a total view difference of $2(k' - k)$, the remaining $3k' - 2k - 2(k' - k) = k'$ normal-points must be in k' 3-critical d-polygons. But a 3-critical d-polygon can have exactly one normal-point. So the number of tri-valent point in $\mathcal{T}$ is k'.

The problem is in NP Given $\mathcal{S}$ and a valid assignment for $\mathcal{S}$, for each direction of $\mathcal{S}$, we check whether the number of normal-points in its visible hemisphere is equal to the corresponding integer or not. This takes no more than $O(K + N)$ time.

References

1. M. S. Bazaraa, J. J. Jarvis, and H. D. Sherali. *Linear Programming and Network Flows*. John Wiley, 2005.
2. S. Bereg. 3D realization of two triangulations of a convex polygon. In *20th Eur. Work. Comp. Geom.*, pages 49–52, Seville, Spain, March 2004.
3. T. Biedl, G. Kant, and M. Kaufmann. On triangulating planar graphs under the four-connectivity constraint. *Algorithmica*, 19(4):427–446, 1997.
4. A. Bottino, L. Jaulin, and A. Laurentini. Reconstructing 3D objects from silhouettes with unknown viewpoints: The case of planar orthographic views. In *8th Iberoamerican Congress on Patt. Recog.*, pages 153–162, Havana, Cuba, November 2003.
5. A. Bottino and A. Laurentini. Introducing a new problem: Shape-from-silhouette when the relative positions of the viewpoints is unknown. *IEEE PAMI*, 25(11):1484–1493, 2003.
6. E. D. Demaine and J. Erickson. Open problems on polytope reconstruction. Manuscript.
7. H. Edelsbrunner. *Algorithms in Combinatorial Geometry*. Springer-Verlag, 1986.
8. R. Gardner. *Geometric Tomography*. Cambridge University Press, 1995.
9. M. Hasan. *Reconstruction and visualization of polyhedra using projections*. PhD thesis, School of Computer Science, University of Waterloo, Canada, 2005.
10. C. H. Hoffman. *Geometric and Solid Modelling*. Morgan Kaufmann, 1989.
11. H. S. Kasan and K. D. Kumar. *Introductory Operations Research: Theory and Applications*. Springer, 2004.
12. A. Laurentini. How many 2D silhouettes does it take to reconstruct a 3D object? *Comp. Vis. Image Unders.*, 67(1), 1997.
13. H. Lipson and M. Shpitalni. Optimization-based reconstruction of a 3D object from a single freehand line drawing. *Computer Aided Design*, 28(8):651–663, 1996.
14. G. Markowsky and M. Wesley. Fleshing out wire frames. *IBM J. Res. Dev.*, 24:582–597, 1980.

15. G. Markowsky and M. A. Wesley. Fleshing out projections. *IBM J. Res. Dev.*, 25(6):934–954, 1981.
16. B. Marlin and G. Toussaint. Constructing convex 3-polytopes from two triangulations of a polygon. In *14th Can. Conf. Comp. Geom.*, pages 36–39, Lethbridge, Alberta, August 2002.
17. W. Matusik, C. Buehler, R. Raskar, S. J. Gortler, and L. McMillan. Image-based visual hulls. In *SIGGRAPH 2000*, pages 369–374, New Orleans, Louisiana, July 2000.
18. I. Nagendra and U. Gujar. 3-D objects from 2-D orthographic views– a survey. *Computer and Graphics*, 12(1):111–114, 1988.
19. M. Penna. A shape from shading analysis for a single perspective image of a polyhedron. *IEEE PAMI*, 11(6):545–554, 1989.
20. K. Sugihara. A necessary and sufficient condition for a picture to represent a polyhedral scene. *IEEE Trans. Patt. Anal. Mach. Intell.*, 6(5):578–586, 1984.
21. K. Sugihara. *Machine Interpretation of Line Drawing*. MIT Press, 1986.
22. R. Thomas. An update on four-color theorem. *Notices of American Mathematical Society*, 45(7):848–859, 1998.
23. P. A. C. Varley. *Automatic creation of boundary-representation models from single line drawings*. PhD thesis, Department of Computer Science, University of Wales College of Cardiff, 2003.
24. W. Wang and G. G. Grinstein. Survey of 3d solid reconstruction from 2d projection line drawings. *Computer Graphics Forum*, 12(2):137–158, 1993.
25. Q.-W. Yan, C. L. P. Chen, and Z. Tang. Efficient algorithm for the reconstruction of 3d objects from orthograpgics projections. *Computer Aided Design*, 26(9):699–717, 1994.

Appendix

Proof of Lemma 3 The following pseudocode will assign the elements of M.

```
for i = 1 ... m, let t_i = R_i
for j = 1 ... n, let u_j = C_j
let j = 1
for i = 1 ... m
    while t_i > 0
        M_{i,j} = min{t_i, u_j}
        t_i = t_i - M_{i,j}
        u_j = u_j - M_{i,j}
        if u_j = 0, then j = j + 1
```

Consider the justification now. $\sum_{i=1}^{m} R_i \leq \sum_{j=1}^{n} C_j$ implies that for any i, $R_i \leq \sum_{j=1}^{n} C_j$. Since R_i intersects all columns, its elements can be assigned a maximum value of $\sum_{j=1}^{n} C_j$ in total. So from the algorithm, the elements of the first row of M are assigned a total value of R_1. After assigning the first row the elements of second row can be assigned a total of $\sum_{j=1}^{n} C_j - R_1$, which is at least R_2. So the elements of the second row are assigned a total value of R_2. In this way for all i, the elements of i-th row get a total value of R_i. Therefore the sum of all elements of M is $\sum_{i=1}^{m} R_i$.

For the columns of M, for all j we keep track of the total value assigned to the j-th column by checking u_j to be zero. Therefore the j-th column is assigned a total value of no more than the initial value of u_j, which is C_j.

Finally, we increment j only if $u_j = 0$. Since $\sum_{i=1}^{m} R_i \leq \sum_{j=1}^{n} C_j$, u_j becomes zero at most n times before the assignment is complete, so during the assignment the value of j does not exceed n.

If the outputs are only the elements that have been assigned and all other elements are zero, then the time complexity is clearly $O(m+n)$. $\square$

Proof of Lemma 4 [Sketch only] **Two planes have a pair of common directions.** Assume the opposite d-i pairs $\langle \overline{d}_0, \overline{n}_0 \rangle$ and $\langle \overline{d}_{\overline{K}}, \overline{n}_{\overline{K}} \rangle$ of $\overline{\mathcal{S}}$ are also in $\widetilde{\mathcal{S}}$. So the great circle of $\overline{h}_0$ and $\overline{h}_{\overline{K}}$ passes through the poles of $\overline{\mathcal{S}}$ and $\widetilde{\mathcal{S}}$. This great circle also divides the d-polygons into two $\overline{K} \times \widetilde{K}$ "matrices"—one is $\overline{h}_0$ and the other one is $\overline{h}_{\overline{K}}$. (See Figure 3(d)(ii)).

After possible renaming assume that $\overline{D} = \min\{\overline{D}, \widetilde{D}\}$. Assign $\widetilde{D}$ normal-points to the d-polygons of $\overline{h}_0$ by using Lemma 3 with the rows of M having sum $\overline{\delta}_0 + \frac{1}{2}(\widetilde{D} - \overline{D})$, $\overline{\delta}_1, \ldots, \overline{\delta}_{\overline{K}-1}$ and the columns having sum $\widetilde{\delta}_0, \ldots, \widetilde{\delta}_{\widetilde{K}-1}$. Similarly, assign a total value of $\widetilde{D}$ to the d-polygons of $\overline{h}_{\overline{K}}$ with the rows of M having sum $\overline{\delta}_{\overline{K}} + \frac{1}{2}(\widetilde{D} - \overline{D})$, $\overline{\delta}_{\overline{K}+1}, \ldots, \overline{\delta}_{2\overline{K}-1}$ and the columns having sum $\widetilde{\delta}_{\widetilde{K}}, \ldots, \widetilde{\delta}_{2\widetilde{K}-1}$. Finally, if $\widetilde{D} < N$, then increase both $\Delta_{0,0}$ and $\Delta_{\overline{K},\widetilde{K}}$ (which are opposite to each other) by $\frac{1}{2}(N - \widetilde{D})$.

Since $\overline{h}_i = \bigcup_{l=i+\overline{K}}^{i-1} \overline{\theta}_l$ and $\overline{h}_i$ includes either $\overline{\theta}_0$ or $\overline{\theta}_{\overline{K}}$, the total number of normal-points in $\overline{h}_i$ is $\sum_{l=i+\overline{K}}^{i-1} \overline{\delta}_l + \frac{1}{2}(\widetilde{D} - \overline{D}) + \frac{1}{2}(N - \widetilde{D})$, which by Lemma 2 is $\overline{n}_i$. Similarly, for $\widetilde{h}_j$ the total number of normal-points is $\widetilde{n}_j$. So the assignment is valid and if

$\max\{\overline{D}, \widetilde{D}\} < N$, then the two opposite d-polygons $\theta_{0,0}$ and $\theta_{\overline{K},\widetilde{K}}$ have at least one normal-point. Finally, using Lemma 3 twice takes $O(\overline{K} + \widetilde{K})$ time.

Two planes have no common direction. Two poles of $\overline{\mathcal{S}}$ (resp. $\widetilde{\mathcal{S}}$) are strictly within two opposite d-lunes, say $\widetilde{\theta}_0$ and $\widetilde{\theta}_{\overline{K}}$ (resp. $\overline{\theta}_0$ and $\overline{\theta}_{\widetilde{K}}$), of $\widetilde{\mathcal{S}}$ (resp. $\overline{\mathcal{S}}$). Observe that both $\overline{\theta}_0$ and $\overline{\theta}_{\overline{K}}$ (both $\widetilde{\theta}_0$ and $\widetilde{\theta}_{\widetilde{K}}$) intersect all d-lunes of $\widetilde{\mathcal{S}}$ ($\overline{\mathcal{S}}$). (See Figure 6(a).)

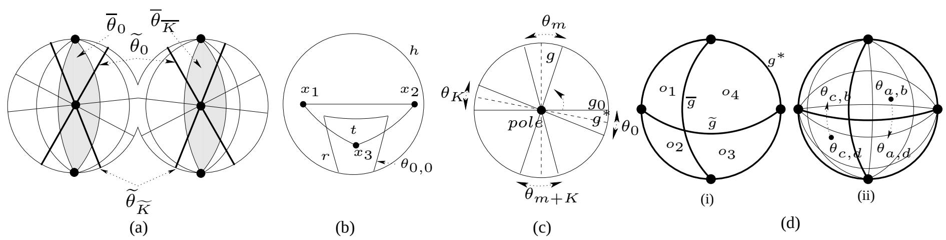

Fig. 6. (a) Lemma **??**, (b) Lemma 5, (c) Lemma 7, (d) Lemma 8: (i) Four octants o_1, o_2, o_3 and o_4, and (ii) Claim 4.

Consider all the d-polygons that are *not* within $\overline{\theta}_0, \widetilde{\theta}_0, \overline{\theta}_{\overline{K}}, \widetilde{\theta}_{\widetilde{K}}$. These form two $(\overline{K}-1) \times (\widetilde{K}-1)$-matrices. We place as many normal points as possible within these d-polygons; one can express exactly how many that is, and find them with Lemma 3. The remaining normal-points must all be within $\overline{\theta}_0, \widetilde{\theta}_0, \overline{\theta}_{\overline{K}}, \widetilde{\theta}_{\widetilde{K}}$, which yields lower bounds (and hence necessary conditions) onto $\overline{\delta}_0, \widetilde{\delta}_0, \overline{\delta}_{\overline{K}}, \widetilde{\delta}_{\widetilde{K}}$. These conditions turn out to be sufficient as well, and we find the assignment using Lemma 3 twice more: we first put as many normal-points as possible in the remaining d-polygons except $\theta_{0,0}$, $\overline{\theta}_{\overline{K},0}$, $\widetilde{\theta}_{0,\widetilde{K}}$, $\theta_{\overline{K},\widetilde{K}}$ (the four d-polygons incident to both poles), and finally put the remaining normal-points into these four polygons. In the process, the two opposite d-polygons $\theta_{0,0}$ and $\theta_{\overline{K},\widetilde{K}}$ obtain all "extra" normal-points, which in particular means that they are both positive if $\mathbf{max}\{\overline{D}, \widetilde{D}\} < N$. $\square$

Proof of Lemma 5 From Lemma 4 we can compute a valid assignment such that two opposite d-polygons $\theta_{0,0}$ and $\theta_{\overline{K},\widetilde{K}}$ have at least one normal-point each. Choose all normal-points arbitrarily according to the valid assignment, except one from $\theta_{0,0}$ and another one $\theta_{\overline{K},\widetilde{K}}$, which are chosen as follows. Let x_1 and x_2 be two of the already chosen $N-2$ normal-points. If x_1 and x_2 are antipodal, then move one of them slightly within its respective d-polygon. Now consider a hemisphere h that contains x_1 and x_2 strictly within it; this exists since x_1 and x_2 are not antipodal. For rest of the proof please refer to Figure 6(b). At least one of $\theta_{0,0}$ and $\theta_{\overline{K},\widetilde{K}}$ intersects h, say $r = \theta_{0,0} \cap h \neq \emptyset$. Choose the second last normal-point $x_3 \neq x_1, x_2$ strictly within r such that x_1, x_2, x_3 do not lie on a great circle. Consider the spherical triangle defined by the three segments $\overline{x_1x_2}$, $\overline{x_2x_3}$ and $\overline{x_3x_1}$, which intersects r, say in area t. Let the opposite of t be t', which is a subset of $\theta_{\overline{K},\widetilde{K}}$. Choose as last normal-point an arbitrary point x_4 strictly within t'.

Note that any hemisphere h'' that contains x_1, x_2, x_3 strictly within also contains t, so t' (and hence x_4) is strictly outside h''. So no hemisphere can contain all points x_1, x_2, x_3 and x_4.

From Lemma 3 we know the list of positive d-polygons. So the time for the selection is $O(N)$. □

Proof of Lemma 7 Let g_0 be the first great-circle, in counter-clockwise direction after g^*, that is the boundary of a visible hemisphere (so either $g_0 = g^*$, or g^* is strictly inside θ_0.) We initialize g to be g_0, and then rotate it counter-clockwise until the lunes satisfy the conditions. To be precise, let m be minimal such that $\sum_{i=1}^{m} \delta_i \geq 2$ and $\sum_{i=K+1}^{m+K} \delta_i \geq 2$, and choose g to be strictly inside θ_m (and θ_{m+K}). See also Figure 6(c).

We claim that the four lunes of g^* and g contain at least 2 normal points each, at least if we distribute normal-points in $\theta_0, \theta_m, \theta_K$ and θ_{m+K} suitably. By $N = D$, there is only one valid assignment: each θ_i contains exactly δ_i normal-points. Also, we know $\min\{\delta_i, \delta_{i+K}\} = 0$ for any i. We may therefore (after renaming, if needed) assume that $\delta_{m+K} = 0$. On the other hand, we must have $\delta_m > 0$ and $\sum_{i=1}^{m-1} \delta_i \leq 1$ by minimality of m. Now we consider the four lunes of G^* and g, which we describe by the d-lunes that they strictly contain:

- The lune containing $\theta_{K+1}, \ldots, \theta_{m+K-1}$: This lune contains at least $\sum_{i=1}^{m+K-1} \delta_i$ normal-points, which is at least 2 by choice of m and $\delta_{m+K} = 0$.
- The lune containing $\theta_1, \ldots, \theta_{m-1}$: This lune contains at least 2 normal-points by choice of m if we include all normal-points from θ_m. However, since some normal-points from θ_m may be needed elsewhere, we will only use $2 - \sum_{i=1}^{m-1} \delta_i$ normal-points from θ_m for it, which gives exactly 2 normal-points for this lune.
- The lune containing $\theta_{m+K+1}, \ldots, \theta_{2K-1}$: Note that no normal-points from θ_0 have been used for the previous lune, so we will include all of them (if any) here. Hence the number of normal-points is $\sum_{i=m+K-1}^{2K-1} \delta_i + \delta_0 = \sum_{i=m+K-1}^{m-1} \delta_i - \sum_{i=1}^{m-1} \delta_i \geq n_m - 1 \geq 3$, since $\sum_{i=1}^{m-1} \delta_i \leq 1$, $\sum_{i=m+K-1}^{m-1} \delta_i = n_m$ by Lemma 2 and $D = N$, and $n_m \geq 4$.
- The lune containing $\theta_{m+1}, \ldots, \theta_{K-1}$: Note that no normal-points from θ_K have been used for the first lune, so we will include all of them (if any) here. Also, this lune gets $\delta_m - (2 - \sum_{i=1}^{m-1} \delta_i)$ normal-points from θ_m. So the number of normal-points is $\sum_{i=m+1}^{K-1} \delta_i + \delta_K + \delta_m + \sum_{i=1}^{m-1} \delta_i - 2 = n_1 - 2 \geq 2$, since $\sum_{i=1}^{K} \delta_i = n_1$ by Lemma 2 and $D = N$, and $n_1 \geq 4$.

Clearly the minimal m satisfying the condition, and the point distribution, can be found in $O(K)$ time. □

Proof of Lemma 8 Choose g^* to be the great circle through the four poles, and $\overline{g}$ and $\widetilde{g}$ by applying Lemma 7 to $\overline{\mathcal{S}}$ and $\widetilde{\mathcal{S}}$, respectively. Now we need to find at least one normal-point in each octant of these three great-circles. We will only show how to select normal-points from one hemisphere of g^*, the other hemisphere case is similar. Any normal-point will avoid the boundary of the d-polygons.

Let o_1, o_2, o_3 and o_4 be the four octants of a hemisphere of g^*, and assume that o_1 and o_2 are in one hemisphere of $\overline{g}$ and o_2 and o_3 are in one hemisphere of $\widetilde{g}$. (See

Figure 6(d)(i)). Each two consecutive octants together (i.e., $o_1 \cup o_2$, $o_2 \cup o_3$, $o_3 \cup o_4$ and $o_4 \cup o_1$) contain at least two normal-points by choice of $\overline{g}$ and $\widetilde{g}$. Unfortunately, this does not imply that each octant contains a normal-point, but if this is not the case, then we can modify the valid assignment to achieve it. We call an octant *empty* if in the given valid assignment all of its d-polygon are empty.

Claim. If there are empty octants, then we can change the valid assignment to another one without empty octants.

Proof. Assume that o_1 is empty. Then o_2 and o_4 contain at least two normal points each. So let $\theta_{a,b}$ and $\theta_{c,d}$ be d-polygons in o_2 and o_4 that contains at least one normal-point each. Recall that $\theta_{a,b} = \overline{\theta}_a \cap \widetilde{\theta}_b$ and $\theta_{c,d} = \overline{\theta}_c \cap \widetilde{\theta}_d$. (See Figure 6(d)(i)). Observe that the intersection of $\overline{\theta}_c$ and $\widetilde{\theta}_b$, which is the d-polygon $\theta_{c,b}$, intersects o_1. Similarly, the d-polygon $\theta_{a,d}$, intersect o_3. Now change the valid assignment by moving one normal-point from each of $\theta_{a,b}$ and $\theta_{c,d}$ to $\theta_{a,d}$ and $\theta_{c,b}$ respectively. See Figure 6(d)(ii). Then the normal-points in any d-lune has not been changed, and we still have a valid assignment. Also, o_1 and o_3 have gained one normal-point each, and o_2 and o_4 have lost one each. Since o_2 and o_4 had at least two normal-points before, now no octant is empty. □

So the valid selection is done as follows: Change the valid assignment, if needed, to avoid empty octants. Then assign coordinates to normal-points arbitrarily as long as each is strictly within its assigned d-polygon. This gives a valid selection by Lemma 6. It is not hard to see that this selection can be found in $O(N + \overline{K} + \widetilde{K})$ time, since during the construction of the valid assignment we can get a list of the (at most N) d-polygons that contain a normal-point, and all other steps require at most scanning this list and doing $O(\overline{K} + \widetilde{K})$ work to change the valid assignment. □