J. G. Gan
Nanyang Technological University,
Singapore

T. C. Woo

K. Tang

University of Michigan,
1205 Beal Avenue,
Ann Arbor, MI 48109-2117

Spherical Maps: Their Construction, Properties, and Approximation

The Gaussian map and its allied visibility map on a unit sphere find wide applications for orientating the workpiece for machining by numerical control machines and for probing by coordinate measurement machines. They also provide useful aids in computerized scene analysis, computation of surface-surface intersection, component design for manufacturing and fabrication procedures. Spherical convex hulls and spherical circles are two geometric constructs used to approximate the Gaussian maps and the visibility maps. The duality and the complementarity of these spherical maps are examined so as to derive efficient algorithms.

1 Introduction

The Gaussian map is a representation of the surface normals. Gauss introduced the concept of mapping the surface normals onto the surface of a unit sphere more than one-and-a-half centuries ago to define the local curvature of a given point (Hilbert, 1983). The Gaussian map finds many applications. In numerical control machining applications, surface normals have long been used to provide tool orientations (Faux, 1979) such that offset surfaces may be generated (Chen, 1987; Farouki, 1986). In component design and fabrication applications (Gan, 1989), the Gaussian map provides such information as the directions which may potentially be used for deep drawing or stamping, and the lower bound on the number of inserts or cores necessary in the molding or casting process. In geometric modelling, the Gaussian map enhances surface-surface intersection algorithms by providing a means of intersection loop detection (Hohmeyer, 1991; Kriezis, 1991; Sederberg, 1988; Sinha, 1985). In computer vision applications, the Gaussian map provides the basis for the generation of aspect graphs (Sripradisvarakul, 1989) used in 3-D shape representation and for object recognition (Besl, 1988).

But it was not until recently that the tool geometry was incorporated in Gaussian maps. Surface normals can be "enhanced" by cutting tool geometries to yield the notion of a visibility map. Fast algorithms for intersection, union, and convex hull on the sphere have been reported (Chen, 1989). Opening a new vista in an allied application of coordinate measurement (Spyridi, 1990), a similar notion of an "accessibility map" is introduced based on measuring probe geometries. Uncovering the basic assumption that these derivatives of a Gaussian map imply a 3-axis machine application, (Tang, 1989) investigates the algorithmic aspect of 4- and 5-axis applications in the context of spherical geometry. In the three aforementioned works, the existence of a visibility (or accessibility) map is assumed.

This paper reports the properties and the computational procedures for constructing the spherical maps. It generalizes the results for a specific surface representation (Kim, 1990). In Section 2, the maps are defined and their construction and properties detailed. Because the Gaussian maps are not always convex, while the visibility maps are, the combination of two operators, (spherical) convex hull SCH and (spherical) dual, yields a pleasant surprise: taking the dual of the SCH of a Gaussian map gives the SCH of a visibility map. Since the visibility map is shown to be convex, the process, as reported in Section 3, gives a fast way to compute the visibility map by first approximating a Gaussian map. The paper goes one step further with another approximation—(spherical) circles. It shows in Section 4 that the in-circle of a visibility map can be computed from the circum-circle of a Gaussian—map. There, two different operators, the (spherical) Voronoi diagram and the (spherical) complement, are invoked.

2 Spherical Maps and Their Construction

In this section, the Gaussian map and the visibility map are defined and their geometric constructions discussed.

2.1 Gaussian Map. Gauss introduced the notion of mapping of surface normals onto the surface of a unit sphere by means of "parallel normals" (Laugwitz, 1965; Hilbert, 1983; Calladine, 1986), where each point on the map is the result of the intersection of the surface normal vector, transferred parallelly such that it emanates from the center of the unit sphere, with the surface of the unit sphere. The region on the surface of the unit sphere so produced has come to be known as the *Gaussian map*. The following definition of the Gaussian map is from do Carmo (1976):

Definition (Gaussian Map)
Let $S \subset \mathbf{R}^3$ be a surface with an orientation N. The map N:

Contributed by the Design Automation Committee for publication in the JOURNAL OF MECHANICAL DESIGN. Manuscript received April 1991. Associate Technical Editor: G. A. Gabriele.

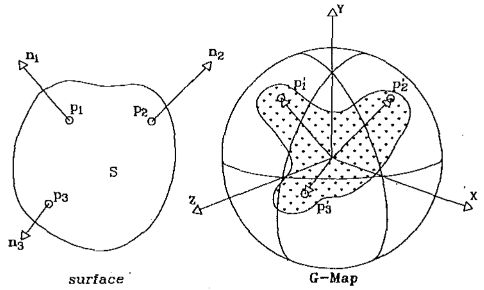

Fig. 2-1 Definition of Gaussian map

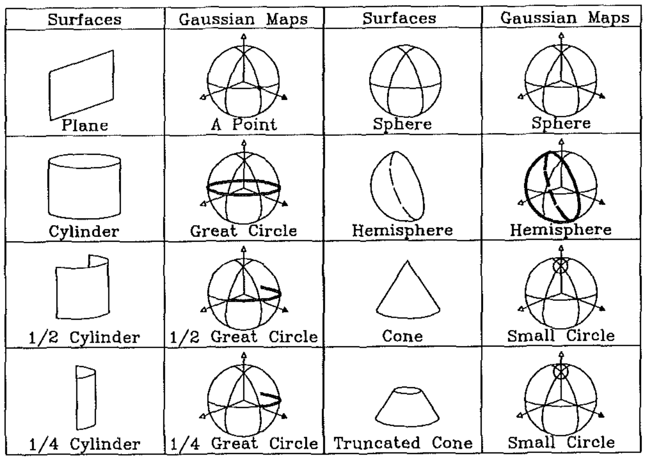

Fig. 2-2 Surfaces and their Gaussian maps

$S \rightarrow \mathbf{R}^3$ takes its values in the unit sphere $\mathbf{S}^2 = \{(x, y, z) \in \mathbf{R}^3 | x^2 + y^2 + z^2 = 1\}$. The map $N: S \rightarrow \mathbf{S}^2$, thus defined, is called the Gaussian map of S.

The geometric interpretation of a Gaussian map, or G-Map for abbreviation, is illustrated with the aid of Fig. 2-1. The surface normals n_1, n_2 and n_3 of three sample points, p_1, p_2 and p_3, on the surface S, are transferred to the unit sphere such that the directions do not change, and that the transferred vectors pass through the center of the sphere. The resulting points, p_1', p_2' and p_3' on the G-map are obtained from the intersection of the transferred normal vectors with the surface of the unit sphere. When this is done for all the points on the surface S, the G-Map for the surface S is produced.

The G-Maps for some of the elementary surfaces are illustrated in Fig. 2-2. For a plane, everywhere on the surface, the normal is the same. Therefore the G-Map for a plane is a single point. For a cylinder, the surface normals everywhere point in directions away from the axis of the cylinder. Therefore, the G-Map for a cylinder is a great circle. It follows that for a 1/2 or a 1/4 cylinder the corresponding G-Map should be 1/2 or 1/4 of a great circle. A sphere contains all the surface normals. Therefore its G-Map is also a sphere. Similarly, the G-Map of a hemisphere is another hemisphere. Due to the fact that the transfer of surface normal vectors is parallel, the position of the G-Map is the same as that of the progenitor hemispherical surface. For a cone, the surface normals everywhere incline at a fixed angle from the axis of the cone. Therefore, its G-Map is a small circle whose center corresponds to the axis of the cone, and whose radius (measured in terms of the angle subtended by the circle at the center of the unit sphere) corresponds to the angle of inclination of the surface normal of the cone from its axis. Since the surface normals of a truncated cone are identical to those of a cone, the G-Map of a truncated cone is identical to that of a cone. Here, it is understood that the location of a G-Map on the unit sphere is a function of the orientation of a given surface in the Euclidean space.

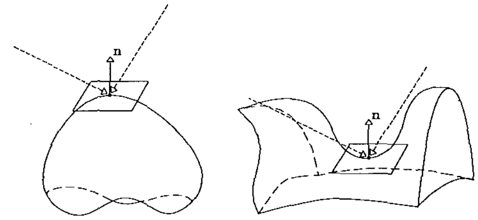

Fig. 2-3 Tangent plane and visibility

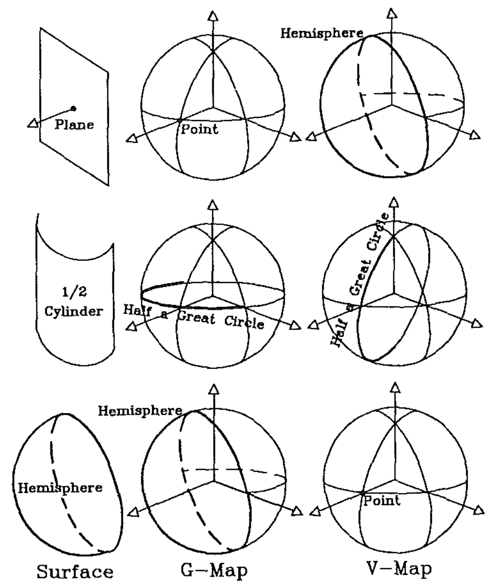

Fig. 2-4 Visibility maps for plane, cylinder, and hemisphere

2.2 Visbility Map. The normal of a point on a surface gives the direction in which the point is visible from infinity if it admits no intersection. But the same point may be visible from other directions as well. As illustrated in Fig. 2-3, a tangent plane yields all other directions from which the point in question is locally visible. Based on this observation the notion of a visibility map is introduced as a representation of a collection of directions, each of which has the potential of viewing the complete surface.

Definition (Visibility Map)
A *visibility map* or *V-Map* of a surface S is the set of points, p_i in $\mathbf{S}^2$, each of which differs from every normal n_j on S by at most 90 degrees. Symbolically, V-Map $= \{p_i | p_i \cdot n_j \geq 0; p_i \in \mathbf{S}^2; n_j \in \mathbf{S}^2\}$.

An illustration of V-Maps with the aid of Fig. 2–4 is in order. A plane has only a single normal, so its G-Map is a single point in $\mathbf{S}^2$. The set of orientations which differs from this point by at most 90 degrees is a hemisphere having this point as the pole, or spherical center. The G-Map of a 1/2 cylinder is half-a-great-circle. Since the V-Map is the set of orientations which differ from this half-a-great-circle by at most 90 degrees, the V-Map is another half-a-great-circle, perpendicularly bisecting it. Since the G-Map of a sphere is a sphere, the G-Map of a hemisphere is also itself. The V-Map of a hemisphere is a point at its pole—it is the only point from which the normals of the hemisphere differ by no more than 90 degrees.

The V-Map is a set of points in $\mathbf{S}^2$ such that each point in the set differs from every point in the G-Map by at most *90*

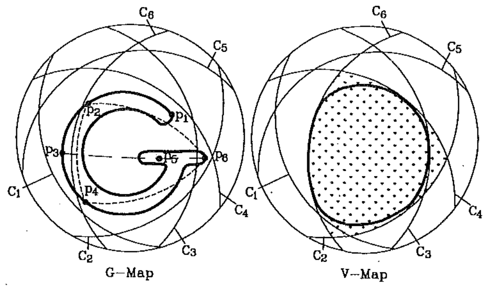

Fig. 2-5 Construction of visibility map

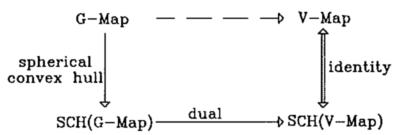

Fig. 3-1 Finding the visibility map

degrees. The geometric construction of a V-Map given a G-Map (for an arbitrary parametric surface) is now shown.

The V-Map of a surface can be constructed by computing the intersection of hemispheres, each having as its pole a point on the G-Map, i.e.,

$$V\text{-}Map(S) = \bigcap_{p_i \in G\text{-}Map(S)} hemi(p_i)$$

Suppose the figure "G" in Fig. 2-5 represents the G-Map of a given surface. Take a point p_1 on the G-map. The set of points in $\mathbf{S}^2$ that deviate from p_1 by at most *90* degrees is a hemisphere $hemi(p_1)$ bounded by the great circle C_1 having the point p_1 as its pole. Similarly, corresponding to the points p_2 to p_6, the sets of points that deviate from each point by at most *90* degrees is each a hemisphere, denoted by $hemi(p_2)$ to $hemi(p_6)$. Each of the hemispheres are bounded by the great circles C_2 to C_6. The intersection of the hemispheres corresponding to the three points p_2, p_4 and p_6 is shown by the dotted figure on the right. The intersection of all the hemispheres $hemi(p_i)$ for all points p_i in "G" yields the oval shape indicated in bold.

The procedure for constructing a V-Map is immediate.

Procedure V-Map(G-Map)
step 1 V-Map: = complete sphere
step 2 For each sample point p_i on the G-Map Do
V-Map: = V-Map $\cap$ hemi(p_i)

While the complexity of *step 2* of *Procedure V-Map* appears to be $O(n^2)$, where n is the number of hemispheres or points approximating the G-Map , it is $O(n \log n)$ by rewriting it in a divide-and-conquer form as *Procedure V-Map**. This procedure and a recursive *Function Partition* take as input a set of points P in $\mathbf{S}^2$ and return as output the intersection of hemispheres whose spherical centers are the input points. These algorithms are given below.

Procedure V-Map (G-Map)*
step 1 V-Map = Partition(G-Map)

Function Partition(P)
step 1 If cardinality(P) > 1 Then
step 2 Divide P into P_1 and P_2 of approximately equal sizes
step 3 $Part_1$: = Partition(P_1)
step 4 $Part_2$: = Partition(P_2)
step 5 Return $Part_1 \cap Part_2$
step 6 Else Return hemi(P)

Though *Procedure V-Map** appears straightforward, the number of sample points it takes is from the entire G-Map. In the following section, it is shown that only the points from the boundary of a G-Map—and, in particular, only the points on the convex hull of a G-Map—are necessary and sufficient.

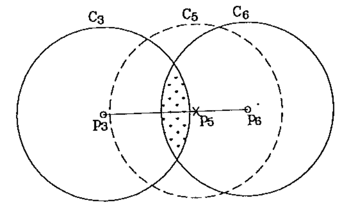

Fig. 3-2 Contributions to V-map

3 Efficient Computation of V-Map

The strategy for computing a V-Map efficiently from a G-Map is illustrated in Fig. 3-1. First, the spherical convex hull SCH of a G-Map is found and then its dual is generated. For each vertex in the SCH(G-Map), there corresponds an edge in the spherical convex hull of the V-Map, SCH(V-Map).

This strategy yields the SCH(V-Map). As the rest of this section is devoted to finding the spherical convex hull and to finding the dual, the convexity of the V-Map is first established, i.e, SCH(V-Map) is identical to V-Map.

Definition (Spherical Convexity)
A set of points in $\mathbf{S}^2$ is (*spherically*) *convex*, if for any two points p_1 and p_2 in the set, the segment $\mathbf{p_1p_2}$ (the shorter of the two great arcs joining p_1 and p_2) lies entirely within the set.

From the construction:

$$V\text{-}Map(S) = \bigcap_{p_i \in G\text{-}Map(S)} hemi(p_i)$$

a V-Map is the intersection of hemispheres, each corresponding to a point in the given G-Map. A hemisphere is, by definition, convex. (Its correspondence to a half plane, by central projection, is alternatively given in the Appendix.) As the intersection of convex sets is always convex, a V-Map is therefore convex. Hence, it is identical to its convex hull. The discussion now moves to the first of two operations: finding the spherical convex hull of a G-Map.

3.1 Spherical Convex Hull. It may be interesting to note that the points in the interior of a G-Map do not contribute to the construction of a V-Map. In Fig. 2-5, the point p_5 is in the interior of a G-Map. Its corresponding hemisphere $hemi(p_5)$ bounded by great circle C_5 is a superset of the eventual V-Map, no part of the boundary of which comes from C_5. By constrast, p_3 and p_6, which are on the boundary of the G-Map in Fig. 2-5, give rise to the V-Map. These observations are not difficult to rationalize. Suppose p_3 and p_6 are the end points of a segment (of a great circle) and p_5 lies somewhere in the segment. As long as the segment is less than or equal to a semi great circle, the intersection of the two hemispheres bounded by C_3 and C_6 will be completely contained in C_5. An analog in the plane is shown in Fig. 3-2.

The observation that only the points on the boundary of a G-Map contributes to a V-Map is formalized by the following.

Lemma 3-1
For the determination of the V-Map, the set of points on the boundary of the G-Map is sufficient.
Proof:
Let p_i be any point in the interior of the G-Map and C any great circle through it. There always exist on C two other boundary points p_b and p_b' on either side of p_i, as shown in Figure 3-3, such that

$$hemi(p_b) \cap hemi(p_b') \subseteq hemi(p)_i$$

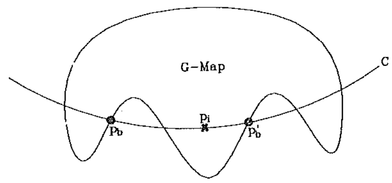

Fig. 3-3 Contributions from boundary points of the Gaussian map

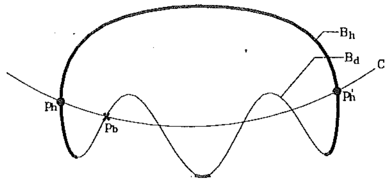

Fig. 3-4 Contributions from convex hull points

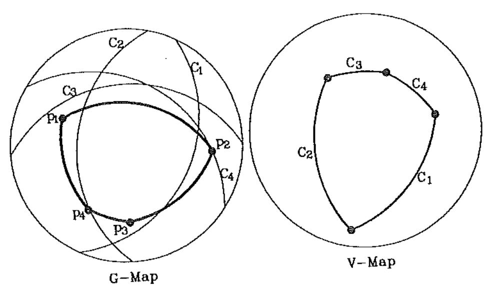

Fig. 3-5 Duality of spherical convex hulls

Since a V-Map results from the intersection of hemispheres and since $hemi(p_b) \cap hemi(p_b') \cap hemi(p_i) = hemi(p_b) \cap hemi(p_b')$, only boundary points p_b and p_b' contribute to the V-Map. **Q.E.D.**

For the construction of a V-Map, Lemma 3-1 reduces the range of data drastically from the domain of an entire G-Map to that of its boundary. A further reduction is made possible by invoking the notion of convexity. It is useful to note that the points on the boundary of a G-Map lying inside its convex hull do not contribute to the construction of a V-Map.

Definition (Spherical Convex Hull)
The *spherical convex hull SCH* of a set of points in $\mathbf{S}^2$ is the boundary of the smallest convex set containing it.

A reasoning similar to the proof for Lemma 3-1 proceeds as follows, with the aid of Fig. 3-4 in which the portion of the boundary that coincides with the convex hull is darkened.

Let the boundary B of a G-Map be divided into two disjoint subsets: the portion that lies on the convex hull B_h and the portion that lies on the convex deficiency B_d. Let p_b be any point in B_d. There always exist a great circle C through p_b such that C intersects B_h at exactly two points: p_h and p_h' on either side of p_b. The point p_b in the center does not contribute to the V-Map, but the other two points on B_h do contribute to the V-Map. The following lemma ensures further reduction of the range of data.

Lemma 3-2
For the determination of the V-Map, the set of points on the boundary of a G-Map that coincides with its convex hull is necessary.

The procedure for determining the spherical convex hull of a G-Map is given in the Appendix.

3.2 Duality. While the notion of the convex hull helps to reduce the number of points n which contribute to the V-Map, the notion of duality enables efficiency in finding the V-Map from a G-Map by providing an $O(n)$ algorithm.

Definition: (Duality of Spherical Convex Hulls)
Two spherical convex hulls, SCH_1 and SCH_2, are *dual* to each other if a vertex V_i of SCH_1 corresponds to an edge E_i in SCH_2 such that E_i is a great arc having V_i as the pole, and the vertices in SCH_1 are in the same order as the corresponding edges in SCH_2; and vice versa.

The two spherical convex hulls given in Fig. 3-5 are dual to each other. Great arcs C_1 to C_4 have their poles at vertices p_1 to p_4 respectively, and they are ordered in the same sense.

Lemma 3-3
SCH(V-Map) is the dual of SCH(G-Map)
Proof:
Take a spherical convex hull whose vertices are p_1, p_2, p_3 and p_4, in clockwise order (see Fig. 3-5). The set of points in $\mathbf{S}^2$ which deviate from vertex p_1 by at most 90 degrees is a hemisphere centered on p_1 (shown as a hemisphere bounded by the great circle C_1). This establishes the duality between a vertex in SCH(G-Map) and an edge in SCH(V-Map). The ordering of the edges in SCH(V-Map) could be different from that of the vertices in SCH(G-Map) if some of the vertices p_i were not used. By Lemmas 3-1 and 3-2, the vertices of SCH(G-Map) are necessary and sufficient. In other words, if the duals of the vertices of SCH(G-Map) are taken in turn, the intersection of the great circles must yield edges for SCH(G-Map) in the order taken.
Now, SCH(V-Map) is the V-Map itself because hemispheres are convex sets and the intersection of convex sets is convex. The SCH(V-Map) is therefore the dual of the SCH(G-Map). **Q.E.D.**

Lemma 3-3 suggests that the V-Map of a set of points in $\mathbf{S}^2$ can be obtained by first finding the spherical convex hull of the G-Map and then dualizing the vertices of its spherical convex hull. It becomes clear then, given ν convex hull vertices of a G-Map, a V-Map can be constructed in $O(\nu)$ time as the corresponding hemispheres can be intersected sequentially.

Procedure V-Map(G-Map)
step 1 Determine SCH(G-Map)
step 2 Determine the dual of SCH(G-Map) to get SCH (V-Map)

The overall complexity of the *Procedure V-Map* is however $O(n \log n)$ because, while *step 2* is $O(\nu)$ where ν is the number of vertices in the convex hull of G-Map *step 1* is $O(n \log n)$ which dominates the time complexity.

4 Spherical Circles as Approximations

Though a V-Map is convex, an in-circle may be used as a further approximation.

Definition (In-Circle)
The largest circle within the boundary of a spherical region in $\mathbf{S}^2$.

The in-circle is appropriate as an approximation to the boundary of a V-Map because it is the largest circle within the boundary of the V-Map and it does not not enclose any point not lying within the V-Map. Finding an in-circle of an n-sided spherical polygon efficiently, however, is difficult as there can be combinatorially many solutions: up to ${}^nC_3$ circles may be tangential to three of these n-edges and not all of them are valid candidates for the in-circle.

A novel approach to the determination of the In-Circle (V-Map) is introduced. It is computed indirectly, in much the same way as the V-Map is computed from the dual of the convex hull of a G-Map, by way of the complement of the circum-circle for a G-Map.

Definition (Circum-Circle)

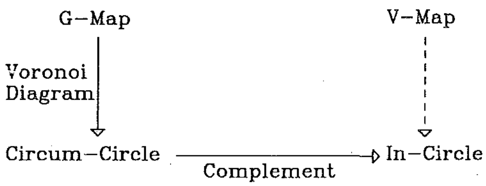

Fig. 4-1 Finding the in-circle of a visibility map

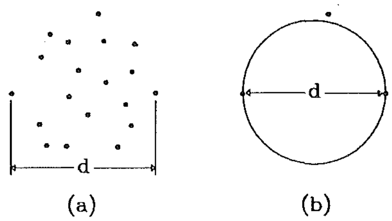

Fig. 4-2 Diameter and point inclusion

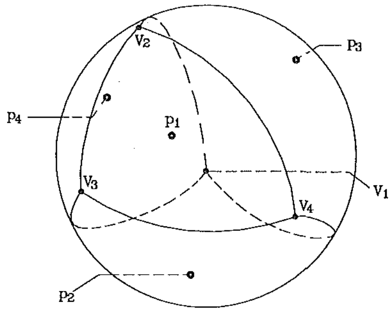

Fig. 4-3 Spherical closest point Voronoi diagram

The *Circum-Circle* of a set of points P in $\mathbf{S}^2$ is the smallest circle in $\mathbf{S}^2$ bounding P.

The strategy for approximating a V-Map is illustrated in Fig. 4-1.

4.1 Finding the Circum-Circle. An immediate method for finding the circum-circle for a set of points is to determine the diameter of the set.

Definition (Diameter of a Set of Points)
The *diameter* of a set of points P in $\mathbf{S}^2$ is the maximum distance between two of them, measured along a great arc.

Figure 4-2 (a) illustrates the diameter of a set of points. Since the SCH of a set of points in $\mathbf{S}^2$ can be computed in time $O(n \log n)$ where n is the number of points in the set, and given the SCH the diameter of the set can be computed in time $O(\nu)$ [Preparata 85] where ν is the number of vertices in SCH, the diameter of a set of points can be computed in time $O(n \log n)$. As noted from Fig. 4-2 (b), however a circle of diameter d may not cover the entire set.

Due to the necessity of additional testing for point inclusion in $O(n)$ time notwithstanding, merely finding the diameter for a circum-circle is not sufficient. A remedy is sought.

The concept of a *Farthest Point Voronoi diagram* is invoked to develop the procedure for determining the three-point circum-circle of a set of points in $\mathbf{S}^2$. A more familiar version of the closest point Voronoi diagram is given first.

Definition (Spherical Closest Point Voronoi Diagram)
The *spherical closest point Voronoi diagram* of a finite set of points P in $\mathbf{S}^2$ is a partitioning of $\mathbf{S}^2$ so that each region of the partition is the locus of points which are closer to one member of P than to any other member.

A closest point Voronoi diagram for a set of four points is

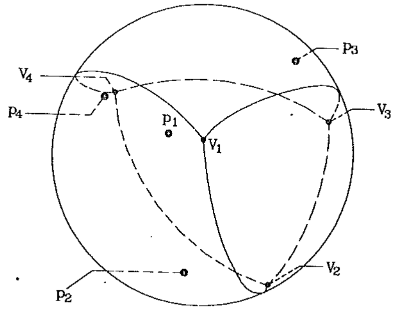

Fig. 4-4 Spherical farthest point Voronoi diagram

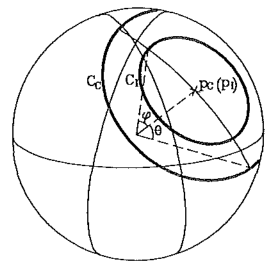

Fig. 4-5 Complement circles

illustrated in Fig. 4-3. Every point within the spherical polygonal region whose vertices are V_2, V_3 and V_4 is closer to the point p_1 than to any other point in the set. Similarly, point p_2 is closest to polygonal region V_1-V_4-V_3, point p_3 is closest to polygonal region V_1-V_2-V_4, and point p_4 is closest to polygonal region V_2-V_1-V_3.

The complement to the spherical closest point Voronoi diagram is defined below:

Definition (Farthest Point Voronoi Diagram)
The (*spherical*) *farthest point Voronoi diagram* of a finite set of points P in $\mathbf{S}^2$ is a partitioning of $\mathbf{S}^2$ so that each region of the partition is the locus of points which are farther from one member of P than from any other member.

A farthest point Voronoi diagram (FPVD) for the same set of four points as that for the closest point Voronoi diagram is illustrated in Fig. 4-4. Every point within the spherical polygonal region whose vertices are V_2, V_3 and V_4 is farthest from the point p_1 than from any other point in the set. Similarly, every point in the polygonal region V_1-V_3-V_4 is farthest from point p_2, every point in the polygonal region V_1-V_4-V_2 is farthest from point p_3, and every point in the polygonal region V_1-V_2-V_3 is farthest from point p_4.

The $O(n \log n)$ algorithm (Shamos, 1975) for determining the closest point Voronoi diagram can also be used to determine the FPVD. This is based on the observation that the FPVD for a set of points P is actually the closest point Voronoi diagram of a set of antipodal points, each of which lies diametrically on the opposite side of the sphere to a point in the set P (Anderson, 1985).

A FPVD vertex V_i has the property that it is equally distant to the set of three points $\mathbf{p}_i$ that contribute to the construction of V_i. Thus there exists a circle centered at a FPVD vertex V_i which passes through $\mathbf{p}_i$.

Lemma 4-1
The circle centered on a vertex of a FPVD, with a radius equal to the distance from this vertex to any of the three points that define it, encloses all the points in the set from which the Voronoi diagram is constructed, i.e., all the points lie on the same side of the circle as the vertex.

Proof:
Each polygon in the FPVD is the locus of a point that is farther from a given point in the set than from any other

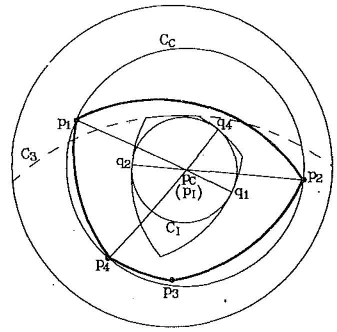

Fig. 4-6 Tangency of C_I and complementarity of C_C

point in the set. Therefore, an FPVD vertex, which is the intersection of three adjacent polygons, is farther from each of the three points in the set than from all the other points in the set. That means, there is no point in the set outside the circle defined by the three points that determine the FPVD vertex. **Q.E.D.**

It is important to note that the number of such three-point circles does not reach the naive bound of ${}^nC_3$. By the definition of the spherical Voronoi diagram on a set of n points in $\mathbf{S}^2$, $\mathbf{S}^2$ is partitioned into n regions (spherical polygons). The dual of a Voronoi diagram is the triangulation of $\mathbf{S}^2$ (Preparata, 1985). Therefore, the triangulated $\mathbf{S}^2$ has n vertices. By the application of Euler's formula (Bollobas, 1979; Swamy, 1981) to triangulation, triangulating a set of n vertices in $\mathbf{S}^2$ partitions $\mathbf{S}^2$ into *2n-4* spherical triangles. *2n-4* triangulated regions dual to *2n-4* vertices in the FPVD. Hence, there are *2n-4* such three-point circles.

The determination of the circum-circle of a set of points P in $\mathbf{S}^2$ can be summarized by the following algorithm:

Procedure Circum-Circle(G-Map)
step 1 *Final diameter d of the set of sample points P in the G-Map*
step 2 *If spherical circle with diameter d covers P then Circum-Circle is found otherwise, go to step 3*
step 3 *Find Spherical Farthest Point Voronoi Diagram of P*
step 4 *Circum-Circle = Smallest (three-point circles)*

step 1 takes $O(n \log n)$ time, *step 2* $O(n)$ time, *step 3* $O(n \log n)$ time, and *step 4* $O(n \log n)$ time. Therefore, the overall complexity is $O(n \log n)$.

4.2 Complementarity

Definition (Complementarity of Spherical Circles)
Two spherical circles, C_C and C_I, are complementary to each other if they have the same spherical centers and the angle subtended by C_C at the center of the unit sphere and that subtended by C_I are *180* degrees complement of each other.

Let C_C be the circum-circle approximating the G-Map and C_I the corresponding in-circle approximating the V-Map. The following results are immediate by the definition of the V-Map.

Lemma 4-2
If the centers of C_C and C_I are p_C and p_I, and the angles subtended by C_C and C_I are 2θ and 2φ, respectively. Then
(i) $p_C = p_I$; and
(ii) $\varphi + \theta = 90$ degrees.

The in-circle C_I approximating the V-Map is computed by taking the complement of the circum-circle C_C for the corresponding G-Map. Having asserted the existence of two complementary circles, it is necessary to show that C_I thus obtained is indeed tangential to the V-Map, with the aid of Fig. 4-6.

Lemma 4-3
Let p_i be a vertex on the G-Map which also lies on C_C. Then, there exists a point q_i such that it is the point of tangency between C_I and the great arc centered at p_i.
Proof:
By Lemma 4-2, since p_i is a point on C_C, there exists a point q_i on C_I such that the great arc p_iq_i subtends an angle of *90* degrees at the center of the unit sphere, and the great arc p_iq_i passes through p_c (and p_I). The point q_i therefore lies on the great arc centered at p_i. The result follows.
Q.E.D.

While Lemma 4-3 shows the tangency of C_I to a V-Map, it is insufficient unless C_I is also shown to be an in-circle to the V-Map, that is, C_I is a conservative approximation to a V-Map.

Lemma 4-4
None of the great arcs, centered at vertices p_j not lying on C_C, intersects C_I.
Proof:
Let p_j be a vertex on the G-Map which does not also lie on C_C. Since p_j lies within C_C, everywhere on the circle C_I differs from p_j by less than *90* degrees. Therefore, the edge on the V-Map which forms part of the great arc centered at p_j does not intersect C_I. **Q.E.D.**

In Fig. 4-6, p_3 does not lie on C_C. Its corresponding great circle C_3, while giving rise to the V-Map, does not intersect the in-circle C_I approximating the V-Map.

The idea that the in-circle to a V-Map can be obtained by taking the complement (via Lemma 4-2) of the circum-circle of a G-Map (by Procedure Circum-Circle) is now complete.

Lemma 4-5
In-Circle (V-Map) is the complement of Circum-Circle (G-Map).
Proof:
By Lemma 4-3, C_I is the circle which is tangent to edges of the V-Map which are parts of the great arcs, centered at the vertices of the G-Map lying on the Circum-Circle (G-Map). By Lemma 4-4, C_I lies fully within the SCH(V-Map). This completes the proof. **Q.E.D.**

Lemma 4-5 above gives the basis that the in-circle of the V-Map can be found by complementing the circum-circle of the G-Map, as an alternative to finding the spherical convex hulls of the G-Map and the V-Map.

Procedure In-Circle (G-Map)
step 1 *Determine Circum-Circle of G-Map*
step 2 *In-Circle (V-Map) is given by the Complement of the Circum-Circle found in step 1*

4 Summary

Procedures for computing a G-Map from a given surface and for computing a V-Map from a G-Map have been presented. In addition to the procedure V-Map*, which takes the set intersection of hemispheres, each of which corresponds to a point in the G-Map, two efficient schemes for computing a V-Map by approximation are given.

Since a V-Map is convex, it is identical to its spherical convex hull SCH(V-Map). It is shown in Lemma 3-3 that SCH(V-Map) can be obtained by taking the dual of the SCH(G-Map): for each point p_i on the convex hull of a G-Map there corresponds an edge C_i in the convex hull of a V-Map, and such C_i is a part of the great circle centered at p_i on the Gaussian sphere.

An in-circle is proposed as a further approximation to a V-Map. Lemma 4-5 eases the difficult computation by complementing the circum-circle of a G-Map: that the two circles have identical centers on the sphere with solid angles 180 degrees complement to each other. A three-point circum-circle for a G-Map is obtained from its farthest point Voronoi diagram.

References

Anderson, D. P., 1985, "Efficient Algorithms for Automatic Viewer Orientation," *Comp. & Grap*, Vol. 9, No. 4, pp. 407–413.

Besl, P. J., 1988, *Surfaces in Range Image Understanding*, Springer-Verlag, NY.

Bollobas, B., 1979, *Graph Theory, An Introductory Course*, Springer-Verlag, NY.

Calladine, C. R., 1986, "Gaussian Curvature and Shell Structure," *The Mathematics of Surfaces*, Gregory JA, ed., Clarendon Press, Oxford.

Chen, Y. J., and Ravani, B., 1987, "Offset Surface Generation and Contouring in Computer Aided Design," ASME Journal of Mechanisms, Transmissions, and Automation in Design, Sept.

Chen, L. L., and Woo, T. C., 1992, "Computational Geometry on the Sphere with Application to Automated Machining," ASME Journal of Mechanisms, Transmissions, and Automation in Design, Vol. 114, No. 2, pp. 288–295.

do Carmo, M. P., 1976, *Differential Geometry of Curves and Surfaces*, Prentice-Hall, NJ.

Farouki, R. T., 1985, "The Approximation of Non-Degenerate Surfaces," *Comp. Aided Geometric Design*, Vol. 2, No. 1, pp. 257–279.

Faux, I. D., and Pratt, M. J., 1979, *Computational Geometry for Design and Manufacture*, Halsted Press, NY.

Gan, J., 1989, "Spherical Algorithms for Setup Orientations of Workpieces with Sculptured Surfaces," Ph.D. Dissertation, IOE Dept, U of Michigan.

Hilbert, D., and Cohn-Vossen, S., 1983, *Geometry and the Imagination*, Translated by P Nementi. Chelsea, NY.

Hohmeyer, M., 1991, "A Surface Intersection Algorithm Based on Loop Detection," *Proc. Symp. on Solid Modeling Foundations and CAD/CAM Applications, J., ACM Siggraph*, Rossignac J. and Turner, eds., pp. 197–208, Jun.

Kim, D. S., 1990, "Cones on Bezier Curves and Surfaces," Ph.D. Dissertation, IOE Dept, U. of Michigan.

Kriezis, G. A., and Patrikalakis, N. M., 1991, "Rational Polynomial Surface Intersections," *Proc. 17th ASME Design Automation Conf., Miami Fl., Advances in Design Automation: CAD*, Vol. II, pp. 43–53, Sept.

Laugwitz, D., 1965, *Differential and Riemannian Geometry*, translated by Steinhardt F., Academic Press, NY.

Preparata, F. P., and Shamos, M. I., 1985, *Computational Geometry—An Introduction*, Springer-Verlag, NY.

Sederberg, T. W., and Myers, R. J., 1988, "Loop Detection in Surface Patch Intersections," *Comp. Aided Geometric Design*, Vol. 5, pp. 161–171.

Sedgewick, R., 1983, *Algorithms*, Addison Wesley, Reading, MA.

Shamos, M. I., and Hoey, D., 1975, "Closest-point Problems," In *16th Annual Symposium on Foundations of Computer Science*, pp. 155–162.

Sinha, P., Klassen, E. E. E., and Wang, K. K., 1985, "Exploiting Topological and Geometric Properties for Selective Subdivision," *Proc. ACM Symp. on Computational Geometry*, Baltimore, MD, pp. 39–45.

Sypridi, A. J., and Requicha, A. A. G., 1990, "Accessibility Analysis for the Automatic Inspection of Parts," *Proc. Int'l. Conf. on Robotics and Automation*, Cincinnati, OH, pp. 1284–1289, May 13–18.

Sripradisvarakul, T., and Jain, R., 1989, "Generating Aspect Graph for Curved Objects," *Proc. Workshop on Interpretation of 3D Scenes*, Austin, TX, pp. 109–115, Nov. 27–29.

Swamy, M. N. S., and Thulasiraman, K., 1981, "Graphs, Networks, and Algorithms, John Wiley, NY.

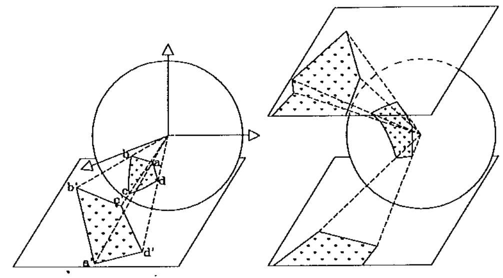

Fig. A-1 Projection of spherical convex hull

Tang, K., Woo, T. C., and Gan, J. G., 1992, "Maximum Intersection of Spherical Polygons and Workpiece Orientation for 4- and 5-axis Machining," ASME Journal of Mechanical Design, Vol. 114, No. 3, pp. 447–485.

APPENDIX

Determining the Spherical Convex Hull

Convex hull algorithms for a set of points lying on a plane are well researched (Preparata, 1985; Sedgewick, 1983). If the set of points P in S^2 are hemispherical such that they can be contained within a hemisphere, then it is possible to map P into a set of points P' in the plane by central projection. (If P is not hemispherical, then central projection onto two parallel planes suffice. See Chen (1989) for details.) Since the central projection is a geodesic projection, the great arcs are projected as straight lines which form the edges of the convex hull in the plane corresponding to the great arcs in the spherical convex hull. Figure A-1 illustrates the projection of the spherical convex hull *abcd* onto a plane as a planar convex hull $a'b'c'd'$. *Procedure SCH* below outline the steps involved in finding the spherical convex hull via central projection.

Procedure SCH(G-Map)
step 1: Centrally Project the points in the G-Map onto a plane
step 2: Determine the convex hull in the projected plane
step 3: Reverse the projection to get the spherical convex hull

Step 1 can be performed in time $O(n)$ where n is the number of given points. *Step 2* can be performed by using Graham Scan which is $O(n \log n)$ (Preparata, 1985). *Step 4* is the reverse projection and it can be done in time $O(n)$. Therefore, the overall complexity of *Procedure SCH* is $O(n \log n)$.