

Efficient Algorithms for Approximating the Centroids of Monotone Directions in a Polyhedron

Jong-Sung Ha

Department of Game and Contents

Woosuk University, 490 Hujongri, Samrae-Up, Wanju-Kun, Chonbuk 565-701 Republic of Korea

Kwan-Hee Yoo

Department of Digital Informatics and Convergence

Chungbuk National University, 410 Seongbongro, Heungukgu, Cheongju, Chungbuk 361-763 Republic of Korea

ABSTRACT

We present efficient algorithms for computing centroid directions for each of the three types of monotonicity in a polyhedron: strong, weak, and directional monotonicity, which can be used for optimizing directions in many 3D manufacturing processes. Strongly- and directionally-monotone directions are the poles of great circles separating a set of spherical polygons on the unit sphere, the centroids of which are shown to be obtained by applying the previous result for determining the maximum intersection of the set of their dual spherical polygons. Especially in this paper, we focus on developing an efficient method for approximating the weakly-monotone centroid, which is the pole of one of the great circles intersecting a set of spherical polygons on the unit sphere. The original problem is approximately reduced into computing the intersection of great bands for avoiding complicated computational complexity of non-convex objects on the unit sphere, which can be realized with practical linear-time operations.

Key words: Polyhedron Monotonicity, Centroid Direction, Gaussian Map, Visibility Map, Spherical Algorithm

1. INTRODUCTION

The notion of geometric *monotonicity* that has arisen in the theoretical area of computational geometry [3] is closely related with 3D manufacturability problems, while the problems of machinability and demouldability in application field areas [4], [5] have been formulated being based on the notion of surface *visibility*.

In this paper, we consider efficient and robust implementation for computing the monotone directions of a polyhedron, based on the characterization [1] of polyhedron monotonicities categorized as [3].

- A polyhedron is *strongly-monotone* in a direction d , if the intersection of the polyhedron with every plane parallel to d is a monotone polygon in d .
- A polyhedron is *weakly-monotone* in a direction d , if the intersection of the polyhedron with every plane orthogonal to d is a simple polygon.

- A polyhedron is *directionally-monotone* (directionally-convex) in a direction d , if every line parallel to d intersects the boundary of the polyhedron at most two times

Similarly to the formulations of surface machinability [5], the polyhedron monotonicities have been characterized as geometric problems to find great circles *separating or intersecting a set of spherical polygons* that are derived from sub-surfaces of the polyhedron and its convex hull [1]. Consequently, all directions for the three monotonicity can be constructed in $O(nk \log k + n \log n)$ time, where n and k are the numbers of all faces and all sub-surfaces, respectively.

This paper suggests efficient algorithms for finding a centroid of all monotone directions in a polyhedron, which will be called the *monotone centroid* in short. The centroid in a set of directions is defined so as to *maximize the minimum distance between the centroid and all directions* in the set. This centroid is very useful for optimizing directions in many manufacturing processes; minimize the cusp height in machining, minimize the frictional forces in assembling and moulding, and so on. Furthermore, our efficient methods or approximating the centroid can also replace the feasibility test for the next

* Corresponding author, Email: khyoo@chungbuk.ac.kr
 Manuscript received Feb. 24, 2016; revised Apr. 07, 2016;
 accepted Apr. 18, 2016

designing processes or before applying more complicated algorithms for constructing all solutions.

The strongly- and directionally-monotone centroids will be obtained by simply applying a discrete algorithm [2] that finds maximum intersection of dual spherical polygons for approximating centroids among the poles of great circles intersecting a set of spherical polygons. The weakly-monotone centroid will be effectively approximated by intersecting other spherical objects called *great bands* instead of non-convex spherical regions. Our approximation can be efficiently realized with linear time operations such as the smallest enclosing sphere [6] and linear programming [7]-[9].

2. NOTATION AND DEFINITIONS

The space on the boundary of the unit sphere centered at origin in three dimension is described as $S^2 = \{p \mid \|p\|=1\}$. A point p on S^2 is a unit vector in three dimensional Euclidean space E^3 .

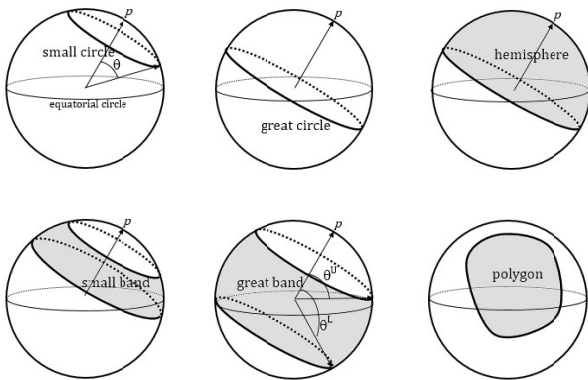


Fig. 1. Geometric Primitives on S^2

As illustrated in Fig. 1, we define *geometric primitives* on S^2 , which are described by the set of points with the operation of inner product.

A circle on S^2 is determined by the intersection of the unit sphere with a plane. A part of a circle is called an *arc* (edge). If the plane contains the origin, the intersection is called a *great circle*; otherwise, it is called a *small circle*. The circle is denoted by $Cr(p, \theta) = \{x \mid p \cdot x = \cos\theta, \text{ and } p, x \in S^2\}$. We call p the pole of the circle, and θ the size of the circle. A great circle will be shortly written by $GC(p)$. It is obvious that $Cr(p, \theta) = Cr(-p, \pi - \theta)$ and $GC(p) = GC(-p)$.

We define two regions bounded by circles on S^2 : *hemisphere* denoted by $HS(p) = \{x \mid p \cdot x \geq 0, \text{ and } p, x \in S^2\}$, and *great band* denoted by $GB(p, \theta^L, \theta^U) = \{x \mid \cos\theta^L \leq p \cdot x \leq \cos\theta^U, \text{ and } p, x \in S^2\}$, where $\theta^L > \frac{\pi}{2}$ and $\theta^U < \frac{\pi}{2}$. A great circle can be included in a great band, while a band excluding any great circle is called the *small band*.

A (*spherical*) *polygon* is another region on S^2 , which is bounded by great arcs; a convex polygon is also the intersection of hemispheres.

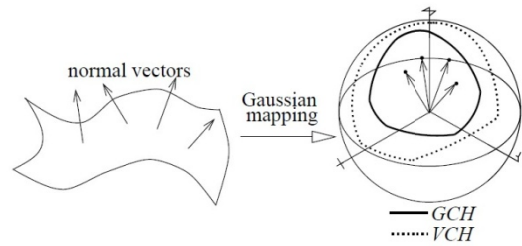


Fig. 2. Gaussian Polygon and Its Dual Visibility Polygon

As illustrated in Fig. 2, let $U = \{u_1, \dots, u_n\}$ be the set of outward unit normal vectors to a surface S , which is called the *Gaussian map* of S . The set of inward unit normal to S will be represented with $-U$. The *visibility map* of S , which is the set of directions visible to S , can be computed by $\bigcap_{i=1}^n HS(u_i)$.

The spherical convex hulls of the Gaussian map and visibility map of S will be, respectively, denoted by $GCH(U)$ and $VCH(U)$, or by $GCH(S)$ and $VCH(S)$ for convenience. In short, they will be, respectively, called (*positive*) *Gaussian polygon* and (*positive*) *visibility polygon* in this paper. $GCH(-U)$ and $VCH(-U)$ or $GCH(-S)$ and $VCH(-S)$ are, respectively, *negative* Gaussian and visibility polygons. The two spherical polygons of Gaussian and visibility are dual to each other; a vertex of $GCH(U)$ for all corresponds to an edge (arc) in $VCH(U)$, and vice versa [10].

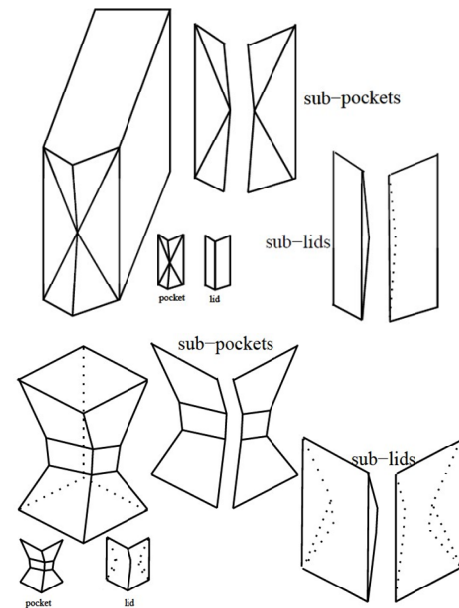


Fig. 3. Sub-pockets and Sub-lids of Polyhedrons

As illustrated in Fig. 3, we define the sub-surfaces of a polyhedron with pockets $\{P_i\}$, lids $\{L_i\}$, sub-pockets $\{SP_i\}$, and sub-lids $\{SL_i\}$, which are used for characterizing the monotone directions of the polyhedron.

3. APPROXIMATING STRONGLY- AND DIRECTIONALLY-MONOTONE CENTROIDS IN A POLYHEDRON

The strongly- and directionally-monotone directions can be established by *finding great circles separating a set of Gaussian polygons* of pockets $\{P_i\}$ and lids $\{L_i\}$ (sub-pockets $\{SP_i\}$) of a polyhedron as follows.

Lemma 1. *A polyhedron P is strongly-monotone in a direction d , if and only if $GC(d)$ separates $GCH(P_i \cup -L_i)$ for all $i = 1 \dots n$ [1].*

Lemma 2. *A polyhedron P is directionally-monotone in a direction d , if and only if $GC(d)$ separates $GCH(SP_i)$ for all $i = 1, \dots, n$ [1].*

All poles of great circles separating a set of Gaussian polygons $\{GCH_1, \dots, GCH_n\}$ can be found by partitioning the unit sphere S^2 with the set of their dual positive or negative visibility polygons $\{VCH_1, \dots, VCH_n, -VCH_1, \dots, -VCH_n\}$. If a partitioned face is contained in VCH_i or $-VCH_i$ for each $i = 1, \dots, n$, the face represents the set of poles of great circles separating $\{GCH_1, \dots, GCH_n\}$ [4].

In order to compute great circles separating Gaussian polygons derived from sub-surfaces of a polyhedron, we are going to use an efficient algorithm maximizing the intersection number of visibility polygons on S^2 .

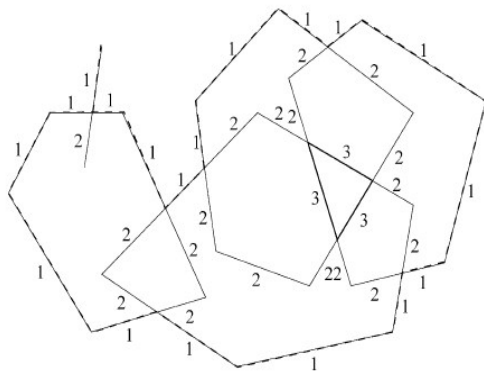


Fig. 4. Counting Ownership Numbers of Edges Partitioned by Visibility Polygons

For maximizing the number of intersected visibility polygons, motivated from the machinability problem for setting models and cutting tools, we have presented a novel method [2] called edge-based partition of S^2 with great circles, which is robust to avoid topological errors and handle geometric degeneracies as well as practical to be implemented in an efficient way. As shown in Fig. 4, the ownership number of each partitioned edge is counted by incrementally partitioning the visibility polygons. The final ownership numbers of all partitioned edges are scanned for finding the maximum ownership number, and then the edges with the maximum number are gathered. Using the convexity of the intersected solution regions bounded by the gathered edges, the centroids

of the solution regions can be obtained in linear time with a discrete method without constructing the solution boundary.

We can observe the following property in the problem of finding the ownership number of edges partitioned by the set of visibility polygons.

Property 1. *A set $\{GCH_1, \dots, GCH_n\}$ is separated by a great circle, if and only if the maximum ownership number of edges partitioned with $\{VCH_1, \dots, VCH_n, -VCH_1, \dots, -VCH_n\}$ is just n . (The maximum ownership number is at most n , since $VCH_i \cap -VCH_i = \emptyset$ for any $i = 1, \dots, n$).*

Our problem of separating Gaussian polygons is a special case of the maximization problem finding a region intersected by all of their dual (positive or negative) visibility polygons; in other words, we check the maximum ownership number is the same as the total number of Gaussian polygons. It becomes possible to construct indirect steps for computing the strongly and directionally-monotone centroids in a polyhedron; given a set of Gaussian polygons of sub-surfaces (in Lemma 1 and 2) of a polyhedron, we first compute the region that is maximally intersected by their dual visibility polygons, and then determine the centroid of the completed region only if the ownership number of the region is the same as to the total number of Gaussian polygon.

Consequently, the strongly-monotone centroids of a polyhedron can be obtained by simply applying the previous results of [1], [2] as the following algorithm. The directionally-monotone centroids can be similarly computed.

procedure Strongly Monotone Centroid

input: the pockets $\{P_i\}$ and the lids $\{L_i\}$ of P

output: the strongly-monotone centroids of P

- 1 For all $i = 1, \dots, n$, compute $\{VCH_i(P_i \cup -L_i)\}$.
- 2 Maximum Intersection ($\{VCH_i\} \cup \{-VCH_i\}$) [2].
- 3 If the maximum ownership number $\neq n$ then, terminate with the infeasibility.
- 4 Return with the centroid solutions.

endProcedure Strongly Monotone Centroid

4. APPROXIMATING WEAKLY-MONOTONE CENTROIDS IN A POLYHEDRON

Weakly-monotone directions can be established by *finding great circles intersecting with visibility polygons* of sub-pockets $\{SP_i\}$ and sub-lids $\{SL_i\}$ of a polyhedron as follows.

Lemma 3. *A polyhedron P is weakly-monotone in a direction d , if and only if $GC(d)$ intersects $VCH(SP_i \cup -SLF_{ij})$ for all $i = 1, \dots, n_1, j = 1, \dots, n_2$, where SLF_{ij} is each face $\in SL_i$ [1].*

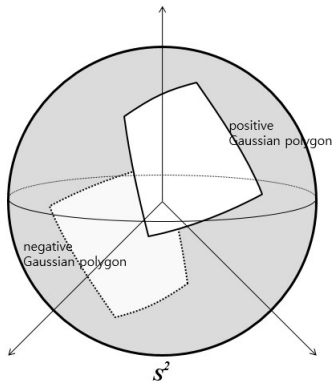


Fig. 5. Complement of Two Dual Gaussian Polygons for a Visibility Polygon

We will characterize a region representing the set of poles of great circles intersecting a visibility polygon by using its dual positive and negative Gaussian polygons. Property 2 leads to that the set of poles of great circles for this intersection is the complement of its dual positive and negative Gaussian polygons, as illustrated in Fig. 5.

Property 2. *Intersecting a visibility polygon VCH by great circles is complementary to separating (not intersecting) it by great circles, and the set of poles of great circles separating VCH is the dual positive and negative Gaussian polygons GCHU–GCH.*

In Section 3 approximating the strongly- and directionally-monotone centroids of a polyhedron, the great circles separating Gaussian polygons of subsurface of a polyhedron could be efficiently computed with the intersection of their dual visibility polygons by using the convexity. Unfortunately, however, the complement of two positive and negative Gaussian polygons, which can be described as $\sim(GCHU-GCH)$, is a non-convex region. In any space, it is very difficult to compute the intersection of non-convex objects, compared to convex objects.

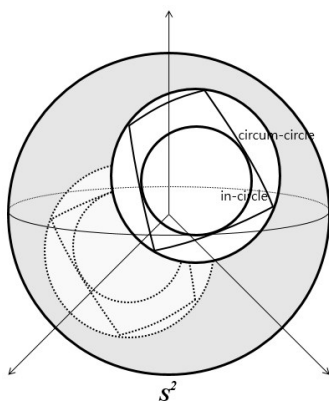


Fig. 6. Great Bands Bounded by Each Pair of In-circles and Circum-circles of Gaussian Polygons

We develop an approximation method to use primitive objects bounding an original object in order to avoid complicated operations for non-convex objects, as illustrated in Fig. 6. Two small circles can bound a polygon on S^2 ; in-circle

and circum-circle that are, respectively, the largest circle within the polygon and the smallest circle enclosing the polygon.

When we replace a polygon on S^2 with its bounding circles, the non-convex region that is the complement of a pair of two positive and negative Gaussian polygons is approximated by a primitive object called the great band. Great bands are well-known geometric primitives on S^2 , which can be often found in 4- and 5-axis machinability problems. The great band approximated by the in-circles can be used for the feasibility test since it is the necessary condition for the original solutions, while another great band approximated by the circum-circles is a sufficient condition for quickly finding the centroid among solutions.

Let $Cr(p_{ij}^I, \theta_{ij}^I)$ and $Cr(p_{ij}^C, \theta_{ij}^C)$ be, respectively, the in-circle and the circum-circle of $GCH(SP_i \cup -SLF_{ij})$ for all i, j in Lemma 3, then we can summarize our approximation method as follows.

Property 3. $\bigcap_{i,j=1}^{n_1, n_2} GB(p_{ij}^C, \theta_{ij}^C, \pi - \theta_{ij}^C) \subset$ the set of weakly-monotone directions $\subset \bigcap_{i,j=1}^{n_1, n_2} GB(p_{ij}^I, \theta_{ij}^I, \pi - \theta_{ij}^I)$.

Let's consider the approximately reduced problem intersecting great bands under the extension of the geometric space from the spherical space S^2 space into three-dimensional Euclidean space E^3 . Since the two planes passing small circles bounding a great band $GB(p, \theta^U, \theta^L)$ can be, respectively, expressed by $\{x \mid (x - \cos\theta^U p) \cdot p = 0, x \in E^3\}$ and $\{x \mid (x + \cos\theta^L p) \cdot p = 0, x \in E^3\}$, the space volume bounded by the two planes in E^3 can be represented with Definition 1. Then, a great band is the intersection between this space volume and S^2 ; $GB(p, \theta^U, \theta^L) = SV(p, \theta^U, \theta^L) \cap S^2$.

Definition 1. $SV(p, \theta^U, \theta^L) = \{x \mid (x - \cos\theta^U p) \cdot p < 0 \text{ and } (x + \cos\theta^L p) \cdot p > 0, x \in E^3\}$

In order to check the intersection of a set of great bands, we observe an important property as follows.

Property 4. *Given a set of great bands $\{GB(p_1, \theta_1^U, \theta_1^L), \dots, GB(p_n, \theta_n^U, \theta_n^L)\}$, $\bigcap_{i=1}^n GB(p_i, \theta_i^U, \theta_i^L) = \emptyset$, if and only if $\bigcap_{i=1}^n SV(p_i, \theta_i^U, \theta_i^L)$ is completely enclosed by S^2 .*

In order to check the feasibility without constructing all boundary of $\bigcap_{i=1}^n SV(p_i, \theta_i^U, \theta_i^L)$, we find its extreme point with a maximizing problem such as Formulation 1. After determining the extreme point x^* of $\bigcap_{i=1}^n SV(p_i, \theta_i^U, \theta_i^L)$, a simple test is performed whether or not $\|x^*\| < 1$. Even though Formulation 1 has the form of a quadratic objective function different from general linear programming (LP) formulations, it can be solved in linear time by an efficient LP technique since its objective function is convex [8].

Formulation 1. Maximize $\|x^2\|$ subject to $SV(p_i, \theta_i^U, \theta_i^L)$, for all $i = 1, \dots, n$.

Clearly, the centroid of $\bigcap_{i=1}^n GB(p_i, \theta_i^U, \theta_i^L)$, which maximizes the minimum distance between the centroid and all

points in the solution, is the center of circle inscribing the region of $\cap_{i=1}^n GB(p_i, \theta_i^U, \theta_i^L)$, the boundary of which is composed of the parts of small circles $\{Cr(p_1, \theta_1^U), \dots, Cr(p_n, \theta_n^U)\}$ and $\{Cr(p_1, \theta_1^L), \dots, Cr(p_n, \theta_n^L)\}$. In other words, a weakly-monotone centroid can be obtained by computing the center of an inscribing circle similarly to a strongly-monotone centroid in Section 3; the difference is that the one is bounded by small circles and the other by great circles. The methods to compute the centers of inscribing circles within small and great circle boundaries are quite different from each other.

The center of circle inscribing a convex region bounded by great circles could be obtained in linear time with a discrete method of edge-based partition [2]. In the case of the region bounded by small circles, we prove that the solution of Formulation 1 is just the centroid of $\cap_{i=1}^n GB(p_i, \theta_i^U, \theta_i^L)$ as.

Lemma 4. *The unit vector of a solution x^* maximizing $\|x^2\|$ subject to $SV(p_i, \theta_i^U, \theta_i^L)$ for all $i = 1, \dots, n$ is the centroid of $\cap_{i=1}^n GB(p_i, \theta_i^U, \theta_i^L)$, if $\|x^2\| > 1$.*

Proof. As illustrated in Fig. 7, consider a cross-section of a plane passing through three points: the origin, the center of a circle c_a and the solution x^* . Clearly, the arc length l , which is just the minimum distance between $\frac{x^*}{\|x^*\|}$ and c_a , becomes longer as the x goes farther from the origin. In other words, the extreme point x^* determines the centroid maximizing the minimum distance between the centroid and all small circles bounding the solution region. (Q.E.D.)

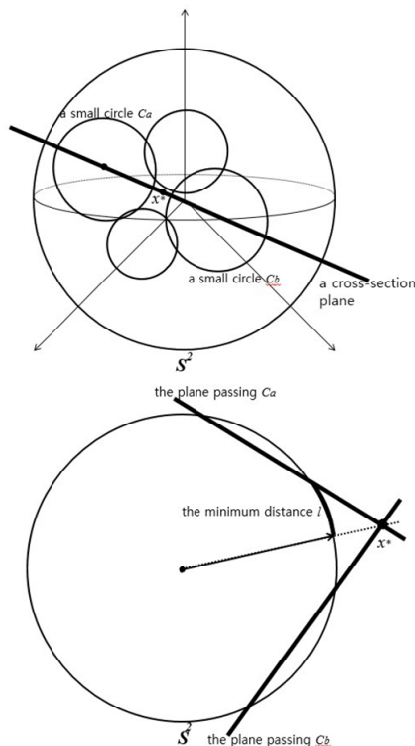


Fig. 7. Top View and Cross-section View about the Extreme Point of a Solution Space

By combining Lemma 4 and the LP technique [8] for quadratic objective function, we can get a result about the centroid in the intersection of great bands as follows.

Theorem 1. *Given a set of great bands $\{GB_1, \dots, GB_n\}$ on the unit sphere S^2 , the centroid of the intersection $\cap_{i=1}^n GB_i$ can be computed in $O(n)$ time.*

The next discussion is how to construct the two bounding circles: in-circle and circum-circle. The circum-circle sometimes called *the smallest enclosing circle* in two dimensional space can be constructed with a simpler formulation in an efficient time $O(n)$ [6]. The circumcircle bounding a spherical polygon on S^2 can be obtained by constructing *the smallest sphere* enclosing a set of points in 3D as Lemma 5. This paper presents more detailed theoretical foundation for this technique that was used in [2] without proof.

Lemma 5. *The intersection of S^2 and the smallest sphere s^* enclosing the vertices of a polygon on S^2 is the circum-circle of the spherical polygon.*

Proof. Let s^* be the smallest sphere enclosing the vertices of a spherical polygon in the three-dimensional space, as illustrated in Fig. 8. It can be shown that the intersection of s^* and S^2 is a great circle of s^* by a simple contradiction; if we assume the circle c generated by this intersection of the two spheres is not a great circle of s^* , there is another enclosing sphere smaller than s^* . As well as this great circle c of s^* circumscribes the vertices of the spherical polygon on S^2 , the center of c on S^2 and the center of s^* in E^3 have the same direction from the origin in E^3 . (Q.E.D.)

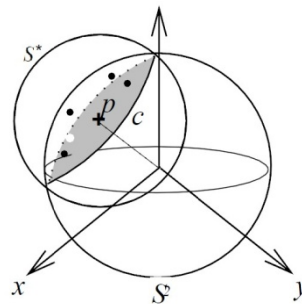


Fig. 8. The Smallest Sphere Enclosing a Set of Points

Finding the three edges of a convex polygon in two-dimensional space for its in-circle is a combinatorial problem. Up to ${}_nC_3$ circles may be tangential to three of n edges in the polygon. Even though there is an optimal $O(n \log n)$ algorithm [11] for this problem that is sometimes called *the largest empty circle*, the direct investigation takes costly implementation such as Voronoi diagrams. A novel result [12] of the relation between the in-circle and the circum-circle of a polygon on S^2 has been presented as.

Lemma 6. *If $Cr(p^C, \theta^C)$ and $Cr(p^I, \theta^I)$ are, respectively, the circum-circle and the in-circle of a polygon and its dual polygon on S^2 , then $p^C = p^I$ and $\theta^C + \theta^I = \frac{\pi}{2}$, and vice versa [12].*

By combining Lemma 5, 6 and the LP technique [6] for the smallest enclosing sphere, we can get a surprising result for determining the in-circle of a polygon on S^2 as follows.

Theorem 2. *The center and the size of the circle inscribing a polygon on S^2 can be determined in $O(n)$ time. (Note that it takes $O(n \log n)$ for a polygon in E^2 .)*

Finally, we can provide an efficient algorithm for obtaining the weakly-monotone centroid with sub-pockets and sub-lids of a polyhedron as follows.

procedure Weakly Monotone Centroid

input : the sub-pockets $\{SP_i\}$ and the sub-lids $\{SL_j\}$ of P

output : the weakly-monotone centroid of P

- 1 For all $i = 1, \dots, n_1, j = 1, \dots, n_2$, get $\{GCH_k(SP_i \cup SLF_{ij})\}, \{VCH_k(SP_i \cup SLF_{ij})\}$, where SLF_{ij} is each face $\in SL_j$ [2].
- 2 Determine the smallest $Cr(p_k^V, \theta_k^V)$ enclosing $\{VCH_1, \dots, VCH_n\}$ [13], [14]
- 3 Find the extreme x^* of $\bigcap_{k=1}^n SV(p_k, \theta_k^V, \pi - \theta_k^V)$ [15], [16], where $p_k = p_k^V$ and $\theta_k^V = \frac{\pi}{2} - \theta_k^V$.
- 4 If $\|x^*\| < 1$ then, terminate with the infeasibility.
- 5 Determine the smallest $Cr(p_k^G, \theta_k^G)$ enclosing $\{GCH_1, \dots, GCH_n\}$ [13], [14]
- 6 Find the extreme x^C of $\bigcap_{k=1}^n SV(p_k, \theta_k^C, \pi - \theta_k^C)$ [15], [16], where $p_k = p_k^G$ and $\theta_k^C = \theta_k^G$.
- 7 If $\|x^C\| \geq 1$ then return with the solution.

endProcedure Weakly Monotone Centroid

Different from the method for obtaining the strongly-and directionally-monotone centroids with original spherical polygons, the above procedure uses the small circles bounding the spherical polygons internally and externally. The following statement can be attached after the step 7 for the complete computation instead of our approximation. Our algorithm is just to avoid this full partitioning, which requires very complicated computations with $O(nk \log k + n \log n)$ time complexity and is only theoretical until now as far as the authors know.

”else partition S^2 with $\{GCH_1, \dots, GCH_n\}$ ”

The best advantage of our algorithm is easy and efficient to implement with linear time complexity since the robust and fast codes of which are available [13]-[16]. Furthermore, the steps 1-5 can be used when the feasibility should be quickly checked with light computations.

5. CONCLUSION

In this paper, we revisited the problems of determining the monotone directions of a polyhedron, which have been characterized with the sub-surfaces of the polyhedron [1], in order to find the centroid directions without constructing the boundary of all solutions.

The strongly- and directionally-monotone centroids of a polyhedron are shown to be obtained by simply applying the previous result [2] maximizing the intersection number of the positive and negative visibility polygons of sub-surfaces of the polyhedron.

The weakly-monotone directions of a polyhedron are reinterpreted as the intersection of a complementary set of the

positive and negative Gaussian polygons of sub-surfaces of the polyhedron. In order to avoid the computational complexity of non-convex regions, we generate two bounding circles: the in-circle and circum-circle of each Gaussian polygon; the complement for a pair of two positive and negative Gaussian polygons is approximated by a geometric primitive great band that can be treated more efficiently.

This paper presents an efficient algorithm for approximating the weakly-monotone centroid with the set of great bands that are often found in 3D manufacturability problems based on two theoretical results: Theorem 1 and 2.

ACKNOWLEDGEMENT

This research was supported by Woosuk University and Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2014R1A1A2055379).

REFERENCES

- [1] J. Ha and K. Yoo, “Characterization of polyhedron monotonicity,” *Computer-Aided Design*, vol. 38, no. 1, 2006, pp. 48-54.
- [2] J. Ha and K. Yoo, “Approximating centroids for the maximum intersection of spherical polygons,” *Computer-Aided Design*, vol. 37, no. 8, 2005, pp. 783-790.
- [3] G. Toussaint, *Computational Geometry*, Elsevier Science, 1985.
- [4] S. C. L. L. Chen and T. Woo, “Separating and intersecting spherical polygons: computing machinability on three-four- and five-axis numerically controlled machines,” *ACM Tr. on Graphics*, vol. 12, no. 4, 1993, pp. 305-326.
- [5] Lin-Lin Chen, Shuo-Yan Chou, and Tony C. Woo, “Parting directions for mould and die design,” *Computer-Aided Design*, vol. 25, no. 12, 1993, pp. 762-768.
- [6] E. Welzel, “Smallest enclosing disks (balls and ellipsoids),” *New Results and New Trends in Computer Science*, Springer Lecture Notes in Computer Science, vol. 555, 1991, pp. 359-370.
- [7] N. Megiddo, “Linear programming in linear time when the dimension is fixed,” *Journal of ACM*, vol. 31, no. 1, 1984, pp. 114-127.
- [8] M. Dyer, “Linear time algorithms for two- and three-variable linear programs,” *SIAM. Journal on Computing*, vol. 13, no. 1, 1984, pp. 31-45.
- [9] R. Seidel, “Small-dimensional linear programming and convex hulls made easy,” *Discrete Computing Geometry*, vol. 6, no. 5, 1991, pp. 423-434.
- [10] T. Woo, “Visibility maps and spherical algorithms,” *Computer Aided Design*, vol. 26, no. 1, 1994, pp. 6-16.
- [11] G. Toussaint, “Computing largest empty circles with location constraints,” *International Journal of Computer and Information Sciences*, vol. 12, no. 5, 1983, pp. 347-358.
- [12] J. G. Gan, T. C. Woo, and K. Tang, “Spherical maps: their construction, properties, and approximation,” *ASME J. Mechanical Design*, vol. 116, 1994, pp. 357-363.

- [13] N. Capens, an implementation of [6], [http://www.flipcode.com/archives/smallest enclosing spheres.shtml](http://www.flipcode.com/archives/smallest_enclosing_spheres.shtml)
- [14] CGAL, an implementation of [6], [http://www.cgal.org/manual/latest/dochtml/cgalmanual/bounding volumes ref/class min .sphere of spheres d.html](http://www.cgal.org/manual/latest/dochtml/cgalmanual/bounding_volumes_ref/class_min_sphere_of_spheres_d.html).
- [15] M. Hohmeyer, an implementation of [9], <http://www3.cs.stonybrook.edu/~algorithm/implement/linprog/distrib/source.a>.
- [16] CGAL, an implementation of [9], [http://www.cgal.org/manual/latest/doc/html/cgal manual/qp solver/chapter main.html](http://www.cgal.org/manual/latest/doc/html/cgal_manual/qp_solver/chapter_main.html).



Jon-Sung Ha

He is a professor who is working for department Game and Contents at Woosuk University, Korea. He received the B.E. in computer of computer engineering at Seoul National University, Korea in 1984, and also received M.S., Ph.D. in computer science from KAIST (Korea Advanced Institute of Science and Technology), Korea in 1986 and 1996, respectively. His research interests include computational geometry, computer graphics, CAD/CAM, and 3D digital contents.



Kwan-Hee Yoo

He is a professor who is working for department of Computer Education and department of IIE(information industrial engineering) at Chungbuk National University, Korea. He received the B.S. in computer Science from Chonbuk National University, Korea in 1985, and also received M.S., Ph.D. in computer science from KAIST (Korea Advanced Institute of Science and Technology), Korea in 1988 and 1995, respectively. His research interests include u-Learning system, computer graphics, 3D character animation, dental/medical applications.