POLYGON PROPERTIES CALCULATED FROM THE VERTEX NEIGHBORHOODS

Wm. Randolph Franklin

Electrical, Computer, and Systems Engineering Dept.
6026 J.E.C.
Rensselaer Polytechnic Institute
Troy, NY, 12180, USA

(518) 266-6077

franklin@csv.rpi.edu

ABSTRACT

Calculating properties of polyhedra given only the set of the locations and neighborhoods of the vertices is easy. Possible properties include volume, surface area, and point containment testing. No global topological information at all is explicitly needed (although the complete global topology could be recovered). The neighborhood of the vertex means the directions of the edges and faces on it but not their extents. These vertex-based formulae are dual to the usual formulae that use the faces. They have been implemented and the stability against inconsistent data tested. Alternative data structures and formulae for polyhedron calculation are important since special cases are a function partly of the data structure, and because different methods have different numerical accuracy and error detection properties.

ACKNOWLEDGEMENTS

This work was supported by the National Science Foundation under PYI grant no. ECS-8351942. It was partly performed at the Center for Interactive Computer Graphics at RPI.

INTRODUCTION

Topological considerations have been an important element of computer aided design for some time. An early polygon boolean combination algorithm was presented by [Eastman72]. A rigorous theoretical basis for manifold modeling and and some non-manifold modeling are discussed in [Weiler86]. Some of the topological degeneracies in hidden surface removal are solved by [Blinn]. Euler space operations are handled by[Mantyla82] and [Mantyla84]. An early mention of some topological issues in cartography was by [White77]. Using topology to convert wireframe models to surface models is presented by [Courter86]. Dual data structures for planar graphs are presented in [Guibas83]. An earlier version of some of the vertex neighborhood concepts was presented in [Franklin83].

Consider a polygon P that may be concave with multiple disconnected nested components, and whose complete description requires a directed tree showing the components' inclusions. It is in fact sufficient for most purposes to ignore the global topology, and work with only the set of edges, each containing the ordered list of its 2 vertices. However here each edge doesn't explicitly know which other components it is adjacent to, although that could be determined. For example, to determine point containment, we count how many edges cross a ray without needing to know how the edges are globally structured. To determine any mass property, such as area, we construct a triangle from each edge to the origin, calculate the property on that triangle, and sum. We can also determine Boolean combinations of the polygons with this data structure, [Franklin82]. Many others have also independently used similar algorithms. All the above also applies to polyhedra.

In this paper, we will dualize the formulae, first in 2-D, then in 3-D to work with the set of vertex neighborhoods, V_i, instead of edges. Each vertex will be represented thus:

$$V_i = (x_i, y_i, R_i, S_i)$$

where its location is (x, y). R and S are rays from V in the direction of the two edges incident on V. If we rotate from R to S in a positive direction we will sweep inside the polygon. See figure 1. Note that we do not know explicitly the other vertex that R passes through and also have no link in the data structure with the corresponding ray from that vertex that is opposite to R.

These data structures are particularly appropriate when we have a fixed set of infinite lines and are creating various polygons of complicated topology from a subset of the lines' intersections. We can determine such a polygon's mass and point containment properties without ever determining its topology.

The next sections will show i) the determination of a characteristic function $X(x,y)$ for the polygon from the vertex neighborhood, ii) an alternative data structure, the augmented ray, iv) the derivation of the mass properties, v) the 3-D case, vi) error detection, and finally vii) implementation.

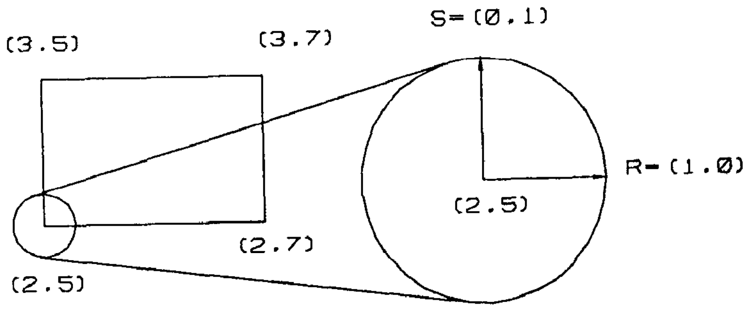

Figure 1: The Vertex Data Structure

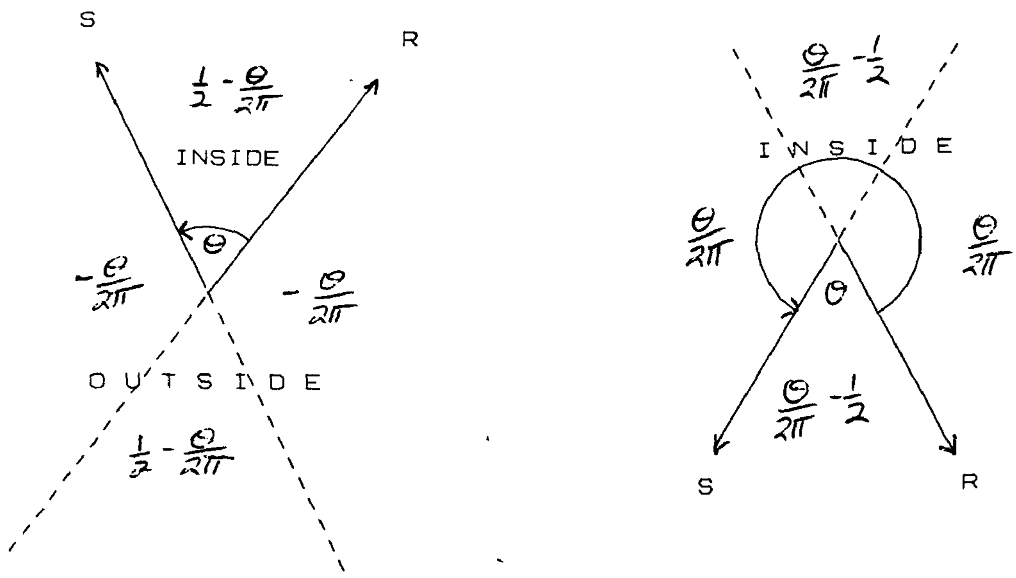

Figure 2: χ Function on Convex and Concave Vertices

POLYGON CHARACTERISTIC FUNCTION

Definition

$X(x,y)$ is a characteristic function of polygon P iff

$$X(x,y) = \begin{cases} 1 & \text{if } (x,y) \in P \\ 0 & \text{otherwise} \end{cases}$$

Definition

A *cross* function, $\chi(x,y)$ on each vertex is defined as shown in Figure 2. χ depends on the location of the vertex, the directions of the 2 edges adjacent to it, and which partial plane, or wedge, determined by the edges is on the inside. There are 2 cases depending on whether the angle, ϕ, at the vertex has $\phi > \pi$ or not. For convenience, we define θ as the angle $< \pi$ between the two edges. That is,

$$\theta = \begin{cases} \phi & \text{if } \phi < \pi \\ 2\pi - \phi & \text{otherwise} \end{cases}$$

χ assigns a value to every point, Q, in the plane depending on its relationship to V.

$$\chi(Q) = \begin{cases} \frac{1}{2} - \frac{\theta}{2\pi} & \text{if } \phi < \pi \text{ \& Q is in a wedge of size } \theta \\ -\frac{\theta}{2\pi} & \text{if } \phi < \pi \text{ \& Q is in a wedge of size } \pi - \theta \\ \frac{\theta}{2\pi} - \frac{1}{2} & \text{if } \phi > \pi \text{ \& Q is in a wedge of size } \theta \\ \frac{\theta}{2\pi} & \text{if } \phi > \pi \text{ \& Q is in a wedge of size } \pi - \theta \end{cases}$$

Example

In figure 3, we see the case of a $\pi/3$ angle.

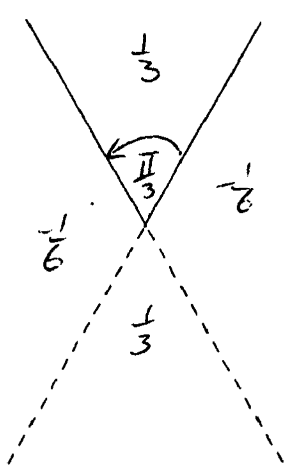

Figure 3: χ Function on $\frac{\pi}{3}$ Angle

Lemma

For all $R > 0$ $\int_{|Q-V| < R} \chi(Q)dQ = 0$.

Theorem

The characteristic function of the whole polygon is the sum of the χ functions of the vertices.

$$X(x,y) = \sum_{v \in P} \chi_v(x,y)$$

Proof

1. This can be proven for a triangle ABC by exhaustively considering the 7 regions of the plane around it, as shown in figure 4, where for several of the regions, we show the sum of, in order, $\chi_\alpha + \chi_\beta + \chi_\gamma$. The simplification uses $\alpha+\beta+\gamma = \pi$.

2. Then the formula is clearly true for a multiple component polygon composed of a disjoint set of triangles since the characteristic function of each component will add.

3. The characteristic function of an isolated hole is 0 outside and −1 inside so if the formula holds for the components of a polygon then it also holds when those components have holes.

4. To prove the case of general N-gons, we must show that if two adjacent triangles coalesce into one then the χ function at their common vertices adds. This is easy, as is shown in figure 5. Thus the characteristic function obtained by triangulating a polygon is the same as that obtained directly from the polygon's vertices.

Lemma

To determine whether a general point, Q, is inside polygon P, calculate

$$X_P(Q) = \sum_{v \in P} \chi_v(Q)$$

Then $Q \in P$ iff $X(Q) = 1$.

We have demonstrated a point containment formula that uses only the set of vertex neighborhoods.

χ was determined by postulating the existence of some such general function and then solving for the actual weights for each region. The solution given is unique. Alternatively the weights can be derived from the angular deficiency $(\pi-\phi)$ at each vertex since the angle deficiencies sum to 2π.

This 2-D characteristic function may be compared to the 1-D case. A polygon corresponds to a set of nonoverlapping line segments (l_i, r_i). The cross function for a general point, x, relative to a left endpoint, l_i is

$$\chi(x) = \begin{cases} \frac{1}{2} & \text{if } x > l_i \\ -\frac{1}{2} & \text{otherwise} \end{cases}$$

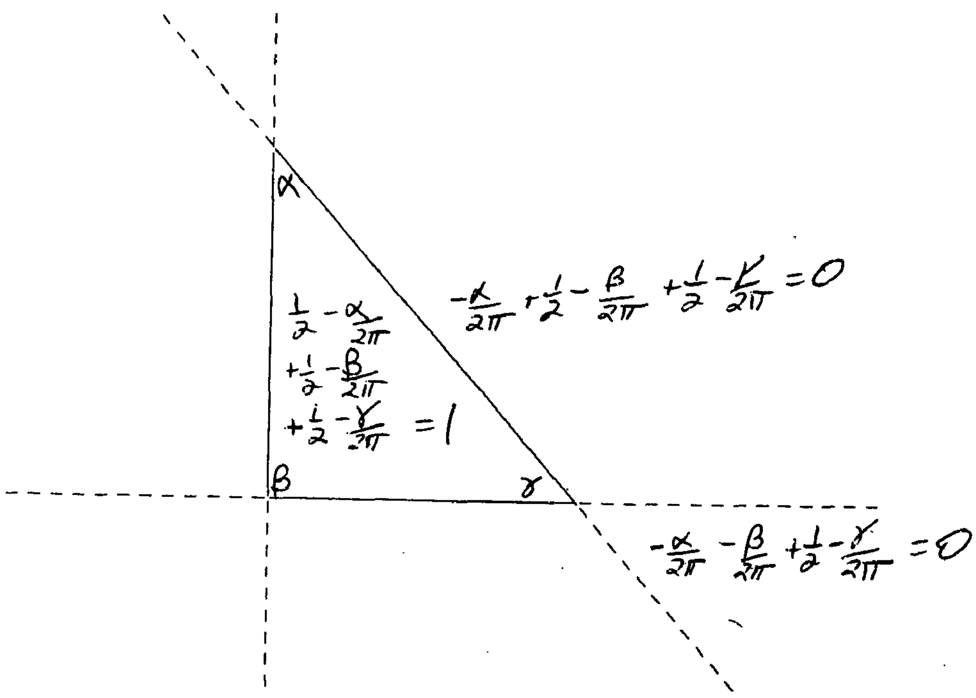

Figure 4: Proof of Characteristic Function for a Triangle

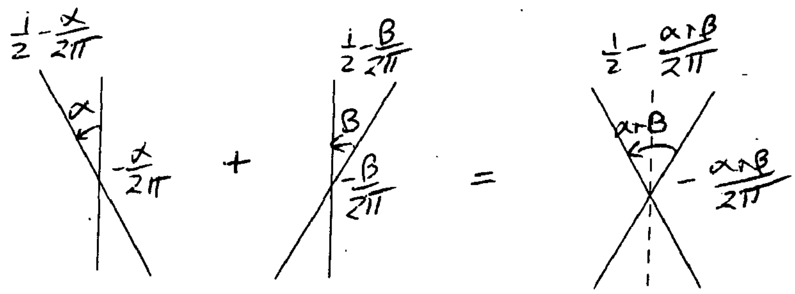

Figure 5a: Proof of Addition of χ Function for $\alpha+\beta<\pi$

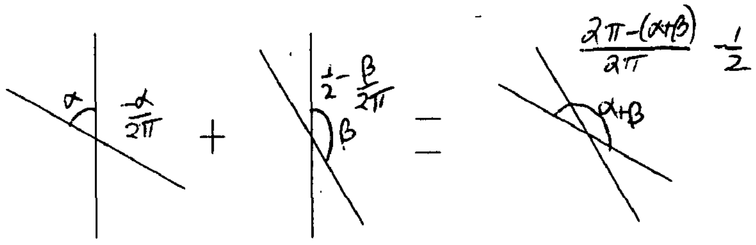

Figure 5b: Addition of χ for $\alpha+\beta>\pi$

The right endpoint is the reverse.

POLYGON MASS PROPERTIES

The characteristic function can be used to determine P's area, for example. We cannot calculate

$$\int_x \int_y \chi(x,y)\, dx\ dy$$

as the area of one χ since that diverges. However, we can use a sequence of weight functions, w_R, that get broader and broader and satisfy the following.

1. $\lim_{x^2+y^2 \to \infty} w_R(x,y) \to 0$ and

2. for all $\epsilon > 0$ and for all r there exists R_0 such that $R > R_0$ and $x^2+y^2 < r \ \rightarrow\ 1-\epsilon < w_R(x,y) < 1+\epsilon$

Then we could calculate the area as

$$A = \lim_{R \to \infty} \sum_{v \in V} \int_x \int_y w_R(x,y)\, \chi(x,y)\, dx\ dy$$

The principle is illustrated by the trivial 1-D case, where the weighted lengths of the cross functions for l_i and r_i are $-l_i$ and r_i respectively, giving the length of the line as $\sum r_i - \sum l_i$. This can be derived from the following weight sequence:

$$w_R(x) = \begin{cases} 1 & \text{if } |x| < R \\ 0 & \text{otherwise} \end{cases}$$

It is not necessary to know which l_i corresponds to which r_i, although that could be determined by sorting.

In 2-D we will use the sequence of weight functions:

$$w_R(x,y) = \begin{cases} 1 & \text{if } |x| < R \text{ and } |y| < \rho R \\ 0 & \text{otherwise} \end{cases}$$

ρ is a constant that is chosen so that for sufficiently large R no extended edges intersect the top or bottom of the weight box, but all go through the sides. That is, if the edges' slopes are m_i then $\rho > \max_i \{m_i\}$. If we assume that

i) the angles of the two edges of the χ (relative to the X-axis) are α and β,

ii) $-\pi/2 < \alpha < \beta < \pi/2$, and

iii) the center point of the χ is (x_0, y_0)

then the weighted area of χ is

$$A = \int_x \int_y w_R(x,y)\, \chi(x,y)\, dx\ dy$$

$$= \begin{cases} \left(\frac{1}{2} R^2 + x_0^2\right)(\tan\alpha - \tan\beta) - \frac{\alpha-\beta}{\pi}\rho R^2 \\ \quad \text{if the point } (0,\infty) \text{ has a positive weight} \\ -\left(\frac{1}{2} R^2 + x_0^2\right)(\tan\alpha - \tan\beta) - \frac{\alpha-\beta}{\pi}\rho R^2 \\ \quad \text{otherwise} \end{cases}$$

Although normally one can ignore terms tending to zero in the limit, here we ignore terms tending to ∞ in the limit, since we know that for well-formed polygons they will cancel. This gives a modified weighted area of χ as

$$A' = \begin{cases} \frac{1}{2} x^2 (\tan\alpha - \tan\beta) \\ \quad \text{if the point } (0,\infty) \text{ has a positive weight} & (1) \\ -\frac{1}{2} x^2 (\tan\alpha - \tan\beta) & \text{otherwise} \end{cases}$$

These modified weighted areas are not unique; other choices of w_R will give others. However so long as we are consistent for every vertex of a given polygon, then the mass properties are invariant irrespective of the w_R.

Example

Consider the triangle $A(0,0)$, $B(2,0)$, $C(1,\sqrt{3})$. The weighted areas of the cross functions at each vertex are, respectively,

$$\frac{1}{2}\sqrt{3}\, x_A^2$$

$$\frac{1}{2}\sqrt{3}\, x_B^2$$

$$-\sqrt{3}\, x_C^2$$

giving the triangle's area, correctly, as $\sqrt{3}$.

An alternative derivation of χ's weighted areas without using limits follows from the observation that equation (1) is the area that the leftmost wedge of χ has when cut by the y-axis. The other possible w_R are obtained by cutting the cross function by other lines, which need not be straight. For example, if all the vertices are in the first quadrant, then a cutting line composed of the positive x-axis and positive y-axis is sufficient.

An alternative mass property calculation method is given in the next section.

In [Franklin83] each vertex neighborhood is split into a set of two *augmented rays*, as shown in figure 6.

Definition

An augmented ray is the tuple

$$A = (V, N, b)$$

where $V = (x, y)$ is the starting vertex, N is a unit vector in the direction of the edge, and b is a bit flag that is 1 if the polygon's inside is to the left of the ray and -1 otherwise. Once the vertices are reformated as rays, the two rays from each vertex are not explicitly connected in the data structure.

Now for any mass property, such as area, there is a functional from it to a function, f_{area}, on the rays such that the total polygon area is

$$\text{area} = \sum_{A \epsilon P} f_{area}(A)$$

Lemma

$$f_{area}(A) = \frac{-b}{2} v \cdot N \;\; V \times N \cdot e_3$$

where e_3 is the unit vector (0,0,1), and $\cdot$ and $\times$ are the vector dot and cross products.

Proof

Consider the triangle defined by O, V, and F, the foot of the perpendicular dropped from O to the ray from V. The two rays defining an edge have the same F. See figure 7, where the edge V_1V_2 defines the 2 augmented rays A_1 and A_2. If

$$N_1 = \frac{V_1V_2}{|V_1V_2|}$$

and

$$N_2 = -N_1$$

then

$$A_1 = (V_1, N_1, -1)$$

and

$$A_2 = (V_2, N_2, 1)$$

Lemma

$$f_{perimeter} = - V \cdot N$$

Proof

Sum the length VF for each ray.

A modified version of this could also be used for point containment, although using characteristic functions is simpler. Let

$$f_{contain} = \begin{cases} 1 & \text{if } (Q-V) \cdot N > 0 \;\&\; (Q-V) \times N \cdot e_3 > 0 \\ 0 & \text{otherwise} \end{cases}$$

Then $Q \epsilon P$ iff $\mathrm{mod}(\sum f_{contain}(V), 2) = 1$.

POLYGONS WITH CURVED EDGES

Polygons whose edges are finite segments of curves can also be represented by rays provided that each curve has a unique distinguished point that is independent of the segment's endpoints. Although dropping a normal from the origin to the curve is not always unique, the distinguished point of an ellipse, for example, can be the point at y_{max}.

3-D POINT CONTAINMENT

Definition

A cross function $\chi(Q)$ for a trihedral vertex V of a polyhedron assigns a value to each point, Q of E^3 depending on Q's relation to the planes of the 3 faces on V. Let the faces' internal angles at V be α, β, and γ for F_1, F_2, and F_3 resp. Whereas the 2-D χ was symmetric around V, the 3-D case is antisymmetric.

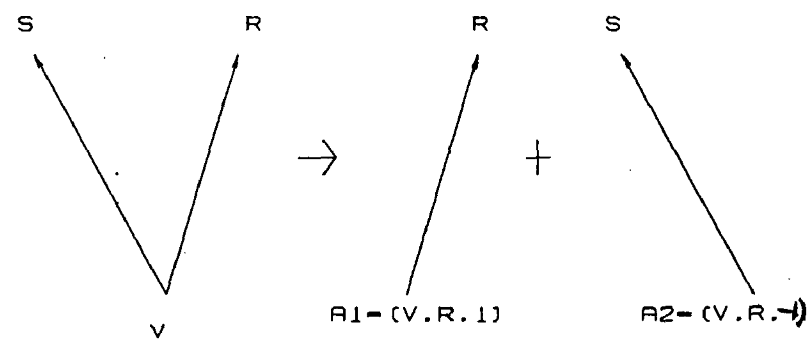

Figure 6: Splitting a Vertex into 2 Rays

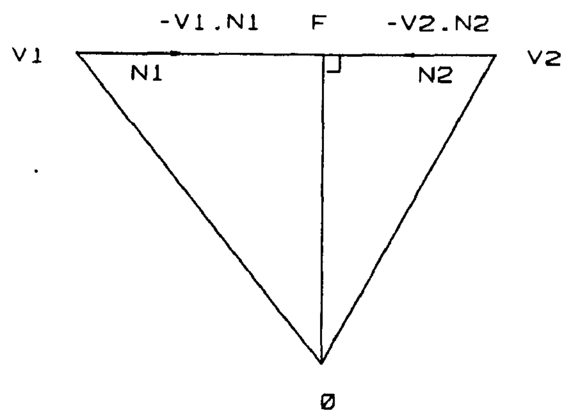

Figure 7: Calculation of the Area of V_1V_2O from the 2 Rays

$$\chi(Q) = \begin{cases} \frac{2\pi-\alpha-\beta-\gamma}{4\pi} & \text{if } Q \text{ is inside all 3 face planes} \\ \frac{\alpha-\beta-\gamma}{4\pi} & \text{if } Q \text{ is outside } F_1 \text{ but inside } F_2 \text{ and } F_3 \\ \frac{\alpha+\beta-\gamma}{4\pi} & \text{if } Q \text{ is outside } F_1 \text{ and } F_2 \text{ but inside } F_3 \\ \frac{\alpha+\beta+\gamma-2\pi}{4\pi} & \text{if } Q \text{ is outside all 3 face planes} \end{cases}$$

The other 4 cases, such as outside F_2 and inside F_1 and F_3 are similar.

Example

For a tetrahedron vertex, $\alpha = \beta = \gamma = \frac{\pi}{3}$, and $\chi(Q) = \frac{1}{4}, -\frac{1}{12}, \frac{1}{12}$, and $-\frac{1}{4}$ for points inside 3, 2, 1, or 0 face planes respectively.

Lemma

The characteristic function of a polyhedron, all of whose vertices have 3 faces, is obtained by adding the cross functions of its vertices.

Proof

Similar to the 2-D case.

Definition

The deficiency, δ_V, at a vertex, V, of a polyhedron is the amount by which the sum of the face angles at that vertex is less than 2π.

Example

$\delta_V = \pi/2$ for each vertex of a cube.

Theorem

For a polyhedron of genus 0, $\sum_V \delta_V = 4\pi$.

The 3-D characteristic formula could be derived from this theorem, which is a discrete version of the Gauss-Bonnet theorem over a closed surface:

$$\iint_S K \, dA = 4\pi(1-g)$$

where K is the Gaussian curvature at each point and g is the genus of the surface.

When more than three faces meet at a vertex, it may be simplest to decompose it into trihedral components.

3-D MASS PROPERTIES

Definition

A 3-D *wedge* is the analog to a 2-D augmented ray. It is the triple

$$W = (V, R, S)$$

where $V = (x, y, z)$ is the vertex of the polyhedron P, and R and S are the rays of 2 adjacent edges on the vertex that have a face between them. $R \times S$ is a vector out of the plane of the face. $|R| = |S| = 1$.

Just as in 2-D, there are mass property functions corresponding to each mass property.

Lemma

$$f_{edgelength} = -V \cdot R - V \cdot S$$

and the total edge length of the polyhedron is

$$\sum_{W \in P} f_{edgelength}(W)$$

Lemma

$$f_{volume} =$$

$$\frac{1}{6} \frac{V \cdot R \times S}{1-(R \cdot S)^2} \left(2V \cdot R \; V \cdot S - R \cdot S\left(\left(V \cdot R\right)^2 + \left(V \cdot S\right)^2\right)\right)$$

Proof

Drop lines from the origin onto rays R and S and onto the plane defined by them. Together with V, they define a tetrahedron whose volume is the above expression. The set of such tetrahedra partitions the polyhedron.

ERROR DETECTION AND ANALYSIS

For a valid polygon or polyhedron to be defined the cross functions and rays must be mutually consistent. If the rays on which the cross functions are defined are not consistent, then there will be regions of the plane, between the two rays that should coincide, in which the so-called characteristic function will have a value that is neither 0 nor 1, so the error will be self-evident. This behavior was verified by implementation as described in the next section.

If the rays being used to determine mass properties do not coincide, then the calculated error will be proportional to the discrepancy. More precisely, assume that the data for each ray is perturbed:

$$V_i' = V_i + dV$$

$$N_i' = N_i + dN$$

Although each vertex and normal is represented twice in the ray data structure, we assume that these two occurrences are perturbed independently so that the data structure is inconsistent.

Lemma

The effect on the perimeter is

$$dP = \sum_i dV_i \cdot N_i + V_i \cdot dN_i + dV_i \cdot dN_i$$

which is linear in the input error.

Lemma

The error in the area formula is linear in the input error.

Lemma

The point containment formula produces an error only when the point lies between the two inconsistent rays that should be coincident.

Thus these data structures are reasonably stable.

IMPLEMENTATION

The characteristic function algorithm was implemented and use to test point inclusion with various points on some convex polygons. The only interesting matter was calculating the angle inclusions correctly; given this, the results were as expected for points not on an extended edge. Extended edges are complicated since they are the boundary between two regions of some cross function.

To test the numerical robustness of the algorithm, the database was perturbed so that the cross functions did not align. Points between two lines that should have been coincident gave a value between 0 and 1 for the characteristic function, while points is a consistent region of the plane gave the correct result. This was as expected. The probability of a point being in an inconsistent region is proportional to the size of the angle errors in the cross functions.

SUMMARY

We have presented a new method for determination of polygon and polyhedron properties, calculation from the vertices and their local neighborhoods, that is dual to the usual method of calculation from the faces. The method is efficient, and is stable against inconsistencies in the input data.

REFERENCES

Blinn.

James F. Blinn, *A Method for Removing Some Degeneracies in Hidden Line and Ray Tracing Calculations,* Jet Propulsion Lab, undated.

Courter86.

S. Mark Courter and John A. Brewer III. "Automated Conversion of Curvilinear Wire-Frame Models to Surface Boundary Models; A Topological Approach," *ACM Computer Graphics,* vol. 20, no. 4, pp. 171-178, August 1986. SIGGRAPH'86 Conference Proceedings.

Eastman72.

C.M. Eastman and C.I. Yessios, *An Efficient Algorithm for Finding the Union, Intersection, and Differences of Spatial Domains,* Sept. 1972. Carnegie-Mellon University, Dept. of Computer Science

Franklin82.

Wm. Randolph Franklin, "Efficient Polyhedron Intersection and Union," *Proc. Graphics Interface'82,* pp. 73-80, Toronto, 19-21 May 1982.

Franklin83.

Wm. Randolph Franklin, "Rays - New Representation for Polygons and Polyhedra," *Computer Graphics and Image Processing* , vol. 22, pp. 327-338, 1983.

Guibas83.

L.J. Guibas and J. Stolfi, "Primitives for the Manipulation of General Subdivisions and the Computation of Voronoi Diagrams," *Proc. 15th ACM Symposium on the Theory of Computing,* pp. 221-234, April 1983. also in ACM T. Graph. 4, 1985, 74-123.

Mantyla82.

M. Mantyla and R. Sulonen, "GWB: A Solid Modeler with the Euler Operators," *IEEE Computer Graphics and Applications,* vol. 2, no. 7, pp. 17-31, September 1982.

Mantyla84.

M. Mantyla, "A Note on the Modeling Space of Euler Operators," *Computer Vision, Graphics, and Image Processing,* vol. 26, pp. 45-60, 1984.

Weiler86.
Kevin J. Weiler, Topological Structures for Geometric Modeling, Center for Interactive Computer Graphics, Rensselaer Polytechnic Institute, August 1986. PhD Thesis.

White77.
Denis White, "A New Method of Polygon Overlay," *An Advanced Study Symposium on Topological Data Structures for Geographic Information Systems,* Laboratory for Computer Graphics and Spatial Analysis, Harvard University, Cambridge, MA, USA, 02138, Oct. 16-21, 1977.