AGGRES: A Program for Computing Power Crusts of Aggregates

Javier Bernal

National Institute of Standards and Technology,

Gaithersburg, MD 20899, USA

Abstract

AGGRES, a Fortran 77 program for computing a power crust of an aggregate, is discussed. AGGRES takes a finite set of points from the surface of an aggregate, i. e., a three-dimensional object with no holes that contains its center of mass in its interior, and computes a piecewise-linear approximation of the surface of the object called a power crust. AGGRES is based on an algorithm by Amenta, Choi and Kolluri for computing power crusts using power diagrams. Besides a power crust of the object, the program also produces the area of the power crust and the volume of the solid it encloses.

1 Introduction

Let F be the surface of an object in 3−dimensional space ($\mathcal{R}^3$), and let G be a finite set, possibly empty, of points in $\mathcal{R}^3$. By the *medial axis* of $F \cup G$ we mean the closure of the set of points in $\mathcal{R}^3$ with more than one closest point in $F \cup G$. Note that the medial axis of $F \cup G$ is usually a 2−dimensional surface that includes both a part inside F and a part outside F. In addition, each point in the medial axis is the center of a *medial ball*, i. e., a maximal ball that does not contain points of $F \cup G$ in its interior.

Let S be a finite set of points in $\mathcal{R}^3$. For each point p in S let w_p be a real-valued weight assigned to p. Given p in S and a point x in $\mathcal{R}^3$, the *power distance* of x from p, denoted by $\pi_p(x)$, is defined by

$$\pi_p(x) \equiv |xp|^2 - w_p,$$

where $|xp|$ is the Euclidean distance between x and p. For a point p in S define the *power cell* of p as the set of points in $\mathcal{R}^3$ that are as close in power distance to p as they are to any other point in S. Each power cell is a convex polyhedron, possibly empty, and the collection of power cells, called the *power diagram* of S [4], is a partition of $\mathcal{R}^3$. We observe that if the weights of the points in S are all equal then the power diagram of S is also called the *Voronoi diagram* of S [9], and the power cell of a point p in S is called the *Voronoi cell* of p. A point v that is a vertex of a Voronoi cell in the Voronoi diagram of S is called a *Voronoi vertex* of S and has the property that it is the center of a ball, called the *Voronoi ball* at v, that contains at least four points of S in its boundary but no points of S in its interior.

Let F be the surface of an object in $\mathcal{R}^3$, let S be a finite set of points in F, and let G be a finite set, possibly empty, of points in $\mathcal{R}^3$. Let I and E be disjoint non-empty subsets of the set of Voronoi vertices of $S \cup G$, and for each point in $I \cup E$ assign a weight to the point equal to the square of the radius of the Voronoi ball at the point. Using these weights, the *power crust* of the object relative to S, G, I and E is the collection of polygons that are facets in common of power cells of points in I and power cells of points in E in the power diagram of $I \cup E$.

In this paper we discuss AGGRES, a Fortran 77 program for computing a power crust of an aggregate, i. e., a $3-$dimensional object that has no holes and that contains its center of mass in its interior, in such a way that the computed power crust is a piecewise-linear approximation of the surface of the aggregate. Let F be the surface of the aggregate, let S be a finite set of points in F, and let G be the finite set made up of the center of mass of the aggregate and the eight vertices of a large box containing the aggregate. Program AGGRES first computes the Voronoi diagram of $S \cup G$. Assuming that the set of points S is sufficiently dense in the surface F [2, 3], program AGGRES then chooses a subset P of the set of Voronoi vertices of $S \cup G$ that approximates the medial axis of $F \cup G$. To each point in P program AGGRES assigns a weight equal to the square of the radius of the Voronoi ball at the point, and with these weights computes the power diagram of P. Using this power diagram, program AGGRES then identifies disjoint non-empty subsets I and E of P whose union is P. Program AGGRES chooses I

so that it approximates the part of the medial axis of $F \cup G$ inside F, and chooses E so that it approximates the part of the medial axis of $F \cup G$ outside F. Consequently, the union of the power cells of points in I in the power diagram of P approximates the union of the Voronoi balls at points in I, which in turn approximates the representation of the aggregate in terms of the union of the medial balls with centers inside F [2, 3]. Finally, program AGGRES produces the power crust of the aggregate relative to S, G, I and E, together with the area of this power crust and the volume of the solid it encloses. A copy of program AGGRES as well as of other programs it calls such as REGTET [5], PRWVTX, can be obtained from `http://math.nist.gov/~JBernal`.

2 The Algorithm

Program AGGRES is based on an algorithm for computing power crusts developed and analyzed by Amenta, Choi and Kolluri [2, 3]. Given an aggregate, let F be its surface, let S be a finite set of points in F, and let G be the finite set made up of the center of mass of the aggregate and the eight vertices of a large box containing the aggregate. Assuming that S is sufficiently dense in F [2, 3], the algorithm consists essentially of five steps:

1. Compute Voronoi diagram of $S \cup G$.

2. Identify a subset P of the set of Voronoi vertices of $S \cup G$ that approximates the medial axis of $F \cup G$.

3. Assign to a each point in P a weight equal to the square of the radius of the Voronoi ball at the point, and compute the power diagram of P with these weights.

4. From power diagram of P partition P into two sets I and E so that I approximates the part of the medial axis of $F \cup G$ inside F, and E approximates the part outside F.

5. Identify the collection of facets in common of power cells of points in I and power cells of points in E, i. e., the power crust of the aggregate relative to S, G, I and E.

Program AGGRES computes the Voronoi diagram in step 1 above by first calling program REGTET [5] which computes a Delaunay tetrahedralization of $S \cup G$ [6], and then calling program PWRVTX which extracts the Voronoi diagram from the Delaunay tetrahedralization. Similarly, program AGGRES computes the power diagram in step 3 above by first calling program REGTET which computes a regular tetrahedralization of P [8], and then calling program PWRVTX which extracts the power diagram from the regular tetrahedralization.

In what follows we describe the processes with which program AGGRES identifies and partitions in steps 2 and 4 above the subset P of the set of Voronoi vertices of $S \cup G$.

3 Identification of Poles

Let p be a point in S. By the choice of the set G, the Voronoi cell of p in the Voronoi diagram of $S \cup G$ is bounded. Two points, called the *poles* of p, are associated with p [1]. The first pole, say p_1, is the Voronoi vertex of p farthest from p. The second pole, say p_2, is the Voronoi vertex of p farthest from p such that the inner product of $p_1 - p$ and $p_2 - p$ is negative. Assuming that the surface F is smooth, that S is sufficiently dense in F, and that there is no noise in the data associated with the coordinates of the points in S, the Voronoi cell of p is then thin and perpendicular to F, and the poles of p lie near the medial axis of $F \cup G$, one inside F, the other outside F. In addition, the set of poles of all points in S approximates the medial axis of $F \cup G$ [2, 3].

Because F may not be smooth, S may not be sufficiently dense in F, and there may be noise in the data associated with the coordinates of the points in S, the set of poles of all points in S may not be a good approximation of the medial axis of $F \cup G$. However a subset of this set may be a good enough approximation for the purpose of computing as described above a power crust that approximates the surface F.

Program AGGRES identifies in step 2 of the algorithm a subset P of the set of Voronoi vertices of $S \cup G$ which is actually a subset of the set of poles of all points in S. Again let p be a point in S, and let p_1 and p_2 be the poles of p with p_1 the pole farthest from p. Program AGGRES decides whether p_1 and/or p_2 are part of the set P according to a criterion by Dey, Giesen, Leekha and Wenger [7]. Let p' be a point in the Voronoi cell of p farthest from p such that the inner product of $p_1 - p$ and $p' - p$ is zero. Given a parameter ρ (usually equal to 8), program AGGRES identifies p_1 as a point in P if

$$\rho \cdot |pp'| \leq |pp_1|$$

where $|pp'|$ and $|pp_1|$ are the Euclidean distances between p and p', and p and p_1, respectively. Similarly for p_2.

4 Partitioning of Poles

The set of poles P is chosen to approximate the medial axis of $F \cup G$. Given a pole in P, the Voronoi ball at the pole is called an *inner ball* if the pole is in the subset of P that approximates the medial axis of $F \cup G$ inside F, and it is called an *outer ball* if the pole is in the subset of P that approximates the medial axis of $F \cup G$ outside F. Assuming that the surface F is smooth, that S is sufficiently dense in F, and that there is no noise in the data associated with the coordinates of the points in S, each inner ball is then almost entirely inside F and each outer ball is almost entirely outside F [2, 3]. Consequently, the union of the inner balls approximates the representation of the aggregate in terms of the union of the medial balls with centers inside F, and an inner ball and an outer ball can only intersect in a shallow manner [2, 3].

Let v and w be poles in P that form a *power neighbor pair*, i. e., whose power cells in the power diagram of P have a facet in common. If the Voronoi balls at v and w have nonempty intersection and one does not contain the other, let q be a point that lies in both the surface of the Voronoi ball at v and the surface of the Voronoi ball at w. The *intersection angle* of the two balls is then defined as $\pi - \alpha$ where α is the angle at the vertex q of the triangle with vertices v, w, q.

The intersection angle of two Voronoi balls corresponding to a power neighbor pair is a measurement of the depth of the intersection of the two balls: the larger the intersection angle the deeper the intersection. In step 4 of the algorithm program AGGRES partitions Voronoi balls into inner and outer balls by taking advantage of the fact that the intersection angle of an inner ball and an outer ball corresponding to a power neighbor pair must be small. It does this by first computing all intersection angles, one per power neighbor pair, and placing them in a stack in decreasing order. The first Voronoi balls that are marked as either outer or inner are those whose surfaces contain at least one point in G. If the surface of the ball contains a corner of the big box that contains the aggregate then the ball is marked as outer. If it contains the center of mass of the aggregate then it is marked as inner. Next, the intersection angle at the top of the stack is repeatedly removed until the stack is empty. Each time the intersection angle is removed the corresponding power neighbor pair is identified. If the Voronoi ball at one and only one of the poles forming the pair has been previously marked as either inner or outer then the other ball is marked in the same manner. By traversing the power diagram other Voronoi balls that were previously marked as being associated with the newly marked ball are similarly marked. If neither of the Voronoi balls at the poles forming the pair has been previously marked as either inner or outer then the two balls are marked as being associated with each other, and by traversing the power diagram other balls that were previously marked as being associated with either of the two balls are marked as being associated with both, and vice versa. Finally the set of poles P is partitioned into sets I and E, where I is the set of poles that are centers of inner balls, and E is the set of poles that are centers of outer balls.

We note that as long as the set of poles P is a good enough approximation of the medial axis of $F \cup G$, program AGGRES usually partitions P as described above in an adequate manner. This is the case even when F is not smooth, S is not sufficiently dense in F, or there is noise in the data associated with the coordinates of the points in S.

5 Estimate of Center of Mass

Program AGGRES computes a point that is a rough estimate of the center of mass of the aggregate. Using this point instead of the actual center of mass works well as long as the point is in the interior of the aggregate and sufficiently far away from its surface.

Define real numbers $xmin$, $xmax$, $ymin$, $ymax$, $zmin$, $zmax$ by

$$\begin{aligned}
xmin &\equiv \min\{x : \exists\ y,\ z,\ (x,y,z) \in S\},\\
xmax &\equiv \max\{x : \exists\ y,\ z,\ (x,y,z) \in S\},\\
ymin &\equiv \min\{y : \exists\ x,\ z,\ (x,y,z) \in S\},\\
ymax &\equiv \max\{y : \exists\ x,\ z,\ (x,y,z) \in S\},\\
zmin &\equiv \min\{z : \exists\ x,\ y,\ (x,y,z) \in S\},\\
zmax &\equiv \max\{z : \exists\ x,\ y,\ (x,y,z) \in S\}.
\end{aligned}$$

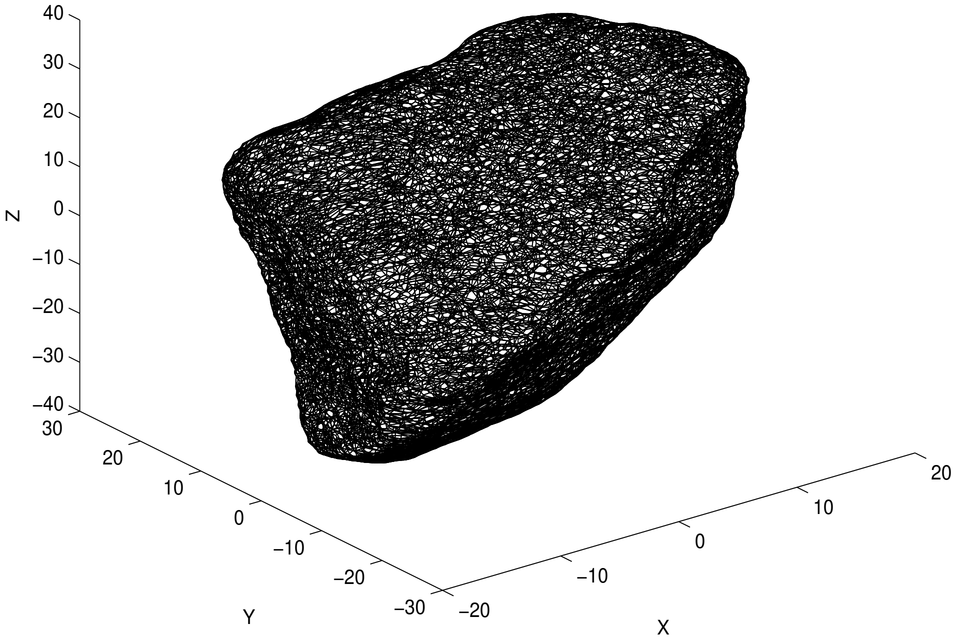

Figure 1: The surface of an aggregate as approximated by the power crust of the aggregate.

Define real numbers $xctr$, $yctr$,$zctr$ by

$$\begin{aligned} xctr &\equiv (xmax + xmin)/2, \\ yctr &\equiv (ymax + ymin)/2, \\ zctr &\equiv (zmax + zmin)/2. \end{aligned}$$

The estimate of the center of mass of the aggregate, say $\overline{p}$, can now be defined by

$$\bar{p} \equiv (xctr, yctr, zctr).$$

6 Numerical Results

Program AGGRES has been executed for several data sets. Figure 1 shows a reconstruction of the surface of an aggregate as a power crust of the aggregate that was computed with program AGGRES. The input data set consisted of 43,026 laser scanned points. Program AGGRES identified 34,920 poles and the resulting power crust had 47,147 facets. The area of the power crust and the volume of the solid enclosed by the power crust as computed by program AGGRES were respectively 8,205.5363 sq mm and 45,856.2344 cu mm. Finally, the execution time of the program was about 14 CPU minutes on a Sun Ultra80.[1]

Another data set consisted of 25,000 randomly generated points on the surface of a sphere of radius 100 (unspecified units). Program AGGRES then computed the surface area and volume of the sphere to be 125,637.266 and 4,186,665.058, respectively. Since the true surface area and volume of a sphere of radius 100 are approximately 125,663.706 and 4,188,790.200, respectively, it follows that the differences in surface area and volume are about 0.02 % and 0.05 %, respectively, of the true values.

References

[1] Amenta, N., Bern, M.: Surface reconstruction by Voronoi filtering. Discr. Comput. Geom. **22** (1999) 481–504.

[2] Amenta, N., Choi, S., Kolluri, R.: The power crust, unions of balls and the medial axis transform. Comput. Geom.: Theory Applications **19** (2001) 127–153.

[3] Amenta, N., Choi, S., Kolluri, R.: The power crust. Proc. 6th Annu. Sympos. Solid Modeling Applications (2001) 249–260.

[4] Aurenhammer, F.: Power diagrams: properties, algorithms and applications. SIAM J. Comput. **16** (1987) 78–96

[5] Bernal, J.: REGTET: A program for computing regular tetrahedralizations. NISTIR 6786 (2001)

[6] Delaunay, B.: Sur la sphère vide. Bull. Acad. Sci. USSR (VII), Classe Sci. Mat. Nat. (1934) 793–800

[1]The identification of any commercial product or trade name does not imply endorsement or recommendation by the National Institute of Standards and Technology.

[7] Dey, T. K., Giesen, J., Leekha, N., Wenger, R.: Detecting boundaries for surface reconstruction using co-cones. Int. J. Comput. Graphics & CAD/CAM **16** (2001) 141–159

[8] Edelsbrunner, H., Shah, N. R.: Incremental topological flipping works for regular triangulations. Algorithmica **15**(3) (1996) 223–241

[9] Voronoi, G.: Nouvelles applications des paramètres continus à la théorie des formes quadratiques. J. Reine Angew. Math. **134** (1908) 198–287