Equidecomposable Quadratic Regions

Thomas C. Hales

Math Department, University of Pittsburgh, Pittsburgh, PA 15217
E-mail: hales@pitt.edu

Abstract

This article describes an algorithm that decides whether a region in three dimensions, described by quadratic constraints, is equidecomposable with a collection of primitive regions. When a decomposition exists, the algorithm finds the volume of the given region. Applications to the 'Flyspeck' project are discussed.

1 Introduction

From ancient times, a common approach to computing the volume of a region A is to dissect A into finitely many pieces and then reassemble those pieces into finitely many new regions whose volumes have been previously determined.

This is the motivating idea behind Hilbert's third problem on polyhedra. He asked if any two polyhedra of the same volume are equidecomposable. That is, can one polyhedron be cut into finitely many polyhedral pieces, which can be reassembled into the other polyhedron. M. Dehn answered this question negatively in 1902. For example, a regular tetrahedron is not equidecomposable with a cube of the same volume. The proof of this result is that the cube and the tetrahedron have different Dehn invariants, but all equidecomposable polyhedra have the same Dehn invariant.

In two dimensions, the corresponding result is true: any two polygons of the same area are equidecomposable. That is, if they have the same area, the first polygon can be cut into finitely many triangles in such a way that they can be reassembled into the second polygon.

The mathematical literature on equidecomposability has various extensions. The Banach-Tarski paradox is one of the best known of these results: it is possible to cut a ball into finitely many (non-measurable) pieces and reassemble them into a two balls of the same radius as the first. Another is the classical squaring-the-circle problem: M. Laczkovich proved that it is possible to cut a square into finitely many pieces that can be reassembled into a disk of the same area [6].

In this article we look at regions that are defined by a restricted class of quadratic constraints. Each face of the boundary is required to be planar, spherical, or right conical. We call these quadratic regions. Among the quadratic regions are a special subset that we call primitive. All the primitive regions are familiar shapes. The question we ask is when a quadratic region is equidecomposable with a finite disjoint collection of primitive quadratic regions. When such a decomposition can be produced, we obtain a formula for the volume of the quadratic region in terms of primitives.

This article describes an algorithm to decide whether a quadratic region admits such an equidecomposition and to produce the decomposition when it exists. Our analysis of the algorithm is reminiscent of the Dehn invariant, which is an invariant attached to the edges of a polyhedron. Similarly, most of our analysis is focused on the curves formed by the intersection of two quadric surfaces.

The study of intersections of conics is classical. In Descriptive Geometry, this was one of the topics "which possess industrial utility and which develop the qualities of mind so essential in a draftsman" [5]. More recently, the intersection of quadric surfaces has become relevant for Computer Aided Geometric Design. For a guide to the recent literature on this subject, see the survey in the introduction to [2].

This algorithm has applications to the proof of the Kepler conjecture, and the ongoing project to formalize that proof. All of the volume calculations that arise in the proof of the Kepler conjecture are volumes of quadratic regions. In every case, these regions are equidecomposable with a finite disjoint collection of primitive regions. Thus, all of the volume calculations of that proof can be handled by an automated procedure.

2 Petal Figures

To motivate what is to come, we begin with the simple case of two dimension petal figures. To limit ourselves to

This research has been supported by NSF grant 0503447.

the area of bounded regions in the plane, we introduce a large square $\Omega = (-L, L)^2$, which will function as our "universe." When we speak of the complement of an open set $A \subset \Omega$, we mean the set $\Omega \setminus A$.

Definition 1. A petal is a convex region in Ω, whose boundary is formed by a finite set of line segments and arcs of circles. A petal figure is a finite union of petals and complements of petals.

The boundary of a petal has zero area, so it does not matter for our purposes whether it is included in the petal or not. We will disregard sets of measure zero.

We define the primitive petals to be two different types of sets, which we will call *caps* and *triangles* respectively:

1. Cap: the part interior to a circle, bounded by the circle and a line intersecting the circle.

2. Triangle (including the interior).

Let $\chi(A)$ be the characteristic function of a set A. All characteristic functions will be considered modulo the subspace of functions generated by characteristic functions of sets of measure zero.

Problem 2. *Given a petal figure A, express $\chi(A)$ as an integral linear combination of characteristic functions of primitive petals.*

The solution to this problem is a simple matter. Draw the secant to each circular arc forming part of the boundary of A, and take the corresponding caps C_i. Let $\sigma_i = 1$, if locally along the circular arc, A and C_i lie on the same side of the arc. Set $\sigma_i = -1$ otherwise. Working at the level of characteristic functions, we see that $\chi(A) - \sum_i \sigma_i \chi(C_i)$ is a linear combination of characteristic functions of polygons. By triangulating each polygon, we obtain triangles T_j. Then we obtain

$$\chi(A) = \sum_i \sigma_i \chi(C_i) + \sum_j \chi(T_j).$$

2.1 Petal Figures on a Sphere

We can repeat this construction, using a sphere instead of the plane. We take Ω to be the unit sphere, and we define a spherical petal figure to be a finite boolean combination (intersections, unions, and complements) of regions bounded by circles on the unit sphere.

We take the primitives to be the interior of geodesic (great circle) triangles and the two-sided regions bounded by an arc of a circle and an arc of a great circle.

Following the same procedure, we can rewrite the characteristic function of a spherical petal function as a linear combination of characteristic functions of primitives.

3 Primitive Volumes

Definition 3. The open ball $B(x, r)$ with center x and radius r is the set

$$\{y \in \mathbb{R}^3 \mid |x - y| < r.\}$$

Definition 4. The solid lune $L(x, v, w)$ is specified by a point $x \in \mathbb{R}^3$, and two unit vectors v and w. It consists of points y such that

$$(y - x) \cdot w > 0 \ \wedge \ (y - x) \cdot v > 0.$$

This is the intersection of two half-spaces.

Definition 5. The solid triangle $ST(x, r, v_1, v_2, v_3)$ is specified by a point $x \in \mathbb{R}^3$, a radius $r \geq 0$, and unit vectors v_1, v_2, v_3. It is the intersection of two lunes with a ball:

$$ST(x, r, v_1, v_2, v_3) = L(x, v_1, v_2) \cap L(x, v_2, v_3) \cap B(x, r).$$

The vector v_2 is repeated, so it has three bounding planar faces, rather than four.

Definition 6. The solid cap $SC(x, v, r, c)$ is specified by an apex $x \in \mathbb{R}^3$, a radius $r \geq 0$, a non-zero vector v giving direction, and constant c. The solid cap is the intersection of the ball $B(x, r)$ with a half-space:

$$\{y \in B(x, r) \mid (y - x) \cdot v > c\}.$$

Definition 7. The frustum $FR(x, v, h', h, a)$ is specified by an apex $x \in \mathbb{R}^3$, heights $0 \leq h' \leq h$, a unit vector v giving its direction, and $a \in [0, 1]$. The set FR is given as

$$\{y \mid (y - x) \cdot v > a|y - x| \ \ \wedge \ \ h' < (y - x) \cdot v < h\}.$$

By squaring the first inequality, we get the equation of a frustum as a quadratic constraint:

$$((y - x) \cdot v)^2 > a^2((y - x) \cdot (y - x)).$$

Definition 8. A tetrahedron $S(v_1, \ldots, v_4, c_1, \ldots, c_4)$ is the intersection of four open half-spaces (when the intersection is nonempty).

$$y \cdot v_i < c_i, \quad i = 1, 2, 3, 4.$$

These sets have all been defined by linear and quadratic constraints.

Definition 9. A primitive region is any of the following.

1. A solid triangle ST.

2. A tetrahedron S.

3. A wedge of a frustum; that is, the intersection of a lune with a frustum:

$$FR(x, u, h', h, a) \cap L(x, v, w),$$

where $h' < h$ and $u \cdot v = u \cdot w = 0$.

4. A wedge of a solid cap; that is, the intersection of a lune with a solid cap:

$$SC(x, r, u, h) \cap L(x, v, w), \text{ where } u \cdot v = u \cdot w = 0.$$

The solid triangle has one face that is spherical. The curvilinear edges of that face are arcs of great circles. A wedge of a solid cap also has one face that is spherical. The curvilinear edges of that face are circles. Two edges are great circles and the third edge is not necessarily a great circle. A wedge of a frustum has one face that is conical. Two of the edges of that face are line segments. The other edges are arcs of a circle (a conic section).

The volumes of these primitive regions are well-known. We will not repeat them here. These are elementary integrals. The volume formula for a solid triangle goes back over 400 years to T. Harriot.

4 Statement of Results

The main problem in rough terms is to determine whether a given quadratic region is equidecomposable with a finite disjoint sum of the primitive regions. This section gives a precise formulation of the problem.

If the coefficients of the linear and quadratic constraints used to define a primitive region are algebraic numbers, then we say that the region is definable. We restrict ourselves to definable sets, to avoid computations with arbitrary real numbers.

Let $\mathcal{A}$ be the collection of all sets A obtained by finite boolean combinations (intersection, union, and difference) of definable primitive regions. The operations of intersection, finite union, and difference carry bounded sets to bounded sets. We call the sets in $\mathcal{A}$ *quadratic regions*. Let $\mathcal{F}$ be the vector space of all rational linear combinations of characteristic functions of sets $A \in \mathcal{A}$ (considered modulo rational linear combinations of characteristic functions of sets of measure zero).

We define the equivalence relation $f_1 \sim f_2$ of $\mathcal{F}$ generated by $\chi(A) \sim \chi(B)$, if there is a rigid motion of $\mathbb{R}^3$ carrying A to B, with $A, B \in \mathcal{A}$. Let $\mathcal{F}_0$ be the vector space of all rational linear combinations of characteristic functions of definable primitive regions.

Problem 10.
- *Given $f \in \mathcal{F}$ (explicitly presented), determine whether it is equivalent to a function in $\mathcal{F}_0$.*
- *If $f \in \mathcal{F}$ is equivalent to a function in $\mathcal{F}_0$, find a representative of f in $\mathcal{F}_0$.*

Equivalent functions have the same (Lebesgue) integral. Thus, an affirmative answer to the problem for a function f leads to its integral in terms of volumes of primitive regions.

Theorem 11. *There is an algorithm that answers Problem 10.*

In particular, the algorithm finds a way, if it exists, to move non-primitive pieces by rigid motions so that they can be assembled into primitives. For instance, start with a single primitive region, cut it into two regions that separately cannot be expressed in terms of primitives, then move the pieces apart by rigid motions. The resulting characteristic function is not in $\mathcal{F}_0$, but it is equivalent to a function in $\mathcal{F}_0$. The algorithm finds the rigid motion that moves these pieces back together.

The method we use is similar to a familiar method of solving a two-dimensional jigsaw puzzle, by picking up a single piece at a time, and trying to match its edge with every other piece in turn, until a match is found. (If several pieces have an identical edge, extra bookkeeping is involved.)

I have made no attempt to find the most efficient possible algorithm. It is quite obvious that significant improvements are possible over what I have presented.

There are various more general problems along these lines that can be posed. There is no reason to restrict the primitive regions to the particular regions that are used here. The same problem can be posed with more general collections of primitive regions. I have not investigated these more general problems. It would be interesting to do so.

5 The Algorithm in Overview

Each $A \in \mathcal{A}$ is bounded by finitely many surfaces. Each surface is planar, spherical, or conical. Two surfaces intersect along a segment of a curve. Since the surfaces are quadrics, each curve has degree at most four. If the degree of the curve is one, it is a line. If the degree of the curve is two, then it is a conic section. In particular it is a planar curve. When the intersection of a cone and a sphere is reducible, the irreducible components are planar. When the intersection of two cones is reducible, the intersection consists either of planar curves or of a line and an irreducible nonplanar cubic. The algorithm proceeds by making a careful analysis of these curves and the local geometry near the curve.

The idea is to proceed in stages, starting with the most complicated types of curves, and using the equivalence relation to rewrite the function f in equivalent form with curves of lesser complexity. As we progress, the curves appearing in the representation of the function become progressively simpler, until finally, the curves are all lines and circles. At this point, we are able to recognize the primitive regions. The algorithm reports that f is not equivalent to a function in $\mathcal{F}_0$ if at any stage, it is impossible to eliminate the curves of a given type. The only way that the algorithm fails to find a representative of f in $\mathcal{F}_0$ is for no such representative to exist.

For each $f \in \mathcal{F}$, we work with a representation of f as a linear combination $f = \sum a_i \chi(A_i)$ of quadratic regions. We will call the elements A_i the constituents of f. As the algorithm progresses we may change the representation of f and hence its constituents.

As the algorithm progresses we will subtract off various known quantities from the function f, until finally it becomes the zero function. Thus, the function f does not denote a fixed element of $\mathcal{F}$, but rather a dynamic quantity that depends on the stage of the algorithm.

We will repeatedly use the identity:

$$\chi(A) = \chi(B) - \chi(B \setminus A) + \chi(A \setminus B).$$

The set A will be the region we are trying to decompose. The set B will be a set of known volume that matches A along a surface and a curvilinear edge of that surface. By matching B along the 'most complex feature' of the region A, the remainder $\chi(A \setminus B) - \chi(B \setminus A)$ will have lower complexity than A, and this will ensure that the procedure eventually terminates.

Although the expressions $\chi(A) - \chi(B)$ and $\chi(A \setminus B) - \chi(B \setminus A)$ are equal as functions, the data that represent them differ, and it matters in the algorithm which expression we use. In fact, it is best to retain the expression $\chi(A) - \chi(B)$ rather than rewrite it in terms of $A \setminus B$ and $B \setminus A$. Rewriting it would require an analysis of the intersection of the boundaries of A and B, and this would introduce further complications.

6 Capabilities

The algorithm will require us to construct regions with certain properties, and it will require us to decide the truth value of various statements. This section collects some of these auxiliary results upon which the algorithm rests.

6.1 Elementary Theory of the Reals

In the course of the algorithm, there are various statements in the elementary theory of the reals whose truth value we need to determine. Since the elementary theory of the reals is decidable, we can make unrestricted use of such in our algorithm. For an overview of methods of quantifier elimination with applications to geometry, see [1] and [7]. For example, we may have a point $x \in \mathbb{R}^3$ and a set $A \in \mathcal{A}$ and ask whether every sufficiently small neighborhood of x is contained in A. This statement can be expressed as a sentence in the first order theory of the reals. In our description of the algorithm, we will make use of such statements, sometimes without explicitly mentioning that they are elementary. When we use the term 'elementary' in this paper, it is in this technical sense.

6.2 Jumps across Surfaces

Let C be an irreducible component of the intersection of two surfaces:

$$C \subset \{x \mid f_1(x) = f_2(x) = 0\},$$

defined by polynomials f_1 and f_2, where the zero sets Z_i of f_i are planes, spheres, or cones.

Let x be a nonsingular point on C that does not lie on any other irreducible component of the zero set of (f_1, f_2). If the two surfaces are smooth at x and if their tangent planes to the surfaces at x are not equal, then by the implicit function theorem, in sufficiently small balls around x, the topology of the triple (x, C, Z_i) is that of a point, diameter, and plane through the diameter.

If C is nonplanar, these conditions are satisfied for all but finitely many points of C. In fact, C meets another component of $Z_1 \cap Z_2$ with at most finite multiplicity. The number of singular points on C is finite. The set of points x for which the tangent planes are equal is clearly Zariski closed. Hence if the tangent planes are equal at infinitely many x, they are equal for all $x \in C$. Now if the two surfaces are cones, the tangent planes to the curve at three different points, none on the same ray through the apex, determine the apex, and hence determine the cone. Thus, the intersection of two distinct cones cannot give the same collection of tangent planes along the curve. If the two surfaces are a cone and a sphere, the segments from the apex to the curve are tangents to the sphere. Hence they have the same length. This forces the curve of intersection to be a circle, contrary to the assumption that C is nonplanar. So indeed these conditions are satisfied for all but finitely many points of C.

In particular, locally at x, C separates Z_1 into two connected components

$$E(f_1, \pm f_2) = \{x \mid f_1(x) = 0,\ \pm f_2(x) > 0\}.$$

Let A be a quadratic region. For every f_1, f_2, consider the elementary statement:

$P(f_1, f_2, C, x, A)$: *For every sufficiently small open ball* $B(x, \epsilon)$ *around* x *and for every*

$$y \in B(x, \epsilon) \cap E(f_1, f_2),$$

there is a nonempty open ball $B(y, \delta) \subset B(x, \epsilon)$ such that

$$B(y, \delta) \cap \{z \mid f_1(z) > 0\} \subset A.$$

Let $\psi(f_1, f_2, C, x, A) = 1$ if $P(f_1, f_2, C, x, A)$ is true and 0 otherwise. If $f = \sum_i a_i \chi(A_i)$, we have a corresponding value

$$\begin{aligned} J(f, f_1, f_2, C, x) \quad &= \textstyle\sum_i a_i(\psi(f_1, f_2, C, x, A_i) - \\ &\qquad \psi(-f_1, f_2, C, x, A_i)). \end{aligned}$$

This function measures the jump in the value of the function f across the zero set of f_1, on the side $E(f_1, f_2)$. It is elementary to compute it for a given $x \in C$.

If every intersection of $B(x, \epsilon) \cap \{y \mid f_1(y) = 0\}$ with a surface $f_2 = 0$ defining A is contained in C, then except for finitely many $x \in C$, the element y in the statement $P(f_1, f_2, C, x, A)$ does not lie on any surfaces defining A except $f_1 = 0$. (We will only use the jump function when this condition is met.) Thus, the set $B(y, \delta) \cap \{z \mid f_1(z) > 0\}$ for δ sufficiently small is either disjoint from A or is entirely contained in A.

Under these same conditions, the jump $J(f, f_1, f_2, C, x)$ is a locally constant function of x on $C \setminus X$, where X is a computable finite set of exceptional points. The set $C \setminus X$ is a finite number of topological intervals. Thus, $J(f, f_1, f_2, C, x)$ can be considered a function on a finite domain, computed by picking a test point x on each interval. (We have more to say in Section 6.3 about how to specify the topological intervals.)

We say that C is a *boundary curve* of $f \in \mathcal{F}$ if there is a jump

$$J(f, f_1, f_2, C, x) \neq J(f, f_1, -f_2, C, x),$$

for all x in some interval of C for some (f_1, f_2). We also say that C is a boundary curve of $A \in \mathcal{A}$, if it is a boundary curve of $\chi(A) \in \mathcal{F}$. We say that $f_1 = 0$ is a *boundary surface* of $A \in \mathcal{A}$ if

$$J(\chi(A), f_1, \pm f_2, C, x) \neq 0.$$

for some choice of sign $\pm$ (again for x in some interval of C), and some f_2 such that C is an irreducible component of the zero set of (f_1, f_2). We say that the jump through $f_1 = 0$ is coherent across C (at x) if

$$J(f, f_1, f_2, C, x) = J(f, f_1, -f_2, C, x),$$

for every f_2 such that C is an irreducible component of the zero set of (f_1, f_2).

6.3 Intervals on a Curve

There are several arguments that make use of a simple arc of an algebraic curve on a quadric surface. Since we do not have parameterizations of the curves, we use the surface to simplify matters.

Suppose that C is a nonplanar curve on a cone S, given as an irreducible component of the intersection of two cones. The lines through the apex and the circles form isothermal coordinates on S that can be used to partition the cone into small 'rectangles.'

We claim that the rectangles can be chosen so that the intersection of C with each rectangle is a simple arc (except at singular points of C and the apex of the cone). Simply pick lines on the rectangular grid that include every tangent line to C that lies on the cone, and circles that include every circle on S tangent to C. On each such rectangle, each branch of C is a continuous function of the isothermal coordinates, and is therefore homeomorphic to an interval. By restricting the size of the rectangles further, we may assume that there is a single branch of C in each rectangle. (Checking that C is a univalent function on a rectangle is an elementary test.)

A similar procedure works for a curve C on a sphere S. We may pick two pairs of antipodal points (that avoid C), and then take a collection of great circles that pass through one pair or the other. This gives a system of 'rectangles' that can be used in a similar manner to break C into a finite set of intervals. We will eventually relate these rectangles to primitive regions. With that in mind, we draw a diagonal to each rectangle, breaking it into two spherical triangles.

The possible singularities of C are mild in the case of the intersection of a cone with another cone or sphere. It can be verified that if there are two branches of the curve C meeting at a singular point x on the curve, then it is always possible to separate the branches in a neighborhood of x by a plane passing through x. This is true even if x is an apex of a cone. We can choose the rectangles so that the singular points of C always lie on an edge of a rectangle. By cutting the rectangles by finitely many planes, we arrive at a situation where we separate all the branches of the curve C. We refer to these rectangles, possibly cut into smaller pieces by branch-separating planes as *charts* for C (with respect to a sphere or cone S). When the surface is a sphere, by triangulating if necessary, we may assume that each chart is a spherical triangle.

We can choose the charts sufficiently small, so that on the intervals I they define, the jump function J is constant on each interval. We write $J(f, f_1, f_2, C, I)$ for the common value of $J(f, f_1, f_2, C, x)$, for $x \in I \subset C$.

6.4 Automorphisms of Curves

A bijective rigid motion $T : A \to B$ between subsets $A, B \subset \mathbb{R}^3$ is a *congruence*. A congruence of a set A with itself is an *automorphism* of that set. A nonplanar irreducible algebraic curve in $\mathbb{R}^3$ has only finitely many automorphisms.

Given an irreducible nonplanar component C of the intersection of a cone with a sphere or another cone, we can determine the finite group of rigid motions that map the curve to itself. This is elementary. A rigid motion T is determined by the finitely many coordinates of its underlying affine transformation

$$T(x) = Ax + b.$$

The statement *'T is automorphism of C'* can be expressed in the elementary language of the real numbers, with free parameters A and b. By eliminating quantifiers from that statement, we obtain explicit equations for the possible automorphisms T.

For each automorphism T, we can solve $T(x) = x$ for the set of fixed points, and then intersect the set of fixed points with C to determine the finite set of fixed points of T on C. By passing to smaller charts in Section 6.3, we can assume that each point of C that is fixed by some automorphism is an endpoint of an interval, so that no fixed points appear on an open-ended interval.

We may repeatedly bisect the charts (and the intervals of the curve C they contain) until each (open-ended) interval I is so small that the only automorphism T such that $T(I)$ meets I is the identity automorphism.

6.5 Automorphisms and Boundary Edges

If $f = \sum a_i\chi(A_i) \in \mathcal{F}$, we can form a list of all of its boundary curves $C_1, \ldots, C_k$. By applying quantifier elimination on each of the statements *'T is a rigid motion carrying C_i bijectively to C_j'*, we can explicitly determine all of the finitely many congruences between the boundary curves of f. We can subdivide each boundary curve C_i into small open-ended intervals I_{ir} (that cover C_i except for the finitely many endpoints), as explained in Section 6.4, so that $T(I_{ir})$ does not meet I_{ir}, whenever T is a non-trivial automorphism of C_i.

Fix one boundary curve $C = C_0$, and consider all congruences from the other curves C_i to C (including C_0 itself). Mark all endpoints of the intervals $T(I_{ir}) \subset C$, as we run over all congruences T for all i. Use these finitely many points so obtained, to further refine the intervals I_{0r} into smaller intervals $I'_{0r'}$. If some $I'_{0r'}$ meets some $T(I_{ir})$, then $I'_{0r'}$ is contained in $T(I_{ir})$.

By construction, if $T' : C \to C_i$ is a congruence, then $T'(I'_{0r'})$ is contained in a single interval I_{ir} of C_i. The intervals $I'_{ir'} = T'(I'_{0r'})$ then give a refinement of the intervals I_{ir} that is independent of the choice of $T' : C \to C_i$. In this way, we obtain a collection of intervals $I'_{ir'}$ on each curve C_i (that cover except for finitely many endpoints) such that every congruence carries intervals bijectively to intervals.

6.6 The Edge Coherence Condition

Let C be an irreducible curve that has only a finite number of automorphisms.

Let $f \in \mathcal{F}$. Let $C_1, \ldots, C_k$ be the boundary curves of f that are congruent to C. Let $\mathcal{T}$ be the set of all congruences $T : C_i \to C$, as i ranges from 1 to k.

The group of motions of $\mathbb{R}^3$ acts on $\mathcal{F}$ by

$$(T_* f)(x) = f(T^{-1}x).$$

With this action, $T_*\chi(A) = \chi(TA)$. We define $D_C(f) = \sum_{T\in\mathcal{T}} T_* f \in \mathcal{F}$. This is well-defined in the sense that it does not depend on the expression for f as a sum of characteristic functions. Note that D_C depends on f through the list $C_1, \ldots, C_k$ of boundary curves, so it is not a linear operator on $\mathcal{F}$.

If $f = \sum_i a_i\chi(A_i)$. Then

$$D_C(f) = \sum_{T\in\mathcal{T}} \sum_i a_i\chi(TA_i).$$

Partition each C_i into intervals I'_{ir} as described in Section 6.5. Define intervals I_r on C by $I_r = T(I'_{ir})$ for any congruence $T : C_i \to C$.

We have the following necessary condition for equidecomposability. If the following condition is not met, the function f is not equivalent to a function in $\mathcal{F}_0$, and the algorithm terminates.

(Edge Coherence Condition) For every nonplanar irreducible curve C, every interval I_r of C as constructed above, and every nonzero quadratic function f_1 whose zero set contains C, the function $D_C(f)$ has jumps through $f_1 = 0$ that are coherent across C along I_r.

This is stated as a condition on infinitely many curves and quadratic functions. However, it reduces to a finite calculation. We can restrict to curves C that are congruent to an edge curve of f, because otherwise the condition always holds. We can restrict further, to curves C that equal one of the edge curves $C_1, \ldots, C_k$. We can restrict the condition further to quadratic functions f_1 that have the form $T_* f'_1$, where f'_1 defines a boundary surface of some constituent A_i of f, the zero set of f'_1 contains some C_j, and $T : C_j \to C$ is a congruence. This reduces the edge coherence condition to a finite condition.

We will omit the proof that any function equivalent to a function in $\mathcal{F}_0$ satisfies the edge coherence condition. There are two main ingredients to the proof. The first ingredient is to check that the edge coherence condition holds with respect to $D_C(f)$ if and only if it holds for a function $D'_C(f)$, where D'_C is defined exactly as D_C, but with respect to a larger set of curves $\{C_1, \ldots, C_k, C_{k+1}, \ldots, C_\ell\}$, where the additional curves are congruent to C, but otherwise arbitrary. The second ingredient is to check that if $A \in \mathcal{A}$ and T' is any rigid motion, the function $f' = f + a(\chi(A) - \chi(T'A))$ satisfies the edge coherence condition if and only if f does. The proof of this second ingredient is based on the first ingredient.

We note that the edge coherence condition is a Dehn invariant type condition. With the original Dehn invariant, all the edges are line segments, which are all locally congruent. This helps to explains why the Dehn invariant, which is a single invariant for polyhedra, must be expanded to a collection of conditions for general quadratic regions. We have a separate condition for each congruence type of curve.

7 Remove Nonplanar Intersections

We are ready to give details of the algorithm. The first goal is to adjust f by a known quantity so that the jumps are coherent across all nonplanar curves. At this stage of the algorithm, the edge coherence condition has been tested, and it is assumed to be valid.

We show how to make the jumps of f coherent across irreducible nonplanar components of an intersection of a cone with another cone or with a sphere.

Let $f = \sum a_i\chi(A_i)$, and let C be an irreducible nonplanar curve for which the edge coherence condition holds. We assume that it is congruent to some boundary curve C_j of f. Fix an irreducible quadratic function f_2 whose zero set contains C. Partition C into interval I_r as was done in the construction of $D_C(f)$. Automorphisms of C permute the intervals I_r. Let $\mathcal{I}$ be a minimal set of intervals in the sense that every interval I_r is congruent to exactly one element of $\mathcal{I}$.

Let $A = A_i$ be a constituent of f that has C_j as a boundary curve. Let f_1 be an irreducible quadratic function whose zero set contains C_j and such that the jumps through $f_1 = 0$ are not coherent across C_j along some interval I'' of C_j. There is a unique $I \in \mathcal{I}$ and unique congruence $T : C \to C_j$ such that $I'' = T(I)$.

First we handle the cases when T_*f_2 and f_1 are not proportional functions.

Assume also that $f_1 = 0$ defines a cone. Recall that each interval $I'' = T(I)$ was defined on some small chart, which is a rectangle on the cone, possibly cut into smaller pieces by planes. Each rectangle on the surface of the cone uniquely determines a wedge of a frustum. The planes cutting the rectangle into a chart cut the frustum into smaller pieces. One of those pieces F corresponds to the interval $T(I)$. The function T_*f_2 cuts F into two pieces

$$F_\pm = \{x \in F \mid \pm(T_*f_2)(x) > 0\}.$$

The fresh boundary from this cut meets the chart along $T(I)$.

When $f_1 = 0$ defines a sphere, the argument is almost the same. In this case the charts lie on the surface of a sphere. The chart is a spherical triangle, which is the boundary of a uniquely determined solid triangle F. The function T_*f_2 cuts F into two pieces. In both cases (both cone and sphere), the pieces $F_\pm$ belong to $\mathcal{A}$. The only nonplanar edge on these pieces is the interval I'' on C_j.

There are unique constants $b_\pm$ such that

$$g = a_i\chi(A_i) + b_+\chi(F_+) + b_-\chi(F_-)$$

has

$$J(g, f_1, \pm T_*f_2, C_j, I'') = 0.$$

In particular, with this choice of constants, the jumps are coherent through $f_1 = 0$ across I''. We replace f with the function

$$f + b_+(\chi(F_+) - \chi(T^{-1}F_+)) + b_-(\chi(F_-) - \chi(T^{-1}F_-)).$$

It has several important properties. It is equivalent to f. It satisfies the edge coherence condition if and only if the function f does. The incoherent jump across C_j along $T(I)$ has been translated by the rigid motion T to an incoherent jump across C along I.

Call this equivalent function f. Repeat this procedure until we have moved all incoherent jumps to intervals $I \in \mathcal{I}$, except possibly when f_1 is proportional to T_*f_2. If we take a small loop around an interval I'', we get a jump in value in the function f each time we cross a boundary surface. As we make a full loop, we return to the original value of the function f. Thus, the sum of the jumps as we complete a loop is zero. Once all the jumps around C_j are zero, except those along $T_*f_2 = 0$, then the zero sum condition forces the jumps along $T_*f_2 = 0$ to be coherent. Thus, coherence of this final surface is automatic, and we find that the only incoherent jumps are confined to interval $I \in \mathcal{I}$.

By construction, there are no congruences between different intervals in $\mathcal{I}$. We have 'used up' all the congruences and automorphisms. This implies that the edge coherence condition for $D_C(f)$ yields that the jumps of f are coherent through every f_1 across every interval I''.

This completes this stage of the algorithm: we have replaced f with an equivalent function that has the property that all jumps are coherent through irreducible nonplanar curves.

8 Planar Intersections

At this point in the algorithm, every jump that is not coherent is across a planar curve. The final steps are to eliminate spherical boundaries and to eliminate conical boundaries. What remains will be a polyhedron, which can be triangulated into tetrahedra, which are primitive regions.

8.1 Spherical Surfaces

In this step we assume that all boundary curves are planar. For quadratic regions, this implies that the curves are lines or conic sections.

We work through the spherical surfaces in groups according to the radius of the sphere, starting with the largest radius. In this step of the algorithm there are no necessary conditions for equidecomposability. The spherical surfaces can always be eliminated.

When one of the boundary surfaces is a sphere, the intersections are always circles. Thus, the spherical part of the boundary of a quadratic region forms a spherical petal figure. We have seen in Section 2.1 how to decompose a spherical petal figure into spherical caps and spherical triangles. Corresponding to this decomposition of the spherical petal figure is a decomposition of the solid ball into wedges of solid caps and solid triangles. These are primitives. Thus, we can always eliminate a given spherical surface.

The boundary of the wedge of a solid cap consists of three planar surfaces and a spherical surface. The boundary of the solid triangle also consists of three planar surfaces and a spherical surface. In this process, we may introduce new jumps along lines C given by the intersection of two planes. These will be handled at a later stage of the algorithm.

8.2 Conic Sections

At this point of the algorithm we assume that every jump occurs along is bounded by planar and conical surfaces, and that every boundary curve is planar (a line or conic section).

The next step is to produce coherence of jumps along cones across conic sections (other than lines and circles). We will deal with lines and circles later. This part of the algorithm is similar to the elimination of nonplanar curves.

We eliminate boundary curves that are not lines and circles. We have a necessary condition that must be satisfied. If this condition is not satisfied, then equidecomposability fails, and the algorithm terminates.

(Conic Section Coherence Condition) For every conic section C, other than circles and lines, every interval I_r of C constructed as above, and every nonzero quadratic function f_1 defining a cone that contains C, the function $D_C(f)$ has jumps through $f_1 = 0$ that are coherent across C along I_r.

The procedure is essentially identical to the process of eliminating nonplanar curves on cones. We fix a conic section C that satisfies the conic section coherence condition. We let F be a suitable frustum, which we cut into two pieces $F_\pm$ by a plane that meets the conic boundary along an interval $I'' \subset C_j$, for some conic section C_j that is congruent to C. The pieces $F_\pm$ lie in $\mathcal{A}$. The edges of the pieces $F_\pm$ consist of $I'' \subset C_j$, and linear and circular segments. By means of a congruence $T : C \to C_j$, we transport the jumps so that they occur along an interval of C, rather than an interval of C_j. Once all the jumps lie along intervals of C, and once we have 'used up' the congruences and automorphisms, the conic section coherence condition for conic section implies coherence of jumps.

8.3 Circles

At this stage of the algorithm we assume that all boundary curves are lines and circles. These lines and circles form isothermal coordinates on each cone. Using these lines and circles, we break the surface into curvilinear rectangles on the cone. Each such rectangle is the conic surface of a uniquely defined wedge of a frustum, which is a primitive region. Subtracting off these frustums, we obtain a region in which the jump across each conical surface is zero. In other words, they no longer form part of the boundary of the quadratic regions. Thus, after subtracting the frustums, we are left with a polyhedron.

8.4 Polyhedra

As we mentioned above, once all spherical and conical surfaces have been eliminated, we are left with a polyhedron. This can always be triangulated into tetrahedra, which are primitives. Thus, the algorithm is complete.

9 Example

In this section we show an explicit example of a volume computed by this algorithm. The region we consider occurs many times in the proof of the Kepler conjecture [3]. It is called a *quoin*.

Let a, c, t be constants with $a < c$ and $0 < t$. We define the following quadratic region $Q(a, c, t)$:

$$\{(x, y, z) \mid z > 0,\ a < y < tx,\ x^2 + y^2 + z^2 < c^2\}.$$

Note that if $Q(a, c, t)$ is nonempty, we have

$$(a/t)^2 + a^2 + 0^2 < x^2 + y^2 + z^2 < c^2.$$

This condition implies that $t = a/\sqrt{b^2 - a^2}$ with $a < b < c$. We assume that this condition holds for some b. The volume $q(a, c, t)$ is then given explicitly as follows:

$$\begin{aligned} 6\,q(a,c,t) \quad &= (a+2c)(c-a)^2 \arctan(e) \\ &\quad +a(b^2-a^2)e \\ &\quad -4c^3 \arctan(e(b-a)/(b+c)), \end{aligned} \tag{1}$$

where $e \geq 0$ is given by $e^2(b^2 - a^2) = (c^2 - b^2)$.

This formula is obtained by applying the algorithm to the given quadratic region. All the boundary curves are planar. In fact, every curve is a circular arc or a line segment. No surfaces are cones. Thus, the volume is computed by a particularly simple application of the algorithm.

We see that the intersection of $Q(a, b, t)$ with the sphere of radius c is a spherical petal figure. After subtracting off the contribution from the petal figure, we are left with a tetrahedron. This leads to the given formula for volume.

10 Applications to Flyspeck

In 1998, Sam Ferguson and I gave a proof of the Kepler Conjecture, which asserts that no packing of congruent balls has density greater than the face-centered cubic packing. The proof relies on a significant number of computer calculations.

The refereeing process took several years. As a result of the difficulties in checking the correctness of the Kepler conjecture, I have become interested in formal theorem proving, as a way of checking complex computer proofs. In 2003, I announced a project called Flyspeck, whose purpose is to give a completely formal proof of the Kepler Conjecture [4]. The name '**F**lys**p**ec**k**' comes as an expansion of the acronym 'F*P*K', which stands for the '**F**ormal **P**roof of **K**epler.'

In addition to the computer part of the proof, the proof of the Kepler Conjecture involves nearly 300 pages of traditional mathematical arguments. The Flyspeck project aims to formalize the traditional mathematical portions of the proof as well. The non-computer parts of the project are not nearly so far along. In my view, a major impediment to completing this part of this project is a lack of modularity in the design of the original proof.

A significant part of these 300 pages consists of volume calculations of quadratic regions. In the original proof, these were all obtained by hand, using a variety of techniques. Every one of the volume calculations falls within the scope of the algorithm of this paper. (They can all be expressed as a linear combination of primitives.) As a result of this paper, these calculations can be entirely automated.

References

[1] S. Basu, R. Pollack, M.-F. Roy, *Algorithms in real algebraic geometry,* Second edition, Algorithms and Computation in Mathematics, 10, Springer-Verlag, Berlin, 2006.

[2] E. Berberich, Exact Arrangements of Quadric Intersection Curves, Master's thesis, Saarbrücken, 2004.

[3] S. Ferguson and T. Hales, The Kepler Conjecture, Disc. and Comp. Geom. (36), no. 1, 2006, 1–269.

[4] T. Hales, The Flyspeck Fact Sheet, 2003; revised 2007. http://www.math.pitt.edu/˜ thales/

[5] F. Higbee, *The Essentials of Descriptive Geometry*, Second Edition, John Wiley and Sons, New York, 1917.

[6] M. Laczkovich, Equidecomposability and discrepancy; a solution of Tarski's circle-squaring problem, J. Reine Angew. Math. (404) 1990, 77-117.

[7] B. Mishra, Computational Real Algebraic Geometry, *Handbook of Discrete and Computational Geometry*, ed. J. E. Goodman, J. O'Rourke, 743–764, CRC Press, Boca Raton FL, 1997.