7. Algorithmic aspects

Objekttyp: **Chapter**

Zeitschrift: **L'Enseignement Mathématique**

Band (Jahr): **40 (1994)**

Heft 1-2: **L'ENSEIGNEMENT MATHÉMATIQUE**

PDF erstellt am: **30.06.2024**

that we can reduce our search to a finite number of vertices (P, p) of Γ_∞. We focus attention on one of these cases. We start to explore $\Gamma_{P,p}$. If we find more than b_0 vertices, we know Z is not complete. Otherwise we find a generating set of circuits in $\Gamma_{P,p}$ and check for each of these that a consistent horosphere can be constructed. □

7. Algorithmic aspects

We will now look more closely at the algorithmic aspects of Poincaré's Theorem. We wish to produce a mechanical procedure which takes as input a finite number of finite-sided convex polyhedra in $\mathbf{E}^n$ or $\mathbf{S}^n$ or $\mathbf{H}^n$, together with a finite set of face-pairings, and which outputs "Yes" or "No" to the question of whether these polyhedra and face-pairings give a tessellation of the appropriate space. In the case that the answer is "Yes", it also outputs a presentation for the group of symmetries of this tessellation with the given finite union of finite polyhedra as a fundamental domain.

What kind of mathematical model of a computing machine is necessary in order to carry out the procedure described in the preceding pages? It is not appropriate to use a Turing machine model. A Turing machine is not capable of taking as input a list of real numbers and coming out with the answer "Yes" or "No". We need to be able to handle real numbers not as sequences of bits but as entities. We need to be able to compare two real numbers for equality or inequality in a one-step operation, and likewise for addition and multiplication and division of real numbers.

Such a mathematical model has been described in [BSS89]. Their model is devoted to the study of polynomial and rational maps, and it is assumed that computation of a polynomial can be carried out in a single step. In most computations in hyperbolic or spherical geometry, trigonometric and hyperbolic trigonometric functions are likely to arise, and so it seems at first sight that a model of computation able to carry out only polynomial operations would not be relevant. However, in the case of Poincaré's Theorem it happens that the computation can be expressed in polynomial terms. Since the BSS scheme has been thought out and developed far enough to be a reasonable tool, we use it.

However, for more general computations in geometry, it seems that it would be more satisfactory to have a computational model with a library of functions, satisfying certain axioms. It might, for example, be assumed that any of the functions in the library could be computed with complete accuracy in a single step. We put this forward in the hope of encouraging someone to develop such an approach.

Let us go through the steps of the computation to see what kind of operations are necessary. We need to start by making a decision as to how to represent the input data. We first need to decide how to represent $\mathbf{E}^n$, $\mathbf{S}^n$ and $\mathbf{H}^n$. It is convenient in each case to embed the space in $\mathbf{R}^{n+1}$. In order to be able to change basis easily, we will describe the situation in a general manner.

Suppose we are given a positive definite symmetric $(n+1) \times (n+1)$ real matrix M_S defining a positive definite inner product on $\mathbf{R}^{n+1}$. We define $\mathbf{S}^n$ to be the set of vectors v of unit length with respect to this inner product. We will frequently represent a point in $\mathbf{S}^n$ by a non-zero vector which does not have unit length; conceptually this can be normalized, but computationally we will not normalize. The reason for avoiding normalization is that BSS machines are capable of polynomial operations, but not of taking square roots.

We take $\mathbf{E}^n$ to be an affine subspace of $\mathbf{R}^{n+1}$ which does not contain the origin. We can think of the subspace as specified by a non-zero linear map $\lambda : \mathbf{R}^{n+1} \to \mathbf{R}$ as follows: $\mathbf{E}^n = \{v : \lambda(v) = 1\}$. As in the case of $\mathbf{S}^n$, we assume that $\mathbf{R}^{n+1}$ has a positive definite inner product, given by a matrix M_E. We will often represent a point in $\mathbf{E}^n$ by an element $v \in \mathbf{R}^{n+1}$, such that $\lambda(v) > 0$, without supposing $\lambda(v) = 1$. Multiplying by a positive scalar, we find a vector in $\mathbf{E}^n$.

If we are given a real $(n+1) \times (n+1)$ symmetric matrix M_H with n positive eigenvalues and one negative eigenvalue, we obtain a non-degenerate indefinite inner product on $\mathbf{R}^{n+1}$ of type $(n, 1)$. We define $\mathbf{H}^n$ to be one sheet of the hyperboloid $\{v \in \mathbf{R}^{n+1} : \langle v, v \rangle = -1\}$. We specify such a sheet by fixing a linear map $\lambda : \mathbf{R}^{n+1} \to \mathbf{R}$, such that the sheet lies in the half-space $\lambda > 0$. A vector v such that $\langle v, v \rangle < 0$ and $\lambda(v) > 0$ represents a well-defined point of $\mathbf{H}^n$, obtained by multipling by a suitable positive scalar. However this scalar cannot be computed by our BSS machine, since the computation involves taking a square root.

If $\mathbf{X}^n$ is any of the three spaces, we specify a codimension-one $\mathbf{X}$-subspace by means of a single linear equation, and a general $\mathbf{X}$-subspace by means of a finite number of linear equations (with no constant term). The condition on the subspace in the hyperbolic case is that the coefficient vectors of the linear equations define a positive definite subspace with respect to M_H. In the euclidean case the condition is that the coefficient vector of λ, the linear map defining $\mathbf{E}^n$, is not linearly dependent on the set of coefficient vectors of the linear inequalities. In the spherical case there are no conditions. Such a subspace is therefore determined by a finite list of real numbers.

Face-pairings are represented by matrices. A half-space is determined by a linear inequality (with no constant term).

Each finite collection of linear equalities and inequalities (satisfying appropriate conditions to give a codimension-one subspace or a codimension-zero half-space) defines either the nullset or some i-dimensional convex polyhedron in $\mathbf{X}^n$. If there are exactly $n - i$ linearly independent equalities and if each of the half-spaces is essential, the defining collection of equalities and inequalities is minimal. If $i = n$, there is a unique minimal set of defining half-spaces (see Proposition 2.5). If $i < n$, the number of inequalities in a minimal collection is equal to the number of codimension-one faces, but neither the equalities nor the inequalities are uniquely determined. Any collection of equalities and inequalities defining the i-dimensional polyhedron can be transformed into a minimal collection by changing some of the inequalities to equalities and then omitting some of the equalities and inequalities. (For example the two conditions $\mu \geqslant 0$ and $\mu \leqslant 0$ are equivalent to the one condition $\mu = 0$.)

THEOREM 7.1 (BSS polyhedron computation). *There is a BSS program which carries out the following computation. We input* $\mathbf{E}^n$, $\mathbf{H}^n$ *or* $\mathbf{S}^n$, *represented by a real non-singular symmetric* $(n+1) \times (n+1)$ *matrix* M_E, M_H *or* M_S, *and a linear map* $\lambda : \mathbf{R}^{n+1} \to \mathbf{R}$. *We also input a finite set of linear equalities and inequalities defining codimension-one subspaces and codimension-zero half-spaces in* $\mathbf{E}^n$, $\mathbf{H}^n$ *or* $\mathbf{S}^n$ *respectively. The output from the program is the dimension* i *of the convex polyhedron defined by the intersection of these subspaces and half-spaces, the combinatorial structure of its faces, for each face a minimal subset of the defining equalities and inequalities (with some of the defining inequalites converted to equalities). The program also outputs for each face* F *an element* $x_F \in \mathbf{R}^{n+1}$ *representing a point in the relative interior of the face.*

Proof of 7.1. If $n = 0$ or $n = 1$, the result is clearly true. Inductively we assume the result is known for dimensions less than n.

If the collection of equalities and inequalities input includes one or more equality, then the result follows by induction on n. To see this, we transform by a matrix which changes one of the equalities to $x_{n+1} = 0$. This changes the matrix of the inner product and the coordinates of λ. In the hyperbolic case we next check that $(0, \ldots, 0, 1)$ is a positive vector (otherwise the plane $x_{n+1} = 0$ is not a plane in hyperbolic space). In the euclidean case, we check that λ does not have the form $cx_{n+1} = 0$ in the new coordinates. (If these checks fail, then the input data was inconsistent). We then apply the BSS program which has been inductively constructed. So we may asume that our input consists of inequalities only.

We now assume that we have j inequalities, and that we have constructed a program which gives the required output for any collection of $j-1$ inequalities satisfying the induction assumptions. Let $P \subset \mathbf{X}^n$ be the convex polyhedron defined by the first $j-1$ inequalities. Let $f \geqslant 0$ be the j-th inequality. We first construct a point x_f representing a point $y_f \in \mathbf{X}^n$, such that $f(x_f) = f(y_f) = 0$.

One of the following situations must hold, and we want to construct a BSS program to find which.

CONDITION 7.2 (situation for (P, f)).

(a) $P \subset \{f = 0\}$. In the other cases, we assume that P is not a subset of $\{f = 0\}$.

(b) $P \subset \{f > 0\}$.

(c) $P \subset \{f \geqslant 0\}$ and the codimension-one subspace $f = 0$ meets P.

(d) P meets $f > 0$ and $f < 0$.

(e) $P \subset \{f \leqslant 0\}$ and the codimension-one subspace $f = 0$ meets P.

(f) $P \subset \{f < 0\}$.

Assuming we know which case we are in, the inductive proof deals with all cases except Condition 7.2(d), when we need also to compute the new face structure and to find a representative for a point in the relative interior of each new face.

We proceed as follows, assuming that we are in case Condition 7.2(d). If P has no faces, then $P = \mathbf{X}^n$. The new polyhedron has two cells, namely $f \geqslant 0$ and $f = 0$. It is easy to find representatives for points in these two faces. (Solving linear equations can be done by row operations.)

If P does have faces, we first tackle the same problem for each proper face. Let F be a face and let S be the smallest $\mathbf{X}$-subspace containing F. If the plane $f = 0$ meets S, then either $f = 0$ contains S, which we can check by a linear independence computation (row operations), or $f = 0$ meets S in a codimension-one subspace of S. In both cases we can treat the problem by induction on n. If the plane $f = 0$ does not meet S, then take the point $x_F \in \operatorname{Int}(F)$ given by our induction, and evaluate $f(x_F)$. The value is either negative, in which case we must be in case 7.2(f) for the pair (F, f), or it is positive, in which case we must be in case 7.2(b) for (F, f).

The construction of the point in the relative interior of $\{f \geqslant 0\} \cap P$ in case 7.2(d) is as follows. By induction we find a point in the relative interior of $\{f = 0\} \cap P$. This means that all the equalities required for the definition of P are satisfied for this point, and all the inequalities required are satisfied as strict inequalities. We can therefore move the point a little so that the equalities and the strict inequalities continue to hold. In addition we move it in a direction away from $\{f = 0\}$ so as to increase f.

Now let us see how to recognize which case we have for (P, f). Using our minimal set of equalities and inequalities for P, a linear independence check (row operations) tells us whether or not we are in case 7.2(a).

We are in case 7.2(b) for (P, f), if firstly for each proper face F of P, (F, f) is in case 7.2(b), which shows that $f = 0$ does not meet ∂P, and secondly we check that that $x_f \notin P$. Case 7.2(f) is treated in the same way.

We are in case 7.2(c) if the following three conditions are satisfied: firstly for each proper face F of P we have case 7.2(a) or case 7.2(b) or case 7.2(c) for (F, f), secondly we are in case 7.2(a) for some face F, and thirdly $f(x_P) > 0$ for the point x_P already constructed in the relative interior of P. Similarly for the case 7.2(e), except that the signs are changed.

To see if we are in case 7.2(d), we change coordinates so that $f = 0$ becomes the plane $x_{n+1} = 0$, and then look at the intersection P' of this plane with P. We take a point in the relative interior of P' and check whether it is in the relative interior of P. $\square$

To complete the discussion of the algorithmic approach, suppose we are given a finite set $\mathscr{P}$ of finite-sided polyhedra and maps $R : \mathscr{F}(\mathscr{P}) \to \mathscr{F}(\mathscr{P})$ and $A : \mathscr{F}(\mathscr{P}) \to \mathrm{Isom}(\mathbf{X}^n)$, where $\mathscr{F}(\mathscr{P})$ is the set of codimension-one faces of the polyhedra in $\mathscr{P}$. There is obviously no problem in checking Pairing$(\mathscr{P}, R, A)$, Finite$(\mathscr{P})$ and Connected$(\mathscr{P}, R)$. To check Cyclic$(\mathscr{P}, R, A)$, we need to be more explicit about the form in which the face-pairings are given. We will assume each face-pairing is given by an $(n+1) \times (n+1)$ matrix of real numbers which preserves the appropriate structure. Then we can check Cyclic$(\mathscr{P}, R, A)$ by multiplying such matrices together. The fact that a certain product is the identity on a codimension-two face can be checked by a linear independence calculation, applied to the coefficient vectors of the planes defining the $\mathbf{X}$-subspace spanned by the face. The fact that the angle of rotation has the form $2\pi/m$ can be checked by seeing whether the m-th power of a certain group element is the identity. We can see approximately which values of m to use by means of floating point arithmetic.

Finally we indicate circumstances under which it seems that a Turing machine could do all the relevant checks. Suppose we are given a finite set of matrices each of which is an isometry of $\mathbf{X}^n$, and such that each entry is an algebraic number. We can hold the algebraic numbers in the computer by holding the coefficients of its irreducible polynomial, together with a floating point approximation to the number. Suppose we are also given a finite set of finite-sided polyhedra, given approximately using floating point numbers, together with face-pairings each of which is equal to one of our given matrices. We can then check the condition Cyclic($\mathscr{P}, R, A$) precisely, using integer arithmetic, by checking on a certain product of face-pairings. (We can use floating point arithmetic to see which words in the face-pairings need to have checks performed.)

8. Special cases

One case of Poincaré's Theorem which is often used is the case where there is a single element of $\mathscr{P}$ and all face-pairings are reflections. In that case completeness is a consequence of Lemma 5.4, provided the other axioms are satisfied. This enables a number of important examples to be constructed.

As a minor point, we note that it enables us to construct infinitely generated fuchsian groups with an arbitrary subset of the positive integers being the set of exponents of maximal cyclic subgroups. These and other applications of Poincaré's Theorem are well-known.

Poincaré's Theorem works in an especially simple way in dimension two. In this dimension, a face-pairing is called an *edge-pairing*. The following result is essentially due to de Rham [dR71].

Theorem 8.1 (dimension two). *Suppose we have a finite set $\mathscr{P}$ of finite-sided polygons in $\mathbf{H}^2$ and an edge-pairing (R, A) of the boundary edges satisfying Pairing($\mathscr{P}, R, A$), Connected($\mathscr{P}, R$) and Cyclic($\mathscr{P}, R, A$). Then the quotient Q of $\bigsqcup_{P \in \mathscr{P}} P$ by the edge-pairing is a two-dimensional hyperbolic orbifold which is obtained from a complete orbifold with geodesic boundary by removing the compact boundary components. The hyperbolic structure on Q is induced in an obvious way from the hyperbolic structure on the hyperbolic polygons used to define it. The group G generated by the edge-pairings in the manner described in Definition 4.2 is discrete. If all the polygons are compact, then Q is a compact orbifold without boundary. (But it may have mirrors.)*

Remark 8.2. The main feature of this result is that for the two-dimensional case it describes the quotient Q even when this is not complete. (For the conditions under which Q is complete the reader is referred to Lemma 5.4 and Theorem 6.3.)