Computational Geometry 46 (2013) 541–551

Contents lists available at SciVerse ScienceDirect

Computational Geometry: Theory and Applications

www.elsevier.com/locate/comgeo

Computing the bounded subcomplex of an unbounded polyhedron

Sven Herrmann [a], Michael Joswig [b,*], Marc E. Pfetsch [c]

[a] School of Computing Sciences, University of East Anglia, Norwich, NR4 7TJ, UK
[b] Fachbereich Mathematik, TU Darmstadt, 64289 Darmstadt, Germany
[c] Institute for Mathematical Optimization, TU Braunschweig, 38106 Braunschweig, Germany

A R T I C L E I N F O

Article history:
Received 14 June 2010
Received in revised form 12 November 2010
Accepted 10 November 2011
Available online 15 November 2011

Keywords:
Unbounded polyhedron
Bounded subcomplex
Hasse diagram
Vertex-incidences
Tight span

A B S T R A C T

We study efficient combinatorial algorithms to produce the Hasse diagram of the poset of bounded faces of an unbounded polyhedron, given vertex–facet incidences. We also discuss the special case of simple polyhedra and present computational results.

1. Introduction

The *bounded subcomplex* of an (unbounded) convex polyhedron, is its set of bounded (or, equivalently, compact) faces, partially ordered by inclusion. Polytopal complexes of this type arise in several situations each of which is interesting in its own right. Prominent examples are the *tight spans* of finite metric spaces of Dress [11], see also Isbell [16], and the *tropical polytopes* of Develin and Sturmfels [9]. The purpose of this note is to present algorithms to deal with such objects.

Typically, in applications a bounded complex of a polyhedron P is given only implicitly by a set of inequalities defining P. The primary goal is to establish algorithms to make the bounded subcomplex explicit from this input. A direct approach is to enumerate the full face lattice $\mathcal{L}(\overline{P})$ of a polytope $\overline{P}$ projectively equivalent to P (which exists if P is pointed) and to filter it to obtain $\mathcal{B}(P)$, the poset of bounded faces. However, since $\mathcal{L}(\overline{P})$ often is much larger than $\mathcal{B}(P)$, this is not efficient.

One natural approach is to start with a (dual) convex hull computation which also yields the vertex–facet incidences. Here we focus on combinatorial algorithms which take these vertex–facet incidences as input. Other approaches are discussed briefly. Our Algorithm 1 in Section 3.1 is a modification of a combinatorial algorithm for face lattice enumeration [20]. It uses the vertex–facet incidences of $\overline{P}$. This method of *selective generation* is best possible, in the sense that its running time is linear in the size of $\mathcal{B}(P)$. If only the vertex–facet incidences of P are given, that is, no information about the unbounded edges is present, one can still compute $\mathcal{B}(P)$, although at the cost of a higher running time (quadratic in the size of $\mathcal{B}(P)$). The corresponding Algorithm 2 is based on interleaving the computation of the bounded faces with an incremental computation of the poset's Möbius function.

* Corresponding author.
E-mail addresses: s.herrmann@uea.ac.uk (S. Herrmann), joswig@mathematik.tu-darmstadt.de (M. Joswig), m.pfetsch@tu-bs.de (M.E. Pfetsch).

0925-7721/$ – see front matter
doi:10.1016/j.comgeo.2011.11.002

For the sake of simplicity of exposition we usually consider complexity questions in the RAM model. However, it is easy to modify each of our results such that the bit complexity can be determined: The only potential source of non-polynomiality in terms of RAM complexity arises from calling LP type oracles. Therefore, whenever necessary, we explicitly mention the relevant sizes of the linear programming problems which need to be solved.

If additional structural information is available, specialized algorithms come into focus. For instance, in the case where the polyhedron P is simple, it turns out that $\mathcal{B}(P)$ is determined by the (directed) vertex–edge graph of P, see [15, Section 2]. This is an unbounded version of a result of Blind and Mani-Levitska [5] obtained by applying techniques due to Kalai [21]. In Section 4, we employ the reverse search approach of Avis and Fukuda [3] to generate the (directed) vertex–edge graph of $\mathcal{B}(P)$ from an inequality description of P, provided that P is simple. This can be used to either construct the bounded faces or to efficiently compute the face numbers of P and $\mathcal{B}(P)$.

In Section 5, computational experiments with Algorithm 1 are presented. We investigate five different cases in which it is interesting to compute the bounded subcomplex. The computations show that some (surprisingly) large instances can be handled. The paper closes with a list of open problems.

We are grateful to an anonymous referee for meticulous reading and for requiring a correction in Algorithm 2. This also contributed to a cleaner description of this method.

2. Preliminaries and notation

Let P be a polyhedron. The lattice of all faces (*face lattice*) of P is denoted by $\mathcal{L}(P)$. We define $\varphi(P) = |\mathcal{L}(P)|$, that is, the number of all faces of P. Moreover, let n be the number of vertices of P and m be its number of facets.

A polyhedron is called *pointed* if it does not contain an entire affine line. In the sequel we always assume that P is pointed but unbounded. For our purposes this does not mean any loss of generality, since the bounded subcomplex of a polyhedron is non-empty if and only if it is pointed. In this case, the polyhedron P is projectively equivalent to a polytope $\overline{P}$. Each such polytope is called a *projective closure* of P. A projective closure of P has a special face F_∞ (the *far face*) corresponding to the face at infinity. Fixing an admissible projective transformation γ which maps P to $\overline{P}$, the face F_∞ is the unique maximal face among the faces of $\overline{P}$ that are not images of faces of P under γ. Note that the combinatorial type of F_∞ depends on the geometry of P, not only its combinatorial structure, and that its dimension can be any number in $\{0, \dots, \dim P - 1\}$.

We begin with the description of an algorithm to compute the polytope $\overline{P}$ from an inequality description of $P \subseteq \mathbb{R}^d$. We refer to Ziegler's monograph [26] for a general discussion of admissible projective transformations and [19, §3.4] for an explicit construction. First, we compute one vertex v of P by solving a linear program similar to Phase I of the Simplex Algorithm. Such a vertex exists as P is pointed. In a second step we determine the active constraints at v, that is, those inequalities which are satisfied with equality at v. Let τ be the affine transformation moving v into the origin. In the image $\tau(P)$ the constraints active at 0 correspond to homogeneous linear equations. Any subset of those constraints defines a polyhedral cone containing the translated polyhedron $\tau(P)$. Among the constraints active at 0 we choose a dual basis by Gauß–Jordan-elimination. Next we pick a linear transformation ρ mapping this dual basis to the basis which is dual to the standard basis of $\mathbb{R}^d$. Then the image $\rho(\tau(P))$ is a polyhedron with vertex 0 such that the boundary hyperplanes of the positive orthant define valid inequalities. In particular, $\rho(\tau(P))$ is contained in the positive orthant.

In general, intersecting the image of a polyhedron in $\mathbb{R}^d$ under a projective transformation in $\mathrm{PGL}_d(\mathbb{R})$ with $\mathbb{R}^d$ may yield something non-convex. However, if $M \in \mathrm{GL}_{d+1}(\mathbb{R})$ is an invertible matrix with non-negative coefficients, then the induced projective linear transformation $[M] \in \mathrm{PGL}_d(\mathbb{R})$ maps the positive orthant into itself, and the image of any polyhedron inside is again a polyhedron. Now let μ be the unique projective linear transformation which fixes each coordinate hyperplane of $\mathbb{R}^d$ and which additionally maps the hyperplane at infinity to the affine hyperplane $\sum x_i = 1$. This transformation can be written as a $(d+1) \times (d+1)$-matrix of zeroes and ones. The polyhedron

$$\overline{P} := \mu\big(\rho\big(\tau(P)\big)\big)$$

is contained in the simplex given by $\sum x_i \leqslant 1$ and $x_i \geqslant 0$ for $i \in [d]$. In particular, $\overline{P}$ is bounded and projectively equivalent to P. The equation $\sum x_i = 1$ defines a hyperplane supporting $\overline{P}$, and its intersection with $\overline{P}$ is the far face F_∞. For given matrix A and right-hand side b, let $\ell(A, b)$ be the time complexity of solving the linear feasibility problem to find a point x satisfying $Ax \leqslant b$. The discussion above can be summarized as follows.

Proposition 1. *If $P = P(A, b)$ is given by inequalities, there exists an explicit algorithm to compute $\overline{P}$ in $O(\ell(A, b))$ time.*

The unbounded edges of P correspond to edges of $\overline{P}$ that contain exactly one vertex in F_∞. The vertices of P correspond to vertices of $\overline{P}$ that are not contained in F_∞. Via these relations it is easy to describe the face lattice of P in terms of $\overline{P}$ and vice versa. The following relationship holds between the size $\varphi = \varphi(P)$ of the face posets of P and the size $\overline{\varphi} = \varphi(\overline{P})$ of its projective closure.

Lemma 2. *We have $\overline{\varphi} \leqslant 2(\varphi - 1)$.*