



# Formalizing the Face Lattice of Polyhedra

Xavier Allamigeon<sup>1,2(✉)</sup>, Ricardo D. Katz<sup>3</sup>, and Pierre-Yves Strub<sup>4</sup>

<sup>1</sup> Inria, Palaiseau, France

xavier.allamigeon@inria.fr

<sup>2</sup> CMAP, CNRS, Ecole Polytechnique, Institut Polytechnique de Paris,  
Palaiseau, France

<sup>3</sup> CIFASIS–CONICET, Rosario, Argentina

katz@cifasis-conicet.gov.ar

<sup>4</sup> LIX, CNRS, École Polytechnique, Institut Polytechnique de Paris,  
Palaiseau, France

pierre-yves@strub.nu

**Abstract.** Faces play a central role in the combinatorial and computational aspects of polyhedra. In this paper, we present the first formalization of faces of polyhedra in the proof assistant COQ. This builds on the formalization of a library providing the basic constructions and operations over polyhedra, including projections, convex hulls and images under linear maps. Moreover, we design a special mechanism which automatically introduces an appropriate representation of a polyhedron or a face, depending on the context of the proof. We demonstrate the usability of this approach by establishing some of the most important combinatorial properties of faces, namely that they constitute a family of graded atomistic and coatomistic lattices closed under sublattices.

## 1 Introduction

A face of a polyhedron is defined as the set of points reaching the maximum (or minimum) of a linear function over the polyhedron. Faces are ubiquitous in the theory of polyhedra, and especially in the complexity analysis of optimization algorithms. As an illustration, the simplex method, one of the most widely used algorithms for solving linear programming, finds an optimal solution by iterating over the *graph* of the polyhedron, i.e. the adjacency graph of vertices and edges, which respectively constitute the 0- and 1-dimensional faces. The problem of finding a pivoting rule, i.e. a way to iterate over the graph, which ensures to reach an optimal vertex in a polynomial number of steps, is a central problem in computational optimization, related with Smale’s ninth problem for the twenty-first century [25]. Faces of polyhedra are also involved in the worst-case complexity analysis of other

---

The first author was partially supported by the ANR CAPPs project (ANR-17-CE40-0018), and by a public grant as part of the Investissement d’avenir project, reference ANR-11-LABX-0056-LMH, LabEx LMH, in a joint call with Gaspard Monge Program for optimization, operations research and their interactions with data sciences. The third author was partially supported by the ANR SCRYPT project (ANR-18-CE25-0014).

© Springer Nature Switzerland AG 2020

N. Peltier and V. Sofronie-Stokkermans (Eds.): IJCAR 2020, LNAI 12167, pp. 185–203, 2020.

[https://doi.org/10.1007/978-3-030-51054-1\\_11](https://doi.org/10.1007/978-3-030-51054-1_11)

optimization methods, such as interior point methods; see [2, 13]. This has motivated several mathematical problems on the combinatorics of faces, which are of independent interest. For example, the question of finding a polynomial bound on the diameter of the graphs of polyhedra (in the dimension and the number of defining inequalities) is still unresolved, despite recent progress [6, 7, 23]. We refer to [12] for a recent account on the subject.

Other applications of polyhedra and their faces arise in formal verification, in which passing from a representation by inequalities to a representation as the convex hull of finitely many points and vice versa, is a critical computational step. The correctness analysis of the algorithms solving this problem, extensively relies on the understanding of the mathematical structure of faces, in particular of vertices, edges and facets (i.e. 1-co-dimensional faces).

In this paper, we formalize a significant part of the properties of faces in the proof assistant COQ. As usually happens in the formalization of mathematics, one of the key difficulties is to find the right representation for objects in the proof assistant. For polyhedra and their faces, the choice of the representation depends on the context. In more detail, every polyhedron admits infinitely many descriptions by linear inequality systems. In mathematics textbooks, proofs are carried out by choosing one (often arbitrary) inequality system for a polyhedron  $\mathcal{P}$ , and then manipulating the faces of  $\mathcal{P}$  or other subsequent polyhedra through inequality systems which derive from the one chosen for  $\mathcal{P}$ . Proving that these are valid inequality systems is usually trivial for the reader, but not for the proof assistant. We exploit the so-called *canonical structures* of COQ in order to achieve this step automatically. This allows us to obtain proof scripts which only focus on the relevant mathematical content, and which are closer to what mathematicians write.

Thanks to this approach, we show that the faces of a polyhedron  $\mathcal{P}$  form a finite lattice, in which the order is the set inclusion, the bottom and top elements are respectively the empty set and  $\mathcal{P}$ , and the meet operation is the set intersection. We establish that the face lattice is both atomistic and coatomistic, meaning that every element is the join (resp. the meet) of a finite set of atoms (resp. coatoms). Atoms and coatoms respectively correspond to minimal and maximal elements distinct from the top and bottom elements. Moreover, we prove that the face lattice is graded, i.e. every maximal chain has the same length. Finally, we show that the family of face lattices of polytopes (convex hulls of finitely many points) is closed under taking sublattices, i.e. any sublattice of the face lattice of a polytope is isomorphic to the face lattice of another polytope. As a consequence of that, we prove that any sublattice of height two is isomorphic to a diamond.

Formalizing these results requires the introduction of several important and non-trivial notions. First of all, our work relies on the construction of a library manipulating polyhedra, which provides all the basic operations over them, including intersections, projections, convex hulls, as well as special classes of polyhedra such as affine subspaces. Dealing with faces also requires to formalize the dimension of a polyhedron, and its relation with the dimension of its affine hull, i.e. the smallest affine subspace containing it. Some classes of faces also

retain a particular attention, such as vertices, edges and facets. For instance, we formalize the vertex figure, which is a geometric construction to manipulate the faces containing a fixed vertex.

Throughout this work, we have drawn much inspiration from the textbooks of Schrijver [24] and Ziegler [28] to guide us in our approach. The source code of our formalization is done within the Coq-Polyhedra project, and is available at <https://github.com/Coq-Polyhedra/Coq-Polyhedra/tree/IJCAR-2020>, in the directory `theories`. We rely on the Mathematical Components library [18] (abridged MathComp thereafter) for basic data structures such as finite sets, ordered fields, and vector spaces.

The paper is organized as follows. In Sect. 2, we present how we define the basic operations and constructions over polyhedra. Section 3 deals with the central problem of finding an appropriate representation of faces, and explains how this leads to a seamless formalization of important properties like the dimension. Section 4 demonstrates the practical usability of our approach, by presenting the formalization of the face lattice and its main characteristics. Finally, we discuss related work in Sect. 5. A link to the relevant source files is given beside section titles in order to help the reader finding the results in the source code of the formalization.

## 2 Constructing a Library Manipulating Polyhedra

### 2.1 The Quotient Type of Polyhedra<sup>1,2</sup>

We recall that a (*convex*) *polyhedron* of  $\mathbb{R}^n$  is defined as the intersection of finitely many *halfspaces*  $\{x \in \mathbb{R}^n : \langle \alpha, x \rangle \geq \beta\}$ , where  $\alpha \in \mathbb{R}^n$ ,  $\beta \in \mathbb{R}$ , and  $\langle \cdot, \cdot \rangle$  is the Euclidean scalar product, i.e.  $\langle y, z \rangle := \sum_{1 \leq i \leq n} y_i z_i$ . Equivalently, a polyhedron can be represented as the solution set of a linear affine system  $Ax \geq b$ , where  $A \in \mathbb{R}^{m \times n}$  and  $b \in \mathbb{R}^m$ , in which case each inequality  $A_i x \geq b_i$  corresponds to a halfspace.

Throughout the paper, we use the variable `n : nat` to represent the dimension of the ambient space. Instead of dealing with polyhedra over the reals, we introduce a variable `R : realFieldType` which represents an abstract ordered field with decidable ordering. In this setting, `'cV[R]_n` (or `'cV_n` for short) stands for the type of column vectors of size `n` over the field `R`.

As we mentioned earlier, the representation by inequalities (or halfspaces) of a convex polyhedron  $\mathcal{P}$  is not unique. The first step in our work is to introduce a quotient structure, in order to define the basic operations (membership of a point, inclusion, etc.) regardless of the exact representation of the polyhedron. The quotient structure is based on a concrete type denoted by `'hpoly[R]_n` (or simply `'hpoly_n`, when `R` is clear from the context). The prefix letter “h” is taken from the terminology *H-polyhedron* or *H-representation* which is used to refer to

<sup>1</sup> <https://github.com/Coq-Polyhedra/Coq-Polyhedra/tree/IJCAR-2020/theories/hpolyhedron.v>.

<sup>2</sup> <https://github.com/Coq-Polyhedra/Coq-Polyhedra/tree/IJCAR-2020/theories/polyhedron.v>.

representations by halfspaces. The elements of `'hpoly_n` are records consisting of a matrix  $A \in \mathbb{R}^{m \times n}$  and a vector  $b \in \mathbb{R}^m$  representing the system  $Ax \geq b$ :

**Record** `hpoly` := `HPoly { m : nat; A : 'M_(m,n); b : 'cV_m }`.

We equip `'hpoly_n` with a membership predicate stating that, given  $P : 'hpoly_n$  and  $x : 'cV_n$ , we have  $x \text{ \texttt{in} } P$  if and only if  $x$  satisfies the system of inequalities represented by  $P$ . Two H-polyhedra are *equivalent* when they correspond to the same solution set, i.e. their membership predicate agree. We prove that this equivalence relation is decidable, by exploiting the implementation of the simplex method of [3]. The latter allows us to check that an inequality  $\langle \alpha, x \rangle \geq \beta$  is valid over an H-polyhedron  $P : 'hpoly_n$  by minimizing the linear function  $x \mapsto \langle \alpha, x \rangle$  over  $P$ , and checking that the optimal value is greater than or equal to  $\beta$ . Then, deciding whether  $P Q : 'hpoly_n$  are equivalent amounts to checking that each inequality in the system defining  $Q$  is valid over  $P$ , and vice versa.

The quotient structure is built following the approach of [10]. This introduces a quotient type, denoted here by `'poly[R]_n` (or simply `'poly_n`). Its elements are referred to as *polyhedra* and represent equivalence classes of H-polyhedra. In practice, each polyhedron is a record formed by a canonical representative of the class, and the proof that the representative is indeed the canonical one. We point out that the notion of canonical representative has no special mathematical meaning or structure.

We define the membership predicate of each  $P : 'poly_n$  as the membership predicate of its canonical representative. As expected, equality between two polyhedra of `'poly_n` and extensional equality (denoted `=i` below) of their membership predicates are equivalent properties:

**Lemma** `poly_eqP` `{P Q : 'poly_n} : (P = Q) <-> (P =i Q)`.

## 2.2 Operations over Polyhedra<sup>3</sup>

We first lift a number of basic primitives from the type `'hpoly_n` to the quotient type `'poly_n`, including the subset relation  $P \text{ \texttt{<=}} Q$  and the intersection operation  $P \text{ \texttt{\&}} Q$ . The related properties are also lifted by using the fact that the membership predicate of any element of `'hpoly_n` is extensionally equivalent to the membership predicate of its equivalence class in `'poly_n`.

Even though we now work on the quotient type, we still need a way to build polyhedra from sets of inequalities. While H-polyhedra rely on inequality constraints under the matrix form, we choose now to be closer to the mathematical definition of polyhedra as the intersection of finitely many halfspaces. To this end, we introduce the type `\rel[R]_n` (or simply `\rel_n` when  $R$  is clear from the context), which is isomorphic to the cartesian product  $'cV_n * R$  of vectors of size  $n$  and elements of  $R$ . This type is used to construct linear affine inequalities

<sup>3</sup> <https://github.com/Coq-Polyhedra/Coq-Polyhedra/tree/IJCAR-2020/theories/polyhedron.v>.

or equalities. In more detail, if  $e$  represents the pair  $(\alpha, \beta) \in \mathbb{R}^n \times \mathbb{R}$ , then the polyhedron `[hs e]` corresponds to the halfspace  $\langle \alpha, x \rangle \geq \beta$ :

**Lemma in\_hs** (`e : lrel_n`) `x : x \in [hs e] <-> ([e.1,x] >= e.2)`.

Similarly, the element  $e$  is used to build a hyperplane denoted `[hp e]`:

**Lemma in\_hp** (`e : lrel_n`) `x : x \in [hp e] <-> ([e.1,x] = e.2)`.

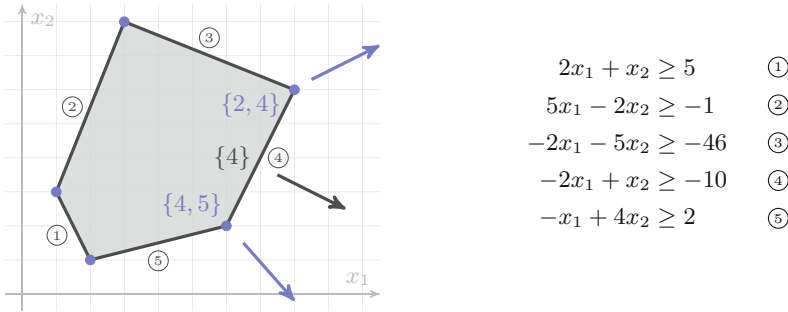
(In the last two statements, the terms `e.1` and `e.2` respectively stand for the first and second component of the pair formed by  $e$ , while `[. . .]` stands for the scalar product between two vectors.)

We can now construct polyhedra defined by sets of inequalities. To this aim, we use the type `{fset lrel_n}` of finite sets of elements of type `lrel_n`. Then, given `base : {fset lrel_n}`, the polyhedron denoted by `'P(base)` is defined as the intersection of the halfspaces `[hs e]` for `e \in base`. In particular, we introduce the empty polyhedron `[poly0]` and the full polyhedron `[polyT]`, which are defined by the inequality  $1 \leq 0$  and by no inequality respectively. As we shall see in Sect. 3, the formalization of faces requires us to manipulate polyhedra defined by systems mixing inequalities and equalities. We denote such a polyhedron by `'P^(base; I)`, where both `base` and `I` are of type `{fset lrel_n}`. It represents the intersection of the polyhedron `'P(base)` with the hyperplanes `[hp e]` for `e \in I`.

The cornerstone of more advanced constructions is the primitive `proj0`, which, given `P : 'poly_(n.+1)`, builds its projection on the last  $n$  components. This is carried out by implementing Fourier–Motzkin elimination algorithm (see e.g. [24, Chapter 12]). In short, this algorithm starts from a system of linear inequalities, and constructs pairwise combinations of them in order to eliminate the first variable. The result is that the new system is a valid representation of the projected polyhedron. This is written as follows:

**Theorem proj0P** (`P : 'poly_(n.+1)`) :  
`reflect (exists2 y : 'cV_(n.+1), x = row' 0 y & y \in P) (x \in proj0 P)`.

where `row' 0 y : 'cV_n` is the projection of  $y$  on the last  $n$  components, and `reflect` stands for a logical equivalence between the two properties. This projection primitive then allows us to construct many more polyhedra. For example, we can build the image of a polyhedron  $\mathcal{P}$  by the linear map represented by a matrix  $A \in \mathbb{R}^{k \times n}$ . The latter is obtained by embedding  $\mathcal{P}$  in a polyhedron over the variables  $(x, y) \in \mathbb{R}^{n+k}$ , intersecting it with the equality constraints  $y = Ax$ , and finally projecting it on the last  $k$  components. The construction of the convex hull of finitely many points immediately follows. Indeed, the convex hull of a finite set  $V = \{v^1, \dots, v^p\} \subset \mathbb{R}^n$  can be defined as the image of the simplex  $\Delta_p := \{\mu \in (\mathbb{R}_{\geq 0})^p : \sum_{i=1}^p \mu_i = 1\}$  by the linear map  $\mu \mapsto \sum_{i=1}^p \mu_i v^i$ . We denote the convex hull by `conv V` where `V : {fset 'cV_n}` represents a finite set of points, and we obtain (cf. **Lemma in\_convP**) that `x \in conv V` if and only if  $x$  is a barycentric combination of the points of  $V$ . The convex hull constructor yields some other elementary yet very useful constructions, such as polyhedra



**Fig. 1.** A polyhedron, defined by the inequalities on the right, and its faces. The vertices (0-dim. faces) are represented by blue dots, while the edges (1-dim. faces) are depicted in black. Arrows correspond to linear functions associated with some of the faces, in the sense of Definition 1. We also indicate beside them the set  $I$  of the defining inequalities turned into equalities, as in Theorem 1. (Color figure online)

reduced to a single point (denoted [pt  $x$ ] where  $x : 'cV\_n$ ) or segments between two points (denoted [segm  $x$ ;  $y$ ] where  $x\ y : 'cV\_n$ ).

Finally, we recover some important results of the theory of polyhedra which were proved in [3]. In more detail, we lift a version of Farkas Lemma expressed on the type 'hpoly\_n, and then obtain the Strong Duality Theorem, the complementary slackness conditions (which are conditions characterizing the optimality of solutions of linear programs), and some separation results. We refer to Section Separation and Section Duality for further details on these statements.

### 3 Representing Faces of Polyhedra<sup>4</sup>

#### 3.1 Equivalent Definitions of Faces

Faces are commonly defined as sets of optimal solutions of linear programs, i.e. problems consisting in minimizing a linear function over a polyhedron.

**Definition 1.** A set  $\mathcal{F}$  is a face of the polyhedron  $\mathcal{P} \subset \mathbb{R}^n$  if  $\mathcal{F} = \emptyset$  or there exists  $c \in \mathbb{R}^n$  such that  $\mathcal{F}$  is the set of points of  $\mathcal{P}$  minimizing the linear function  $x \mapsto \langle c, x \rangle$  over  $\mathcal{P}$ .

We note that  $\mathcal{P}$  is a face of itself (take  $c = 0$ ). Figure 1 provides an illustration of this definition.

In formal proving, the choice of the definition plays a major role on the ability to prove complex properties of the considered objects. A drawback of the previous definition is that it does not directly exhibit some of the most basic properties of faces: for instance, the fact that a face is itself a polyhedron, the fact that the intersection of two faces is a face, or the fact that a polyhedron

<sup>4</sup> [https://github.com/Coq-Polyhedra/Coq-Polyhedra/tree/IJCAR-2020/theories/poly\\_base.v](https://github.com/Coq-Polyhedra/Coq-Polyhedra/tree/IJCAR-2020/theories/poly_base.v).

has finitely many faces. In contrast, these properties are straightforward consequences of the following characterization of faces:

**Theorem 1.** *Let  $\mathcal{P} = \{x \in \mathbb{R}^n : Ax \geq b\}$ , where  $A \in \mathbb{R}^{m \times n}$  and  $b \in \mathbb{R}^m$ . A set  $\mathcal{F}$  is face of  $\mathcal{P}$  if and only if  $\mathcal{F} = \emptyset$  or there exists  $I \subset \{1, \dots, m\}$  such that*

$$\mathcal{F} = \mathcal{P} \cap \{x \in \mathbb{R}^n : A_i x = b_i \text{ for all } i \in I\}. \tag{1}$$

Nevertheless, Theorem 1 is expressed in terms of a certain H-representation of the polyhedron  $\mathcal{P}$ , while the property of being a face is intrinsic to the set  $\mathcal{P}$ . This raises the problem of exploiting the most convenient representation of  $\mathcal{P}$  to apply the characterization of Theorem 1. We illustrate this on the proof of the following property, which is used systematically (or even implicitly) in almost every proof of statements on faces:

**Proposition 1.** *If  $\mathcal{F}$  is a face of  $\mathcal{P}$ , then any face of  $\mathcal{F}$  is a face of  $\mathcal{P}$ .*

Assume  $\mathcal{P}$  is represented by the inequality system  $Ax \geq b$ , and take  $I$  as in (1). Let  $\mathcal{F}'$  be a nonempty face of  $\mathcal{F}$ . We apply Theorem 1 with  $\mathcal{F}$  as  $\mathcal{P}$ , by using the following H-representation of  $\mathcal{F} : Ax \geq b$  and  $-A_i x \geq -b_i$  for  $i \in I$ . We get that  $\mathcal{F}' = \mathcal{F} \cap \{x \in \mathbb{R}^n : A_i x = b_i \text{ for all } i \in I'\}$  for a certain set  $I' \subset \{1, \dots, m\}$ . We deduce that  $\mathcal{F}' = \mathcal{P} \cap \{x \in \mathbb{R}^n : A_i x = b_i \text{ for all } i \in I \cup I'\}$ , and conclude that  $\mathcal{F}'$  is a face of  $\mathcal{P}$  by applying Theorem 1. While the choice of the H-representation of  $\mathcal{P}$  is irrelevant, we point out that the proof would not have been so immediate if we had initially chosen an arbitrary H-representation of  $\mathcal{F}$ .

### 3.2 Working Within a Fixed Ambient H-Representation

Theorem 1 leads us to the following strategy: when dealing with the faces of a polyhedron, and possibly with the faces of these faces, etc., we first set an H-representation of the top polyhedron, and then manipulate the subsequent H-representations of faces in which some inequalities are strengthened into equalities, like in (1).

The top H-representation will be referred to as the *ambient representation*, and is formalized as a term `base` of type `{fset lrel_n}` representing a finite set of inequalities. Then, we introduce the type `{poly base}`, which corresponds to the subtype of `'poly_n` whose inhabitants are the polyhedra  $Q$  satisfying the following property:

**Definition `has_base`** `base Q := (Q != [poly0]) -> exists I : {fsubset base}, Q = 'P^=(base; I).`

where `{fsubset base}` is the type of subsets of `base`. We recall that `'P^=(base; I)` denotes the polyhedron defined by the inequalities in `base`, with the additional constraint that the inequalities in the subset `I` are satisfied with equality. This means that `{poly base}` corresponds to the polyhedra defined by equalities or inequalities in `base`. The choice of the name `base` is reminiscent of the terminology used in fiber bundles. Indeed, as we shall see in the next sections, several proofs will adopt the scheme of fixing a base locally, and then working on polyhedra

of type `{poly base}`. Following this analogy, the latter may be thought of as a fiber.

We now present a first formalization of the set of faces relying on the subtype `{poly base}`:

```
Definition pb_face_set (P : {poly base}) :=
  [set Q : {poly base} | Q `<= ` P].
```

It defines the set of faces of  $P : \{\text{poly base}\}$  as the set of elements of `{poly base}` contained in  $P$ . With this definition, some properties of faces come for free. For instance, the finiteness of the set of faces follows from the fact that there are only finitely many inhabitants of the type `{fsubset base}`, and subsequently of `{poly base}`. Another example is that Proposition 1 straightforwardly derives from the transitivity of the inclusion relation ``<= ``.

Some other properties come at the price of proving that a polyhedron inhabits the type `{poly base}`. As an example, if  $P : \{\text{poly base}\}$  and  $c : 'cV\_n$ , the polyhedron `argmin P c : 'poly_n` is defined as the set of points of  $P$  minimizing the function `fun x => '[c,x]`. Showing that `argmin P c` is a face of  $P$  essentially amounts to proving the following property:

```
Lemma argmin_baseP (P : {poly base}) c : has_base base (argmin P c).
```

Indeed, the inclusion `argmin P c `<= ` P` is immediate from the definition of the polyhedron `argmin P c`. However, even once Lemma `argmin_baseP` is proved, we cannot yet write a statement of the form `argmin P c \in pb_face_set P` due to the fact that `argmin P c` has type `'poly_n`. In order to turn it into an element of the subtype `{poly base}`, we need to explain in more detail how this type is defined. The type `{poly base}` is a short-hand notation for the following inductive type:

```
Inductive poly_base base :=
  PolyBase { pval :> 'poly_n ; _ : has_base base pval }.
```

In other words, an element of type `{poly base}` is a record formed by an element `pval : 'poly_n` and a proof that the property `has_base base pval` holds. While we could construct the element `PolyBase (argmin_baseP P c)`, we introduce a more general scheme to cast elements of type `'poly_n` to `{poly base}` whenever possible. This scheme relies on COQ canonical structures, which provide an automatic way to recover a term of record type from the head symbol. The association is declared as follows:

```
Canonical argmin_base (P : {poly base}) c := PolyBase (argmin_baseP P c).
```

One restriction of COQ is that canonical structures are resolved only when unifying types, and not arbitrary terms. This is why our primitive `poly_base_of`, which casts a  $Q : 'poly\_n$  to a `{poly base}`, encapsulates the value  $Q$  in a *phantom type*, i.e. a type isomorphic to the unit type, but with a dependency to  $Q$ .



**Definition** `poly_base_of` ( $Q : \{\text{poly base}\}$ ) ( $\_ : \text{phantom 'poly\_n } Q$ ) :=  $Q$ .

**Notation** "`Q %:poly_base`" := (`poly_base_of (Phantom _ Q)`).

In consequence, writing `(argmin P c)%:poly_base` triggers the unification algorithm between the term `argmin P c` and a value of type `{poly base}`, which is resolved using the `Canonical` declared above. We finally end up with the following statement

**Lemma** `argmin_pb_face_set` `base` ( $P : \{\text{poly base}\}$ )  $c :$   
`(argmin P c)%:poly_base \in pb_face_set P.`

whose proof is trivial: it just amounts to proving the inclusion `argmin P c `<=` P`.

We declare other canonical structures over elementary constructions for which the property `has_base base _` can be shown to be satisfied. This includes the intersection `P `&` Q` of two elements  $P Q : \{\text{poly base}\}$ , the empty set `[poly0]`, or polyhedra of the form `'P(base)` or `'P^=(base; .)`. This allows us to cast complex terms to the type `{poly base}`, or, said differently, to prove automatically that they satisfy the property `has_base base _`. As an example, the term

`('P^=(base; I) `&` argmin 'P(base) c)%:poly_base`

typechecks thanks to multiple resolutions of canonical structures on the aforementioned declarations, without requiring extra proof from the user. We refer to [21] for the use of canonical structures in formal mathematics.

We point out that `Lemma argmin_pb_face_set` is a proof of one side of the equivalence between the definition of faces brought by `pb_face_set` and `Definition 1` (i.e. the equivalence in `Theorem 1`). The other side can be written as follows:

**Theorem** `pb_faceP` `base` ( $P Q : \{\text{poly base}\}$ ) :  
`Q \in pb_face_set P -> Q != [poly0] ->`  
`exists c, Q = (argmin P c)%:poly_base.`

When  $Q$  is nonempty, we use a set  $I$  such that  $Q = 'P^=(base, I)$ , and we build  $c$  as the sum of the vectors `-e.1 : 'cV_n` for  $e \in I$ . The equality  $Q = \text{argmin } P \ c$  follows from a routine verification of the complementary slackness conditions.

### 3.3 Getting Free from Ambient Representations

So far, we have worked with a fixed ambient representation `base`, and restricted the formalization of faces to polyhedra that can be expressed as terms of type `{poly base}`. We now describe how to formalize the set of faces of any polyhedron of type `'poly_n` as a finite set of polyhedra of the same type, without sacrificing the benefits brought by `{poly base}`.

First, we exploit the observation that for each polyhedron  $P : 'poly\_n$ , there exists `base : {fset lrel_n}` and  $P' : \{\text{poly base}\}$  such that  $P = \text{pval } P'$  (recall that `pval` also stands for the coercion from the type `{poly base}` to `'poly_n`). This can be proved by exploiting the definition of the quotient type `'poly_n`.

Indeed,  $P$  admits a representative  $\text{hrepr } P : \text{'hpoly\_n}$  corresponding to a certain  $H$ -representation, from which we can build a term  $\text{base} : \{\text{fset } \text{lrel\_n}\}$  such that  $P = \text{pval } \text{'P}(\text{base})\%:\text{poly\_base}$ .

Second, we introduce another quotient structure over the type  $\text{'poly\_n}$ , in order to deal with the fact that a polyhedron may correspond to several elements of type  $\{\text{poly } \text{base}\}$  for different values of  $\text{base}$ . Our construction amounts to showing that  $\text{'poly\_n}$  is isomorphic to the quotient of the dependent sum type  $\sum_{\text{base}} \{\text{poly } \text{base}\}$  by the equivalence relation in which  $Q1 : \{\text{poly } \text{base1}\}$  and  $Q2 : \{\text{poly } \text{base2}\}$  are equivalent if  $\text{pval } Q1 = \text{pval } Q2$ . Given a polyhedron  $P$  of type  $\text{'poly\_n}$ , this construction provides us a canonical ambient representation denoted  $\backslash\text{repr\_base } P : \{\text{fset } \text{lrel\_n}\}$ , and an associated canonical representative  $\backslash\text{repr } P$  of type  $\{\text{poly } (\backslash\text{repr\_base } P)\}$  satisfying  $P = \text{pval } (\backslash\text{repr } P)$ .

We are now ready to define the set of faces of  $P$  in full generality:

```
Definition face_set (P : 'poly_n) :=
  [fset (pval F) | F in pb_face_set (\repr P)]%fset.
```

which corresponds to the image by the coercion  $\text{pval}$  of the face set of  $\backslash\text{repr } P$  (here,  $\text{pval}$  has type  $\{\text{poly } (\backslash\text{repr\_base } P)\} \rightarrow \text{'poly\_n}$ ). Of course, we need to check that this definition is independent of the choice of the representative of  $P$  in this new quotient structure. This is written as follows:

```
Lemma face_set_morph (base : {fset lrel_n}) (P : {poly base}) :
  face_set P = [fset pval F | F in pb_face_set P]%fset.
```

The proof relies on the geometric properties of the elements of  $\text{pb\_face\_set}$  established in Sect. 3.2. Indeed, they imply that, regardless of the choice of the ambient representation, the set  $[\text{fset } \text{pval } F \mid F \text{ in } \text{pb\_face\_set } P]$  always consists of the empty set  $[\text{poly}\emptyset]$  and the polyhedra of the form  $\text{argmin } P \text{ c}$ .

Now that this architecture is settled, we can prove some of the basic properties of faces. Most of the proof make use of the following elimination principle:

```
Lemma polybW (Pt : 'poly_n -> Prop) :
  (forall (base : {fset lrel_n}) (Q : {poly base}), Pt Q) ->
  (forall P : 'poly_n, Pt P).
```

which means that, given a property to be proved on any polyhedron  $P : \text{'poly\_n}$ , it is sufficient to prove it over the type  $\{\text{poly } \text{base}\}$  for any choice of  $\text{base}$ . In practice, **Lemma polybW** is used to introduce an ambient representation. Let us illustrate it on the proof that the intersection of two faces of  $P$  is a face of  $P$ :

```
Lemma face_set_polyI (P Q1 Q2 : 'poly_n) :
  Q1 \in face_set P -> Q2 \in face_set P -> Q1 `&` Q2 \in face_set P.
```

**Proof.**

```
elim/polybW: P => base P.
```

```
case/face_setP => {}Q1 Q1_sub_P.
```

```
case/face_setP => {}Q2 Q2_sub_P.
```

```
by rewrite face_setE ?(poly_subset_trans poly_subsetIl) ?pvalP.
```

**Qed.**

The first line destructs  $P$ , and introduces the ambient representation `base` and an element still named  $P$ , now of type `{poly base}`. The second and third lines successively consume the assumptions that  $Q1$  and  $Q2$  are faces, then introduce two elements of type `{poly base}` having the same name and respectively satisfying  $Q1 \leq P$  and  $Q2 \leq P$ . Finally, the tactics `rewrite face_setE` replaces the goal  $Q1 \ \& \ Q2 \ \text{in} \ \text{face\_set} \ P$  by the following two subgoals:  $Q1 \ \& \ Q2 \leq P$  and `has_base base (Q1 \& Q2)`. Since  $(Q1 \ \& \ Q2) \leq Q1$  and  $Q1 \leq P$ , the former is proved by transitivity of the subset relation. The latter is automatically provided by the canonical structure mechanism described in Sect. 3.2, triggered by the generic statement

**Lemma** `pvalP` `base (P : {poly base}) : has_base base P.`

### 3.4 From Faces to the Affine Hull and Dimension

We argue that the approach that we have introduced to represent faces of polyhedra also perfectly fits the formalization of the affine hull and dimension of polyhedra. Recall that the *affine hull* of a polyhedron refers to the smallest (inclusionwise) affine subspace of  $\mathbb{R}^n$  containing it, and the *dimension* of the polyhedron is defined as the dimension of this subspace (i.e., the dimension of the underlying vector subspace).

To this end, given an ambient representation `base` and a polyhedron  $P$  of type `{poly base}`, we introduce the set of *active inequalities* of  $P$ , i.e. the set of `e \ \text{in} \ \text{base}` such that the corresponding inequality is satisfied as equality over  $P$ . This is written as the inclusion  $P \leq [hp \ e]$  (recall that  $[hp \ e]$  is the hyperplane  $\{[e.1, x] = e.2\}$ ). The active inequalities form a subset of `base` denoted `{eq P}`. Equivalently, when  $P$  is non-empty, `{eq P}` corresponds to the largest (inclusionwise) subset  $I$  such that  $P = P^{\wedge}(\text{base}; I)$ .

It is a classic property of polyhedra that the affine hull of a non-empty polyhedron is the affine subspace defined by the equalities in `{eq P}`. We take this property as a definition:

**Definition** `pb_hull` `base (P : {poly base}) :=`  
`if P != [poly0] then affine << {eq P} >> else [poly0].`

**Definition** `hull` `(P : 'poly_n) := pb_hull (\repr P).`

The second definition lifts the affine hull from `{poly base}` to `'poly_n`. Of course, we show that the resulting affine subspace `hull P` is independent of the choice of `base` (cf. **Lemma** `hullE`). We establish that this definition is correct w.r.t. the usual mathematical definition discussed above, i.e.:

**Lemma** `hullP` `P U : (P \leq affine U) <-> (hull P \leq affine U).`

Here,  $U$  corresponds to a vector subspace of `\rel_n`, and the term `affine U` stands for the affine subspace given by the intersection of the affine equalities represented by the elements of  $U$ . (The term `<< {eq P} >>` above corresponds to the vector subspace spanned by the elements of `{eq P}`.)

We follow the same scheme to formalize the dimension  $\dim P$  of a polyhedron  $P : \text{'poly\_n}$ , which we define as one plus the co-dimension of the vector span of  $\{\text{eq } P\}$ . The shift by one originates from the fact that  $\dim P$  ranges over the type  $\text{nat}$  of natural numbers. Therefore, we have to set the dimension of the empty set  $[\text{poly}\emptyset]$  to 0, while it is common to set it to  $-1$  in the literature. As expected, we obtain the following statement:

**Lemma `dim_hull`** ( $P : \text{'poly\_n}$ ) :  $\dim P = \dim (\text{hull } P)$ .

Like in mathematics textbooks, **Lemma `dim_hull`** is the natural way to establish the basic statements concerning the dimension, i.e. by reducing to elementary proofs over vector spaces. For instance, we establish that the dimension is monotone (**Lemma `dimS`**), and compute the dimension for important classes of polyhedra. This includes the fact that segments of two distinct points have dimension 2 (remember the shift by one of our formalization):

**Lemma `dim_seg`** ( $x \ y : \text{'cV\_n}$ ) :  $\dim [\text{segm } x; y] = (x \neq y).+1$ .

and that, conversely, any compact polyhedron of dimension 2 is a segment of two points:

**Lemma `dim2P`** ( $P : \text{'poly\_n}$ ) :  $\text{compact } P \rightarrow \dim P = 2 \rightarrow$   
 $\text{exists } x, \text{exists2 } y, P = [\text{segm } x; y] \ \& \ x \neq y$ .

(We point out that  $\text{compact } P$  is simply defined as the fact that  $P$  is a bounded set, as polyhedra are topologically closed.) Similarly, we prove that polyhedra reduced to a single point are precisely the ones having dimension 1, that proper hyperplanes have codimension 1, etc. We refer to **Section `Dimension`** for a detailed account of our results.

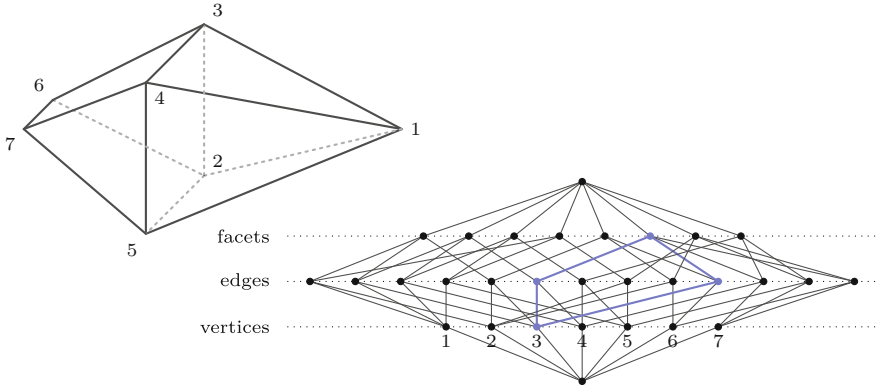
## 4 The Face Lattice<sup>5</sup>

In this section, we illustrate how the framework that we have introduced in Sect. 3 serves as a foundation for formalizing the structural properties of faces. We refer to Fig. 2 for an example of the properties presented below.

At the core of the formalization lies the theory of ordered structures such as partial orders, semilattices and lattices. Some of these structures have been very recently introduced in the MathComp library – for instance, the non-distributive lattice structure has been introduced in early 2020. However, as we shall see in this section, the formalization of the face lattice requires to implement additional objects, such as graded lattices, sublattices, and lattice homomorphisms. This development is gathered in the module `xorder.v` of the Coq-Polyhedra project.

The first property that we can immediately formalize following the results of Sect. 3 is the finite lattice structure over the set `face_set`  $P$  for  $P : \text{'poly\_n}$ . The

<sup>5</sup> [https://github.com/Coq-Polyhedra/Coq-Polyhedra/tree/IJCAR-2020/theories/poly\\_base.v](https://github.com/Coq-Polyhedra/Coq-Polyhedra/tree/IJCAR-2020/theories/poly_base.v).



**Fig. 2.** A three-dimensional polytope (left) and the Hasse diagram of its face lattice (right). A interval of height 2 is depicted in blue. (Color figure online)

partial order is given by the polyhedron inclusion  $\leq$ , the meet operator is the intersection  $\&$  (as a consequence of [Lemma face\\_set\\_polyI](#)), while the bottom and top elements are respectively  $[\text{poly}\theta]$  and  $\mathcal{P}$ . As a finite lattice, the join operator  $Q \vee Q'$  can be built as the meet of all the faces of  $\mathcal{P}$  containing both  $Q$  and  $Q'$ .

### 4.1 Facets and Gradedness

Recall that a lattice  $(L, \prec)$  is said to be *graded* if there exists a rank function  $\Phi: L \rightarrow \mathbb{N}$  such that: (i)  $\Phi(u) < \Phi(v)$  whenever  $u \prec v$ , (ii)  $u \preceq v$  and  $\Phi(u) + 1 < \Phi(v)$  implies that there exists  $w \in L$  satisfying  $u \prec w \prec v$ . Equivalently, this is a lattice in which all maximal chains have the same length.

In the case of the face lattice, the rank function can be defined as the dimension of the face. Property (i) is proved as follows. If  $Q$  and  $Q'$  are both faces of  $\mathcal{P}$  and  $Q \prec Q'$ , then  $\dim Q \leq \dim Q'$ , as the dimension is monotone. Moreover,  $\text{hull } Q \leq \text{hull } Q'$ . If we assume  $\dim Q = \dim Q'$ , then we can prove that  $\text{hull } Q$  is equal to  $\text{hull } Q'$  (as affine subspaces of the same dimension). We conclude that  $Q = Q'$  by the fact that  $F = \mathcal{P} \& \text{hull } F$  for any face  $F$  of  $\mathcal{P}$ .

The proof of Property (ii) (see [Lemma graded](#)) relies on the formalization of facets of polyhedra, and their combinatorial characterization in terms of active inequalities. We recall that a *facet* of a non-empty polyhedron  $\mathcal{P}$  is a face of  $\mathcal{P}$  of dimension  $\dim \mathcal{P} - 1$ . A classical result states that when  $\mathcal{P}$  is given by a *non-redundant* system of inequalities  $Ax \geq b$  (i.e. the H-representation is minimal inclusionwise), the facets are precisely the polyhedra of the form  $\mathcal{P} \cap \{x \in \mathbb{R}^n : A_i x = b_i\}$  for any  $i$  such that  $\mathcal{P} \not\subseteq \{x \in \mathbb{R}^n : A_i x = b_i\}$ . The formalization of this statement first goes through the construction of non-redundant bases for any polyhedron, and the proof of the following elimination principle:

```

Lemma non_redundant_baseW (Pt : 'poly_n -> Prop) :
  (forall base, non_redundant base -> Pt 'P(base)%:poly_base) ->
  (forall P : 'poly_n, Pt P).

```

This allows to specialize  $P$  to a polyhedron of the form  $'P(\text{base})$  where  $\text{base}$  is a minimal set of inequalities defining  $P$ . Using the techniques of Sect. 3, we switch to a proof environment dealing with polyhedra in  $\{\text{poly base}\}$ , and establish that the facets of  $P$  are precisely the polyhedra of the form  $'P^=(\text{base}; [\text{fset } i])$  for  $i \notin \{\text{eq } P\}$  (where  $[\text{fset } i]$  is the singleton consisting of  $i$ ). We refer to the statements **Lemma dim\_facet** and **Lemma facetP** for the exact description.

Going back to the description of the proof of Property (ii), we assume that  $Q$  and  $Q'$  are two faces of  $P$  satisfying  $Q \preceq Q'$  and  $(\dim Q).+1 < \dim Q'$ . We first consider the case where  $Q' = P$ . Since  $Q \prec P$ , we can pick an element  $i$  in  $\{\text{eq } Q\}$  but not in  $\{\text{eq } P\}$ , and verify that the facet  $F := 'P^=(\text{base}; [\text{fset } i])$  satisfies  $Q \prec F \prec P$ . The general case where  $Q'$  is a proper face of  $P$  is handled by using the fact that  $Q \in \text{face\_set } P$  and  $Q \preceq Q'$  ensures that  $Q$  is a face of  $Q'$  (see **Lemma face\_set\_of\_face**).

## 4.2 Vertices, Atomicity and Coatomicity

The *atoms* of a lattice  $L$  are the elements  $u \in L \setminus \{\perp\}$  such that there is no  $v \in L$  satisfying  $\perp \prec v \prec u$ , where  $\perp$  denotes the bottom element of  $L$ . In the face lattice of a polyhedron  $P$ , they correspond to the faces  $F$  of  $P$  such that  $\dim F = 1$ , i.e. to the vertices of  $P$  (remember the shift by one of our formalization). This motivates the introduction of the vertex set of  $P$ , which satisfies the following two characteristic properties:

```

Lemma in_vertex_setP (P : 'poly_n) x :
  (x \in vertex_set P) <-> ([pt x] \in face_set P).
Lemma face_dim1 (P Q : 'poly_n) : Q \in face_set P -> dim Q = 1 ->
  exists2 x, Q = [pt x] & x \in vertex_set P.

```

A central property is that if  $P$  is compact, then it coincides with the convex hull of its vertices:

```

Theorem conv_vertex_set (P : 'poly_n) :
  compact P -> P = conv (vertex_set P).

```

Remark that this shows that any compact polyhedron is a polytope. Together with the converse statement (**Lemma compact\_conv** in `polyheron.v`), this brings a proof of Minkowski Theorem.

The latter result allows us to prove that, when  $P$  is compact, its face lattice is atomistic, meaning that any face of  $P$  is the join of a finite set of atoms:

```

Lemma atomisticP (Q : face_set P) :
  reflect (exists2 S, (forall x, x \in S -> atom x) & Q = \join_(x in S) x)
  (atomistic Q).
Lemma face_atomistic (Q : face_set P) : atomistic Q.


```

To prove this statement for  $Q$ , we set  $S$  to the set of vertices of  $Q$ . The latter are vertices of  $P$  as well, and thus correspond to atoms of the face lattice of  $P$ . The inclusion  $Q \leq \bigvee_{x \in \text{vertex\_set } Q} x$  is established by substituting  $Q$  by  $\text{conv } (\text{vertex\_set } Q)$  thanks to [Lemma `conv\_vertex\_set`](#), which makes the statement obvious by construction of the convex hull and the join operator. The opposite inclusion  $Q \geq \bigvee_{x \in \text{vertex\_set } Q} x$  is trivial by property of the join operator, and this concludes the proof.

The *coatoms* of  $L$  are defined dually: these are the elements  $u \in L \setminus \{\top\}$  such that there is no  $v \in L$  satisfying  $u \prec v \prec \top$ , where  $\top$  denotes the top element of  $L$ . The coatomicity of  $\text{face\_set } P$  means that any face of  $P$  is the intersection of facets of  $P$ . Our proof exploits the characterization of facets presented in [Sect. 4.1](#). We refer to [Lemma `face\_coatomic`](#) for more details.

### 4.3 Closedness Under Taking Sublattices

The *closedness under sublattices* of the face lattices of polytopes states that if  $Q$  and  $Q'$  are two faces of a polytope  $P$  such that  $Q \leq Q'$ , then the interval  $[\langle Q; Q' \rangle]$ , i.e. the sublattice formed by the faces  $F : \text{face\_set } P$  satisfying  $Q \leq F \leq Q'$ , is isomorphic to the face lattice of a polytope of dimension  $\dim Q' - \dim Q$ .

The interest of this property is that it allows involved induction schemes on the height of the face lattice. As an example, we can establish the so-called *diamond property*, namely that every sublattice of height 2 of the face lattice consists of precisely four faces ordered as . Equivalently, this means that for any two faces  $\mathcal{F}$  and  $\mathcal{F}'$  of a polytope  $\mathcal{P}$  such that  $\dim \mathcal{F}' = \dim \mathcal{F} + 2$  and  $\mathcal{F} \subset \mathcal{F}'$ , there are precisely two faces between them (see [Lemma `diamond`](#) for the statement, and [Fig. 2](#) for an illustration). The proof exploits the closedness by sublattices, and the subsequent isomorphism of any interval of height 2 with the face lattice of a polytope  $P'$  verifying  $\dim P' = 2$ . [Lemma `dim2P`](#) reduces it to the face lattice of a segment  $[\text{segm } x; y]$ , which is given by the following characterization:

[Lemma `face\_set\_seg`](#) ( $x \ y : \text{'cV\_n}$ ) :

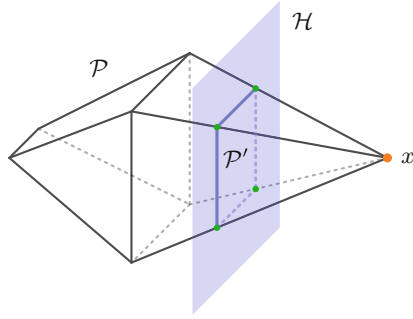
`face_set [segm x; y] = [fset [poly0]; [pt x]; [pt y]; [segm x; y]].`

The proof of the closedness by sublattices is done as follows. First, we reduce to the case where  $Q' = P$ , since the face lattice of  $Q'$  is isomorphic to the sublattice of the faces of  $P$  contained in  $Q'$ . We are left with the following statement:

[Lemma `closed\_by\_interval\_r`](#) ( $Q : \text{face\_set } P$ ) :

`exists (P' : \text{'compact_poly_n}) (f : {omorphism '[< Q; P >] -> face_set P'}),`  
`bijective f.`

The proof is done by induction on the dimension of  $Q$ . We restrict the exposition to the base case  $\dim Q = 1$ , i.e. when  $Q$  corresponds to a vertex  $x$  of  $P$ , since the general case is just handled by iterating the process. When  $\dim Q = 1$ , the construction of the polyhedron  $P'$  is achieved by the *vertex figure* method. It consists in slicing the polytope  $P$  with a hyperplane  $[\text{hp } e]$  separating the vertex



**Fig. 3.** The vertex figure construction, illustrated on the vertex  $x$  of the polytope  $\mathcal{P}$ . The hyperplane  $\mathcal{H}$  (light blue) separates  $x$  from the other vertices of the polytope. In the sliced polytope  $\mathcal{P}'$ , the vertices (green) and edges (blue) are respectively in one-to-one correspondence with the edges and facets of  $\mathcal{P}$  containing  $x$ . (Color figure online)

$x$  from the other vertices (see Fig. 3 for an illustration). We define  $\mathcal{P}'$  as the sliced polytope. It has dimension  $(\dim \mathcal{P}) - 1$ , and its face lattice can be shown to be isomorphic to the sublattice  $\{ \langle [pt \ x]; \mathcal{P} \rangle \}$ . Once again, the isomorphism is proved by exposing the polyhedron  $\mathcal{P}$  to the subtype `{poly base}` for some ambient representation `base`, and reducing to basic manipulations of sets `{eq _}` of active inequalities of faces. Interestingly, two distinct ambient representations are used in the proof: `base` for the original polytope  $\mathcal{P}$ , and its union `e +|` base` with the singleton `{e}` for the sliced polytope  $\mathcal{P}'$ . Our use of canonical structures still applies to this setting, and provides the proof that any face of  $\mathcal{P}$  sliced with the hyperplane `[hp e]` writes down over the base `e +|` base` of the sliced polytope  $\mathcal{P}'$ .

## 5 Related Work

Many software developments related with convex polyhedra have been motivated by applications to formal verification. Several libraries have been developed for this purpose, e.g. [4, 20], and, despite being informal, it is worth noting that they are also used by mathematicians to perform computation over polyhedra and polytopes, for instance in [16, 27]. Initiatives on the development of formally verified polyhedral algorithms are more recent. The works of [26] and [8] in Isabelle/HOL aim at providing a formally proven yet practical and efficient algorithm to decide linear rational arithmetic for SMT-solving. The Micromega tactic [5] relies on polyhedra to prove automatically arithmetic goals over ordered rings in Coq. The *Verified Polyhedral Library* [9, 15] targets abstract interpretation, and brings the ability to verify polyhedral computations a posteriori in Coq.

There are far fewer developments focusing on formal mathematics. Euler formula, which relates the number of vertices, edges and facets of three-dimensional



polytopes, has been proved in [14] in COQ and in [1] in Mizar. The generalization to polytopes in arbitrary dimension, namely Euler–Poincaré formula, has been formally proved in HOL-Light [19], together with several intermediate properties of polyhedra and faces. In the intuitionistic setting, we are not aware of any work concerning faces and their properties. We point out that Fourier–Motzkin elimination has been formalized in COQ by [22].

## 6 Conclusion and Future Work

In this work, we have formalized a substantial part of the theory of polyhedra and their faces, which has allowed us to obtain some of the essential properties of face lattices. Beyond the mathematical results formally proven, a special attention has been paid to the usability of the library. This goes through a mechanism to bring the right representation of faces according to the context, and the automatic proof that these representations are valid thanks to the use of canonical structures.

This foundational work opens several perspectives. First, it has raised that an important development over ordered structures is still needed, in particular for the manipulation of ordered substructures such as sublattices, and the interplay between them through morphisms. The formalization of finite groups and subgroups in [17] may provide a possible source of inspiration to solve this problem. Second, there are many other interesting properties in relation with polyhedra and their faces to be formalized, such as getting upper bounds on the diameter of polytopes, or more generally, on the number of faces (the so-called f-vector theory). However, beyond the interest of formalizing already known mathematical results, we are even more interested in using proof assistants to help getting new ones. We think of mathematical results relying on computations that are not accessible by hand. To this extent, we aim at providing a way to compute with the objects introduced in this work, directly within the proof assistant, and to introduce all the needed mechanisms for the design and development of large scale reflection tactics. A basic goal is to compute the face lattice (or part of it) of a polyhedron defined by a set of inequalities. This requires us to formalize some algorithms based on faces, and to find a way to execute them on efficient data structures, in the spirit of the approach of [11].

**Acknowledgments.** We are grateful to Assia Mahboubi for helpful discussions on the subject. We thank the anonymous reviewers for their detailed comments and their suggestions to improve the presentation of the paper.

## References

1. Alama, J.: Euler’s polyhedron formula in mizar. In: Fukuda, K., Hoeven, J., Joswig, M., Takayama, N. (eds.) ICMS 2010. LNCS, vol. 6327, pp. 144–147. Springer, Heidelberg (2010). [https://doi.org/10.1007/978-3-642-15582-6\\_26](https://doi.org/10.1007/978-3-642-15582-6_26)

2. Allamigeon, X., Benchimol, P., Gaubert, S., Joswig, M.: Log-barrier interior point methods are not strongly polynomial. *SIAM J. Appl. Algebra Geom.* **2**(1), 140–178 (2018). <https://doi.org/10.1137/17M1142132>
3. Allamigeon, X., Katz, R.D.: A formalization of convex polyhedra based on the simplex method. *J. Autom. Reason.* **63**(2), 323–345 (2019). <https://doi.org/10.1007/s10817-018-9477-1>
4. Bagnara, R., Hill, P.M., Zaffanella, E.: The Parma Polyhedra Library: toward a complete set of numerical abstractions for the analysis and verification of hardware and software systems. *Sci. Comput. Program.* **72**(1–2), 3–21 (2008)
5. Besson, F.: Fast reflexive arithmetic tactics the linear case and beyond. In: Altenkirch, T., McBride, C. (eds.) *TYPES 2006. LNCS*, vol. 4502, pp. 48–62. Springer, Heidelberg (2007). [https://doi.org/10.1007/978-3-540-74464-1\\_4](https://doi.org/10.1007/978-3-540-74464-1_4)
6. Bonifas, N., Di Summa, M., Eisenbrand, F., Hähnle, N., Niemeier, M.: On sub-determinants and the diameter of polyhedra. *Discret. Comput. Geom.* **52**(1), 102–115 (2014)
7. Borgwardt, S., De Loera, J.A., Finhold, E.: The diameters of network-flow polytopes satisfy the Hirsch conjecture. *Math. Program.* **171**(1), 283–309 (2018)
8. Bottesch, R., Haslbeck, M.W., Thiemann, R.: Verifying an incremental theory solver for linear arithmetic in Isabelle/HOL. In: Herzig, A., Popescu, A. (eds.) *FroCoS 2019. LNCS (LNAI)*, vol. 11715, pp. 223–239. Springer, Cham (2019). [https://doi.org/10.1007/978-3-030-29007-8\\_13](https://doi.org/10.1007/978-3-030-29007-8_13)
9. Boulmé, S., Maréchal, A., Monniaux, D., Périn, M., Yu, H.: The verified polyhedron library: an overview. In: 2018 20th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), September 2018, pp. 9–17 (2018). <https://doi.org/10.1109/SYNASC.2018.00014>
10. Cohen, C.: Pragmatic quotient types in Coq. In: Blazy, S., Paulin-Mohring, C., Pichardie, D. (eds.) *ITP 2013. LNCS*, vol. 7998, pp. 213–228. Springer, Heidelberg (2013). [https://doi.org/10.1007/978-3-642-39634-2\\_17](https://doi.org/10.1007/978-3-642-39634-2_17)
11. Cohen, C., Dénès, M., Mörtberg, A.: Refinements for free!. In: Gonthier, G., Norrish, M. (eds.) *CPP 2013. LNCS*, vol. 8307, pp. 147–162. Springer, Cham (2013). [https://doi.org/10.1007/978-3-319-03545-1\\_10](https://doi.org/10.1007/978-3-319-03545-1_10)
12. De Loera, J.: Algebraic and topological tools in linear optimization. *Not. Am. Math. Soc.* **66**, 1 (2019). <https://doi.org/10.1090/noti1907>
13. Deza, A., Terlaky, T., Zinchenko, Y.: Central path curvature and iteration-complexity for redundant Klee-Minty cubes. In: Gao, D., Serali, H. (eds.) *Advances in Applied Mathematics and Global Optimization. Advances in Mechanics and Mathematics*, vol. 17, pp. 223–256. Springer, Boston (2009). [https://doi.org/10.1007/978-0-387-75714-8\\_7](https://doi.org/10.1007/978-0-387-75714-8_7)
14. Dufourd, J.F.: Polyhedra genus theorem and Euler formula: a hypermap-formalized intuitionistic proof. *Theor. Comput. Sci.* **403**(2–3), 133–159 (2008)
15. Foulhe, A., Boulmé, S.: A certifying frontend for (sub)polyhedral abstract domains. In: Giannakopoulou, D., Kroening, D. (eds.) *VSTTE 2014. LNCS*, vol. 8471, pp. 200–215. Springer, Cham (2014). [https://doi.org/10.1007/978-3-319-12154-3\\_13](https://doi.org/10.1007/978-3-319-12154-3_13)
16. Gawrilow, E., Joswig, M.: polymake: a framework for analyzing convex polytopes. In: Kalai, G., Ziegler, G.M. (eds.) *Polytopes—Combinatorics and Computation. DMV Seminar*, vol. 29, pp. 43–73. Birkhäuser, Basel (2000). [https://doi.org/10.1007/978-3-0348-8438-9\\_2](https://doi.org/10.1007/978-3-0348-8438-9_2)
17. Gonthier, G., et al.: A machine-checked proof of the odd order theorem. In: Blazy, S., Paulin-Mohring, C., Pichardie, D. (eds.) *ITP 2013. LNCS*, vol. 7998, pp. 163–179. Springer, Heidelberg (2013). [https://doi.org/10.1007/978-3-642-39634-2\\_14](https://doi.org/10.1007/978-3-642-39634-2_14)

18. Gonthier, G., Mahboubi, A., Tassi, E.: A small scale reflection extension for the Coq system. Research report RR-6455, Inria Saclay Ile de France (2016)
19. Harrison, J.: The HOL Light theory of Euclidean space. *J. Autom. Reason.* **50**, 173–190 (2013). <https://doi.org/10.1007/s10817-012-9250-9>
20. Jeannet, B., Miné, A.: APRON: a library of numerical abstract domains for static analysis. In: Bouajjani, A., Maler, O. (eds.) CAV 2009. LNCS, vol. 5643, pp. 661–667. Springer, Heidelberg (2009). [https://doi.org/10.1007/978-3-642-02658-4\\_52](https://doi.org/10.1007/978-3-642-02658-4_52)
21. Mahboubi, A., Tassi, E.: Canonical structures for the working Coq user. In: Blazy, S., Paulin-Mohring, C., Pichardie, D. (eds.) ITP 2013. LNCS, vol. 7998, pp. 19–34. Springer, Heidelberg (2013). [https://doi.org/10.1007/978-3-642-39634-2\\_5](https://doi.org/10.1007/978-3-642-39634-2_5)
22. Sakaguchi, K.: Vass (2016). <https://github.com/pi8027/vass>
23. Santos, F.: A counterexample to the Hirsch conjecture. *Ann. Math.* **176**(1), 383–412 (2012)
24. Schrijver, A.: Theory of Linear and Integer Programming. Wiley, New York (1986)
25. Smale, S.: Mathematical problems for the next century. *Math. Intell.* **20**(2), 7–15 (1998). <https://doi.org/10.1007/BF03025291>
26. Spasić, M., Marić, F.: Formalization of incremental simplex algorithm by stepwise refinement. In: Giannakopoulou, D., Méry, D. (eds.) FM 2012. LNCS, vol. 7436, pp. 434–449. Springer, Heidelberg (2012). [https://doi.org/10.1007/978-3-642-32759-9\\_35](https://doi.org/10.1007/978-3-642-32759-9_35)
27. Stein, W., et al.: Sage Mathematics Software (Version 9.0). The Sage Development Team (2020). <http://www.sagemath.org>
28. Ziegler, G.M.: Lectures on Polytopes. Springer, New York (1995). <https://doi.org/10.1007/978-1-4613-8431-1>