Convexity Recognition of the Union of Polyhedra

Alberto Bemporad [a] Komei Fukuda [b,c] Fabio D. Torrisi [a]

[a] *Automatic Control Laboratory, ETHZ, ETL I 26 CH-8092 Zurich, Switzerland*

[b] *Institute for Optimization Research, ETHZ, CLP D2 CH-8092 Zurich, Switzerland*

[c] *Dept. of Mathematics, EPFL, CH-1015 Lausanne, Switzerland*

Abstract

In this paper we consider the following basic problem in polyhedral computation: Given two polyhedra in $\mathbb{R}^d$, P and Q, decide whether their union is convex, and, if so, compute it. We consider the three natural specializations of the problem: (1) when the polyhedra are given by halfspaces (H-polyhedra) (2) when they are given by vertices and extreme rays (V-polyhedra) (3) when both H- and V-polyhedral representations are available. Both the bounded (polytopes) and the unbounded case are considered. We show that the first two problems are polynomially solvable, and that the third problem is strongly-polynomially solvable.

Key words: Polyhedron, Union, Convexity
PACS:

---

1 Introduction

A *convex polyhedron* is the intersection of a finite set of halfspaces of the Euclidean space $\mathbb{R}^d$, and a *convex polytope* is a bounded convex polyhedron. Every convex polyhedron has two natural representations, a halfspace representation (*H-polyhedron*) and a generator representation (*V-polyhedron*). We have seen in the recent years various new techniques of geometric computations associated with convex polyhedra. Since convex polyhedra arise frequently as

Email addresses: bemporad@aut.ee.ethz.ch (Alberto Bemporad), fukuda@ifor.math.ethz.ch (Komei Fukuda), torrisi@aut.ee.ethz.ch (Fabio D. Torrisi).

critical objects in fundamental problems of mathematical programming, computational geometry, statistics, material sciences, control engineering, etc., basic computational techniques for convex polyhedra often turn out to be extremely important for solving and analyzing the problems. Perhaps the best way to observe this new trend is to look at popular homepages on polyhedral/geometric computations [5,8] and an excellent handbook of Discrete and Computational Geometry [9]. Converting a representation of a convex polytope to another, triangulating a point set and removing redundancy from a system of linear inequalities are typical problems in polyhedral computation. We expect that a critical mass of state-of-the-art techniques for these fundamental problems will eventually crystallize as a foundation of polyhedral computation.

In this paper, we study the following polyhedral computation problem. Let P and Q be convex polyhedra in the d-dimensional Euclidean space $\mathbb{R}^d$. Our problem, in generic form, is to efficiently check whether their union $P \cup Q$ is convex, and if so, to find a minimal representation. A motivation for studying such a problem stems from the field of control engineering. In particular, in [4] the authors propose multiparametric quadratic programming to explicitly solve optimal control problems. The solution turns out to be piecewise linear over a polyhedral partition of the space of sensor measurements. In order to reduce the complexity of the solution, and hence allow cheaper control hardware, convexity recognition algorithms are successfully used to join polyhedral regions where the linear solution is the same.

The problem of convexity recognition of the union of polyhedra needs to be specified further depending on the forms of input and output. We shall consider three natural cases: (1) when both input and output are H-polyhedra, (2) when they are V-polyhedra and (3) when they are VH-polyhedra (i.e. given by both representations). We do not consider the case that input and output are of different forms. Such cases require the computation of a representation conversion that is a well-known fundamental problem and should be treated independently. We also investigate a natural extension of our problems to the union of k H-polyhedra for $k \geq 3$. While our problem of convexity recognition in R^d does not seem to have been investigated before, there are various studies on the union of polyhedra. Franklin [7] studied union and intersection operations of 2 and 3 dimensional nonconvex polyhedra. Aronov, Sharir and Tagansky [1] studied the complexity of the union of k convex 3-polyhedra and proposed a randomized algorithm for computing it. Balas [2] presented a formulation of a H-polyhedra that contains the convex hull of several H-polyhedra to deal with mixed integer 0/1 programs.

Our algorithms are based on simple characterizations of the convexity of the union of convex polyhedra, Theorem 3 for H-polyhedra and Theorem 4 for V-polyhedra, given in Section 3 and Section 5, respectively. It follows quite naturally from these theorems that the convexity recognition of the union of two H-(V-)polyhedra can be verified in polynomial time via linear programming [13]. For the H-polyhedra case, one can generalize Theorem 3 and show the polynomial solvability for any fixed number of H-polyhedra. We also present an extension of Theorem 4, Theorem 5, but we do not know whether it can be used to design an efficient algorithm, and thus the efficient solvability is still open for k V-polyhedra with $k \geq 3$. Finally we present a strongly polynomial algorithm for two VH-polyhedra.

2 Preliminaries on Convex Polyhedra

Let us recall some basic notions and theorems on convex polyhedra we shall use in the sequel.

For two subsets P and Q of $\mathbb{R}^d$, their *Minkowski's sum*, denoted by $P+Q$, is the set $\{x : x = p+q, p \in P, q \in Q\}$. For a finite subset $V = \{v_1, \ldots, v_n\}$ of $\mathbb{R}^d$, its convex hull $\mathrm{conv}(V)$ and the conic hull $\mathrm{cone}(V)$ are defined by

$$\mathrm{conv}(V) := \{x : x = \sum_{i=1}^{n} \lambda_i v_i, \lambda \geq 0, \sum_{i=1}^{n_1} \lambda_i = 1\}$$
$$\mathrm{cone}(V) := \{x : x = \sum_{i=1}^{n} \lambda_i v_i, \lambda \geq 0\}.$$

Here follow two fundamental results on convex sets and on convex polyhedra.

Theorem 1 (Carathéodory's Theorem [13, p. 94]) *If $A \subseteq \mathbb{R}^d$ and $x \in \mathrm{conv}(A)$ then exists $S \subseteq A$, such that $x \in \mathrm{conv}(S)$ and $|S| \leq d+1$.*

Theorem 2 (Motzkin's Theorem [13, p. 88], [14, p. 30]) *For a subset $P \subseteq \mathbb{R}^d$, the following two statements are equivalent:*

(a) *$P = \mathrm{conv}(V) + \mathrm{cone}(R)$ for some finite subsets V and R of $\mathbb{R}^d$;*
(b) *$P = \{x : \ Ax \leq \alpha\}$ for some matrix $A \in \mathbb{R}^{m \times d}$ and some vector $\alpha \in \mathbb{R}^m$.*

A subset P of $\mathbb{R}^d$ represented by either (a) or (b) is called a *convex polyhedron* or simply *polyhedron*. A *convex polytope* or simply *polytope* is a bounded convex polyhedron. A representation (V, R) of a convex polyhedron P is called a *V-representation*, and a representation (A, α) is an *H-representation*. A convex polyhedron given by V-representation (H-representation, both V- and H-representations) is called *V-polyhedron* (*H-polyhedron*, *VH-polyhedron*). We denote by $d(P)$ the *dimension* of P, the dimension of the affine span of P.

The following basic lemma, whose proof naturally follows from the definition

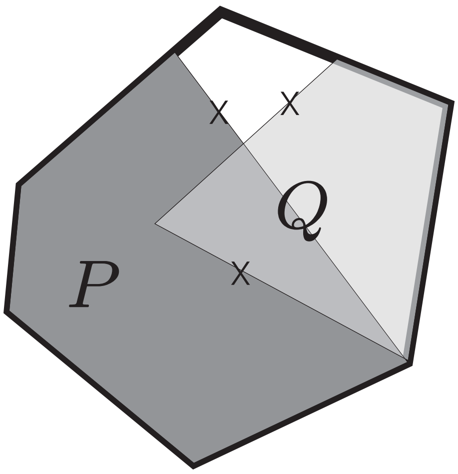


Fig. 1. Construction of the envelope $\mathrm{env}(P, Q)$. The removed inequalities are marked by $\times$

of convexity (see e.g. [3]), will be used in the sequel to prove our main results.

Lemma 1 *Let P and Q be convex polyhedra with V-representations (V, R) and (W, S), respectively. Then $P \cup Q$ is convex if and only if $P \cup Q$ is a convex polyhedron with V-representation $(V \cup W, R \cup S)$.*

Let P be a convex polyhedron in $\mathbb{R}^d$. For $a \in \mathbb{R}^d$ and $b \in \mathbb{R}$, the inequality $a^T x \leq b$ is called *valid* for P if it is satisfied by all points in P. The *faces* of P are the sets of form $P \cap \{x : a^T x = b\}$ for some valid inequality $a^T x \leq b$. We say the face $P \cap \{x : a^T x = b\}$ is *determined by the inequality* $a^T x \leq b$. The faces of dimension 0, 1 and $d(P) - 1$ are called *vertices*, *edges* and *facets*. A valid inequality $a^T x \leq b$ is said to be a *facet inequality* if it determines a facet, and a *linearity inequality* if it determines the polytope itself.

For an H-polyhedron $P = \{x : \ Ax \leq \alpha\}$, an ith inequality $A_i x \leq \alpha_i$ is said to be *redundant* for P if its removal preserves the polyhedron, i.e.

$$P = \{x : \ Ax \leq \alpha\} = \{x : \ A_j x \leq \alpha_i \text{ for all } j \neq i\}.$$

Note that redundancy is a relative notion and removing two redundant inequalities may not preserve the polyhedron. Every nonredundant inequality is either a facet inequality or a linearity inequality. If a facet inequality is redundant, then P contains another inequality that determines the same facet.

Finally, given a collection of n points V in $\mathbb{R}^d$, we say that $v \in V$ is *redundant* for V if $\mathrm{conv}(V) = \mathrm{conv}(V \setminus \{v\})$.

3 Key Theorem for H-Polyhedra

Let P and Q be (possibly unbounded) H-polyhedra,

$$P = \{x \in \mathbb{R}^d : \; Ax \leq \alpha\}, \tag{1}$$
$$Q = \{x \in \mathbb{R}^d : \; Bx \leq \beta\}. \tag{2}$$

We define the *envelope* of two H-polyhedra P and Q as the polyhedron defined by

$$\text{env}(P, Q) = \{x \in \mathbb{R}^d : \; \overline{A}x \leq \overline{\alpha}, \overline{B}x \leq \overline{\beta}\}, \tag{3}$$

where $\overline{A}x \leq \overline{\alpha}$ is the subsystem of $Ax \leq \alpha$ obtained by removing all the inequalities not valid for the other polyhedron Q, and $\overline{B}x \leq \overline{\beta}$ is defined similarly with respect to P. An example is shown in Fig. 1, where the removed inequalities are marked by $\times$. By definition, it is easy to see that

$$P \cup Q \subseteq \text{env}(P, Q). \tag{4}$$

Note that the set $\text{env}(P, Q)$ does not depend on the H-representations of P and Q.

Theorem 3 *$P \cup Q$ is convex $\Leftrightarrow$ $P \cup Q = \text{env}(P, Q)$.*

Proof of Theorem 3: The "$\Leftarrow$" part is trivial, as $\text{env}(P, Q)$ is a convex set by construction. In order to prove the "$\Rightarrow$" part, assume $K := P \cup Q$ is convex and without loss of generality P, Q are full dimensional. The case $\dim(P) = \dim(Q) < d$ can be reduced to the case of full dimensional polyhedra in the common embedding subspace. In the case $\dim(P) < \dim(Q)$, $P \cup Q$ convex implies $P \subset Q$, and therefore the proof is trivial.

Let (V, R) and (W, S) be a V-representation of P and Q, respectively. By Lemma 1, $K = \text{conv}(V \cup W) + \text{cone}(R \cup S)$ is a polyhedron, and by Theorem 2, has an H-representation. As, by (4), $K \subseteq \text{env}(P, Q)$, it is enough to prove $\text{env}(P, Q) \subseteq K$, by showing that all the inequalities in the unique minimal representation are already in the inequalities representing $\text{env}(P, Q)$. Suppose there is a facet inequality $r'x \leq s$ for K that is missing in the H-representation of $\text{env}(P, Q)$. Let $H = \{x \in \mathbb{R}^d : r'x = s\}$. Since the inequality is missing in the H-representations of P and Q, $\dim(P \cap H) \leq d-2$ and $\dim(Q \cap H) \leq d-2$ because it is valid for P and is not in $\text{env}(P, Q)$. This implies that the facet $K \cap H$ of K cannot be the union of two convex sets $P \cap H$ and $Q \cap H$, because they have smaller dimensions than $K \cap H$. This contradicts $K = P \cup Q$. ■

Remark 1 *Theorem 3 can be naturally generalized to the union of k polytopes, for any positive integer k.*

4 Algorithm for H-Polyhedra

Theorem 3 represents a result for convexity recognition of the union of two H-polyhedra. It also leads to an algorithm for checking convexity of the union, and for generating an H-representation of the union when it is convex.

Algorithm 4.1

1 Construct $\mathrm{env}(P, Q)$ by removing non-valid constraints (see Fig. 1),
let $\tilde{A}x \leq \tilde{\alpha}$, $\tilde{B}x \leq \tilde{\beta}$ be the set of removed constraints, and
let $\mathrm{env}(P, Q) = \{x : \; Cx \leq \gamma\}$ the resulting envelope;

2 Remove from $\mathrm{env}(P, Q)$ possible duplicates
$(B_j, \beta_j) = (\sigma A_i, \sigma \alpha_i)$, $\sigma > 0$;

3 **for** each pair $\tilde{A}_i x \leq \tilde{\alpha}_i$, $\tilde{B}_j x \leq \tilde{\beta}_j$ **do**

4 Determine ϵ^* by solving the linear program

$$\epsilon^* = \max_{(x,\epsilon)} \epsilon$$

$$\text{subj. to } \; \tilde{A}_i x \geq \tilde{\alpha}_i + \epsilon$$

$$\tilde{B}_j x \geq \tilde{\beta}_j + \epsilon$$

$$Cx \leq \gamma;$$

/* $\epsilon^* = -\infty$ if the LP is infeasible,
$\epsilon^* = \infty$ if the LP is unbounded */

5 **if** $\epsilon^* > 0$, **stop**; **return** nonconvex;

6 **endfor**;

7 **return** $\mathrm{env}(P, Q)$. /* $P \cup Q$ is convex. */

Note that if $\epsilon^* = 0$ for each i, j as defined in step *3*, then by Theorem 3 the union is convex and equals $\mathrm{env}(P, Q)$. On the other hand, $\epsilon^* > 0$ indicates the existence of a point $x \in \mathrm{env}(P, Q)$ outside $P \cup Q$. Note that $\epsilon^* > 0$ for the pairs of constraints marked by $\times$ in Fig. 1.

For recognizing convexity and computing the union of k polyhedra, the test can be modified by checking each k-tuple of removed constraints. Let $\tilde{m}_1$, …, $\tilde{m}_k$ be the number of removed constrains from the polyhedra $P_1, \ldots, P_k$, respectively. Then similarly to step *4*, $\prod_{i=1}^{k} \tilde{m}_i$ linear programs need to be solved in the general case.

Algorithm 4.1 provides an H-representation for $P \cup Q$ (step 1). We prove in next Proposition 1 that such representation is *minimal*, i.e. it contains no redundant inequalities.

Proposition 1 *If P and Q are given by minimal H-representation and are full-dimensional then Algorithm 4.1 outputs a minimal H-representation of $P \cup Q$.*

Proof: Suppose P and Q are d-dimensional and given in minimal H-representation. Take any inequality T given by the algorithm. We may assume it comes from the representation of P. By the minimality and full-dimensionality, it is a facet inequality for P. By definition, the facet F determined by T contains d affinely independent points of P. Since these points are also in $P \cup Q$, T is a facet inequality for $P \cup Q$. By the step 2 of the algorithm, there is no other inequality in the output that determines the same facet F. Therefore the output is minimal. ■

Note that the full dimensionality of P and Q are necessary in Proposition 1. In general, redundancies of a representation can be removed efficiently using standard algorithms based on linear programming, see e.g. [12].

Now, we evaluate the complexity of Algorithm 4.1. Clearly, the major computational burden comes from solving LPs. Since the time complexity of solving an LP depends on the choice of algorithm (e.g. interior-point methods, simplex methods, randomized methods [10, Chapter 39]), we consider an LP to be oracle and evaluate the complexity of a given algorithm by the maximum number of LPs that must be solved.

Namely, we denote by $\mathbf{lp}(m, d)$ a time complexity of solving an $m \times d$ canonical LP:

$$\max_x \; c^T x$$

$$\text{subj. to } Ax \leq b, x \geq \mathbf{0},$$

where $A \in \mathbb{R}^{m \times d}$, $c \in \mathbb{R}^d$ and $b \in \mathbb{R}^m$. Since its dual LP is a $d \times m$ canonical LP, we may assume that $\mathbf{lp}(d, m)$ is of the same order as $\mathbf{lp}(m, d)$.

The time complexity of solving an LP might depend on the length L of binary encodings of the input A, b and c, but we are neglecting this dependency since we believe that practically it hardly reflects the actual behavior of real implementations. In any case, one can read the function $\mathbf{lp}(m, d)$ as $\mathbf{lp}(m, d, L)$ when more appropriate.

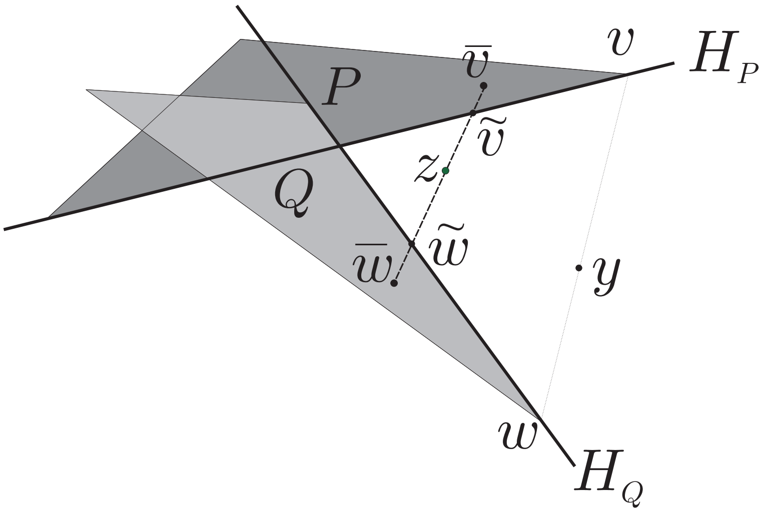


Fig. 2. Proof of Theorem 4

Note that there are polynomial algorithms for LP such as the ellipsoid method and interior-point methods, but they are *not strongly polynomial*, that is, the number of arithmetic operations needed to solve an LP by any such an algorithm is not polynomially bounded by m, d only (but depends on L). It is still an open problem whether or not there exists a strongly polynomial algorithm for LP.

Now we can evaluate the complexity of Algorithm 4.1.

Proposition 2 *Let the row size of A and B be m_1 and m_2, respectively. Then the time complexity of Algorithm 4.1 is $O(m_1 m_2 \mathbf{lp}(O(d, m_1 + m_2)))$.*

Proof: While step *1* requires $m_1 + m_2$ LPs to be solved, the hardest part of Algorithm 4.1 is step *4*. Each LP in this step can be reduced to a canonical LP of size $O((m_1+m_2)\times d)$. Since there are $O(m_1 m_2)$ LPs to be solved in the worst case, the time complexity of the algorithm is $O(m_1 m_2 \mathbf{lp}(O(d, m_1 + m_2)))$. ■

5 Key Theorem for V-Polyhedra

In this section we present simple criteria for recognizing the convexity of the union of two V-polyhedra. For simplicity of presentation, we assume that the given V-polyhedra are bounded, i.e. they are V-polytopes. The results can be easily generalized to the unbounded case, as shown in [3].

Theorem 4 *Let P, Q be polytopes with V-representation V and W, respec-*

tively. Then

$$P \cup Q \textit{ is convex} \Leftrightarrow [v, w] \subseteq P \cup Q,\ \forall v \in V,\ \forall w \in W. \tag{5}$$

Moreover, a stronger characterization of convexity holds

$$P \cup Q \textit{ is nonconvex} \Leftrightarrow \exists v \in V,\ w \in W \textit{ such that } (v, w) \cap (P \cup Q) = \emptyset. \tag{6}$$

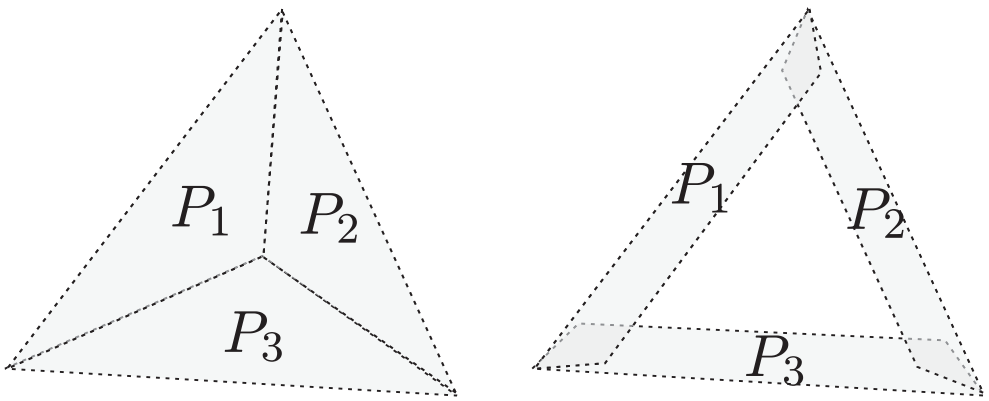
Proof: Let $Ax \le \alpha$ and $Bx \le \beta$ be H-representations of P, Q, respectively.

First we prove the statement (5). The implication ($\Rightarrow$) follows immediately by convexity. We prove the ($\Leftarrow$) part by contradiction. Suppose that there exist $\bar{v} \in P$, $\bar{w} \in Q$, and $\gamma \in (0,1)$ such that $z := \gamma\bar{v} + (1-\gamma)\bar{w} \notin P \cup Q$. It follows that $\bar{v} \notin Q$,$\bar{w} \notin P$. (See Fig. 2.) Since $z \notin P$, there exists a hyperplane $H_P := \{x :\ A_i x = \alpha_i\}$ such that the line segment $[z, \bar{v}]$ intersects H_P in some point $\tilde{v} \in P$, and $A_i z > \alpha_i$, where subscripting by i denotes the row. Denote by $H_P^+ := \{x :\ A_i x > \alpha_i\}$, and $H_P^- := \{x :\ A_i x < \alpha_i\}$. Similarly, there exists $H_Q := \{x :\ B_j x = \beta_j\}$ intersecting $[z, \bar{w}]$ in $\tilde{w} \in Q$, and $z \in H_Q^+$. By convexity, $(\tilde{v}, \bar{w}] \cap P = \emptyset$ and $(\tilde{w}, \bar{v}] \cap Q = \emptyset$, and in particular $\tilde{w} \in H_P^+$, $\tilde{v} \in H_Q^+$. As $\tilde{v} \in P \cap H_P$ lies in the halfspace H_Q^+, there exists a vertex v of $P \cap H_P$ in H_Q^+. Note that $v \in V$ because $P \cap H_P$ is a face of P. Similarly, there exists a vertex $w \in W$ such that $w \in H_P^+ \cap H_Q$. Consider now the polyhedron $T := \{x :\ A_i x \ge \alpha_i,\ B_j x \ge \beta_j\}$. Clearly, $v,\ w \in T$. By convexity of T, $[v, w] \subseteq T$, and either $(v, w) \cap \mathring{T} \neq \emptyset$, or $(v, w) \subseteq \partial T$, where ∂T denotes the boundary of T, and $\mathring{T}$ the interior, $\mathring{T} := T - \partial T$. In the first case, all points $y_\gamma = \gamma v + (1-\gamma) w$, $\gamma \in (0,1)$, $y_\gamma \in \mathring{T}$, and therefore $y_\gamma \notin P$, $y_\gamma \notin Q$, i.e. $y_\gamma \notin P \cup Q$, a contradiction. In the second case, $[v, w] \subseteq H_P$ or $[v, w] \subseteq H_Q$. In both cases, there is a contradiction, as $v \in H_Q^+$ and $w \in H_P^+$, respectively. As this holds for all $\gamma \in (0,1)$, the statement (6) is also proved. ■

The theorem can be generalized to the union of $k \ge 3$ polytopes. By referring to Fig. 3(a), a natural conjecture would be to consider the union V of the vertices of P_1, P_2, P_3, and check that for all $v_i, v_j \in V$, the line segment $[v_i, v_j]$ is contained in $P_1 \cup P_2 \cup P_3$. However, Fig. 3(b) shows that such a conjecture is false. Before introducing the generalized theorem, we prove the following lemma, which is a straightforward consequence of Carathéodory's theorem. For the remaining part of this section we assume that $d \ge 0$, V_i is a finite set of $\mathbb{R}^d$ for $i = 1, \ldots, k$, $P_i := \mathrm{conv}(V_i)$, $P := \cup_{i=1}^k P_i$, $V := \cup_{i=1}^k V_i$ and $\hat{P} := \mathrm{conv}(V)$. Note that each V_i is possibly empty, that is important for the inductive proof below.

Lemma 2 *Let* $P = \cup_{i=1}^k \mathrm{conv}(V_i)$ *and* $V := \cup_{i=1}^k V_i$. *Then*

$$P \textit{ is convex} \Leftrightarrow \forall S \subseteq V,\ |S| \le d+1,\ \mathrm{conv}(S) \subseteq P. \tag{7}$$



(a) Convex case (b) Nonconvex case

Fig. 3. Union of three V-polytopes

Proof: The "⇒" part is trivial. To prove the "⇐" part, we assume by contradiction that the RHS of (7) is satisfied but P is not convex. Then there exists $\bar{x} \in \hat{P} \setminus P$. Since $\bar{x} \in \text{conv}(V) = \hat{P}$, by Carathéodory's theorem, there exists $S \subseteq V$, $|S| \leq d+1$ such that $\bar{x} \in \text{conv}(S)$, contradicting the assumption. ■

Now we can state and prove the following theorem [1] for k V-polytopes.

Theorem 5 *Let* $P = \cup_{i=1}^{k} \text{conv}(V_i)$ *and* $V := \cup_{i=1}^{k} V_i$. *Then*

$$P \textit{ is convex} \Leftrightarrow \forall S \subseteq V,\ |S| \leq k,\ \text{conv}(S) \subseteq P. \tag{8}$$

Proof: The direction "⇒" is trivial.

To prove the direction "⇐", we proceed by induction on d. When $d = 0$, the implication is trivial. Assume by inductive hypothesis that the implication holds in dimension $d-1$ ($d \geq 1$), and we shall prove it in dimension d. Suppose by contradiction that there exists a counterexample, namely we suppose that the RHS of (8) is satisfied but there exists $\bar{x} \in \hat{P} \setminus P$. First we remark that $\dim(\hat{P}) = d$, since otherwise the counterexample can be embedded in a lower dimensional space contradicting the inductive hypothesis. Because $\bar{x} \notin P_i$ for all $i = 1, \ldots, k$, there exists a valid inequality $A_i x \leq \alpha_i$ for P_i, violated at $\bar{x}$. By Lemma 2, $k \leq d$ and thus the region $C = \{x : A_i x > \alpha_i, i = 1, \ldots, k\}$ is unbounded (see Fig. 4). Let ℓ be an unbounded direction, and let $\bar{x}'$ be the intersection of the line $L = \{\bar{x} + \lambda\ell, \lambda \geq 0\}$ and the boundary of $\hat{P}$, and let F be a facet of $\hat{P}$ containing $\bar{x}'$. Consider $V_i' := V_i \cap F$ and $P_i' := \text{conv}(V_i') = P_i \cap F$, where the last equality follows from the fact that F is a face of $\hat{P}$. Finally let $V' := V \cap F = (\cup V_i) \cap F = \cup(V_i \cap F) = \cup V_i'$, $P' := \cup P_i' = \cup(P_i \cap F) =$

[1] This is a joint work with L. Finschi [6].

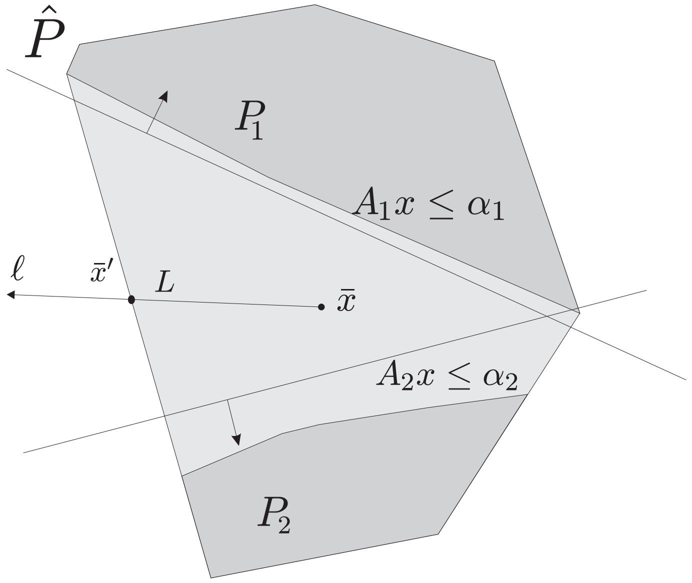


Fig. 4. Proof of Theorem 5

$(\cup P_i) \cap F = P \cap F$ and $\hat{P}' := \mathrm{conv}(V')$. Clearly $\mathrm{conv}(S') \subseteq P \; \forall S' \subseteq V'$, $|S'| \leq k$, as $V' \subseteq V$. Moreover $\mathrm{conv}(S') = \mathrm{conv}(S') \cap F \subseteq P \cap F = P'$ and F has dimension $\dim(F) = d - 1$. Thus the RHS of (8) is satisfied for P' in dimension $d - 1$. On the other hand, P' is not convex, because $\bar{x}' \in \hat{P}' \setminus P'$. This is a contradiction to the inductive hypothesis. ■

6 Algorithms for V-Polyhedra

In this section we present two algorithms for convexity recognition and computation of the union of two V-polyhedra. Unfortunately, it is not clear how to make use of Theorem 5 for efficient convexity recognition for tree or more polyhedra. For the sake of simplicity, we consider the bounded case only.

Let P and Q be V-polytopes with given representation $V = \{v_1, \ldots, v_{n_1}\}$ and $W = \{w_1, \ldots, w_{n_2}\}$, respectively. We present two algorithms based on the two characterizations of the convexity of $P \cup Q$ in Theorem 4. The first algorithm makes use of a "ray-shooting" technique in order to detect points z on line-segments $[v_i, w_j]$ which do not belong to the union $P \cup Q$.

Consider all pairs of vertices $v_i \in V$, $w_j \in W$, and let $\ell := (w_j - v_i)$, as shown in Fig. 6. Consider the problem of finding the "last" point $z \in P$ on the ray $v_i + \lambda_0 \ell$, $\lambda_0 \geq 0$, such that λ_0 is maximized. This can be computed by the

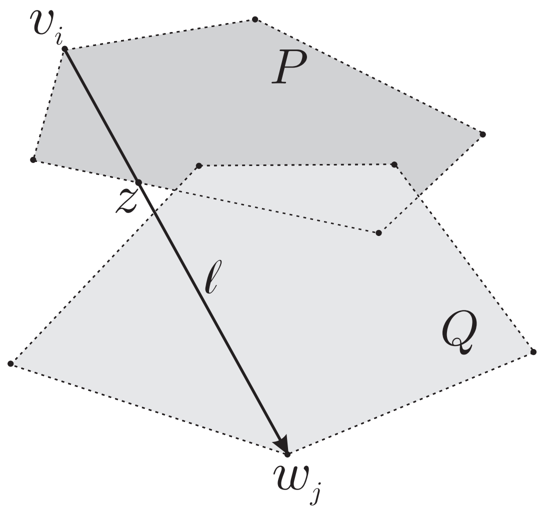


Fig. 5. Ray-shooting Algorithm for V-polytopes

following linear program (LP)

$$
\begin{aligned}
\lambda_0^* := \max_{\lambda_0,\ldots,\lambda_{n_1}} \ & \lambda_0 \\
\text{subj. to} \quad & v_i + \lambda_0 \ell = \textstyle\sum_{i=1}^{n_1} \lambda_i v_i \\
& \textstyle\sum_{i=1}^{n_1} \lambda_i = 1 \\
& \lambda_i \geq 0, \ i = 0, \ldots, n_1
\end{aligned}
\tag{9}
$$

Let $z := v_i + \lambda_0^* \ell$. If $z \notin Q$, then $P \cup Q$ is nonconvex. This is proved by the following corollary of Theorem 4.

Corollary 1 *$P \cup Q$ is convex $\Leftrightarrow$ $\forall$ pair v_i, w_j of vertices of P, Q, respectively, the vector $v_i + \lambda_0^*(w_j - v_i) \in Q$, where λ_0^* is determined by the LP in (9).*

The condition $z \in Q$ can be checked via the following LP feasibility test:

$$z = \sum_{i=1}^{n_2} \mu_j w_j, \ \sum_{i=1}^{n_2} \mu_j = 1, \ \mu_j \geq 0, \ j = 0, \ldots, n_2. \tag{10}$$

The LP problem (9) does not need to be solved for vertices $\tilde{v}_i \in Q$ and $\tilde{w}_j \in P$, as by convexity of P and Q the resulting vector z would be in $P \cup Q$. Therefore, it is useful to detect such vertices and apply the LP in (9) only for the remaining ones. Such a detection is solved by the LP feasibility test. Note that although determining $\tilde{v}_i$ costs an LP, it avoids solving the LP in (9) for all pairs $\tilde{v}_i$, w_j. Moreover, as a by-product, the V-representation of $P \cup Q$ is given by $(V \backslash \{\tilde{v}_i\}) \cup (W \backslash \{\tilde{w}_j\})$. In summary, the algorithm to determine the convexity of the union of convex polyhedra P, Q given in V-representation, and eventually the V-representation of $P \cup Q$, is the following:

Algorithm 6.1

1 Determine vertices $\tilde{v}_i \in Q$, $i = 1, \ldots, k_1$, and vertices $\tilde{w}_j \in P$, $k = 1, \ldots, k_2$, by solving LP feasibility tests;
2 **let** $\bar{V} := V \backslash \{\tilde{v}_i\}_{i=1}^{k_1}$, $\bar{W} := W \backslash \{\tilde{w}_j\}_{j=1}^{k_2}$;
3 **for** each pair $v_i \in \bar{V}$, $w_j \in \bar{W}$ **do**
4   Find the corresponding vector $z := v_i + \lambda_0^*(w_j - v_i)$ by solving the LP (9);
5   Determine if $z \in Q$ via the feasibility test (10);
6   **if** $z \notin Q$ **then stop**;**return** nonconvex;
7 **endfor**;
8 **let** X be the set of points in $V \cap W$ that are extreme in $P \cup Q$;
9 **stop**; **return** $(\bar{V} \cup \bar{W} \cup X)$.   /* $P \cup Q$ is convex. */

The algorithm above can be made more efficient by switching P and Q whenever $n_1 < n_2$, so that (10) is a feasibility test over n_1 variables.

Remark 2 *At the step 1, all vertices in $V \cap W$ will be removed. These vertices might be vertices of $P \cup Q$. To obtain a minimal V-representation, it is necessary to check whether each such vertex is extreme in $P \cup Q$ by an LP in the step 8.*

Algorithm 6.1 provides a V-representation for $P \cup Q$. We prove in next Proposition 3 how to obtain a *minimal* V-representation.

Proposition 3 *Let P and Q be V-polytopes given by minimal V-representation, and let $P \cup Q$ be convex. Then the set of vectors $\bar{V} \cup \bar{W} \cup X$ determined by Algorithm 6.1 is a minimal representation of $P \cup Q$.*

Proof: Suppose V and W are minimal V-representations. In general, every vertex of P that does not belong to Q is a vertex of $P \cup Q$. This means all points in $\bar{V} \cup \bar{W}$ are extreme in $P \cup Q$. Since the vertices of $P \cup Q$ are contained in $V \cup W$, the remaining vertices of $P \cup Q$ must be in $V \cap W$. When $P \cup Q$ is convex, the algorithm must process step *8* and thus the output is complete and minimal. ■

Proposition 4 *The time complexity of Algorithm 6.1 is $O(n_1 n_2(\mathbf{lp}(O(d, n_1)) + \mathbf{lp}(O(d, n_2))))$.*

Proof: It is easy to see that the for-loop starting the step *3* is dominating

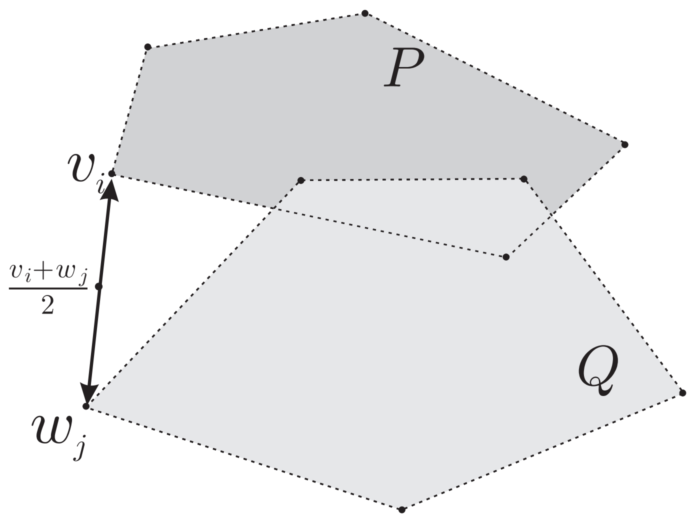


Fig. 6. Midpoint Algorithm for V-polytopes

other parts in terms of time complexity. This loop cycles $O(n_1 n_2)$ times in the worst case. There are two LPs to be solved in each loop. The first one, LP (9), is reducible to a canonical LP of size $O(d \times n_1)$. The second LP, the feasibility test (10), can be reduced a canonical LP of size $O(d \times n_2)$. Thus the claimed complexity follows. ■

Next we shall present a second algorithm that exploits the stronger characterization (6): $P \cup Q$ is nonconvex if and only if there exists an open segment (v_i, w_j) which lies totally outside $P \cup Q$. This characterization implies a more practical equivalence: $P \cup Q$ is nonconvex if and only if there exists an open segment (v_i, w_j) whose midpoint $z := \frac{v_i + w_j}{2}$ is not in $P \cup Q$. This yields the following algorithm.

Algorithm 6.2

```
Remove vertices of P which are in Q, and vice-versa, and
let V̄, W̄ the sets of remaining vertices;
for each pair v_i ∈ V̄, w_j ∈ W̄ do
   let z := (v_i + w_j)/2;
   Determine if z ∈ P ∪ Q via an LP feasibility test
   similar to (10);
   if z ∉ P ∪ Q, then stop; return nonconvex;
endfor;
let X be the set of points in V ∩ W that are extreme in P ∪ Q;
stop; return (V̄ ∪ W̄ ∪ X).    /* P ∪ Q is convex. */
```

The same argument observed in Remark 2 can be repeated for Algorithm 6.2. The output is a minimal V-representation of $P \cup Q$ and the proof is essentially the same as for Proposition 3.

Proposition 5 *The time complexity of Algorithm 6.2 is* $O(n_1 n_2(O(\mathbf{lp}(d, n_1)) + O(\mathbf{lp}(d, n_2))))$.

Proof: One can easily see that the for-loop starting step *2* is dominating other parts in terms of time complexity. This loop cycles $O(n_1 n_2)$ times in the worst case. There are two LPs to be solved in each loop (to check if $z \in P \cup Q$ is enough to check if $z \in P$ and $z \in Q$). This LPs are similar to the feasibility test (10), thus the claimed complexity follows. ■

Even though the two algorithms have the same complexity, the ray-shooting Algorithm 6.1 might stop earlier, depending on how likely the union is convex. In fact, the algorithm stops as soon as a point on (v_i, w_j) is outside $P \cup Q$, while Algorithm 6.2 might try several pairs before finding one such that the midpoint $(v_i + w_j)/2$ lies outside $P \cup Q$. On the other hand, as it will be explained in Section 7, Algorithm 6.2 can be efficiently modified when both V- and H-representation for P and Q are available.

We remark that if the polytopes have a convex union, all the pairs will be tested.

7 Algorithm for VH-Polyhedra

In this section, we present a strongly polynomial algorithm for the convexity recognition of $P \cup Q$ when both V- and H-representations of P and Q are given:

$$P = \mathrm{conv}(V) = \{x : Ax \le \alpha\} \tag{11}$$
$$Q = \mathrm{conv}(W) = \{x : Bx \le \beta\}. \tag{12}$$

First of all, the next theorem follows immediately from Theorem 3 and Lemma 1.

Theorem 6 *Let P and Q be VH-polytopes in $\mathbb{R}^d$ Then, if $P \cup Q$ is convex,* $\mathrm{conv}(V \cup W) = \mathrm{env}(P, Q) = P \cup Q$.

Note that the converse is not true, as shown in Fig. 7. Nevertheless, assuming that P and Q are given by VH-representations, $(V, Ax \le \alpha)$ and $(W, Bx \le \beta)$, we can design an efficient algorithm by modifying Algorithm 6.2. For the sake of simplicity, we consider polytopes, it is not hard to extend the algorithm to the general case.

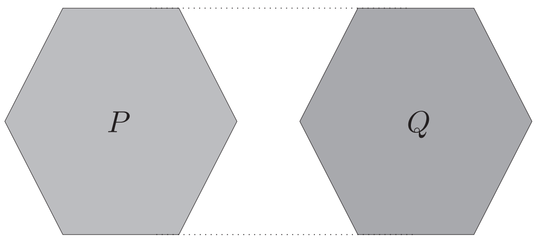


Fig. 7. Counterexample for the converse of Theorem 6

Algorithm 7.1

1 Determine vertices $v_i \in V \cap P$ which are also in Q, and vice versa vertices $w_j \in W \cap P$, by checking fulfillment of the inequalities $Aw_j \leq \alpha$, $Bv_i \leq \beta$;

2 Find inequalities in P which are not valid for Q, and vice versa, by checking if $\exists j$ such that $A_i w_j > \alpha_i$, and $\exists i$ such that $B_j v_i > \beta_j$;

3 Remove vertices found in *1* and inequalities found in *2*; **let** $\bar{V}$, $\bar{W}$ the sets of remaining vertices, and $(\bar{A}, \bar{\alpha})$, $(\bar{B}, \bar{\beta})$ the set of remaining inequalities with possible duplications eliminated;

4 **for** each pair $v_i \in \bar{V}$, $w_j \in \bar{W}$ **do**

5   **let** $z := \frac{v_i + w_j}{2}$;

6   determine if $z \notin P \cup Q$ by checking if $\exists i, j$ such that $A_i z > \alpha_i$ and $B_j z > \beta_j$;

7   **if** yes, **then stop**; **return** nonconvex;

8 **endfor**;

9 **let** X be the set of points in $V \cap W$ that are extreme in $P \cup Q$, without duplicates;

10 **return** $(\bar{V} \cup \bar{W} \cup X)$,$\{x \in \mathbb{R}^d : \ \bar{A}x \leq \bar{\alpha}, \ \bar{B}x \leq \bar{\beta}\}$.
    /* $P \cup Q$ is convex. */

Step *9* requires some explanation. As for the algorithms discussed in earlier sections, this step is necessary for the minimality of the output V-representation. In case of VH-polytopes, this checking is much easier.

An inequality is said to be *active at x* if it is satisfied by x with equality. A point $y \in V \cup W$ is extreme in $P \cup Q$ if and only if the active set $A(y)$ of inequalities in $\{\bar{A}x \leq \bar{\alpha}, \ \bar{B}x \leq \bar{\beta}\}$ at y is maximal over $V \cup W$, i.e. there is no other point $y' \in V \cup W$ with $A(y) \subsetneq A(y')$.

Proposition 6 *The time complexity of Algorithm 7.1 is $O(n_1 n_2 d(m_1 + m_2))$.*

Proof: Step *1* and Step *2* require $O((n_1 m_2 + n_2 m_1)d)$ time, Step *9* requires $O(\min(n_1, n_2)d(m_1+m_2))$ time. The for-loop of Step *4* repeats at most $O(n_1 n_2)$ times. Each iteration of the loop costs $O((m_1 + m_2)d)$ time. This loops clearly dominates the others including the time for Step *9*, and thus the claimed complexity follows. ■

Acknowledgements

The authors thank the referees for their careful reading and many constructive comments.

References

[1] B. Aronov, M. Sharir, and B. Tagansky. The union of convex polyhedra in three dimensions. *SIAM J. Comput.*, 26:1670–1688, 1997.

[2] E. Balas. On the convex hull of the union of certain polyhedra. *Operations Research Letters*, 7:279–283, 1988.

[3] A. Bemporad, K. Fukuda, and F.D. Torrisi. Convexity recognition of the union of polyhedra. Technical Report AUT00-13, Automatic Control Laboratory, ETH Zurich, Switzerland, 2000. `http://control.ethz.ch/`.

[4] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. In *Proc. American Contr. Conf.*, pages 872–876, Chicago, IL, June 2000.

[5] J. Erickson. Computational geometry pages, list of software libraries and codes. `http://compgeom.cs.uiuc.edu/~jeffe/compgeom/`.

[6] L. Finschi. Private communication, June 2000.

[7] Wm. R. Franklin. Efficient polyhedron intersection and union. In *Proc. Graphics Interface*, pages 73–80, 1982.

[8] K. Fukuda. Polyhedral computation FAQ. Both html and ps versions available from `http://www.ifor.math.ethz.ch/~fukuda/fukuda.html`.

[9] J.E. Goodman and J. O'Rourke (eds.). *Handbook of Discrete and Computational Geometry.* CRC Press, 1997.

[10] J.E. Goodman and Joseph O'Rourke (Eds.). *Handbook of Discrete and Computational Geometry.* Discrete Mathematics and Its Applications. CRC Press, New York, 1997.

[11] B. Grünbaum. *Convex Polytopes*, volume 16 of *Pure and Applied Mathematics.* Interscience Publishers John Wiley & Sons, Inc., New York, 1967.

[12] Th. Ottmann, S. Schuierer, and S. Soundaralakshmi. Enumerating extreme points in higher dimensions. In E.W. Mayer and C. Puech, editors, *STACS 95: 12th Annual Symposium on Theoretical Aspects of Computer Science*, Lecture Notes in Computer Science 900, pages 562–570. Springer-Verlag, 1995.

[13] A. Schrijver. *Theory of Linear and Integer Programming.* Wiley, 1986.

[14] G. M. Ziegler. *Lectures on Polytopes*, volume 152 of *Graduate Texts in Mathematics.* Springer, New York, 1994.