The Negative Cycles Polyhedron and Hardness of Checking Some Polyhedral Properties

Endre Boros* Khaled Elbassioni† Vladimir Gurvich‡
Hans Raj Tiwary§

Abstract

Given a graph $G = (V, E)$ and a weight function on the edges $w : E \mapsto \mathbb{R}$, we consider the polyhedron $P(G, w)$ of negative-weight flows on G, and get a complete characterization of the vertices and extreme directions of $P(G, w)$. Based on this characterization, and using a construction developed in [11], we show that, unless $P = NP$, there is no output polynomial-time algorithm to generate all the vertices of a 0/1-polyhedron. This strengthens the NP-hardness result of [11] for non 0/1-polyhedra, and comes in contrast with the polynomiality of vertex enumeration for 0/1-polytopes [8]. As further applications, we show that it is NP-hard to check if a given integral polyhedron is 0/1, or if a given polyhedron is half-integral. Finally, we also show that it is NP-hard to approximate the maximum support of a vertex a polyhedron in $\mathbb{R}^n$ within a factor of $12/n$.

Keywords: Flow polytope, 0/1-polyhedron, vertex, extreme direction, enumeration problem, negative cycles, directed graph, half-integral polyhedra, maximum support, hardness of approximation.

1 Introduction

A convex polyhedron $P \subseteq \mathbb{R}^n$ is the the intersection of finitely many halfspaces, determined by the *facets* of the polyhedron. A *vertex* or an *extreme point* of P is a point $v \in \mathbb{R}^n$ which cannot be represented as a convex combination of two other points of P, i.e., there exists no $\lambda \in (0, 1)$ and $v_1, v_2 \in P$ such that $v = \lambda v_1 + (1-\lambda)v_2$. A *direction* of P is a vector $d \in \mathbb{R}^n$ such that $x_0 + \mu d \in P$ whenever $x_0 \in P$ and $\mu \geq 0$. An *extreme direction* of P is a direction d that cannot be written as a conic combination of two other directions, i.e., there exist no non-negative numbers $\mu_1, \mu_2 \in \mathbb{R}_+$ and directions d_1, d_2 of P such that $d = \mu_1 d_1 + \mu_2 d_2$. Denote respectively by $\mathcal{V}(P)$ and $\mathcal{D}(P)$ the sets of extreme points and directions of polyhedron P. A bounded polyhedron, i.e., one for which $\mathcal{D}(P) = \emptyset$ is called a *polytope*.

*RUTCOR, Rutgers University, 640 Bartholomew Road, Piscataway NJ 08854-8003; (boros@rutcor.rutgers.edu)

†Max-Planck-Institut für Informatik, Saarbrücken, Germany; (elbassio@mpi-sb.mpg.de)

‡RUTCOR, Rutgers University, 640 Bartholomew Road, Piscataway NJ 08854-8003; (gurvich@rutcor.rutgers.edu)

§Universität des Saarlandes, Saarbrücken, D-66123 Germany; (hansraj@cs.uni-sb.de)

The well-known Minkowski-Weyl theorem states that any convex polyhedron can be represented as the Minkowski sum of the convex hull of the set of its extreme points and the conic hull of the set of its extreme directions (see e.g. [16]). Furthermore, for pointed polyhedra, i.e., those that do not contain lines, this representation is unique. Given a polyhedron P by its facets, obtaining the set $\mathcal{V}(P) \cup \mathcal{D}(P)$, required by the other representation, is a well-known problem, studied in the literature under different (but polynomially equivalent) forms, e.g. the *vertex enumeration* problem[7], the *convex hull* problem [2] or the *polytope-polyhedron problem* [12]. Clearly, the size of the extreme set $\mathcal{V}(P) \cup \mathcal{D}(P)$ can be (and typically is) exponential in n and the number of facets m, and thus when we consider the computational complexity of the vertex enumeration problem, we are interested in *output-sensitive* algorithms, i.e., whose running time depends not only on n, m, but also on $|\mathcal{V}(P) \cup \mathcal{D}(P)|$. Alternatively, we may consider the following, polynomially equivalent, decision variant of the problem:

Dec$(\mathcal{C}(P), \mathcal{X})$**:** Given a polyhedron P, represented by a system of linear inequalities, and a subset $\mathcal{X} \subseteq \mathcal{C}(P)$, is $\mathcal{X} = \mathcal{C}(P)$?

In this description, $\mathcal{C}(P)$ could be either $\mathcal{V}(P)$, $\mathcal{D}(P)$, or $\mathcal{V}(P) \cup \mathcal{V}(P)$. It is well-known and also easy to see that the decision problems for $\mathcal{D}(P)$ or for $\mathcal{V}(P) \cup \mathcal{D}(P)$ are equivalent to that for $\mathcal{V}(P')$ where P' is some polytope derived from P. It is also well-known that if the decision problem is NP-hard, then no output polynomial-time algorithm can generate the elements of the set $\mathcal{C}(P)$ unless P=NP (see e.g. [6]).

The complexity of some interesting restrictions of these problems have already been settled. Most notably, it was shown in [8], that in the case of 0/1-polytopes, i.e., for which $\mathcal{V}(P) \subseteq \{0, 1\}^n$, the problem of finding the vertices given the facets can be solved with polynomial delay (i.e. the time to produce each vertex is bounded by a polynomial in the input size) using a simple backtracking algorithm. Output polynomial-time algorithms also exist for enumerating the vertices of simple and simplicial polytopes [3, 4, 7], network polyhedra and their duals [14], and some other classes of polyhedra [1]. More recently, it was shown in [11] that for general unbounded polyhedra problem $\mathrm{Dec}(\mathcal{V}(P), \mathcal{X})$ of generating the vertices of a polyhedron P is NP-hard. On the other hand, for special classes of 0/1-polyhedra, e.g. the polyhedron of s-t-cuts in general graphs [10], the polyhedra associated with the incidence matrix of bipartite graphs, and the polyhedra associated with 0/1-network matrices [6], the vertex enumeration problem can be solved in polynomial time using problem-specific techniques. This naturally raises the question whether there exists a general polynomial-time algorithm for the vertex enumeration of such polyhedra, extending the result of [8] for 0/1-polytopes. Here, we show that this is only possible if P=NP. Our result strengthens that in [11], which did not apply to 0/1-polyhedra, and uses almost the same construction, but goes through the characterization of the vertices of the polyhedron of negative weight-flows of a graph, defined in the next section. We show that this polyhedron could be highly unbounded, by also characterizing its extreme directions, and leave open the hardness of enumerating these directions, with its immediate consequences on the hardness of vertex enumeration for polytopes.

Ding, Feng and Zang [9] have recently shown, among other results, that distinguishing whether a polyhedron, given by its facets, is either a 0/1 polyhedron or fractional, is an NP-hard problem. Their construction uses a polyhedron with exponentially many 0/1-vertices, and a constraint matrix having exactly two ones per column. In contrast, we shall see that our construction gives a similar result, but with the further restriction that the polyhedron is *integral*, has only polynomially many 0/1-vertices, and the constraint matrix is of the form $\left[\frac{A}{r}\right]$, where A is a totally unimodular matrix having at most one "+1" and one "−1" per column, and r is a row of $\pm 1'$s.

Another consequence of our construction is that checking if a polyhedron is *half-integral*, i.e., if all the vertices have components in $\{0, 1, 1/2\}$ is an NP-hard problem.

For a polyhedron $P = \{x \in \mathbb{R}^n : \ Ax = b, \ x \geq 0\}$, the support of a vertex $x \in \mathcal{V}(P)$ is defined as the number of positive components of x. Finding a vertex of a polyhedron with maximum support includes several interesting problems, such as MAX-CUT in undirected graphs, and LONGEST-CYCLE in directed graphs. It follows from our construction that is it NP-hard to approximate such maximum support within a factor bigger than $12/n$.

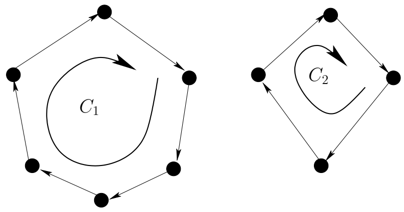
2 The polyhedron of negative-weight flows

Given a directed graph $G = (V, E)$ and a weight function $w : E \to \mathbb{R}$ on its arcs, consider the following polyhedron:

$$P(G,w) = \left\{ y \in \mathbb{R}^E \;\middle|\; \begin{array}{lrcll} (F) & \displaystyle\sum_{v:(u,v)\in E} y_{uv} - \sum_{v:(v,u)\in E} y_{vu} & = & 0 & \forall \; u \in V \\ (N) & \displaystyle\sum_{(u,v)\in E} w_{uv} y_{uv} & = & -1 & \\ & y_{uv} \geq 0 & & & \forall \; (u,v) \in E \end{array} \right\}.$$

If we think of $w_{u,v}$ as the cost/profit paid for edge (u, v) per unit of flow, then each point of $P(G, w)$ represents a *negative-weight circulation* in G, i.e., assigns a non-negative flow on the arcs, obeying the *conservation of flow* at each node of G, and such that total weight of the flow is strictly negative.

A negative- (respectively, positive-, or zero-) weight cycle in G is a directed cycle whose total weight is negative (respectively, positive, or zero). We represent a cycle C by the subset of arcs appearing on the cycle, and denote by $V(C)$ the nodes of G on the cycle (we assume all cycles considered to be directed and simple). Let us denote the families of all negative, positive, and zero-weight cycles of G by $\mathcal{C}^-(G, w)$, $\mathcal{C}^+(G, w)$, and $\mathcal{C}^0(G, w)$, respectively. Define a *2-cycle* to be a pair of cycles (C_1, C_2) such that $C_1 \in \mathcal{C}^-(G, w), C_2 \in \mathcal{C}^+(G, w)$ and $C_1 \cup C_2$ does not contain any other cycle of G. It is not difficult to see that a 2-cycle is either the edge-disjoint union of a negative cycle C_1 and a positive cycle C_2, or the edge-disjoint union of 3 paths P_1, P_2 and P_3 such that $C_1 = P_1 \cup P_2$ is a negative cycle, and $C_2 = P_1 \cup P_3$ is a positive cycle (see Figure 1). In the next section, we show that the set of vertices $\mathcal{V}(P(G, w))$ are in one-to-one correspondence with the set of negative cycles $\mathcal{C}^-(G, w)$, while the set of extreme directions $\mathcal{D}(P(G, w))$ is in one-to-one correspondence with the set $\mathcal{C}^0(G, w) \cup \{(C, C') \; : \; (C, C') \text{ is a 2-cycle}\}$.



(a)

Figure 1: 2-cycle.

3 Characterization of vertices and extreme directions of $P(G, w)$

For a subset $X \subseteq E$, and a weight function $w : E \mapsto \mathbb{R}$, we denote by $w(X) = \sum_{e \in X} w_e$, the total weight of X. For $X \subseteq E$, we denote by $\chi(X) \in \{0, 1\}^E$ the characteristic vector of X: $\chi_e(X) = 1$ if and only if $e \in X$, for $e \in E$.

Theorem 1 *Let $G = (V, E)$ be a directed graph and $w : E \rightarrow \mathbb{R}$ be a real weight on the arcs. Then*

$$\mathcal{V}(P(G, w)) = \left\{ \frac{-1}{w(C)} \chi(C) : \ C \in \mathcal{C}^-(G, w) \right\}, \tag{1}$$

$$\mathcal{D}(P(G, w)) = \mathcal{D}_1 \cup \mathcal{D}_2, \tag{2}$$

where

$$\mathcal{D}_1 = \{ \frac{1}{|C|} \chi(C) : \ C \in \mathcal{C}^0(G, w) \},$$
$$\mathcal{D}_2 = \{ \mu_{C_1,C_2} \chi(C_1) + \mu'_{C_1,C_2} \chi(C_2) \ : \ (C_1, C_2) \text{ is a 2-cycle} \},$$

and

$$\mu_{C_1,C_2} = \frac{w(C_2)}{w(C_2)|C_1| - w(C_1)|C_2|}, \ \mu'_{C_1,C_2} = \frac{-w(C_1)}{w(C_2)|C_1| - w(C_1)|C_2|}.$$

are non-negative numbers computed from cycles C_1 and C_2.

Proof. Let $m = |E|$ and $n = |V|$. We first prove (1). It is easy to verify that any element $y \in \mathbb{R}^E$ of the set on the right-hand side of (1) belongs to $P(G, w)$. Moreover, any such $x = -\chi(C)/w(C)$, for a cycle C, is a vertex of $P(G, w)$ since there are m linearly independent inequalities of $P(G, w)$ tight at x, namely: the conservation of flow equations at $|C| - 1$ vertices of C, the equation $\sum_{e \in C} w_e y_e = -1$, and $m - |C|$ equations $y_e = 0$, for $e \in E \setminus C$.

To prove the opposite direction, let $y \in \mathbb{R}^E$ be a vertex of $P(G, w)$. Let $Y = \{e \in E : \ y_e > 0\}$. The proof follows from the following 3 claims.

Claim 1 *The graph (V, Y) is the disjoint union of strongly connected components.*

Proof. Consider an arbitrary strongly connected component X in this graph, and let X^- be the set of components reachable from X (including X). Summing the conservation of flow equations corresponding to all the nodes in X^- implies that all arcs going out of X^- have a flow of zero. $\square$

Claim 2 *There exists no cycle $C \in \mathcal{C}^0(G, w)$ such that $C \subseteq Y$.*

Proof. If such a C exists, we define two points y' and y'' as follows.

$$y'_e = \begin{cases} y_e + \epsilon, & \text{if } e \in C \\ y_e, & \text{otherwise,} \end{cases} \qquad y''_e = \begin{cases} y_e - \epsilon, & \text{if } e \in C \\ y_e, & \text{otherwise,} \end{cases}$$

for some sufficiently small $\epsilon > 0$. Then y', y'' clearly satisfy (F). Moreover, (N) is satisfied with y' since

$$\sum_{e \in E} w_e y'_e = \sum_{e \notin C} w_e y_e + \sum_{e \in C} w_e (y_e + \epsilon) = \sum_{e \in E} w_e y_e + w(C)\epsilon = -1.$$

Similarly for y''. Thus $y', y'' \in P(G, w)$ and $y = (y' + y'')/2$ contradicting that y is a vertex. $\square$

Claim 3 *There exist no distinct cycles $C_1, C_2 \in \mathcal{C}^-(G, w) \cup \mathcal{C}^+(G, w)$ such that $C_1 \cup C_2 \subseteq Y$.*

Proof. If such C_1 and C_2 exist, we define two points y' and y'' as follows.

$$y'_e = \begin{cases} y_e + \epsilon_1, & \text{if } e \in C_1 \setminus C_2 \\ y_e + \epsilon_2, & \text{if } e \in C_2 \setminus C_1 \\ y_e + \epsilon_1 + \epsilon_2, & \text{if } e \in C_1 \cap C_2 \\ y_e, & \text{otherwise,} \end{cases} \qquad y'_e = \begin{cases} y_e - \epsilon_1, & \text{if } e \in C_1 \setminus C_2 \\ y_e - \epsilon_2, & \text{if } e \in C_2 \setminus C_1 \\ y_e - \epsilon_1 - \epsilon_2, & \text{if } e \in C_1 \cap C_2 \\ y_e, & \text{otherwise,} \end{cases}$$

where $\epsilon_1 = -\frac{w(C_2)}{w(C_1)}\epsilon_2$, for some sufficiently small $\epsilon_2 > 0$ (in particular, to insure non-negativity of y', y'', ϵ_2 must be upper bounded by the minimum of $\min\{y_e : e \in C_2 \setminus C_1\}$, $\frac{|w(C_1)|}{|w(C_2)|}\min\{y_e : e \in C_1 \setminus C_2\}$, and $\frac{|w(C_1)|}{|w(C_1) - w(C_2)|}\min\{y_e : e \in C_1 \cap C_2\}$). Then it is easy to verify that y', y'' satisfy (F). Moreover, (N) is satisfied with y'since

$$\begin{aligned} \sum_{e \in E} w_e y'_e \;\; &= \sum_{e \notin C_1 \cup C_2} w_e y_e + \sum_{e \in C_1 \setminus C_2} w_e (y_e + \epsilon_1) + \sum_{e \in C_2 \setminus C_1} w_e (y_e + \epsilon_2) \\ &\quad + \sum_{e \in C_1 \cap C_2} w_e (y_e + \epsilon_1 + \epsilon_2) = \sum_{e \in E} w_e y_e + w(C_1)\epsilon_1 + w(C_2)\epsilon_2 = -1. \end{aligned}$$

Similarly for y''. Thus $y', y'' \in P(G, w)$ and $y = (y' + y'')/2$ contradicting that y is a vertex of $P(G, w)$. $\square$

The above 3 claims imply that the graph (V, Y) consists of a single cycle C and a set of isolated vertices $V \setminus V(C)$. Thus $y_e = 0$ for $e \notin C$. By (F) we get that y_e is the same for all $e \in C$, and by (N) we get that $y_e = -1/w(C)$ for all $e \in C$, and in particular that $C \in \mathcal{C}^-(G,w)$. This completes the proof of (1).

We next prove (2). As is well-known, the extreme directions of $P(G,w)$ are in one-to-one correspondence with the vertices of the polytope $P'(G,w)$, obtained from $P(G,w)$ by setting the right-hand side of (N) to 0 and adding the normalization constraint (N') : $\sum_{e\in E} y_e = 1$.

We first note as before that every element of $\mathcal{D}_1 \cup \mathcal{D}_2$ is a vertex of $P'(G,w)$. Indeed, if $y \in \mathcal{D}_2$ is defined by a 2-cycle (C_1, C_2), then there are m linearly independent inequalities tight at y. To see this, we consider two cases: (i) When C_1 and C_2 are edge-disjoint, then there are $|C_1|-1$ and $|C_2|-1$ equations of type (F), normalization equations (N) and (N'), and $m-|C_1|-|C_2|$ non-negativity inequalities for $e \in E\setminus(C_1 \cup C_2)$. (ii) Otherwise, $C_1 \cup C_2$ consists of 3 disjoint paths P_1, P_2, P_3 of, say m_1, m_2 and m_3 arcs, respectively. Then $C_1 \cup C_2$ has $m_1+m_2+m_3-1$ giving $m_1+m_2+m_3-2$ linearly independent equation of type (F), which together with (N), (N') and $m-m_1-m_2-m_3$ non-negativity constraints for $e \in E \setminus (C_1 \cup C_2)$ uniquely define y.

Consider now a vertex y of $P'(G,w)$. Let $Y = \{e \in E : \ y_e > 0\}$. Clearly, Claim 1 is still valid for Y. On the other hand, Claims 2 and 3 can be replaced by the following two claims.

Claim 4 *There exist no 3 distinct cycles C_1, C_2, C_3 such that $C_1 \in \mathcal{C}^-(G,w)$, $C_2 \in \mathcal{C}^+(G,w)$, and $C_1 \cup C_2 \cup C_3 \subseteq Y$.*

Proof. If such C_1, C_2 and C_3 exist, we define two points y' and y'' as follows: $y'_e = y_e + \sum_{i=1}^3 \epsilon_i \chi_e(C_i)$ and $y''_e = y_e - \sum_{i=1}^3 \epsilon_i \chi_e(C_i)$, for $e \in E$, where $\epsilon_3 > 0$ is sufficiently small, and ϵ_1 and ϵ_2 satisfy

$$\begin{aligned} \epsilon_1 w(C_1) + \epsilon_2 w(C_2) &= -\epsilon_3 w(C_3) \\ \epsilon_1 |C_1| + \epsilon_2 |C_2| &= -\epsilon_3 |C_3|. \end{aligned} \tag{3}$$

Note that ϵ_1 and ϵ_2 exist since $\alpha \overset{\text{def}}{=} w(C_1)|C_2| - w(C_2)|C_1| < 0$. Furthermore, since $\epsilon_1 = (w(C_2)|C_3| - w(C_3)|C_2|)\epsilon_3/\alpha$ and $\epsilon_2 = (w(C_3)|C_1| - w(C_1)|C_3|)\epsilon_3/\alpha$, we can select ϵ_3 such that $y', y'' \geq 0$. By definition of y' and y'', they both satisfy (F), and by (3) they also satisfy (N) and (N'). However, $(y' + y'')/2 = y$ contradicts that $y \in \mathcal{V}(P'(G,w))$. □

Claim 5 *There exist no 2 distinct cycles C_1, C_2 such that $C_1, C_2 \in \mathcal{C}^0(G,w)$, and $C_1 \cup C_2 \subseteq Y$.*

Proof. If such C_1 and C_2 exist, we define two points y' and y'' as follows: $y'_e = y_e + \epsilon_1 \chi_e(C_1) + \epsilon_2 \chi_e(C_2)$ and $y''_e = y_e - \epsilon_1 \chi_e(C_1) - \epsilon_2 \chi_e(C_2)$, for $e \in E$, where $\epsilon_2 > 0$ is sufficiently small, and $\epsilon_1 = -\epsilon_2 |C_2|/|C_1|$. Then $y', y'' \in P'(G,w)$ and $y = (y' + y'')/2$. □

As is well-known, we can decompose y into the sum of positive flows on cycles, i.e., write $y = \sum_{C \in \mathcal{C}'} \lambda_C \chi(C)$, where $\mathcal{C}' \subseteq \mathcal{C}^-(G,w) \cup \mathcal{C}^+(G,w) \cup \mathcal{C}^0(G,w)$, and $\lambda_C > 0$ for

$c \in \mathcal{C}'$. It follows from Claim 4 that $|\mathcal{C}'| \leq 2$. Using (N), we get $\sum_{c \in \mathcal{C}'} \lambda_C w(C) = 0$, which implies by Claim 5 that either $\mathcal{C}' = \{C\}$ and $w(C) = 0$ or $\mathcal{C}' = \{C_1, C_2\}$ and $w(C_1) < 0$, $w(C_1) > 0$. In the former case, we get that $y \in \mathcal{D}_1$, and in the latter case, we get by Claim 4 that (C_1, C_2) is a 2-cycle, and hence, that $y \in \mathcal{D}_2$. □

In the next section we construct a weighted directed graph (G, w) in which all negative cycles have unit weight. We show that generating all negative cycles of G is NP-hard, thus implying by Theorem 1 that generating all vertices of $P(G, w)$ is also hard.

4 NP-hardness construction

The construction is essentially the same as in [11]; only the weights change. We include a sketch here.

In the next section we will reduce number of decision problems, concerning polyhedra, from the following CNF satisfiability problem: Is there a truth assignment of N binary variables satisfying all clauses of a given conjunctive normal form $\phi(x_1, \ldots, x_N) = C_1 \wedge \ldots \wedge C_M$, where each C_j is a disjunction of some literals in $\{x_1, \overline{x}_1, \ldots, x_N, \overline{x}_N\}$?

Given a CNF ϕ, we construct a weighted directed graph $G = G(\phi) = (V, E)$ on $|V| = 5 \sum_{j=1}^{M} |C_j| + M - N + 1$ vertices and $|E| = 6 \sum_{j=1}^{M} |C_j| + 1$ arcs (where $|C_j|$ denotes the number of literals appearing in clause C_j) as follows. For each literal $\ell = \ell^j$ appearing in clause C_j, we introduce two paths of three arcs each: $\mathcal{P}(\ell) = (p(\ell), a(\ell), b(\ell), q(\ell))$, and $\mathcal{P}'(\ell) = (r(\ell), b'(\ell), a'(\ell), s(\ell))$. The weights of these arcs are set as follows:

$$w((p(\ell), a(\ell))) = \tfrac{1}{2}, \quad w((a(\ell), b(\ell))) = -\tfrac{1}{2}, \quad w((b(\ell), q(\ell))) = 0,$$

$$w((r(\ell), b'(\ell))) = 0, \quad w((b'(\ell), a'(\ell))) = -\tfrac{1}{2}, \quad w((b'(\ell), s(\ell))) = \tfrac{1}{2}.$$

These edges are connected in G as follows (see Figure 2 for an example):

$$G = v_0 \; \mathcal{G}_1 \; v_1 \; \mathcal{G}_2 \; v_2 \ldots \; v_{N-1} \; \mathcal{G}_n \; v_N \; \mathcal{G}'_1 \; v'_1 \; \mathcal{G}'_2 \; v'_2 \ldots \; v'_{M-1} \; \mathcal{G}'_M \; v'_M,$$

where $v_0, v_1, \ldots, v_N, v'_1, \ldots, v'_{M-1}, v'_M$ are distinct vertices, each $\mathcal{G}_i = \mathcal{Y}_i \vee \mathcal{Z}_i$, for $i = 1, \ldots, N$, consists of two parallel chains $\mathcal{Y}_i = \wedge_j \mathcal{P}(x_i^j)$ and $\mathcal{Z}_i = \wedge_j \mathcal{P}(\overline{x}_i^j)$ between v_{i-1} and v_i, and each $\mathcal{G}'_j = \vee_{i=1}^{|C_j|} \mathcal{P}'(\ell_i^j)$, for $j = 1, \ldots, M$, where $\ell_1^j, \ell_2^j, \ldots$ are the literals appearing in C_j.

Finally we add the arc (v'_M, v_0) with weight $-h$, where $h \geq 0$ will be specified later, and *identify* the pairs of nodes $\{a(\ell), a'(\ell)\}$ and $\{b(\ell), b'(\ell)\}$ for all ℓ, (i.e. $a(\ell) = a(\ell')$ and $b(\ell) = b(\ell')$ define the same nodes).

Clearly the arcs $(a(\ell), b(\ell))$ and $(b'(\ell), a'(\ell))$ form a directed cycle of total weight -1, for every literal occurrence ℓ. Let $\mathcal{S} \subseteq \mathcal{C}^-(G, w)$ be the set of such *short* cycles. Note that $|\mathcal{S}| = \sum_{j=1}^{M} |C_j|$.

Call a cycle of G *long* if it contains the vertices $v_0, v_1, \ldots, v_N, v'_1, \ldots, v'_{M-1}, v'_M$. Any long cycle has weight $-h$. The crucial observation is the following.

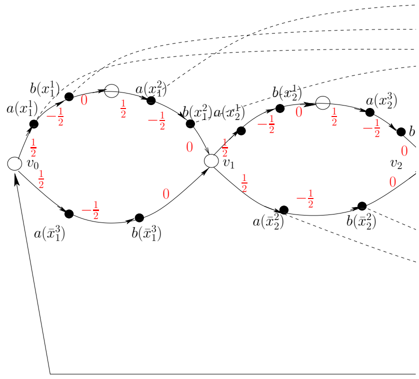


Figure 2: An example of the graph construction in the proof of Theorem 2 with CNF $C = (x_1 \vee x_2 \vee \overline{x}_3) \wedge (x_1 \vee \overline{x}_2 \vee x_3) \wedge (\overline{x}_1 \vee x_2 \vee \overline{x}_3)$.

Lemma 1 *Any negative cycle $C \in \mathcal{C}^-(G, w) \setminus \mathcal{S}$ must either be long or have weight at least $-h + 1$.*

Proof. Consider any cycle $C \notin \mathcal{S}$, and let us write the traces of the nodes visited on the cycle (dropping the literals, and considering a, a' and b, b' as different copies), without loss of generality as follows:

$$p\ a\ b\ p\ a\ b\ p\ \cdots\ a\ a'\ s\ b'\ a'\ s\ b'\ \cdots\ b'\ b\ p\ a\ b\ \cdots\ p.$$

Note that the sequences $a'\ a$ and $b\ b'$ are not allowed since otherwise C contains a cycle from $\mathcal{S}$.

Let us compute the distance (i.e., the total weight) of each node on this sequence starting from the initial node p. Call the subsequences $a\ a'$ and $b'\ b$, a- and b-jumps respectively. Then it is easy to verify that each a-jump causes the distance to eventually increase by 1 while each b-jump keeps the distance at its value. One also observes that, if the sequence has a b-jump, and it contains the nodes v_0 and v'_M, then it must also contain an a-jump. Thus it follows from the definition of $d(x)$ that any cycle with a jump must be either non-negative, if it does not contain the nodes v_0 and v'_M, or have weight at least $-h + 1$, if it contains v_0 and v'_M. So the only possible negative cycle, not in $\mathcal{S}$, of weight less than or equal to $-h$ must be long. □

Corollary 1 *If $h = 1$ in the construction above, then all negative cycles are either short or long.*

The following claim was established in [11].

Lemma 2 *The CNF formula ϕ is satisfiable if and only if $G(\phi)$ contains a long cycle.*

In particular, if $h = 1$, then it is NP-hard to check if $\mathcal{C}^-(G(\phi), w) = \mathcal{S}$.

Remark 1 *Unlike [11], we use "asymmetric" weighting in the above construction in order to force all negative cycles to have the same weight. In fact, in this construction edges have 4 different weights, while the construction in [11] used only 2 different values, namely $+1$ and -1.*

5 Hardness of checking some polyhedral properties

Let $P(G(\phi), w)$ be the polyhedron defined by the graph $G(\phi)$ and the arc weights w, constructed for an input CNF formula ϕ as in the previous section. Let $\mathcal{X} \subseteq \mathcal{V}(P(G(\phi), w))$ be the vertices of $P(G(\phi), w)$ corresponding to the set $\mathcal{S} \subseteq \mathcal{C}^-(G(\phi), w)$, defined in the previous section.

5.1 Generating all vertices of a $0/1$-polyhedron is hard

Let us now show that the following problem is CoNP-complete:

VE-$0/1$**:** Given a polyhedron $P = \{x \in \mathbb{R}^n | \ Ax \leq b\}$, where $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, and $\mathcal{V}(P) \subseteq \{0, 1\}^n$, and a subset $\mathcal{X} \subseteq \mathcal{V}(P)$, decide if $\mathcal{X} = \mathcal{V}(P)$.

Then, no algorithm can generate all elements of $\mathcal{V}(P)$ in incremental or total polynomial time, unless P=NP.

Theorem 2 *Problem VE-*$0/1$ *is NP-hard.*

Proof. Set $h = 1$ in the construction. Then by lemma 1, any negative cycle must either belong to $\mathcal{S}$ or is long. Any such cycle has weight -1. This implies by Theorem 1 that all vertices of $P(G, w)$ are $0/1$, and further that checking if $\mathcal{V}(P(G(\phi), w)) = \mathcal{X}$ is equivalent to checking if $\mathcal{C}^-(G(\phi), w) = \mathcal{S}$. The latter problem is NP-hard by Lemma 2. $\square$

Thus, it is NP-hard to generate all vertices of a $0/1$-polyhedron. However, (2) shows that the above construction cannot be used to imply the same hardness result for polytopes, since the numbers of positive and negative cycles can be exponential and, hence, polyhedron $P(G, w)$ can be highly unbounded. In fact, for the negative cycle polyhedron arising in the construction of Theorem 2, we have the following.

Proposition 1 *For the directed graph $G = (V, E)$ and weight $w : E \to \mathbb{R}$ used in the proof of Theorem 2, both sets $\mathcal{D}(P(G, w))$ and $\mathcal{V}(P(G, w)) \cup \mathcal{D}(P(G, w))$ can be generated in incremental polynomial time.*

Proof. This follows from the fact that for every positive cycle in G there is a negative cycle, edge-disjoint from it, and vice versa (assuming no clause consists of only one literal) as one can easily verify. Hence, the number of 2-cycles and thus the number of extreme directions of $P(G, w)$ satisfy $|\mathcal{D}(P(G, w))| \geq \max\{|\mathcal{C}^+(G, w)|, |\mathcal{C}^-(G, w)|\} + |\mathcal{C}^0(G, w)|$. Thus $\mathcal{D}(P(G, w)$ and $\mathcal{V}(P(G, w)) \cup \mathcal{D}(P(G, w))$ can be generated by generating all cycles of G, which can be done with polynomial delay [15]. $\square$

However, it is open whether the same holds for general graphs. In fact, there exist weighted graphs in which the number of positive cycles is exponentially larger than the number of 2-cycles. Consider for instance, a graph G composed of a directed cycle $(x_1, y_1, \ldots, x_k, y_k)$ of length $2k$, all arcs with weight -1, and $2k$ additional paths

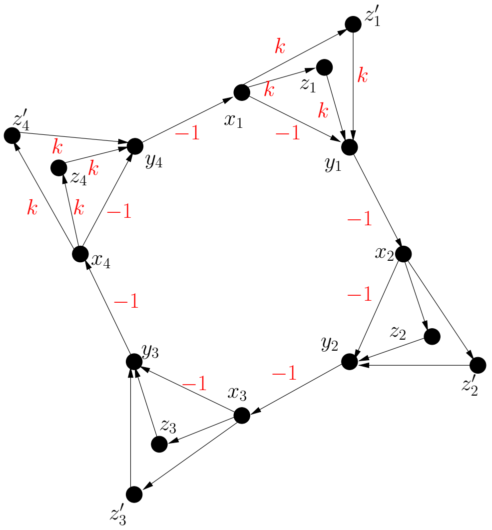


Figure 3: An example where there are exponentially more positive cycles than 2-cycles ($k = 4$).

$\mathcal{P}_1, \mathcal{P}_1', \ldots, \mathcal{P}_k, \mathcal{P}_k'$ where $\mathcal{P}_i = (x_i, z_i, y_i)$ and $\mathcal{P}_i' = (x_i, z_i', y_i)$, of two arcs each going the same direction parallel with every second arc along the cycle, each having a weight of $2k$ (see Figure 3 for an example with $k = 4$). Then we have more than 2^k positive cycles, but only $2k$ 2-cycles. Note that proving that enumerating 2-cycles of a given weighted graph is NP-hard, will imply the same for the vertex enumeration problem for polytopes, whose complexity remains open.

5.2 Recognizing $0/1$-polyhedra is hard

The problem of checking if a polyhedron $P = \{x \in \mathbb{R}^n : \ Ax \geq \mathbf{e},\ x \geq 0\}$, where $A \in \{0,1\}^{V \times E}$ is the vertex-edge incidence matrix of an undirected graph $G = (V, E)$ and $\mathbf{e}$ is the vector of all ones, is a $0/1$ polyhedron, was shown in [9] to be NP-hard. The $0/1$-vertices of P are the *edge-covers* of G, whose number is exponential in n for the graph used in the NP-hardness construction of [9], and these are the only vertices of P which could possibly be integral. In other words, the result of [9] implies that is hard to distinguish if a polyhedron is $0/1$ or fractional. In contrast, we show here the following.

Theorem 3 *Given a polyhedron $P = \{x \in \mathbb{R}^n \ : \ Ax = b, x \geq 0\}$, such that $A = \left[\frac{A'}{r}\right]$ has exactly one "+1" and one "-1" per column, $r \in \{-1, +1\}^m$, $b = [0, \ldots, 0, -2]^T$, and $\mathcal{V}(P) \subseteq \{0,1,2\}^n$, it is NP-hard to tell if $\mathcal{V}(P) \subseteq \{0,1\}^n$, even if P has a polynomial number of $0/1$-vertices.*

Proof. We set $h = \frac{1}{2}$. It follows that all short negative cycles have weight -1, any long negative cycle has weight $-1/2$, and by Lemma 1, there exists no other negative cycles. In particular, by (1), $\mathcal{V}(P(G,w))$ can be partitioned into two sets $\mathcal{V}_1, \mathcal{V}_2$, where $\mathcal{V}_1$ is the set of vertices corresponding to short negative cycles, and $\mathcal{V}_2$ are the ones corresponding to long negative cycles. The theorem follows from the following facts

(i) $P(G,w)$ is integral: $\mathcal{V}(P(G,w)) = \mathcal{V}_1 \cup \mathcal{V}_2 \subseteq \{0,1,2\}^E$,

(ii) $\mathcal{V}_1 \subseteq \{0,1\}^E$ and $\mathcal{V}_2 \subseteq \{0,2\}^E$,

(iii) $|\mathcal{V}_1| \leq |E|$, and

(iv) checking if $\mathcal{V}_2$ is non-empty is NP-hard (see Lemma 2).

$\square$

Papadimitriou and Yannakakis [13] showed that checking if a polytope, given by its facets, is integral is an NP-hard problem. It will be interesting to extend their result by showing that checking if a given integral polytope is 0/1 is also hard.

5.3 Checking for half-integrality is hard

Let f be an integer. A polyhedron $P \subseteq \mathbb{R}^n$, with $\mathcal{V}(P) \in [0,1]^n$, is said to be $\frac{1}{f}$*-integral* [17] if $\mathcal{V}(P) \subseteq \{0, \frac{1}{f}, \frac{1}{f-1}, \ldots, \frac{1}{2}, 1\}^n$. In particular, for $f = 1/2$, such a polyhedron is called *half-integral*. For instance, if $A \in \{0,1\}^{E \times V}$ is the edge-vertex incidence matrix of a graph $G = (V,E)$, then the polyhedron $P = \{x \in \mathbb{R}^E \ : \ Ax \geq \mathbf{e}, \ x \geq 0\}$ is half-integral. The importance of half-integral (or more generally $\frac{1}{f}$-integral) polyhedra $P = \{x \in \mathbb{R}^n : \ Ax = b, \ x \geq 0\}$ is that, for any $c \in \mathbb{R}^n$, one can approximate the optimum of $\min\{cx \ : \ x \in P \cap \{0,1\}^n\}$ within a relative factor of 2, by solving the linear programming relaxation $\min\{cx \ : \ x \in P\}$, and rounding to 1 all variables with value at least 1/2. Thus it will be interesting to be able to recognize such classes of polyhedra in polynomial time. The next result states that this is highly unlikely, unless P=NP.

Theorem 4 *Given a polyhedron P, with $\mathcal{V}(P) \subseteq [0,1]^n$ and an integer $f \geq 1$. It is NP-hard to decide if P is $\frac{1}{f}$-integral.*

Proof. We set $h = (f+1)$ in the construction. Then Lemma 1 implies that any non-long negative cycle has weight of at least $-f$, while a long negative cycle has weight $-(f+1)$. It follows that each vertex of $P(G,w)$ corresponding to a non-long negative cycle has components in $\{0, 1, \frac{1}{2}, \ldots, \frac{1}{f}\}$. Thus the only negative cycle corresponding to a vertex with some component possibly less than $1/f$ is a long negative cycle. But checking for the existence of such cycle is NP-hard by Lemma 2. $\square$

5.4 Hardness of approximating the maximum support

For a polyhedron $P = \{x \in \mathbb{R}^n : \ Ax = b, \ x \geq 0\}$, where $A \in \mathbb{R}^{m \times n}$, the support of a vertex $x \in \mathcal{V}(P)$ is defined as $\text{supp}(x) = |\{i \in [n] \ : \ x_i > 0\}|$ the number of positive components of x. Let $\text{max-supp}(P) = \max\{\text{supp}(x) \ : \ x \in \mathcal{V}(P)\}$. Given an unweighted directed graph $G = (V, E)$, let us assign weight -1 to each arc. Then $P(G, w)$ is polytope whose vertices are in one-to-one correspondence with the directed simple cycles of G. It follows that the vertex with maximum support in $\mathcal{V}(P)$ corresponds to the longest cycle in G. It was shown in [5] that it is not possible to approximate the longest cycle in directed graph within a factor $|V|^{1-\epsilon}$, for any $\epsilon > 0$, unless $P = NP$. It follows that max-supp(P), for a polytope P, is NP-hard to approximate within a factor of $n^{1-\epsilon}$, for any $\epsilon > 0$. Here we show a stronger result for polyhedra.

Theorem 5 *For a polyhedron $P = \{x \in \mathbb{R}^n : \ Ax = b, \ x \geq 0\}$, the following problems are NP-hard:*

(i) Checking if P has a vertex of support more than 2.

(ii) Checking if P has a vertex with $x_i > 0$ for a given i.

(iii) Approximating max-supp(P) *within a factor of bigger than* $12/n$.

Proof. Set $h = 1$ in the construction of Section 4. (i) follows from the observation that vertices of $P(G(\phi), w)$ corresponding to short cycles have support 2, while the existence of a vertex of bigger support is equivalent to the existence of a long cycle. (ii) follows from the observation that a long cycle, if one exists, must contain the arc (v'_M, v_0). To see (iii), observe that the hardness construction remains valid even if we assume that the CNF formula ϕ has 3 literals per clause, and each literal appears in at least one of the clauses. With such an assumption and using the notation of Section 4, we have $n = |E| = 18M + 1$. Any long cycle has length at least $3(N + M) + 1 \geq n/6$ while a short cycle has length 2. Thus we can check the existence of a long cycle if we cannot approximate max-supp($P(G(\phi), w)$) within a factor bigger than $\frac{12}{n}$.

$\square$

References

[1] S. D. Abdullahi, M. E. Dyer, and L. G. Proll, *Listing vertices of simple polyhedra associated with dual LI(2) systems*, DMTCS: Discrete Mathematics and Theoretical Computer Science, 4th International Conference, DMTCS 2003, Proceedings, 2003, pp. 89–96.

[2] D. Avis, B. Bremner, and R. Seidel, *How good are convex hull algorithms*, Computational Geometry: Theory and Applications **7** (1997), 265–302.

[3] D. Avis and K. Fukuda, *A pivoting algorithm for convex hulls and vertex enumeration of arrangements and polyhedra*, Discrete and Computational Geometry **8** (1992), no. 3, 295–313.

[4] ______, *Reverse search for enumeration*, Discrete Applied Mathematics **65** (1996), no. 1-3, 21–46.

[5] A. Björklund, T. Husfeldt, and S. Khanna, *Approximating longest directed paths and cycles*, ICALP, 2004, pp. 222–233.

[6] B. Boros, K. Elbassioni, V. Gurvich, and K. Makino, *Generating vertices of polyhedra and related monotone generation problems*, DIMACS Technical Report 2007-03, Rutgers University, 2007.

[7] D. Bremner, K. Fukuda, and A. Marzetta, *Primal-dual methods for vertex and facet enumeration*, Discrete and Computational Geometry **20** (1998), 333–357.

[8] M. R. Bussieck and M. E. Lübbecke, *The vertex set of a 0/1 polytope is strongly $\mathcal{P}$-enumerable*, Computational Geometry: Theory and Applications **11** (1998), no. 2, 103–109.

[9] G. Ding, L. Feng, and W. Zang, *The complexity of recognizing linear systems with certain integrality properties*, Math. Program., Ser. A., to appear (2007).

[10] N. Garg and V. V. Vazirani, *A polyhedron with all s-t cuts as vertices, and adjacency of cuts*, Math. Program. **70** (1995), no. 1, 17–25.

[11] L. Khachiyan, E. Boros, K. Borys, K. Elbassioni, and V. Gurvich, *Generating all vertices of a polyhedron is hard*, Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2006, 2006, pp. 758–765.

[12] L. Lovász, *Combinatorial optimization: some problems and trends*, DIMACS Technical Report 92-53, Rutgers University, 1992.

[13] C. H. Papadimitriou and M. Yannakakis, *On recognizing integer polyhedra*, Combinatorica **10** (1990), no. 1, 107–109.

[14] J.S. Provan, *Efficient enumeration of the vertices of polyhedra associated with network lp's*, Mathematical Programming **63** (1994), no. 1, 47–64.

[15] R. C. Read and R. E. Tarjan, *Bounds on backtrack algorithms for listing cycles, paths, and spanning trees*, Networks **5** (1975), 237–252.

[16] A. Schrijver, *Theory of linear and integer programming*, Wiley, New York, 1986.

[17] V. V. Vazirani, *Approximation algorithms*, Springer Verlag, Berlin, Heidelberg, New York, 2001.