RUTCOR
RESEARCH
REPORT

GENERATING ALL VERTICES OF A POLYHEDRON IS HARD

Leonid Khachiyan [a] Endre Boros[b] Konrad Borys[c]
Khaled Elbassioni[d] Vladimir Gurvich[e]

RRR 18-2005, June 2005

RUTCOR
Rutgers Center for
Operations Research
Rutgers University
640 Bartholomew Road
Piscataway, New Jersey
08854-8003
Telephone: 732-445-3804
Telefax: 732-445-5472
Email: rrr@rutcor.rutgers.edu
http://rutcor.rutgers.edu/∼rrr

[a]Our friend and colleague, Leo Khachiyan passed away with tragic suddenness while we were preparing this manuscript.

[b]RUTCOR, Rutgers University, 640 Bartholomew Road, Piscataway NJ 08854. Email: boros@rutcor.rutgers.edu

[c]RUTCOR, Rutgers University, 640 Bartholomew Road, Piscataway NJ 08854. Email: kborys@rutcor.rutgers.edu

[d]Max-Planck-Institut für Informatik, Saarbrücken, Germany. Email: elbassio@mpi-sb.mpg.de

[e]RUTCOR, Rutgers University, 640 Bartholomew Road, Piscataway NJ 08854-8003. Email: gurvich@rutcor.rutgers.edu

Rutcor Research Report
RRR 18-2005, June 2005

Generating all vertices of a polyhedron is hard

Leonid Khachiyan Endre Boros Konrad Borys
Khaled Elbassioni Vladimir Gurvich

Abstract. We show that generating all negative cycles of a weighted graph is a hard enumeration problem, in both the directed and undirected cases. More precisely, given a family of (directed) negative cycles, it is NP-complete problem to decide whether this family can be extended or there are no other negative (directed) cycles in the graph, implying that (directed) negative cycles cannot be generated in polynomial output time, unless P=NP. As a corollary, we solve in the negative two well-known generating problems from linear programming: (i) Given an (infeasible) system of linear inequalities, generating all minimal infeasible subsystems is hard. Yet, for generating maximal feasible subsystems the complexity remains open. (ii) Given a (feasible) system of linear inequalities, generating all vertices of the corresponding polyhedron is hard. Yet, in case of bounded polyhedra the complexity remains open
keywords polytope, polyhedron, polytope-polyhedron problem, vertex, face, facet, enumeration problem, vertex enumeration, facet enumeration, graph, cycle, negative cycle, linear inequalities, feasible system

---

Acknowledgements: This research was partially supported by the National Science Foundation (Grant IIS-0118635), and by DIMACS, the National Science Foundation's Center for Discrete Mathematics and Theoretical Computer Science.

1 Introduction and Main Results

Let $G = (V, E)$ be a directed graph (digraph)and $w : E \to \mathbb{R}$ be a real-valued weight function defined on its arcs. We will call such a pair a *weighted digraph* and denote it by (G, w). For every subset of arcs $F \subseteq E$ its weight is defined as the total weight of all its arcs, $w(F) = \sum_{e \in F} w(e)$. In particular, a simple directed cycle is called *negative* if its weight is negative. Finally, let us denote by $\mathcal{C}^- = \mathcal{C}^-(G, w)$ the family of negative cycles of (G, w), i.e., $\mathcal{C}^- = \{C \subseteq E \mid C$ is a simple directed cycle, with $w(C) < 0\}$.

First we consider the problem of generating exhaustively all negative cycles of a given weighted directed graph (G, w), in other words the problem of enumerating the family $\mathcal{C}^-(G, w)$. Since the number of negative cycles may be exponential in the size of input description, i.e., the size of G and w, the efficiency of such enumeration algorithms is measured customarily in both the input and output sizes (see e.g., Valiant [1979], Lawler et al. [1980]). More precisely, such an enumeration problem is said to be solvable in *polynomial total time* if the output can be generated in time polynomial in the input and output sizes. It is easy to see that a family $\mathcal{C}$ is enumerable in polynomial total time if and only if for each subfamily $\mathcal{X} \subseteq \mathcal{C}$, the problem of deciding $\mathcal{X} \neq \mathcal{C}$, and if yes, finding $C \in \mathcal{C} \setminus \mathcal{X}$, is solvable in time polynomial in $size(G, w)$ and $|\mathcal{X}|$. On the other hand, when this decision problem is NP-hard, the enumeration problem is called NP-hard, too (see Lawler et al. [1980]). Thus, NP-hard enumeration problems are unlikely to have efficient solutions, unless P=NP.

Our main result claims that enumerating negative directed cycles of a weighted directed graph is a hard enumeration problem.

Theorem 1 *Given a weighted digraph $G = (V, E)$, $w : E \to \mathbb{R}$ and a family $\mathcal{X} \subseteq \mathcal{C}^-$ of its negative directed cycles, it is an NP-complete problem to decide whether $\mathcal{X} \neq \mathcal{C}^-$, even if w takes only two different values.*

Let us remark that all directed cycles of a directed graph can be enumerated efficiently, e.g., by a simple backtracking algorithm, just like all simple cycles of an undirected graph (see e.g., [Read and Tarjan, 1975]). Let us also add that the analogous hardness result can be shown for undirected graphs, as well. Let us also denote by $\mathcal{C}^- = \mathcal{C}^-(G, w)$ the family of all simple cycles of an undirected graph $G = (V, E)$, the edges in which have a negative total weight with respect to a given weight function $w : E \to \mathbb{R}$.

Theorem 2 *Given a weighted undirected graph $G = (V, E)$, $w : E \to \mathbb{R}$, and a family $\mathcal{X} \subseteq \mathcal{C}^-(G, w)$ of its negative cycles, it is an NP-complete problem to decide whether $\mathcal{X} \neq \mathcal{C}^-$, even if w takes only two different values.*

Let us note that if w takes the same value for all edges (arcs), then negative (directed) cycles either do not exist, or all (directed) cycles are negative. Thus, the enumeration problems for both directed and undirected graphs can efficiently be solved, as we noted above. Furthermore, when w takes only two different values, those can be assumed to be integers, and hence by edge (arc) splitting) the input can be transformed to one, in which all edges (arcs) have weight ± 1. Though this transformation may increase the size of the input in a non polynomial way, in case of the specific constructions we provide in the proofs of the above two theorems, it is a polynomial transformation, implying that generating all negative (directed) cycles is NP-hard even if all edges (arcs) have weights ± 1.

We shall derive several consequences of the above results, including the hardness of generating all vertices of a (possibly unbounded) polyhedron, generating all minimal infeasible subsystems of a system of linear inequalities, etc. We prove Theorems 1 and 2 in sections 2 and 3, respectively.

1.1 Negative Cycles and Minimal Infeasible Subsystems

Let us first note that deciding the existence and finding a negative cycle in a weighted directed graph are polynomially solvable tasks. Gallai [1958] proved that (G, w) has no negative cycle if and only if by a potential transformation all edge weights can be changed to nonnegative values, while Karp [1978] provided an $O(|V|^3)$ algorithm to find a directed cycle with the minimum average weight, which of course must be negative if the graph has negative cycles at all. We shall utilize Gallai's approach to reformulate the problem and derive some interesting consequences.

To a weighted digraph (G, w), where $G = (V, E)$ and $w : E \to \mathbb{R}$, let us associate a polyhedron $P(E, w)$ defined by

$$P(E, w) = \{x \in \mathbb{R}^V \mid x_u - x_v \geq -w(u, v) \text{ for all arcs } (u, v) \in E\} \tag{1}$$

Let us note that every vector $x \in P(E, w)$ is a potential in Gallai's sense, proving that G is negative cycle free. Namely, defining $w'(u, v) = w(u, v) + x_u - x_v$ for all arcs $(u, v) \in E$ we get another weighting of the arcs of G, such that $w'(C) = w(C)$ for all directed cycles $C \subseteq E$, and for which $w'(u, v) \geq 0$ for all arcs $(u, v) \in E$, according to the definition of $P(E, w)$. This latter shows that G is indeed negative cycle free.

Applying thus Gallai's result to subgraphs of G we obtain that $P(E', w) = \emptyset$ for some $E' \subseteq E$ if and only if the subgraph $G' = (V, E')$ contains a negative cycle with respect to the weight function w. Therefore, the minimal infeasible subsystems of the system of linear inequalities (1) correspond in a one-to-one way to the negative cycles of (G, w). Hence, Theorem 1 implies the following result.

Corollary 1 *Enumerating all minimal infeasible subsystems of a system of linear inequalities is an NP-hard enumeration problem, even if we restrict the input to linear systems involving at most two variables in each inequality.* □

The problems of finding minimal infeasible subsystems of a system of linear inequalities, sometimes called *IIS* (*Irreducible Inconsistent Subsystems*) or *Helly systems*, and its natural dual of finding maximal feasible subsystems received ample attention in the literature, see e.g., [Ryan, 1996, Pfetsch, 2002]. The optimization versions of these problems, i.e., finding a maximum cardinality feasible subsystem, and finding a minimum cardinality infeasible subsystem are known to be NP-hard, see e.g., [Johnson and Preparata, 1978, Chakravarti, 1994, Pfetsch, 2002].

1.2 Negative Cycles and Vertex Enumeration

Let us recall that the infeasibility of a system of linear inequalities is well characterized by Farkas' lemma: either the system $Ax \geq b$ has a solution, or there exists a nonnegative vector $y \geq 0$ such that $y^T A = 0$ and $y^T b > 0$, but not both (see [Farkas, 1901]). Using this claim, Gleeson and Ryan [1990] associated to a system of linear inequalities $Ax \geq b$, $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$, a so called *alternative polyhedron* defined as $Q = \{y \in \mathbb{R}^m_+ \mid y^T A = 0,\ y^T b = 1\}$, and observed that minimal infeasible subsystems of $Ax \geq b$ are in a one-to-one correspondence with vertices of Q. Indeed, for every vector $y \in Q$ let us consider the subsystem of $Ax \geq b$ corresponding to the support set $S(y) = \{i \mid y_i \neq 0\}$. By Farkas' lemma, we have that these corresponding subsystems are indeed infeasible. Conversely, if S is the index set of an infeasible subsystem of $Ax \geq b$, then again by Farkas' lemma we have a vector $y \in Q$ for which $S(y) \subseteq S$. Thus, minimal infeasible subsystems correspond to vectors $y \in Q$ with minimal support sets, and hence those are indeed vertices of Q.

This observation, coupled with Corollary 1 implies the hardness of enumerating the vertices of polyhedra.

Corollary 2 *Enumerating all vertices of a rational polyhedron, given as the intersection of finitely many closed half-spaces, is an NP-hard enumeration problem.*

Proof: Let us consider an infeasible system of rational linear inequalities $Ax \geq b$, and its alternative polyhedron Q. We can write Q equivalently as $Q = \{y \in \mathbb{R}^m \mid y \geq 0,\ A^T y \geq 0,\ -A^T y \geq 0,\ b^T y \geq 1,\ -b^T y \geq -1\}$, i.e., as the intersection of $m + 2n + 2$ closed half-spaces. Thus, by the above observation, enumerating the vertices of this rational polyhedron would also enumerate all minimal infeasible subsystems of $Ax \geq b$, which is an NP-hard enumeration problem according to Corollary 1. □

Vertex enumeration is a fundamental problem in computational geometry and polyhedral combinatorics (see e.g., Dyer and Proll [1977] for a list of applications), and has many equivalent formulations. Most notably for bounded polyhedra vertex enumeration is equivalent with *facet generation*, i.e., enumerating the facets of a polytope given by an explicit list of its vertices(see e.g., the so called polytope-polyhedron problem in Lovász [1992]).

Let us emphasize that whenever the system of equations $A^T y = 0$, $b^T y = 0$ has a nontrivial solutions for which $y \geq 0$, then Q in the above Corollary 2 is an unbounded polyhedron. Thus, our reduction through Theorem 1 yields in general, unbounded polyhedra, and hence does not imply the hardness of vertex generation for bounded polyhedra, which still remains an open problem. Furthermore and equivalently, the complexity of enumerating together vertices and extreme rays of polyhedra is also an open problem.

Numerous algorithmic ideas have been introduced in the literature (either for vertex or for facet enumeration, see e.g., Charnes et al. [1953], Motzkin et al. [1953], Balinski [1961], Chand and Kapur [1970], Mattheiss [1973], Dyer and Proll [1977], Chvátal [1983], Dyer [1983], Swart [1985], Seidel [1986], Avis and Fukuda [1992], Provan [1994], Avis and Fukuda [1996], Bremner et al. [1998], Bussieck and Lübbecke [1998], Abdullahi [2003]). Efficient algorithms (typically linear in the number of vertices) were proposed for several special cases, including non-degenerate polyhedra, i.e. in which every vertex is incident with exactly n facets, [Avis and Fukuda, 1992], network polytopes [Provan, 1994], polytopes with zero-one vertices [Bussieck and Lübbecke, 1998], and polyhedra in which every facet defining inequality involves at most two nonzero coefficients [Abdullahi, 2003]. However, no method proved to be efficient (yet) for the general case. In fact several publications [Avis et al., 1997, Fukuda et al., 1997, Bremner, 1999] analyzed the proposed general purpose methods for vertex/facet enumeration, and showed that all of the known algorithms may require in the worst case superpolynomial time in the output size. Along the same lines, Corollary 2 shows that vertex enumeration is indeed a hard enumeration problem for unbounded polyhedra (unless of course P=NP).

In analyzing the reasons why backtracking methods are not efficient for vertex enumeration, in general, Fukuda et al. [1997] noted that such methods require solving repeatedly decision problems, which turn out to be NP-hard. In particular, they showed that for a given rational polyhedron P and an open rational half-space $H = \{x \in \mathbb{R}^n \mid \alpha^T x > \beta\}$, it is NP-hard to decide if P has a vertex in H. Let us note that the same decision problem for bounded polyhedra is much easier, since it can be decided by maximizing $\alpha^T x$ over P, which is a linear programming problem, known to be polynomially solvable by [Khachiyan, 1979]. We can show, as a next corollary of Theorem 1 that the enumerative version of this decision problem is hard, already for polytopes.

To arrive to this claim, let us recall that the nontrivial vertices of the circulation polytope

$$P(G) = \left\{ y \in \mathbb{R}^E \;\middle|\; \begin{array}{rcll} \sum\limits_{v:(u,v)\in E} y_{uv} - \sum\limits_{w:(w,u)\in E} y_{wu} & = & 0 & \forall \;\; u \in V \\ \sum_{(u,v)\in E} y_{uv} & = & 1 & \\ 0 \leq y_{uv} & & & \forall \;\; (u,v) \in E \end{array} \right\}$$

of a directed graph $G = (V, E)$ correspond to simple directed cycles of G. Let us remark that $P(G)$ is a frequently occurring polytope in the optimization literature, the vertices and facial structure of which is well studied and understood. In particular, its vertices can be generated in linear (output) time either by cycle enumeration Read and Tarjan [1975] or by the method proposed in Bussieck and Lübbecke [1998].

Associating further to a rational weight function $w : E \to \mathbb{R}$ an open rational half-space defined by

$$H = \left\{ y \in \mathbb{R}^E \;\middle|\; \sum_{(u,v)\in E} w(u,v) y_{uv} \; < \; 0 \right\}$$

we get that the support sets of vertices of $P(G)$ belonging to H are exactly the negative cycles of the weighted directed graph (G, w). Thus, Theorem 1 readily implies the following claim.

Corollary 3 *Given a rational polyhedron P and an open rational half-space H, it is NP-hard to enumerate all vertices of P which belong to H, even if P is bounded.* □

Many applications (see e.g. Dyer and Proll [1977]) call for the enumeration of all those basic feasible solutions to a linear programming problem (i.e., vertices of the corresponding polyhedron), the corresponding objective function value of which is above a given threshold. Corollary 3 indicates that unfortunately such enumeration problems might be difficult, in general, unless P=NP.

A further consequence of Theorem 1 is that enumerating all vertices of a polytope P which do not belong to a given face of P is also hard, in general.

Corollary 4 *Given a polytope P and a face F of it, it is NP-hard to enumerate the vertices of P which do not belong to F.*

Proof: Note that $P' = P(G) \cap H$, as defined above, is a polytope, for which H is facet defining, and the vertices of which outside H correspond in a one-to-one way to the negative cycles of the weighted graph (G, w) to which we associated H and $P(G)$. Thus the claim follows by Theorem 1. □

1.3 Four Geometric Enumeration Problems

Let us finally recall four strongly related geometric enumeration problems. Let $\mathcal{A} \subseteq \mathbb{R}^n$ be a given subset of vectors in $\mathbb{R}^n$, fix a point $z \in \mathbb{R}^n$ called the *center*, and consider the following four definitions:

- A *simplex* is a minimal subset $X \subseteq \mathcal{A}$ containing the center in its convex hull, i.e., $z \in conv(X)$.

- An *anti-simplex* is a maximal subset $X \subseteq \mathcal{A}$ not containing the center in its convex hull, i.e., $z \notin conv(X)$.
- A *body* is a minimal (full-dimensional) subset $X \subseteq \mathcal{A}$ containing the center in the interior of its convex hull, i.e., $z \in int(conv(X))$.
- An *anti-body* is a maximal subset $X \subseteq \mathcal{A}$ not containing the center in the interior of its convex hull, i.e., $z \notin int(conv(X))$.

Equivalently, a simplex (body) is a minimal collection of the given vectors not contained in an *open* (*closed*) half-space through the center, while an anti-simplex (anti-body) is a maximal collection of vectors contained in an open (closed) half space through the center. It can be seen easily that $|X| \leq n+1$ for a simplex, and that $n+1 \leq |X| \leq 2n$ for a body.

For a given point set $\mathcal{A} \subseteq \mathbb{R}^n$ and center $z \in \mathbb{R}^n$, let us denote respectively by $\mathcal{S}$ and $\mathcal{B}$ the hypergraphs on the base set $\mathcal{A}$, consisting of all simplices, and respectively all bodies of $\mathcal{A}$. The corresponding families of maximal independent sets of these two hypergraphs are respectively all anti-simplices and anti-bodies of $\mathcal{A}$, denoted respectively by $\mathcal{S}^*$ and $\mathcal{B}^*$, i.e.,

$$\begin{aligned} \mathcal{S}^* &= \{\ X \subseteq \mathcal{A} \mid X \text{ is maximal such that } X \not\supseteq S\ \forall S \in \mathcal{S}\ \}, \\ \mathcal{B}^* &= \{\ Y \subseteq \mathcal{A} \mid Y \text{ is maximal such that } Y \not\supseteq B\ \forall B \in \mathcal{B}\ \}. \end{aligned}$$

Simplices, anti-simplices, bodies and anti-bodies can naturally be related to minimal infeasible or maximal feasible subsystems of certain linear systems of inequalities. Namely, let us denote by $A \in \mathbb{R}^{m \times n}$, where $m = |\mathcal{A}|$, the matrix whose row vectors are the vectors of $\mathcal{A}$, and let $e \in \mathbb{R}^m$ denote the m-dimensional vector of all ones.

It follows from the above definitions that simplices and anti-simplices are in a one-to-one correspondence respectively with the minimal infeasible and maximal feasible subsystems of the linear system of inequalities:

$$Ax \geq \mathbf{e}, \quad x \in \mathbb{R}^n. \tag{2}$$

Similarly, it follows that bodies and anti-bodies correspond, in a one-to-one way, respectively to the minimal infeasible and maximal feasible subsystems of the system:

$$Ax \geq \mathbf{0}, \quad x \neq \mathbf{0}. \tag{3}$$

As of the complexity of these enumeration problems, it is known that the generation of anti-bodies is a hard problem:

Proposition 1 (Boros et al. [2004]) *Given a set of vectors $\mathcal{A} \subseteq \mathbb{R}^n$, and a partial list $\mathcal{X} \subseteq \mathcal{B}^*$ of the anti-bodies of S, it is NP-hard to determine if the given list is incomplete, i.e. $\mathcal{X} \neq \mathcal{B}^*$, or not. Equivalently, given an infeasible system* (3), *and a partial list of its maximal feasible subsystems, it is NP-hard to determine if the given partial list is incomplete, or not.* □

Enumeration of bodies turns out to be at least as hard as the well-known *hypergraph transversal problem* [Eiter and Gottlob, 1995], which is not known to be solvable in incremental polynomial time.

Proposition 2 (Boros et al. [2004]) *The problem of incrementally enumerating bodies, for a given set of $m+n$ points $\mathcal{A} \subseteq \mathbb{R}^n$ and center $z = \mathbf{0}$, includes as a special case the problem of enumerating all minimal transversals for a given hypergraph $\mathcal{H}$ with n hyperedges on m vertices. Equivalently, generating minimal infeasible subsystems of* (3) *is at least as hard as hypergraph transversal generation.* □

It should be added that the best currently known algorithm for the hypergraph transversal problem runs in incremental *quasi-polynomial* time (see Fredman and Khachiyan [1996]).

The problem of generating simplices turns out to be equivalent, in general, with the problem of enumerating the vertices of bounded polyhedra, or enumerating the vertices and extreme rays of possibly unbounded polyhedra. To see this, let us consider a vector set $\mathcal{A} \subseteq \mathbb{R}^n$, and center $z = \mathbf{0}$, and consider the polyhedron $P = \{x \in \mathbb{R}^n \mid Ax = -b,\ x \geq 0\}$, where $b \in \mathcal{A}$, and the column vectors of matrix $A = [a_1, ..., a_n]$ are the vectors of $\mathcal{A} \setminus \{b\}$ (i.e., $n = |\mathcal{A}| - 1$). Recall that for a vector $y \in \mathbb{R}^n$ we called the set $S(y) = \{i \mid y_i \neq 0\}$ its support set.

Proposition 3 *If $y \in P$ is a vertex of P then the set $\{a_i \mid i \in S(y)\} \cup \{b\}$ is a simplex of $\mathcal{A}$, while if $y \in P$ is an extreme ray of P then the set $\{a_i \mid i \in S(y)\}$ is a simplex of $\mathcal{A}$ (assuming $z = \mathbf{0}$ as center, in both cases). Furthermore, every simplex of $\mathcal{A}$ with center $z = \mathbf{0}$ correspond in this way either to a vertex or to an extreme ray of P.*

Proof: The first two claims are easy to see by the definitions. For the last claim, let $S \subseteq \mathcal{A}$ be a simplex, i.e., a minimal subset for which $\mathbf{0} \in conv(S)$. If $b \in S$, then we have for some $\lambda_a \geq 0$, $a \in S \setminus \{b\}$ and $\lambda_b \geq 0$, with $\lambda_b + \sum_{a \in S \setminus \{b\}} \lambda_a = 1$ that

$$-\lambda_b b = \sum_{a \in S \setminus \{b\}} \lambda_a a.$$

Since S is minimal, we must have all these coefficients positive, and thus

$$-b = \sum_{a \in S \setminus \{b\}} \frac{\lambda_a}{\lambda_b} a.$$

Thus the vector $x \in \mathbb{R}^n$, defined by

$$x_i = \begin{cases} \frac{\lambda_{a_i}}{\lambda_b} & \text{if } a_i \in S \setminus \{b\}, \\ 0 & \text{otherwise} \end{cases}$$

for $i = 1, ..., n$, is a vertex of P, again by the minimality of S. While if $b \notin S$, then we have

$$\mathbf{0} = \sum_{a \in S} \lambda_a a$$

for some positive coefficients $\lambda_a > 0$, $a \in S$ for which $\sum_{a \in S} \lambda_a = 1$, and thus the vector $x \in \mathbb{R}^n$, defined by

$$x_i = \begin{cases} \lambda_{a_i} & \text{if } a_i \in S, \\ 0 & \text{otherwise} \end{cases}$$

for $i = 1, ..., n$, is a vertex of P, once more by the minimality of S. □

In particular, if $P = \{x \in \mathbb{R}^n \mid Ax = b,\ x \geq 0\}$ is a bounded polyhedron, i.e., if $Ax = \mathbf{0}$ has no nontrivial nonnegative solutions, then the vertices of P correspond in a one-to-one way to the simplices of the set $\mathcal{A}$ formed by the column vectors of A and b, with center $z = \mathbf{0}$.

For the special case of vectors $\mathcal{A} \subseteq \mathbb{R}^n$ in general position, we have $\mathcal{B} = \mathcal{S}$, and consequently the problem of enumerating bodies of $\mathcal{A}$ turns into the problem of enumerating vertices of the polytope $\{x \in \mathbb{R}^n \mid Ax = 0, \mathbf{e}^T x = 1, x \geq \mathbf{0}\}$, each vertex of which is non-degenerate and has exactly $n + 1$ positive components. For such kinds of *simple* polytopes there exist algorithms that generate all vertices with polynomial delay (see e.g., Chvátal [1983], Avis and Fukuda [1992]).

Let us finally mention that, although the status of the problem of enumerating all maximal feasible subsystems of (2) is not known in general, the situation changes if we fix a consistent subfamily of inequalities, and ask for enumerating all its extensions to a maximal feasible subsystem. In fact, such a problem turns out to be NP-hard, even if we fix only non-negativity constraints.

Proposition 4 (Boros et al. [2004]) *Let $A \in \mathbb{R}^{m \times n}$ be an $m \times n$ matrix, $b \in \mathbb{R}^m$ be an m-dimensional vector, and assume that the system*

$$Ax \geq b, \quad x \in \mathbb{R}^n \tag{4}$$

has no solution $x \geq \mathbf{0}$. Let $\mathcal{F}$ be the family of all maximal subsystems of (4) which can be satisfied by a non-negative solution x. Then, given a partial list $\mathcal{X} \subseteq \mathcal{F}$, it is an NP-complete problem to determine if the list is incomplete, i.e., if $\mathcal{X} \neq \mathcal{F}$, even if b is a unit vector, and entries in A are either, $-1, 1,$ or 0. □

We conclude with the observation that the problem of finding, for an infeasible system

$$Dx \geq f, \quad D'x \geq f', \tag{5}$$

all maximal feasible subsystems extending the feasible subsystem $D'x \geq f'$, naturally includes both the problems of generating anti-simplices and simplices. Clearly, the former problem can be written in the form (5) by considering (2) and all maximal extensions of an empty subsystem. For the latter problem, note that the vertices of the polytope $\{x \in \mathbb{R}^n \mid Ax = b,\ x \geq 0\}$, where $b \neq 0$, are in one-to-one correspondence with the maximal feasible extensions of the subsystem $Ax = b,\ x \geq 0$ in the infeasible system $Ax = b,\ x \geq 0,\ x \leq 0$. Although the general problem, of generating maximal feasible extensions, is NP-hard as shown above, the special cases of generating simplices and anti-simplices remain open.

2 Proof of Theorem 1

In this section we prove Theorem 1 by a reduction from satisfiability, a well-known NP-complete problem (see Cook [1971]).

Let us consider n propositional Boolean variables X_j, $j = 1,..,n$, denote by $\overline{X} = 1 - X$ the negation of X, call variables and their negations *literals*, and elementary disjunctions of literals *clauses*. Let us next consider an arbitrary conjunctive normal form (CNF) $\phi = C_1 \wedge C_2 \ldots \wedge C_m$, i.e., where C_i, $i = 1,...,m$ are clauses. A truth assignment to the variables is called *satisfying* for the CNF ϕ, if ϕ evaluates to true, i.e., if at least one literal evaluates to true in each of the clauses of ϕ.

In what follows, we shall associate to ϕ a weighted directed graph (G, w) and set $\mathcal{X}$ of negative cycles of G such that (G, w) has a negative cycle not belonging to $\mathcal{X}$ if and only if ϕ has a satisfying assignment. Because (G, w) and $\mathcal{X}$ is constructed from ϕ in $O(mn)$ time, and the weight function w uses only two different values (1 and -2), Theorem 1 follows readily from this construction by Cook [1971], since the decision problem "*Is there a negative cycle in* (G, w) *which does not belong to* $\mathcal{X}$*?*" clearly belongs to NP. To complete the proof of Theorem 1, we provide below a construction with these properties, such that there is a one-to-one correspondence between satisfying assignments to ϕ and negative cycles of (G, w) which not belong to $\mathcal{X}$.

To describe our construction, let us denote for $j = 1,...,n$ respectively by o_j and $\bar{o}_j$ the number of occurrences of literal X_j and its negation $\overline{X}_j$, denote by x_j^k the kth occurrence of X_j, $k = 1,...,o_j$, and by $\bar{x}_j^k$ the kth occurrence of $\overline{X}_j$, $k = 1,...,\bar{o}_j$, and let L denote the set of all literal occurrences, i.e.,

$$|L| = \sum_{i=1}^{m} |C_i| = \sum_{j=1}^{n} o_j + \bar{o}_j.$$

Since monotone literals can be easily eliminated from a satisfiability problem, we can assume without any loss of generality that $o_j > 0$ and $\bar{o}_j > 0$ hold for all variables $j = 1,...,n$.

For instance, if $n = 3$ and

$$\phi = (X_1 \vee X_2 \vee \overline{X}_3) \wedge (X_1 \vee \overline{X}_2 \vee X_3) \wedge (\overline{X}_1 \vee X_2 \vee \overline{X}_3), \tag{6}$$

then we have $o_1 = 2$, $\bar{o}_1 = 1$, $o_2 = 2$, $\bar{o}_2 = 1$, $o_3 = 1$, $\bar{o}_3 = 2$,

$$L = \{x_1^1, x_2^1, \bar{x}_3^1, x_1^2, \bar{x}_2^1, x_3^1, \bar{x}_1^1, x_2^2, \bar{x}_3^2\}$$

and

$$\phi' = (x_1^1 \vee x_2^1 \vee \bar{x}_3^1) \wedge (x_1^2 \vee \bar{x}_2^1 \vee x_3^1) \wedge (\bar{x}_1^1 \vee x_2^2 \vee \bar{x}_3^2).$$

We define the vertex set of the graph $G = (V, E)$ associated to ϕ as

$$V \;=\; U \cup Q \cup \bigcup_{j=1}^{n} (Y_j \cup Z_j),$$

where U, Q, and Y_j and Z_j for $j = 1, ..., n$ are pairwise disjoint, defined as

$$\begin{aligned} U &= \{u_k \mid k = 0, 1, ..., m+n\}, \\ Q &= \{a(\ell), b(\ell) \mid \ell \in L\}, \\ Y_j &= \{y_{jk} \mid k = 1, ..., o_j - 1\} \;\text{ for } j = 1, ..., n, \;\text{ and} \\ Z_j &= \{z_{jk} \mid k = 1, ..., \bar{o}_j - 1\} \;\text{ for } j = 1, ..., n. \end{aligned}$$

The graph itself has a ring structure, the skeleton of which is the set U. For every variable X_j of ϕ we have two parallel directed paths from u_{j-1} to u_j. The first path corresponding to X_j contains vertices Y_j (and some other vertices), while the second path, corresponding to $\overline{X}_j$ passes through vertices of Z_j ($j = 1, ..., n$). For convenience, we also introduce the notations

$$y_{j0} = z_{j0} = u_{j-1} \quad \text{and} \quad y_{j,o_j} = z_{j,\bar{o}_j} = u_j \tag{7}$$

for $j = 1, ..., n$. To every clause C_i of ϕ we associate $|C_i|$ parallel directed paths from u_{n+i-1} to u_{n+i}, one for each of the literals in C_i ($i = 1, ..., m$). Finally vertices $a(\ell)$ and $b(\ell)$ correspond exclusively to literal occurrence $\ell \in L$.

Let us consider next the weighted graph $H(a, b, p, q, r, s)$ (see Figure 1) on six nodes a, b, p, q, r and s, having six arcs, the weights of which are as follows:

$$\begin{aligned} w(a, b) \;=\; w(b, a) \;=\; -2 \quad \text{and} \\ w(p, a) \;=\; w(b, q) \;=\; w(r, b) \;=\; w(a, s) \;=\; 1. \end{aligned} \tag{8}$$

To every literal occurrence $\ell \in L$ we associate a disjoint copy of $H(a, b, p, q, r, s)$, and denote by $a(\ell)$, $b(\ell)$, etc., its nodes, and by E_ℓ its arc set. Note that each of these small subgraphs can be decomposed into two directed paths of $3 - 3$ arcs, $E_\ell = E_\ell^v \cup E_\ell^c$, where

$$\begin{aligned} E_\ell^v &= \{\; (p(\ell), a(\ell)),\; (a(\ell), b(\ell)),\; (b(\ell), q(\ell)) \;\}, \text{ and} \\ E_\ell^c &= \{\; (r(\ell), b(\ell)),\; (b(\ell), a(\ell)),\; (a(\ell), s(\ell)) \;\}. \end{aligned}$$

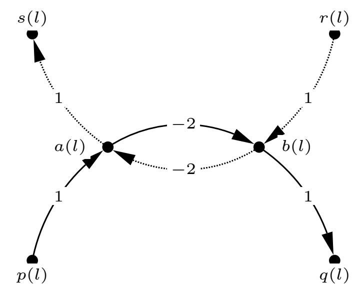

Figure 1: The directed graph $H(a, b, p, q, r, s)$ associated with literal occurrences.

Finally we set

$$E = E_0 \cup \bigcup_{\ell \in L} E_\ell$$

where $E_0 = \{(u_{m+n}, u_0)\}$ with weight $w(u_{m+n}, u_0) = -2$.

In each of the subgraphs corresponding to the literal occurrences $\ell \in L$, we have the nodes $a(\ell)$ and $b(\ell)$ already introduced in $Q \subseteq V$, while the nodes $p(\ell)$, $q(\ell)$, $r(\ell)$ and $s(\ell)$ for $\ell \in L$ are corresponding to some other vertices of G, according to the following definitions:

$$\begin{array}{llll} p(\ell) = y_{j,k-1} & \text{and} & q(\ell) = y_{jk} & \text{if } \ell = x_j^k, \\ p(\ell) = z_{j,k-1} & \text{and} & q(\ell) = z_{jk} & \text{if } \ell = \bar{x}_j^k, \text{ and} \\ r(\ell) = u_{n+i-1} & \text{and} & s(\ell) = u_{n+i} & \text{if } \ell \in C_i. \end{array}$$

In other words, for every literal occurrence ℓ of clause C_i the set E^c_ℓ forms a 3-arc directed path from u_{n+i-1} to u_{n+i}. Furthermore by (7) and by the above definitions, the sets E^v_ℓ for $\ell = x_j^1, x_j^2, ..., x_j^{o_j}$ form a directed path from u_{j-1} to u_j through the vertices of Y_j, consisting of $3o_j$ arcs, for every variable X_j. Similarly, the sets E^v_ℓ for $\ell = \bar{x}_j^1, \bar{x}_j^2, ..., \bar{x}_j^{\bar{o}_j}$ form another directed path from u_{j-1} to u_j through the vertices of Z_j, consisting of $3\bar{o}_j$ arcs.

In summary, $G = (V, E)$ consists of $|V| = 3|L| + m - n + 1$ vertices and $|E| = 6|L| + 1$ arcs, and the weight function w takes only values in $\{-2, 1\}$.

Returning to the example CNF ϕ given in (6), the corresponding graph $G = (V, E)$ is shown in Figure 2. To make the drawing of such a graph visually more clear, nodes $a(\ell)$ and $b(\ell)$ of G are represented by two-two separate points of the picture, labeled as $a(\ell)$ and $a'(\ell)$, and as $b(\ell)$ and $b'(\ell)$, respectively. Similarly, node u_n is represented by two points in the figure, labeled by u_n and u'_n. Arcs in the sets E^c_ℓ for $\ell \in L$ are drawn as dashed lines, while those belonging to E^v_ℓ for $\ell \in L$ are drawn as solid lines.

Let us observe first that the arcs $(a(\ell), b(\ell))$ and $(b(\ell), a(\ell))$ form a directed cycle of total weight -4 for every literal occurrence $\ell \in L$. Let us denote by $\mathcal{X}$ the set of these directed cycles, i.e., $|\mathcal{X}| = |L|$, and let us denote by $\mathcal{F}$ the set of all directed negative cycles of G.

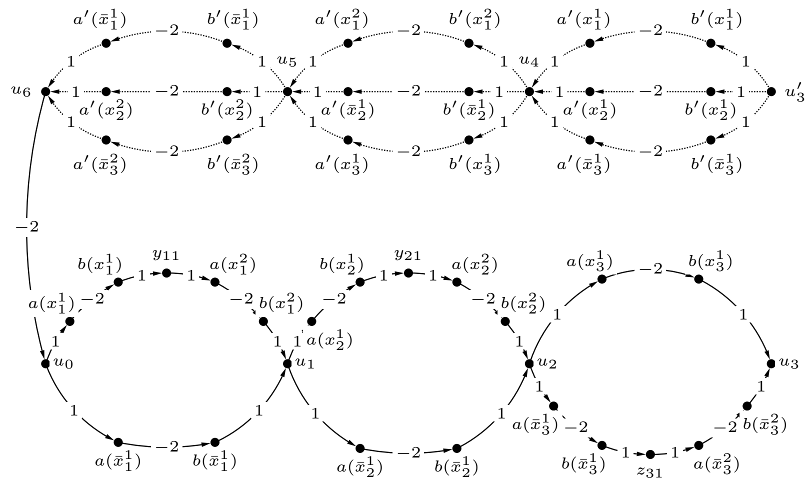

Figure 2: G is obtained by identifying vertices $a(l)$, $a'(l)$, and $b(l)$, $b'(l)$, for each literal l of ϕ', and u_3, u_3' in the graph above.

We claim that from every satisfying assignment X of ϕ we can construct a directed negative cycle $D^X \in \mathcal{F} \setminus \mathcal{X}$, and conversely, from every directed negative cycle $D \in \mathcal{F} \setminus \mathcal{X}$ we can construct a satisfying assignment X^D of ϕ. As we noted at the beginning of this section, this claim implies Theorem 1.

To see this claim, let us first consider a satisfying assignment $X = (X_1, ..., X_n) \in \{0,1\}^n$ of ϕ. Since X satisfies ϕ, we have a literal $\ell_i \in C_i$ in every clause $i = 1, ..., m$ such that ℓ_i evaluates to true at X (i.e., $\ell_i(X) = 1$). Let us also denote by W the set of all those literal occurrences, which evaluate to false at X, i.e., $W = \{\ell \in L \mid \ell(X) = 0\}$. Clearly, $\ell_i \notin W$ for $i = 1, ..., m$ by the above definitions. Then, the set of arcs

$$D^X = \left(\bigcup_{i=1}^{m} E^c_{\ell_i}\right) \cup \left(\bigcup_{\ell \in W} E^v_\ell\right) \cup \{(u_{m+n}, u_0)\}.$$

forms a simple directed cycle in G not belonging to $\mathcal{X}$. Since we have $w(E^c_\ell) = w(E^v_\ell) = 0$ for all literal occurrences $\ell \in L$, it follows by the above definitions that $w(D^X) = w(u_{m+n}, u_0) = -2$, i.e., $D^X \in \mathcal{F} \setminus \mathcal{X}$ as claimed.

Before proving the reverse direction of our main claim, let us first observe some simple properties of our construction. To simplify notation, recall that $E_\ell = E^c_\ell \cup E^v_\ell$ for $\ell \in L$, and that the 6-vertex subgraphs induced by the arc set E_ℓ have the same structure and weights, as in Figure 1, for all $\ell \in L$. The following property of these subgraphs will be instrumental in our proof.

Lemma 1 *Given a simple directed cycle $D \subseteq E$ of G, not belonging to $\mathcal{X}$, and given a literal occurrence $\ell \in L$, we have*

$$w(D \cap E_\ell) \in \{0, 2, 4\}.$$

Moreover, $w(D \cap E_\ell) = 0$ *only if the set* $D \cap E_\ell$ *is one of the following three subsets of* E_ℓ*:* E^c_ℓ*,* E^v_ℓ*, or* $\emptyset$*.*

Proof: Since D is a simple cycle not belonging to $\mathcal{X}$, D cannot contain both arcs $(a(\ell), b(\ell))$ and $(b(\ell), a(\ell))$. Thus, denoting by $A_\ell = \{(p(\ell), a(\ell)), (a(\ell), s(\ell))\}$ and $B_\ell = \{(r(\ell), b(\ell)), (b(\ell), q(\ell))\}$ we have that $D \cap E_\ell$ is one of the following six sets: $\emptyset$, A_ℓ, B_ℓ, $A_\ell \cup B_\ell$, E^c_ℓ, and E^v_ℓ. Since we have $w(\emptyset) = w(E^c_\ell) = w(E^v_\ell) = 0$, $w(A_\ell) = w(B_\ell) = 2$ and hence $w(A_\ell \cup B_\ell) = 4$, the statement follows. □

Returning to the reverse direction of our main claim, let us consider a simple directed negative cycle $D \in \mathcal{F} \setminus \mathcal{X}$ of G. Since

$$w(D) = \sum_{\ell \in L} w(D \cap E_\ell) + w(D \cap \{(u_{m+n}, u_0)\})$$

we must have by Lemma 1 that $(u_{m+n}, u_0) \in D$ and

$$w(D \cap E_\ell) = 0 \quad \text{for all} \quad \ell \in L. \tag{9}$$

We show first that D passes through all vertices in U, includes exactly one of the two parallel paths between u_{j-1} and u_j for $j = 1, ..., n$, and exactly one of the parallel paths between u_{n+i-1} and u_{n+i} for all $i = 1, ..., m$.

As we observed above, we have u_0 as a vertex of D. Thus D must contain an arc leaving u_0, say it contains $(u_0, a_{x^1_1})$. Then, by (9) and by Lemma 1 we must have $E^v(x^1_1) \subseteq D$, i.e., D must pass through vertex y_{11}. Since only $(y_{11}, a(x^2_1))$ is leaving y_{11}, by repeating the above argument we can conclude that we must also have $E^v_{x^2_1} \subseteq D$, etc., finally arriving to $E^v_{x^{o_1}_1} \subseteq D$, i.e., that D includes u_2 as a vertex. Repeating the same argument, we can prove by induction that for all indices $j = 1, ..., n$, if $E^v_{x^1_j} \subseteq D$, then we must have $E^v_{x^k_j} \subseteq D$ for all $k = 1, ..., o_j$, and that if $E^v_{\bar{x}^1_j} \subseteq D$, then we must also have $E^v_{\bar{x}^k_j} \subseteq D$ for all $k = 1, ..., \bar{o}_j$. Let us then define a truth assignment X^D by

$$X^D_j = \begin{cases} 1 \text{ if } E^v_{\bar{x}^1_j} \subseteq D, \\ 0 \text{ if } E^v_{x^1_j} \subseteq D. \end{cases}$$

Furthermore, repeating a similar argument for vertices u_n, u_{n+1}, ..., u_{n+m-1}, u_{n+m} we can also conclude that D must contain the set $E^c_{\ell_i}$ for exactly one of the literals $\ell_i \in C_i$, for each clauses C_i of ϕ. Since D is a simple cycle in which no vertex $a(\ell)$ or $b(\ell)$ is repeated, we must have that $\ell_i(X^D) = 1$ for all $i = 1, ..., m$, i.e., that X^D is indeed a satisfying assignment of ϕ.

These observations prove the reverse direction of our main claim, and hence conclude the proof of Theorem 1. □

3 Proof of Theorem 2

We can repeat essentially the same proof as for the directed case, with the exception that we define $w(u_0, u_{m+n}) = -1$ now, and associate with every literal occurrence $\ell \in L$ a different subgraph denoted by E_ℓ: Let us associate now with $\ell \in L$ six nodes, $a = a(\ell), b = b(\ell), c = c(\ell), d = d(\ell), e = e(\ell_)$, and $f = f(\ell)$, and and the following 10 edges

$$E_\ell = \{(a, b), (b, c), (c, d), (d, e), (e, f), (a, f), (a, p), (b, q), (d, r), (e, s)\},$$

where nodes $p = p(\ell)$, $q = q(\ell)$, $r = r(\ell)$ and $s = s(\ell)$ are identified with the other nodes of G, in the same way as in the previous proof. To simplify notation, we shall omit the reference to ℓ, whenever it is clear from the context which literal occurrence we talk about. The weights of the edges of E_ℓ are defined as

$$w(a, p) = w(b, q) = w(d, r) = w(e, s) = \tfrac{5}{2}, \text{ and}$$
$$w(a, b) = w(b, c) = w(c, d) = w(d, e) = w(e, f) = w(a, f) = -1.$$

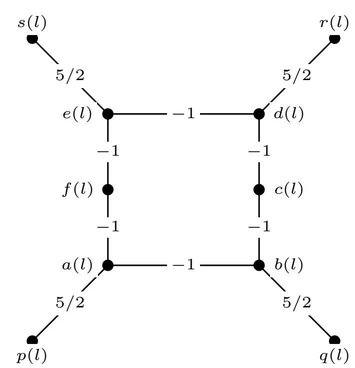

Figure 3: The undirected graph associated with literal occurrences.

Let us note that in each of these subgraphs there is a negative cycle (see Figure 3), formed by the six edges $D_\ell = \{(a, b), (b, c), (c, d), (d, e), (e, f), (a, f)\}$. Let us denote by $\mathcal{X} = \{D_\ell \mid \ell \in L\}$ the collection of these negative cycles, and let $\mathcal{F}$ denote the family of all negative cycles in G.

By an analogous proof as in the previous section, we can show that there exists a negative cycle belonging to $\mathcal{F} \setminus \mathcal{X}$ if and only if ϕ has a satisfying assignment. The key observation in this case, the analogue of Lemma 1, is the following claim, which can easily be verified e.g., by looking at Figure 3.

Lemma 2 *For a simple cycle D of G not belonging to $\mathcal{X}$ and literal occurrence $\ell \in L$ we have*

$$w(D \cap E_\ell) \in \{0, 1, 2, 4\}$$

and it is equal to 0 *only if* $D \cap E_\ell$ *is one of the following there sets:* $\emptyset$,

$$E_\ell^v = \{(b,c),(c,d),(d,e),(e,f),(a,f),(a,p),(b,q)\}, \text{ or}$$
$$E_\ell^c = \{(a,b),(b,c),(c,d),(e,f),(a,f),(d,r),(e,s)\}.$$

□

References

S.D. Abdullahi. *Vertex Enumeration and Counting for Certain Classes of Polyhedra*. PhD thesis, Computing (Computer Algorithms) Leeds University U.K., 2003.

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

D. Avis and K. Fukuda. A pivoting algorithm for convex hulls and vertex enumeration of arrangements and polyhedra. *Discrete and Computational Geometry*, 8(3):295–313, 1992. ISSN 0179-5376.

D. Avis and K. Fukuda. Reverse search for enumeration. *Discrete Applied Mathematics*, 65 (1-3):21–46, 1996. ISSN 0166-218X. doi: http://dx.doi.org/10.1016/0166-218X(95)00026-N.

M.L. Balinski. An algorithm for finding all vertices of convex polyhedral sets. *SIAM Journal on Applied Mathematics*, 9:72–81, 1961.

E. Boros, K. Elbassioni, V. Gurvich, and L. Khachiyan. Enumerating minimal dicuts and strongly connected subgraphs and related geometric problems. In D. Bienstock and G. Nemhauser, editors, *Integer Programming and Combinatorial Optimization, 10th International IPCO Conference, New York, NY, USA*, volume 3064 of *Lecture Notes in Computer Science*, pages 152–162, Berlin, Heidelberg, New York, June 7-11 2004. Springer. (RRR 36-2003).

D. Bremner. Incremental convex hull algorothms are not output sensitive. *Discrete and Computational Geometry*, 21:57–68, 1999.

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

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(2):103–109, 1998.

N. Chakravarti. Some results concerning post-infeasibility analysis. *European Journal of Operations Research*, 73:139–143, 1994.

D.R. Chand and S.S. Kapur. An algorithm for convex polytopes. *Journal of the Association for Computing Machinery*, 17(1):78–86, 1970. ISSN 0004-5411. doi: http://doi.acm.org/10.1145/321556.321564.

A. Charnes, W.W. Cooper, and A. Henderson. *An Introduction to Linear Programming*. Wiley, New York, 1953.

V. Chvátal. *Linear Programming*. Freeman, San Francisco, CA, 1983.

S.A. Cook. The complexity of theorem proving procedures. In *Proceedings of the Third Annual ACM Symposium on Theory of Computing*, pages 151–158, 1971.

M.E. Dyer. The complexity of vertex enumeration methods. *Mathematics of Operations Research*, 8:381–402, 1983.

M.E. Dyer and L.G. Proll. An algorithms for determining all extreme points of a convex polytope. *Mathematical Programming*, 12:81–96, 1977.

T. Eiter and G. Gottlob. Identifying the minimal transversals of a hypergraph and related problems. *SIAM Journal on Computing*, 24:1278–1304, 1995.

J. Farkas. Theorie der einfachen ungleichungen. *Journal für die reine und angewandte Mathematik*, 124:1–27, 1901.

M. Fredman and L. Khachiyan. On the complexity of dualization of monotone disjunctive normal forms. *Journal of Algorithms*, 21:618–628, 1996.

K. Fukuda, Th. M. Liebling, and F. Margot. Analysis of backtrack algorithms for listing all vertices and all faces of a convex polyhedron. *CGTA*, 8:1–12, 1997.

T. Gallai. Maximum-minimum sätze über graphen. *Acta Mathematicae, Academiae Scientiarum Hungaricae*, 9:395–434, 1958.

J. Gleeson and J. Ryan. Identifying minimally infeasible subsystems of inequalities. *ORSA Journal on Computing*, 2(1):61–63, 1990.

D.S. Johnson and F.P. Preparata. The densest hemisphere problem. *Theoretical Computer Science*, 6:93–107, 1978.

R. Karp. A characterization of the minimum cycle mean in a digraph. *Discrete Mathematics*, 23:309–311, 1978.

L. Khachiyan. A polynomial algorithm in linear programming. *Soviet Math. Dokl.*, 20: 191–194, 1979.

E. Lawler, J. K. Lenstra, and A. H. G. Rinnooy Kan. Generating all maximal independent sets: NP-hardness and polynomial-time algorithms. *SIAM Journal on Computing*, 9: 558–565, 1980.

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

T.H. Mattheiss. An algorithm for determining irrelevant constraints and all vertices in systems of linear inequalities. *Operations Research*, 21:247–260, 1973.

T.S. Motzkin, H. Raiffa, G.L. Thompson, and R.M. Thrall. The double description method. In H.W. Kuhn and A.W. Tucker, editors, *Contributions to the Theory of Games*, volume II, pages 51–73, 1953.

M. E. Pfetsch. *The Maximum Feasible Subsystem Problem and Vertex-Facet Incidences of Polyhedra*. Dissertation, TU Berlin, 2002.

J.S. Provan. Efficient enumeration of the vertices of polyhedra associated with network lp's. *Mathematical Programming*, 63(1):47–64, 1994. ISSN 0025-5610.

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

J. Ryan. IIS-hypergraphs. *SIAM Journal on Discrete Mathematics*, 9(4):643–653, 1996.

R. Seidel. *Output-size sensitive algorithms for constructive problems in computational geometry*. Computer science, Cornell University, Ithaka, NY, 1986.

G. Swart. Finding the convex hull facet by facet. *Journal of Algorithms*, 6:17–48, 1985.

L.G. Valiant. The complexity of enumeration and reliability problems. *SIAM Journal on Computing*, 8:410–421, 1979.