Les cahiers
du laboratoire
Leibniz

# Integer polyhedra : Combinatorial Properties and Complexity

A. Sebo

Laboratoire Leibniz-IMAG, 46 av. Félix Viallet, 38000 GRENOBLE, France -
ISSN : 1298-020X

n° 24
Mai 2001

# Integer polyhedra : Combinatorial Properties and Complexity

András Sebő *

May 30, 2001

**Abstract**

A polyhedron having vertices is called *integer* if all of its vertices are integer. This property is co$\mathcal{NP}$-complete in general. Recognizing integral set-packing polyhedra is one of the biggest challenges of graph theory (perfectness test). Various other special cases are major problems of discrete mathematics.

The focus of this talk is *not* the recognition of classes of integer polyhedra. We aim at communicating their relevant properties, and satisfactory alternatives to their recognition, whenever this latter is difficult; for instance,

- combinatorial or geometric properties related to integrality; connections between integer polyhedra, graphs and numbers;
- a general co$\mathcal{NP}$ characterization of minimal noninteger structures that contains the known special cases for set-packing, set-covering, and particular mixtures of these.
- complexity questions related to integer polyhedra, some of which do not fit into the P-NPC axis.

We are giving priority to some topics that may need to be further explored.

*CNRS, Leibniz-IMAG, 38000 Grenoble, France, Andras.Sebo@imag.fr

# Introduction

A polyhedron having vertices is called *integer* if all of its vertices are integer. It is co-$\mathcal{NP}$-complete to check in general whether a polyhedron $\{x \in \mathbb{R}^n : Ax \leq b\}$ is integer [87], even if the constraint matrix is $0-1$ and the polyhedron is in the nonnegative orthant. However, it is one of the biggest challenges of graph theory to decide the complexity of this problem if the right hand side is also 1. (This is polynomially (and easily) equivalent to certifying the perfectness of a graph.) Various other special cases are major problems in different subfields of discrete mathematics.

The significance of this problem is obvious: when optimizing on polyhedra having vertices, there is always an optimal solution which is a vertex. Therefore on integer polyhedra linear programming, which is solvable in polynomial time by [64] is equivalent to integer programming, in turn $\mathcal{NP}$-hard in general.

We do *not* provide in this paper a full classification of properties that ensure a polyhedron to be integer, even if we provide a survey of a relevant part of the results; we do certainly not want to restrict ourselves to such characterizations; we will not treat integer programming algorithms (for an introduction to various aspects of integer programming see [99], [80] [84] etc.), or provide efficient tests about the integrality of the polyhedron defined by a program, even if we will point at some of these algorithms. We will not deal with the structure of polytopes from the convex geometry viewpoint (which does not take into account integrality) ; we do not decompose problems that behave well with respect to some integer programs; the complexity will not mean classifying the arising problems into polynomially solvable and NP-hard ones, even if we will mention the most crucial results; we will also not analyse the complexity of particular polytopes such as the travelling salesman, or the stable-set polytope.

We wish to show combinatorial $\mathcal{NP}$ and *co*$\mathcal{NP}$ characterization theorems for integrality in some interesting cases; some of these involve elementary properties of numbers, some others structural properties of hypergraphs, etc.

We will explain some combinatorial aspects of integer polyhedra, some of which have come to the surface recently, and point at some new challenges concerning these, in particular

– explain purely combinatorial or algebraic formulations of structural questions concerning simple objects such as parallelepipeds and simplices. These come up as helpful tools in more general questions about integer polyhedra. We summarize the state of the art about the Hilbert-property of cones, together with the most recent developments and the questions that remain. We arrive at a simple set of problems to study concerning the structure of integer parallelepipeds and simplices.

– show some common features of integer packing, covering and of the corresponding minimal noninteger structures that *provide co*$\mathcal{NP}$ *characterizations* for such problems, including the intersection of packing and covering problems. We survey some subclasses of integer polyhedra and state the main open problems about them.

– exhibit some new complexity results and questions concerning integer polyhedra, including some new occurances of 'total search problems', where the searched object always exists, but it is not evident to find. Such a problem may also be 'complete' in a well defined complexity class containing difficult problems. This is a new tool proposed by Papadimitriou and his coauthors to show that *finding* some combinatorial objects might be desperately difficult, even if *deciding* the existence of the same object is easy.

We will not go into the details of algorithms, which does not mean that we do not care about them. *The main benefit of the integrality of a polyhedron is that integer programming problems collaps to linear programming for them.* Therefore the main customers of results about integer polyhedra are integer programmers. The results we study have the same type of relation to integer programming algorithms as for instance 'total unimodularity': they are not directly concerned with algorithms, but many of them turn out to be helpful.

We will be satisified with a brief survey and the main references about more or less closed theories, and give priority to open subjects.

The role played by simple combinatorial arguments involving the divisibility of numbers in the existence of integer solutions to linear programs is not surprising: even the simplest integer program $ax = b$, $x \geq 0$, $(a, b, x$ are integers) involves divisibility. Indeed, it has a solution if and only if $a|b$ and $a$, $b$ have the same sign.

We suppose basic knowledge of linear algebra, and some definitions and facts from [99], or [28]. A *lattice* generated by a finite set $B$ of integer vectors is the set of linear combinations of vectors in $B$ with integer coefficients. Lattices can always be generated by linearly independent vectors. In the questions we are studying it is usually not an essential restriction of generality to suppose that the underlying lattice is the lattice of integers – the generalizations to arbitrary lattices are straightforward. The greatest common divisor and the least common multiple of the integers $a_1, \ldots, a_n \in \mathbb{Z}$ will be denoted by $\gcd(a_1, \ldots, a_n)$, and $\mathrm{lcm}(a_1, \ldots, a_n)$ respectively.

In Section 1 we treat the simplest possible integer polyhedra: (shifted) linear subspaces, cones, parallelepipeds, simplices.

A set $S \subseteq \mathbb{R}^n$ is called a *simplex*, if $S = \mathrm{conv}\{0, v_1, \ldots, v_n\}$, where $0 \in \mathbb{R}^n$ and $v_1, \ldots, v_n$ are linearly independent. (In other words a simplex is a polytope whose vertices are affinely independent; our definition restricts a simplex to be full dimensional, and to have 0 as one of its vertices.) Under the same condition, $\mathrm{cone}\{v_1, \ldots, v_n\}$ is called a *simplicial cone.*

For simplices, testing integrality itself is trivial, on the other hand if a simplex is integer, finding out about other integer points seems to be difficult. Moreover, this turns out to provide a common language to more and more relevant problems.

In Section 2 we define some classes of polyhedra and characterize them with some combinatorial properties. After a short survey of the classical theorems we would like to show some more recent results on packing, covering and their intersection.

In Section 3 we study some complexity questions related to integer polyhedra. One of the complexity measures is provided by cutting plane methods, and these lead also to some computational questions. We then show some integer polyhedra for which an integer solution can be desperately difficult to find even if we surely know and can certify that one exists (and therefore they cannot be $\mathcal{NP}$-complete).

Summarizing: we will exhibit some of the *combinatorial properties underlying the integrality of polyhedra*, along with some *new computational complexity phenomena behind them.*

# 1 The Simplest Integer Polyhedra

The simplest possible polyhedra from the point of view of integrality, are shifted (affine) linear subspaces. The structure of integer points of cones or parallelepipeds lead to some simple questions about Abelian groups. These also provide natural tools for treating some most basic questions concerning simplices, of which occur to be difficult.

Parallelepipeds are a useful intermediate tool implying at the same time the basic simple affine subspaces as simple corollaries and also acting as a stepping stone towards simplices and general polyhedra. Therefore we make parallelepipeds the main subject of this section, and show several examples of their concrete use. Among these we will touch in Section 1.2 the well-known and easier affine subspaces and in 1.2.2, 1.2.3 several other problems are formulated as special cases.

For the sake of an example, let us forecast how simplices will occur as a special case of parallelepipeds. It is trivial to decide whether a simplex is integer, no matter how it is given: any natural way of providing the input for a simplex allows computing the coordinates of all of its vertices and all of its facets. (We do not wish to deal with the computational problems related to approximation of irrational data.)

However, the second question one can ask, seems to be already open in $\mathbb{R}^4$:

**Problem 1** *[102] Given an integer simplex $S$, is the property that $S$ has no integer point besides its vertices in $\mathcal{NP}$ ? Moreover, can it be decided in polynomial time ?*

We will call an integer simplex containing no integer point besides its vertices *empty*. For the results in $\mathbb{R}^3$ see [94], [96], [124], [102].

This will arise as a special case of finding an integer vector in a cone generated by a set of linearly independent vectors, for which the (uniquely determined) coefficients satisfy some particular inequalities. The main point is *to mix equations modulo n with linear inequalities*. We believe that this is an interesting problem; we will show some applications of it, and we believe that many others will show up in the future. This explains why we spend a relatively big amount of space to an introduction to parallelepipeds.

## 1.1 Parallelepipeds

Let $V \subseteq \mathbb{Z}^n$, $V = \{v_1, \ldots, v_n\}$ be linearly independent, and define

$$\mathrm{par}(V) := \{x \in \mathbb{Z}^n : x = \sum_{i=1}^{n} \lambda_i v_i : 0 \leq \lambda_i < 1\,(v \in V)\},$$

and call it a *parallelepiped.* If in addition $|V| = n$, then $\det(V)$ denotes the *absolute value of the determinant* of the matrix whose rows are the elements of $V$. More generally, if $|V| < n$ then let $\det(V)$ denote the greatest common divisor of the $|V| \times |V|$ subdeterminants of the matrix whose rows are the elements of $V$. For $x \in \mathbb{R}^n$ the vector $\mathrm{coeff}(x, V) := \lambda$, where $\lambda$ is defined by the unique combination $x = \lambda_1 v_1 + \ldots + \lambda_n v_n$), will be called the $V$*-coefficient* vector of $x$.

**Theorem 1.1** *Let $V$ be as above. Then $|\mathrm{par}(V)| = \det(V)$, in particular,*

$$\mathrm{par}(V) = \{0\} \text{ if and only if } \det(V) = 1.$$

*Moreover, $x \in \mathbb{Z}^n$ if and only if det$(V)\,\mathrm{coeff}(x, V)$ is integer; for all $x \in \mathbb{Z}^n$ there exists a unique vector $x' \in \mathrm{par}(V)$, such that $x - x'$ is on the lattice generated by $V$; $\mathrm{coeff}(x', V)$ is the residue vector of* $\mathrm{coeff}(x', V) \bmod 1$.

Let us call the $x'$ defined in the theorem the *residue* of $x$ mod $V$, and denote it $\mathrm{mod}(x, V)$.

The main content is $|\mathrm{par}(V)| = \det(V)$, which is a basic and often used fact in the geometry of numbers, and can be proved in several essentially different ways (see for instance Cassels [20]). It is basic and well-known. We provide a full elementary proof (sketched in [100]), in order to show the underlying combinatorial structure, and to be prepared for a translation between group terminology and polyhedral combinatorics:

**Proof.** Let $V = \{v_1, \ldots, v_m\} \subseteq \mathbb{Z}^n$, $(m, n \in \mathbb{N})$, and let $M_V$ be the $m \times n$ matrix whose $i$-th row is $v_i$, $(i = 1, \ldots, m)$. Denote by $\Lambda(V)$ the set of $V$-coefficients of integer vectors multiplied by $\det(V)$. For square matrices the rest is evident from 'Cramer's rule', and for arbitrary matrices it will also be an easy byproduct of the proof. Let $a_i$ denote the $i$-th column of $M_V$, $(i = 1, \ldots, n)$. The following fact is obvious:

**Claim 1**: *Replacing a column $a_i$ by $a_i \pm a_j$ $(i \neq j)$, or interchanging $a_i$ and $a_j$ and denoting the new row-set by $V'$,*

$$\Lambda(V) = \Lambda(V'), \text{ moreover, } |\operatorname{par}(V)| = |\operatorname{par}(V')|$$

Let $\lambda = det(V)(\lambda_1, \ldots, \lambda_m) \in \mathbb{Z}^m$ be the $V$-coefficient of an integer vector. Replace a column $a_i$ by $a_i \pm a_j$ $(i \neq j)$, or interchange two columns; denoting the new row-set by $V'$, $\lambda$ is also the $V'$-coefficient of an integer vector. $\Lambda(V) \subseteq \Lambda(V')$ follows. Since the operations in this claim are reversible, we have in fact $\Lambda(V) = \Lambda(V')$. Moreover, there is a bijection between $\operatorname{par}(V)$ and the set

$$\Lambda_0 := \{\lambda = (\lambda_1, \ldots, \lambda_m) \in \Lambda(V) : 0 \geq \lambda_i < 1\},$$

and also between $\operatorname{par}(V')$ and this set: we have therefore $|\operatorname{par}(V)| = |\operatorname{par}(V')|$.

In other words, $\Lambda_0$ does not change through the operations of Claim 1, and we also know that the determinant does not change. With such operations, called *elementary column operations* one can easily arrive at a lower triangular matrix $M_U$ (every entry is 0 except the lower $m \times m$ corner), where $U = \{u_1, \ldots, u_m\}$ denotes the set of the row vectors of the obtained matrix. Requiring in addition that the matrix is nonnegative and the diagonal element of each row is the unique maximum of the row, the resulting matrix is unique and is called the *Hermite normal form* of $M_V$, see [99] p.45. (All this is straightforward to see. It can also be determined in polynomial time with some extra work, as well as the Smith normal form below, see [72], [99].)

It is immediate to check that the greatest common divisor of the $m \times m$ determinants does not change during the procedure, and at the end, it is equal to the product of the entries in the main diagonal.

Let the elements in the main diagonal of $M_U$ be $d_1, \ldots d_m$. We have proved:

**Claim 2**: $|\operatorname{par}(V)| = |\operatorname{par}(U)|$, and $\det(V) = \det(U) = d_1 d_2 \ldots, d_m$.

Therefore we are remained only with:

**Claim 3**: $|\operatorname{par}(U)| = d_1 d_2 \ldots, d_m$.

Let $\lambda \in \mathbb{R}^D$. For $M_D\lambda$ to be in $\operatorname{par}(v_1, \ldots, v_m)$, we have $d_m$ different choices for $\lambda_m$: $\frac{t}{d_m}$ $(t = 0, \ldots, d_m - 1)$. Similarly, if $\lambda_m, \ldots, \lambda_{m-i+1}$ have already been chosen, and the $m - i$-th component of $\sum_{j=0}^{i-1} \lambda_{m-j} a_{m-j}$ is $x$, then the possible choices for $\lambda_{m-i}$ are restricted by the fact that that $m-i$-th coordinate of the result does not depend on rows with smaller index than $m-i$:

$\lceil x \rceil - x + \frac{t}{d_{m-i}}$, $(t = 0, \ldots, d_{m-i} - 1)$: for all possible choices of $\lambda_m, \ldots, \lambda_{m-i+1}$ we have exactly $d_{m-i}$ choices for $\lambda_{m-i}$. We conclude that $\mathrm{par}(u_1, \ldots, u_m)$ has $d_1 \ldots d_m$ elements.

In the additional claim the only nontrivial fact is that the $V$-coefficient vector $det(V)(\lambda_1, \ldots, \lambda_n)$ of $x$ is integer. Equivalently, the denominators of the coefficients in a combination $x = \alpha_1 v_1 + \ldots + \alpha_n v_n$ all divide $\det V$. However, because of Claim 1, $\Lambda(V) = \Lambda(U)$, and it is clear that $d_1 \ldots d_n$ is a common denominator of the solutions exhibited in the proof of Claim 3. $\square$

The above proof hopefully enlightens the definition of $\det(V)$ when $m = |V| < n$, and also the basic structure of $V$-coefficients.

Let us say that $\alpha \in \mathbb{Z}^V$ is a $\mathrm{par}(V)$-*coefficient* vector, or if there is no ambiguity, a *parallelepiped coefficient vector*, if $\alpha = \det(V)\,\mathrm{coeff}(x, V)$ for some $x \in \mathrm{par}(V)$. With the notations of the above proof $\lambda = (\lambda_1, \ldots, \lambda_m) \in \mathbb{R}^n$ is a $\mathrm{par}(V)$-coefficient vector, if and only if $\lambda \in \Lambda(V)$, and $0 \geq \lambda_i < \det(V)$ for $i = 1, \ldots, m$. Attention ! For $x \in \mathrm{par}(V)$ the $\mathrm{par}(V)$-coefficient is $\det(V)$ times the $V$-coefficient, and it is integer.

The $\mathrm{par}(V)$-coefficient vectors form a commutative group with respect to $\mathrm{mod}\,\det(V)$ addition. Denote this group by $G(V)$.

The last part of the proof evades Cramer's rule, and is in fact equivalent to it (but it also include the 'non-full-dimensional' case of it). We can see from it that $G(V)$ is isomorphic to the group formed by the $\mathrm{par}(V)$-coefficient vectors (which are integer vectors) with respect to $\mathrm{mod}\,\det(V)$ addition. We will also exploit the fact that parallelepipeds are *symmetric*: if $x \in \mathrm{par}(v_1, \ldots, v_m)$, then $v_1 + \ldots + v_m - x \in \mathrm{par}(v_1, \ldots, v_m)$. In other words, if $(\lambda_1, \ldots, \lambda_m) \in G(V)$, then $(D - \lambda_1, \ldots, D - \lambda_m) \in G(V)$. Parallelepiped coefficients, and these simple properties have been extensively exploited in [100], and we will apply here the same approach. This is similar to the method of 'barycentric coordinates' used by Reznick in [93] for somewhat different purposes.

We will use the notation $G(V)$ for the set of $\mathrm{par}(V)$-coefficient vectors (even when we do not use the group operation).

We now generalise Problem 1:

**Parallelepiped Programming** (PP)
INPUT: A set $V \subseteq \mathbb{Z}^n$ of linearly independent vectors, $m := |V|$, and

$S_j \subseteq \{1, \ldots, m\}$, $l_j \in [0,1]$, $u_j \in [0,1]$, $k \in \mathbb{Z}$, $(j = 1, \ldots, k.)$.
QUESTION: Is there a $\lambda = (\lambda_1, \ldots, \lambda_n) \in G(V)$ satisfying the inequalities

$$l_i \leq \sum_{i \in S_j} \lambda_i \leq u_i, \text{ for all } j = 1, \ldots, k.$$

We restricted the problem to $0-1$ inequalities, since this is the case that occurs in all of our applications. For instance in the special $k = 1$ and $S_1 = \{1, \ldots, m\}$, $l_i = D + 1$, $u_i = \infty$, the question becomes exactly: is the simplex conv$\{\{0\} \cup V\}$ empty ?

If we do not restrict ourselves to $0-1$ constraints we get close to Gomory's corner polyhedra see 1.2.3, but the restriction does not help.

**Theorem 1.2** *Parallelepiped programming is NP-complete, and it remains so if there are only upper bound constraints.*

**Proof.** It is clearly in $\mathcal{NP}$. We reduce PARTITION [52] to it. Let $a_1, \ldots, a_{n-1} \in \mathbb{N}$ (an instance of PARTITION) and $A := \sum_{i=1}^{n} a_i/2$. Consider now the $n-1$-dimensional cone (homogenized knapsack polytope) $C$ in $\mathbb{R}^n$ defined by one equality and the nonnegativity constraints:

$$\sum_{i=1}^{n-1} a_i x_i - A x_n = 0, \; x_i \geq 0 \, (i = 1, \ldots, n).$$

Clearly, the (least integer multiples of) extreme rays of this cone are
$v_1 := (b_1, 0, \ldots, 0, c_1)$, $v_2 = (0, b_2, 0, \ldots, 0, c_2)$ $\ldots$, $v_{n-1} := (0, \ldots, 0, b_{n-1}, c_{n-1})$, where
$b_i = \text{lcm}(a_i, A)/a_i$, $c_i = \text{lcm}(a_i, A)/A$. Let $V := \{v_1, \ldots, v_{n-1}\}$.

**Claim**: If $x = (x_1, \ldots, x_n) \in C \cap \mathbb{Z}^n$, $x_i \leq 1$ $(i = 1, \ldots, n-1)$, $0 < \sum_{i=1}^{n-1} x_i < n-1$, then $x_n = 1$.

Indeed, it is sufficient to prove $0 < x_n < 2$. We have from the assumption that not all the $x_i$, $(i = 1, \ldots, n-1)$ are 0, and not all are 1, so

$$0 < \sum_{i=1}^{n-1} a_i x_i < \sum_{i=1}^{n-1} a_i = 2A.$$

Since $x \in C$, we can substitute here $\sum_{i=1}^{n-1} a_i x_i = A x_n$, that is,

$$0 < A x_n < 2A,$$

as claimed.

It follows that PARTITION with input $a_1, \ldots, a_n$ has a solution if and only if there exists $x \in \mathrm{par}(V)$ such that $x_i \leq 1$ $(i = 1, \ldots, n-1)$, $0 < \sum_{i=1}^{n-1} x_i < n-1$. Indeed, by the claim, such a solution is in $\{0,1\}^n$, and because of $x_n = 1$ it determines a solution of PARTITION; conversely, a solution of PARTITION defines such a $0-1$ solution.

Now the existence of $x \in \mathrm{par}(V)$ such that $x_i \leq 1$ $(i = 1, \ldots, n-1)$, $0 < \sum_{i=1}^{n-1} x_i < n-1$ is a parallelepiped programming problem. Let $x \in \mathrm{par}(V)$, and let $\lambda \in \mathbb{Z}^n$ be its $\mathrm{par}(V)$-coefficient. Now note that $x_i \leq 1$ is equivalent to $\lambda_i \leq \det(V)/b_i$; $0 < \sum_{i=1}^{n-1} x_i$ holds for every vector in $\mathrm{par}(V)$; $\sum_{i=1}^{n-1} x_i < n-1$ is equivalent to $\sum_{i=1}^{n} \lambda_i \leq (\sum_{i=1}^{n} \det(V)/b_i) - 1$. The decision problem about the existence of a parallelepiped coefficient vector satisfying these inequalities, is a parallelepiped programming problem, and it has only upper bound constraints. $\square$

Parallelepiped programming matches the group structure with linear inequalities. In order to treat interesting particular cases of this problem and its applications, one should have in mind a synthesis and interrelation of these two aspects.

A recent beautiful work of Alan Hoffman [70] about linear inequalities over Abelian groups mixes the group structure with linear inequalities. It is not straightforward to see a direct connection of this work to Parallelepiped Programming, since the group $G(V)$ with the ordering of numbers is not an ordered group.

## 1.2 The Group and the Inequalities

First note that the general definition of determinants allows a simple formulation of the (well-known) theorem characterizing when an affine subspace contains integer vectors:

**Fact 1.1** *Given an $m \times n$ integer matrix $A$ with linearly independent rows, and an integer vector $b \in \mathbb{Z}^m$, the set $\{x \in \mathbb{Z}^n : Ax = b\}$ (the set of integers in an affine subspace) is nonempty, if and only if* $\det(A) = \det(A, b)$.

The proof is easy: The vector $b$ is on the lattice generated by the columns of $A$ if and only if adding $b$ to the matrix as a new column, it can be zeroed by elementary column operations. (Elementary operations do not change the determinant, and adding a 0 column does not change it either. On the other hand, for matrices in Hermite normal form with nonzero diagonal, the statement is obvious.

The problem PP combines this linear structure with some linear inequalities. In the following we wish to study first the groups alone, and then together with the inequalities.

### 1.2.1 The Group Structure

The group $G(V)$ is a finite Abelian group: it has a simple and well-known structure that we would like to present from our biased viewpoint. We will go through some elementary facts well-known from algebra in terms of the parallelepiped coefficient vectors of $G(V)$. This is necessary for some subsequent arguments, and for a good understanding of the problems we are stating. We would also like to give a dictionary between the terminology of polyhedral combinatorics and algebra.

By Theorem 1.1, for the integers $0 \leq \lambda_i \leq \det(V)$ we have $(\lambda_1, \ldots, \lambda_m) \in G(V)$ if and only if $a_1\lambda_1 + \ldots, a_m\lambda_m \equiv 0 \mod \det(V)$ for every column $a = (a_1, \ldots, a_m)$ of $M_V$.

The group $G(V)$ is actually nothing else but a general finite Abelian group. Indeed, let $G$ be such a group on $n$ vertices and $\{g_1, \ldots, g_m\}$ a set of generators. Let us use the additive notation for the group operation. The vectors

$$\{(a_1, \ldots, a_m) \in \mathbb{Z}^m : a_1 g_1 + \ldots + a_m g_m = 0\}$$

form a lattice. The finiteness of the group implies that there exists $k_i \in \mathbb{N}$ such that $k_i g_i = 0$ $(i = 1, \ldots, n)$, so there are at least $n$ linear independent points on this lattice; then the lattice has a basis consisting of $n$ vectors, that can be xhosen to be the columns of an $n \times n$ matrix $M_U$. (The rows $U$ of $M_U$ will arise from $M_V$ by a unimodular transformation.)

Choose any basis of this lattice to be the columns of the matrix $M_V$, and let $V$ denote the rows of $M_V$.

$G(V)$ is the group defined from the free group on $n$ elements 'presented' with the relations given by the columns of $M_V$. (Any generating set of the above equations is called a presentation.) Clearly, such a group is unique, whence $G$ is isomorphic to $G(V)$. The matrix $M_V$ and $G = G(V)$, mutually determine one another.

For the rest of this section we fix $V \subseteq \mathbb{Z}^n$, $|V| = m$, $M_V$ is the $m \times n$ matrix whose row-set is $V$ (in arbitrary order), and $G = G(V)$.

The elementary column operations (see the proof of Theorem 1.1) correspond to replacing a presentation by an equivalent one, and this transformation does not change the vectors in $G(V)$ representing $G$ (the par-coefficients of the rows). Similarly, if $g_1$ and $g_2$ are two generators of the group, then any of them can be replaced by $g_1 \pm g_2$, and the order of the generators can also be changed (and this changes the par($V$)-coefficients, that is, the isomorphism from $G$ to $\mathbb{Z}^m$), the group of parallelepiped coefficient vectors of the rows remains unchanged. These correspond to adding or subtracting a row of $M_V$ to another row, or interchanging the order of the rows, and we will refer to them as *elementary row operations*.

It is straightforward to see that one can pursue these operations until arriving at the Smith normal form [115], [99]:

**Theorem 1.3** *(Smith normal form) If $M$ is an $m \times n$ integer matrix with linearly independent columns or rows, then it can be brought by elementary row and column operations into a form where the only nonzero elements are on the diagonal of the leftmost and uppermost $m \times m$ submatrix, and denoting by $d_1, \ldots, d_m$ the diagonal of this matrix, $d_i | d_{i+1}$ $(i = 1, \ldots, m-1)$.*

The Smith normal form of a matrix can be determined in polynomial time [72].

It is easy to see that the effect of a sequence of elementary column operations on a matrix corresponds to multiplying it by a unimodular matrix (a square matrix of determinant 1) from the right; conversely, multiplying by a unimodular matrix from the right can be decomposed to column operations. (This is easy to see by bringing the unimodular matrix to its Hermite normal form.) Similarly, a sequence of elementary row transformations is equivalent to a single multiplication with a unimodular matrix from the left. Recall that elementary column operations do not change the set of par-coefficients, whereas the row operations *do change them*; since row operations correspond to changing the generators of the group, $G(V)$ remains isomorphic to $G(V')$, but the inequalities do change.

Algebrists call these elementary operations *Tietze-transformations*. Theorem 1.3 is called the *fundamental theorem of finite Abelian groups*, or *Kronecker's theorem*. (See [116], [99].)

The following corollary allows decomposing some parallelepiped programming problems into a polynomial number of subproblems:

**Corollary 1.1** *The group $G(V)$ is isomorphic to the group of parallelepiped coefficient vectors of the rows of the Smith normal form of $M_V$ which itself is the direct sum of cyclic groups of size $d_i$ $(i = 1, \ldots)$; in particular, this group is the direct sum of at most $\log_2(\det(V))$ cyclic groups.*

Indeed, from Theorem 1.3 we get immediately $d_{m-i} \leq d_m/2^i$ $(i = 1, \ldots, m-1)$, so at most $d_m \leq \det(M_V) = \det(V) \leq \log_2 \det(V)$ entries are bigger than 1 in the Smith normal form. We have already checked in our previous remarks that $G(V)$ does not change through elementary row and column operations on $M_V$, so it is isomorphic to $G(U)$ where $U$ is the set of rows of the Smith normal form. But then $G(U)$ is the direct sum of the cyclic groups on $d_i$ elements, for all $i = 1, \ldots, n$ such that $d_i > 1$.

### 1.2.2 Cyclic groups and jumps

We conclude that parallelepiped coefficient vectors form a group which behaves computationally well: it is the direct product of a polynomial number of cyclic groups. However, we cannot solve Problem 1 even if $G(V)$ is cyclic. It is so actually if and only if the diagonal of the Smith normal form is $(1, \ldots, 1, d)$. This allows to establish that $G(V)$ is cyclic in various cases. In particular, the following will be important for us:

**Fact 1.2** *If $V \subseteq \mathbb{Z}^n$ are linearly independent vectors and the parallelepiped generated by at least one of $n-1$-element subset of $V$ is equal to $\{0\}$, then $G(V)$ is cyclic.*

Indeed, choose the $n-1$-element set of the condition to be the first $n-1$ rows of $M_V$. If in the Hermite normal form not all the first $n-1$ entries of the diagonal were 1, then it is easy to exhibit (in the same way as in the proof of Theorem 1.1) a vector in the parallelepiped which is a linear combination of these first $n-1$ rows, contradicting the condition. (The statement can also be seen easily directly, without the normal forms.)

The condition of this fact is not necessary: for instance, if all prime factors of $d$ are at the first power, then the condition is not necessarily satisfied, but the group is also cyclic. In the following we mostly focus on the cyclic special case since the structure is simple to think about, according to Corollary 1.1 this is not an essential restriction of generality, moreover, most of the examples will involve directly this case.

Let us now consider the inequalities together with the group structure. Let us suppose that $G(V)$ is cyclic and try to keep track of how the parallelepiped coefficients vary. Let $\alpha \in G(V)$ be a generator of $G(V)$, that is, the $V$-coefficients are $(\alpha_1/\det(V), \ldots, \alpha_n/\det(V))$. Then the $V$-coefficients of the $det(V) - 1$ nonzero vectors in $\mathrm{par}(V)$ are:

$$(\{i\alpha_1/\det(V)\}, \ldots, \{i\alpha_n/\det(V)\}),\ (i = 1, \ldots, det(V) - 1),$$

where $\{x\} := x - \lfloor x \rfloor$ is the *fractional part* of $x$. Therefore an 'atom' of PP is the following problem, and actually we can reduce the general problem to this:

**Residue Inequalities**
INPUT: Rational numbers $a_i \in \mathbb{Q}$ $0 < a_i < 1$, $i = 1, \ldots, n$, sets $S_i \subseteq \{1, \ldots, k\}$ and numbers $u_i \in \mathbb{Q}_+$.
QUESTION: Does there exist a $z \in \mathbb{Z}$ so that

$$\sum_{j \in S_i} \{za_j\} \leq u_i \, (i = 1, \ldots k)?$$

**Theorem 1.4** *Parallelepiped Programming can be reduced in polynomial time to Residue Inequalities, in other words the latter is also $\mathcal{NP}$-complete.*

**Proof.** (Sketch) By Corollary 1.1 we can list a generating set of $G(V)$ whose size is polynomial in the input. For the multiples of each generator, it is straightforward to see that the set of inequalities to check is equivalent to a Residue Inequalities problem. $\Box$

We do not know, however, the complexity of the problem if there is only one inequality. Moreover, in the applications the $u_i$ are integer. (They can always be supposed to have the same denominator as $x$: multiplying by this denominator all the data are integer, and we get linear congruences.)

**Problem 2** *What is the complexity of the following problem: Given rational numbers $0 < a_i < 1$, $i = 1, \ldots, n,$, and $u \in \mathbb{Z}$, does there exist $z \in \mathbb{N}$ so that*

$$\{za_1\} + \ldots + \{za_n\} \leq u?$$

*Note that the complementary problem is to decide whether for the same input it is true for all $z \in I\!N$ that*

$$\{za_1\} + \ldots + \{za_n\} > u.$$

*Is this problem also in $\mathcal{NP}$ ? Is there a particular and 'interesting' subset of inputs for which some of these problems can be solved ?*

The first condition is in $\mathcal{NP}$, but not the second, since we have to check the condition for all $z \in \mathbb{N}$, equivalently for $z = 1, \ldots, D-1$, where $D := \mathrm{lcm}(a_1, \ldots, a_n)$, and the input size contains only the logarithm of $D$. (In the special case we solve below the condition will actually be independent of $D$.)

This, and other problems we will mention, make sense for irrational numbers as well, and can mostly be reduced to rational numbers.

Note that the emptiness of a simplex is exactly the special case $u = 1$ of this problem. Indeed, we can suppose without loss of generality that one of the vertices of the simplex is 0, and use Corollary 1.1 to assume that the parallelepiped group generated by the others is cyclic.

If $n = 3$ this decomposition does not have to be used: if the parallelepiped generated by some face, say the face of $v_1$ and $v_2$ is not equal to $\{0\}$, then any $v \in \mathrm{par}(v_1, v_2) \setminus \{0\}$ or $v_1 + v_2 - v$ which is also in the parallelepiped (this property will be called 'symmetry') are in the simplex, contradicting emptiness. Then by Fact 1.2 the parallelepiped is cyclic. This argument can be straightforwardly generalized to prove that in even dimension if *the sum of the coordinates of every parallelepiped coefficient vector is at least $n/2$*, then the parallelepiped group is cyclic. (Then it can be easily checked using the symmetry again, that the sum of every parallelepiped coefficient vector must be equal to $n/2$.) Therefore the following conjecture *would provide a good characterization of such parallelepipeds.*

**Conjecture 1** *[102] If $n$ is even, and $0 < a_1 < \ldots < a_n < 1$, $i = 1, \ldots, n$, are rational numbers with denominator $D$, then for all $i = 1, \ldots, D-1$ we have*

$$\{ia_1\} + \ldots + \{ia_n\} \geq n/2,$$

*if and only if $a_j + a_{n-j} = 1$ for $j = 1, \ldots, n/2$, and $\gcd(a_iD, D) = 1$ $(i = 1, \ldots, n)$.*

We do not know of direct applications of this conjecture, but it can be a good stepping stone to other problems, for instance to empty simplices. For n=4 it implies a characterization for three dimensional empty simplices. Moreover, the key lemma that supports the case $n = 4$ provides new insight to Hilbert bases and the key-lemma leading to the solution provides a bridge towards another circle of applications (problems involving fractional parts, stated by number theorists) we show in the next section. Let us state this key-lemma, and sketch its applications.

Let $x \in \mathbb{Q}$, $x = d/D$, $\gcd(d, D) = 1$, $d, D \in \mathbb{N}$, $d < D$. Let us say that $i \in \{1, \ldots, D-2\}$ is a *jump* for $x \in \mathbb{R}$, $0 < x < 1$, if $\{ix\} + x \geq 1$, that is, if $\{(i+1)x\} \neq \{ix\} + x$. (This means that between $ix$ and $(i+1)x$ we 'jump' over an integer.) Let us state two straightforward, but crucial facts from [102] :

**Fact 1.3** *The set of jumps for $x \in \mathbb{R}$, $0 < x < 1$ is $\{\lfloor z/x \rfloor : z = 1, \ldots, D-2\}$.*

**Fact 1.4** *(Symmetry) If $x \in \mathbb{R}$, $0 < x < 1$, and $ix$ is not integer, then $i$ is a jump for $x$ if and only if it is not a jump for $1 - x$. In particular, if $x$ is not rational, the jumps of $x$ and $1 - x$ bipartition $\mathbb{N}$, and if $x = d/D$ where $d, D \in \mathbb{N}$, $\gcd(d, D) = 1$, then the jumps for $x$ and for $1 - x$ bipartition $\{1, \ldots, D-2\}$.*

If $0 < x < y < 1$ and $y$ is a multiple of $x$, then the jumps of $y$ contain all jumps of $x$. Our key-lemma tells that this can be reversed if $x, y \leq 1/2$ :

**Lemma 1.1** *[102] Let $a_1, a_2 \in \mathbb{Q}$, $0 < a_1, a_2 \leq 1/2$. Then every jump of $a_1$ is a jump of $a_2$ if and only if $a_1 | a_2$.*

Without the condition the statement is not true, see[102].

This simple, elementary fact is not as trivial to prove as it looks. It can be proved as a difficult exercise in more than one way, but a short and elegant proof (as it could be expected) is still to be found. It is a key to empty simplices, a simple proof for unimodular triangulation in 3-space, and using Fact 1.3 a stepping stone to some new applications shown in the next subsubsection.

The following straightforward corollary, however, opens the way to several applications:

**Lemma 1.2** *Let* $V := \{v_1, v_2, v_3\} \subseteq \mathbb{Z}^3$ *be linearly independent. If* $\mathrm{conv}(0, v_1, v_2, v_3)$ *is empty, then there exists* $\lambda = (\lambda_1, \lambda_2, \lambda_3) \in \mathrm{par}(V)$ *and* $i \in \{1, 2, 3\}$ *so that*

$$\lambda_1 + \lambda_2 + \lambda_3 = det(V) + 1, \ \ and \ \lambda_i = 1$$

.

**Proof.** Suppose $\mathrm{conv}(0, v_1, v_2, v_3)$ is empty, and let $\lambda = (\lambda_1, \lambda_2, \lambda_3)$ be any generator of $G(V)$. Let $D := \det(V)$. From the emptiness $\gcd(v_i, D) = 1$ $(i = 1, 2, 3)$, and using also the symmetry of parallelepipeds we get that

$$D + 1 \leq \{i\lambda_1\} + \{i\lambda_2\} + \{i\lambda_3\} \leq 2D - 1, \ (i = 1, \dots D - 1),$$

and it follows that

$$\lambda_1 + \lambda_2 + \lambda_3 = D + 1$$

can be supposed without loss of generality. (See more details if necessary of this first part of the proof in [100] or [102].) Suppose without loss of generality $\lambda_1 \geq \lambda_2 \geq \lambda_3$.

It follows that the jumps of $\lambda_1/D$, of $\lambda_2/D$ and of $\lambda_3/D$ partition $\{1, \dots, D-2\}$. Applying this to $i = 1$ we get that $\lambda_1/D > 1/2 > \lambda_2/D \geq \lambda_3/D$. Therefore, by Fact 1.4 every jump of $\lambda_2/D$ or $\lambda_3/D$ is a jump of $(1-\lambda_1)/D$, and all these are at most $1/2$, so Lemma 1.1 can be applied: both $\lambda_2$ and $\lambda_3$ divide $D - \lambda_1$. If neither of them is equal to $D - \lambda_1$, then both are smaller than $)D - \lambda_1)/2$ and $\lambda_1 + \lambda_2 + \lambda_3 \leq D$ follows, contradicting the second equation above.

It follows that $\lambda_2 = D - \lambda_1$ and applying the equation again, $\lambda_3 = 1$, as claimed. $\square$

For $n = 3$ several solutions of the empty simplex problem have been found, mostly independently of one antother, see [94], [96], [124], [93]. The following was found as an application of Lemma 1.1, and it turned out to imply all these results:

**Theorem 1.5** *Let* $V := \{v_1, v_2, v_3\} \subseteq \mathbb{Z}^3$ *be linearly independent. Then* $\mathrm{conv}(0, v_1, v_2, v_3)$ *is empty, if and only if* $G(V)$ *has a generator* $g$ *so that* $\mathrm{coeff}(g, V)$ *has two coordinates which sum up to* 1. *(Equivalently, all parallelepiped coefficient vectors have two such coordinates.)*

The proof follows straightforwardly from Lemma 1.2. The same Lemma contains the core of the proof of unimodular partitions of Hilbert cones of 3-space.

For an introduction to Hilbert bases we refer to [99]. A *triangulation* of a cone is a covering by cones with linearly independent extreme rays (simplicial cones), which do not have a common inner point.

**Theorem 1.6** *[100] Let $H \subseteq \mathbb{R}^3$ be a Hilbert basis. Then $H$ can be triangulated with cones whose (linearly independent) extreme rays are in $H$ and are Hilbert-bases themselves.*

A weakening – by deleting the requirement that the simplicial Hilbert cones don't have a common interior – of the statement of this theorem was hoped to be true in general, and became known as the 'Unimodular Covering Conjecture'. (The original proof in [100] used the weakening of Lemma 1.2 where the existence of $i \in \{1, 2, 3\}$ with $\lambda_i = 1$ is not proved. The surplus of our statement here requires some additional effort, but simplifies the proof of the unimodular covering conjecture for $n = 3$.)

Bouvier and Gonzalez-Sprinberg provided a counterexample to unimodular partitioning in 4-space [13]. Triangulations have some significance for toric varieties see [117], [48].

Bruns and Gubeladze have shown, a counterexample to the unimodular covering conjecture [14]. The same six dimensional (pointed) Hilbert cone contains an integer vector that is not a nonnegative integer combination of at most six vectors from the Hilbert basis (only of seven vectors), as Henk, Martin and Weismantel noticed in the new variant [15]. The conjecture is open in four- and five-space.

### 1.2.3 Other Applications

Parallelepiped Programming has in fact been already investigated in other terms, with the objective of solving integer programs: the heart of Gomory's approach [62], [126], [99] is a kind of description of the convex hull of integer points of a cone.

The parallelepiped groups defined in the previous section are in fact a special case of Gomory's groups, if one restricts the cones to have linearly independent extreme rays. It could be useful to work out more about the connections of the results themselves.

Let us now relate some other well-known problems to parallelepiped programming.

A connection between Hilbert bases and diophantine approximation with interesting results open problems has been pointed our in [67]. The similarity of the methods applied to Parallelepiped Programming (Hilbert bases, empty simplices, ... ) and those used to solve some other particular diophantine equations see for instance [8], is also apparent. This peculiar problem seems to formulate the essence of some diophantine approximation problems [125], a geometric problem like view obstruction [38] and combinatorial problems like nowhere zero flows in graphs and matroids.

**Problem 3** *(The Lonely Runner Problem [125], [38], [8])*
*Given* $v = (v_1, \ldots, v_n) \in \mathbb{Z}^n$, *does there exist for all* $k \in \{1, \ldots, n\}$ *a* $t = t(k) \in [0, 1]$ *so that the following Lonely Runner Inequality (LRI) holds:*

$$(LRI) \;\; vt \equiv u \bmod 1, \;\; \textit{and } |u_i - u_k| \geq 1/n, \; (i = 1, \ldots, n)?$$

Clearly, adding a constant to $v$ the statement does not change. In particular, one can suppose that there exists $i \in \{1, \ldots, n\}$ such that $v_i = 0$. For a relatively simple proof of the cases $n \leq 5$, see [8]. For $n > 6$ the problem is open.

The Lonely Runner Problem is a common generalization of some problems in combinatorial geometry, and diophantine approximation. We give the following reformulation as a parallelepiped programming problem, but we omit the proof:

**Fact 1.5** *Let* $v \in \mathbb{Z}^{n-1}$, *and define* $d$ *to be the least common multiple of the set of numbers* $\{v_i + v_j : i \neq j = 1, \ldots, n\}$, *and* $V := \{e_1, \ldots, e_{n-1}, r\} \subseteq \mathbb{R}^n$, *where* $r = (v_1, \ldots, v_{n-1}, d) \in \mathbb{R}^{n+1}$. *Then (LRI) holds for the numbers* $v = (0, v_1, \ldots, v_{n-1})$ *and k=1 if and only if there exists a* $\mathrm{par}(V)$*-coefficient that satisfies the inequalities*

$$\frac{1}{n+1} \leq \lambda_i \leq \frac{n-1}{n+1} \, (i = 1, \ldots, n) \, ?$$

Note that it is not true in general that any parallelepiped having an integer point which is not on any of the facets, contains an integer point satisfying (LRI): if $n = 2$, $V := \{(2, 1), (1, 3)\}$, we have $\det(V) = 5$ and in every par($V$)-coefficient vector at least one of the two coordinates is at most $1/5 < 1/3$ or bigger than $4/5 > 2/3$.

At last, I would like to mention some newly discovered connections of jumps to some classical work by Skolem [114], [3]. A scheduling problem studied by Brauner and Crama [11] revealed some further connections of parallelepiped programming to 'number-oriented-combinatorics'.

In the revised version I may say a few more words and state a generalization of Fact 1.3 as a conjecture leading to a general problem containing Skolem, Brauner-Crama, Beatty sequences, empty simplices etc.

Moreover, (through pointers of Gerhart Woeginger concerning the problem of Brauner and Crama) one can arrive at further connections to number theory problems of similar nature, involving fractional parts, see Tijdeman's work [120] about particular cases of Fraenkel's conjecture [49].

# 2 Classes of Integer Polyhedra

There are several exhaustive surveys of the classical results concerning basic general results and characterizations of subclasses of integer polyhedra (see Schrijver's survey paper [98] or book [99], or the more recent survey of [24] on $\pm 1$ perfect, ideal and balanced matrices, and Gérard Cornuéjols's book [32] provide a detailed account of the results - a transitive closure of the pointers from these covers the literature of the subject.) In this survey we restrict ourselves to a brief overview of the *classical hierarchy* of integer polyhedra followed by an account of recent results concerning intersections of known classes of polyhedra *mixing* constraints of different kind. Some details of several subsubsections are left out until the revised version.

## 2.1 Classical hierarchy

The four cornerstones of each subsubsection of this subsection are

> the *definition* of the subclass treated there. This definition usually dates from the early days of integer programming or of graph theory, together with
>
> the proof of the *integrality* of polyhedra in the subclass. For some of these subclasses

*combinatorial characterizations* were proved before the construction of

*recognition* algorithms with polynomial complexity bounds.

These four cornerstones will be indicated in italics in each subclass.

Recognition algorithms exist only for the two simplest classes of totally unimodular and balanced matrices. These are among the most difficult problems, and most complicated algorithms of combinatorial optimization. The complexity of recognizing more general classes is still not known, and even the problem of nonalgorithmic characterizations (that would put some classes of problems in $\mathcal{NP}$) is open. This is the subject of ongoing research.

### 2.1.1 Totally Unimodular Matrices

A matrix is *totally unimodular* if each of its square subdeterminants is 0, 1 or $-1$.

Totally unimodular matrices were *introduced in* [68], which also establishes a *link to integer polyhedra*: a matrix $A$ is totally unimodular, if $\{x : Ax \leq b\}$ is an integer polyhedron for arbitrary integer vector $b$.

Several other useful *combinatorial characterizations* have been shown and are surveyed in [99]. However, these provide only co$\mathcal{NP}$ certificates to total unimodularity, like the definition itself. Tutte [123] has proved that the matroids representable by a totally unimodular matrix over the reals, are representable over an arbitrary field. Such matroids are called *regular*. He characterized such matroids in [121] with three simple *excluded minors*. For a simple proof see Gerards [56].

The *recognition* of totally unimodular matrices is therefore equivalent to testing whether a matroid contains the three minors of Tutte's theorem. Seymour developped a decomposition procedure of regular matroids providing a polynomial algorithm and a simpler $\mathcal{NP}$ certificate for them, reproving Tutte's theorem. A decomposition can be found in polynomial time using Cunningham and Edmonds' reduction [37] to matroid intersection; the 'bricks' of the decomposition are graphic matroids for which a recognition algorithm has already been designed by Tutte [122], and several improved algorithms followed [9], for an account see [86].

Besides its self-interest, Seymour's method also paved the way of many similar decomposition theorems for various problems. Variants of the selfcontained algorithm are presented in [86] and [99], the latter in terms of totally unimodular matrices.

### 2.1.2 Balanced Matrices

A matrix with all entries from $\{0,1\}$ is *balanced*, if it does not contain a square submatrix with an odd number of rows and columns, and all entries equal to 0 except exactly two entries in each row and each column. Since the determinant of such a matrix is 2 we have immediately that nonnegative totally unimodular matrices are balanced.

Balanced matrices were *introduced in* [6], see also [7], where many of their interesting properties are shown.

The main *link to integer polyhedra* is shown in [51]: a matrix $A$ is balanced, if and only if $\{x \in \mathbb{R}^n : A'x \leq 1\}$ is an integer polyhedron for any subset of rows, or equivalently, if and only if $\{x \in \mathbb{R}^n : A'x \geq 1\}$ is an integer polyhedron for an arbirary subset of rows. By linear programming duality we also get other characterizations. These characterizations are clearly less restrictive than those for totally unimodular matrices.

Several other useful *combinatorial characterizations* have been shown and are surveyed in [99]. However, these provide only co$\mathcal{NP}$ certificates of balancedness, like the definition itself.

No $\mathcal{NP}$ characterization was proved before the *recognition* of totally unimodular matrices could be solved with a polynomial time algorithm by Conforti, Cornuéjols and Rao [25]. While the most interesting questions about totally unimodular matrices seem to be essentially closed the same cannot be said about balanced matrices: there is no natural $\mathcal{NP}$-characterization simpler than the algorithm itself.

### 2.1.3 Packing and Perfect Matrices

A packing type of polyhedron is a polyhedron of the form $P := \{x \in \mathbb{R}^n : Ax \leq 1,\ x \geq 0\}$.

In this section we summarize the results about packing type polyhedra that are integer.

Let us call an $n \times n$ matrix $A$ with all entries from $\{0,1\}$ *perfect*, if for every $c \in \{0,1\}^n$ the polytope $P := \{x \in \mathbb{R}^n : Ax \leq 1,\ x \geq 0\}$ contains an integer vector that maximizes $c^T x$. That is, there exists $x_0 \in P \cap \{0,1\}$ such that $cx_0 \geq cx$ for all $x \in P$. Let us then also say that the polytope $P$ is perfect.

In other words, the definition requires from the matrix $A$ that the polytope $P$ is integer 'in the direction of $0-1$ objective functions'. Clearly, balanced matrices are perfect.

The *integrality* of perfect polyhedra is equivalent to the so called 'Lovász replication lemma' modulo basic polyhedral combinatorics see for instance in [99]. Fulkerson [50] showed that for packing type of polyhedra integrality is equivalent to total dual integrality.

**Theorem 2.1** *Perfect polyhedra are integer, moreover the minimal system of inequalities that describes them is totally dual integral.*

Chvátal [23] noticed:

**Corollary 2.1** *A matrix $M$ is perfect if and only if there exists a perfect graph $G$ so that the rows of $M$ are exactly the characteristic vectors of inclusionwise maximal cliques of $G$.*

See some detailed algorithmic comments about the connections of these results to three different certificates of imperfectness in [92].

the corollary that perfect matrices are exactly the clique matrices of *perfect* (in which the size of a maximum clique is equal to the chromatic number for every induced subgraph). In the revised version a short survey of some points to know will follow in the style of the preceding sections. This will include a short co$\mathcal{NP}$ certificate (in accordance with analogous short certificates for ideal matrices), having also in mind to prepare 2.2.1. The short proofs in [53], [55] help preparing the generalization.

The subsection will finish with the statement of the SPGC and some words about perfectness test. Theta will be mentioned and Shepherd's lemma [112] for testing partitionability will be stated. Questions for analogous results in the ideal case will be stated. (All this will not be more than one page altogether.)

### 2.1.4 Covering and Ideal Matrices

In the revised version the definitions and summary of results (mainly Lehman) will occur by analogy with the results on perfect graphs, including partitionabilityn, and having in mind 2.2.1 (about half a page).

We will also have to define binary clutters, down-matroids, up-matroids, we will need these for 3.1 as well.

We include here the full details of the end of this subsection:

The complete list of minimal nonideal matrices is not known, even not as a conjecture like for perfect graphs; furthermore, the existing examples show that a complete list would be hard to establish [85], [34] [82]. However, a wide class of important applications is provided by binary clutters, including theorems on packing paths on surfaces, themselves equivalent to multiflows. (For a survey on these see [57].)

**Theorem 2.2** *[65] Seymour's conjecture is true for binary clutters whose up-matroid is graphic or the blockers of such clutters.*

A lack in Guenin's theorem: there is one major class of ideal clutters that it does not contain, which also plays a role as a building block for further classes in [107]. Cornuéjols and Guenin [33] made one more step towards Seymour's conjecture: they generalize Guenin's theorem so that it contains Edmonds and Johnson's theorem [44] on the idealness of $T$-join clutters.

Couldn't the theta function be generalized to covering polyhedra, maybe with the help of Lovász-Schrijver cuts [81] ? Can partitionability be tested for ideal clutters ?

### 2.1.5 Total Dual Integral Systems and Hilbert Bases

Totally dual integral systems were defined by Edmonds and Giles [43] as follows:

A system of inequalities $Ax \leq b$ ($A$ is an $m \times n$ matrix, $b \in \mathbb{R}^m$) is called *totally dual integral* (TDI), if any inequality $c^T x \leq d$, and $c \in \mathbb{Z}^n$ which is their consequence (in other words if $Ax \leq b$ is satisfied for $x \in \mathbb{R}^n$, then $c^T x \leq d$ is also satisfied), arises by weakening a non-negative integer combination of inequalities in the system (that is, there exists $y \in \mathbb{Z}+^n$, $yA = c$, $y^T b \leq d$).

**Theorem 2.3** *[43] A TDI system with integer right hand sides defines an integer polyhedron.*

Giles and Pulleyblank [60] observed that this can also be reversed. We combine their result with the unicity result of Schrijver [99]:

**Theorem 2.4** *Every full dimensional polyhedron $P$ has a unique TDI defining system, and $P$ is integer if and only if the right hand sides in this system are integer.*

Hilbert bases and TDI systems have many occurances already from the early days of the theory of integer programming see for instance [41], [42], [35], [36], for surveys see [98], [99], [100]. Some well-known examples (among these) are matching polytopes and their generalizations, (poly)matroid polytopes and their intersections, submodular flows, etc.

The most fundamental problem about TDI systems raised by Edmonds and Giles [43] remains open:

**Problem 4** *Given an integer matrix $A$ and an integer vector $b$, is the system $Ax \leq b$ TDI ?*

Through the above correspondance between TDI systems and Hilbert bases, a special case that contains the essence of the problem is the following:

**Problem 5** *Given a set $H \subseteq \mathbb{Z}^n$ of integer vectors, do they form a Hilbert basis ?*

If the dimension (somewhat more generally the rank) is bounded by a constant, then these problems have been solved by polynomial algorithms [30].

Testing whether a given element is in the Hilbert basis is $\mathcal{NP}$-complete, see [100].

In the revised version we will profit some more on the reformulation of Hb properties to TDI systems, including a 'greedy' way of deducing dual solutions. This insight will be used later on for subclasses of polyhedra (for instance for the TDI-ness of packing-polyhedra or $\pm 1$ matrices. Some simple properties of Hilbert bases will allow us to include more proofs, for instance to Edmonds-Giles. All this will be less than a page.

## 2.2 Mixing packing and covering

The subject of this subsection can be viewed to be the intersection of different kinds of integer polyhedra. The intersection of integer polyhedra is of course not necessarily integer. (For a simplest example, partition the edges of a circuit on $2n + 1$ vertices into two nonempty sets $E_1$ and $E_2$, and let $P_i :=$

$\{x_u + x_v \in \mathbb{R}^{2n+1} : uv \in E_i\}$, $i = 1, 2$. The all $1/2$ vector is a vertex of the intersection.)

The matroid intersection theorem is an example where the intersection of pairs of polyhedra is integer (moreover the facet inducing inequalities form a TDI system) by the theorem of Edmonds [42]. We do not study this theorem in details, it is treated in any book on combinatorial optimization or matroid theory in the past three decades [28], and at a general level we cannot say more about them than about TDI systems. But why aren't there other intersection theorems:

**Problem 6** *Given two classes of integer polyhedra in $\mathbb{R}^n$ which belong to one of the known subclasses. Is it possible to identify conditions, or types of pairs of polyhedra for which the intersection is also be integral ?*

In the first half of this section (Subsection 2.1) almost all the considered polyhedra had $0 - 1$ constraints, only the first and the last subclass, totally unimodular matrices and totally dual integral systems, were allowed to have negative coefficients. Typical applications of these are to packing and covering problems (polyhedra). Writing all inequalities in the smaller or equal form, covering polyhedra have zero-minus-one constraint matrices. The best would therefore be to state results in general for $\pm 1$ constraint matrices.

This is difficult though. The results that have been reached so far do require strong conditions on the system.

**Problem 7** *Is it possible to identify conditions depending both on $A$ and $b$, where $A$ is a $0 - \pm 1$ matrix, under which the polytope $P := \{x : Ax \leq b,\ x \geq 0\}$ is integral, and so that the conditions are satisfied both by packing and covering polyhedra ? Are there co$\mathcal{NP}$ characterizations of the nonintegrality of $P$ in terms of $A$ and $b$, at least for subclasses containing both the characterization of minimal imperfect and minimal nonideal matrices ?*

This last problem has found some first solutions explained in the following subsubsection. The other subsubsection surveys the so called $\pm 1$ constraint matrices which arise by flipping packing constraints or covering constraints but without mixing the two.

### 2.2.1 Generalizing minimal imperfect and minimal nonideal clutters

This subsubsection will state the simplest common generalization of minimal imperfect and minimal nonideal clutters [55]. We explain what are the main obstacles for more powerful results, how these are treated in some other papers, and what is needed for a solution. Some more explanations will follow in one direction, involving elementary number theory again: a lemma based on divisibility, implying tight properties for a somewhat more general minimal noninteger structure, and also the full dimensionality of minimal noninteger polyhedra will be proved in a few lines. These give an idea of the proof and lead to interesting questions and also to answering a question of Shepherd's on clutters that are both minimal imperfect and minimal nonideal. (About one page.)

### 2.2.2 Flipped packing and covering

This subsubsection will provide an account of the so called $\pm 1$ balanced, perfect and ideal matrices. The constraints of these are flipped packing or covering constraints, and don't mix these two basic cases. A simple treatment reducing the perfect case to perfect graphs will be sketched, and some direction concerning the references will be given [], including also balanced and ideal $\pm 1$ matrices. (About one page.)

### 2.2.3 Kernels

Besides the self-interest of the integrality of the following polyhedra, they will provide us a main sample example in Section 3. After some definitions and an introduction (borrowed from [92]), we develop some of their aspects related to integer polyhedra.

The notion of kernels originate in game theory. Some related problems belong to the kernel of the theory of integer polyhedra, and also of perfect graphs.

We allow the presence of cycles of length two in a directed graph. If an arc is not contained in such a cycle we will call it *strictly oriented*. A path, an arborescence (a rooted tree where every vertex can be reached from the root) or a cycle will be called strictly oriented, if all of its arcs are strictly oriented.

Given a directed graph $G$, a *kernel* is a stable set $S$, such that $S \cup N^-(S) = V(G)$. $G$ is called kernel-perfect, if all of its (induced) subgraphs has a kernel.

For a directed graph $G$ to be kernel-perfect, it is obviously necessary that $G$ is *clique-acyclic*, that is, cliques in $G$ do not contain a strictly oriented cycle.

A *fractional kernel* is a vector $x \in [0,1]^n$ such that $x(N^+[v]) \geq 1$ for all $v \in V$.

*The subject of this subsection is the intersection of the set-covering type polyhedron consisting of the fractional kernels, and of the set-packing polyhedron which is the convex hull of stable-sets of perfect graphs.*

This is certainly the intersection of a packing and a covering polyhedron, whence it has its place in the present subsubsection.

Aharoni and Holzman [2] defined an *intermediary notion between kernels and fractional kernels*: $x \in [0,1]^n$ is a *strong fractional kernel* if it is a fractional kernel, and $x(K) = 1$ for some clique $K \subseteq N^+[v]$. Clearly, any kernel is a strong fractional kernel; a strong fractional kernel is a kernel if and only if it is integer.

**Theorem 2.5** *[2] If $G$ is a directed graph, it has a strong fractional kernel.*

The proof of this theorem uses a theorem of Scarf of linear programming character, developed to treat problems arising in game theory; the use of such tools is due to Boros and Gurvich to prove the following theorem. Aharoni and Holzman's prove through (0.3) the following:

**Theorem 2.6** *[12] If $G$ is perfect, then it has a kernel.*

Strong fractional kernels were introduced as an auxiliary notion suited for the proof of Theorem 2.6. It seems to be also a *remarkable compromise between kernels and their fractional relaxation.*

The following three main questions arise naturally:

**Problem 8** *Does Scarf's algorithm have a version that terminates in polynomial time ?*

**Problem 9** *Can a strong fractional kernel in a directed graph $G$ be found in polynomial time ?*

**Problem 10** *Let $G$ be a clique-acyclic orientation of a perfect graph. Can a kernel in $G$ be found in polynomial time ?*

These questions match a very interesting phenomenon in complexity theory, and therefore we will come back to it: in Section 3.2.3 we will see the alternative to a polynomial algorithm. (This problem cannot be $\mathcal{NP}$ complete, since the answer to the corresponding decision problem is always Yes; yet it can be 'complete' in another class of problems, and the meaning of completeness is again that a polynomial algorithm is not likely to exist.)

Any negative result on this problem (see more explanations in 3.2.3) *would suggest that perfectness cannot be tested in polynomial time* : indeed, if such a test works with a decomposition procedure into simple building blocks in polynomial running time, then kernels in $G$ would likely be reconstructable from those in the building blocks; moreover, in particular classes of graphs kernels can often be found in polynomial time, so one can expect this to be the case in the building blocks. Is this a reason to think that a kernel in a perfect graph can be found in polynomial time ? To enlighten this question is one of the guiding goals (but not the only objective) in the following section.

# 3 Complexity

We do not wish to treat here the classical results on the well-known complexity of testing for special cases of integer programming: the most basic results treated already in Garey and Johnson's book [52], and in the ongoing guide of the same authors in the Journal of Algorithms; the more recent books on integer programming also mention the most interesting results; Schrijver [99] contains a full chapter on the complexity of integer linear programming (Chapter 18) and some related problems. The complexity of most of the particular problems mentioned earlier in this paper will also not be considered again (most of these are in Section 2, with references to the classical results in Subsection 2.1).

The recognition problems for classes of integer polyhedra have actually weaker alternatives that are more attractive and more useful in some sense. This alternative is finding either a combinatorial object crucial for all instances of the class or a certificate that a given instance is not in the class. For example an algorithm that either colors a graph or provides a certificate that the graph is not perfect occurs to be closer to what we need, than testing perfectness. Indeed, according to [63] one can either color a graph with the same number of colors as its clique number $k$ or certify that it is not perfect. In the first case we can be satisfied with the coloring and we don't care whether the graph is perfect : in fact our main interest is to $k$-color, and perfectness is defined only as a substitute of this $\mathcal{NP}$-hard problem; in the second case we can show a certificate of inperfectness. The recognition of similar examples lead to the complexity investigation below, see Subsection 3.2, and further remarks on this particular example in Subsubsection 3.2.1.

## 3.1 Gomory-Chvátal cuts

In this subsection we show some complexity questions related to Gomory-Chvátal cuts and the Chvátal closure of polyhedra. These lead already to some complexity questions that get us out from the $\mathcal{P}$-$\mathcal{NP}$ axis. Furthermore, the complexity of the cuts themselves, and the length of the cutting plane procedures leading to the convex hull of integer points already express the complexity of the problem in some sense, [27], [46].

If $P$ is a polyhedron and the inequality $a^T x \leq a_0$ ($a \in \mathbb{Z}^n$, $a_0 \in \mathbb{R}$) is satisfied by all $x \in P$, then the inequality $a^T x \leq \lfloor a_0 \rfloor$ is called a Gomory-Chvátal cut. The intersection of the Gomory-Chvátal cuts is the (first) Chvátal-closure of $P$ (cf. [28], [99] or [22], [61]). The $k$-th Chvátal closure $P^{(k)}$ of $P$ is the Chvátal closure of $P^{(k-1)}$ ($k = 1, \ldots,$), where $P_0 := P$. Chvátal [22] proved for polytopes and Schrijver [99] for arbitrary polyhedra that $P^{(k)} = \mathrm{conv}(P \cap \mathbb{Z}^n)$ for some $k \in \mathbb{Z}$. One of the goals of Polyhedral Combinatorics is to find (or to optimize, or to 'separate') on $\mathrm{conv}(P \cap \mathbb{Z}^n)$ for given $P$. Indeed, if this goal is fulfilled for a class of polyhedra, then integer programming on the class is reduced to linear programming.

There is an 'explicit construction' of $P^{(1)}$ from $P$ (if it is full dimensional) by Schrijver [99]: round down the right hand sides in a TDI description $\{x : Ax \leq b\}$ of $P$, where $A$ is an integer matrix. (The size of a minimal TDI description is exponential in general.)

Note that the integrality of polyhedra can be expressed in terms of Gomory-Chvátal cuts: *a polyhedron $P$ is integer, that is, $P = \mathrm{conv}(P \cap \mathbb{Z}^n)$ if and only if $P^{(1)} = P$.* (This is an obvious fact – the proof does not have to use any of the mentioned deeper results.) In this case one can test membership in, or optimize on $\mathrm{conv}(P \cap \mathbb{Z}^n)$ with linear programming. However, the problem that stands one step further seems to be already open:

**Problem 11** *Given an integer matrix $A$ and an integer vector $b$ of appropriate dimensions so that for $P := \{x \in \mathbb{R}^n : Ax \leq b\}$, $P^{(1)} = \mathrm{conv}(P \cap \mathbb{Z}^n)$ holds, can one optimize on $P \cap \mathbb{Z}^n$ ?*

Note that Lovász and Schrijver [81] provide another cutting plane algorithm that makes possible separation and optimization on the 'first Lovász-Schrijver closure' of 'solvable' polyhedra in polynomial time. This method uses semidefinit programming; the domain of problems for which it works is different from that of Gomory-Chvátal cuts'. For instance the 'Lovász-Schrijver rank' of matching polyhedra is high (Cook, Dash ?), whereas the clique inequalities satisfied by stable sets can be derived from the edge inequalities in one step. Since Gomory-Chvátal cuts concern arbitrary integer programs, and therefore they are closely related to divisibility and $\mathrm{mod}\, D$ computations whereas Lovász-Schrijver cuts concern $0-1$ programs, and are not closely related with the methods we are discussing, we will not study them in this paper.

A special case of this problem has been solved in the already mentioned work [59]. The following more difficult variant was also a well-known open problem for a long time (see [99] ) until Eisenbrand [45] settled it:

**Theorem 3.1** *Given an $m \times n$ integer matrix $A$, integer vectors $b \in \mathbb{Z}^m$, $c \in \mathbb{Z}^n$ and $k \in \mathbb{Z}$ the problem of deciding whether*

$$\max\{c^T x : x \in P^{(1)}\} \geq k,$$

*where $P := \{x \in \mathbb{R}^n : Ax \leq b\}$, is $\mathcal{NP}$-complete.*

The proof is based on Caprara and Fischetti's reduction [19] of finding the minimum weight of a set in a binary clutter to separating a given vector from a well-defined superset of the Chvátal closure. Eisenbrand observes that this superset is in fact equal to the Chvàtal closure, and concludes the $\mathcal{NP}$-completeness. On the positive side, Hartmann, Queyranne and Wang [66] show examples where some simple and quite general sufficient conditions for proving that the Chvátal rank of some given inequalities is at least two, do work. In this context it is natural to ask the following more particular question, mentioned to the author by Fritz Eisenbrand, which is clearly in $\mathcal{NP}$:

**Problem 12** *What is the complexity of the following problem: Given an $m \times n$ integer matrix $A$, integer vectors $b \in \mathbb{Z}^m$, $c \in \mathbb{Z}^n$, decide whether $Q^{(1)} = \emptyset$, where $Q := \{x \in \mathbb{R}^n : Ax \leq b\}$.*

Theorem 3.1 leaves the possibility for the 'reason of the $\mathcal{NP}$-completeness' to be that separating (in the sense of Grötschel, Lovász and Schrijver [64]) *exactly* from the Chvátal closure can be difficult:

The difficulty could be preventing 'to cut more than the Chvátal closure'. But why to prevent an event that makes us happier ? It is completely satisfying to separate on a polyhedron *contained in* the first Gomory-Chvátal closure and *still containing* $P \cap \mathbb{Z}^n$. A simpler special case of this problem is stated in Problem 11.

Note that the more difficult questions of deciding whether the Chvátal rank of given polyhedra is zero, or whether it is at most one, seem to be too difficult.

**Problem 13** *Is the following problem in $\mathcal{NP}$ ? Is it in co$\mathcal{NP}$ ? Given an $m \times n$ integer matrix $A$, an integer vector $b \in \mathbb{Z}^m$, $P := \{x \in \mathbb{R}^n : Ax \leq b\}$, decide whether the Chvátal rank of $P$ is at most one.*

In Section 3.2.5 we show how a negative answer to both of these questions could be possible.

Note however, that Edmonds and Johnson [44] and Gerards and Schrijver [59] exhibit two important classes of polyhedra where the Chvátal closure is equal to the convex hull of integer points. The first includes the convex hull of the solution of (generalized) matching problems, the second the convex hull of stable-sets of special ($t$-perfect) graphs.

Note also that Lovász and Schrijver's [81] cutting plane algorithm does have the property that one can separate on the 'first closure' of 'solvable' polyhedra.

## 3.2 Searching what Surely Exists

In this subsection we would like to explore how difficult it can be to find an integer vertex of a polyhedron even if we surely know that it exists. (You can experience the difficulty of finding a surely existing object, when you lose your surely existing glasses or keys in your own house !) One meets this problem already when learning the four-color problem, or kernels in perfect graphs: it is not easy to debug the contradiction between the dumbness of the decision problems (modulo the four color theorem and Theorem 2.6 respectively) and the difficulty of finding the certificates.

As we have already mentioned in the introduction of this section, whenever the recognition of a property $P_1$ is needed because it is the condition of a problem $P_2$, in fact much less is sufficient than the recognition of $P_1$: what we need then, is to decide whether '$P_2$ or not $P_1$' holds. At one hand this can be much easier than recognition.

On the other hand, even if there is a theorem that states '$P_1$ implies $P_2$', that is, if '$P_2$ or not $P_1$' always holds, the problem 'Find either $P_2$ or an obstacle for $P_1$' can be difficult. It cannot be $\mathcal{NP}$-hard, since the corresponding decision problem can be solved with a 'dumb yes'. The results we present here show that yet there are possibilities for showing that such a problem is difficult.

We use the notation and terminology of [52] for the basics of complexity theory. Two documents had a revealing nature for the author: the summary on 'total problems' in Cristina Bazgan's thesis [5] written under the supervision of Miklós Sántha and Papadimitriou's book [88]. The following sections present this theory from the viewpoint of a combinatorial optimizer, with the not hidden goal of stimulating its use for our problems. We will show the clarifying effect of this on one major problem concerning the integrality of polyhedra.

### 3.2.1 Examples

The list of natural sources of such problems is endless: first of all, *finding the good certificates* for well-characterized problems provides one of the sources (and actually all other sources are in a precise sense equivalent to this, see Fact 3.1 below). For some of these problems there is still no polynomial algorithm for *finding* the good certificates.

FOUR-COLORING OF PLANAR GRAPHS (FCPG) is a well-known example: according to Appel and Haken's celebrated results [1] the answer to the corresponding decision problem is *identically yes*, that is, can be solved in 0 time. But *how to find* the coloring ? According to Appel and Haken [1], or Robertson, Seymour, Sanders, Thomas [95] it can also be found in polynomial time. Yet this is already an example of a problem where a dumb 'always yes' answer solves the decision problem and *finding the good certificate* is less evident.

Another example from graph theory: COLORING WITH CLIQUE CERTIFICATE OR SHOWING IMPERFECTNESS (CCCI). For given graph as input either find a coloration (a partition into sets of vertices into classes such that none of the edges has both of its endpoints in the same class) and a clique of the same size, or a certificate that the graph is not perfect.

A graph is called *perfect* if its maximum clique-size is equal to its chromatic number. According to a sharpening of the perfect graph theorem see Lovász [78], perfectness is in co$\mathcal{NP}$ [79], for a short certificate see [92]. Therefore it is straightforward to see that one of the certificates (either the clique and the coloration or the certificate for imperfectness) *always exists.* As an implicit consequence of Grötschel, Lovász and Schrijver's results [64] (see also [63]), at least one of these certificates is present in every graph ! In other words, the following question is solved if 'combinatorial' is deleted.

**Problem 14** *Is there a 'combinatorial algorithm' that solves CCCI in polynomial time ?*

This is a variant of the coloration problem for perfect graphs. It does not imply an algorithm for testing the perfectness of graphs. However, if it does not end up with a certificate of imperfectness it does solve a difficult integer program:

Let $G$ be a graph, and

$$Q(G) := \{x \in \mathbb{R}^n : Ax \leq 1\},$$

where the rows of $A$ are the characteristic vectors of the cliques of $A$.

According to Corollary 2.1, $Q(G)$ *is integer if and only if* $G$ *is perfect.*

If the matrix $A$ is given explicitly (which is not the case for the cliques of graphs) , there is no need of any theorem, the results about the existence of certificates are obvious in this case. Let us generalize:

Suppose we are given an $m \times n$ integer matrix $A$ and integer vectors $b \in \mathbb{Z}^m$, $c \in \mathbb{Z}^n$, $Q := \{x \in \mathbb{R}^n : Ax \leq b\}$. Since the ellipsoid method can find a vertex $z$ with $c^T z = \max_{x \in Q} c^T x$ in polynomial time, with some care it can be turned to an algorithm that either finds an optimal point of $Q$ which is integer, or a certificate of nonintegrality. (If $Q$ has vertices, then a fractional vertex $x_0$ and $n$ linearly independent valid inequalities for $Q$, each satisfied by $x_0$ with equality, suffice. For more details concerning certificates of nonintegrality, see [92]. ) Linear programming for integer (or $0-1$ ?) polyhedra could be easier:

**Problem 15** *Given $A, b, c$ find the integer optimum or a noninteger vertex of $Q$ with a 'combinatorial algorithm', in polynomial time.*

Another example, one of the most applied problems of complexity theory is the prime factorization of numbers (PFN). According to introductory courses of number theory a factorization to prime numbers *surely exists* (and is unique), it can also be certified to be one (by Pratt's famous elementary lemma see [88] a number can be certified to be a prime), but it is strongly believed that there is no polynomial algorithm to *find one* ! (See any book on complexity theory, for instance [88].)

A most illuminating example can be found in [71]: LOKL (Local Optimum for Kerninghan-Lin), which could also be called the 'local max-cut problem'. Find a cut in an edge-weighted graph that has the property, that it cannot be increased by moving one vertex to the 'other side'. Such a cut *surely exists.* Can one be *found* in polynomial time ? (It is known that the procedure of moving one by one vertices from one side to the other does not always terminate after a polynomial number of steps, which does not mean that a locally maximum cut cannot possibly be found in shorter time.)

More generally, in a sequence of papers Papadimitriou and his coauthors define several classes of problems where a (directed or undirected) Huge Graph (it can have exponentially many edges and vertices) is associated with each instance. (Think about the vertices of this graph as corresponding to solutions and 'quasi-solutions' to the problem; for LOKL the quasi-solutions are bipartitions of the input graph, and the solutions those bipartitions where the weight of the determined cut cannot be increased by putting a vertex to the other side; we put an edge between two vertices if one arises from the other by putting a vertex to the other side; we orient the edge towards the bigger; we delete edges between vetices of equal weight.)

The following two properties have to be assumed (we define them informally, but in a way that can be easily made precise):

> the problem can be defined as finding 'a vertex with a given particular property in the graph'. (The property is usually simple, for instance

'having degree equal to 1'; the difficulty comes from the big size of the graph.) For LOKL this property is to be a sink that is, to have no out-neighbor.

the graph can be 'locally explored in polynomial time' (for example with an algorithm that computes for any given vertex as input the list of all of its neighbors (or out- and in-neighbors in directed graphs), and this list is of polynomial size). For LOKL this is obviously satisfied.

In general, the class $\mathcal{PLS}$ (Polynomial Local Search) is defined as the class of problems where the QUESTION is to find a vertex of the Huge (but finite) acyclic Graph which is the best (according to an objective function computable in polynomial time), and an initial solution can be found in polynomial time, moreover, for any vertex of the Huge Graph it can be decided whether the corresponding solution of the problem is at least as good as the objective values of its neighbors in the Huge Graph, and if not, a better neighbor can be found in polynomial time.

Cameron and Edmonds [17], [18] and Poljak [90] provide a rich collection of other problems with a dumb yes answer. However, they do not explain the Optimizers' chances of proving her or his *unability* of providing a polynomial algorithm for such a problem. The strength of a series of papers by Papadimitriou and his coauthors is that they recognize typical proof styles, that allow them to define classes of optimization problems that can contain complete problems, and can be used similarly to the theory of $\mathcal{NP}$-completeness for telling easy problems from the difficult ones. The message is that a problem with an *identically yes answer can be difficult as hell*, and can be proved to be so similarly to $\mathcal{NP}$-completeness proofs. Let us sketch the main points of this theory and apply what we will have understood to a major problem of the present paper.

### 3.2.2 Search Problems

We would like to clarify here the differences between search problems and the decision problems concerning the existence of the searched object.

Let $\Sigma = \{0, 1\}$ and let $\Sigma^*$ be the set of all 0-1 series. A *relation* is $R \subseteq \Sigma^* \times \Sigma^*$; with an abuse of notation, whenever it does not cause a misunderstanding, we will not distinguish in the notation the characteristic function $R : \Sigma^* \times \Sigma^* \longrightarrow \{0, 1\}$ of the relation.

The notation $FR$ will be used for the problem of computing for any given $x \in \Sigma^*$ as input, a $y \in \Sigma^*$ so that $(x, y) \in R$, or if such a $y$ does not exist, giving the answer 'no'. We will say that $R$ *recognizes*

$$L_R := \{x \in \Sigma^* : \text{ there exists } y \in \Sigma^*, (x, y) \in R \},$$

and that $FR$ is the *search problem* associated to $R$.

We will also say that $L_R$ (or the problem 'Is $x \in L_R$ ?') is the *decision problem associated to* $FR$. A search problem is always defined along a relation, so every search problem is associated to a unique decision problem. However, it is not at all true that a decision problem (language) is associated with a unique search problem. For any language $L$ there are many relations $R$ so that $L = L_R$, of various nature and difficulty. (We will actually see many interesting relations $R$ so that $L_R = \Sigma^*$.) Therefore, it does not make any sense to speak about one search problem associated to a decision problem.

If $R$ is a relation, and there exists a polynomial $p(n) = p_R(n)$ such that for all $x \in L_R$ there exists $y \in \Sigma^*$, $|y| \leq p(|x|)$ so that $(x, y) \in R$, and $R(x, y)$ can be computed in $p(|x|)$ time, then $R$ will be called an $\mathcal{NP}$-relation. A substitution $x \in \Sigma^*$ for the first variable is called an *input*, and then for $x \in L_R$, a $y$ as in the definition is called a *certificate* for $x$. If in addition $L_R \in \mathcal{P}$, then $R$ will be called a $\mathcal{P}$-relation.

For a language $L \in \mathcal{NP}$ a problem of *searching after a certificate for* $L$ is a search problem $FR$, where $R$ is an $\mathcal{NP}$-relation, and $L = L_R$. There can be many such relations for $L$ , but the role of all is the same: they provide a certificate for $x \in L$, that can be checked in polynomial time. Many other examples show several essentially different relations $R$: for instance imperfectness can be certified with a partitionable subgraph, and if the SPGC is true, also with an odd hole or odd antihole.

Recall

$$\mathcal{NP} := \{L \subseteq \Sigma^* : \text{ there exists an } \mathcal{NP}\text{-relation } R \text{ such that } L_R = L\}.$$

$$F\mathcal{NP} := \{FR : R \text{ is an } \mathcal{NP}\text{-relation}\}.$$

$$F\mathcal{P} := \{FR : R \text{ is a } \mathcal{P}\text{-relation}\}.$$

The polynomial reductions from one problem to another have to be understood as usual (we can think of Cook reductions, but everything would work as well by using Karp reductions see [52]). Of course, for search problems not only the input of problem A has to be mapped in polynomial time to the input of problem B, but also the output of problem B has to be mapped back to an output of problem B. In both of these requirements hold, we will say that problem $A$ is *reducible* to problem $B$ in polynomial time. We say that $A$ and $B$ are *polynomially equivalent*, if each can be reduced to the other in polynomial time.

We have to distinguish from $F\mathcal{P}$ the class $\mathcal{FP}$ of *search problems solvable in polynomial time* (in the literature only this one is used). ($F\mathcal{P} \neq \mathcal{FP}$, unless $\mathcal{P} = \mathcal{NP}$ see Section 3.2.3) !

If HAMILTONIAN PATH $(G, P)$ is defined to be 1 if and only if $P$ is a Hamiltonian path of the graph $G$, then FHAMILTONIAN PATH is the problem of *finding a Hamiltonian path*. It is easy to reduce this problem (polynomially) to the existence of a Hamiltonian path. (Delete the edges one by one, and ask whether there is still a Hamiltonian path.) It is quite commonly thought that decision problems describe well the problems we meet 'in life', and that defining the complexity of search problems with the complexity of deciding the existence of the searched object is a reasonnable simplification.

This common prejudice occurs to be false ! The easier the problem $L$ is, the bigger the difference between complexity of $L$ and $FL$ can be ! The prejudice comes from the fact that for $\mathcal{NP}$-complete problems, the polynomial equivalence of $L$ and $FL$ is usually easy to prove. Selman [109] proved in general:

**Theorem 3.2** *If $R$ is an $\mathcal{NP}$-relation and $L_R$ is $\mathcal{NP}$-complete, then $L_R$ and $FR$ are polynomially equivalent.*

One cannot state, however, a similar result about problems in $\mathcal{NP} \cap co\mathcal{NP}$, as will be explained in the following Section 3.2.3. While most of *the decision problems in $\mathcal{NP} \cap co\mathcal{NP}$ turn out to be polynomially solvable, in $F(\mathcal{NP} \cap co\mathcal{NP})$ there are classes of polynomially equivalent problems for which this is not believed to be the case ! For some of these problems finding the good certificate occurs to be essentially more difficult than the decision problem (even if this, like many other negative results, cannot be proved).*

Even concerning $\mathcal{NP}$-complete problems one must be careful with the relation of the optimization problems and the corresponing decision problems:

for instance Integer Programming is not the search problem of finding any kind of certificate to the corresponding decision problem or its complement (see 3.2.5) !

### 3.2.3 Total Relations

A relation $R \subseteq \Sigma^* \times \Sigma^*$ is called a *total relation* if for all $x \in \Sigma^*$ there exists $y \in \Sigma^*$ so that $(x, y) \in R$. We will be interested in total $\mathcal{NP}$-relations, that is, in total $\mathcal{NP}$-relations for which every input $x \in \Sigma^*$ has a polynomial certificate.

A polynomial certificate for the set of *all* words ? We are more used to certifying a nontrivial set of words (language). In 3.2.1 though we saw such nontrivial search problems where deciding the existence of the searched object is just the recognition of $\Sigma^*$ ('dumb yes'); we will see some more examples to this in the next subsubsection. Don't forget that we are no more recognizing languages, but searching after certificates for relations. This subsubsection is devoted to clarifying this difference.

In other words, $R$ is a total relation if and only if $L_R = \Sigma^*$. Define

$$TF\mathcal{NP} := \{FR : R \text{ is an } \mathcal{NP}\text{-relation, } L_R = \Sigma^* \ \}.$$

That is, $TF\mathcal{NP}$ consists of the problems of searching after the certificates for total $\mathcal{NP}$-relations.

If $R$ is a total relation, even though $L_R (= \Sigma^*)$ *is trivial to recognize, the examples already show that $FR$ can be nontrivial !* This warns against associating search problems merely to languages, as it is erronously written in the books and papers I saw on the subject. Different relations may of course recognize the same language. (For instance: total relations all recognize $\Sigma^*$; this leads to a big variety of relations and search problems that recognize $\Sigma^*$ by considering $R' := R \cup \bar{R}$, where $R$ and $\bar{R}$ are the polynomially testable relations recognizing a well-caracterized language and its complement, respectively. (We have just borrowed the proof of Fact 3.1 below.) It is therefore important to associate search problems to *relations*, and not to decision problems !

Total relations have occurred much earlier in the literature than the recognition of differences between decision and search, or the completeness results concerning subclasses of $TF\mathcal{NP}$. Poljak [90] calls them 'existence theorems', and such theorems are mentioned first in [91]. Cameron and Edmonds [17] call such relations EP-theorems, where EP stands for existentially polytime. Papadimitriou and his coauthors' theory realize the difference between decision and search, present subclasses and provide for them theorems analogous to Cook's theorem.

The following fact is inspired by the equality $TF\mathcal{NP} = F(\mathcal{NP} \cap co\mathcal{NP})$ of Meggido and Papadimitriou [83]. (This formulation drags the inaccuracy of the definition of a search problem along. Formally, a search problem does not search after a certificate of the complementary language.)

Let $\mathcal{P}_1$ and $\mathcal{P}_2$ be two classes of problems. Let us say that the *the problems in the classes $\mathcal{P}_1$ and $\mathcal{P}_2$ are polynomially equivalent*, if every $P_1 \in \mathcal{P}_1$ is polynomially equivalent to some $P_2 \in \mathcal{P}_2$, and every $P_2 \in \mathcal{P}_2$ is polynomially equivalent to some $P_1 \in \mathcal{P}_1$. (That is, if in the undirected bipartite graph of polynomial equivalences between the problems in these two classes as vertices there is no isolated point.)

**Fact 3.1** *The problems in the three classes $F\mathcal{P}$, $TF\mathcal{NP}$ and 'searching after certificates both for $L$ and for $\Sigma^* \setminus L$, $L \in \mathcal{NP} \cap co\mathcal{NP}$' are polynomially equivalent.*

**Proof.** Suppose first $L \in \mathcal{P}$, and show that $FL$ is polynomially equivalent to a problem in $TF\mathcal{NP}$. Indeed, fix a polynomial algorithm that recognizes $L$, and for all $x \in \Sigma^*$ let $r(x)$ encode the running of the algorithm with input $x$. Define $R := \{(x, r(x)) : x \in \Sigma^*\}$. Then $R$ can be computed in polynomial time by running the algorithm, and $L_R = \Sigma^*$. Hence $FR \in TF\mathcal{NP}$.

Now let $FR \in TF\mathcal{NP}$. Then $L_R = \Sigma^*$, and indeed, $FR$ itself is searching after a certificate for $\Sigma^*$, and it is polynomially equivalent to itself. Choosing any trivial certificate for $\emptyset = \Sigma^* \setminus L$, the equivalence remains true.

Last let $L \in \mathcal{NP} \cap co\mathcal{NP}$, let $R$ be an $\mathcal{NP}$ relation so that $L_R = L$, and $\bar{R}$ be an $\mathcal{NP}$ relation so that $L_R = \Sigma^* \setminus L$. (That is, $FR$ is searching after a certificate for $L$, and $F\bar{R}$ for $\Sigma^* \setminus L$.) We show a relation $T$ such that $L_T \in \mathcal{P}$, and $FT$ is polynomially equivalent to solving both $FR$ and $F\bar{R}$. Define $T := R \cup \bar{R}$. We have indeed, $L_T = \Sigma^* \in \mathcal{P}$. Let us show the polynomial equivalence of $FT$ with solving both $FR$ and $F\bar{R}$. Indeed, if $FT$ can be solved in polynomial time, then for $x \in L$ it provides $y$ such that $(x, y) \in R$, and for $x \in \Sigma^* \setminus L$ we have for the $y$ it provides: $(x, y) \in \bar{R}$. Therefore $FR$, $F\bar{R} \in F\mathcal{P}$. Conversely, if $FR$, $F\bar{R} \in F\mathcal{P}$ it follows similarly that $FT \in F\mathcal{P}$. $\Box$

We get the following immediate corollary:

**Fact 3.2** *If $\mathcal{P} \neq \mathcal{NP}$, then $F\mathcal{P} \neq \mathcal{FP}$.*

### 3.2.4 Subclasses and their difficult problems

We are getting now near to a most interesting question: how to prove about a 'trivial' decision problem with a dumb yes answer, that it is difficult. For instance, do we have any chance of proving that finding a kernel in a perfect graph (or a strong fractional kernel in an arbitrary graph) is difficult ?

Here again, there is no chance for lower bounds on the running time, and since by Fact thm:equal this problem is at most as difficult as any well-characterized problem there is also no chance of proving $\mathcal{NP}$-completeness. On the other hand, there are no completeness results known for problems in $\mathcal{NP}\cap$co$\mathcal{NP}$. However, it turns out that there are interesting completeness results in some subclasses containing many different kinds of problems !

A class that turns out to be crucial for our ultimate goal is the class $\mathcal{PPA}$. We do not wish to go into the formal details, but explain the main ideas instead, as one can understand from [89] or [5]:

A typical problem in this class is SMITH: given a cubic graph $G$ with a given Hamiltonian circuit $H$, find another Hamiltonian circuit. Let us consider the Huge Graph whose vertices are the Hamiltonian paths of $G$, and the edge $(H_1, H_2)$ is in the Huge Graph, if $H_1$ and $H_2$ have $n-2$ common edges. Thomason [119] gave an algorithmic proof of Smith's theorem stating that a cubic graph that has a Hamiltonian circuit has a second Hamiltonionian circuit, along the following lines: delete an edge of $H$, denote the endpoints of the resulting Hamiltonian path by 1 and 2; construct a series of Hamiltonian paths with 1 as endpoint, by adding to the last Hamiltonian path $P$ with endpoints 1 and $P$, the unique edge of $G$ incident to $p$ which is neither in $P$ nor was the deleted edge in the previous step; if $p1 \in E(G)$, a Hamiltonian circuit is found; if $p1 \notin E(G)$, there is a unique edge to delete, different from the one that was added, so that the result is a Hamiltonian path with endpoint 1. Each step of this algorithm uniquely determines the previous and the next step. Consequently the algorithm cannot cycle, and therefore $p1 \in E(G)$ holds after a finite number of steps, and again because of the unicity of the preceding and following steps the found Hamiltonian circuit must be different from the initial one.

It is noteworthy that *this algorithm leaves the executor unemployed: it (the executor) has no choice all along the execution.* It goes one by one to uniquely determined Hamiltonian paths, executing each step in polynomial time, until a Hamiltonian circuit is determined. In other words, the Huge Graph is the union of vertex disjoint paths in this case, and the algorithm does not do anything else but walking along one of the paths, without branching with 'if' commands. Let us call an algorithm 'without if commands' *dumb*. Surprisingly, the result of a dumb algorithm *cannot be foreseen without executing the algorithm*, Papadimitriou [88] actually proved that this is $\mathcal{NP}$-hard to find, (and even PSPACE-complete). It is known that Thomason's particular algorithm, does not always solve SMITH in polynomial time, see Cameron's analysis of Krawczyk's example [16]. (For readers who are not yet confused, let us mention that Papadimitriou proved: the problem of finding 'the other endpoint' of a fixed path, which is exactly the problem solved by Thomason's algorithm, is complete in a complexity class containing $\mathcal{NP}$, whereas SMITH is included in a subclass of $\mathcal{NP} \cap co\mathcal{NP}$ ! Therefore, if $\mathcal{NP} \neq co\mathcal{NP}$, then finding 'the other endpoint' is more difficult than 'finding an endpoint'.)

Papadimitriou [89] first notes that SMITH belongs to a general class of problems that can be put into the form: given a graph and an odd degree vertex of a Huge Graph related to the problem that can be explored in polynomial time, find another odd degree vertex. This class of problems is called $\mathcal{PPA}$. Moreover, he shows that every problem in $\mathcal{PPA}$ can be reduced to a problem on a Huge Graph whose local complexity differs from the original one only by a polynomial factor, and can be solved by a dumb algorithm. However, SMITH is not proved to be $\mathcal{PPA}$-complete, and in general no $\mathcal{PPA}$-complete problem is shown in [89].

Therefore a subclass of $\mathcal{PPA}$ is defined, where the edges of the Huge Graph are directed forming directed paths. This orientation provides an additional structure to the problem: in SMITH, given two Hamiltonian Paths we do not know which one will be reached first by the algorithm, if they will be reached at all. The subclass where the orientation of the edges is known without running the algorithm is denoted by $\mathcal{PPAD}$.

Examples of problems in $\mathcal{PPAD}$ are beyond the scope of this paper, we only point at [89] again. We especially recommand the reader to look at SPERNER, BROWER or P-LCP.

The only relation known between the classes is $\mathcal{PPAD} \subseteq \mathcal{PPA}$. Is it really impossible to prove $\mathcal{PPA} = \mathcal{PPAD} = \mathcal{PLS}$ ? One thing is sure: these three classes are syntactic, and therefore contain complete problems. Johnson, Papadimitriou and Yannakakis [71] prove for instance that LOKL is $\mathcal{PLS}$-complete. The status of SMITH is open. Completeness here has to be interpreted similarly to $\mathcal{NP}$-completeness: for instance, $\mathcal{PLS}$-completeness means that it a polynomial combinatorial algorithm for one of the problems would imply one for all the problems. On the other hand, a polynomial algorithm for so many different problems like LOKL, Linear programming, etc., is unlikely.

(The polynomial time reductions do not necessarily respect the borders of the subclasses, but we can afford here not to care. Indeed, accepting for instance that a $\mathcal{PPAD}$-complete problem A is 'difficult', we get a valuable negative result by reducing such a problem to a problem B in $\mathcal{PPA}$: we can get a '$\mathcal{PPAD}$-hard' problem even if the problem is not in $\mathcal{PPAD}$.)

Let us turn now back to kernels, and try to situate the complexity of the problems we are interested in. Recall kernels - and 'strong fractional kernels' as a compromise between fractional and integer solutions introduced in Subsection 2.2.3 - moreover it concerns a characterizing property of perfectness which may therefore be related to the complexity of perfectness test, as explained before. The notions could of course be useful for a much wider range of problems in the practice of combinatorial optimization.

Let us define STRONG FRACTIONAL KERNELS (SFK) to be the problem of finding a strong fractional kernel in an undirected graph, and KERNELS IN PERFECT GRAPHS (KPG) is the problem of finding a kernel in a perfect graph (see Section 2.2.3). The former is more general: a polynomial algorithm for SFK implies a polynomial algorithm for KPG using Aharoni and Holzman's method and G'rötschel, Lovász, Schrijver's algorithm [64]. The algorithmic solutions of these are based on an algorithm of Scarf [96] to a more general problem on matrices that we will denote by SCARF. The precise definition and details of these are beyond the scope of this paper; yet we will state some results that are not difficult to check by readers who know the details. We think these are important open problems on the borderline of graph theory, complexity theory and polyhedral combinatorics; at the same time at the border of polynomial solvability and completeness (see 3.2.3). Therefore, we wish to explain the phenomena related to their complexity, in more details.

**Fact 3.3** *SCARF, SFK, KPG* $\in \mathcal{PPA}$.

**Proof.** (Sketch) Let us define the vertices of a Huge Graph to be the sets of $m$ linearly independent columns of an $m \times m$ input matrix, ($m \in \mathbb{N}$). The problem of finding a basis with the wished properties can be formulated as the problem of finding a vertex of degree 1 different from an initial vertex in this Huge graph. This problem contains SFK according to Aharoni and Holzman [2], and SFK contains KPG using [2] and a polynomial algorithm for finding a maximum clique in a perfect graph [64]. (A formal proof matched with a corresponding treatment of Scarf's algorithm is worked out in [103]. ) $\square$

Scarf's algorithm is of the same spirit as Lemke's algorithm for the linear complementarity problem P-LPMC.

Note that again, like in Lemke's or Thomason's algorithm, Scarf's algorithm finds a degree 1 vertex in a given component of the Huge graph (which, for problems in $\mathcal{PPA}$ in general, is $PSPACE$-hard). A reduction from P-LPMC or SMITH to these problems looks hopeful.

**Problem 16** *Are SCARF, SFK, KPG* $\mathcal{PPA}$*-complete ? Are they at least* $\mathcal{PPAD}$*-hard ?*

Poljak [90], and Cameron, Edmonds [17], [18] provide further nice examples of total problems among others a rich collection of problems in the class $\mathcal{PPA}$. For those that look difficult why aren't there new completeness results ? No $\mathcal{PPA}$-complete problem has been exhibited although it is a syntactic class. Problems similar to Problem 16 could be asked for all of them, and actually for many more problems in combinatorial optimization.

The missing attention turned towards these questions comes from the ignorance of the existence of complete problems that provide the possibility of proving negative results.

Besides putting light to some aspects of the complexity of problems Papadimitrou and his coauthors' results put various combinatorial algorithms into a few number of boxes according to 'proof styles'. The phenomena pointed out by these procedures deserve further attention from Combinatorial Optimizers, as well as the related complexity classes. The latter could enlarge their possibilities of proving new negative results by proving completeness (or hardness) in the new complexity classes.

### 3.2.5 Beyond $\mathcal{NP}$

In 3.2.3 it was explained that a trivially existing object may be difficult to find. In this section we provide a tool for showing that a given problem $L$ is not likely to be in $\mathcal{NP}$ at all. This consists essentially in showing a reduction $f : \Sigma^* \longrightarrow \Sigma^*$ computable in polynomial time, such that $x \in K$ if and only if $f(x) \in \Sigma^* \setminus L$, where $K$ is an $\mathcal{NP}$-complete problem. Then $L \in \mathcal{NP}$ would imply $K \in$co$\mathcal{NP}$, and since $K$ is $\mathcal{NP}$-complete, this can hold only if $\mathcal{NP}$=co$\mathcal{NP}$.

This is useful, since many of the problems studied in combinatorial optimization are neither trivially in $\mathcal{NP}$ nor in co$\mathcal{NP}$. For some problems one of the two belongings can be proved, but is far from being trivial (for instance perfectness). We must be prepared to assume the disaster to renounce to both types of necessary and sufficient conditions, and to have an evidence for this ! Fortunately for perfectness this is not the case, even if the co$\mathcal{NP}$ certificate is not trivial to prove. I will either skip this section or collect examples: Ageev, Kostochka, Szigeti: nontrivial caract for $T$-joins. Open problems and maybe co-completeness results for list colorings.

In order to prove that $L$ is neither in $\mathcal{NP}$ nor in co$\mathcal{NP}$ we have to show two reductions, one for $L$ the other for its complementary problem $\bar{L} := \Sigma^* \setminus L$. This helps in excluding some kinds of 'nice' conjectures that one should *not* expect to be true. Another reason for introducing these notions here is to complete our explanations on search problems and to realize that one should be somewhat careful with the relation of decision an optimization problems.

Let IP (Integer Programming) denote the problem of deciding for a given $m \times n$ integer matrix $A$, integer vectors $b \in \mathbb{Z}^m$, $c \in \mathbb{Z}^n$, and $k \in \mathbb{Z}$, whether

$$k \leq \max\{c^T x : x \in Q\}, \text{ where } Q := \{x \in \mathbb{R}^n : Ax \leq b\} .$$

EXACTIP is the problem of deciding whether $k = \max\{c^T x : x \in Q\}$ holds. Let OPTIP be the corresponding optimization problem, that is, the problem of *finding the maximum, or even $x \in Q$ with maximum objective value $c^T x$.*

The following discussion is a variant of results in [88]. We choose integer programming as an example, instead of satisfiability or the travelling salesman, and this is not a big difference. However, with this example we would like to inspire sharpenings of the results we are stating: prove $\mathcal{DP}$- or $\mathcal{FP}^{\mathcal{NP}}$-completeness results for more particular problems !

OPTIP is not at all equal to any natural search problem FIP that searches after a certificate for the decision problem IP$\in \mathcal{NP}$. While FIP$\in \mathcal{FNP}$, already the problem of finding only *the value* of OPTIP clearly includes the problem of deciding whether a given number $w$ is *equal to the optimum*, that is at least a particular case of EXACTIP; despite its polynomial equivalence to the TSP the decision problem EXACTIP is neither trivially in $\mathcal{NP}$ nor trivially in co$\mathcal{NP}$ ! Probably it is in neither of them, (and also not if $k$ is optimum), as the following results suggest:

The language $L \subseteq \Sigma^*$ is said to be in $\mathcal{DP}$ if there exist $L_1 \in \mathcal{NP}$ and $L_2 \in \mathcal{NP}$ such that $L = L_1 \cap L_2$. $L \in \mathcal{DP}$ is said to be $\mathcal{DP}$-complete, if any language in $\mathcal{DP}$ can be reduced to it.

The following straightforward statement unveils what one should actually do for proving $\mathcal{DP}$-completeness:

**Fact 3.4** *Let $L$ be a language. Then $L \in \mathcal{DP}$ if and only if $\bar{L} \in \mathcal{DP}$. Moreover, $L$ is $\mathcal{DP}$-complete if and only if $L \in DP$, and there exists two reductions $f, g : \Sigma^* \longrightarrow \Sigma^*$ both computable in polynomial time, and two $\mathcal{NP}$-complete problems $L_1$ and $L_2$ such that $x \in L_1$ if and only if $f(x) \in L$ and $x \in L_2$ if and only if $g(x) \in \Sigma^* \setminus L$; if $L$ is $\mathcal{DP}$-complete, then $L \in \mathcal{NP}$ if and only if $\mathcal{NP}$=co$\mathcal{NP}$.*

Applying this, one can easily get:

**Fact 3.5** *[88] EXACTIP is $\mathcal{DP}$-complete.*

We also introduce the first class of the polynomial hierarchy because of its seemingly frequent occurrence in integer optimization, where problems like OPTIP find their place: $\mathcal{P}^{\mathcal{NP}}$ is the class of problems solvable in polynomial time extending the usual operations by calls of an arbitrary $\mathcal{NP}$-complete problem $N$; $\mathcal{FP}^{NP}$ is the family of search problems that can be solved in polynomial time using also the oracle $N$ as operation. (Clearly, the definition does not depend on the choice of $N$. Don't convuse $\mathcal{FP}$ with $F\mathcal{P}$, recall that the latter is the search problem defined by a $\mathcal{P}$-relation.)

Obviously, $\mathcal{DP} \subseteq \mathcal{P}^{\mathcal{NP}}$.

**Fact 3.6** *[88] OPTIP is $\mathcal{FP}^{NP}$-complete.*

A similar claim to Fact 3.4 holds for $\mathrm{F}\mathcal{P}^{NP}$. Did any combinatorial optimizer ever care of providing two reductions for $\mathcal{NP}$-hard problems, one for each of $L$ or $\bar{L}$ corresponding to the '$\mathcal{NP}$-part' of the reduction?

If such results are proved for a problem, it is not advisable to look for $\mathcal{NP}$-characterizations. The same kind of negative result may hold for Problem 13:

**Problem 17** *Prove or disprove that the following problem is $\mathcal{DP}$-complete: Given an $m \times n$ integer matrix $A$, an integer vector $b \in \mathbb{Z}^m$, $Q := \{x \in \mathbb{R}^n : Ax \leq b\}$, is the Chvátal rank of $Q$ at most one.*

We hope the readers will find more examples of $\mathcal{DP}$ completeness or $\mathcal{PPA}$-completeness (or $\mathcal{LS}$-completeness) results, etc., or in lack of these, get encouragement for finding polynomial algorithms for their favorit problems. To reach this latter goal for testing the integrality of polyhedra one may need to use similar methods to those mentioned or referred in the previous sections.

# References


[1] Appel and Haken, The solution of the four-color-map problem, Sci. Amer. 237 (1977) 108-121.

[2] R. Aharoni, R. Holzman, Fractional Kernels in Digraphs, Journal of Combinatorial Theory/B, May 1998.

[3] Th. Bang, On the sequence $[n\alpha]$, $n = 1, 2, \ldots$ Supplementary note to the preceding paper by Th. Skolem., Math. Scand., 5, (1957), 69–76.

[4] A. Barvinok, A polynomial time algorithm for counting integral points in polyhedra when the dimension is fixed, *Math. Oper. Res.*, 19 (1994), 769–779.

[5] C. Bazgan, Approximation de problèmes d'optimisation et de fonctions totales de $\mathcal{NP}$, thèse de Docteur en Sciences de l'université Paris XI Orsay.

[6] C. Berge, The rank of a family of sets and some applications in graph theory, in Recent Progress in Combinatorics (Proceedings of the Third Waterloo Conference on Combinatorics, 1968, W. Tutte editor), Academic Press, New York 1969, 49–57.

[7] C. Berge, Balanced Matrices, Math. Programming, 2, 1972, 19–31.

[8] W. Bienia, L. Goddyn, P. Gvozdjak, A. Sebő, M. Tarsi, Flows, View Obstructions, and the Lonely Runner, J. Comb. Theory/B, Vol 72, No 1, 1998.

[9] R. Bixby, W. Cunningham, Converting linear programs to network problems, Math. Oper. Res., 5, 321-357.

[10] A. Bockmayr, F. Eisenbrand, Cutting Planes and the Elementary Closure in Fixed Dimension, Technical Report, MPI-I-99-2-008

[11] N. Brauner, Y. Crama, Some Facts and Questions about the Maximum Deviation Just-in-time Problem, Papiers de Recherches, G.E.M.M.E, No 0013, September 2000.

[12] E. Boros, V. Gurvich, Perfect graphs are kernel solvable, Discrete Math. (1996), 159, No.1-3, 35-55.

[13] C. Bouvier, G. Gonzalez-Sprinberg, Système gnrateurs minimal, diviseurs essentiels et $G$-dsinularisations de variétés toriques, Tôhoku Math. J., 46 (1994), 125–149.

[14] W. Bruns, J. Gubeladze, Normality and Covering Properties of Affine Semigroups, manuscript.

[15] W. Bruns, J. Gubeladze, M. Henk, A. Martin, and R. Weismantel, A counterexample to an integer analogue of Caratheéodory's theorem, to appear in *J. reine angew. Math.*.

[16] K. Cameron, Krawczyk's graphs show Thomason's algorithm fo finding a second Hamilton cycle through a given edge in a cubic graph is exponential, Fifth Czech-Slovak International Symposium on Combinatorics, Graph THoeory and Applications, Prague, July 1998.

[17] C. Cameron, J. Edmonds, Existentially polytime theorems, DIMACS series, Discrete Mathematics and Theoretical Computer Science, 1 (1990) 83–99.

[18] C. Cameron, J. Edmonds, Some Graphic Use of an Even Number of Odd Nodes, Annales de l'Institut Fourier, Tome 49 (3) (1999), Proceedings of the Symposium in memoriam Franois Jaeger, 815–827.

[19] A. Caprara, M Fischetti, $\{0, 1/2\}$-Chvátal-Gomory cuts, Mathematical Programming, 74, (1996), 221–235.

[20] J.W.S. Cassels, An Introduction to the Geometry of Numbers, Springer, Berlin, 1959.

[21] I. Charon, A. Germa, O. Hudry, Scores and Slater Index of Tournaments, École Nationale des Télécommunications, manuscript, 1998.

[22] V. Chvátal, Edmonds' polytopes and a hierarchy of combinatorial problems, Discrete Mathematics, 1973, 305–337.

[23] V. Chvátal "On certain polytopes associated with graphs", Journal of Combinatorial Theory/B, 18, 1975, 138–154.

[24] M. Conforti, G. Cornuéjols, C. Vušković, Perfect, ideal and balanced matrices, in Dell'Amico, Mauro (ed.) et al., Annotated bibliographies in combinatorial optimization, Chichester, Wiley. 81-94 (1997).

[25] M. Conforti, G. Cornuejols, M.R. Rao, Decomposition of Balanced Matrices, Journal of Combinatorial Theory, Series B 77, 292-406 (1999).

[26] W. Cook, On box totally dual integral polyhedra, Mathematical Programming 34 (1986), 48–61.

[27] W. Cook,C.R. Coullard, and Gy. Turán, On the Complexity of Cutting Plane Proofs, Discrete Applied Mathematics, 18, (1987), 25–38.

[28] W. Cook, W. H. Cunningham, W. R. Pulleyblank, A. Schrijver, Combinatorial Optimization, Wiley, 1998.

[29] W. Cook, J. Fonlupt, A. Schrijver, An integer analogue of Carathéodory's theorem, Journal of Combinatorial theory (B) **40** (1986) 63–70.

[30] W. Cook, L. Lovász, A. Schrijver, A polynomial time test for total dual integrality in fixed dimension, Mathematical Programming Study, 22 (1984), 64–69.

[31] D. Cox, J. Little, D. O'Shea, Ideals, Varieties and ALgorithms, An Introduction to Computational Algebraic Geometry and Commutative Algebra, Springer (1991).

[32] G. Cornuéjols, Combinatorial Optimization, SIAM, CBMS-NSF, Regional conference series in Applied Mathematics, 74 (2001).

[33] G. Cornuéjols, B. Guenin, Ideal Binary Clutters, Connectivity and a Conjecture of Seymour, preprint, Carnegie Mellon University, Pittsburgh, PA (2000).

[34] G. Cornuéjols, B. Novick, Ideal 0,A Matrices, Journal of Combinatorial Theory Series B 60 (1994) 145–157.

[35] W. H. Cunningham, A. B Marsh, A primal algorithm for optimum matching, Math. Programming Study, 8 (1987), 50-72.

[36] W. H. Cunningham, Testing membership in matroid polyhedra, J. Combinatorial Theory B 36, (1987) 161-188.

[37] W. H. Cunningham, J. Edmonds, A combinatorial decomposition theory, Canadian Journal of Mathematics, 32, 734-765.

[38] T. W. Cusick, View-obstruction problems in $n$-dimensional geometry, *J. Combin. Theory Ser. A* **16** (1974), 1–11.

[39] T. W. Cusick, View-obstruction problems. II, *Proc. Amer. Math. Soc.* **84** (1982) 25–28.

[40] T. W. Cusick, C. Pomerance, View-obstruction problems. III, *J. Number Theory* **19** (1984) 131–139.

[41] J. Edmonds (1967) Edge-disjoint branchings, in: Combinatorial algorithms, Academic Press, New York.

[42] J. Edmonds (1967), Submodular functions, matroids, and certain polyhedra, in: R. Guy, H. Hanani, N. Sauer and J. Schnheim eds., Combinatorial Structures and their Applications, Gordon and Breach, New York, 1970, 69–87.

[43] J. Edmonds, R. Giles, Total dual integrality of linear inequality systems, in: Progress in Combinatorial Optimization, Jubilee Conference, W.R. Pulleyblank ed., Academic Press '1984) , Toronto

[44] J. Edmonds, E. Johnson, Matching: A well solved class of integer linear programs, in 'Combinatorial Structures and Their Applicatuions' (R. Guy et al eds) pp 89–92, Gordon and Breach, New York, 1976.

[45] F. Eisenbrand, A note on the membership problem for the first elementary closure of a polyhedron, Combinatorica, 19 (2), (1999), 297–300.

[46] F. Eisenbrand, A. Schulz, Bounds on the Chvátal Rank of Polytopes in the 0/1-Cube., in Lecture Notes in Computer Science 1610, (Cornuéjols, Burkard, Woeginger Eds.), Springer, June 1999.

[47] F. Eisenbrand, private communication.

[48] G. Ewald, Combinatorial Convexity and Algebraic Geometry, Springer Verlag, New York, 1996.

[49] A. S. Fraenkel, Complementing and exactly covering sequences, Journal of Combinatorial Theory, Ser. A, 14 (1973), 8–20.

[50] D.R. Fulkerson, "The perfect graph conjecture and the pluperfect graph theorem", in: R.C. Bose et al. eds., Procedings of the Second Chapel Hill Conference on Combinatorial Mathematics and its applications, University of North Carolina, 1970, 171–175.

[51] D.R. Fulkerson, A. Hoffman, R. Oppenheim, On balanced matrices, Math. Programming Study, 1, 1974, 120-132.

[52] M.R. Garey, and D.S. Johnson (1979), Computers and Intractability: a Guide to the Theory of $\mathcal{NP}$-completeness, Freeman, San Francisco

[53] G. Gasparyan, Minimal imperfect graphs: A simple approach, *Combinatorica* 16 (2) (1996), 209–212.

[54] G. Gasparyan, Bipartite designs, in IPCO 6 (Houston), (Bixby and Rios-Mercado eds.), Lecture Notes in Computer Science, 1412, Springer, May 1998, pp 23-35.

[55] G. Gasparyan, M. Preissmann, A. Sebő, Imperfect and Nonideal Clutters: a Simple Approach, manuscript, submitted to Combinatorica.

[56] A. M. H. Gerards, A short proof of Tutte's characterization of totally unimodular matrices, Linear ALgebra Appl. 114/115 (1989), 207–212.

[57] A. M. H. Gerards, Multicommodity Flows and Polyhedra, CWI Quarterly, Volume 6, Number 3 (1993).

[58] A. M. H. Gerards, A. Sebő, Total dual integrality implies local strong unimodularity, Mathematical Programming, 8 (1987), 69–73

[59] A. M. H. Gerards, A. Schrijver, Matrices with the Edmonds-Johnson property, Combinatorica, 6 (1986), 365-379.

[60] R Giles, W. Pulleyblank (1979), Total dual integrality and integer polyhedra, Linear algebra and its applications, 25, 191–196.

[61] R.E. Gomory, Outline of an algorithm for integer solutions to linear programs, Bulletin of the American Mathematical Society, 64 (1958), 275–278.

[62] R.E. Gomory, Some Polyhedra Related to Combinatorial Problems, Linear Algebra and Its Applications, 2 (1969), 451–558.

[63] M. Grötschel, L. Lovász, A. Schrijver, *Polynomial Algorithms for Perfect Graphs*, Topics on Perfect Graphs in Annals of Discrete Mathematics (21), 1984.

[64] M. Grötschel, L. Lovász, A. Schrijver, *Geometric Algorithms and Combinatorial Optimization*, Springer Verlag, Berlin Heidelberg (1988).

[65] B. Guenin, A Characterization of weakly bipartite graphs, in Integer Programming and Combinatorial Optimization (Bixby, Boyd, Rios-Mercado, eds.) Lecture Notes in Computer Science 1412 (1998) 9-22.

[66] M. Hartmann, M. Queyranne and Y. Wang [66], On the Chvátal rank of certain inequalities, in Lecture Notes in Computer Science 1610, 7th International IPCo conference, (Cornuéjols, Burkard, Woeginger Eds.), Springer Verlag, Graz, June 1999.

[67] M. Henk, R. Weismantel, Hilbert bases of cones related to simultaneous Diphantine approximations and linear Diophantine equations, preprint (1997).

[68] A. J. Hoffman,J. B. Kruskal (1956), Integral boundary points of convex polyhedra, in Linear Inequalities and Related Systems (Kuhn and Tucker eds), Princeton University Press (1956), pp 223–246.

[69] A. J. Hoffman, R. Oppenheim (1978), Local unimodularity in the matching polytope, Annals of Discrete Mathematics, 2, 201–209

[70] A. J. Hoffman, On Polyhedral Combinatorics over a Totally Ordered Abelian Group, manuscript, 1999.

[71] D. S. Johnson, C.H. Papadimitriou, M. Yannakakis, How Easy Is Local Search ?, Journal of Computer and System Sciences, 37 (1988), 79–100.

[72] R. Kannan, A. Bachem, Polynomial algorithms for computing the Smith and the Hermite normal forms of an integer matrix, SIAM Journal on Computing, 8 (1979), 499–507.

[73] R. Kannan, Minkowski's convex body theorem and integer programming, Mathematics of Operations Research 12, (1987) 415–440.

[74] J-M. Kantor, On the width of lattice-free simplices, Composition Mathematica, 1999.

[75] L. Kronecker, Auseinandersetzung einiger Eigenschaften der Klassenanzahl idealen complexer Zahlen, Werke Vol. 1, Chelsea, New York (1968).

[76] A. Lehman, The Width-Length Inequality and Degenerate Projective Planes, Cook, W. and P. D. Seymour (eds.), *Polyhedral Combinatorics* DIMACS vol.1, (1990) 101–105.

[77] H.W. Lenstra Jr., Integer programming with a fixed number of variables, Mathematics of Operations Research, 8, (1983), 538–548.

[78] L. Lovász, A characterization of perfect graphs, *J. of Comb. Theory*/B, 13 (1972) 95–98.

[79] Lovász, L. Perfect graphs, Selected topics in graph theory 2, 55-87 (1983).

[80] L. Lovász, "An algorithmic theory of Numbers, Graphs and Convexity", SIAM (CBMS-NSF Regional conference series in applied mathematics).

[81] L. Lovász, A. Schrijver, Cones of matrices, set-functions and 0-1 optimization, SIAM J. Optim. 1, No.2, 166-190 (1991).

[82] C. Lütolf, F. Margot, A Catalog of Minimally Nonideal Matrices, Mathematical Methods of Operations Research 47 (1998) 221–241.

[83] N. Megiddo, C.H. Papadimitriou, On total functions, existence theorems, and computational complexity, Theoretical Computer Science 81 (1991) 317–324.

[84] G. L. Nemhauser, L. A. Wolsey, Integer and Combinatorial Optimization, Wiley, New York, 1988.

[85] B. Novick, Ideal 0,1 Matrices, Ph. D. dissertation, Carnegie Mellon University, Pittsburgh, PA (1990).

[86] J.G. Oxley, Matroid Theory, Oxford University Press (1992), Oxford, New York, Tokyo.

[87] C.H. Papadimitriou, M. Yannakakis, On Recognizing Integer Polyhedra, Combinatorica, **10** (1), 1990, pp 107-109.

[88] C.H. Papadimitriou, Computational Complexity, Addison-Wesley Publishing Company Inc, 1994.

[89] C.H. Papadimitriou, The Complexity of the Parity Argument and Other Inefficient Proofs of Existence, Journal of Computer and Sciences, 48 (1994), 498–532.

[90] S. Poljak, On Existence Theorems, Discrete Mathematics, 111 (1993), 423–434.

[91] S. Poljak, P. Pudlák, D. Turzik, Extension of $k$-subsets to $k+1$-subsets, Comm. Math. Univ. Car. 23 (1982), 337–349.

[92] M. Preissmann, A. Sebő, Some Aspects of Minimal Imperfect Graphs, to appear in a volume on perfect graphs (Ramirez-Alfonsin, Reed editors)

[93] B. Reznick, Lattice point simplices, Discrete Mathematics, 60, 1986, 219–242.

[94] J.E.Reeve, On the volume of lattice polyhedra, Proc. London Math. Soc. (3) 7 (1957), 378–395.

[95] N. Robertson, P. D. Seymour, D.Sanders, R. Thomas, The four-colour theorem, J. Combin. Theory Ser. B 70 (1997) 2-44 05C15

[96] H. Scarf, *The core of an $N$-person game*, Econometrica, 35, No 1, (1967), 50-59.

[97] H. Scarf, Integral polyhedra in three space, Math. Oper. Res., 10 (1985), 403–438.

[98] A. Schrijver, Min Max Results in Combinatorial Optimization, Mathematical Programming: The State of The Art, Bonn 1982, (Bachem, Grötschel, Korte eds.), Springer Verlag, Berlin, Heidelberg, New York, Tokyo.

[99] A. Schrijver, 'Theory of Integer and Linear Programming', Wiley, Chichester,1986.

[100] A. Sebő, Hilbert bases, Caratheodory's theorem and Combinatorial Optimization, Integer Programming and Combinatorial Optimization 1, (R. Kannan and W. Pulleyblank eds), University of Waterloo Press, Waterloo 1990, 431–456.

[101] A. Sebő, Characterizing Noninteger Polyhedra with 0-1 constraints, in IPCO 6 (Houston), (Bixby and Rios-Mercado eds.), Lecture Notes in Computer Science, 1412, Springer, May 1998, pp 37–52.

[102] A. Sebő, An introduction to empty lattice-simplices, in Lecture Notes in Computer Science 1610, (Cornuéjols, Burkard, Woeginger Eds.), Springer Verlag, June 1999.

[103] A. Sebő, Total Search, Parity Completeness and the Perfectness Test, to be submitted.

[104] E. C. Sewell, Binary integer programs with two variables per inequality, Mathematical Programming, 75 (1996) 467–476.

[105] P .D. Seymour, The matroids with the max-flow-min-cut property, J. Comb. Theory, Ser. B., 23, (1977), 189–222.

[106] P .D. Seymour, Decomposition of Regular Matroids, J. Comb. Theory, Ser. B., 28, (1980), 305-359.

[107] P .D. Seymour, Matroids and Multicommodity Flows, European Journal of Combinatorics, 2, (1981), 257-290.

[108] P .D. Seymour, On Lehman's width-length characterization, in Polyhedral Combinatorics (W. Cook, P. D. Seymour, eds.) DIMACS 1, AMS Providece, RI (1990) 107-117.

[109] A. Selman, Natural self-reducible sets, SIAM Journal on Computing, 17 (1988), 989-996.

[110] F.B. Shepherd, Applying Lehman's theorems to packing problems, Mathematical Programming, 71 (1995), 353–367.

[111] F.B. Shepherd, private communication.

[112] F. B. Shepherd, The Theta body and imperfection, to appear in 'Perfect Graphs', J. Ramirez Alfonsin and B. Reed eds, (2001).

[113] D. Shmoys, É. Tardos, Computational Complexity, Chapter in the handbook of Combinatorics II, Graham, Grötschel, Lovász eds, Elsevier (1995).

[114] Th. Skolem., On certain distributions of integers in pairs, with given differences, Math. Scand., 5, (1957), 57–68.

[115] H. J. S. Smith, On systems of linear indeterminate equations and congruences, Philosophical Transactions of the Royal Society of London (A) 151 (1861) 293–326.

[116] J. Stillwell, Classical Topology, and Combinatorial Group Theory, Graduate Texts in Mathematics, Springer Verlag 1993.

[117] B. Sturmfels, Gröbner Bases and Convex Polytopes, University Lecture Series, Volume 8, American Mathematical Society, Providence, Rhode Island (1996).

[118] A. Tamura, Y. T. Ikebe, Perfect bidirected graphs, manuscript, May 1996.

[119] A. Thomason, Hamilton cycles and uniquely edge-colourable graphs, Annals of Discrete MAthematics 3 (1978), 259–268.

[120] R. Tijdeman, Fraenkel's conjecture for six sequences, Discrete Mathematics 222 (2000) 223–234.

[121] W. T. Tutte, A homotopy theorem for matroids, I, II. Trans. Amer. MAth. Soc. 88, 144-174.

[122] W. T. Tutte, An algorithm for determining whether a given binary matroid is graphic, Proc. Amer. Math. Soc., 11 (1960), 905–917.

[123] W. T. Tutte, Lectures on Matroids, J. of Res. NAt. Bur. Standards, Sect. B 69B, (1965), 1–47.

[124] G. K. White, Lattice tetrahedra, Canadian J. Math. 16 (1964), 389–396.

[125] J. M. Wills, Zwei Sätze über inhomogene diophantische Approximation von Irrationalzahlen. *Monatsch. Math.* **71** (1967) 263–269.

[126] L. A. Wolsey, Cutting Plane Methods, in Operations Research Support Methodology (A. G. Holzman, ed.) Marcel Dekker, New York, 1979, 441–466.

Le laboratoire Leibniz est fortement pluridisciplinaire. Son activité scientifique couvre un large domaine qui comprend aussi bien des thèmes fondamentaux que des thèmes très liés aux applications, aussi bien en mathématiques qu'en informatique.

Les recherches sur les Environnements Informatiques d'Apprentissage Humain et la didactique des mathématiques ouvrent cette pluridisciplinarité sur les sciences humaines, elles jouent un rôle particulier en favorisant les coopérations entre différentes composantes du laboratoire.

* mathématiques discrètes et recherche opérationnelle
* logique et mathématique pour l'informatique
* informatique de la connaissance
* EIAH et didactique des mathématiques

*Les cahiers du laboratoire Leibniz* ont pour vocation la diffusion des rapports de recherche, des séminaires ou des projets de publication réalisés par des membres du laboratoire. Au-delà, Les cahiers peuvent accueillir des textes de chercheurs qui ne sont pas membres du laboratoire Leibniz mais qui travaillent sur des thèmes proches et ne disposent pas de tels supports de publication. Dans ce dernier cas, les textes proposés sont l'objet d'une évaluation par deux membres du Comité de Rédaction.




Contact Gestion & Réalisation : Jacky Coutin
Directeur de la publication : Nicolas Balacheff

ISSN : 1298-020X 