

## THE ELLIPSOID METHOD AND ITS CONSEQUENCES IN COMBINATORIAL OPTIMIZATION

M. Grötschel,\* L. Lovász,\*\* and A. Schrijver\*\*\*†

*\*Institut für Ökonometrie und Operations Research, Universität Bonn*

*\*\*Bolyai Institute, Josef Attila University, Szeged*

*\*\*\*Mathematisch Centrum, Amsterdam*

L.G. Khachian recently published a polynomial algorithm to check feasibility of a system of linear inequalities. The method is an adaptation of an algorithm proposed by Shor for non-linear optimization problems. In this paper we show that the method also yields interesting results in combinatorial optimization. Thus it yields polynomial algorithms for vertex-packing in perfect graphs; for the matching and matroid intersection problems; for optimum covering of directed cuts of a digraph; for the minimum value of a submodular set function; and for other important combinatorial problems. On the negative side, it yields a proof that weighted fractional chromatic number is NP-hard.

†Research supported by the Netherlands Organisation for the Advancement of Pure Research (ZWO).

0. INTRODUCTION.

A typical problem in combinatorial optimization is the following. Given a finite set  $S$  of vectors in  $\mathbb{R}^n$  and a linear objective function  $c^T x$ , find

$$(1) \quad \max\{c^T x \mid x \in S\}.$$

Generally  $S$  is large (say exponential in  $n$ ) but highly structured. For example,  $S$  may consist of all characteristic vectors of perfect matchings in a graph. We are interested in finding the value of (1) by an algorithm whose running time is polynomial in  $n$ . Therefore, enumerating the elements of  $S$  is not a satisfactory solution.

The following approach was proposed by Edmonds [1965], Ford and Fulkerson [1962] and Hoffman [1960], and is the classical approach in combinatorial optimization. Let  $P$  denote the convex hull of  $S$ . Then clearly

$$(2) \quad \max\{c^T x \mid x \in S\} = \max\{c^T x \mid x \in P\}.$$

The right hand side here is a linear programming problem: maximize a linear objective function on a polytope. Of course, to be able to apply the methods of linear programming, we have to represent  $P$  as the set of solutions of a system of linear inequalities. Such a representation, of course, always exists, but our ability to find the necessary inequalities depends on the structure of  $S$ . However, in many cases these inequalities (the facets of  $P$ ) can be described. There are some beautiful theorems of this kind, e.g. Edmonds' description of the matching polytope. In these cases, the methods of linear programming can be applied to solve (1). However, until about a year ago there were two main obstacles in carrying out the above program even for nice sets  $S$  like the set of perfect matchings. First, no algorithm to solve linear programming with polynomial running time in the worst case was known. Second, the number of inequalities describing  $S$  is typically large (exponential) and hence even to formulate the linear program takes exponential space and time. Indeed, the well-known efficient combinatorial algorithms, like Edmonds' matching algorithm [1965] or Lucchesi's algorithm to find optimum coverings for directed cuts [1976] are based on different - ad hoc - ideas.

A recent algorithm to solve linear programs due to L.G. Khachian [1979], based on a method of Shorr [1970], removes both difficulties. Its running time is polynomial; also, it is very insensitive to the number of constraints in the following sense: we do not need to list the faces in advance, but only need a

subroutine which recognizes feasibility of a vector and if it is infeasible then computes a hyperplane separating it from  $P$ . Searching for such a hyperplane is another combinatorial optimization problem which is often much easier to solve. It is interesting that if we want to apply the same method to this second problem, we get back the first one.

The main purpose of this paper is to exploit this equivalence between problems. After formulating the optimization problem in Chapter 1 exactly, we survey the "ellipsoid method" in Chapter 2. In Chapter 3 we prove the equivalence of the optimization and the separation problem, and their equivalence with other optimization problems. So we show that optimum dual solutions can be obtained by the method (since the dual problem has, generally in combinatorial problems, exponentially many variables, the method cannot be applied to the dual directly). Chapter 4 contains applications to the matching, matroid intersection, and branching problems, while in Chapter 5 we show how to apply the method to minimize a submodular set function and, as an application, to give algorithmic versions of some results of Edmonds and Giles [1977] and Frank [1979]. These include an algorithm to find optimum covering of directed cuts in a graph, solved first by Lucchesi [1976].

It is interesting to point out that these applications rely on the deep theorems characterizing facets of the corresponding polytope. This is in quite a contrast to previously known algorithms, which typically do not use these characterizations but quite often give them as a by-product.

The efficiency of the algorithms we give is polynomial but it seems much worse than those algorithms developed before. Even if we assume that this efficiency can be improved with more work, we do not consider it the purpose of our work to compete with the special-purpose algorithms. The main point is that the ellipsoid method proves the polynomial solvability of a large number of different combinatorial optimization problems at once, and hereby points out directions for the search for practically feasible polynomial algorithms.

Chapter 6 contains an algorithm to find maximum independent sets in perfect graphs. The algorithm makes use of a number  $\psi(G)$  introduced by one of the authors as an estimation for the Shannon capacity of a graph (Lovász [1979]). Finally, in Chapter 7 we note that the vertex-packing problem of a graph is in a sense equivalent to the fractional chromatic number problem, and comment on the phenomenon that this latter problem is an example of a problem in NP which is NP-hard but (as for now) not known to be NP-complete.

1. OPTIMIZATION ON CONVEX BODIES: FORMULATION OF THE PROBLEMS AND THE RESULTS

Let  $K$  be a non-empty convex compact set in  $\mathbb{R}^n$ . We formulate the following two algorithmic problems in connection with  $K$ .

- (1) Strong optimization problem: given a vector  $c \in \mathbb{R}^n$ , find a vector  $x$  in  $K$  which maximizes  $c^T x$  on  $K$ .
- (2) Strong separation problem: given a vector  $y \in \mathbb{R}^n$ , decide if  $y \in K$ , and if not, find a hyperplane which separates  $y$  from  $K$ ; more exactly, find a vector  $c \in \mathbb{R}^n$  such that  $c^T y > \max\{c^T x \mid x \in K\}$ .

Examples. Let  $K$  be the set of solutions of a system of linear inequalities

$$(3) \quad a_i^T x \leq b_i \quad (i = 1, \dots, m)$$

( $a_i \in \mathbb{R}^n$ ,  $b_i \in \mathbb{R}$ ). Then the strong separation problem can be solved trivially: we substitute  $x = y$  in the constraints. If each of them is satisfied,  $y \in K$ . If constraint  $a_i^T x \leq b_i$  is violated, it yields a separating hyperplane. On the other hand, the optimization problem on  $K$  is just the linear programming problem.

As a second example, let  $K$  be given as the convex hull of a set  $\{v_1, \dots, v_m\}$  of points in  $\mathbb{R}^n$ . Then the optimization problem is easily solved by evaluating the objective function at each of the given points and selecting the maximum. On the other hand, to solve the separation problem we have to find a vector  $c$  in  $\mathbb{R}^n$  such that

$$(4) \quad c^T y > c^T v_i \quad (i = 1, \dots, m)$$

So this problem requires finding a feasible solution to a system of linear inequalities; this is again essentially the same as linear programming.

Note that the convex hull of  $\{v_1, \dots, v_m\}$  is, of course, a polytope and so it can be described as the set of solutions of a system of linear inequalities as well. But the number of these inequalities may be very large compared to  $m$  and  $n$ , and so their determination and the checking is too long. This illustrates that the solvability of the optimization and separation problems depends on the way  $K$  is given and not only on  $K$ .

We do not want to make any a priori arithmetical assumption on  $K$ . Thus it may well be that the vector in  $K$  maximizing  $c^T x$  has irrational coordinates. In this case the formulation of the problem is not correct, since it is not clear

how to state the answer. Therefore we have to formulate two weaker and more complicated, but more correct problems.

- (5) (Weak) optimization problem: given a vector  $c \in \mathbb{R}^n$  and a number  $\epsilon > 0$ , find a vector  $y \in \mathbb{R}^n$  such that  $d(y, K) \leq \epsilon$  and  $y$  almost maximizes  $c^T x$  on  $K$ , i.e. for every  $x \in K$ ,  $c^T x \leq c^T y + \epsilon$ .
- (6) (Weak) separation problem: given a vector  $y \in \mathbb{R}^n$  and a number  $\epsilon > 0$ , conclude with one of the following: (i) asserting that  $d(y, K) \leq \epsilon$ , (ii) finding a vector  $c \in \mathbb{R}^n$  such that  $|c| \geq 1$  and for every  $x \in K$ ,  $c^T x \leq c^T y + \epsilon$ .

We shall always assume that we are given a point  $a_0$  and  $0 < r \leq R$  such that

- (7)  $S(a_0, r) \subseteq K \subseteq S(a_0, R)$ .

The second inclusion here simply means that  $K$  is bounded, where a bound is known explicitly; this is quite natural to assume both in theoretical and in (possible) practical applications. The first assumption, namely that  $K$  contains an explicit ball, is much less natural and we make it for purely technical reasons. What it really means is that  $K$  is full-dimensional, or at least we can find the affine subspace it spans and also that we can find a ball in this subspace contained in  $K$ . At the end of Chapter 3 we shall show that some assumption like this must be made.

So we define a convex body as a quintuple  $(K; n, a_0, r, R)$  such that  $n \geq 1$ ,  $K$  is a convex set in  $\mathbb{R}^n$ ,  $a_0 \in K$ ,  $0 < r \leq R$  and (7) is satisfied.

Let  $\mathcal{K}$  be a class of compact convex bodies. We assume that each  $K \in \mathcal{K}$  has some encoding. An input of the optimization problem for  $\mathcal{K}$  is then the code of some member  $K$  of  $\mathcal{K}$ , a vector  $c \in \mathbb{R}^n$ , and a number  $\epsilon > 0$ . Inputs of the other problems are defined similarly. The length of the input is defined in the (usual) binary encoding. Thus the length of the input is at least  $n + |\log r| + |\log R| + |\log \epsilon|$ . An algorithm to solve the optimization problem for the class  $\mathcal{K}$  is called polynomial if its running time is bounded by some polynomial of the size of the input.

The fact that the running time must be polynomial in  $|\log \epsilon|$  is crucial: it means that running the algorithm for  $\epsilon = \frac{1}{2}, \frac{1}{4}, \dots$  we get a sequence of approximations which converge exponentially fast in the running time. Other approximation algorithms for linear programming (Motzkin and Schoenberg [1954]) have only polynomial convergence speed. This exponential convergence rate enables

Khachian to obtain exact optimum in polynomial time (essentially by rounding) and us to give the combinatorial applications in this paper.

## 2. THE ELLIPSOID METHOD

Let us first describe the simple geometric idea behind the method. We start with a convex body  $K$ , included in a ball  $S(a_0, R) = E_0$ , and a linear objective function  $c^T x$ . In the  $k$ -th step there will be an ellipsoid  $E_k$ , which includes the set  $K_k$  of those points  $x$  of  $K$  for which  $c^T x$  is at least as large as the best found so far. We look at the centre  $x_k$  of  $E_k$ . If  $x_k \notin K$  then we take a hyperplane through  $x_k$  which avoids  $K$ . This hyperplane cuts  $E_k$  into two halves; we pick that one which includes  $K_k$  and include it in a new ellipsoid  $E_{k+1}$ , "smaller" than  $E_k$ . If  $x_k \in K$  then we cut with the hyperplane  $c^T x = c^T x_k$  similarly. The volumes of the ellipsoids  $E_k$  will tend to 0 exponentially and this guarantees that those centres  $x_k$  which are in  $K$  will tend to an optimum solution exponentially fast.

We now turn to the exact formulation of the procedure. Let  $K \subseteq \mathbb{R}^n$  be a compact convex set,  $S(a_0, r) \subseteq K \subseteq S(a_0, R)$ ,  $c^T x$  a linear objective function,  $|c| \geq 1$  and  $\epsilon > 0$ . Assume that there is a subroutine SEP to solve the separation problem for  $K$ . This means that given a vector  $y \in \mathbb{R}^n$  and  $\delta > 0$ , SEP either concludes that  $y \in S(K, \delta)$  or yields a vector  $d$  such that

$$(1) \quad \max\{d^T x \mid x \in K\} \leq d^T y + \delta.$$

To solve the optimization problem on  $K$  we run the following algorithm. Let

$$(2) \quad N = 4n^2 \left\lceil \log \frac{2R^2 |c|}{\epsilon} \right\rceil,$$

$$(3) \quad \delta = \frac{R^2 - \epsilon}{24(n-1)},$$

and

$$(4) \quad \left\{ \begin{array}{l} p = 5N \left\lceil \log \frac{12\sqrt{n}}{R^2} \right\rceil. \end{array} \right.$$

We now define a sequence  $x_0, x_1, \dots$  of vectors and a sequence  $A_0, A_1, \dots$  of positive definite matrices as follows. Let  $x_0 = a_0$  and  $A_0 = R^2 I$ . Assuming that  $x_k, A_k$  are defined, we run the subroutine SEP with  $y = x_k$  and  $\delta$ . If it concludes that  $x_k \in S(K, \delta)$  we say that  $k$  is a feasible index, and set  $a = c$ . If SEP yields a vector  $d \in \mathbb{R}^n$  such that  $|d| \geq 1$  and

$$(5) \quad \max(d^T x \mid x \in K) \leq d^T x_k + \delta,$$

then we call  $k$  an *infeasible index* and let  $a = -d$ . Next define

$$(6) \quad b_k = A_k a / \sqrt{a^T A_k a},$$

$$(7) \quad x_k^* = x_k + \frac{1}{n+1} b_k,$$

$$(8) \quad A_k^* = \frac{2n^2+3}{2n^2} (A_k - \frac{2}{n+1} b_k b_k^T),$$

and

$$(9) \quad x_{k+1} \approx x_k^*, \text{ and } A_{k+1} \approx A_k^*,$$

where the sign  $\approx$  means that the left hand side is obtained by rounding the right hand side to  $p$  binary digits, taking care that  $A_{k+1}$  is symmetric.

The sequence  $(x_k)$ ,  $k$  feasible, will give good approximations for the optimum solution of our problem. To prove this, we shall need some lemmas, which will also illuminate the geometric background of the algorithm.

First we introduce some further notation. Let

$$(10) \quad E_k = \{x \in \mathbb{R}^n \mid (x-x_k)^T A_k^{-1} (x-x_k) \leq 1\},$$

and

$$(11) \quad E_k^* = \{x \in \mathbb{R}^n \mid (x-x_k^*)^T A_k^{*-1} (x-x_k^*) \leq 1\}.$$

(2.1) LEMMA. The matrices  $A_0, A_1, \dots$  are positive definite. Moreover,

$$(12) \quad \|x_k\| \leq \|a_0\| + R \cdot 2^k, \quad \|A_k\| \leq R^2 \cdot 2^k, \quad \text{and } \|A_k^{-1}\| \leq R^{-2} \cdot 4^k.$$

PROOF. By induction on  $k$ . For  $k = 0$  all the statements are obvious. Assume that they are true for  $k$ . Then note first that

$$(13) \quad \|A_k^*\| = \frac{2n^2+3}{2n^2} \|A_k - \frac{2}{n+1} b_k b_k^T\| \leq \frac{2n^2+3}{2n^2} \|A_k\| \leq (1 + \frac{3}{2n^2}) R^2 \cdot 2^k$$

and so

$$(14) \quad \|A_{k+1}^*\| \leq \|A_k^*\| + \|A_{k+1} - A_k^*\| \leq (1 + \frac{3}{2n^2}) R^2 \cdot 2^k + n \cdot 2^{-P} \leq R^2 \cdot 2^{k+1}.$$

Further,

$$(15) \quad \|b_k\| = \frac{\|A_k a\|}{\sqrt{a^T A_k a}} = \sqrt{\frac{a^T A_k^2 a}{a^T A_k a}} \leq \sqrt{\|A_k\|} \leq R \cdot 2^k,$$

and so

$$(16) \quad \|x_{k+1}\| \leq \|x_k\| + \frac{1}{n+1} \|b_k\| + \|x_{k+1} - x_k^*\| \leq \|a_0\| + R \cdot 2^k + \frac{1}{n+1} R \cdot 2^k + \sqrt{n} \cdot 2^{-P} \leq \|a_0\| + R \cdot 2^{k+1}.$$

Finally we have

$$(17) \quad (A_k^*)^{-1} = \frac{2n^2}{2n^2+3} (A_k^{-1} + \frac{2}{n-1} \cdot \frac{aa^T}{a^T A_k a}),$$

as it is easy to verify by computation, and hence  $A_k$  is positive definite.

Further,

$$(18) \quad \|(A_k^*)^{-1}\| \leq \frac{2n}{2n+3} (\|A_k^{-1}\| + \frac{2}{n-1} \cdot \frac{\|a\|^2}{a^T A_k a}) \leq \frac{2n^2}{2n^2+3} (\|A_k^{-1}\| + \frac{2}{n-1} \|A_k^{-1}\|) < \frac{n+1}{n-1} \|A_k^{-1}\|.$$

Let  $\lambda_0$  denote the least eigenvalue of  $A_{k+1}$  and let  $v$  be a corresponding eigenvector,  $\|v\| = 1$ . Then

$$(19) \quad \lambda_0 = v^T A_{k+1} v = v^T A_k^* v + v^T (A_{k+1} - A_k^*) v \geq \|(A_k^*)^{-1}\|^{-1} - \|A_{k+1} - A_k^*\| \geq \frac{n-1}{n+1} \|A_k^{-1}\|^{-1} - \sqrt{n} \cdot 2^{-P} \geq \frac{n-1}{n+1} \cdot R^2 \cdot 4^{-k} - \sqrt{n} \cdot 2^{-P} > R^2 \cdot 4^{-(k+1)}.$$

This proves that  $A_{k+1}$  is positive definite and also that

$$(20) \quad \|A_{k+1}^{-1}\| = 1/\lambda_0 \leq R^{-2} \cdot 4^{k+1}. \quad \square$$

(2.2) LEMMA. Let  $\mu$  denote the  $n$ -dimensional volume. Then

$$(21) \quad \frac{\mu(E_{k+1})}{\mu(E_k)} < e^{-1/4n}.$$

PROOF. See Gács and Lovász [1979].  $\square$



Set

$$(22) \quad \zeta_k = \max\{c^T x_j \mid 0 \leq j < k, j \text{ feasible}\},$$

and

$$(23) \quad K_k = K \cap \{x \mid c^T x \geq \zeta_k\}.$$

(2.3) LEMMA.  $E_k \supseteq K_k$ , for  $k = 0, 1, \dots, N$ .

PROOF. By induction on  $k$ . For  $k = 0$  the assertion is obvious. Let  $x \in K_{k+1}$ . Then

$$(24) \quad x \in K_k \subseteq E_k,$$

and also

$$(25) \quad a_k^T x \geq a_k^T x_k - \delta,$$

where  $a_k$  equals the auxiliary vector  $a$  used in step  $k$  (if  $k$  is a feasible index we do not even have the  $\delta$  here). Write

$$(26) \quad x = x_k + y + t b_k,$$

where  $a_k^T y = 0$  (since  $b_k$  and  $a_k$  are not perpendicular because of the positive definiteness of  $A_k$ , such a decomposition of  $x$  always exists). By (24),

$$(27) \quad 1 \geq (y + t b_k)^T A_k^{-1} (y + t b_k) = y^T A_k^{-1} y + t^2 b_k^T A_k^{-1} b_k = y^T A_k^{-1} y + t^2.$$

Hence  $t \leq 1$ . On the other hand, (25) yields

$$(28) \quad -\delta \leq t a_k^T b_k = t \sqrt{a_k^T A_k a_k}.$$

Now we have

$$(29) \quad (x - x_{k+1})^T A_{k+1}^{-1} (x - x_{k+1}) \leq (x - x_k^*)^T A_k^{-1} (x - x_k^*) + R_1,$$

where the remainder term  $R_1$  can be estimated easily by similar methods to those in the proof of Lemma (2.1), and it turns out that  $R_1 < 1/12n^2$ .

For the main term we have

$$\begin{aligned}
 (30) \quad (x-x_k^*)^T A_k^{n-1} (x-x_k^*) &= \frac{2n^2}{2n^2+3} \left( (t-\frac{1}{n-1})b_k+y \right)^T \left( A_k^{-1} + \frac{2}{n-1} \frac{aa^T}{a^T A_k a} \right) \left( (t-\frac{1}{n-1})b_k+y \right) \\
 &= \frac{2n^2}{2n^2+3} \left( (t-\frac{1}{n-1})^2 + y^T A_k^{-1} y + \frac{2}{n-1} (t-\frac{1}{n-1})^2 \right) \leq \frac{2n^2}{2n^2+3} \left( \frac{n^2}{n^2-1} - \frac{2t(1-t)}{n-1} \right) \\
 &\leq \frac{2n^4}{(2n^2+3)(n^2-1)} + \frac{4t}{(n-1)\sqrt{a^T A_k a}} \leq 1 - \frac{1}{6n^2} + \frac{4t}{n-1} \|A_k^{-1}\| \leq 1 - \frac{1}{6n^2} + \frac{4tR^{-2} \cdot 4^N}{n-1}.
 \end{aligned}$$

Hence  $(x-x_{k+1})^T A_{k+1} (x-x_{k+1}) \leq 1$ , and so  $x \in E_{k+1}$ .  $\square$

Now we are able to prove the main theorem in this section.

(2.4) THEOREM. Let  $j$  be a feasible index for which

$$(31) \quad c^T x_j = \max(c^T x_k \mid 0 \leq k < N, k \text{ feasible}).$$

Then  $c^T x_j \geq \max(c^T x \mid x \in K) - \epsilon$ .

PROOF. Let us observe first that Lemmas (2.2) and (2.3) imply that

$$(32) \quad \mu(K_N) \leq \mu(E_N) \leq e^{-N/4n} \mu(E_0) = e^{-N/4n} R^n V_n,$$

where  $V_n$  is the volume of the  $n$ -dimensional unit ball. On the other hand, let

$$(33) \quad \zeta = \max(c^T x \mid x \in K)$$

and  $y \in K$  such that  $c^T y = \zeta$ . Consider the cone whose base is the  $(n-1)$ -dimensional ball of radius  $r$  and centre  $x_0$  in the hyperplane  $c^T x = c^T x_0$  and whose vertex is  $y$ . The piece of this cone in the half-space  $c^T x \geq c^T x_j$  is contained in  $K_N$ . The volume of this piece is

$$(34) \quad \frac{V_{n-1} \cdot r^{n-1} \cdot (\zeta - c^T x_0)}{n \|c\|} \left( \frac{\zeta - c^T x_j}{\zeta - c^T x_0} \right)^n \leq \mu(K_N) \leq e^{-N/4n} R^n V_n.$$

Hence

$$(35) \quad \zeta - c^T x_j \leq e^{-N/4n} R \cdot \left( \frac{\zeta - c^T x_0}{r} \right)^{\frac{n-1}{n}} \left( \frac{nV_n}{V_{n-1}} \right)^{1/n} \|c\|^{1/n}.$$

We still need an upper bound on  $\zeta$ . Since

$$(36) \quad |\xi - c^T x_0| = |c^T (y - x_0)| \leq \|c\| \|y - x_0\| \leq R \|c\|,$$

we finally have

$$(37) \quad \xi - c^T x_j < 2e^{-N/4n} \cdot \frac{R^2}{\epsilon} \|c\| \leq \epsilon. \quad \square$$

### 3. EQUIVALENCE OF OPTIMIZATION AND OTHER PROBLEMS

First we prove the equivalence of the separation problem and the optimization problem, for any given  $K$ . More exactly, this means the following.

(3.1) THEOREM. *Let  $K$  be a class of convex bodies. There is a polynomial algorithm to solve the separation problem for the members of  $K$ , if and only if there is a polynomial algorithm to solve the optimization problem for the members of  $K$ .*

A class  $K$  with this property will be called solvable.

PROOF. I. The "only if" part. In view of the results of Chapter 2, the only thing to check is that the algorithm described there is polynomial-bounded. This follows since by assumption, the subroutine SEP is polynomial, hence the number of digits in the entries of  $a$  is polynomial and so the computation of  $x_{k+1}$  and  $A_{k+1}$  requires only a polynomial number of steps. All other numbers occurring have only a polynomial number of digits, by Lemma (2.1). The number of iterations is also polynomial. Hence the algorithm runs in polynomial time.

II. The "if" part. Without loss of generality assume that  $a_0 = 0$ . Let  $K^*$  be the polar of  $K$ , i.e.,

$$(1) \quad K^* = \{u \mid u^T x \leq 1 \text{ for each } x \in K\}.$$

It is well-known that  $K^*$  is a convex body, and

$$(2) \quad S(0, 1/R) \subseteq K^* \subseteq S(0, 1/r).$$

If  $K$  is a class of convex bodies with  $a_0 = 0$ , let  $K^* = \{K^* \mid K \in K\}$ .

(3.2) LEMMA. *The separation problem for a class  $K$  of convex bodies with  $a_0 = 0$  is polynomially solvable iff the optimization problem is polynomially solvable for the class  $K^*$ .*

Since  $(K^*)^* = K$ , this lemma immediately implies the "if" part of the theorem: if the optimization problem is polynomially solvable for  $K$  then the separation problem is polynomially solvable for  $K^*$ . But then by part I, the optimization problem is polynomially solvable for  $K^*$  and so using the lemma again, it follows that the separation problem is polynomially solvable for  $K$ .

PROOF OF THE LEMMA. I. The "if" part. Let  $x^* \in K^*$ ,  $v \in \mathbb{R}^n$  and  $\epsilon > 0$ . Using the optimization subroutine for  $K$ , with objective function  $v$  and error  $\epsilon x$ , we get a vector  $z \in \mathbb{R}^n$  such that  $d(z, K) \leq \epsilon x$ , and

$$(3) \quad v^T z \geq \max\{v^T x \mid x \in K\} - \epsilon x.$$

Now if  $v^T z \leq 1$  then  $v^T x \leq 1 + \epsilon x$  and hence  $v_0 = \frac{1}{1 + \epsilon x} v \in K^*$ , whence  $d(v, K^*) \leq \epsilon$ .

On the other hand, if  $v^T z > 1$  then  $z$  is a solution of the separation problem for  $K^*$ . In fact, let  $z_0 \in K$  such that  $\|z - z_0\| \leq \epsilon x$ . Then for every  $u \in K^*$ ,

$$(4) \quad z^T u = (z - z_0)^T u + z_0^T u \leq \|u\| \cdot \|z - z_0\| + 1 \leq \epsilon + z^T v,$$

which proves that  $z$  is a solution of the separation problem for  $K^*$ .

II. The "only if" part follows by interchanging the roles of  $K$  and  $K^*$ .  $\square$

Let  $K$  and  $L$  be two classes of convex bodies. Define

$$(5) \quad K \wedge L = \{K \cap L \mid K \in K, L \in L, \dim K = \dim L, a_0(K) = a_0(L)\}.$$

(3.3) COROLLARY. If  $K$  and  $L$  are solvable then so is  $K \wedge L$ .

PROOF. The separation problem for  $K \wedge L$  goes trivially back to the separation problems for  $K$  and  $L$ .  $\square$

(3.4) COROLLARY. Let  $K$  be a class of convex bodies with  $a_0 = 0$ . Then  $K$  is solvable iff  $K^*$  is solvable.

The proof is trivial by Lemma (3.2).

Let  $\mathbb{R}_+^n$  be the non-negative orthant in  $\mathbb{R}^n$ . Next we study convex bodies  $K$  such that there are  $\rho > 0$ ,  $R > 0$  with

$$(6) \quad \mathbb{R}_+^n \cap S(0, \rho) \subseteq K \subseteq \mathbb{R}_+^n \cap S(0, R).$$

The anti-blocker of  $K$  is defined by

$$(7) \quad A(K) = \{y \in \mathbb{R}^n \mid y^T x \leq 1 \text{ for every } x \in K\}.$$

Moreover,  $A(K) = \{A(K) \mid K \in \mathcal{K}\}$ .

(3.5) COROLLARY. Let  $K$  be a class of convex bodies satisfying (6). Then  $K$  is solvable iff  $A(K)$  is solvable.

The proof is the same as that of Lemma (3.2).

Next we want to show that without the assumption that  $K$  contains a ball, there is no algorithm at all to solve the optimization problem. More exactly, consider the class of polytopes  $(K_\lambda; 2, 0, *, 1)$ , where the  $*$  means that no ball is supposed to be contained in  $K_\lambda$ , and

$$(8) \quad K_\lambda = \{(x_1, x_2) \mid 0 \leq x_1 \leq \lambda, x_2 = (4 + 4\sqrt{5})x_1\}.$$

First note that if  $\lambda$  is known then both optimization and separation algorithms are easily given, even in the strong sense:  $c^T x$  is maximized by either  $(0, 0)$  or  $(\lambda, (4 + 4\sqrt{5})\lambda)$ ; and if  $(y_1, y_2) \in \mathbb{R}^2$  then let

$$(9) \quad c = \begin{cases} (-1, 0) & \text{if } y_1 < 0; \\ (0, -1) & \text{if } y_1 \geq 0, y_2 < 0; \\ (-y_1 - 2y_2, y_1 + y_2) & \text{if } y_1 \geq 0, y_2 \geq 0, y_1^2 + y_1 y_2 - y_2^2 < 0; \\ (y_1 + 2y_2, -y_1 - y_2) & \text{if } y_1 \geq 0, y_2 \geq 0, y_1^2 + y_1 y_2 - y_2^2 > 0; \\ (1, 0) & \text{if } y_1 > \lambda, y_2 \geq 0, y_1^2 + y_1 y_2 - y_2^2 = 0. \end{cases}$$

If  $0 \leq y_1 \leq \lambda$ ,  $y_2 \geq 0$ , and  $y_1^2 + y_1 y_2 - y_2^2 = 0$  then conclude  $(y_1, y_2) \in K_\lambda$ . It is easy to check that this algorithm solves the separation problem for  $K_\lambda$ .

On the other hand, we show that there is no algorithm at all (even arbitrarily slow) which would use a separation oracle for the class  $\{K_\lambda \mid 0 \leq \lambda \leq 1\}$ , and would be able to maximize the objective function  $x_1$  (i.e.,  $c^T x$  with  $c = (1, 0)$ ) over  $K_\lambda$ . By this we mean an algorithm whose input is a black box which is known to solve the strong separation problem for one of the  $K_\lambda$ 's; but the box cannot be broken open to see which value of  $\lambda$  is there; and the algorithm should work regardless which separation algorithm is used by the black box. In particular, it must work if we use the above-described separation algorithm. But then our algorithm runs the same way for every  $0 \leq \lambda \leq 1$ : the only way it

could distinguish between different  $K_\lambda$ 's is to run the subroutine with an input vector  $(y_1, y_2)$  such that  $y_1 > 0$  and  $y_1^2 + y_1 y_2 - y_2^2 = 0$ . But then one of  $y_1, y_2$  is irrational and so it cannot be the input of an algorithm. So the algorithm cannot determine  $\lambda$ , a contradiction. Thus no such algorithm may exist.

Finally we show that for polytopes many of the results are even nicer. By a rational polytope we mean a quadruple  $(P; n, a_0, T)$  where  $P$  is a full-dimensional polytope (in  $\mathbb{R}^n$ ),  $a_0 \in \text{Int } P$ , and every component of  $a_0$  as well as of every vertex of  $P$  is a rational number with numerator and denominator not exceeding  $T$  in absolute value. (This definition is much in the spirit of our previous discussion: the vertices of  $P$  must be rational in order to be able to explicitly present them and explicit bounds must be known for their complexity.)

(3.6) THEOREM. Let  $(P; n, a_0, T)$  be a rational polytope. Then  $S(a_0, r) \subseteq P \subseteq S(a_0, R)$ , where  $R = 2nT$  and  $r = (2T)^{-n^2-n}$ . Furthermore, every facet of  $P$  can be written as  $a^T x \leq b$ , where  $a (\neq 0)$  is an integral vector,  $b$  is an integer, and the entries of  $a$  as well as  $b$  are less than  $T' = (nT)^n$ .

Thus every rational polytope can be viewed as a convex body, with  $r$  and  $R$  as above. A certain converse of this assertion holds as well.

(3.7) THEOREM. Let  $P \subseteq \mathbb{R}^n$  be a polytope,  $a_0 \in \text{Int } P$ , and assume that every component of  $a_0$  is a rational number with numerator and denominator less than  $T$  in absolute value. Also assume that every facet of  $P$  can be written as  $a^T x \leq b$ , where  $a (\neq 0)$  is an integral vector,  $b$  is an integer and the entries of  $a$  as well as  $b$  are less than  $T$  in absolute value. Then  $(P; n, a_0, T')$  is a rational polytope where  $T' = (nT)^n$ .

The proof of these two theorems is rather straightforward arithmetic and is omitted (cf. Lemmas 1-2 in Gács and Lovász [1979]).

(3.8) THEOREM. Let  $K$  be a class of rational polytopes. Suppose that  $K$  is solvable. Then the strong optimization problem and the strong separation problem are solvable for  $K$  in time polynomial in  $n$ ,  $\log T$ , and  $\log \|c\|$  (respectively  $\log S$ , where  $S$  is the maximum of the absolute values of the numerators and denominators occurring in  $y$ ).

PROOF. Let  $(P; n, a_0, T) \in K$ ,  $Q = 2T^2$ , and

$$(10) \quad d = Q^n c + (1, Q, \dots, Q^{n-1})^T.$$

We prove that  $\max\{d^T x \mid x \in P\}$  is attained at a unique vertex of  $P$  and that this vertex maximizes  $c^T x$  as well.

For let  $x_0$  be a vertex of  $P$  maximizing  $d^T x$  and let  $x_1$  be another vertex. Write

$$(11) \quad x_0 - x_1 = \frac{1}{a} z,$$

where  $0 < a < T^2$  is an integer and  $z = (z_1, \dots, z_n)^T$  is an integral vector with  $|z_1| < 2T^2 = Q$ . Then

$$(12) \quad 0 \leq d^T(x_0 - x_1) = \frac{1}{a} \left\{ Q^n c^T z + \sum_{j=1}^n Q^{j-1} z_j \right\}.$$

Here  $c^T z \geq 0$ , since it is an integer and if  $c^T z \leq -1$ , then the first term in the bracket is larger in absolute value than the second. Hence

$$(13) \quad c^T z = c^T(x_0 - x_1) \geq 0$$

for every vertex  $x_1$ , i.e.,  $x_0$  indeed maximizes the objective function  $c^T x$  over  $P$ . Also note that the second term is non-zero since  $z \neq 0$ . Hence

$$(14) \quad d^T(x_0 - x_1) \geq \frac{1}{a} \geq \frac{1}{T^2}$$

and so  $x_0$  is the unique vertex of  $P$  maximizing the objective function  $d^T x$ .

Now use the hypothesized polynomial algorithm to find a vector  $y \in \mathbb{R}^n$  such that

$$(15) \quad d(y, P) \leq \epsilon = \frac{1}{2} \|d\|^{-1} T^{-6}$$

and  $d^T y \geq d^T x_0 - \epsilon$ . We claim that

$$(16) \quad \|y - x_0\| \leq \frac{1}{2T^2}.$$

For let  $y_0$  be the point of  $P$  next to  $y$ . Represent  $y_0$  as a convex combination of  $n+1$  vertices of  $P$ , one of which is  $x_0$ :

$$(17) \quad y_0 = \sum_{i=0}^n \lambda_i x_i, \quad \lambda_i \geq 0, \quad \sum_{i=0}^n \lambda_i = 1.$$

Then by (14)

$$(18) \quad d^T y = d^T (y - y_0) + d^T y_0 \leq \epsilon \|d\| + \sum_{i=0}^n \lambda_i d^T x_i \leq \epsilon \|d\| + d^T x_0 - \frac{1 - \lambda_0}{T^2}.$$

Hence

$$(19) \quad \frac{1 - \lambda_0}{T^2} \leq \epsilon (\|d\| + 1) \leq 2\epsilon \|d\|$$

and

$$(20) \quad \|y - x_0\| \leq \|y - y_0\| + \|y_0 - x_0\| \leq \epsilon + (1 - \lambda_0) \left\| \sum_{i=1}^n \frac{\lambda_i}{1 - \lambda_0} x_i - x_0 \right\| \leq \epsilon + (1 - \lambda_0) 2T^2 \leq \epsilon + 2\epsilon \|d\| \cdot 2T^4 \leq \frac{1}{2T^3}.$$

Now it is rather clear how to conclude: round each entry of  $y$  to the next rational number with denominator less than  $T$ ; the resulting vector is  $x_0$ . The rounding can be done by using the technique of continued fractions. We leave the details to the reader.

The separation algorithm can be obtained by applying the previous algorithm to  $P^*$  (assuming that  $a_0 = 0$ , possibly after translation).  $\square$

If the strong separation problem concludes that  $y \in P$  then it is nice to have a "proof" of that, i.e., a representation of  $y$  as a convex combination of vertices of  $P$ . This problem can also be solved.

**(3.9) THEOREM.** Let  $K$  be a solvable class of rational polytopes. Then there exists an algorithm which, given  $(P; n, a_0, T) \in K$  and a rational vector  $y \in P$ , yields vertices  $x_0, x_1, \dots, x_n$  of  $P$  and coefficients  $\lambda_0, \lambda_1, \dots, \lambda_n \geq 0$  such that  $\lambda_0 + \lambda_1 + \dots + \lambda_n = 1$  and  $\lambda_0 x_0 + \lambda_1 x_1 + \dots + \lambda_n x_n = y$ , in time polynomial in  $n$ ,  $\log T$  and  $\log S$ , where  $S$  is the maximum absolute value of numerators and denominators of components of  $y$ .

**PROOF.** We construct a sequence  $x_0, x_1, \dots, x_n$  of vertices,  $y_0, y_1, \dots, y_n$  of points and  $F_1, F_2, \dots, F_n$  of facets of  $P$  as follows. Let  $x_0$  be any vertex of  $P$ , and let  $y_0 = y$ . Assume that  $x_i, y_i$  and  $F_i$  are defined for  $i \leq j$ . Let  $y_{j+1}$  be the last point of  $P$  on the semi-line from  $x_j$  through  $y_j$ , let



$$(21) \quad y'_{j+1} = y_{j+1} + \epsilon(y_j - x_j)$$

where  $0 < \epsilon < (nT)^{-3n^3}$ . Let  $F_{j+1}$  be a facet separating  $y'_{j+1}$  from  $P$ , and let  $x_{j+1}$  be a vertex of  $F_1 \cap \dots \cap F_{j+1}$ . It is straightforward to prove by induction that  $x_i, y_i \in F_j$  for  $j \geq i$ ,  $y \in \text{conv}(x_0, \dots, x_i, y_i)$ , and  $\dim(F_1 \cap \dots \cap F_j) = n-j$ . Hence  $x_n = y_n$  and so  $y$  is contained in the convex hull of  $x_0, \dots, x_n$ .

The procedure described above is easy to follow with computation. The vertex of  $F_1 \cap \dots \cap F_j$  can be obtained as follows. Let  $a_i^T x \leq b_i$  be the inequality corresponding to facet  $F_i$ ; then maximize the objective function  $(\sum_{i=1}^j a_i)^T x$ . We leave the details to the reader.  $\square$

The "dual" form of this theorem will also play an important role in the sequel. It shows that if we consider optimization on  $P$  as a linear program, an optimal dual basic solution can be found in polynomial time, if the class is solvable.

(3.10) THEOREM. Let  $K$  be a solvable class of rational polytopes. Then there exists a polynomial-bounded algorithm which, given  $(P, n, a_0, T) \in K$ ,  $c \in \mathbb{Z}^n$ , provides facets  $a_i^T x \leq b_i$  ( $i = 1, \dots, n$ ) and rationals  $\lambda_i \geq 0$  ( $i = 1, \dots, n$ ) such that  $\sum_{i=1}^n \lambda_i a_i = c$  and  $\sum_{i=1}^n \lambda_i b_i = \max\{c^T x \mid x \in P\}$ .

The proof is easy by considering  $K^*$ .

#### 4. MATROID INTERSECTION, BRANCHINGS AND MATCHINGS

We now apply the methods described in the previous chapters to a number of combinatorial problems. As said in the introduction our main aim is to show the existence of polynomial algorithms for certain combinatorial problems, and these algorithms are not meant as substitutes for the algorithms developed for these problems before (see Lawler [1976] for a survey). However, in the next chapters we shall show the existence of polynomial algorithms also for certain problems which were not yet solved in this sense. The algorithms found there, though polynomial, in general do not seem to have the highest possible rate of efficiency, and the challenge remains to find better algorithms.

First we apply the ellipsoid method to matroid intersection (cf. Edmonds [1970, 1979], Lawler [1970]). Note that given a matroid  $(V, x)$ , the corresponding matroid polytope is the convex hull of the characteristic vectors of independent sets. The idea is very simple: given an integral "weight" function  $w$  on  $V$ ,

the trivial "greedy algorithm" finds an independent set  $V'$  maximizing  $\sum_{v \in V'} w(v)$ . That is, it finds a vertex  $x$  of the corresponding matroid polytope maximizing the objective function  $w^T x$ , in time bounded by a polynomial in  $|V|$  and  $\log \|w\|$ . So the class of matroid polytopes is solvable. Therefore, by Corollary (3.3) also intersections of matroid polytopes are solvable. Since the intersection of two matroid polytopes has integer vertices again, this provides us with a polynomial algorithm for matroid intersection. (In fact, we obtain a polynomial algorithm for common "fractional" independent sets for any number of matroids.) Obviously, we may replace "matroid" by "polymatroid". In Chapter 5 we shall extend this algorithm to a more general class of polytopes, and we shall show there how to obtain optimal integral dual solutions.

In this application, and in the following examples we leave it to the reader to check that without loss of generality we may restrict the classes of polytopes to full-dimensional polytopes, and to find a vector  $a_0$  and a number  $T$  such that (i) each numerator and denominator occurring in the components of the vertices of the polytope, and in those of  $a_0$ , do not exceed  $T$  in absolute value, (ii)  $a_0$  is an internal point of the polytope, and (iii)  $\log T$  is bounded by a polynomial in the size of the original combinatorial problem (in most cases we have  $T = 1$ ).

Also the second application is illustrative for the use of the method. It shows the existence of a polynomial algorithm for finding optimum branchings in a directed graph (cf. Chu and Liu [1965], Edmonds [1967]). Let  $D = (V, A)$  be a digraph, and let  $r$  be some fixed vertex of  $D$ , called the root. A branching is a set  $A'$  of arrows of  $D$  making up a rooted directed spanning tree, with root  $r$ . A rooted cut is a set  $A'$  of arrows with  $A' = \delta^-(V')$  for some non-empty set  $V'$  of vertices not containing  $r$ , where  $\delta^-(V')$  denotes the set of arrows entering  $V'$ . It follows from Edmonds' branching theorem [1973] that the convex hull of the (characteristic vectors of) the sets of arrows containing a branching as a subset (i.e., the sets intersecting each rooted cut), is a polytope  $P$  in  $\mathbb{R}^A$  defined by the following linear inequalities:

- (i) (i)  $0 \leq x(a) \leq 1$   $(a \in A)$ ,
- (ii)  $\sum_{a \in A'} x(a) \geq 1$   $(A' \text{ rooted cut})$ .

So there exists an algorithm which, given a digraph  $D = (V, A)$ , a root  $r$ , and a nonnegative integral weight function  $w$  defined on  $A$ , determines a branching of minimum weight, in time polynomially bounded by  $|V|$  and  $\log \|w\|$ , if and only if the strong optimization problem is solvable for the class of polytopes  $P$  arising in this way. By Theorem (3.1) and (3.8) it is enough to show that the

strong separation problem is solvable. Indeed, if  $x \in \mathbb{R}^A$  one easily checks condition (i) above and one finds a separating hyperplane in case of violation. To check condition (ii), we can find a rooted cut  $A'$  minimizing  $\sum_{a \in A} x(a)$  in time polynomially bounded by  $|V|$  and  $\log T$  (where  $T$  is the maximum of the numerators and denominators occurring in  $x$ ), namely by applying Ford-Fulkerson's max flow-min cut algorithm to the corresponding network with capacity function  $x$ , source  $r$  and sink  $s$ , for each  $s \neq r$ . If the minimum is not less than 1 we conclude  $x \in P$ , and otherwise  $A'$  determines a separating hyperplane. (Again, see Chapter 5 for a more general approach.)

In fact this branching algorithm is one instance of a more general procedure. Let  $\mathcal{E}$  be a clutter, i.e., a finite collection of finite sets no two of which are contained in each other. The blocker  $B(\mathcal{E})$  of  $\mathcal{E}$  is the collection of all minimal sets intersecting every set in  $\mathcal{E}$  (minimal with respect to inclusion). E.g., if  $\mathcal{E}$  is the collection of branchings in a digraph, then  $B(\mathcal{E})$  is the collection of minimal rooted cuts. One easily checks that  $B(B(\mathcal{E})) = \mathcal{E}$  for every clutter  $\mathcal{E}$ . Sometimes an even stronger duality relation may hold. Let  $V = \cup E$ , and let  $P$  be the convex hull of the characteristic vectors (in  $\mathbb{R}^V$ ) of all subsets of  $V$  containing some set in  $\mathcal{E}$ . Clearly, each vector  $x$  in  $P$  satisfies:

$$(2) \quad \begin{array}{ll} (i) & 0 \leq x(v) \leq 1 \quad (v \in V), \\ (ii) & \sum_{v \in V'} x(v) \geq 1 \quad (V' \in B(\mathcal{E})), \end{array}$$

as these inequalities hold for characteristic vectors of sets in  $\mathcal{E}$ . In case  $P$  is completely determined by these linear inequalities,  $\mathcal{E}$  (or the hypergraph  $(V, \mathcal{E})$ ) is said to have the  $\mathbb{Q}_+$ -max flow-min cut-property or the  $\mathbb{Q}_+$ -MFMC-property (cf. Seymour [1977]). Thus the clutter of all branchings in a digraph has the  $\mathbb{Q}_+$ -MFMC-property, as we saw above. Fulkerson [1970] showed the interesting fact that a clutter  $\mathcal{E}$  has the  $\mathbb{Q}_+$ -MFMC-property if and only if its blocker  $B(\mathcal{E})$  has the  $\mathbb{Q}_+$ -MFMC-property.

Now one easily extends the derivation of a polynomial algorithm for branchings from such an algorithm for rooted cuts as described above, to the following theorem.

**(4.1) THEOREM.** Let  $\mathcal{C}$  be a class of clutters with the  $\mathbb{Q}_+$ -MFMC-property, such that there exists an algorithm which finds, given  $E \in \mathcal{C}$  and  $w \in \mathbb{Z}_+^V$  (where  $V = \cup E$ ), a set  $V'$  in  $E$  minimizing  $\sum_{v \in V'} w(v)$ , in time bounded by a polynomial in  $|V|$  and  $\log \|w\|$ . Then the same is true for the class of blockers of clutters in  $\mathcal{C}$ .

$r$ - $s$ -paths all have the  $\mathbb{Z}_+$ -MFMC-property (proved by Fulkerson [1974], Edmonds [1973], Lucchesi and Younger [1978], Ford and Fulkerson [1956], and Fulkerson [1968], respectively), and the clutters of  $T$ -joins, two-commodity cuts, two-commodity paths, and multicommodity cuts in planar graphs (with commodities on the boundary), have the  $\frac{1}{2}\mathbb{Z}_+$ -MFMC-property (proved by Edmonds and Johnson [1970], Hu [1963], Seymour [1978], and Okamura and Seymour [1979], respectively). Edmonds and Giles [1977] posed the problem whether the clutter of directed cuts has the  $\mathbb{Z}_+$ -MFMC-property.

We were not able to derive from the ellipsoid method in general a polynomial algorithm for optimum (half-)integer dual solutions, if such solutions exist. However, in the case of optimum packings of (rooted, directed,  $r$ - $s$ -,  $T$ -) cuts we can find by Theorem (3.10) an optimum fractional solution in which the number of cuts with nonzero coefficient is at most  $|V|$ . Hence, by well-known techniques (cf. Edmonds and Giles [1977], Frank [1979], Lovász [1975]) we can make these cuts laminar (i.e., non-crossing) in polynomial time, and we can find (half-)integer coefficients for the new collection of cuts, again in polynomial time, thus yielding an optimum (half-)integer packing of cuts.

We do not know whether the class  $\mathcal{C}$  of all clutters with the  $\mathbb{Q}_+$ -MFMC-property is polynomially solvable in the sense of Theorem (4.1) (in which case Theorem (4.1) would become trivial). In Chapter 6 we shall see this indeed is the case for its anti-blocking analogue.

In this chapter, as well as in the next chapters we see that the existence of polynomial algorithms can be derived from the ellipsoid method for many problems for which such algorithms have been designed before. However, we were not able to derive such an algorithm for the following two problems; for which (complicated) polynomial algorithms are known: the problem of finding a maximum independent set of vertices in a  $K_{1,3}$ -free graph (Minty [1977]), and that of finding a maximum collection of independent lines in a projective space (Lovász [1978]). A main obstacle to derive such algorithm from the ellipsoid method is that so far no characterizations in terms of facets of the corresponding convex-hull polytopes have been found. Such characterizations, previously considered as more theoretical of nature, might now be useful to derive polynomial algorithms even for the weighted versions of these problems.

## 5. SUBMODULAR FUNCTIONS AND DIRECTED GRAPHS

In this chapter we show the existence of a polynomial algorithm finding the minimum value of a submodular set function, and we derive polynomial algorithms for the optimization problems introduced by Edmonds and Giles [1977] and by Frank [1979].

Let  $X$  be a finite set, let  $F$  be a collection of subsets of  $X$  closed under union and intersection, and let  $f$  be an integer-valued submodular function defined on  $F$ , that is, let  $f: F \rightarrow \mathbb{Z}$  be such that

$$(1) \quad f(X') + f(X'') \geq f(X' \cap X'') + f(X' \cup X'')$$

for  $X', X'' \in F$ . Examples of submodular functions are the rank functions of matroids, and the function  $f$  defined on all sets  $V'$  of vertices of a capacitated digraph by:  $f(V')$  is the sum of the capacities of the arrows leaving  $V'$ . We shall give an algorithm to find  $X'$  in  $F$  minimizing  $f(X')$ , in time polynomially bounded by  $|X|$  and  $\log B$ , where  $B$  is some (previously known) upper bound for  $|f(X')|$  ( $X' \in F$ ). So as special cases we can decide in polynomial time, given a matroid  $(X, r)$  and a weight function  $w$  on  $X$ , whether  $w$  is in the corresponding matroid polytope (i.e., whether  $\sum_{x \in X'} w(x) \leq r(X')$  for each subset  $X'$  of  $X$ ), and we can derive a polynomial algorithm for finding minimum capacitated cuts in networks.

We need to make some requirements on the way  $F$  and  $f$  are given. First we should know an upper bound  $B$  for  $|f(X')|$  ( $X' \in F$ ). Secondly, we must know in advance the sets  $\cap F$  and  $\cup F$ , as well as for which pairs  $x_1, x_2$  in  $X$  there exists  $X' \in F$  with  $x_1 \in X'$  and  $x_2 \notin X'$ . This makes it possible to decide whether a given subset  $X'$  of  $X$  is in  $F$ , and it allows us to assume without loss of generality that  $\cap F = \emptyset$  and  $\cup F = X$ . Finally, given  $X'$  in  $F$  we must be able to find  $f(X')$ . It is enough to know that  $f(X')$  can be calculated in time polynomial in  $|X|$  and  $\log B$ , or that some oracle gives the answer. Most of the special-case submodular functions fulfil these requirements.

We shall reduce the minimization problem for submodular functions to the strong separation problem for polymatroid polytopes. Since the class of polymatroid polytopes is solvable (an optimization can be done by the greedy algorithm - see Edmonds [1970]), this will solve the problem.

Since we know an upper bound  $B$  for  $|f(X')|$  we can find the minimum value of  $f$  by applying binary search. So it suffices to have a polynomial algorithm finding an  $X'$  in  $F$  with  $f(X') < K$ , or deciding that no such  $X'$  exists, for any

given  $K$ . Since adding a constant to the values of  $f$  does not violate submodularity we can take  $K = 0$ . Now if  $f(\emptyset) < 0$  we can take  $X' = \emptyset$ . Hence we may assume that  $f(\emptyset) = 0$ .

Let  $g$  be the function defined on  $F$  by

$$(2) \quad g(X') = f(X') + 2B \cdot |X'|,$$

for  $X' \in F$ . So  $g$  is nonnegative, integral, monotone and submodular. Moreover,  $f(X') < 0$  if and only if  $g(X') < 2B \cdot |X'|$ . Next define for each subset  $X'$  of  $X$  the set

$$(3) \quad \overline{X'} = \cap \{X'' \in F \mid X' \subseteq X''\}.$$

(Note that  $\overline{X'}$  can be determined in polynomial time.) Let  $h(X') = g(\overline{X'})$  for each subset  $X'$  of  $X$ . One easily checks that  $h$  again is nonnegative, integral, monotone and submodular. Moreover, the problem of the existence of an  $X'$  in  $F$  with  $g(X') < 2B \cdot |X'|$  is equivalent to that of the existence of a subset  $X'$  of  $X$  with  $h(X') < 2B \cdot |X'|$ . But the latter problem is just a special case of the strong separation problem for the polymatroid polytope corresponding to  $h$ :

$$(4) \quad \{v \in \mathbb{R}_+^X \mid \sum_{x \in X'} v(x) \leq h(X') \text{ for all } X' \subseteq X\}$$

(by Theorem (3.8) the separation algorithm yields facets as separating hyperplanes, i.e., subsets  $X'$  of  $X$  violating the inequality). As the optimization problem is solvable for the class of these polytopes we are finished.

We apply the algorithm for finding the minimum value of a submodular function to theorems of Edmonds and Giles and of Frank.

Let  $D = (V, A)$  be a digraph, and let  $F$  be a collection of subsets of  $V$  such that if  $V', V'' \in F$  and  $V' \cap V'' \neq \emptyset$  and  $V' \cup V'' \neq V$  then  $V' \cap V'' \in F$  and  $V' \cup V'' \in F$ . Let  $f$  be an integer-valued function defined on  $F$  such that for all  $V', V'' \in F$  with  $V' \cap V'' \neq \emptyset$  and  $V' \cup V'' \neq V$  we have

$$(5) \quad f(V') + f(V'') \geq f(V' \cap V'') + f(V' \cup V'').$$

Denote by  $\delta^+(V')$  and  $\delta^-(V')$  the sets of arrows leaving (entering, respectively) the set  $V'$  of vertices. Let vectors  $b, c, d \in \mathbb{Z}^A$  be given, and consider the linear programming maximization problem

$$(6) \quad \text{maximize } \sum_{a \in A} c(a) \cdot x(a),$$

where  $x \in \mathbb{R}^A$  such that

$$(7) \quad \begin{aligned} (i) \quad & d(a) \leq x(a) \leq b(a) && (a \in A), \\ (ii) \quad & \sum_{a \in \delta^+(V')} x(a) - \sum_{a \in \delta^-(V')} x(a) \leq f(V') && (V' \in F). \end{aligned}$$

Edmonds and Giles showed that this problem has an integer optimum solution; this is equivalent to the fact that the polytope defined by the linear inequalities (7) has integral vertices. Edmonds and Giles also showed that the dual minimization problem can be solved with integral coefficients.

As special cases of Edmonds and Giles' result one has Ford and Fulkerson's max flow-min cut-theorem, the Lucchesi-Younger theorem on packing directed cuts and minimum coverings, Edmonds' (poly-)matroid intersection theorem, and theorems of Frank [1979] on orientations of undirected graphs. Moreover, one may derive the theorem due to Frank that if  $f$  is an integral submodular function defined on a collection  $F$  and  $g$  is an integral supermodular function on  $F$  (i.e.,  $-g$  is submodular) such that  $g(X') \leq f(X')$  for all  $X'$  in  $F$ , then there exists an integral modular function  $h$  on  $F$  (i.e., both sub- and super-modular) such that  $g(X') \leq h(X') \leq f(X')$  for all  $X'$  in  $F$ .

We shall give an algorithm which solves the maximization problem (6) in time polynomially bounded by  $|V|$  and  $\log B$ , where  $B$  is some (previously known) upper bound on  $|f(X')|$  ( $X' \in F$ ),  $\|b\|$ ,  $\|c\|$  and  $\|d\|$ . We must know in advance for each pair of vertices  $v_1, v_2$  of  $D$  whether  $v_1 \in V'$  and  $v_2 \notin V'$  for some  $V' \in F$  (this makes it possible to decide whether  $V' \in F$ ). Moreover we must have a subroutine calculating  $f(V')$  if  $V' \in F$ , in time polynomially bounded by  $|V|$  and  $\log B$ .

First of all, we may suppose that  $d(a) < b(a)$  for each arrow  $a$ , since if  $d(a) > b(a)$  the polytope (7) is empty, and if  $d(a) = b(a)$  we can remove the arrow  $a$  from the digraph and replace  $f(V')$  by  $f(V') \pm d(a)$  if  $a \in \delta^{\pm}(V')$ . We may even assume that the polytope (7) is full-dimensional and that we know an interior point  $x$  whose components have numerators and denominators not larger than a polynomial in  $|V|$  and  $B$ . Otherwise we can extend the digraph  $D$  with one new vertex  $v_0$  and with new arrows  $(v_0, v)$  for each "old" vertex  $v$  of  $D$ . Define  $d(a) = -4B$ ,  $b(a) = 0$  and  $c(a) = 2nB^2$  for the new arrows  $a$ . One easily checks that the corresponding new polytope is full-dimensional, and one easily finds an  $x$  as required. Moreover, the solutions of the original optimization problem correspond exactly to those solutions  $x$  of the new problem with  $x(a) = 0$  for each new arrow  $a$ .

Assuming the polytope (7) to be full-dimensional, by Theorem (3.8) it is

enough to show that the strong separation problem is solvable. Let  $x \in \mathbb{R}^A$ . One easily checks in polynomial time whether condition (i) is fulfilled. In case of violation we find a separating hyperplane. To check condition (ii) it suffices to find a set  $V'$  in  $F$  minimizing

$$(8) \quad g(V') := f(V') - \sum_{a \in \delta^+(V')} x(a) + \sum_{a \in \delta^-(V')} x(a)$$

in time polynomial in  $\log B$  and  $\log T$ , where  $T$  is the maximum of the numerators and denominators occurring in  $x$ . Note that  $g$  is submodular, hence we can appeal to the algorithm finding the minimum value of a submodular function. To this end we have to multiply the values by a factor to make the function integral (this factor is bounded above by  $T^{|V|}$ ), and we have to apply the algorithm for each  $v_1, v_2$  in  $V$  with  $v_1 \neq v_2$ , to the function restricted to  $\{V' \in F \mid v_1 \in V', v_2 \notin V'\}$ , since this collection is closed under union and intersection. Note that these requisites do not affect the polynomial boundedness of the required time.

So we proved that the class of "Edmonds-Giles" polytopes is solvable. Hence, by Theorem (3.10) we can find an optimum solution for the dual linear programming problem. In general, this solution will be fractional, but one can make this solution integral by making the collection of sets in  $F$  with non-zero dual coefficient laminar, by the well-known techniques (see Edmonds and Giles [1977]), in polynomial time. Now the (possibly fractional) coefficients can be replaced by integer coefficients, and these coefficients can be found by solving a linear program of polynomial size.

We leave it to the reader to derive by similar methods a polynomial algorithm for finding the solution to the following optimization problem, designed by Frank [1979]. Let  $D = (V, A)$  be a digraph, and let  $F$  be a collection of subsets of  $V$  such that if  $V', V'' \in F$  and  $V' \cap V'' \neq \emptyset$  then  $V' \cap V'' \in F$  and  $V' \cup V'' \in F$ . Let  $f$  be a nonnegative integral function defined on  $F$  such that

$$(9) \quad f(V') + f(V'') \leq f(V' \cap V'') + f(V' \cup V'')$$

if  $V', V'' \in F$  and  $V' \cap V'' \neq \emptyset$ . Let  $b, c, d \in \mathbb{Z}_+^A$ . Consider the linear programming problem

$$(10) \quad \text{minimize } \sum_{a \in A} c(a) \cdot x(a),$$

where  $x \in \mathbb{R}^A$  such that



$$(11) \quad \begin{array}{ll} (i) & d(a) \leq x(a) \leq b(a) \quad (a \in A), \\ (ii) & \sum_{a \in S'(V')} x(a) \geq f(V') \quad (V' \in F). \end{array}$$

Frank showed that this problem, and its dual, has an integer solution. As special cases one may derive again Ford and Fulkerson's max flow-min cut theorem and Edmonds polymatroid intersection theorem, and also Fulkerson's theorem on minimum weighted branchings [1974].

We finally remark that the algorithm for finding a set  $V'$  in  $F$  minimizing  $f(V')$ , where  $f$  is a submodular function defined on  $F$ , can be modified to a polynomial algorithm for finding a set  $V'$  in  $F$  of odd size minimizing  $f(V')$ . This extends Padberg and Rao's algorithm [1979] to find minimum odd cuts. More generally, let  $G \subseteq F$  be such that if  $V' \in G$  and  $V'' \in F \setminus G$  then  $V' \cap V'' \in G$  or  $V' \cup V'' \in G$ . (E.g.,  $G$  is the collection of sets in  $F$  intersecting  $V_0$  in a number of elements not divisible by  $k$ , for some fixed subset  $V_0$  of  $V$  and some natural number  $k$ .) Then there exists a polynomial algorithm to find  $V'$  in  $G$  minimizing  $f(V')$  (by this we mean:  $f(V') = \min\{f(V'') \mid V'' \in G\}$ ). This algorithm needs, besides the prerequisites for  $F$  and  $f$  as above, a polynomial subroutine deciding whether a given set  $V'$  is in  $G$ . Without loss of generality we may assume that  $\emptyset \notin G$  and  $V \notin G$ .

The algorithm is defined by induction on  $|V|$ . Suppose the algorithm has been defined for all such structures with smaller  $|V|$ . Find a set  $V'$  in  $F$  such that  $\emptyset \neq V' \neq V$  which minimizes  $f(V')$ . This can be done by applying the polynomial algorithm described above to the function  $f$  restricted to the collection  $\{V' \in F \mid v_1 \in V', v_2 \notin V'\}$ , for all  $v_1, v_2$  in  $V$ . If  $V' \in G$  we are finished. If  $V' \notin G$  there will be a set  $V''$  in  $G$  minimizing  $f(V'')$  such that  $V'' \subseteq V'$  or  $V' \subseteq V''$ . Indeed, if  $V'' \in G$  minimizes  $f(V'')$  then either  $V' \cap V'' \in G$  or  $V' \cup V'' \in G$ ; in the former case we have

$$(12) \quad \begin{array}{l} f(V' \cap V'') + f(V' \cup V'') \leq f(V') + f(V''), \\ f(V' \cap V'') \geq f(V''), \\ f(V' \cup V'') \geq f(V'), \end{array}$$

as  $V'$  and  $V''$  minimize  $f(V')$  and  $f(V'')$  for  $V' \in F$  and  $V'' \in G$ , respectively. Hence  $f(V' \cap V'') = f(V'')$ . If  $V' \cup V'' \in G$  we can exchange  $u$  and  $n$  in this reasoning. Now there exists an algorithm finding  $V'' \in G$  with  $V'' \subseteq V'$  minimizing  $f(V'')$ , and an algorithm finding  $V'' \in G$  with  $V'' \supseteq V'$  minimizing  $f(V'')$ , for these algorithms follow straightforwardly from the previously defined algorithms for sets of size  $|V'|$  and  $|V \setminus V'|$ . We leave it to the reader to check that this gives us a polynomial algorithm.

6. INDEPENDENT SETS IN PERFECT GRAPHS

In the previous chapters we have applied the ellipsoid method to classes of polytopes. We now apply the method to a class of non-polytopal convex sets, in order to obtain a polynomial algorithm finding maximum (weighted) independent sets and minimum colourings for perfect graphs (cf. Lovász [1972]).

Let  $G = (V, E)$  be an undirected graph, and let  $\alpha(G)$  denote the independence number of  $G$ , i.e., the maximum number of pairwise non-adjacent vertices. Let  $\alpha^*(G)$  denote the fractional independence number of  $G$ , i.e., the maximum value of  $\sum_{v \in V} c(v)$  where the  $c(v)$  are nonnegative real numbers such that  $\sum_{v \in C} c(v) \leq 1$  for each clique  $C$  of  $G$ . So  $\alpha(G) \leq \alpha^*(G)$ , and furthermore  $G$  is perfect if and only if  $\alpha(G') = \alpha^*(G')$  for each induced subgraph  $G'$  of  $G$ . Since  $\alpha^*(G)$  is the optimum of a linear programming problem, we could try to calculate  $\alpha^*(G)$  by means of the ellipsoid method; but the size of this problem is not polynomially bounded as there can exist too many cliques  $C$ .

However, the following number  $\theta(G)$  was introduced in Lovász [1979]. Suppose  $V = \{1, \dots, n\}$ . Then  $\theta(G)$  is the maximum value of  $\sum_{i,j=1}^n b_{ij}$ , where  $B = (b_{ij})$  belongs to the following convex body of matrices

$$(1) \quad (B = (b_{ij}) \mid B \text{ is positive semi-definite with trace } 1, \text{ and } b_{ij} = 0 \text{ if } i \text{ and } j \text{ are adjacent vertices of } G (i \neq j)).$$

If  $B$  belongs to this class we shall say that  $B$  represents  $G$ . It was shown that  $\alpha(G) \leq \theta(G) \leq \alpha^*(G)$ . (In fact,  $\theta(G)$  is an upper bound for the Shannon capacity of  $G$ .) We show that  $\theta(G)$  can be calculated (approximated) by the ellipsoid method in time bounded by a polynomial in  $|V|$ . This allows us to find  $\theta(G)$  for graphs  $G$  with  $\alpha(G) = \theta(G)$ , in particular for perfect graphs.

In fact we exhibit an algorithm finding the maximum weight  $\sum_{v \in A} w(v)$ , where  $A$  is an independent set in a perfect graph, given some nonnegative integral weight function on  $V$ , in time polynomially bounded by  $|V|$  and  $\log \|w\|$ . Obviously, this maximum weight is equal to  $\alpha(G_w)$ , where the graph  $G_w$  arises from  $G$  by replacing each vertex  $v$  of  $G$  by  $w(v)$  pairwise non-adjacent new vertices and where two vertices of  $G_w$  are adjacent iff their originals in  $G$  are adjacent. Note that if  $G$  is perfect then also any  $G_w$  is perfect (cf. Lovász [1972]). Moreover,  $\theta(G_w)$  is equal to the maximum value of

$$(2) \quad \sum_{i,j=1}^n \sqrt{w_i w_j} \cdot b_{ij}$$

where  $B = (b_{ij})$  represents  $G$ . This can be seen as follows. If  $B = (b_{ij})$  represents  $G$  then, by replacing each entry  $b_{ij}$  by a matrix of size  $w_i \times w_j$  with constant entries  $(w_i w_j)^{-1} b_{ij}$ , we obtain a matrix  $B'$  representing  $G_w$ , with

$$(3) \quad \sum_{i,j=1}^n \sqrt{w_i w_j} \cdot b_{ij} = \sum_{i,j=1}^n b'_{ij}.$$

Conversely, if  $B'$  represents  $G_w$ , then, by replacing the  $w_i \times w_j$  submatrix induced by the copies of  $i$  and  $j$ , by the sum of its entries divided by  $\sqrt{w_i w_j}$ , we obtain a matrix  $B$  representing  $G$ , satisfying (3) again.

To approximate  $\phi(G_w)$  up to an error of at most  $\epsilon > 0$ , we can replace  $\sqrt{w_i w_j}$  of (2) by some rational number  $w_{ij}$  with

$$(4) \quad |w_{ij} - \sqrt{w_i w_j}| < \epsilon/2n^2$$

(taking  $w_{ij} = w_{ji}$ ), where the denominators of the  $w_{ij}$  are at most  $2n^2/\epsilon$ . Then  $\phi(G_w)$  differs by at most  $4\epsilon$  from the maximum value of  $\sum_{i,j} w_{ij} \cdot b_{ij}$  with  $B = (b_{ij})$  representing  $G$ . So we need to approximate this last number with accuracy  $4\epsilon$ , which can be done by the ellipsoid method.

To apply the ellipsoid method we replace the set (1) by a full-dimensional convex body, by forgetting the coordinates below the main diagonal, as well as the coordinates  $(i,j)$  for adjacent  $i$  and  $j$ , and one diagonal coordinate. We end up with a full-dimensional convex body in the  $(n + \binom{n}{2} - |E| - 1)$ -dimensional space. One easily finds an interior point  $a_0$  in it, and radii  $r$  and  $R$  such that the convex body contains  $S(a_0, r)$  and is contained in  $S(a_0, R)$ , and such that the logarithms of  $r$  and  $R$  and of the numerators and denominators occurring in  $a_0$  are bounded (in absolute value) by a polynomial in  $n$  (fixed over all graphs  $G$ ). So we may apply Theorem (3.1). We show that the separation problem is solvable for the class of convex bodies obtained in this way. Let  $b$  be some vector in the  $(n + \binom{n}{2} - |E| - 1)$ -dimensional space. Extend this vector, in the obvious way, to a symmetric  $n \times n$ -matrix  $B = (b_{ij})$  with trace 1 and with  $b_{ij} = 0$  if  $i$  and  $j$  are adjacent vertices of  $G$ . Find a principal minor  $B'$  of  $B$  such that  $\text{rank}(B') = \text{rank}(B)$  and  $B'$  is nonsingular (this is easy by Gaussian elimination). Without loss of generality assume that  $B' = (b_{ij})_{i,j=1}^k$ . The matrix  $B$  is positive semi-definite iff

$$(5) \quad \det B_\epsilon = \det (b_{ij})_{i,j=1}^{\epsilon} \geq 0,$$

for  $\epsilon = 0, 1, \dots, k$ . Since these determinants can be calculated in polynomial time, thereby we have checked in polynomial time whether  $B$  belongs to the con-

vertex set (1). If, moreover, we find that B is not positive semidefinite then let t be the smallest index for which  $\det B_t < 0$ . Let  $\phi_i$  denote  $(-1)^i$  times the  $(i,t)$ -th minor of  $B_t$  ( $i = 1, \dots, t$ ), and  $\phi_i = 0$  if  $i > t$ . Then

$$(6) \quad \sum_{i,j=1}^n \phi_i \phi_j B_{ij} \geq 0$$

for every positive semidefinite matrix  $(B_{ij})$ . By definition, and by simple computation,

$$(7) \quad \sum_{i=1}^n \sum_{j=1}^n \phi_i \phi_j b_{ij} = \det B_k \cdot \det B_{k-1} \leq 0.$$

So the matrix  $(\phi_i \phi_j)_{i,j=1}^n$  is a solution of the separation problem.

Therefore, by Theorem (3.1), we can approximate the maximum value of  $\sum w_{ij} b_{ij}$  for  $B = (b_{ij})$  in (1) with accuracy  $\frac{1}{2}\epsilon$  (and hence  $\vartheta(G_w)$  with accuracy  $\epsilon$ ), in time polynomially bounded by  $|V|$ ,  $|\log \epsilon|$  and  $\log T$  where T is the maximum among the denominators and numerators occurring in  $(w_{ij})$ . If we know that  $\alpha(G_w) = \vartheta(G_w)$  it follows that  $\vartheta(G_w)$  is an integer, and we can take  $\epsilon = \frac{1}{2}$ . In particular there exists an algorithm which calculates  $\alpha(G_w)$  for perfect graphs G in time polynomially bounded by  $|V|$  and  $\log \|w\|$ .

We can find an explicit maximum weighted independent set in a perfect graph as follows. Compare  $\alpha(G_w)$  with  $\alpha(G'_w)$ , where  $G'$  and  $w'$  arise from G and w by removing vertex 1 from G and the corresponding component from w. If  $\alpha(G'_w) = \alpha(G_w)$  we replace G by  $G'$  and w by  $w'$ ; otherwise we leave G and w unchanged. Next we try to remove vertex 2 similarly, and so on. At the end we are left with a collection of vertices forming a maximum weighted independent set in G.

So given a perfect graph  $G = (V,E)$  and a weight function w on V we can find an independent set  $V'$  maximizing  $\sum_{v \in V'} w(v)$ . This implies that the strong optimization problem is solvable for the class of convex hulls of the independent sets in perfect graphs. For perfect graphs  $G = (V,E)$  this convex hull is given by the linear inequalities

$$(8) \quad \begin{array}{ll} (i) & x(v) \geq 0 & (v \in V), \\ (ii) & \sum_{v \in C} x(v) \leq 1 & (C \text{ clique}). \end{array}$$

This yields that also the strong separation problem is solvable for this class, but this is not interesting anymore, as it amounts to finding a maximum weighted clique in a perfect graph, i.e., a maximum weighted independent set in the complementary graph, which is perfect again.

However, by Theorem (3.9) we can find an optimal (fractional) dual solution for the corresponding linear programming problem. So, given  $w \in \mathbb{Z}_+^V$ , we can find cliques  $C_1, \dots, C_t$  and positive real numbers  $\lambda_1, \dots, \lambda_t$  ( $t \leq |V|$ ) such that  $\lambda_1 + \dots + \lambda_t = \alpha(G_w)$  and

$$(9) \quad \sum_{i \in C_j} \lambda_j = w_i$$

for each vertex  $i$ , in polynomial time. But for perfect graphs  $\alpha(G)$  is equal to the minimum number of cliques needed to cover  $V$  (i.e., to the chromatic number of the complementary graph), which means that there exist integers  $\lambda_1, \dots, \lambda_t$  with the required properties. Indeed we can find such integers as follows.

First, if  $w \equiv 1$ , each clique  $C_j$  with  $\lambda_j > 0$  intersects all maximum-sized independent sets. So we can remove clique  $C_1$  from  $G$ , thus obtaining a graph  $G'$  with  $\alpha(G') = \alpha(G) - 1$ , and we can repeat the procedure for  $G'$ . After  $\alpha(G)$  repetitions we have found  $\alpha(G)$  cliques covering  $V$ .

If  $w$  is arbitrary, let  $\lambda'_j$  be the lower integer part of  $\lambda_j$ , and let

$$(10) \quad w'_i = w_i - \sum_{i \in C_j} \lambda'_j.$$

Since  $(\lambda_j - \lambda'_j) < 1$  we know by (9) that  $w'_i < t \leq |V|$ . Therefore,  $G_{w'}$  has at most  $|V|^2$  vertices, and as in the previous paragraph a covering with  $\alpha(G_{w'})$  cliques can be found in time polynomially bounded by  $|V|^2$ . This covering, together with the covering by  $C_1, \dots, C_t$  with coefficients  $\lambda'_1, \dots, \lambda'_t$ , yields an optimum integral dual solution as required.

We remark that the algorithm to find  $\alpha(G)$  clearly works for all graphs  $G$  with  $\alpha(G) = \beta(G)$  but that our method to find an explicit maximum independent set requires that  $\alpha(G') = \beta(G')$  for each induced subgraph  $G'$  of  $G$ ; this is the case if and only if  $G$  is perfect, as was shown by Lovász [1979].

## 7. INTRACTABILITY OF VERTEX-PACKING AND FRACTIONAL COLOURING

The ellipsoid method yields a certain "polynomial equivalence" of combinatorial problems, in the sense that there exists a polynomial algorithm for one problem iff such an algorithm exists for some other problem. We can use this principle also in the negative: if some problem is "hard" (e.g., NP-complete) then it follows that also certain other problems are hard.

We apply this to the problem of determining the independence number  $\alpha(G)$

of a graph  $G$ , which is known to be NP-complete (cf. Garey and Johnson [1979]). More precisely, and more generally: given a graph  $G = (V, E)$ , a weight function  $w: V \rightarrow \mathbb{Z}_+$  and a number  $K$ , the problem of deciding whether there exists an independent set  $V'$  of vertices such that  $\sum_{v \in V'} w(v) \geq K$  (i.e., whether  $\alpha(G_w) \geq K$ ) is NP-complete. To formulate this in terms of polytopes, let  $P(G)$  be the convex hull of the characteristic vectors of independent sets in  $G$ . Then the strong optimization problem for the class of polytopes  $P(G)$ , is NP-complete.

Now consider the anti-blocker  $A(P(G))$  of  $P(G)$  (cf. Chapter 3). By Corollary (3.5) the strong optimization problem for the class of polytopes  $A(P(G))$  is solvable iff it is solvable for the class of polytopes  $P(G)$ . This remains to be true if we restrict  $G$  to a subclass of the class of all graphs.

Now the strong optimization problem for  $A(P(G))$  asks for a maximum weighted fractional clique, i.e., for a vector  $x$  in  $\mathbb{R}_+^V$  such that  $\sum_{v \in V'} x(v) \leq 1$  for each independent set  $V'$ , and such that  $\sum_{v \in V} w(v) \cdot x(v)$  is as large as possible, given some weight function  $w$  in  $\mathbb{Z}_+^V$ . By linear programming duality this maximum is equal to the weighted fractional chromatic number, i.e., to the minimum value  $\gamma_w^*(G)$  of  $\lambda_1 + \dots + \lambda_c$ , where  $\lambda_1, \dots, \lambda_c$  are positive numbers for which there exist independent sets  $V_1, \dots, V_c$  such that for every vertex  $v$  we have

$$(1) \quad \sum_{\substack{j \\ v \in V_j}} \lambda_j = w(v)$$

(we can take  $c \leq |V|$ ). Hence, given a class of graphs, there exists a polynomial algorithm determining  $\alpha(G_w)$  for each graph  $G$  in this class and for each weight function  $w$ , iff such an algorithm exists determining the fractional chromatic number for each such  $G$  and  $w$ . In fact, the ellipsoid method shows that both problems are "Turing reducible" to each other (cf. Garey and Johnson [1979]). This implies that, since the former problem for the class of all graphs is NP-complete, the latter problem is both NP-hard and NP-easy, i.e., NP-equivalent.

In fact the problem of determining the fractional chromatic number belongs to the class NP, as in order to show in polynomial time that  $\gamma_w^*(G) \leq K$  we can bound the numerators and denominators of the  $\lambda_j$  by  $\|w\| \cdot |V|^{|V|}$ . So the fractional colouring problem is not only Turing reducible, but even polynomial reducible to the independence number problem, but we do not know the other way around.

Since the problem of determining  $\alpha(G)$  is already NP-complete if we restrict  $G$  to planar cubic graphs the problem of determining  $\gamma_w^*(G)$  remains to be NP-equivalent if  $G$  is restricted similarly. The problem of determining the fractional chromatic number  $\gamma_w^*(G)$  and that of determining the chromatic number

$\gamma(G)$  seem to be incomparable with respect to hardness. For cubic graphs  $G$ ,  $\gamma(G)$  can be determined easily in polynomial time, but the problem of determining  $\gamma_w^*(G)$  is NP-equivalent. In contrast to this, for line graphs  $G$  of cubic graphs the problem of determining  $\gamma(G)$  is NP-complete (Holyer [1979]), whereas  $\gamma_w^*(G)$  can be determined in polynomial time (since  $\alpha(G_w)$  can be determined in polynomial time by the matching algorithm).

REFERENCES

- [1965] CHU Yoeng-jin and LIU Tseng-hung, *On the shortest arborescence of a directed graph*, *Scientia Sinica* 4 (1965) 1396-1400.
- [1959] E.W. DIJKSTRA, *A note on two problems in connexion with graphs*, *Numer. Math.* 1 (1959) 269-271.
- [1965] J. EDMONDS, *Maximum matching and a polyhedron with 0,1-vertices*, *J. Res. Nat. Bur. Standards Sect. B* 69 (1965) 125-130.
- [1967] J. EDMONDS, *Optimum branchings*, *J. Res. Nat. Bur. Standards Sect. B* 71 (1967) 233-240.
- [1970] J. EDMONDS, *Submodular functions, matroids, and certain polyhedra*, in: "Combinatorial structures and their applications" (Proc. Intern. Conf. Calgary, Alb., 1969; R. Guy, H. Hanani, N. Sauer, and J. Schönheim, eds.), Gordon and Breach, New York, 1970, pp. 69-87.
- [1973] J. EDMONDS, *Edge-disjoint branchings*, in: "Combinatorial algorithms" (Courant Comp. Sci. Symp. Monterey, Ca., 1972; R. Rustin, ed.), Acad. Press, New York, 1973, pp. 91-96.
- [1979] J. EDMONDS, *Matroid intersection*, *Annals of Discrete Math.* 4 (1979) 39-49.
- [1977] J. EDMONDS and R. GILES, *A min-max relation for submodular functions on graphs*, *Annals of Discrete Math.* 1 (1977) 185-204.
- [1973] J. EDMONDS and E.L. JOHNSON, *Matching, Euler tours and the Chinese postman*, *Math. Programming* 5 (1973) 88-124.
- [1956] L.R. FORD and D.R. FULKERSON, *Maximum flow through a network*, *Canad. J. Math.* 8 (1956) 399-404.
- [1962] L.R. FORD and D.R. FULKERSON, *Flows in networks*, Princeton Univ. Press, Princeton, N.J., 1962.
- [1977] A. FRANK, *On the orientations of graphs*, *J. Combinatorial Theory (B)*, to appear.
- [1979] A. FRANK, *Kernal systems of directed graphs*, *Acta Sci. Math. (Szeged)* 41 (1979) 63-76.
- [1979] A. FRANK, *How to make a digraph strongly connected?*, *Combinatorica*, to appear.



- [1968] D.R. FULKERSON, *Networks, frames, and blocking systems*, in: "Mathematics of the decision sciences" Part I (G.B. Dantzig and A.F. Veinott, eds.), Amer. Math. Soc., Providence, R.I., 1968, pp. 303-334.
- [1970] D.R. FULKERSON, *Blocking polyhedra*, in: "Graph theory and its applications" (Proc. adv. Seminar Madison, Wis., 1969; B. Harris, ed.), Acad. Press, New York, 1970, pp. 93-112.
- [1974] D.R. FULKERSON, *Packing rooted directed cuts in a weighted directed graph*, *Math. Programming* 6 (1974) 1-13.
- [1979] P. GÁCS and L. LOVÁSZ, *Khachian's algorithm for linear programming*, *Math. Programming Studies*, to appear.
- [1979] M.R. GAREY and D.S. JOHNSON, *Computers and intractability: a guide to the theory of NP-completeness*, Freeman, San Francisco, 1979.
- [1960] A.J. HOFFMAN, *Some recent applications of the theory of linear inequalities to extremal combinatorial analysis*, in: "Combinatorial analysis" (Proc. 10th Symp. on Appl. Math. Columbia Univ., 1958; R.E. Bellman and M. Hall, jr, eds.), Amer. Math. Soc., Providence, R.I., 1960, pp. 113-127.
- [1979] I. HOLYER, to appear.
- [1963] T.C. HU, *Multicommodity network flows*, *Operations Res.* 11 (1963) 344-360.
- [1973] T.C. HU, *Two-commodity cut-packing problem*, *Discrete Math.* 4 (1973) 108-109.
- [1979] A.V. KARZANOV, *On the minimal number of arcs of a digraph meeting all its directed cutsets*, to appear.
- [1979] L.G. KHACHIAN, *A polynomial algorithm in linear programming*, *Doklady Akademii Nauk SSSR* 244 (1979) 1093-1096 (English translation: *Soviet Math. Dokl.* 20 (1979) 191-194).
- [1970] E.L. LAWLER, *Optimal matroid intersections*, in: "Combinatorial structures and their applications" (Proc. Intern. Conf. Calgary, Alb., 1969; R. Guy, H. Hanani, N. Sauer, and J. Schönheim, eds.), Gordon and Breach, New York, 1970, pp. 233-235.
- [1976] E.L. LAWLER, *Combinatorial optimization: networks and matroids*, Holt, Rinehart and Winston, New York, 1976.

- [1972] L. LOVÁSZ, *Normal hypergraphs and the perfect graph conjecture*, Discrete Math. 2 (1972) 253-267.
- [1975] L. LOVÁSZ, *2-Matchings and 2-coverings of hypergraphs*, Acta Math. Acad. Sci. Hungar. 26 (1975) 433-444.
- [1978] L. LOVÁSZ, *The matroid parity problem*, Proc. Conf. Algebraic Methods in Graph Theory (Szeged, 1978), to appear.
- [1979] L. LOVÁSZ, *On the Shannon capacity of a graph*, IEEE Trans. on Information Theory 25 (1979) 1-7.
- [1979] L. LOVÁSZ, to appear.
- [1976] C.L. LUCCHESI, *A minimax equality for directed graphs*, Doctoral Thesis, Univ. Waterloo, Waterloo, Ont., 1976.
- [1978] C.L. LUCCHESI and D.H. YOUNGER, *A minimax relation for directed graphs*, J. London Math. Soc. (2) 17 (1978) 369-374.
- [1977] G.J. MINTY, *On maximal independent sets of vertices in claw-free graphs*, J. Combinatorial Theory (B), to appear.
- [1954] T.S. MOTZKIN and I.J. SCHOENBERG, *The relaxation method for linear inequalities*, Canad. J. Math. 6 (1954) 393-404.
- [1979] H. OKAMURA and P.D. SEYMOUR, *Multicommodity flows in planar graphs*, J. Combinatorial Theory (B), to appear.
- [1979] M.W. PADBERG and M.R. RAO, *Minimum cut-sets and b-matchings*, to appear.
- [1977] P.D. SEYMOUR, *The matroids with the max-flow min-cut property*, J. Combinatorial Theory (B) 23 (1977) 189-222.
- [1978] P.D. SEYMOUR, *A two-commodity cut theorem*, Discrete Math. 23 (1978) 177-181.
- [1970] N.Z. SHOR, *Convergence rate of the gradient descent method with dilatation of the space*, Kibernetika 2 (1970) 80-85 (English translation: Cybernetics 6 (1970) 102-108).