Reprinted from JOURNAL OF COMBINATORIAL THEORY, Series B

Vol. 36, No. 3, June 1984
Printed in Belgium

Finding Feasible Vectors of Edmonds–Giles Polyhedra*

ANDRÁS FRANK

Research Institute for Telecommunication, Gabor A. u. 65, Budapest II, Hungary H-1026

Communicated by the Editors

Received October 31, 1981; revised August 1983

A combinatorial algorithm for finding a feasible vector of the Edmonds–Giles polyhedron is presented. The algorithm is polynomially bounded provided that an oracle is available for minimizing submodular functions. A feasibility theorem is also proved by the algorithm and, as a consequence, a good algorithm for finding an integer-valued modular function between a sub- and a supermodular function is deduced. An important idea in the algorithm is due to Schönsleben and Lawler and Martel: the shortest augmenting paths have to be chosen in a lexicographic order.

1. INTRODUCTION

In [4] Edmonds and Giles proved a general min–max relation pertaining to submodular functions on directed graphs. Their main result includes such special cases as Hoffman's circulation theorem [18], Edmonds' matroid and polymatroid intersection theorem [6], and the Lucchesi–Younger theorem [20].

In [9] we described an algorithm which provided a constructive proof for the Edmonds–Giles theorem. That procedure is not only finite but polynomially bounded provided that a fast oracle is available for minimizing a submodular function and that the variables x are restricted by $0 \leqslant x \leqslant 1$.

Among the specializations mentioned, this is the case in the matroid intersection problem and in the Lucchesi–Younger problem. (In fact, the main ideas of the algorithm in [9] were first developed for these special cases; see [10, 11].) Another special case of this kind serves as a good algorithm for finding a minimum cost k-strongly connected orientation of an undirected graph when the two possible orientations of an edge may have different costs [9, 21].

* Research partially supported by Sonderforschungsbereich 21 (DFG, Institut für Ökonometrie und Operations Research, Universität Bonn, West Germany. On leave from Research Institute for Telecommunication, Budapest, Hungary.

0095-8956/84 $3.00

In the general Edmonds–Giles problem, however, when x is bounded by arbitrary integer vectors f and g, i.e., $f \leqslant x \leqslant g$, the complexity of the algorithm in [9] includes a factor proportional to the maximal absolute value M of f and g. This more general case occurs in the minimum cost circulation problem and in the polymatroid intersection problem.

In [9] first an algorithm was described to find an optimal primal solution which started with any feasible solution. It then was stated that the same algorithm could be used for finding a starting feasible solution. However, as Cunningham pointed out, this statement is true only for the special case of intersecting families and not for crossing families. Furthermore, the complexity of this feasibility procedure also includes M; i.e., it is not polynomially bounded in the general $f \leqslant x \leqslant g$ case.

The main purpose of the present paper is to provide a polynomial time combinatorial algorithm for determining a feasible solution to the Edmonds–Giles problem. In order to do this we shall use the approach given in [9] that first considers the intersecting case, combined with an idea of Schönsleben [23] and of Lawler and Martel [19]. Then we prove a feasibility theorem. In Section 4 a discrete separation theorem will be derived along with an adaptation of the algorithm for this case. In Section 5 we reduce the general crossing case to the intersecting case by showing how the algorithm for intersecting families can be used, applying it twice, in the general crossing case. This two-step approach fills in the gap in [9] mentioned above. Finally we show how the Lawler–Martel model can be handled by the present method and briefly outline the relation to other models.

In [2] we described a combinatorial solution algorithm for the optimization problem over the Edmonds–Giles polyhedron. (For a noncombinatorial procedure, see [16].) That algorithm heavily relies on the present work which serves as a fundamental subroutine (like the situation in Edmonds' weighted matroid intersection algorithm [5], where the maximum cardinality matroid intersection is used extensively). See also [14].

We note that a fast oracle for minimizing submodular functions is assumed to be available and, in calculating the complexity of an algorithm, we consider the addition, subtraction, and comparison of two real numbers as one step each.

2. Preliminaries

Throughout the paper we work with a finite groundset V of n elements. If $A \subseteq V$, the complement of A is denoted by $\bar{A}$. Sets $A, B \subseteq V$ are *co-disjoint* if $\bar{A}$ and $\bar{B}$ are disjoint. Sets $A, B \subseteq V$ are *intersecting* if none of $A \cap B$, $A - B$, $B - A$ is empty. If, in addition, $A \cup B \neq V$ then A and B are *crossing*. A family $\mathscr{B}$ of subsets of V is *intersecting* (*crossing*) if $A \cap B, A \cup B \in \mathscr{B}$ for all intersecting (crossing) members of $\mathscr{B}$. $\mathscr{B}$ is called a *ring family* if it is closed under taking any union and intersection. A set function b is *submodular* on A, B if $b(A) + b(B) \geqslant b(A \cap B) + b(A \cup B)$. A function p is *supermodular* if $-p$ is submodular. Sometimes we call a pair $(b, \mathscr{B})$ a *crossing* (*intersecting*) submodular function if $\mathscr{B}$ is a crossing (intersecting) family, b is defined on $\mathscr{B}$ and submodular on crossing (intersecting) sets.

A set A is called a $u\bar{v}$-set if $u \in A$, $v \notin A$. For a vector $z \in R^V$ and for $B \subseteq V$ set $z(B) = \sum (z(v): v \in B)$. Obviously z is modular (i.e., sub- and supermodular) and every modular set function with $z(\varnothing) = 0$ comes in this way. We do not distinguish between a modular function z with $z(\varnothing) = 0$ and a vector $z \in R^V$. Let $G = (V, E)$ be a directed graph with n nodes and m arrows. (We use the term "arrow" rather than directed edge.) Multiple arrows and loops are excluded. An arrow uv *enters* (*leaves*) $B \subset V$ if B is a $v\bar{u}$-set ($u\bar{v}$-set). For a vector $x \in R^E$ denote $\rho_x(B) = \sum (x(e): e \text{ enters } B)$ and $\delta_x(B) = \rho_x(\bar{B})$. For a singleton we use $\rho_x(v)$ instead of $\rho_x(\{v\})$.

Let $(b', \mathscr{B}')$ be a real-valued crossing submodular function. Let A' be a $(0, \pm 1)$ matrix the rows of which correspond to the members of $\mathscr{B}'$, the columns of which correspond to the elements of E and

$$a'_{B,e} = \begin{cases} -1 & \text{if } e \text{ leaves } B, \\ +1 & \text{if } e \text{ enters } B, \\ 0 & \text{otherwise.} \end{cases}$$

Denote by a'_e the column vector of A' corresponding to $e \in E$. Without loss of generality we can assume that $\varnothing, V \notin \mathscr{B}'$. Let $f, g \in R^E$ be two real vectors with $f \leqslant g$ (f, g may include infinite components).

The theorem of Edmonds and Giles states that the linear system

$$A'x \leqslant b', \qquad f \leqslant x \leqslant g, \tag{1}$$

is totally dual integral. For total dual integrality, see also [22, 24].

First we shall be dealing with the special case of intersecting submodular functions. In Section 5 we show how the algorithm developed for intersecting submodular functions extends to the general crossing case.

3. Intersecting Submodular Functions

Instead of $(b', \mathscr{B}')$, let $(b, \mathscr{B})$ be an intersecting submodular function and assume that $V \in \mathscr{B}$ and $b(V) = 0$. Let the matrix A be defined in the same way as A' was.

THEOREM 1. *The linear system*

$$Ax \leqslant b, \qquad f \leqslant x \leqslant g, \tag{2}$$

has a solution if and only if

$$\rho_f(\bigcup B_i) - \delta_g(\bigcup B_i) \leqslant \sum b(B_i) \tag{3}$$

whenever $B_1,\dots,B_k$ are disjoint members of $\mathscr{B}$. Moreover, (2) *has an integral solution if b, f, g are integral and* (3) *holds.*

If (2) has a solution, one may wish to determine the maximum value of $x(t_0 s_0)$, where $t_0 s_0$ is a specified arrow. This is equivalent to asking the maximum possible lower bound $f(t_0 s_0)$ for which (3) holds. (It is convenient to suppose that $g(t_0 s_0) = +\infty$.) Thus Theorem 1 implies the following result where the arrow $t_0 s_0$ is not counted in ρ_f and δ_g.

THEOREM 1A. $\max(x(t_0 s_0): x$ *is a solution to* (2)) $= \min(\sum b(B_i) - \rho_f(\bigcup B_i) + \delta_g(\bigcup B_i): B_1,\dots,B_k \in \mathscr{B}, B_i \cap B_j = \varnothing$ *for* $1 \leqslant i < j < k$, *and* $\bigcup B_i$ *is an* $s_0\bar{t}_0$*-set*).

Notice that the minimum is $+\infty$ if, for any disjoint members B_i of $\mathscr{B}$ such that $\bigcup B_i$ is an $s_0\bar{t}_0$-set , there is an arrow uv either entering $\bigcup B_i$ with $f(uv) = -\infty$ or leaving $\bigcup B_i$ with $g(uv) = +\infty$.

Proof of Theorem 1. *Necessity.* If x is a solution to (2) then we have

$$\sum b(B_i) \leqslant \sum (\rho_x(B_i) - \delta_x(B_i)) = \rho_x(\bigcup B_i) - \delta_x(\bigcup B_i)$$
$$\geqslant \rho_f(\bigcup B_i) - \delta_g(\bigcup B_i).$$

Sufficiency. Adjoin two new nodes t, s to G and for each $v \in V$ construct new arrows e from v to s and from t to v and set $f(e) = 0$, $g(e) = +\infty$. Henceforth we consider the linear system (2) with respect to this graph G_1. Obviously it has a solution by taking, for example, $x(e)$ to be anything with $f \leqslant x \leqslant g$ on old arrows and $x(tv) = 0$, $x(vs) = M$ for each $v \in V$, where M is a large enough number (for example, $M = \max(\rho_x(X) - \delta_x(X) - b(X): X \in \mathscr{B})$ will do). Our purpose is to find another solution in which $x(vs) = 0$ for each $v \in V$. We note that one may use other starting solutions. Actually, in the general case of crossing submodular functions we shall reduce the problem to an intersecting problem and then another starting solution will be needed. In the present case, however, the new arrows tv do not play any role and they can be omitted.

Let $\lambda_x(B) = \rho_x^1(B) - \delta_x^1(B)$, where ρ_x^1 and δ_x^1 concern G_1. Note that λ_x is a modular function and $Ax \leqslant b$ is equivalent to the fact that $\lambda_x(B) \leqslant b(B)$ for each $B \in \mathscr{B}$. A member B of $\mathscr{B}$ is called *b-tight* (with respect to x) or briefly *tight* if $\lambda_x(B) = b(B)$. (In [9] a somewhat confusing term "strict" was used for tight.)

The following two simple lemmas were proved in [9].

LEMMA 1. *If K and L are intersecting tight members of $\mathscr{B}$ then $K \cap L$ and $K \cup L$ are also tight.*

If $v \in V$ is in some tight set, denote by $B(v)$ the intersection of all tight sets containing v. ($B(v)$ depends on x.)

LEMMA 2. (a) *$B(v)$ is tight.*

(b) *If a family of tight sets forms a connected hypergraph, the union is again tight.*

Let us define an auxiliary digraph H_x in which three kinds of arrows may exist (so H_x may contain multiple arrows):

(1) $e_1 = uv$ is an arrow (called *forward*) if $x(uv) < g(uv)$. Its capacity is defined by $c(e_1) = g(uv) - x(uv)$.

(2) $e_2 = vu$ is an arrow (called *backward*) if $x(uv) > f(uv)$. Its capacity is defined by $c(e_2) = x(uv) - f(uv)$.

(3) $e_3 = uv$ $(u \neq s)$ is an arrow (called *jumping*) if there is no tight $u\bar{v}$-set with respect to x. The capacity of e_3 is defined by $c(e_3) = \min(b(B) - \lambda_x(B): B \in \mathscr{B}, B$ is a $u\bar{v}$-set).

In particular, if $x(us) > 0$ for a new arrow us, then su will be in H_x. Or, if $v \in V$ is not in any tight set, then vt will be in H_x. Try to find a path in H_x from s to t. There may be two cases.

Case 1. There is no path from s to t.

Let S be the subset of vertices of V reachable from s. If $S = \varnothing$ then $x(us) = 0$ for each $u \in V$ therefore the restriction of the current solution x to the arrows of G is a solution to (2).

If $S \neq \varnothing$ then any vertex v in S is in a tight set, namely, in $B(v)$, included in S, that is, S is the union of tight sets. From Lemma 2b, S partitions into disjoint tight sets $B_1, B_2,\dots, B_k$. Moreover, $x(uv) = g(uv)$ if an arrow uv in G leaves S and $x(uv) = f(uv)$ if uv enters S. Thus we have

$$\sum b(B_i) = \sum (\rho_x^1(B_i) - \delta_x^1(B_i)) = \rho_f(S) - \delta_g(S) - \sum (x(us): u \in V).$$

This means that $\sum b(B_i) < \rho_f(\bigcup B_i) - \delta_g(\bigcup B_i)$, contradicting (3).

Observe that if the sets $B(v)$ are available, then the sets B_i can be quickly computed since they are the components of the hypergraph $\mathscr{H} = \{B(v): v \in S\}$.

Case 2. In H_x there exists a path from s to t.

Let P be a shortest path from s to t (with respect to the number of arrows). Denote by Δ the minimum capacity of the arrows on P. Δ is called the capacity of P. We see that $\Delta > 0$. Define a new vector x':

$$\begin{aligned} x'(uv) &= x(uv) + \Delta && \text{if } uv \in E \text{ is on } P, \\ &= x(uv) - \Delta && \text{if } (uv \in E \text{ or } v = s) \text{ and } vu \text{ is on } P, \\ &= x(uv) && \text{otherwise.} \end{aligned}$$

We call this change an *augmentation.*

LEMMA 3. *For any member B of $\mathscr{B}$, $\lambda_{x'}(B) = \lambda_x(B) + \Delta \cdot (\delta^j(B) - \rho^j(B))$, where $\rho^j(B)(\delta^j(B))$ stands for the number of jumping arrows of P entering (leaving) B.*

Proof. This is quite easy when $\rho^j(B) = \delta^j(B) = 0$ and, in general, follows by a simple induction on $\rho^j(B) + \delta^j(B)$. ∎

LEMMA 4. *x' is a solution to* (2) (*with respect to G_1*).

Proof. Obviously we have $f \leqslant x' \leqslant g$. Let $\varepsilon(B) = b(B) - \lambda_x(B)$ for $B \in \mathscr{B}$. We are going to prove that $\delta^j(B) \cdot \Delta \leqslant \varepsilon(B)$ for each $B \in \mathscr{B}$. By Lemma 3 this implies that $\lambda_{x'}(B) \leqslant b(B)$ for $B \in \mathscr{B}$, i.e., $Ax' \leqslant b$. We proceed by induction on the value $\delta^j(B)$. The case $\delta^j(B) = 0$ is trivial. Let $\delta^j(B) > 0$ and let uv be the first jumping arrow on P (starting at s) which leaves B. If $v = t$ then there is no other such jumping arrow, i.e., $\delta^j(B) = 1$ and $c(ut) \geqslant \Delta$. On the other hand $c(ut) \leqslant b(B) - \lambda_x(B) = \varepsilon(B)$ and thus $\delta^j(B) \cdot \Delta \leqslant \varepsilon(B)$.

Claim. If $v \neq t$ then $\delta^j(B \cup B(u)) = \delta^j(B) - 1$.

Proof. Since no jumping arrow leaves $B(u)$ and uv does not leave $B \cup B(u)$ we have $\delta^j(B \cup B(u)) \leqslant \delta^j(B) - 1$. On the other hand if qr is another jumping arrow on P leaving B then we claim that $r \notin B(u)$ (that is, qr leaves $B \cup B(u)$, too); in the contrary case ur would be a shortcut arrow to P, contradicting the minimality of P.

Now we have $\varepsilon(B) = \varepsilon(B) + \varepsilon(B(u)) \geqslant \varepsilon(B \cap B(u)) + \varepsilon(B \cup B(u) \geqslant \Delta + \Delta \cdot (\delta^j(B) - 1) = \Delta\delta^j(B)$, as required. Here we made use of the induction hypothesis for $B \cup B(u)$ and the previous claim. ∎

The basic idea behind the algorithm is the same as in the classical Ford–Fulkerson maximum flow algorithm [8]. Again build up the new auxiliary digraph with respect to the new solution x' to (2) and repeat the procedure until Case 1 occurs. We have to prove that the number of subsequent

augmentations can be bounded by a polynomial in $|V|$. To this end, among the various shortest augmenting paths in a given stage, we break ties by a lexicographic ordering. (This means that the run of the algorithm will be completely determined.) Assume that the vertices of G_1 have fixed (different) indices. For notational convenience we do not distinguish between the name and the index of a vertex. That is, for two vertices u, v, $u > v$ means that the index of u is greater than that of v.

For an intermediate solution x denote by $\sigma_x(u)$ ($\tau_x(u)$) the length of the least length path from s to u (from u to t) in the auxiliary digraph H_x. Call an arrow uv in H_x *admissible* if $\sigma_x(u) + \tau_x(v) + 1 = \sigma_x(t)$. Obviously a shortest path from s to t can consist of admissible arrows only. Let us define $\pi_x(v)$ as the minimum index u for which uv is admissible. If no such u exists then $\pi_x(v) = \infty$. The vertices of the augmenting path we will use are t, $\pi(t)$, $\pi(\pi(t), \ldots, s$. Obviously none of these indices is ∞. (These vertices still do not determine uniquely the augmenting path since from u to v may lead three parallel arrows in H_x. It does not matter which one is chosen, but the one of maximum capacity seems to be the most natural.) Henceforth by an augmentation path we mean a path defined this way. Note that a simple modification of the well-known labelling technique finds this path.

The idea of using least augmenting paths is due to Dinitz [3] and to Edmonds and Karp [7] and helps to solve the maximum flow problem efficiently. The idea of lexicographic tiebreaking was suggested by Schönsleben [23] for the polymatroid intersection problem and by Lawler and Martel for the so-called polymatroidal flow problem [19]. Still another application of this idea, due to Cunningham, is a method for testing membership in a matroid polyhedron [1]. The key observation, called the "Splicing Lemma" by Lawler and Martel is as follows in the present context.

SPLICING LEMMA. *Suppose that $\sigma_x(v) > \sigma_x(u)$ and uv is a new jumping arrow in $H_{x'}$, that is, uv is a jumping arrow in $H_{x'}$ but not in H_x. There exist two consecutive nodes v_1, u_1 of P such that $v_1 v$, uu_1 are jumping arrows in H_x and $\sigma_x(u) = \sigma_x(v_1) = \sigma_x(v) - 1 = \sigma_x(u_1) - 1$.*

Proof. Recall that $B(w)$ is the minimal tight set containing w (with respect to x) if w is contained in at least one tight set; if w is not in any tight set then let $B(w) = V \cup \{t\}$. Since uv is a new jumping arrow, $v \notin B(u)$. Let X be a maximal tight $u\bar{v}$-set such that $B(u) \subseteq X$ and (i) for $w \in P \cap (X - B(u))$, $\sigma_x(w) \leqslant \sigma_x(u)$. Since X is not tight with respect to x', by Lemma 3 there is a jumping arrow $v_1 u_1$ of P entering X. By Lemma 1, $X' = B(v_1) \cup X$ is tight. Property (i) holds for X' since $\sigma_x(w) \leqslant \sigma_x(v_1) = \sigma_x(u_1) - 1 \leqslant \sigma_x(u)$ whenever $w \in P \cap B(v_1) - X$. Thus the maximal choice of X implies that X' is not a $u\bar{v}$-set, i.e., $v \in B(v_1)$. Hence $\sigma_x(u_1) = \sigma_x(v_1) + 1 \geqslant \sigma_x(v) \geqslant \sigma_x(u) + 1$ which shows by (i) that $u_1 \in B(u)$. In other words $v_1 v$

and uu_1 are jumping arrows and $\sigma_x(u)+1 \leqslant \sigma_x(v) \leqslant \sigma_x(v_1)+1=\sigma_x(u_1) \leqslant \sigma_x(u)+1$ from which equality follows everywhere. ∎

LEMMA 5. *$\sigma_x(w)$ and $\tau_x(w)$ $(w \in V)$ are nondecreasing.*

Proof. We prove the statement for $\sigma_x(w)$. If uv is a new arrow in $H_{x'}$ for which $\sigma_x(u) < \sigma_x(v)$ then uv is jumping and, by the splicing lemma, $\sigma_x(u) = \sigma_x(v)-1$. Therefore $\sigma_x(w)$ cannot decrease. ∎

By a *phase* we mean a maximal sequence of successive augmentations in which $\sigma(t)$ is unchanged. Obviously the number of phases is at most n.

LEMMA 6. *In one phase $\pi_x(v)$ does not decrease.*

Proof. The only possibility for decreasing $\pi_x(v)$ would be a new admissible jumping arrow uv after making an augmentation. Apply the splicing lemma and consider those vertices v_1, u_1 of P. Then $v_1 v$, uu_1, and $v_1 u_1$ are all admissible arrows in H_x. Thus $\pi_x(v) \leqslant v_1 = \pi_x(u_1) \leqslant u$, i.e., the new jumping arrow uv does not reduce $\pi_x(v)$. ∎

Call an arrow on the augmenting path *critical* if its capacity is Δ.

LEMMA 7. *After making an augmentation, a critical arrow disappears from the auxiliary digraph.*

Proof. The lemma is trivial if either uv is a forward or backward arrow or $u=s$. Assume that uv is a jumping arrow. We prove that there exists a $u\bar{v}$-set X tight with respect to x'. Since uv is critical, $\Delta = \min(b(B)-\lambda_x(B)$: B is a $u\bar{v}$-member of $\mathscr{B}$). Let B be a minimal $u\bar{v}$-set for which $\Delta = b(B)-\lambda_x(B)$.

Claim. $B \subseteq B(u)$.

Proof. $\Delta = \varepsilon(B) = \varepsilon(B)+\varepsilon(B(u)) \geqslant \varepsilon(B \cap B(u)) + \varepsilon(B \cup B(u)) \geqslant \Delta + 0$ from which $\varepsilon(B \cap B(u)) = \Delta$. By the minimality of B, $B = B \cap B(u)$, i.e., $B \subseteq B(u)$.

The claim shows that $X := B$ satisfies the following properties:

(a) X is a $u\bar{v}$-set,

(b) $b(X)-\lambda_x(X)=\Delta$,

(c) $w \in X \cap P \Rightarrow \sigma_x(w) \leqslant \sigma_x(u)$.

Choose a maximal set X satisfying (a)–(c).

Claim. No jumping arrow qr on P enters X.

Proof. Suppose on the contrary that such a qr exists. We are going to show that $X' = X \cup B(q)$ satisfies (a)–(c), contradicting the maximal choice of X.

(a) $\sigma_x(q)+1=\sigma_x(r) \leqslant \sigma_x(u) = \sigma_x(v)-1$; therefore, $v \notin B(q)$.

(b) Denoting $b(X)-\lambda_x(X)$ by $\varepsilon(X)$, we get $\Delta = \varepsilon(X) = \varepsilon(X) + \varepsilon(B(q)) \geqslant \varepsilon(X \cap B(q)) + \varepsilon(X') \geqslant 0 + \Delta$, whence $\varepsilon(X') = \Delta$.

(c) For $w \in (P \cap B(q)) - X$, we have $\sigma_x(w) \leqslant \sigma_x(q) = \sigma_x(r) - 1 < \sigma_x(u)$.

Using Lemmas 3 and 4 we have $b(X) \geqslant \lambda_{x'}(X) = \lambda_x(X) + \Delta(\delta^j(X) - \rho^j(X)) = b(X) - \Delta + \Delta\delta^j(X) \geqslant b(X)$ from which $b(X) = \lambda_{x'}(X)$, as required. ∎

LEMMA 8. *If uv is a critical jumping arrow on an augmenting path P_1 then uv will be no longer a jumping admissible arrow during the whole phase.*

Proof. By Lemma 7 after augmenting along P_1 the arrow uv disappears. At that time we had $\pi_{x_1}(v) = u$; thus, by Lemma 6, $\pi_x(v) \geqslant u$ during the whole phase.

Assume indirectly that later in the same phase we are making an augmentation of the current x along an augmenting path P and uv becomes again a jumping admissible arrow. Applying the splicing lemma, we see that $u \leqslant \pi_x(v) \leqslant v_1 = \pi_x(u_1) \leqslant u$ from which $u = v_1$, that is, uv was a jumping arrow already in H_x, a contradiction. ∎

By now we have proved that within one phase an arrow may be critical at most once. Since there may be three parallel arrows from u to v, the number of successive augmentations is at most $3n^2$ in one phase and thus the overall number of augmentations is at most $3n^3$. Furthermore, if the input data b, f, g are all integral then all arithmetic is integral and thus the final x is also integral.

In order to apply the algorithm we need an oracle which can

(A) minimize $b(B) - z(B)$ over the $u\bar{v}$-members of $\mathscr{B}$,

where z is an arbitrary modular function. (Note that $z = \lambda_x$ is modular.) With the help of this oracle we can compute the auxiliary digraph H_x as well as the capacities of jumping arrows in H_x. Assume this oracle is available with complexity h. One augmenting path and the new $H_{x'}$ with the capacities can be computed in n^2h steps. Therefore the overall complexity of the algorithm can be bounded by $O(n^5h)$.

ALGORITHM FOR SOLVING (2)

Input	$G=(V,E)$:	directed graph,
	$\mathscr{B}$:	intersecting family, $\varnothing \notin \mathscr{B} \subset 2^V$,
	b:	$\mathscr{B} \to \mathbb{R}$, real-valued function, submodular on intersecting pairs,
	f, g:	$E \to \mathbb{R}$, real vectors, $f \leqslant g$,
	M:	upper bound for $b(B)$, $M \geqslant 0$.
Output	**Either**	
	x:	a solution to (2), integer-valued if b, f, g are integer-valued,
or	$\{B_i\}$:	a subfamily of $\mathscr{B}$ which violates (3).

We make use of oracle (A)

Step 0. Form the digraph G_1. Let x be any vector for which $f(e) \leqslant x(e) \leqslant g(e)$ holds for arrows in E and $x(e)=M$ for new arrows $e=us$ $(u \in V)$.

Step 1.

1.1. Using oracle (A), form the auxiliary diagraph H_x and compute the capacities of its arrows.

1.2. Try to find a lexicographically minimal shortest path P from s to t in H_x by the labelling technique. If no such path exists, go to Step 2.

1.3. Compute the minimal capacity Δ of the arrows of P and update

$$x(e) := \begin{cases} x(e)+\Delta & \text{if } e \text{ is a forward arrow on } P \\ x(e)-\Delta & \text{if } x \text{ is a backward arrow on } P \\ 0 & \text{otherwise.} \end{cases}$$

1.4. Go to Step 1.

Step 2.

2.1. Denote by S the set of labelled vertices of V. If $S=\varnothing$ then the restriction of the current x to the original arrows is a solution to (2). HALT.

2.2. If $S \neq \varnothing$, form the components $B_1, B_2, \ldots, B_k$ of the hypergraph $\mathscr{H}=\{B(v): v \in S\}$; $\{B_i\}$ violates (3). HALT.

A slight modification of the algorithm enables us to get a solution x to (2) which maximizes $x(t_0 s_0)$ for a specified arrow $t_0 s_0$. To this end we start with any feasible solution found previously and define an auxiliary digraph in the same way as in Section 3 except that no extra nodes and arrows are needed.

The algorithm consists of finding augmenting paths from s_0 to t_0 in the subsequent auxiliary digraphs. The algorithm terminates by detecting either a solution x for which no augmenting path exists in the corresponding H_x or a solution x such that H_x contains an augmenting path of infinite capacity.

In the first case x is the required maximal solution and sets B_i for which we have equality in Theorem 1A can be obtained as the components of a hypergraph formed by the tight sets in the set of reachable nodes. In the second case the maximum is not finite.

4. SEPARATION THEOREM

As an application of Theorem 1, we prove the following result.

THEOREM 2. *We are given two intersecting families $\mathscr{B}$, $\mathscr{P}$ $(\varnothing \notin \mathscr{B} \cup \mathscr{P})$ and two functions $b: \mathscr{B} \to R$, $p: \mathscr{P} \to R$ which are sub- and supermodular, respectively, on intersecting sets. There exists a vector $m \in R^V$ such that $b(B) \geqslant m(B)$ for $B \in \mathscr{B}$ and $p(P) \leqslant m(P)$ for $P \in \mathscr{P}$ if and only if*

$$\sum p(P_i) \leqslant \sum b(B_j) \tag{4}$$

holds for any disjoint members P_i of $\mathscr{P}$ and B_j of $\mathscr{B}$ such that $\bigcup B_i = \bigcup P_j$. Moreover, if b and p are integer-valued, then m can be chosen to be integer-valued.

(Such an m is said to *separate* b and p.) Note that this theorem easily implies the next one which was proved in [9].

DISCRETE SEPARATION THEOREM. *Let $\mathscr{K}$ be a ring family and b and p integer-valued functions on $\mathscr{K}$ which are sub- and supermodular, respectively, on any pair of members of $\mathscr{K}$. If $p \leqslant b$ then there exists an integer-valued modular function m for which $p \leqslant m \leqslant b$.*

Proof of Theorem 2. The necessity of (4) is straightforward. The sufficiency can be derived from Theorem 1 by a simple elementary construction just as the Discrete Separation Theorem was proved in [9]. Namely, let V' be another copy of the ground set V and construct an arrow from v' to v for each $v \in V$. Let $\mathscr{P}'=\{P': P \in \mathscr{P}\}$ and $\mathscr{F}=\mathscr{P}' \cup \mathscr{B}$. Set $f(B)=b(B)$ for $B \in \mathscr{B}$ and $f(P')=-p(P)$ for $P \in \mathscr{P}$. Then $(f, \mathscr{F})$ is an intersecting submodular function and there is a 1–1 correspondence between the feasible solutions of the Edmonds–Giles problem defined by $(f, \mathscr{F})$ and the modular function m separating b and p. Thus the algorithm of Section 3 applies. Notice that the necessary oracle (A) for this particular Edmonds–Giles problem is available provided that we have an oracle for minimizing

$b(B)-m(B)$ and $m(P)-p(P)$ over $u\bar{v}$-members of $\mathcal{B}$ and $\mathcal{P}$, respectively, for any modular function m.

In fact, the algorithm can be modified to work on the original ground set V. Here we outline this adaptation of the algorithm in Section 3 (of course, without proof). We remark that this discrete separation algorithm will be used for finding an appropriate starting feasible solution to the Edmonds–Giles problem in case of crossing submodular functions. See Section 5.

ALGORITHM FOR SEPARATING SUB- AND SUPERMODULAR FUNCTIONS

Input $\mathcal{B}, \mathcal{P}$: $\subset 2^V$, intersecting families, $\varnothing \notin \mathcal{B}, \mathcal{P}$,

b: $\mathcal{B} \to \mathbb{R}$, real-valued function submodular on intersecting sets,

p: $\mathcal{P} \to \mathbb{R}$, real-valued function supermodular on intersecting sets,

M: nonnegative bound for which $-M \leqslant p$, $b \leqslant M$.

Output **Either**

m: vector in R^V separating p and b which is integer-valued if p and b are integer-valued,

or $\{P_i\}$: disjoint members of $\mathcal{P}$,

$\{B_j\}$: disjoint members of $\mathcal{B}$ such that $\bigcup P_i = \bigcup B_j$ and $\sum p(P_i) > \sum b(B_j)$.

We need an oracle for minimizing $b(B)-m(B)$ and $m(P)-p(P)$ over $u\bar{v}$-members of $\mathcal{B}$ and $\mathcal{P}$, respectively (for any modular m).

Step 0. Choose two modular functions m, d so that $d \geqslant 0$ and $p \leqslant m \leqslant b+d$ (e.g., $m(X)=M|X|$, $d(X)=2M|X|$). Set $b' := b+d$. Call a set X b'-tight if $b'(X)=m(X)$ and p-tight if $p(X)=m(X)$. Let s and t be two new vertices.

Step 1.

1.1. Using the oracle above, form an auxiliary digraph H on $V \cup \{s,t\}$ with the following arrows and capacities:

su: if $d(u)>0$. Let $c(uv)=d(uv)$,

uv: if no b'-tight $u\bar{v}$-set exists in $\mathcal{B}$. Let $c(uv)=\min(b'(B)-m(B)$: $B \in \mathcal{B}$ is a $u\bar{v}$-set). Call such an arrow *blue*.

uv: if not p-tight $u\bar{v}$-set exists in $\mathcal{P}$. Let $c(uv)=\min(m(P)-p(P)$: $P \in \mathcal{P}$ is a $u\bar{v}$-set). Call such an arrow *pink*.

1.2. Try to find a lexicographically minimal shortest path U from s to t in H. If no such path exists, go to Step 2.

1.3. Denote by Δ the minimum capacity of the arrows on U. The vertices of U are $s=v_0, v_1, v_2, \ldots, v_k, v_{k+1}=t$. Update $d(v_1) := d(v_1)-\Delta$.

1.4. The arrows on U are alternately blue and pink. Update m as follows:

$$m(v_1) := \begin{cases} m(v_1) & \text{if } v_1v_2 \text{ is blue,} \\ m(v_1)-\Delta & \text{if } v_1v_2 \text{ is pink.} \end{cases}$$

For $i \geqslant 2$ increase or decrease $m(v_i)$ by Δ according to whether v_i is entered by a pink or blue arrow on U.

Go to 1.1.

Step 2.

2.1. Denote by S the set of labelled vertices of V. If $S=\varnothing$, the current m satisfies the requirements since now $d \equiv 0$. HALT.

2.2. If $S \neq \varnothing$, the components $B_1, B_2, \ldots, B_k$ of the hypergraph $\mathcal{H}_b = \{B(v): v \in S\}$ partition S. Also, the components $P_1, P_2, \ldots, P_l$ of the hypergraph $\mathcal{H}_p = \{P(v): v \in S\}$ partition S. The sets $\{B_j\}$ and $\{P_i\}$ violate (4). HALT.

5. CROSSING SUBMODULAR FUNCTIONS

We turn to the original problem when $(b', \mathcal{B}')$ is a crossing submodular function ($\varnothing, V \notin \mathcal{B}'$). The rough idea is that there exists an intersecting submodular function $(b, \mathcal{B})$ which determines the same Edmonds–Giles polyhedron as $(b', \mathcal{B}')$ and then the algorithm in Section 3 can be applied.

Let us define $\mathcal{B} = \{X: X \neq \varnothing,\ X = \bigcap X_i,\ X_i \in \mathcal{B}',\ \bar{X}_i \cap \bar{X}_j = \varnothing\} \cup \{V\}$. Let $b(X) = \min(\sum b'(X_i): X = \bigcap X_i,\ X_i \in \mathcal{B}',\ \bar{X}_i \cap \bar{X}_j = \varnothing)$ for $X \in \mathcal{B} - \{V\}$ and $b(V)=0$.

Lemma 1 was proved in [9].

LEMMA 1. *$(b, \mathcal{B})$ is an intersecting submodular function.*

LEMMA 2. *For a vector $y \in R^V$ with $y(V)=0$,*

(i′) *$y(B) \leqslant b'(B)$ for each $B \in \mathcal{B}'$ if and only if*

(i) *$y(B) \leqslant b(B)$ for each $B \in \mathcal{B}$.*

Proof. Since $\mathcal{B}' \subset \mathcal{B}$ and $b(B) \leqslant b'(B)$ for $B \in \mathcal{B}'$ the "if" part follows. On the other hand, given a vector y satisfying (i′) and $B \in \mathcal{B}$, $B \neq V$, we have $b(B) = \sum b'(B_i) \geqslant \sum b(B_i) \geqslant \sum y(B_i) = -\sum y(\bar{B}_i) = -y(\bigcup \bar{B}_i) = y(B)$

for some $B_i \in \mathscr{B}'$, where $B = \bigcap B_i$ and $\bar{B}_i \cap \bar{B}_j = \emptyset$. For $B = V$, $b(B) = y(B) = 0$. ∎

Applying Lemma 2 to $y = \lambda_x$ we get

COROLLARY. *$(b, \mathscr{B})$ and $(b', \mathscr{B}')$ define the same Edmonds–Giles polyhedron.*

These lemmas enable us to formulate the feasibility theorem for the crossing case.

THEOREM 1′ *The linear system* (1) *has a solution if and only if*

$$\rho_f(\bigcup B_i) - \delta_g(\bigcup B_i) \leqslant \sum b'(B_{ij}) \tag{5}$$

for any disjoint nonempty sets B_i (possibly not in $\mathscr{B}'$), where each B_i is the intersection of pairwise codisjoint members B_{ij} of $\mathscr{B}'$ $(j = 1, 2,..., k_i)$. Moreover, if b', f, g are integral and (5) *holds,* (1) *has an integral solution.*

We note that Theorem 1A can be extended analogously.

Having the Edmonds–Giles polyhedron in an intersecting form one can try to apply the algorithm of Section 3 to this $(b, \mathscr{B})$. The only question is whether oracle (A) for $(b, \mathscr{B})$ is available provided that the same oracle is available for $(b', \mathscr{B}')$ (when it is denoted by (A′)). There is a small difficulty here to be overcome. In the algorithm of Section 3 at the beginning two extra nodes and a set E_1 of new arrows were adjoined to G and then a starting solution $x \in R^{E \cup E_1}$ was easily found. The difficulty now is that we do not seem to have a method to determine $\min b(B) - \lambda_x(B)$ over $u\bar{v}$-members of $\mathscr{B}$ by using oracle (A′) for arbitrary x. In Lemma 3, however, we shall see that such a method exists if x has the additional property that $\lambda_x(V) = 0$. Therefore we need a precalculation to determine such a starting solution x and later, during the subsequent augmentations, this property of x has to be maintained.

Let $x_0 \in R^E$ be an arbitrary vector with $f \leqslant x_0 \leqslant g$. In the first step we find a vector $z_0 \in R^V$ such that $z_0(V) = 0$ and $z_0(B) + \lambda_{x_0}(B) \leqslant b'(B)$ for $B \in \mathscr{B}'$. This is carried out by the discrete separation algorithm given in Section 4 as follows.

Choose an arbitrary node $r \in V$ and set $\mathscr{B}_1 = \{B : B \in \mathscr{B}', r \notin B\}$, $\mathscr{P} = \{V - B : B \in \mathscr{B}', r \in B\}$. Also define $b_1(B) = b'(B) - \lambda_{x_0}(B)$ for $B \in \mathscr{B}_1$ and $p(P) = -b'(V - P) + \lambda_{x_0}(V - P)$ for $P \in \mathscr{P}$. Obviously, $\mathscr{B}_1, \mathscr{P} \subset 2^{V-r}$ and $(b_1, \mathscr{B}_1)$ and $(p, \mathscr{P})$ are intersecting sub- and supermodular functions, respectively. In order to get a separating modular function $m \in R^{V-r}$, apply the discrete separation algorithm to these functions. Observe that the necessary oracles for this algorithm are available if oracle (A) is available for b'. Namely, minimizing $b(B) - m(B)$ $(m \in R^{V-r})$ over $u\bar{v}$-members of $\mathscr{B}_1$ is equivalent to minimizing $b(B) - m'(B)$ $(m' \in R^V)$ over $u\bar{v}$-members of $\mathscr{B}'$, where $m'(x) = m(x)$ for $x \in V - r$ and $m'(r)$ is a large enough number. The oracle for p can be obtained analogously.

Suppose the algorithm terminates by detecting sets $B_1, B_2,..., B_k$ and $P_1, P_2,..., P_l$, violating (4). This means that (i) $\sum b'(B_i) + \sum b'(\bar{P}_j) < 0$. Choosing $B_{k+1} = V - \bigcup (B_i : i = 1,..., k)$, (5) does not hold for $B_1, B_2,..., B_k, B_{k+1}$ since in this case each $B_i (i = 1,..., k)$ is a member of $\mathscr{B}'$ while $B_{k+1} = \bigcap (\bar{P}_j : j = 1,..., 1)$ and in (5) we get from (i) that $\sum b'(B_{ij}) < 0$ while, because $\bigcup (B_i : i = 1,..., k+1) = V$, $\rho_f(\bigcup B_i) = \delta_g(\bigcup B_i) = 0$.

Assume now that (4) holds and the algorithm finds a separating vector $m \in R^{V-r}$. Let $z_0 \in R^V$ be defined by $z_0(v) = m(v)$ if $v \in V - r$ and $z_0(r) = -m(V - r)$. Then z_0 satisfies the requirements, i.e., $z_0(V) = 0$ and $z_0(B) + \lambda_x(B) \leqslant b'(B)$ for $B \in \mathscr{B}'$.

As in Section 3 adjoin to G two extra nodes s, t and new arrows vs, tv $(v \in V)$. This time, however, set $x = (x_0, x_1)$, where x_0 is as above, $x_1(vs) = -z_0(v)$ if $z_0(v) < 0$, and $x_1(tv) = z_0(v)$ if $z_0(v) > 0$. (Other new arrows can be omitted.)

Let $\lambda_x(B) = \rho_x^1(B) - \delta_x^1(B)$, where ρ_x^1 and δ_x^1 concern the enlarged graph. Observe that $\lambda_x = z_0 + \lambda_{x_0}$ and thus x is a solution to (2) such that $\lambda_x(V) = 0$.

Now we show that the necessary oracle is available for such an x. Let y be a modular function such that $y(\emptyset) = y(V) = 0$ and $y(B) \leqslant b'(B)$ for each $B \in \mathscr{B}'$. The next lemma, when applied to $y = \lambda_x$, shows that $\min(b(B) - \lambda_x(B))$ over $u\bar{v}$-members of $\mathscr{B}$ can be computed with the help of oracle (A′).

LEMMA 3. *Given $(b, \mathscr{B})$, y as above, and $u, v \in V$, we have* $\min(b(B) - y(B) : B \in \mathscr{B}$, *$B$ a $u\bar{v}$-set*$) = \min(b'(B) - y(B) : B \in \mathscr{B}'$, *$B$ a $u\bar{v}$-set*$)$.

Proof. Let γ and γ' be the minimum values of the left- and right-hand sides, respectively. Since $\mathscr{B}' \subseteq \mathscr{B}$ and $b'(B) \geqslant b(B)$ for $B \in \mathscr{B}'$, we have $\gamma' \geqslant \gamma$. On the other hand let B be a $u\bar{v}$-set in $\mathscr{B}$ for which $\gamma = b(B) - y(B)$. Then $b(B) - y(B) = \sum (b'(B_i) : i = 1,..., k) - y(B) = \sum (b'(B_i) - y(B_i))$ for some $B_i \in \mathscr{B}'$ $(i = 1,..., k)$ with $\bar{B}_i \cap \bar{B}_j = \emptyset$. Here we used that $y(B) = -\sum y(\bar{B}_i) = \sum y(B_i)$. Among these sets B_i one is a $u\bar{v}$-set, say B_1; thus $b'(B_1) - y(B_1) \geqslant b(B_1) - y(B_1) \geqslant \gamma$. Hence $\gamma = b(B) - y(B) = b'(B_1) - y(B_1) + \sum (b'(B_i) - y(B_i) : i = 2, 3,..., k) \geqslant b(B_1) - y(B_1) \geqslant \gamma$, whence $b'(B_1) - y(B_1) = \gamma$, as required. ∎

Observe that an augmentation provides another vector $x' = (x_0', x_1')$ for which $\lambda_{x'}(V) = 0$. This is so because, since $b(V) = \lambda_x(V) = 0$, no jumping arrow enters or leaves V in the auxiliary digraph. Therefore, the algorithm can be continued with this x' and the necessary oracle is available throughout the algorithm.

One more difference occurs in the crossing case when no solution exists. In this case the following lemma, taken from [9] helps us.

LEMMA 4. *Any b-tight set $B \in \mathscr{B}$, for which the hypergraph $\{B(u): u \in B$ is connected$\}$, can be obtained constructively as the intersection of pairwise co-disjoint b'-tight members of $\mathscr{B}'$.*

In the original version of this paper from the algorithmic point of view the reduction of the crossing case to the intersecting one was not correct. It was Cunningham who pointed out this error; many thanks are due to him. In fact the same error occurred in [9], too. There the method for finding an optimum solution in the crossing case is correct if a starting feasible solution is already available. However, the method of finding a starting feasible solution works only for intersecting submodular functions and not for crossing ones. The two-step approach presented here overcomes that difficulty.

6. Polymatroidal Network Flows and Other Models

Lawler and Martel [19] and Hassin [17] introduced the concept of polymatroidal network flows. Here we show how their model can be formulated as a feasibility problem of the Edmonds–Giles polyhedron.

We are given a digraph $H=(U,A)$ with a source s and sink t. For each vertex v of H there are specified two capacity functions α_v and β_v. α_v (β_v) is defined on the subsets of the set A_v (B_v) of arrows entering (leaving) v. Both α_v and β_v are submodular on any pair of subsets, monotone nondecreasing (i.e., $\alpha_v(X) \geqslant \alpha_v(Y)$ whenever $X \supset Y$), and $\alpha_v(\varnothing)=\beta_v(\varnothing)=0$.

An *independent flow* f is a non-negative flow from s to t, (i.e., $f \in R^A_+$ and $\rho_f(v)=\delta_f(v)$ for $v \in U-\{s,t\}$) satisfying the inequalities $f(X) \leqslant \alpha_v(X)$ for $X \subseteq A_v$ and $f(X) \leqslant \beta_v(X)$ for $X \subseteq B_v$ $(v \in U)$.

The objective is to find an independent flow of maximum value. Martel and Lawler completely solved this problem by introducing the concept of lexicographically shortest augmenting paths. In order to reduce this problem to (1), assume the arrow $a=t_0s_0$ is in the graph and the objective is to find an independent circulation which maximizes the arc flow on t_0s_0.

Replace each vertex v of U by as many new vertices as there are arrows incident to v. Denote by $\varphi^-(v)$ $(\varphi^+(v))$ the set of new copies of v corresponding to A_v (B_v) and set $\varphi(v)=\varphi^+(v) \cup \varphi^-(v)$. We shall not distinguish between a subset of $\varphi^-(v)$ and the corresponding subset of A_v.

Therefore, we obtain the set V of $2|A|$ elements. The arrows in H determine a partition of V into 2-elements subsets. Denote by e_u and e_v the elements in V corresponding to the arrow $e=uv$ of H. For a subset X of U set $\varphi(X)=\bigcup(\varphi(v): v \in X)$.

Define $\mathscr{B}_1=\{\varphi(X): \varnothing \neq X \subset U\}$ and for $u \in U$ let $\mathscr{A}_u=\{F: \varnothing \neq F \subseteq \varphi^-(u)\}$ and $\mathscr{B}_u=\{F: \varnothing \neq V-F \subseteq \varphi^+(u)\}$. Let $\mathscr{B}'=\bigcup(\mathscr{A}_u \cup \mathscr{B}_u: u \in U) \cup \mathscr{B}_1$. Let us define

$$\begin{aligned} b'(F) &= 0 && \text{if } F \in \mathscr{B}_1, \\ &= \alpha_u(F) && \text{if } F \in \mathscr{A}_u, \\ &= \beta_u(V-F) && \text{if } F \in \mathscr{B}_u. \end{aligned}$$

Now it can easily be seen that $\mathscr{B}'$ is a crossing family, b' is submodular on crossing pairs, and there is a one-to-one correspondence between the solutions to (1) and the independent circulations. Therefore the crossing version of the algorithm mentioned at the end of Section 3 can be applied.

Finally we remark that the present approach makes it possible to solve the polymatroidal flow problem even if the function α_v (β_v) is defined on an intersecting subfamily on A_v (B_v) and is submodular on intersecting pairs only. The monotonicity can also be dropped. Furthermore, extra upper and lower bounds f and g can be accomodated on the arrows.

By the reduction above the optimization algorithm [2] applies to obtaining a minimal cost polymatroidal circulation.

There are other models for submodular functions. One of these is Fujishige's independent flow model [15]. Lawler and Martel showed by a simple elementary construction that this model can be formulated as a polymatroidal network flow problem. Fujishige's algorithm is not proved to be finite when the capacities may be irrational nor polynomial bounded for integral capacities.

Two other models are kernel systems [12] and generalized polymatroids [13]. It can be shown that optimization problems in these problems can be solved by an optimization method for the Edmonds–Giles polyhedron. See [14]. There are models which do not seem to fit into the Edmonds–Giles framework and we do not know any combinatorial algorithm for them. One such example is Hoffman–Schwartz' lattice polyhedron. An excellent survey by Schrijver exhibits a very accurate relationship between the various models [25]. See also [14].

Acknowledgments

I am deeply indebted to W. Cunningham for pointing out an error in the original version and to É. Tardos for the valuable conversations on this topic.

REFERENCES

1. W. H. CUNNINGHAM, Testing membership in matroid polyhedra, *J. Combin. Theory Ser. B* **36** (1984), 000–000.
2. W. H. CUNNINGHAM AND A. FRANK, A primal–dual algorithm for submodular flows, *Math. Oper. Res.*, in press.
3. E. A. DINITZ, Algorithm for solution of a problem of maximum flow in a network with power estimation, *Soviet Math. Dokl.* **11** (1970), 1277–1280.
4. J. EDMONDS AND R. GILES, A min–max relation for submodular functions on graphs, *in* "Studies in Integer Programming Proceedings, Workshop on Programming, Bonn, 1975" (P. L. Hammer, E. L. Johnson, and B. H. Korte, Eds.), Annals of Discrete Math. No. 1, pp. 185–204, North-Holland, Amsterdam, 1977.
5. J. EDMONDS, Matroid intersection, *Ann. Discrete Math.* **4** (1979), 39–49.
6. J. EDMONDS, Submodular functions, matroids, and certain polyhedra, *in* "Combinatorial Structures and Their Applications, Proceedings, International Conference, Calgary, Alberta, 1969," (R. Guy, H. Hanani, N. Sauer, and J. Schönheim, Eds.), pp. 69–87. Gordon & Breach, New York, 1970.
7. J. EDMONDS AND R. M. KARP, Theoretical improvements in algorithmic efficiency for network flow problems, *J. Assoc. Comput. Mach.* **19** (1972), 248–264.
8. L. R. FORD AND D. R. FULKERSON, "Flows in Networks," Princetong Univ. Press, Princeton, N. J., 1962.
9. A. FRANK, An algorithm for submodular functions on graphs, *Ann. Discrete Math.* **16** (1982), 97–120.
10. A. FRANK, How to make a digraph strongly connected, *Combinatorica*, **1**, No. 2 (1981), 145–153.
11. A FRANK, A weighted matroid intersection algorithm, *J. Algorithms* **2** (1981), 328–336.
12. A. FRANK, Kernel systems of directed graphs, *Acta Sci. Math. (Szeged)* **41** (1979), 63–76.
13. A. FRANK, "Generalized Polymatroids," Report 81206-OR, Institut für Operations Research, Bonn, 1981; *in* "Proceedings, Sixth Hungarian Combinatorial Colloquium" (Eger), in press.
14. A. FRANK, Submodular flows, *in* "Proceedings, Silver Jubilee Conference on Combinatorics, Waterloo 1982," in press.
15. S. FUJISHIGE, Algorithms for solving the independent flow problems, *J. Oper. Res. Soc. Japan* **21**, No. 2, (1978), 189–203.
16. M. GRÖTSCHEL, L. LOVÁSZ, AND A. SCHRIJVER, The ellipsoid method and its consequences in combinatorial optimization, *Combinatorica* **1**, No. 2 (1981), 169–197.
17. R. HASSIN, On network flows, *Networks* **12** (1981), 1–21.
18. A. HOFFMAN, Some recent applications of the theory of linear inequalities to extremal combinatorial analysis, *Proc. Sympos. Appl. Math.* **10** (1966), 113–127.
19. E. L. LAWLER AND C. U. MARTEL, Computing maximal "polymatroidal" network flows, *Math. Oper. Res.* **7**, No. 3 (1982).
20. C. LUCCHESI AND D. H. YOUNGER, A minimax theorem for directed graphs, *J. London Math. Soc.* **17** (2), (1978), 369–374.
21. C. ST. J. A. NASH-WILLIAMS, Well-balanced orientations of finite graphs and unobtrusive odd-vertex-pairings, *in* "Recent Progress in Combinatorics," Academic Press, New York, 1969.
22. W. R. PULLEYBLANK, Total dual integrality and *b*-matchings, *Oper. Res. Lett.* **1** No. 1 (1981), 28–30.
23. P. SCHÖNSLEBEN, "Ganzzahlige Polymatroid-Intersektions-Algorithmen," Ph.D. thesis, Eidgenössischen Technischen Hochschule, Zürich, 1980.

24. A. SCHRIJVER, On total dual integrality, *Linear Algebra Appl.* **38** (1981), 27–32.
25. A. SCHRIJVER, Total dual integrality from directed graphs, crossing families, and sub- and supermodular functions, *in* "Proceedings, Silver Jubilee Conference on Combinatorics, Waterloo 1982," in press.
26. U. ZIMMERMANN, Minimization of some nonlinear functions over polymatroidal network flows, *Ann. Discrete Math.* **16** (1982), 287–309.

Printed by the St. Catherine Press Ltd., Tempelhof 41, Bruges, Belgium