COMBINATORICA 1(2) (1981) 145—153

HOW TO MAKE A DIGRAPH STRONGLY CONNECTED

András FRANK

Research Institute for Telecommunication
Gábor Á. u. 65, Budapest, Hungary H—1026

and

JATE Bolyai Institute, Szeged, Hungary H—6720

Received 13 November 1979

Given a directed graph G, a *covering* is a subset B of edges which meets all directed cuts of G. Equivalently, the contraction of the elements of B makes G strongly connected. An $O(n^5)$ primal-dual algorithm is presented for finding a minimum weight covering of an edge-weighted digraph. The algorithm also provides a constructive proof for a min-max theorem due to Lucchesi and Younger and for its weighted version.

1. Introduction

The purpose of this paper is to present a polynomial-bounded algorithm for making a directed graph strongly connected by contracting a minimum number of edges, or more generally a set of edges of minimum total cost. At the same time the algorithm proves an important theorem of Lucchesi and Younger [8] and also its extension due to Edmonds and Giles [1].

We are given a connected digraph $G=(V, E)$ with a non-negative integer cost function d on the edge set E. We say G is *strongly connected* if, for arbitrary vertices x and y of V, there exists a directed path from x to y. As is well known, G is not strongly connected if and only if there is no edge leaving X, for some non-empty proper subset X of V. Such a set X is said to be a *kernel* and the non-empty set $D(X)$ of edges entering X is called a *directed cut,* or dicut (determined by X).

An edge set $B\subseteq E$ is called a *covering* if every dicut uses at least one element of B. The *cost* $d(B)$ of B is $\sum (d(e)\colon e\in B)$. Obviously, B is a covering if and only if by contracting its elements G becomes strongly connected. So we want to find a minimum cost covering. The Lucchesi—Younger theorem concerns this minimum when $d(e)\equiv 1$:

Theorem A. *The minimum cardinality of a covering is equal to the maximum number of (edge) disjoint directed cuts.*

For a brief proof, see Lovász [7]. For the general case we need a definition.

AMS (1980) subject classification: 05 C 20, 90 C 10, 05 C 40; 68 C 25, 68 E 10

A family K of not necessarily distinct dicuts is called *d-independent* if no edge e occurs in more than $d(e)$ members of K.

Theorem B. *The minimum cost τ_d of a covering is equal to the maximum cardinality ν_d of a d-independent family of directed cuts.*

This result has been obtained by Edmonds and Giles from a much more general minimax relation concerning submodular functions, but it can also be derived from Theorem A by elementary construction. In another paper [3] I shall show the method described here can be extended to get an algorithmic proof of this general theorem of Edmonds and Giles. So we shall have an algorithm having specializations (besides the present one) as the weighted matroid intersection algorithm, the minimum cost circulation algorithm, the so called independent flow problem [4] and so on.

The previously used methods for proving the Lucchesi—Younger theorem were based on the same fundamental principles and were not algorithmic in character. The present approach is quite different.

2. Optimality criteria for coverings

We say that an edge $e=(xy)$ *enters* a subset $X\subset V$ if its head y is in X but the tail x is not. An edge *leaves* X if it enters $V-X$.

For $F\subseteq E$, the *indegree* $\varrho_F(X)$ (*outdegree* $\delta_F(X)$) means the number of edges in F entering (leaving) X. A simple counting shows that:

(2.1) $\varrho_F(X)+\varrho_F(Y) = \varrho_F(X\cup Y)+\varrho_F(X\cap Y)$, *for kernels* X, Y.

Let $F^*=\{(xy)\colon (yx)\in F\}$. It will be convenient to consider V to be a kernel. Two kernels X and Y are *intersecting* if $X\cap Y\neq\emptyset$. If, in addition, $X\cup Y\neq V$ then they are *crossing*.

We shall be referring to a covering B throughout the algorithm. The edges of the current B will be called *blue* edges, the remaining edges of G *white* edges.

For the quantities in Theorem B, obviously $\nu_d\leqq\tau_d$. To prove the equality we shall construct a covering B and a family K of dicuts such that:

(2.2) (a) *Every blue edge e is in exactly $d(e)$ dicuts from K.*
(b) *Every white edge e is in at most $d(e)$ dicuts from K.*
(c) $\varrho_B(X)=1$ *for* $X\in K$.

Our method will yield a B and a K satisfying (b), (c) and the following condition:

(a′) *Every blue edge e is in at least $d(e)$ dicuts from K.*

However, in this case we can immediately achieve (a); namely, whenever $e\in B$ and e is in $k>d(e)$ dicuts from K, we omit $k-d(e)$ of these dicuts. The family K will be produced by potentials, which we shall define after presenting some simple notions and propositions.

Denote by $c(X)$ $(X\subseteq V)$ the number of weak components of $G-X$.

(2.3) **Lemma.** *For any kernels X, Y we have* $c(X)+c(Y)\leqq c(X\cap Y)+c(X\cup Y)$.

Proof. Let us consider G as an undirected graph. Denote the set of edges with both end-vertices in $V-X$ by $B(X)$. For the rank of $B(X)$ in the cycle matroid of G, we have $r(B(X))=|V-X|-c(X)$. Furthermore, $B(X\cup Y)=B(X)\cap B(Y)$ and $B(X\cap Y)=B(X)\cup B(Y)$. (The fact that X and Y are kernels and thus there is no edge between $X-Y$ and $Y-X$ is exploited in the second equality.) Now the lemma follows since r is submodular. ▮

It can be seen that, for a kernel X, the dicut $D(X)$ is the union of $c(X)$ disjoint dicuts, the kernels of which are the complements (concerning V) of the components of $G-X$. We say these dicuts *belong* to X and denote their set by $L(X)$. Trivially,

(2.4) For a kernel X and covering B, $\varrho_B(X)\geqq c(X)$.

A kernel X (and the dicut $D(X)$) is said to be *strict* (with respect to B) if $\varrho_B(X)=c(X)$. If, furthermore, $c(X)=1$ then X is *1-strict* (with respect to B). Note that V is always strict and:

(2.5) For a strict X, $L(X)$ consists of 1-strict dicuts which partition $D(X)$.

(2.6) **Lemma.** *For intersecting strict kernels X, Y, both $X\cap Y$* and *$X\cup Y$ are strict.*

Proof. *From* (2.1), (2.3) *and* (2.4) *we have* $c(X)+c(Y)=\varrho_B(X)+\varrho_B(Y)=\varrho_B(X\cap Y)+$ $+\varrho_B(X\cup Y)\geqq c(X\cap Y)+c(X\cup Y)\geqq c(X)+c(Y)$ *which implies* $\varrho_B(X\cap Y)=c(X\cap Y)$ *and* $\varrho_B(X\cup Y)=c(X\cup Y)$. ▮

Repeated applications of (2.4) yields:

(2.7) If a set of strict kernels forms a connected hypergraph then their union is strict again.

Let $P(x)$ denote the intersection of all strict kernels containing a fixed vertex $x\in V$. (Since V is strict, $P(x)$ is well-defined).

From (2.4) we can see that:

(2.8) $P(x)$ is the (unique) least strict set containing x, and $y\in P(x)$ for any edge $(xy)\in E$.

In § 5 we shall show how $P(x)$ can be effectively determined.

A kernel X is called *closed* (with respect to B) if $x\in X$ implies $P(x)\subseteqq X$.

(2.9) **Lemma.** *A kernel X is closed if and only if it is the union of disjoint strict kernels.*

Proof. The "if" part is trivial. Conversely, consider the hypergraph H on X formed by the sets $P(x)$ $(x\in X)$. The union of these sets is X and, by (2.7), the components of H provide the required partition of X. ▮

The (unique) partition $X_1, X_2, \dots, X_k$ obtained in the proof is called the *strict partition* of X.

Let $K(X)=\cup\{L(X_i)\colon i=1,2,\dots,k\}$. Since the dicuts $D(X_i)$ are disjoint and partition $D(X)$ we get by (2.5):

(2.10) For a closed kernel X, $K(X)$ consists of disjoint 1-strict dicuts, the union of which is $D(X)$.

(2.11) **Lemma.** *Let B' be another covering. If X is closed with respect to B and $\varrho_{B'}(X)=\varrho_B(X)$ then X is also closed with respect to B'.*

Proof. For the strict partition of X we have $\varrho_B(X)=\Sigma\varrho_B(X_i)=\Sigma c(X_i)\leqq\varrho_{B'}(X_i)=$ $=\varrho_{B'}(X)$, therefore the X_i are strict with respect to B'. By (2.9) the lemma follows. ▮

3. Dicuts and potentials

By a *potential* p we mean an integer valued function on V.

For an edge $(xy)\in E$ let $\bar{d}(xy)=d(xy)-p(y)+p(x)$. Suppose that we have a covering B and a potential p for which the following optimality criteria hold:

(3.1) (a) *For every blue edge (xy), $\bar{d}(xy)\leqq 0$.*
(b) *For every white edge (xy), $\bar{d}(xy)\geqq 0$.*
(c) *For every $y\in P(x)$, $p(y)\geqq p(x)$.*

In this case we can produce a family K of dicuts which, together with B, satisfy (2.2) (a'), (b) and (c). For a dicut D, x_D will denote the number of its copies occuring in K.

Since any constant can be added to p without destroying the optimality criteria, we can assume that the minimum value of p is zero. Denote by $0=p_0<$ $<p_1<\ldots<p_m$ the different values of p. Define $V_i=\{x: p(x)\geqq p_i\}$ for $i=1, 2, \ldots, m$. Then $\emptyset\neq V_i\subset V$ and one can see by (2.8),

(3.2) Criterion (3.1)(c) is equivalent to the fact that each V_i is a closed kernel.

Let $x_D=\Sigma(p_i-p_{i-1})$ where the summation is taken over those indices i for which $D\in K(V_i)$. (The empty sum is zero).

We assert that K and B satisfy the requirements. By (2.10) K consists of 1-strict kernels, thus (2.2)(c) holds. If e is a blue (or white) edge then, by the optimality criterion (a) ((b), resp.), e occurs in at least (at most, resp.) $d(e)$ dicuts among the $D(V_i)$, whence, by (2.10), (2.2)(a) ((b), resp.) holds.

4. Improving a covering - potential pair

The above argument shows that to prove Theorem B all that is necessary is to construct a covering B and a potential p satisfying the optimality criteria.

The core of our procedure is the following.

(4.1) **Algorithm.**
Input: A covering B, a potential p, and a blue edge (ab) such that (3.1)(b) and (c) hold but (ab) violates (a).
Output: A covering B' and a potential p' such that (3.1)(b) and (c) hold again, (ab) does not violate (a) and if an edge violates (3.1)(a) then it violated (3.1)(a) with respect to B and p.

Assume this algorithm is available. Repeat it successively until there exists no blue edge violating (3.1)(a). At the beginning B may be the edge set of a spanning tree and $p\equiv 0$. Then after no more than $|B|\leqq n-1$ applications of algorithm (4.1) (where $n=|V|$), we get a covering B and a potential p which satisfy all three optimality criteria.

We define an auxiliary digraph $H=(V, A)$ (depending on G, B and p) as follows. Let A consist of the following three not necessarily disjoint parts A_B, A_W and A_R. (H may contain multiple edges.)

(4.2)
$$A_B = \{(xy)\colon (xy)\in B, \bar{d}(xy) \geqq 0\}$$
$$A_W = \{(yx)\colon (xy)\in E-B, \bar{d}(xy) \leqq 0\}$$
$$A_R = \{(xy)\colon y\in P(x), p(y) = p(x)\}.$$

We refer to the elements of A_B, A_W and A_R as blue, white and red edges, respectively.

Let us try to find a directed path from b to a in H. There may be two cases.

Case 1. There is no directed path from b to a in H. That is, $a\notin T=\{y\colon y$ can be reached from b in $H\}$. Change p as follows.

(4.3)
$$p'(x) = \begin{cases} p(x) & \text{if } x\notin T \\ p(x)+\delta & \text{if } x\in T, \end{cases}$$

where $\delta=\min\{\delta_e, \delta_B, \delta_W, \delta_R\}$ *and* $\delta_e=\bar{d}(ab)$, $\delta_B=\min\{-\bar{d}(xy)\colon (xy)\in B, (xy)$ *leaves* $T\}$, $\delta_W=\min\{\bar{d}(xy)\colon (xy)\in E-B, (xy)$ *enters* $T\}$, $\delta_R=\min\{p(y)-p(x)\colon x\in T, y\in P(x)-T\}$.

(The minimum is defined to be ∞ when it is taken over the empty set.)

Note that the definition of T implies that:

(4.4) In H there is no edge leaving T.

(4.5) **Claim.** $\delta>0$.

Proof. $\delta_e>0$ is equivalent to the fact that (ab) violates (3.1)(a) (with respect to B and p). $\delta_B>0$. Otherwise $\bar{d}(xy)\geqq 0$ for some edge $(xy)\in B$ leaving T. Then $(xy)\in A_B$, contradicting (4.4). $\delta_W>0$. Otherwise $\bar{d}(xy)\leqq 0$ for some edge $(xy)\in E-B$ entering T. Then $(yx)\in A_W$, contradicting (4.4). $\delta_R>0$. By (3.1)(c) $p(y)\geqq p(x)$. Thus $\delta_R\leqq 0$ would imply $p(y)=p(x)$, i.e. (xy) would be a red edge leaving T, contradicting (4.4). ∎

For the new $\bar{d}'(xy)=d(xy)-p'(y)+p'(x)$ we have

(4.6)
$$\bar{d}'(xy) = \begin{cases} \bar{d}(xy)+\delta & \text{if } (xy) \text{ leaves } T, \\ \bar{d}(xy)-\delta & \text{if } (xy) \text{ enters } T, \\ \bar{d}(xy) & \text{otherwise.} \end{cases}$$

Claim. *If, for a blue edge, (3.1)(a) was true then it continues to hold.*

Proof. Let $(xy)\in B$ and $\bar{d}(xy)\leqq 0$. If, indirectly, $\bar{d}'(xy)>0$ then because of (4.5) and (4.6), (xy) leaves T and thus $-\bar{d}(xy)\geqq\delta_B\geqq\delta$, i.e., by (4.6), $\bar{d}'(xy)\leqq 0$, a contradiction. ∎

Claim. *(3.1)(b) remains valid.*
Proof. If $(xy)\in E-B$ then $\bar{d}(xy)\geqq 0$ and the indirect assumption $\bar{d}'(xy)<0$ imply, by (4.5) and (4.6), that (xy) enters T. Then $\bar{d}(xy)\geqq\delta_W\geqq\delta$, i.e., by (4.6), $\bar{d}'(xy)\geqq 0$, a contradiction. ▮

Claim. *(3.1)(c) remains valid.*
Proof. Let $y\in P(x)$ ($P(x)$ does not depend on the potential change) and assume, on the contrary, $p'(y)<p'(x)$. Then $x\in T$ and $y\in P(x)-T$. Therefore $p'(y)=p(y)$ and $p'(x)=p(x)+\delta$, whence $p(y)-p(x)<\delta$. On the other hand, $p(y)-p(x)\geqq\delta_R\geqq\delta$, a contradiction. ▮

If $\delta=\delta_e$ then (ab) satisfies (3.1)(a) and thus algorithm (4.1) ends.

If $\delta>\delta_e$ then repeat algorithm (4.1) with the input B, p' and (ab). We observe that in the new auxiliary graph H' the edge set spanned by T is the same as in H. Moreover, the definition of δ assures that H' contains at least one edge leaving T (which is in A_B, A_W or A_R according as δ is equal to δ_B, δ_W or δ_R). Consequently the set T' of vertices which can be reached by a directed path from b in H' properly includes T. Thus, after at most $n-1$ iterations, either $\delta=\delta_e$ or $a\in T$ (case 2) is achieved.

Case 2. In H a can be reached from b. Let U be a ba-path of minimum number of edges. (We shall use only that $U=(x_0=b, x_1, \dots, x_k=a)$ does not span a red "cut off" edge, i.e. $(x_i x_{i+j})$ $(j\geqq 2)$ is not a red edge.)

Since $(ab)\in A_B$, U and (ab) form a directed circuit C in H. Let C_B and C_W denote the set of blue and white edges of C respectively. Now the elements of C_B (C_W^*) correspond to blue (white) edges of G (where C_W^* denotes C_W with orientation reversed).

Let $B'=B-C_B\cup C_W^*$. In other words, change the colors of those edges of G which correspond to the blue and white edges of C.

(4.7) **Lemma.** *B' is a covering.*
Proof. For a kernel X, denote $\varrho_r(X)$ $(\delta_r(X))$ the number of red edges of U entering (leaving) X. Then we have

$$\varrho_{B'}(X) = \varrho_B(X)+\varrho_r(X)-\delta_r(X). \tag{4.8}$$

This is quite clear when $\varrho_r(X)=\delta_r(X)=0$, and hence the general case can also be proved by a simple induction on $\varrho_r(X)+\delta_r(X)$.

Let $\varepsilon(X)=\varrho_B(X)-c(X)$. Then $\varepsilon(X)\geqq 0$ and the equality holds just if X is strict with respect to B. From (2.1) and (2.3) we get

$$\varepsilon(X)+\varepsilon(Y) \geqq \varepsilon(X\cap Y)+\varepsilon(X\cup Y) \quad \text{for intersecting kernels } X,\ Y. \tag{4.9}$$

To prove the lemma we have to verify that $\varrho_{B'}(X)\geqq c(X)$ for each kernel X. This follows from (4.8) and the inequality $\varepsilon(X)\geqq\delta_r(X)$. Before proving the latter, let us consider a kernel X with $\delta_r(X)>0$.

Let (xy) be a red edge of U leaving X such that $p(y)$ $(=p(x))$ is as great as possible and if there is more than one edge of this type then (xy) is the first one of U (starting from b). Let $X'=X\cup P(x)$ ($P(x)$ concerns B).

(4.10) **Claim.** $\delta_r(X') = \delta_r(X)-1.$

Proof. Because no red edge leaves $P(x)$ and (xy) does not leave X', we have $\delta_r(X') \leqq \delta_r(X)-1$. On the other hand, if (st) is another red edge of U leaving X then $t \notin P(x)$ (i.e. (st) leaves X', too): in the contrary case, by (3.1)(c), $p(t) \geqq p(x)$ and thus, because of the maximality of $p(y)$, $p(t)=p(x)$. Consequently, (xt) would be a red (cut off) edge spanned by U, which contradicts the minimal property of U. ▮

Claim. $\varepsilon(X) \geqq \delta_r(X)$.
Proof. We use induction on $\delta_r(X)$. Since $\varepsilon(X) \geqq 0$ we can assume $\delta_r(X)>0$. From (4.9), $\varepsilon(X)=\varepsilon(X)+\varepsilon(P(x)) \geqq \varepsilon(X \cap P(x))+\varepsilon(X')>\varepsilon(X')$. By (4.10) we can apply the induction hypothesis for X' and get $\varepsilon(X)>\varepsilon(X') \geqq \delta_r(X')=\delta_r(X)-1$, that is, $\varepsilon(X) \geqq \delta_r(X)$ which proves the claim. ▮

This completes the proof of Lemma (4.7). ▮

Let us consider what has happened to the optimality criteria.

Claim. *(3.1)(a) is valid for the new blue edges.*
Proof. If (xy) is a new blue edge of G then (yx) was a white edge of H and thus $\bar{d}(xy) \leqq 0$, as required (see (4.2)). ▮

Claim. *(3.1)(b) remains valid.*
Proof. If (xy) is a new white edge of G then (xy) was a blue edge of H and thus $\bar{d}(xy) \geqq 0$, as required. ▮

Claim. *(3.1)(c) remains valid.*
Proof. We suppose again that the minimum of p is zero. By (3.2) it suffices to prove that $V_i=\{x: p(x) \geqq i\}$ is a closed kernel with respect to B', for each positive value i of p. By the definition of A_R and V_i, $\varrho_r(V_i)=\delta_r(V_i)=0$. From these and (4.8) we obtain $\varrho_B(V_i)=\varrho_{B'}(V_i)$. Apply Lemma (2.11). ▮

We have now completed the correctness proof of algorithm (4.1) and consequently the proof of Theorem B. ▮

5. Complexity

At this point we examine how $P(x)$ can effectively be determined. We are given an arbitrary covering B. Using (2.6), it can easily be checked that:

(5.1) A kernel X is strict if and only if X is the non-empty intersection of some 1-strict kernels.

Let $B=\{e_i=(x_i y_i): i=1, 2, \dots, l\}$. Let G_i denote the graph obtained from G by adding a set $(B-e_i)^*$ of new edges (i.e. the new edges are the reversed elements of $B-e_i$). Let $P_i(x)=\{y: y$ can be reached from x in $G_i\}$. Therefore, using the well-known labeling technique [2], $P_i(x)$ can be produced in at most cn^2 steps.
Let $R(x)=\cap\{P_i(x): i=1, 2, \dots, l\}$.

(5.2) **Lemma.** $P(x)=R(x)$.
Proof. $P_i(x)$ is either V or a 1-strict set. Because of (5.1), $P(x)$ is the intersection of 1-strict kernels containing x and thus $P(x) \subseteq R(x)$. On the other hand, if there

exists a vertex y in $R(x)-P(x)$, then $x \in X$, $y \notin X$ for some 1-strict kernel X. If e_i is the only edge of B entering X then $P_i(x) \subseteq X$, i.e. $y \notin P_i(x)$, a contradiction. ∎

Lemma (5.2) enables us to construct $P(x)$ in $O(n^3)$ steps for a fixed x. So determining all $P(x)$'s requires at most $O(n^4)$ steps.

It is conceivable that there may be a better way to find all $P(x)$'s. It would be very useful to have a procedure of complexity $f(n) \leq n^3$ (or perhaps $\leq n^2$) since, as we shall see, computationally this is the crucial part of the algorithm.

The other part of the algorithm (4.1) finds a *ba*-path in H. To this end we can again apply the labeling technique. In Case 1 the set T is just the set of vertices having received a label during the labeling algorithm. In Case 2 the *ba*-path U produced by the labeling algorithm is automatically free of cut off edges.

The labeling algorithm uses at most cn^2 steps. Moreover, if $\delta > \delta_e$ occurs in the course of the algorithm and we apply (4.1) again with the modified potential, then the labels calculated previously can be used ($T \subset T'$). (Note that, in this case the new auxiliary graph arises simply from the old one by joining some new edges leaving T and deleting some old ones entering T.)

Therefore the whole algorithm (4.1) needs at most $cn^2+f(n)$ steps. Since (4.1) is applied at most $(n-1)$ times, the complexity of the algorithm developed here is estimated to be $n(cn^2+f(n)) \leq cn^5$.

Finally, given an optimal pair B, p, we have to construct the corresponding optimal family of dicuts (cf. § 3). This can be done in $O(n^4)$ steps as follows. The parts of the strict partition of each V_i are precisely the components of the graph $G_i=(V_i, E_i)$ where $E_i=\{(xy)\colon y \in P(x), x, y \in V_i\}$. Therefore these parts can be determined in $O(n^2)$ steps, hence $D(V_i)$ can be partitioned into 1-strict dicuts in $O(n^3)$ steps. The $O(n^4)$ bound follows by observing that there are no more than n sets V_i.

6. Formal description of the algorithm

Step 0. (Start) Let B be a covering and p be a potential which satisfy (3.1)(b) and (c). At the beginning, B may be the edge set of a spanning tree and $p \equiv 0$. We call the elements of B and $E-B$ blue and white, respectively.

Step 1.

1.0 Determine $P(x)$ for all $x \in V$.

1.1 If every blue edge satisfies (3.1)(a): *Halt*. The current covering B is optimal.

1.2 Select a blue edge $e=(ab)$ violating (3.1)(a).

1.3 Construct the auxiliary graph H and try to find a *ba*-path in H by the labeling technique (using the labels defined but not yet removed previously). If this path U exists, go to Step 3.

Step 2. (potential change)

2.0 Let T be the set of labeled vertices. Calculate δ and let $p(x):=p(x)+\delta$ for $x \in T$.

2.1 If $\delta=\delta_e$ remove all the labels and go to 1.1.

2.2 Go to 1.3.

Step 3. (covering change)

Let $C=U+e$ and denote C_B and C_W the set of blue and white edges of C, respectively. Let $B:=B-C_B\cup C_W^*$. Go to 1.0.

Remark. We note that the algorithm works without regard to the integrality of the cost function d. The only difference is that p may assume non-integral values as well. In this case Theorem B must slightly be modified as follows.

Theorem C. [1] The minimum weight of a covering is equal to the maximum sum Σx_D where the non-negative variables x_D are associated with the directed cuts D so that the cost $d(e)$ of any edge e is at least the sum of variables associated with dicuts containing e.

Remark. Recently I have learnt that C. L. Lucchesi [9] and A. V. Karzanov [5] gave polynomial algorithms for Theorem A.

References

[1] J. Edmonds and R. Giles, A min-max relation for submodular functions on graphs, in: "*Studies in integer programming*" (Proc. Workshop on Integer Programming Bonn, 1975; P. L. Hammer, E. L. Johnson, B. H. Korte, eds.), *Annals of Discrete Math.* **1** (1977) 185—204.

[2] L. R. Ford, Jr. and D. R. Fulkerson, *Flows in Networks,* Princeton Univ. Press, Princeton, N.J., 1962.

[3] A. Frank, An algorithm for submodular functions on graphs, submitted to *Annals of Discrete Math.*

[4] S. Fujishige, Algorithms for solving the independent flow problems, *J. Operation Res. Soc. Japan,* Vol. **21**, No. 2, June (1978).

[5] A. V. Karzanov, On the minimal number of arcs of a digraph meeting all its directed cutsets, (abstract) *Graph Theory Newsletters,* Vol. **8** (No. 4) March 1979.

[6] E. L. Lawler, *Combinatorial Optimization: Networks and Matroids,* Holt, Rinehart and Winston, New York, 1976.

[7] L. Lovász, On two minimax theorems in graph theory, *J. Combinatorial Theory* (B) **21** (1976) 96—103.

[8] C. L. Lucchesi and D. H. Younger, A minimax relation for directed graphs, *J. London Math. Soc.* (2) **17** (1978) 369—374.

[9] C. L. Lucchesi, *A minimax equality for directed graphs,* Ph. D. Thesis, Univ. of Waterloo, Waterloo, Ont. 1976.