Mathematical Programming 8 (1975) 207–226.
North-Holland Publishing Company

ON THE FOUNDATIONS OF LINEAR AND INTEGER LINEAR PROGRAMMING I*

Jack E. GRAVER
Syracuse University, Syracuse, New York, U.S.A.

Received 3 June 1974
Revised manuscript received 10 January 1975

In this paper we consider the question: how does the flow algorithm and the simplex algorithm work? The usual answer has two parts: first a description of the "improvement process", and second a proof that if no further improvement can be made by this process, an optimal vector has been found. This second part is usually based on duality – a technique not available in the case of an arbitrary integer programming problem. We wish to give a general description of "improvement processes" which will include both the simplex and flow algorithms, which will be applicable to arbitrary integer programming problems, and which will *in themselves assure convergence* to a solution.

Geometrically both the simplex algorithm and the flow algorithm may be described as follows. At the i^{th} stage, we have a vertex (or feasible flow) to which is associated a finite set of vectors, namely the set of edges leaving that vertex (or the set of unsaturated paths). The algorithm proceeds by searching among this special set for a vector along which the gain function is increasing. If such a vector is found, the algorithm continues by moving along this vector as far as is possible while still remaining feasible. The search is then repeated at this new feasible point.

We give a precise definition for sets of vectors, called test sets, which will include those sets described above arising in the simplex and flow algorithms. We will then prove that any "improvement process" which searches through a test set at each stage converges to an optimal point in a finite number of steps. We also construct specific test sets which are the natural extensions of the test sets employed by the flow algorithm to arbitrary linear and integer linear programming problems.

0. Introduction

The purpose of this paper is to give an exposition of linear programming which yields a parallel development for integer linear programming. It is hoped that this treatment will produce some insights into the structures of some of the usual algorithms and into the nature of the difficul-

* This research was supported in part by the National Science Foundation.

ties inherent in integer linear programming problems. This development is not intended for computer consumption and we will dispense with the usual special forms so necessary for actual machine computation.

Consider the question: how does the flow algorithm and the simplex algorithm work? The usual answer has two parts: first a description of the "improvement process" and second a proof that if no further improvement can be made by this process, an optimal vector has been found. This second part is usually based on duality – a technique not available in the case of an arbitrary integer programming problem. We wish to give a general description of "improvement processes" which will include both the simplex and flow algorithms, which will be applicable to arbitrary integer programming problems, and which will *in themselves assure convergence* to a solution.

Geometrically both the simplex algorithm and the flow algorithm may be described as follows. At the i^{th} stage, we have a vertex (or feasible flow) to which is associated a finite set of vectors, namely the set of edges leaving that vertex (or the set of unsaturated paths). The algorithm proceeds by searching among this special set for a vector along which the gain function is increasing. If such a vector is found, the algorithm continues by moving along this vector as far as is possible while still remaining feasible. The search is then repeated at this new feasible point.

In Section 2, sets of vectors like those which arise in the simplex and flow algorithms will be given a precise definition. We will call them test sets. We will then prove that any "improvement process" which searches through a test set at each stage converges to an optimal point in a finite number of steps. In Sections 3 and 4, we construct specific test sets. These are the natural extensions of the test sets which arise in the flow algorithm.

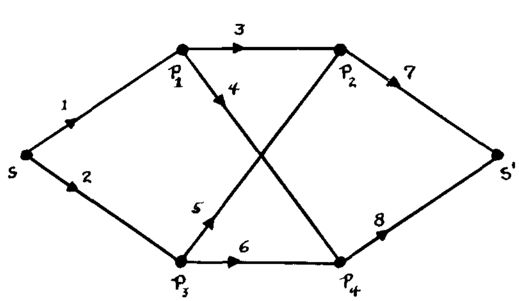

Fig. 1.

At this point we introduce a very simple example which will be used throughout this paper to illustrate the various concepts and results. Let N be the network drawn in Fig. 1. Let s be the source and s' the sink, and assume edges 1, 2, 7, 8 have capacity 4 while edges 3, 4, 5, 6 have capacity 3. Considering this as an integer programming problem (only integral flows are permitted), we make the additional condition that only an even number of units may flow through edge number four. This last condition makes the problem different enough from the usual flow problem that it may be used to illustrate "how things work" in an arbitrary integer linear programming problem.

1. Notation and definitions

The following letters and symbols are fixed throughout this paper.

$\mathbf{Q}$: the rational numbers. The elements of $\mathbf{Q}$ will be denoted by lower case greek letters.

$\mathbf{Z}$: the integers. The elements of $\mathbf{Z}$ are also denoted by lower case greek letters, or m, n, i and k.

$\mathbf{Q}^n$: the vector space of n-tuples of rational numbers. The vectors in $\mathbf{Q}^n$ will be denoted by lower case latin letters.

$\mathbf{Z}^n$: the vectors in $\mathbf{Q}^n$ with integral components.

$v\,w$: the usual inner product of the vectors v and w.

R and S: two fixed, finite, disjoint subsets of $\mathbf{Z}^n$.

$\beta(b)$: a fixed integer for each vector $b \in R \cup S$.

D (domain): the solution set of the system

$$b\,x = \beta(b) \quad \text{for all } b \in R,$$

$$b\,x \geqslant \beta(b) \quad \text{for all } b \in S.$$

$D_{\mathbf{Z}}$: $D \cap \mathbf{Z}^n$. The vectors in D and $D_{\mathbf{Z}}$ are usually called points and will usually be denoted by x or y.

V: the subspace which is the solution set of $b\,x = 0$ for all $b \in R$.

$V_{\mathbf{Z}}$: $V \cap \mathbf{Z}^n$. The vectors of V or $V_{\mathbf{Z}}$ will usually be denoted by u, v, or w.

g: a fixed vector in $\mathbf{Q}^n$ called the gain function, gx is the function value at x.

1.1. Definition. If $x \in D_{(\mathbf{Z})}$, we say that x is *(integral) feasible*. If, in addition, $gx \geqslant gy$ for all $y \in D_{(\mathbf{Z})}$, we say that x is *(integral) optimal*.

Many statements in this paper will be in the above form. The intention is that they be read twice: once as a linear programming statement with all symbols and words in parenthesis deleted, and once again as an integer linear programming statement with the parenthetical symbols and words included.

1.2. Definition. An *(integral) linear program* consists of the sets R and S, constants $\beta(b)$, and gain function g all defined as above. We say that the (integral) linear program has been *solved* when either an (integral) optimal vector is found or it is demonstrated that no such vector exists.

We may put our example in this form. For $i = 1, 2, 3, 5, 6, 7, 8$, let the i^{th} component of the 8-tuple $x = (x_1, x_2, ..., x_8)$ represent the flow through the i^{th} edge of N (in the direction of the arrow). Let x_4 represent half the flow through the fourth edge; hence the condition that x_4 is integral will force the flow through the fourth edge to be even. In order that x be a flow it must satisfy a "conservation condition" at each vertex. These are given by the four vectors in R. $R = \{b_1, b_2, b_3, b_4\}$, where

$$b_1 = (1, 0, -1, -2, 0, 0, 0, 0), \quad b_2 = (0, 0, 1, 0, 1, 0, -1, 0)\,,$$
$$b_3 = (0, 1, 0, 0, -1, -1, 0, 0), \quad b_4 = (0, 0, 0, 2, 0, 1, 0, -1)\,;$$
$$\beta(b_1) = \beta(b_2) = \beta(b_3) = \beta(b_4) = 0\,.$$

In order that x be a feasible flow it must satisfy sixteen inequalities, two for each edge. For $i = 1, 2, 3, 5, 6, 7, 8$, let e_i denote the vector with one in the i^{th} position and zeros elsewhere, let e_4 denote the vector with two in 4^{th} position and zeros elsewhere. Finally let $e_{8+i} = -e_i$ for $i = 1, 2, ..., 8$. Then $S = \{e_1, e_2, ..., e_{16}\}$, $\beta(e_1) = \beta(e_2) = ... = \beta(e_8) = 0$, $\beta(e_9) = \beta(e_{10}) = \beta(e_{15}) = \beta(e_{16}) = -4$, and $\beta(e_{11}) = \beta(e_{12}) = \beta(e_{13}) = \beta(e_{14}) = -3$. Finally the gain function is the sum of the flows out of s, i.e., $g = (1, 1, 0, 0, 0, 0, 0, 0)$.

1.3. Definition. For $x \in D_{(\mathbf{Z})}$ and $v \in V_{(\mathbf{Z})}$,

$$\alpha_{(\mathbf{Z})}(v \mid x) = \max\{\alpha\colon x + \alpha v \in D_{(\mathbf{Z})}\}\,.$$

The next four results follow at once from these definitions.

1.4. Proposition. *If $x, y \in D_{(\mathbf{Z})}$, then $y - x \in V_{(\mathbf{Z})}$.*

1.5. Proposition. *If $bv < 0$ for some $b \in S$, then*

$$\alpha(v \mid x) = \min\left\{\frac{\beta(b) - bx}{bv}: b \in S,\ bv < 0\right\};$$

otherwise $\alpha(v \mid x) = \infty$.

1.6. Proposition. *If $x \in D_{\mathbf{Z}}$, $v \in V_{\mathbf{Z}}$ and m is the greatest common divisor of the components of v, then*

$$\alpha_{\mathbf{Z}}(v \mid x) = \frac{1}{m}\lfloor m\alpha(v \mid x)\rfloor .$$

1.7. Proposition. *Let $x \in D_{(\mathbf{Z})}$ and $v \in V_{(\mathbf{Z})}$ (and let m again be the greatest common divisor of the components of v). Then for $\alpha \geqslant 0$, $x + \alpha v \in D_{(\mathbf{Z})}$ if and only if $\alpha \leqslant \alpha_{(\mathbf{Z})}(v \mid x)$ (and $m\alpha \in \mathbf{Z}$).*

2. Test sets and algorithms

2.1. Definition. An (*integral*) *test set* T for $x \in D_{(\mathbf{Z})}$ is a finite set of vectors, $T \subseteq V_{(\mathbf{Z})}$, such that:

(1) for all $t \in T$, $\alpha_{(\mathbf{Z})}(t \mid x) > 0$;

(2) for all $y \in D_{(\mathbf{Z})}$, $(y - x) = \Sigma_{t \in T}\alpha_t t$, where α_t is a nonnegative rational number (integer).

2.2. Proposition. *If T is an (integral) test set for $x \in D_{(\mathbf{Z})}$ and if for all $t \in T$ we have $gt \leqslant 0$, then x is (integral) optimal.*

Proof. Assume that T is a test set for $x \in D_{(\mathbf{Z})}$ and that $gt \leqslant 0$ whenever $t \in T$. Let $y \in D_{(\mathbf{Z})}$, then

$$y - x = \sum_{i=1}^{k} \alpha_i t_i \alpha_i t_i, \qquad g(y - x) = \sum_{i=1}^{k} \alpha_i (gt_i) .$$

However $\alpha_i > 0$ and $gt_i \leqslant 0$ for all i, then $g(y - x) \leqslant 0$ and $gy \leqslant gx$.

We have yet to prove that such test sets actually exist; this will be done in the next few sections. For the present, we will simply describe how these test sets are to be used.

2.3. Definition. An *(integral) method* consists of a non-empty subset $E \subseteq D_{(\mathbf{Z})}$ and a collection of sets of vectors $\{T_x: x \in E\}$ such that:
(1) T_x is an (integral) test set for x;
(2) for each $x \in E$ and each $t \in T_x$, $x + \alpha_{(\mathbf{Z})}(t \mid x) t \in E$ or $\alpha_{(\mathbf{Z})}(t \mid x) = \infty$.
The method is said to be *finite* if E is finite.

The actual process by which one searches through the test set T_x and decides the next move (in the case x is not optimal) is not discussed in this paper. For our purposes here we simply note that since T_x is finite, any search technique will take only a finite number of steps. We will define a (search) procedure by its effect.

2.4. Definition. A *procedure* for an (integral) method E, $\{T_x\}$ is a function

$$P: E \to E \cup \{\infty\}$$

so that

$$P(x) = \begin{cases} x + \alpha_{(\mathbf{Z})}(t \mid x)\, t \text{ for some } t \in T_x, \text{ where } gt > 0, \\ x \text{ if no such } t \text{ exists,} \end{cases}$$

where $x + \infty t$ is denoted simply by ∞.

2.5. Proposition. *Let P be a procedure for an (integral) method E, $\{T_x\}$. If $P(x) = x$, then x is (integral) optimal; if $P(x) \neq x$, then x is not (integral) optimal; if $P(x) = \infty$, then there is no optimal vector in $D_{(\mathbf{Z})}$.*

Proof. If $P(x) = x$, then $gt \leqslant 0$ whenever $\alpha_{(\mathbf{Z})}(t \mid x) > 0$. Thus by Proposition 2.2, x is (integral) optimal. If $P(x) \neq x$, then $P(x) = x + \alpha_{(\mathbf{Z})}(t \mid x)\, t$, where $gt > 0$ and $\alpha_{(\mathbf{Z})}(t \mid x) > 0$. Thus $gP(x) = gx + \alpha_{(\mathbf{Z})}(t \mid x)(gt) > gx$; it follows that x is not optimal. Finally, if for some t, $gt > 0$ and $\alpha_{(\mathbf{Z})}(t \mid x) = \infty$, then $x_i = x + it \in D_{(\mathbf{Z})}$ and $gx_i = gx + i(gt)$ approaches ∞ with i.

2.6. Definition. An (integral) method E, $\{T_x\}$ with procedure P is an *algorithm* if for each $x \in E$ there exists a positive integer k so that $P^k(x) = P^{k-1}(x)$ or ∞.

2.7. Proposition. *Any finite (integral) method E, $\{T_x\}$ along with any procedure P is an algorithm.*

Proof. Let $x \in D_{(\mathbf{Z})}$. If $P(x) \neq x$ or ∞, then, as we have seen above,

$gP(x) > gx$. Therefore, unless $P^k(x) = P^{k-1}(x)$ or ∞ for some k, we have a sequence of vectors in E: $x, P(x), P^2(x), \ldots$. Furthermore, since $gx < gP(x) < gP^2(x) < \ldots$, they would all have to be distinct. Since E is finite, this is impossible.

2.8. Proposition. *If g is bounded from above on $D_{\mathbf{Z}}$, then any integral method along with any procedure is an algorithm.*

Proof. Assume that g is bounded from above on $D_{\mathbf{Z}}$ and let E, $\{T_x\}$ be some integral method. We may assume without loss of generality that each vector in T_x is in lowest terms, i.e., its components have no common factors. It follows that $\alpha_{\mathbf{Z}}(t \mid x)$, for each such t, is an integer. Turning to g we observe that since its components are rational it is of the form $(1/m)h$ where $h \in \mathbf{Z}^n$. Thus if v is any integral vector, $gv = (1/m)hv$; and since hv is an integer it follows that $gv > 0$ implies that $gv \geqslant 1/m$. Now putting all of the observations together we may conclude that for each $t \in T_x$ either $gt \leqslant 0$ or $gt \geqslant 1/m$.

If P is a procedure and $P(x) \neq x$, then $gP(x) \geqslant gx + 1/m$, and if $x, P(x), P^2(x), \ldots, P^k(x)$ are all distinct, then $gP^k(x) \geqslant gx + k/m$. Since g is bounded from above on $D_{\mathbf{Z}}$, $P^k(x) = P^{k-1}(x)$ for some k.

Conditions under which g will be bounded from above on $D_{(\mathbf{Z})}$ will be discussed in the next section.

3. The universal test set

In this section, we will establish the existence of a test set. In fact, we will produce a single finite set $M \subseteq V$ which will contain a test set for each $x \in D$. To construct this set and to prove it is a test set, we must introduce two relations on the set V.

3.1. Definition. We say that v and w are *compatible* if $(bv)(bw) \geqslant 0$ for all $b \in S$.

We have a natural linear transformation from $\mathbf{Q}^n$ into $\mathbf{Q}^m$, where $S = \{b_1, b_2, \ldots, b_m\}$:

$$v \to v' = (b_1 v, b_2 v, \ldots, b_m v) .$$

v and w are compatible if v' and w' lie in the same "octant" of $\mathbf{Q}^m$. To visualize compatibility in $\mathbf{Q}^n$, let H_b be the hyperplane with equation $bx = \beta(b)$. $\{H_b: b \in S\}$ is the collection of boundary hyperplanes of D. Let $x \in D$, $v \in V$, and $b \in S$. We have $bx \geqslant \beta(b)$. If $bv \geqslant 0$ and $\alpha > 0$, then $x + \alpha v$ is no closer to H than x. In this case we say that v is directed away from or parallel to H_b. Similarly, if $bv \leqslant 0$, we say that v is directed toward or parallel to H_b. Now two vectors are compatible if their directions relative to the boundary hyperplanes are the same.

3.2. Proposition. *Compatibility is a symmetric, reflexive relation.*

3.3. Proposition. *If v is compatible with w and if $\alpha \geqslant 0$, then v is compatible with αw.*

3.4. Proposition. *If v is compatible with both u and w, then it is compatible with $u + w$.*

These propositions follow directly from the definition of compatibility. Under some conditions compatibility is transitive, but before we can show this we must introduce our second relation. This relation is derived from set inclusion by assigning a finite set to each vector.

3.5. Definition. For $v \in \mathbf{Q}^n$, $\operatorname{supp} v = \{b: b \in S,\ bv \neq 0\}$ is called the *support* of v.

The support of v is the set of $b_i \in S$ which correspond to non-zero components of $v' = (b_1 v, b_2 v, ..., b_m v)$. This definition also coincides with the usual notion of the support of a function if we think of v as mapping S into $\mathbf{Q}$ by $v(b) = vb$.

3.6. Proposition. *If u and v are compatible, v and w are compatible, and* $\operatorname{supp} u \subseteq \operatorname{supp} v$, *then u and w are compatible.*

Proof. If $b \in \operatorname{supp} u$, $(bu)(bv) > 0$ and $(bv)(bw) \geqslant 0$; thus $(bu)(bw)(bv)^2 \geqslant 0$. Since $b \in \operatorname{supp} u \subseteq \operatorname{supp} v$, $bv \neq 0$ and $(bv)^2 > 0$. Combined with the above this implies that $(bu)(bw) \geqslant 0$. If $b \notin \operatorname{supp} u$, $(bu)(bw) = 0$. Thus $(bu)(bw) \geqslant 0$ for all $b \in S$.

The next proposition treats one of the difficulties which arise if $S \cup \mathbf{R}$ fails to span $\mathbf{Q}^n$.

3.7. Proposition. (1) *$U = \{v: v \in V$ and $\operatorname{supp} v = \emptyset\}$ is a subspace of V.*

(2) *If $u \in U$, $x \in D_{(\mathbf{Z})}$, then $\alpha_{(\mathbf{Z})}(u \mid x) = \infty$.*

(3) *If $gu \neq 0$ for some $u \in U$, then there is no optimal vector in $D_{(\mathbf{Z})}$.*

Proof. (1) U is the solution space of $bu = 0$ for all $b \in R \cup S$.

(2) If $u \in U$, then $bu \geqslant 0$ for all $b \in S$, thus $\alpha_{(\mathbf{Z})}(u \mid x) = \infty$ for all $x \in D_{(\mathbf{Z})}$.

(3) Let $u \in U$ and let $gu \neq 0$. We may assume that $gu > 0$, otherwise we may replace u with $(-u)$; we may also assume that u is integral. Thus if $D_{(\mathbf{Z})}$ contains one point x it contains the sequence $x_i = x + iu$ and gx_i approaches ∞ with i.

Geometrically U is the subspace of $\mathbf{Q}^n$ orthogonal to the subspace spanned by $R \cup S$. Proposition 3.7 (3) can be restated as follows:

If g is not contained in the subspace spanned by $R \cup S$, then there is no optimal vector.

There are standard techniques for checking whether or not g is in the span of $R \cup S$, and we need not dwell on that problem. One way to make sure that g is in the span of $R \cup S$ is to assume that $R \cup S$ spans $\mathbf{Q}^n$. We will make this assumption and justify it by pointing out that one may easily add constraints to S and R which yield an equivalent problem with $U = \{0\}$.

Henceforth we will assume $U = \{0\}$ or equivalently: if $v \in V$ and $\operatorname{supp} v = \emptyset$, then $v = 0$.

3.8. Definition. $v \in V_{\mathbf{Z}}$ is said to be *minimal* if supp v is a minimal subset in $\{\operatorname{supp} w: w \in V, w \neq 0\}$ and if the components of v have no common factors. $M = \{v: v \text{ is minimal}\}$.

Under the assumption $U = \{0\}$ the linear transformation $v \to v' = (b_1 v, b_2 v, ..., b_m v)$ where $S = \{b_1, b_2, ..., b_m\}$ becomes an isomorphism when restricted to V. Let $V' = \{v': v \in V\}$ be the image of V under this linear isomorphism.

Given any subspace of $\mathbf{Q}^m$, it is natural to single out for special consideration those vectors which have a minimal collection of non-zero components. For example, if $m = 3$ and $\dim V' = 2$, these special vectors would lie on the lines of intersection of V' with the coordinate planes. All vectors (except 0) on the same line have the same support. In general, one can prove that two vectors in the subspace with the same minimal support lie on the same one-dimensional subspace. It seems natural then to single out one (or two) vectors for each minimal support. This is done in our case by insisting that the preimage under $'$ is integral and in "lowest terms".

The most important result concerning the minimal vectors in V' is that they span V' in a very special way: if $w' \in V'$, then w' is a positive linear combination of minimal elements each of which is compatible with w'. The important fact about such a spanning set is that there is no "cancelling out" in this summation, i.e., the j^{th} component of w' is of course the sum of the coefficients times the j^{th} components of the minimal vectors, but in this sum all of the terms have the same sign.

The sequence of definitions and results starting with Definition 3.8 are the lifting back to V of these rather natural theorems for V'.

3.9. Proposition. *If $v \in V$ and $w \in M$ so that* $\operatorname{supp} v = \operatorname{supp} w$, *then $v = \alpha w$. Furthermore, if $v \in M$, then $\alpha = \pm 1$.*

Proof. Since $0 \notin M$, the support of minimal vectors are not empty. Let $b \in \operatorname{supp} w = \operatorname{supp} v$, then bw and bv are both non-zero. Let $\alpha = bv/bw$ and consider $v - \alpha w$. This is a vector in V and $\operatorname{supp}(v - \alpha w) \subseteq \operatorname{supp} w$. However $b \notin \operatorname{supp}(v - \alpha w)$ since $b(v - \alpha w) = bv - \alpha(bw) = 0$. Therefore, $\operatorname{supp}(v - \alpha w) \subset \operatorname{supp} w$. But $w \in M$, therefore $v - \alpha w = 0$ and $v = \alpha w$. Finally, if $v \in M$ also, it, like w, must be integral and have no common factors among its components; it follows that $\alpha = \pm 1$.

3.10. Corollary. *M is finite.*

Proof. As we have seen above, each subset of S is the support of at most two minimal vectors. Since the number of subsets of S is finite, it follows that M is finite.

We wish to show now that for each $x \in D$, $M_x = \{v\colon v \in M, \alpha(v \mid x) > 0\}$ is a test set for that x. To this end we prove:

3.11. Lemma. *If $v, w \in V$, where* $\operatorname{supp} v \subseteq \operatorname{supp} w$ *and if* $S' = \{b \in S\colon (bv)(bw) > 0\} \neq \emptyset$, *then $w - \alpha v$, where* $\alpha = \min\{bw/bv\colon b \in S'\}$, *is compatible with w and* $\operatorname{supp}(w - \alpha v) \subset \operatorname{supp} w$.

Proof. Let α be as defined above and let $b \in S$, we have $(b(w - \alpha v))(bw) = (bw)^2 - \alpha(bv)(bw)$. Now if $b \notin S'$, $(bv)(bw) \leqslant 0$ and $(b(w - \alpha v))(bw) \geqslant$

$(bw)^2 \geqslant 0$. On the other hand, if $b \in S'$, then $\alpha \leqslant bw/bv$ and $\alpha(bv)(bw) \leqslant (bw)^2$; thus $(b(w - \alpha v))(bw) \geqslant 0$. Therefore w and $w - \alpha v$ are compatible.

Assume that $b \notin \operatorname{supp} w$, then $b \notin \operatorname{supp} v$ and $bv = bw = 0$; it follows that $b(w - \alpha v) = 0$, i.e., $b \notin \operatorname{supp}(w - \alpha v)$. Thus $\operatorname{supp}(w - \alpha v) \subseteq \operatorname{supp} w$. Finally, let $b \in S'$ be a vector such that $\alpha = bw/bv$. We have $b \in \operatorname{supp} w$, however $b(w - \alpha v) = 0$, i.e., $b \notin \operatorname{supp}(w - \alpha v)$. Therefore $\operatorname{supp}(w - \alpha v) \subset \operatorname{supp} w$.

3.12. Theorem. *If $w \in V$, then $w = \Sigma_{i=1}^k \alpha_i v_i$, where*

(1) *the v_i are distinct elements of M;*

(2) *each v_i is compatible with w;*

(3) $\operatorname{supp} v_i \subseteq \operatorname{supp} w$ *for each $i = 1, ..., k$;*

(4) *each α_i is positive.*

Proof. The proof will proceed by induction on $|\operatorname{supp} w|$. If $\operatorname{supp} w$ is minimal, the conclusion holds trivially. Now assume that $w \in V$ and that every vector in V with smaller support admits the above decomposition, we may further assume that $w \notin M$. Thus there exists $v \in V$ not 0, so that $\operatorname{supp} v \subset \operatorname{supp} w$. If $(bv)(bw) \leqslant 0$ for all $b \in S$, we may replace v by $-v$; thus we may assume that the hypotheses of Lemma 3.11 are satisfied. We have then: $v' = w - \alpha v$ with the property that v' is compatible with w and $\operatorname{supp} v' \subset \operatorname{supp} w$. Applying Lemma 3.11 again, we get $v'' = w - \alpha' v'$. Thus $w = v'' + \alpha' v'$, where $\operatorname{supp}(\alpha' v')$ and $\operatorname{supp} v''$ are proper subsets of $\operatorname{supp} w$, and we may apply the induction hypothesis to $\alpha' v'$ and v''. We have $\alpha' v' = \Sigma_{i=1}^k \alpha_i v_i$ satisfying conditions 1 through 4 above. Since $\operatorname{supp} v_i \subseteq \operatorname{supp} \alpha' v' \subseteq \operatorname{supp} w$, and v_i is compatible with $\alpha' v'$ while $\alpha' v'$ is compatible with w, we may conclude (Proposition 3.6) that v_i is compatible with w. The corresponding statements hold for the decomposition of v''. We may then add these two decompositions; collecting terms, to get the required decomposition for w.

3.13. Corollary. *For each $x \in D$, $M_x = \{v\colon v \in M, \alpha(v \mid x) > 0\}$ is a test set for x.*

Proof. Let $x, y \in D$, thus $y - x \in V$. We may apply Theorem 3.12 to $y - x$ to get $y - x = \Sigma_{i=1}^k \alpha_i v_i$. We have $v_i \in M$ and $\alpha_i > 0$, thus we need only prove that $\alpha(v_i \mid x) > 0$.

Let $b \in S$, if $bv_i < 0$ we must show that $\beta(b) - bx < 0$. However $bv_i < 0$ implies that $b \in \operatorname{supp} v_i$, it follows that $b \in \operatorname{supp}(y - x)$ and since v_i and $(y - x)$ are compatible, $b(y - x) < 0$. Now $\alpha(y - x \mid x) > 1$, hence $\beta(b) - bx < 0$ for this b.

3.14. Corollary. *If $x, y \in D$, then $y = x + \Sigma_{i=1}^{k} \alpha_i v_i$, where the v_i are distinct elements of M and $0 \leqslant \alpha_i \leqslant \alpha(v_i \mid x)$, i.e., $x + \alpha_i v_i \in D$ for all i.*

Proof. We apply Theorem 3.12 to $y - x$ to get $y - x = \Sigma_{i=1}^{k} \alpha_i v_i$, All that remains to be proved is that $\alpha_i \leqslant \alpha(v_i \mid x)$. Assume that for some j $\alpha_j > \alpha(v_j \mid x)$, then for some $b \in S$, $bv_j < 0$ and $\alpha_j > (\beta(b) - bx)/bv_j$. We have that $y - x$ and v_i are compatible for all i, thus we may conclude that $|b(y-x)| = \Sigma_{i=1}^{k} \alpha_i |bv_i|$. This implies $|b(y-x)| \geqslant \alpha_j |bv_j|$ or $b(y-x) \leqslant \alpha_j bv_j < \beta(b) - bx$; which in turn implies $(\beta(b) - bx)/b(y-x) < 1$. But $(\beta(b) - bx)/b(y - x) \geqslant \alpha(y - x \mid x) \geqslant 1$.

We may now settle one question left open in Section 2.

3.15. Proposition. *Assuming $D_{(\mathbf{Z})} \neq \emptyset$, g is bounded from above on $D_{(\mathbf{Z})}$ if and only if there exists no $v \in M$ such that $gv > 0$ and $bv \geqslant 0$ for all $b \in S$.*

Proof. Assume that $gv > 0$ and that $bv \geqslant 0$ for all $b \in S$. We have then that $\alpha(v \mid x) = \infty$ for all $x \in D_{(\mathbf{Z})}$. Let $x \in D_{(\mathbf{Z})}$ and let $x_i = x + iv$, then gx_i approaches ∞ with i.

Conversely, assume that for any $v \in M$ either $bv < 0$ for some $b \in S$ or $gv \leqslant 0$. Let $x \in D_{(\mathbf{Z})}$ and define $m = gx + \Sigma_{gv>0}\, \alpha(v \mid x)\,(gv)$. Let $y \in D_{(\mathbf{Z})}$ and apply Corollary 3.14 to get $y = x + \Sigma_{i=1}^{k} \alpha_i v_i$; it follows that $gy = gx + \Sigma_{i=1}^{k} \alpha_i g v_i \leqslant m$.

At this point let us consider the geometrical significance of the vectors in M. $\{S - \operatorname{supp} v\}$ is the collection of vectors in S which are orthogonal to v. The geometric interpretation of Proposition 3.9 is that the subspace of V orthogonal to $\{S - \operatorname{supp} v\}$ is the one-dimensional subspace of V spanned by v. $\{x\colon \beta(b) = bx$ for $b \in R \cup \{S - \operatorname{supp} v\}\}$ is then a line in $\mathbf{Q}^n$ parallel to the subspace spanned by v. In fact it is the intersection of the boundary hyperplanes $\{(x\colon \beta(b) = bx)\}$ for $b \in R \cup \{S - \operatorname{supp} v\}$. M then represents the set of one-dimensional subspaces parallel to the lines obtained by intersecting the hyperplanes $\{(x\colon \beta(b) = bx)\}$ for $b \in R \cup S$. The edges of D are subsets of these lines and thus M includes vectors which represent the directions of the edges of D.

Before concluding this section, let us turn to our example. In Table 3.16

we list the minimal vectors for this problem. As in a usual flow problem, the minimal vectors may be easily identified since supp $v = \{e_i, e_{8+i}$: the i^{th} component of v is not zero$\}$. Thus the minimal vectors correspond to elementary paths in N from s to s' or to elementary circuits. These latter minimal vectors have zero gain and hence could be ignored; however they will be of importance in the next section.

3.16. Table. M consists of the following 13 vectors and their negatives.

$m_1 = (1, 0, 1, 0, 0, 0, 1, 0)$, $m_8 = (0, 2, -2, 1, 2, 0, 0, 2)$,
$m_2 = (1, 0, 1, 0, -1, 1, 0, 1)$, $m_9 = (0, 1, 0, 0, 0, 1, 0, 1)$,
$m_3 = (2, 0, 0, 1, 0, 0, 0, 2)$, $m_{10} = (0, 2, 2, -1, 0, 2, 2, 0)$,
$m_4 = (2, 0, 0, 1, 2, -2, 2, 0)$, $m_{11} = (0, 0, 2, -1, -2, 2, 0, 0)$,
$m_5 = (1, -1, 1, 0, -1, 0, 0, 0)$, $m_{12} = (0, 0, 2, -1, 0, 0, 2, -2)$,
$m_6 = (2, -2, 0, 1, 0, -2, 0, 0)$, $m_{13} = (0, 0, 0, 0, 1, -1, 1, -1)$.
$m_7 = (0, 1, 0, 0, 1, 0, 1, 0)$,

To illustrate the use of this test set we will take as our procedure a simple search through M_x in the given order.

We start with the zero flow $x_0 = (0, 0, 0, 0, 0, 0, 0, 0)$. We note that $e_i m_1 = 1$ for $i = 1, 3, 7$, $e_i m_1 = -1$ for $i = 9, 11, 15$; and $e_i m_1 = 0$ for all other values of i. Thus

$$\alpha(m_1 \mid x) = \min \left\{ \frac{\beta(e_i) - 0}{e_i m_1} : i = 9, 11, 15 \right\} = 3 .$$

We conclude that $m_1 \in M_{x_0}$. Since $e_5 m_2 = -1$ and $(\beta(e_5) - e_5 x)/e_5 m_2 = 0$, $\alpha(m_2 \mid x_0) = 0$; hence $m_2 \notin M_{x_0}$. Continuing with these computations we see that $M_{x_0} = \{m_1, m_3, m_7, m_9\}$. In practice, one would not bother to compute all of M_{x_0}. Since $gm_1 = 1$, our procedure uses m_1 in the improvement step. We have $x_1 = P(x_0) = (3, 0, 3, 0, 0, 0, 3, 0)$. Now $M_{x_1} = \{-m_1, m_3, m_7, m_8, m_9\}$ and $x_2 = P(x_1) = (4, 0, 3, \frac{1}{2}, 0, 0, 3, 1)$. Then $M_{x_2} = \{-m_1, -m_3, -m_5, m_7, m_8, m_9, m_{10}, m_{12}\}$. Since $g(-m_1) = g(-m_3) = -1$, $g(-m_5) = 0$, and $g(m_7) = 1$, $x_3 = P(x_2) = (4, 1, 3, \frac{1}{2}, 1, 0, 4, 1)$. Finally, $x_4 = P(x_3) = (4, 4, 3, \frac{1}{2}, 1, 3, 4, 4)$ is a maximal flow. That x_4 is optimal can be seen by computing M_{x_4} and showing $gm \leqslant 0$ for all $m \in M_{x_4}$.

We observe that x_4 is not an integral flow. The difficulty arose in the second step by including $\frac{1}{2}m_3$. Here we had $\alpha(m_3 \mid x_2) = \frac{1}{2}$ while $\alpha_Z(m_3 \mid x_2) = 0$. However, if we had bypassed m_3 here and only made integral improvements, we would have obtained $x_2 = (3, 1, 3, 0, 1, 0, 4, 0)$ and then $x_3 = (3, 4, 3, 0, 1, 3, 4, 3)$. We have $M_{x_3} = \{-m_1, -m_2, m_3, m_6, -m_7, -m_9, -m_{10}, -m_{11}, -m_{12}\}$, but since either $\alpha_Z(m \mid x_3) = 0$ or $gm \leqslant 0$ for each $m \in M_{x_3}$, we can proceed no further. However x_3 is not our optimal integral flow. This illustrates that M_x is not in general an integral test set. Extending M to a universal integral test set is the object of the next section.

4. The universal integral test set

We wish to find the "smallest" collection of integral vectors in V_Z so that every $w \in V_Z$ is a positive *integral* linear combination of vectors from this collection (each of which is compatible with w). As we have seen, M satisfies all of these conditions except that the coefficients may not be integral. Thus it is not surprising that such a collection is generally larger than M. In fact it is not clear at this point that such a collection is finite.

We must introduce still another relation on V_Z.

4.1. Definition. If $v, w \in V_Z$, we say that v *majorizes* w if v and w are compatible and $|bv| \geqslant |bw|$ for all $b \in S$. We write $v \geqslant w$.

Assume that $w = \Sigma_{i=1}^{k} \alpha_i v_i$ (where for all i, α_i is a positive integer, $\operatorname{supp} v_i \subseteq \operatorname{supp} w$, v_i is compatible with w, $v_i \in V_Z$, $v_i \neq 0$). Then w majorizes v_i for all i. Clearly then, if $w \in V_Z$ and majorizes only w and 0, then it has only a trivial integral decomposition as above. Hence it must belong to the universal integral test set. It turns out that vectors of this type are the only ones needed.

4.2. Proposition. (1) *If* $v \geqslant w$, *then* $\operatorname{supp} w \subseteq \operatorname{supp} v$.
(2) $\geqslant$ *is transitive and reflexive.*

Proof. Assume that $v \geqslant w$. If $b \notin \operatorname{supp} v$, $bv = 0$. But $|bv| \geqslant |bw|$. Thus $bw = 0$ and $b \notin \operatorname{supp} w$. Now assume that $v \geqslant w$, $w \geqslant u$. We have u compatible with w, w compatible with v and $\operatorname{supp} u \subseteq \operatorname{supp} w \subseteq \operatorname{supp} v$; thus by Proposition 3.6, u is compatible with v. Finally $|bv| \geqslant |bw| \geqslant |bu|$ for all $b \in S$.

4.3. Definition. We say that $v \in V_{\mathbf{Z}}$ is *indecomposable* if $v \geqslant w$ and $w \in V_{\mathbf{Z}}$ implies that $w = v$ or $w = 0$. $I = \{v\colon v \text{ is indecomposable}\}$.

As with M, we may get a geometric interpretation of I in V'. Let $V'_{\mathbf{Z}}$ be the image of $V_{\mathbf{Z}}$ under $'$ (the map $v \to v'$ is defined just after Definition 3.8). Since each b in S is integral, bv is integral wherever $v \in V_{\mathbf{Z}}$. Hence $V'_{\mathbf{Z}}$ is a sublattice of the lattice of integer vectors in V'. Let us fix some "octant" of $\mathbf{Q}^m$ – for the sake of simplicity let us take the first "octant", i.e., the set of vectors with non-negative components, and let us denote it by O_1. Then V' intersects this octant, O_1, in a cone. The image of M in this cone will generate the cone; the image of I in this cone will generate the sublattice $V'_{\mathbf{Z}}$ under addition only. Assume $\dim V' = 2$, we may then draw a picture of the intersection of V' with the first octant of $\mathbf{Q}^m$ (see Fig. 2). The heavy lines represent the intersections of the plane V' with the coordinate hyperplanes bounding O_1. The upper of the four cones defined by these lines represents $O_1 \cap V'$. The lattice points of $\mathbf{Q}^m$ which lie in V' are indicated by the intersections of the grid lines. The vectors in $V'_{\mathbf{Z}}$ are indicated by an asterisk, the origin is denoted by O. The vectors v_0, v_1, v_2, v_3 and v_4 are the indecomposable

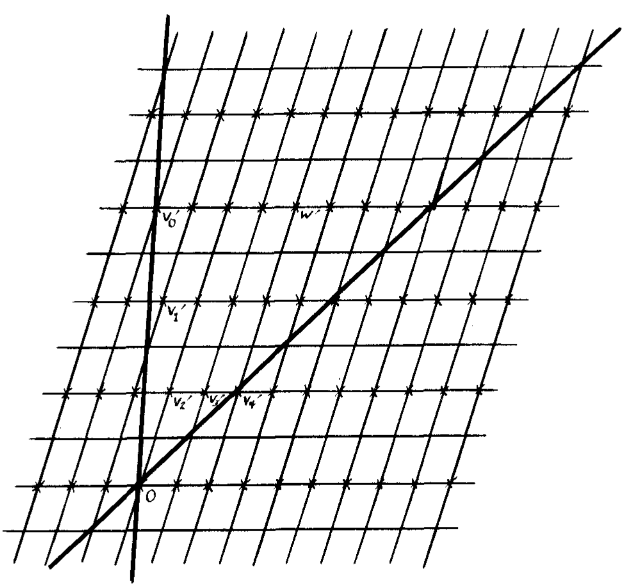

Fig. 2.

vectors whose images v_0', v_1', v_2', v_3' and v_4' lie in O_1, however only v_0 and v_4 are minimal. One can easily see that each vector in $O_1 \cap V'$ can be written as a linear combination of v_0' and v_4' with non-negative rational coefficients. For example: $w' = \frac{1}{2}v_0' + \frac{3}{2}v_4'$. This example also illustrates that the coefficients may indeed be non-integral, even for vectors in $V_{\mathbf{Z}}'$. It is also not difficult to see that every vector in $O_1 \cap V_{\mathbf{Z}}'$ can be written as a linear combination of $v_0', v_1', ..., v_4'$, with non-negative *integral* coefficients. For example: $w' = v_2' + 2v_3'$.

4.4. Proposition. *$M \subseteq I$ and I is finite.*

Proof. Let $v \in M$ and assume that $v \geqslant w$, where $w \in V_{\mathbf{Z}}$. We have then $\operatorname{supp} w \subseteq \operatorname{supp} v$. If $\operatorname{supp} w = \emptyset$, $w = 0$ if $\operatorname{supp} w = \operatorname{supp} v$, $w = \alpha v$. In this case $v \geqslant w$ implies $0 < \alpha \leqslant 1$, but both v and w are integral and the components of v have no common divisors; thus $\alpha = 1$ and $w = v$.

Now let $\epsilon_b = \Sigma_{v \in M} |bv|$ and let $w \in I$. By Theorem 3.12, $w = \Sigma_{i=1}^{k} \alpha_i v_i$. If $\alpha_i > 1$, then $w \geqslant v_i \geqslant 0$ and $w \neq v_i$ which contradicts the indecomposability of w, thus $\alpha \leqslant 1$ for $i = 1, ..., k$. Further,

$$|bw| \leqslant \sum_{i=1}^{k} \alpha_i \, |bv_i| \leqslant \sum_{i=1}^{k} |bv_i| \leqslant \epsilon_b \; .$$

Finally, since $U = \{0\}$, the solution set to:

$$bw = 0 \quad \text{for all } b \in R,$$

$$-\epsilon_b \leqslant bw \leqslant \epsilon_b \quad \text{for all } b \in S,\ w \text{ is integral}\,,$$

is bounded and hence consists of a finite number of integral vectors.

4.5. Theorem. *If $w \in V_{\mathbf{Z}}$, we can write $w = \Sigma_{i=1}^{k} \alpha_i v_i$, where*
(1) *the v_i are distinct elements of I;*
(2) *each v_i is majorized by w;*
(3) *each α_i is a positive integer.*

Proof. The proof will proceed by induction on $\Sigma_{b \in S} |bw|$. If $w \in I$, the conclusions hold trivially, thus we may assume that $v \in V_{\mathbf{Z}}$, $v \neq 0$ or w and $w \geqslant v$. From this we may conclude that $w \geqslant w - v$, where $w - v \neq 0$ or w. It is also easy to see that $\Sigma_{b \in S} |bw| = \Sigma_{b \in S} |bv| + \Sigma_{b \in S} |b(w - v)|$. We may then apply the induction hypothesis to both v and $(w - v)$. Combining the two decompositions and collecting terms we get the required decomposition of w.

4.6. Corollary. *For each* $x \in D_{\mathbf{Z}}$, $I_x = \{v\colon v \in I$ *and* $\alpha_{\mathbf{Z}}(v \mid x) > 0\}$ *is an integral test set for* x.

Proof. Let $x, y \in D_{\mathbf{Z}}$, thus $y - x \in V_{\mathbf{Z}}$. We may apply Theorem 4.5 to $y - x$ to get $y - x = \Sigma_{i=1}^{k} \alpha_i v_i$. We have $v_i \in I$ and α_i a positive integer, thus we need only prove that $\alpha_{\mathbf{Z}}(v_i \mid x) > 0$. Since the components of v_i have no common factors, this is equivalent to proving that $\alpha(v_i \mid x) \geqslant 1$.

Let $b \in S$, if $bv_i < 0$ we must show $(\beta(b) - bx)/bv_i \geqslant 1$. However $(y - x) \geqslant v_i$, thus $|b(y - x)| \geqslant |bv_i|$ from which we conclude $b(y - x) \leqslant bv_i$. Now $\alpha(y - x \mid x) \geqslant 1$, thus $(\beta(b) - bx)/b(y - x) \geqslant 1$. We get $\beta(b) - bx \leqslant b(y - x) \leqslant bv_i$ which implies $(\beta(b) - bx)/bv_i \geqslant 1$.

We now return to our example. Since each indecomposable vector must be the sum of minimal vectors with rational coefficients strictly between 0 and 1, and since it may be arranged so that none of these minimal elements has its support as a proper subset of the union of the supports of the rest, we conclude that indecomposable vectors are sums of $\{m_3, m_4, m_6, m_8, m_{10}, m_{11}, m_{12}\}$ with coefficients 0, $\frac{1}{2}$, or $-\frac{1}{2}$. The indecomposable vectors are listed in Table 4.7.

4.7. Table. I consists of M plus the following 15 vectors and their negatives:

$$
\begin{aligned}
i_1 &= (2, 0, 0, 1, 1, -1, 1, 1) = \tfrac{1}{2}(m_3 + m_4),\\
i_2 &= (2, -1, 0, 1, 0, -1, 0, 1) = \tfrac{1}{2}(m_3 + m_6),\\
i_3 &= (1, 1, -1, 1, 1, 0, 0, 2) = \tfrac{1}{2}(m_3 + m_8),\\
i_4 &= (1, -1, -1, 1, 0, -1, -1, 1) = \tfrac{1}{2}(m_3 - m_{10}) = \tfrac{1}{2}(m_6 - m_{12}),\\
i_5 &= (1, 0, -1, 1, 1, -1, 0, 1) = \tfrac{1}{2}(m_3 - m_{11}) = \tfrac{1}{2}(m_4 - m_{12}) = \tfrac{1}{2}(m_6 + m_8),\\
i_6 &= (1, 0, -1, 1, 0, 0, -1, 2) = \tfrac{1}{2}(m_3 - m_{12}),\\
i_7 &= (2, -1, 0, 1, 1, -2, 1, 0) = \tfrac{1}{2}(m_4 + m_6),\\
i_8 &= (1, -1, -1, 1, 1, -2, 0, 0) = \tfrac{1}{2}(m_4 - m_{10}) = \tfrac{1}{2}(m_6 - m_{11}),\\
i_9 &= (1, 0, -1, 1, 2, -2, 1, 0) = \tfrac{1}{2}(m_4 - m_{11}),\\
i_{10} &= (1, -2, -1, 1, 0, -2, -1, 0) = \tfrac{1}{2}(m_6 - m_{10}),\\
i_{11} &= (0, 0, -2, 1, 1, -1, -1, 1) = \tfrac{1}{2}(m_8 - m_{10}) = -\tfrac{1}{2}(m_{11} + m_{12}),\\
i_{12} &= (0, 1, -2, 1, 2, -1, 0, 1) = \tfrac{1}{2}(m_8 - m_{11}),
\end{aligned}
$$

$$i_{13} = (0, \ \ 1, -2, 1, 1, \ \ 0, -1, 2) = \tfrac{1}{2}(m_8 - m_{12}),$$
$$i_{14} = (0, -1, -2, 1, 1, -2, -1, 0) = \tfrac{1}{2}(m_{10} + m_{11}),$$
$$i_{15} = (0, -1, -2, 1, 0, -1, -2, 1) = \tfrac{1}{2}(m_{10} + m_{12}) .$$

Using this integral test set we may now solve our example problem. Starting as before with $x_0 = (0, 0, 0, 0, 0, 0, 0, 0)$, we have $I_{x_0} = \{m_1, m_3, m_7, m_9\}$. Hence as before $x_1 = P(x_0) = (3, 0, 3, 0, 0, 0, 3, 0)$. We then have

$$I_{x_1} = \{-m_1, m_7, m_8, m_9, i_3, i_6, i_{13}\}, \quad x_2 = P(x_1) = (3, 1, 3, 0, 1, 0, 4, 0) .$$

In our systematic search we find that m_9 is the first vector in I_{x_2} with positive flow; thus $x_3 = P(x_2) = (3, 4, 3, 0, 1, 3, 4, 3)$. As indicated in Section 3, we could get this far using only minimal vectors – but no further. The first vector in I_{x_3} with positive flow is i_5. Hence $x_4 = (4, 4, 2, 1, 2, 2, 4, 4)$. This is optimal since no vector in I_{x_4} has a positive flow.

Let us step back from all of these computations and look at our problem again. If one were asked to describe an algorithm for this problem, he would probably come up with the following obvious algorithm. First, search for unsaturated paths from the source to the sink which if they contain the 4^{th} edge can sustain an additional flow of 2 units. When no further paths of this type can be found, search for pairs of unsaturated paths or one unsaturated path and one unsaturated circuit both passing through the 4^{th} edge. Sending one unit along each of these paths will then send two units along the 4^{th} edge. Steps of the first kind correspond to minimal vectors while steps of the second kind correspond to non-minimal indecomposable vectors. It follows then from Corollary 4.6 and Proposition 2.8 that this natural technique is an algorithm, i.e., it converges to an optimal flow in a finite number of steps. No separate proof based on a duality theory or other ad hoc technique is necessary. For an example of more practical applications of this theorem, see [6].

5. Conclusions and comments

Perhaps we should take a few moments to relate this development and the usual network flow problem. In this case, we can easily compute M. M consists of the unit flows along elementary paths between the source and the sink, or around elementary circuits. An important feature which distinguishes the integer network flow problem from the general integer linear program is that in the former case $I = M$, while in the latter case I is often much larger than M. One may view the size of I relative to the size of M as a rough measure of the complexity introduced into a linear programming problem by the inclusion of the condition that the solution vector be integral.

In the usual solution of the network flow problem $M = I$ is never computed. If x is a feasible integral flow, $P(x)$ is obtained by constructing an element of I_x which has positive inner product with g (an unsaturated path from the source to the sink) or by showing that no such element exists. At each step the construction is started over from scratch. With this technique, we are finding out the important facts about the test set I_x (constructing an element of I_x with positive inner product or showing that no such element exists) without ever constructing I_x. In the network flow problem, this can be done because of the very simple nature of g and of the vectors in R and S. It is anticipated that one will be able to generalize this technique. Indeed, finding efficient methods for generating M, I, M_x, I_x and appropriate subsets of these, is a very important area for further research.

Equally important is the investigation of smaller test sets for both the linear and integral linear cases. Smaller test sets is the subject of a second paper to appear soon. In it, the techniques developed here are first applied to a study of the Simplex Algorithm and then to generalizations of the Simplex Algorithm.

Finally a few comments about the motivation behind this paper are in order. After studying the beautiful theory on network flows due to Ford and Fulkerson [1], it is natural to try to extend their techniques to arbitrary integer linear programming. The essential ideas needed to make this extension arose in a series of two papers by the author with W.B. Jurkat [3], [4]. In these papers, combinatorial designs were considered as elements in various algebraic structures, in fact as the domains of a specific class of integer linear programming problems. The simplest of these problems, the maximum depth problem, is the subject of a survey paper by the author [2]. Special cases of Theorem 3.12 and Proposition 4.4 are included in this paper. This special case of Proposition 4.4 is due to Huckermann and Jurkat [5]. The survey paper also contains references to the recurrence of this problem in game theory and in other branches of combinatorics.

In the special case that S consists of the standard basis vectors for $\mathbf{Q}^n$, as it does in the case of network flows, the support function is the one employed by Tutte [7] in constructing a matroid from a chain group. It is not difficult to see that even in the case of an arbitrary S, the supports of the minimal vectors form the circuits of a matroid on S. The matroid so defined does not capture all of the structure of the minimal elements themselves. Therefore matroid arguments do not enter into the above discussions. However, this underlying matroid does play an important part in its motivation.

References

[1] L.R. Ford, Jr. and D.R. Fulkerson, *Flows in networks* (Princeton University Press).
[2] J.E. Graver, "Maximum depth problem for indecomposable exact covers", Proceedings of the International Colloquium on Infinite and Finite Sets, Keszthely, Hungary (1973).
[3] J.E. Graver and W.B. Jurkat, "Algebra structures for general designs", *Journal of Algebra* 23 (1972) 574–589.
[4] J.E. Graver and W.B. Jurkat, "The module structure of integral designs", *Journal of Combinatorial Theory* 15 (1973) 75–90.
[5] F. Huckermann and W.B. Jurkat, "Finiteness theorems for coverings and semi-groups", to appear.
[6] C.J. Leska, "Covering and matching problems as integer linear programs", to appear.
[7] W.T. Tutte, "Lectures on Matroids", *Journal of Research of the National Bureau of Standards* 69B (1965) 1–49.