# MATROID INTERSECTION ALGORITHMS

by

Eugene L. Lawler

Memorandum No. ERL-M333

October 11, 1971

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

# MATROID INTERSECTION ALGORITHMS

Eugene L. Lawler

Department of Electrical Engineering and Computer Sciences
and the Electronics Research Laboratory
University of California, Berkeley, California 94720

ABSTRACT

Three algorithms for computing optimal matroid intersections are presented: an algorithm for computing an intersection of maximum cardinality, a primal-dual algorithm, and a primal algorithm for computing an intersection of maximum total weight, given a weighting of the elements of the matroids. Each of the algorithms is shown to be computationally efficient, in the sense that the number of computational steps is bounded by a polynomial function of the number of elements, provided there exists a subroutine for testing independence of a given subset of elements which is efficient in the same sense.

Research sponsored by the Air Force Office of Scientific Research, Grant AFOSR-71-2076.

# I. INTRODUCTION

## 1. Matroid Intersection Algorithms

Matroids are combinatorial structures which abstract the notion of linear independence. A set that is "independent" in each of two matroids we call an "intersection" of the two matroids.

The purpose of this paper is to present algorithms for computing optimal matroid intersections. Part I contains definitions and background material. Part II presents an algorithm for computing an intersection that is of maximum (finite) cardinality. Part III presents an algorithm for computing an intersection that is of maximum total weight, given a weighting of the elements of the matroids. This algorithm is of the "primal-dual" variety, employing both primal and dual linear programming variables. In Part IV a simpler "primal" algorithm is presented, in which the dual variables are dispensed with.

Each of the algorithms is computationally efficient, in that the number of computational steps required is bounded by a polynomial function of the number of elements of the matroids, provided there exists a subroutine for testing independence of a given subset of elements which is efficient in the same sense.

## 2. Definitions

A matroid $M = (E, \mathcal{I})$ is a structure in which E is a finite set of elements and $\mathcal{I}$ is a nonempty family of subsets of E (called independent sets) satisfying the axioms:

(2.1) If $I \in \mathcal{I}$ and $I' \subseteq I$, then $I' \in \mathcal{I}$.

(2.2) If $I_p$ and $I_{p+1}$ are sets in $\mathcal{I}$ containing respectively p and p+1 elements, then there exists an element $e \in I_{p+1}-I_p$ such that $I_p+e \in \mathcal{I}$.

(We use I + e and I - e to denote $I \cup \{e\}$ and $I - \{e\}$ respectively. We also denote the symmetric difference of two sets by "$\oplus$", and the number of elements in I by $|I|$.)

A set which is not independent (i.e. not in the family I) is said to be dependent. A minimal dependent set is called a circuit. It is a basic theorem of matroid theory that if I is independent and I + e is dependent, then I + e contains precisely one circuit. If a subroutine exists for testing for independence, then the unique circuit in I + e can be discovered by removing one element at a time from I + e and testing for independence. If the removal of an element produces independence, the element is returned to the set. The subset remaining at the end is the unique circuit.

Let A be an arbitrary subset of E. The rank of A, denoted r(A), is the number of elements in a maximal independent subset $I \subseteq A$. (It is a basic theorem of matroid theory that all such maximal subsets have the same cardinality.) The span of A, denoted sp(A), is the (unique) maximal superset of A having the same rank as A. Intuitively, the span of A is the set composed of A joined with all elements e which form circuits with subsets of A. Clearly, if A is independent and a subroutine exists for testing for indeoendence, it is possible to compute the span of A by testing A + e, for all $e \notin A$. It is assumed in the statement of the algorithms that subroutines are available to test a given set for independence.

## 3. Examples of Matroids

Let E be the columns of an $m \times n$ matrix C, and let $\mathcal{J}$ be the family of linearly independent subsets of columns. The matroid $M = (E, \mathcal{J})$ is said to be the matroid of the matrix C; such a matroid is said to be matric.

Let E be the set of arcs of a linear graph G, and let $\mathcal{J}$ be the family of subsets of arcs which contain no cycles, i.e., subsets which comprise trees or "forests." The matroid $M = (E, \mathcal{J})$ is said to be the matroid of the graph G; such a matroid is said to be graphic. Every graphic matroid is matric, as can be seen by considering the node-arc incidence matrix with arithemetic in the field of two elements.

Let $Q = \{q_i; i = 1, \ldots, m\}$ be a family of (not necessarily distinct) subsets of a set $E = \{e_j; j = 1, \ldots, n\}$. The set $T = \{e_{j(1)}, \ldots, e_{j(t)}\}$, $0 \leq t \leq n$ is called a partial transversal of Q if T consists of distinct elements in E and if there are distinct integers $i(1), \ldots, i(t)$ such that $e_{j(k)} \in q_{i(k)}$ for $k = 1, \ldots, t$. The set is called a transversal or a system of distinct representatives (SDR) of Q if $t = m$.

Now if $\mathcal{J}$ is the family of partial transversals of Q, then $M_a = (E, \mathcal{J})$ is a matroid. Alternatively, if $\mathcal{J}$ is the collection of subfamilies of Q that have transversals, then $M_b = (Q, \mathcal{J})$ is a matroid. As Edmonds and Fulkerson [5] have commented, $M_a$ and $M_b$ both belong to the same abstract class of matroids, because the roles of Q and E are symmetric. Either type of matroid is called a transversal matroid.

Let $P = \{p_i; i = 1, \ldots, m\}$ be a partition of the set E into m blocks or equivalence classes. Let $d_1, d_2, \ldots, d_m$ be given nonnegative integers.

Let $\mathcal{I}$ be the family of all subsets I of E such that

$$|I \cap p_i| \le d_i, \qquad (i = 1, 2, \ldots, m).$$

Then $M = (E, \mathcal{I})$ is a partition matroid. Ordinarily, we assume that $d_1 = d_2 = \ldots = d_m = 1$. Every partition matroid is a transversal matroid.

## 4. Examples of Matroid Intersections

Let $M_1 = (E, \mathcal{I}_1)$ and $M_2 = (E, \mathcal{I}_2)$ be two given matroids. A subset $I \in I_1 \cap I_2$ is said to be an intersection of $M_1$ and $M_2$. Below we give some examples of matroid intersections.

Let C be an $m \times n$ matrix. Suppose we draw a horizontal line through C so that there are $m_1$ rows above the line and $m_2$ below. We can speak of a subset of the columns as being linearly independent both "above the line" and "below the line." In other words, the appropriate projections of those columns as vectors are independent in an "upper" $m_1$-dimensional space, and also in a "lower" $m_2$-dimensional space. Any such subset of columns is a matroid intersection.

Suppose two graphs $G_1$ and $G_2$ are assembled from the same set of arcs E. A subset $I \subseteq E$ is a matroid intersection if it is cycle-free in both $G_1$ and $G_2$.

Let G be a bipartite graph in which each arc extends between a node in a set S and a node in a set T. A matching in G is a subset of the edges, no two of which meet at the same vertex. Let $M_1$ be a partition matroid which has as its independent sets all subsets of arcs, no two of which meet at the same node of S. Let $M_2$ be a partition matroid which has as its independent sets all subsets of arcs, no two of which meet at the same node of T. Every matching is an intersection of matroids $M_1$ and $M_2$, and vice versa.

Of course, the two matroids of an intersection problem do not have to be the same type. For example, let G be a directed graph. Let $M_1$ be the graphic matroid of G (for which the orientations of the arcs are irrelevant). Let $M_2$ be a partition matroid which has as its independent sets all subsets of arcs, no two of which are directed into the same node. An intersection of these two matroids is a union of directed trees rooted from a point. Edmonds [2] has presented a particularly simple and elegant algorithm for finding a maximum-weight rooted tree in a given weighted directed graph.

In addition to the matroids $M_1$ and $M_2$ above, we can define a matroid $M_3$ which has as its independent sets all subsets of arcs, no two of which are directed <u>out</u> of the same node. One can then contemplate the problem of finding a maximum weight set of arcs, subject to the restriction that the arcs are independent in all three matroids. Unfortunately, the methods described in this paper are incapable of coping with the problem of finding optimal intersections of three or more matroids; if they were, one could solve the traveling salesman problem, which can be shown to be equivalent to finding a maximum weight intersection of $M_1$, $M_2$ and $M_3$.

## II. CARDINALITY INTERSECTION ALGORITHM

### 5. Augmenting Sequences

The proposed algorithm starts with any intersection (e.g. the null set) and proceeds to augment it through the mechanism of "augmenting sequences" until an intersection of maximum cardinality is attained. This procedure is exactly analogous to that used for solving maximal network flow problems, assignment problems, etc. The concept of an augmenting sequence corresponds exactly to that of a flow augmenting path.

Let $M_1 = (E, \mathcal{I}_1)$, $M_2 = (E, \mathcal{I}_2)$ be two matroids, and let $sp^1$, $sp^2$ denote spans in $M_1$, $M_2$. Let I be an <u>intersection</u> of $M_1$, $M_2$. Let $S = (e_1, e_2, \ldots, e_m)$, where m is odd, be a sequence of distinct elements in which $e_i \in E - I$ for i odd and $e_i \in I$ for i even. Let $S_i = \{e_1, e_2, \ldots, e_i\}$. S is said to be an <u>augmenting sequence</u> with respect to I if

(1) $I \oplus S_1 \in \mathcal{I}_1$

(2) $sp^1(I \oplus S_i) = sp^1(I \oplus S_1)$, for all odd i.

(3) $sp^2(I \oplus S_i) = sp^2(I)$, for all $i < m$.

(4) $I \oplus S_m \in \mathcal{I}_2$.

It follows that $I \oplus S_i \in \mathcal{I}_1$ for all i, $I \oplus S_i \in \mathcal{I}_2$ for all even i, and $I \oplus S_i$ contains a unique $M_2$-circuit for all odd $i < m$. Most importantly, $I \oplus S$ is an intersection with one more element than I.

Theorem 5.1

Let $I_k$, $I_{k+1}$ be intersections with k, k+1 elements respectively. Then there exists an augmenting sequences $S \subseteq I_k \oplus I_{k+1}$ with respect to $I_k$.

Proof:

Proof is by induction on the number of elements in $I_k \oplus I_{k+1}$. Clearly $S = I_k \oplus I_{k+1}$ is an augmenting sequence if $|I_k \oplus I_{k+1}| = 1$.

Now suppose $|I_k \oplus I_{k+1}| = p > 1$. By matroid axiom (2.2), there must exist some element $e_1 \in I_{k+1} - I_k$ such that $I_k \oplus e_1 \in \mathcal{J}_1$. If $I_k + e_1$ is not independent in $M_2$, there exists a unique $M_2$-circuit $C \subseteq I_k + e_1$. Let $e_2$ be any element belonging to $C - e_1$. $I_K' = I_k + e_1 - e_2$ is an intersection, and $|I_k' \oplus I_{k+1}| = p-2$. By inductive hypothesis, there exists an augmenting sequence $S' \subseteq I_k' \oplus I_{k+1} \subseteq I_k \oplus I_{k+1}$ with respect to $I_k'$. Let $S' = (e_1', e_2', \ldots, e_m')$. There are two possible cases (the proof of which we leave to the reader):

Case 1. There is some element $e_p' \notin I_k$, such that $I_k + e_p' \in \mathcal{J}_1$. Let $e_p'$ be the element of S' with largest index for which this is the case. Then $S = (e_p', e_{p+1}', \ldots, e_m')$ can be shown to be an augmenting sequence with respect to $I_k$.

Case 2. There is no such element $e_p'$. Then $S = (e_1, e_2, e_1', e_2', \ldots, e_m')$ is an augmenting sequence with respect to $I_k$.

The following corollary follows immediately from Theorem 5.1.

Corollary 5.2

An intersection I contains a maximum number of elements if and only if there exists no augmenting sequence with respect to I.

## 6. Labelling Procedure

Augmenting sequences are constructed by a labelling procedure not unlike that used for constructing flow augmenting paths in network flow problems, etc. What makes the procedure efficient is that the successor of a given element $e_j$ in an augmenting sequence does not in any way depend upon its predecessors, but only of $e_j$ itself.

At the start, no elements of E are labelled. Then each element in $E - sp^1(I)$ is given the label $\phi^+$. (If there are no such elements, the existing intersection I is clearly of maximum cardinality.) The symbol "$\phi$" indicates that the element in question is the first element of whatever augmenting sequence it may be found to be a member of. The "+" indicates that the element in question is to be added to I; a "-" sign indicates that the element is to be removed.

Additional elements are labelled by "scanning" existing labels. A "+" label on $e_i$ is scanned by first determining if $I + e_i$ is independent in $M_2$. In this case an augmenting sequence has been discovered, with $e_i$ as the final element. If $I + e_i$ is dependent, the unique circuit $C \subseteq I + e_i$ is found, and the label $i^-$ is given to each unlabelled element in C.

A "-" label on $e_i$ is scanned by giving each unlabelled element e in $sp^1(I) - sp^1(I - e_i)$ the label $i^+$.

The labelling procedure terminates when no further elements can be labelled or when an augmenting sequence is discovered, as described above. The complete augmenting sequence is obtained by "backtracking." I.e., if the label of the last element $e_i$ is "$j^+$" the second-to-last element in the sequence is $e_j$. If the label of $e_j$ is "$k^-$", the third-to-last element is $e_k$, etc. The initial element of the sequence, of course, has the label $\phi^+$.

The reader can verify that the rules of labelling construct augmenting sequences in accordance with the definitions in the previous section.

## 7. Algorithm for Cardinality Intersection Problem

Step 0 (Start)

Let I be any intersection of $M_1$, $M_2$ (possibly the null set). No elements are labelled.

Step 1 (Labelling)

1.0 Give each element in $E - sp^1(I)$ the label $\phi^+$.

1.1 If there are no unscanned labels, go to Step 3. Otherwise, find an element $e_i$ with an unscanned label. If the label is a "+" label (i.e. $e_i \notin I$) go to Step 1.2; if it is a "-" label go to Step 1.3.

1.2 Scan the "+" label as follows. If $I + e_i$ is independent in $M_2$, go to Step 2. Otherwise, identify the unique $M_2$-circuit C in $I + e_i$ and give each unlabelled element in C the label "$i^-$". Return to Step 1.1.

1.3 Scan the "-" label as follows. Give each unlabelled element in $sp^1(I) - sp^1(I-e_i)$ the label $i^+$. Return to Step 1.1

Step 2 (Augmentation)

2.1 An augmenting sequence S has been discovered, of which $e_i$(found in

Step 1.2) is in the last element. The elements in S are identified by "backtracking". I.e. if the label of $e_i$ is $\ell^+$, the second-to-last element is $e_\ell$. If the label of $e_\ell$ is $k^-$, the third-to-last element is $e_k$, etc. The initial element in the sequence has the label $\phi^+$.

2.2 Augment I by adding to I all elements in the sequence with "+" labels and removing from I all elements with "-" labels. Remove all labels from elements. Return to Step 1.0.

Step 3 (Hungarian Labelling)

No augmenting sequence exists and I is of maximum cardinality. The labelling is "Hungarian", and can be used to construct a minimum-rank covering dual to I. Halt.

## 8. Intersection Duality Theorem

The cardinality intersection computation provides a constructive proof of a duality theorem for matroid intersections. This theorem is of the min-max variety, similar to the max flow-min cut theorem of network flows and the Konig-Egervary theorem, of which it represents a proper generalization.

We say that a pair of subsets $E_1$, $E_2$ of E is a covering of E if $E_1 \cup E_2 = E$. With respect to a given pair of matroids $M_1$, $M_2$, we say that the rank of a covering $\mathcal{E} = (E_1, E_2)$ is $r(\mathcal{E}) = r^1(E_1) + r^2(E_2)$.

Lemma 8.1

For any covering $\mathcal{E}$ and any intersection I, $r(\mathcal{E}) \geq |I|$.

Proof:

Let $I_1 = I \cap E_1$, and $I_2 = I \cap (E_2 - E_1)$. Clearly $|I_1| \leq r^1(E_1)$ and $|I_2| \leq r^2(E_2)$ which implies $|I| \leq |I_1| + |I_2| \leq r(\mathcal{E})$.

Matroid Intersection Duality Theorem [3]

For any two matroids $M_1$, $M_2$, the maximum cardinality of an intersection is equal to the minimum rank of a covering.

Proof:

By the lemma, the rank of a covering cannot be less than the cardinality of an intersection. The algorithm enables us to construct a covering whose rank is equal to the cardinality of an intersection, as follows.

At the conclusion of the algorithm (when the labelling has become "Hungarian"), let the set $I_L$ contain the elements of I that are labelled and $I_U$ contain those which are not. Then $E_1 = sp^1(I_U)$, $E_2 = sp^2(I_L)$ are the sets of the desired covering.

To see that $E_1$, $E_2$ is a covering, suppose that there existed an element e that was not a member of either $sp^1(I_U)$ or $sp^2(I_L)$. If $e \notin sp^2(I_L)$, it cannot be labelled, because all such elements are in $sp^2(I_L)$, by the construction in Step 1.2. But if e is not labelled, $e \in sp^1(I-e')$, for all $e' \in I_L$, also by the construction in Step 1.2. But this implies that $e \in sp^1(I_U)$, contrary to assumption.

## 9. Computational Complexity

We wish to establish an upper bound on the length of the computation, as a function of n, the number of elements in E. We assume that a subroutine for independence testing requires c(n) steps.

Clearly there can be at most n augmentations. Let us establish a bound on the length of the computation for each augmentation.

Each element is scanned at most once. Hence there can be at most n such scanning operations. There are two types of scans. The "+" scan of Step 1.2 requires the testing of $I + e_i$ for independence, which is c(n) in length, and, usually, the identification of a circuit in $I + e_i$. This requires no more than $O(nc(n))$ steps. The "-" scan of Step 1.3 requires the computation of $sp^1(I-e_i)$, which requires no more than $O(nc(n))$ steps.

The "backtracking" required to construct an augmenting sequence is $O(n)$ in length, and all other operations required by each augmentation are of this order. Hence, the total number of computational steps per augmentation is $O(n^2c(n))$, resulting from n scanning operations with $O(nc(n))$ steps per scan.

We conclude that the overall complexity of the computation is $O(n^3c(n))$. Thus, if c(n) is a polynomial function of n, the overall computation is polynomial bounded.

# III. PRIMAL - DUAL ALGORITHM FOR WEIGHTED INTERSECTIONS

## 10. Linear Programming Formulation

With respect to a given matroid $M = (E, \mathcal{I})$, we say that a set

$E' \subseteq E$ is <u>self-spanning</u> if $sp(E') = E'$.

Let A be an incidence matrix of self-spanning sets and elements of E. In other words, each row i of A corresponds to a self-spanning set of the matroid (the indexing of these sets being arbitrary) and each column j corresponds to an element $e_j$. We set

$$a_{ij} = 1, \text{ if } e_j \text{ belongs to self-spanning set } i,$$
$$= 0, \text{ otherwise.}$$

Let $r = (r_1, r_2, \ldots, r_m)$ be a vector, where $r_i$ is the rank of self-spanning set i. It has been shown by Edmonds [3,4] that the extreme points of the convex polyhedron defined by the inequalities

$$Ax \leq r$$
$$x \geq 0$$

are in one-one correspondence with the independent sets of M. That is to say, if x is an extreme point, then each component $x_j$ is either 0 or 1, where $x_j = 1$ if element $e_j$ is a member of the independent set identified with the extreme point, and $x_j = 0$, if it is not.

Let A, B be incidence matrices and let r, s be vectors of ranks of self-spanning sets of two given matroids $M_1$, $M_2$ over the same set of elements. A surprising result of Edmonds [3] is that the extreme points of the convex polyhedron defined by the inequalities

$$Ax \leq r$$
$$Bx \leq s$$
$$x \geq 0$$

are in one-one correspondence with the intersections of $M_1$ and $M_2$.

The primal-dual algorithm described below provides a constructive proof of this theorem of Edmonds. That is, it is shown that, regardless of what element weights $w = (w_1, w_2, \ldots, w_n)$ may be chosen, the linear programming problem

$$\text{maximize } wx$$

subject to

$$\begin{aligned} Ax &\leq r \\ Bx &\leq s \\ x &\geq 0 \end{aligned} \tag{10.1}$$

has an optimal solution in zeros and ones.

## 11. Duality and Orthogonality Relations

The primal problem is an indicated in (10.1). The dual problem is

$$\text{minimize } ru + sv$$

subject to

$$A^T u + B^T v \geq w \tag{11.2}$$

$$u, v \geq 0,$$

where each dual variable $u_i$ is identified with a self-spanning set of $M_1$ and $v_k$ with a self-spanning set of $M_2$.

Orthogonality conditions necessary and sufficient for optimality of a pair of feasible primal and dual solutions are

$$x_j > 0 \Rightarrow (A^T u + B^T v)_j = w_j \tag{11.3}$$

$$u_i > 0 \Rightarrow (Ax_i) = r_i \tag{11.4}$$

$$v_k > 0 \Rightarrow (Bx)_k = s_k, \tag{11.5}$$

The algorithm begins with the feasible primal solution $x_j = 0$, for $j = 1, 2, \ldots, n$ (i.e. $I = \phi$), and with the feasible dual solution in which each dual variable $u_i$ or $v_k$ is zero, except $u_E$, the dual variable identified with the self-spanning set E. We set $u_E = \max \{w_j\}$. Thus, at the beginning of the computation the only orthogonality condition which is violated is

$$u_E > 0 \Rightarrow |I| = r^1(E). \tag{11.6}$$

The algorithm proceeds in stages. At each stage either the primal solution is revised by augmenting the existing intersection, or the values of the dual variables are revised. At all times, both primal and dual feasibility are maintained. Moreover, at each stage the only orthogonality condition which is not satisfied is (11.6). After a finite number of stages (in fact, a number bounded by a polynomial function in n, the number of elements in E), the condition (11.6) is also satisfied, and the primal and dual solutions existing at that point are optimal.

For a given pair of primal and dual solutions, the labelling routine of the cardinality intersection algorithm is applied, in an attempt to augment the primal solution. Clearly, the use of any augmenting sequence will result in a new feasible primal solution. However, the labelling routine must be modified in such a way that the only augmenting sequences which can be discovered are those for which all the orthogonality conditions except (11.6) continue to be satisfied.

If the application of the labelling routine, as restricted, does not result in the discovery of an augmenting sequence, then the dual solution is modified. The change in the dual solution must be such as to maintain dual feasibility, maintain satisfaction of all orthogonality conditions except (11.6), and also provide some progress toward the termination of the algorithm with optimal primal and dual solutions.

As a consequence of the fact that (11.6) is the only unsatisfied orthogonality condition, <u>the intersection existing at any intermediate stage of the computation is of maximum weight, relative to all intersections containing $|I|$ or fewer elements</u>. For suppose there were an additional constraint of the form

$$\sum_j x_j \leq k,$$

and we were to incorporate this constraint with the objective function via a Lagrange multiplier $\lambda$. Then an intermediate solution is easily shown to be optimal for $\lambda = u_E$ and therefore for a value of k equal to $|I|$.

## 12. <u>Form of Dual Solution</u>

At each stage of the computation, no more than 2n dual variables are permitted to be nonzero. These nonzero variables except $u_E$ are identical with spans of subsets of I in two different families $\mathcal{U}$ and $\mathcal{V}$

Specifically, let

$$\mathcal{U} = \{U_0, U_1, \ldots, U_p\}$$

and

$$\mathcal{V} = \{V_0, V_1, \ldots, V_q\} ,$$

where

$$U_0 = \phi, U_i \subset U_{i+1}, U_p = I,$$

and

$$V_0 = \phi, V_k \subset V_{k+1}, V_q = I.$$

Associated with subsets $U_i$ and $V_k$ are dual variables $u_i$ and $v_k$, where $u_i$ is identified with the self-spanning set $sp^1(U_i)$ and $v_k$ with $sp^2(V_k)$.

Suppose the primal solution is augmented by the application of the augmenting sequence $S = (e_1, e_2, \ldots, e_m)$. Then the families $\mathcal{U}$ and $\mathcal{V}$ are revised as follows: For $j = 3, 5, \ldots, m$, $e_j$ replaces $e_{j-1}$ in each of the subsets $U_i$ in which $e_{j-1}$ is contained. For $j = 1, 3, \ldots, m-2$, $e_j$ replaces $e_{j+1}$ in each of the subsets $V_k$ in which $e_{j+1}$ is contained. If $u_p > 0$, then p is incremented by one. If $v_q > 0$, then q is incremented by one. Then $U_p$ and $V_q$ are each set equal to I and $u_p$ and $v_q$ are set to zero.

In no case does this revision of the families $\mathcal{U}$ and $\mathcal{V}$ affect the dual solution. (No dual variables are changed in value.) However, unless the augmenting sequence is of a special type, a proper relation will not be maintained between the sets $U_i$, $V_k$ and the dual variables

$u_i$, $v_k$. Specifically, it is necessary that $sp^1(U_i)$ and $sp^2(V_k)$ are unaffected by change in membership of $U_i$ and $V_k$. This is achieved by a modification of the labelling procedure.

## 13. Modification of Labelling Procedure

The labelling procedure is modified in two ways. First, no element is given a label, unless it belongs to the set

$$E^* = \{e_j \mid (A^T u + B^T v)_j = w_j\}.$$

This insures that any augmenting sequence discovered by the labelling procedure will maintain satisfaction of the orthogonality conditions (11.3).

Second, we modify the rules for scanning as follows. Suppose $e_j$ is a labelled element of I. When $e_j$ is scanned, we find the smallest set $U_i$ of which $e_j$ is a member, and label only elements of $E^*$ which are in $sp^1(U_i) - sp^1(U_i - e_j) \subseteq sp^1(I) - sp^1(I - e_j)$. This insures that any augmenting sequence discovered will maintain satisfaction of the orthogonality conditions (11.4) except, of course (11.6). Moreover, when the subsets in $\mathcal{U}$ are revised, they have the same spans as before.

Similarly, suppose $e_j$ is a labelled element not in I. We determine if $I + e_j$ contains a circuit in $M_2$. Of course, if it does not, an augmenting sequence has been discovered. If $I + e_j$ does contain an $M_2$-circuit C, we find the smallest set $V_k$ of which $e_j$ is a member, and label only the elements in $C - V_{k-1}$. This insures that any augmenting sequence discovered will maintain satisfaction of the orthogonality conditions (11.5). Moreover, when the subsets in $\mathcal{U}$ are revised, they will have the same spans as before.

## 14. Revision of Dual Solution

If the labelling procedure, as modified above, terminates without the discovery of an augmenting sequence, then the dual solution is revised. This is done as follows.

First we create additional sets in the families $\mathcal{U}$ and $\mathcal{V}$. Let $I_L$ and $I_U$ denote the subsets of labelled and unlabelled elements of I. If, for i = 1, 2, ..., p, some, but not all, of the elements in $U_i - U_{i-1}$ are labelled, add one to the index of each of the sets $U_i$, $U_{i+1}$, ..., $U_p$, and then set $U_i = U_{i-1} \cup (U_i \cap I_U)$ and $u_i = 0$. I.e. "interpolate" a new set between $U_{i-1}$ and (the old) $U_i$. Similarly, if for k = 1, 2, ..., q, some, but not all, of the elements in $V_k - V_{k-1}$ are labelled, add one to the index of each of the sets $V_k$, $V_{k+1}$, ..., $V_q$, and then set $V_k = V_{k-1} \cup (V_k \cap I_L)$ and $v_k = 0$.

Let $\delta$ be a positive number yet to be determined. The dual variables are changed as follows. $U_E$ is decreased by $\delta$. If the elements of $U_p - U_{p-1}$ are unlabelled, $u_p$ is increased by $\delta$. If, for i = 1, 2, ..., p-1, the elements of $U_i - U_{i-1}$ are labelled (unlabelled) and those of $U_{i+1} - U_i$ are unlabelled (labelled), then $U_i$ is decreased (increased) by $\delta$. If the elements of $V_q - V_{q-1}$ are labelled, $v_q$ is increased by $\delta$. If for k = 1, 2, ..., t-1, the elements of $V_k - V_{k-1}$ are labelled (unlabelled) and those of $V_{k+1} - V_k$ are unlabelled (labelled), then $V_k$ is increased (decreased) by $\delta$. Otherwise, no dual variable is changed in value.

Now consider, for each element $e_j$, the effect of these changes in the dual variables on $(A^Tu + B^Tv)_j$. If there is a subset $U_i$ such that $e_j \in sp^1(U_i)$, let $U_{i(j)}$ be the smallest such subset. Exactly one of the following cases must hold:

<u>Case $E_u^-$:</u> $e_j \in E - sp^1(I)$, or else $e_j \notin E - sp^1(I)$ and the elements in $U_{i(j)} - U_{i(j)-1}$ are labelled.

<u>Case $E_u^0$:</u> $e_j \notin E - sp^1(I)$ and the elements in $U_{i(j)} - U_{i(j)-1}$ are unlabelled.

If there is a subset $V_k$ such that $e_j \in sp^2(V_k)$, let $V_{k(j)}$ be the smallest such set. Exactly one of the following cases must hold:

<u>Case $E_v^0$:</u> $e_j \in E - sp^2(I)$, or else $e_j \notin E - sp^2(I)$ and the elements in $V_{k(j)} - V_{k(j)-1}$ are unlabelled.

<u>Case $E_v^+$:</u> $e_j \notin E - sp^2(I)$ and the elements in $V_{k(j)} - V_{k(j)-1}$ are labelled.

If $e_j \in I$ then either the pair of cases $(E_u^-, E_v^+)$ or $(E_u^0, E_v^0)$ hold. Either way, the net change in $(A^Tu + B^Tv)_j$ is zero, and the orthogonality conditions (11.3) continue to be satisfied.

If $e_j \notin I$, one can examine each of the situations $(E_u^-, E_v^0)$, $(E_u^-, E_v^+)$, ..., $(E_u^0, E_v^+)$ to see when each applies. For example, suppose the pair $(E_u^-, E_v^0)$ applies. Then $e_j$ cannot be labelled, because if it were, $e_j$ would be a single-element augmenting sequence (and if there were an augmenting sequence discovered by the labelling procedure, no change in the dual variables would be called for). The fact that $e_j$ is unlabelled implies that $(A^Tu + B^Tv)_j > w_j$. If $\delta$ is chosen to be no greater than $(A^T u + B^T v) - w_j$, then the net decrease in $(A^T u + B^T v)$ will be such that dual feasibility condition (11.3) will continue to be satisfied.

One can verify that if $e_j$ is labelled (and therefore $(A^T u + B^T v)_j = w_j$), it follows that one of the pairs of cases $(E_u^-, E_v^+)$, $(E_u^0, E_v^0)$ applies. If $e_j$ is unlabelled and $(A^T u + B^T v)_j = w_j$ then either one of the preceding pairs of cases or $(E_u^0, E_0^+)$ applies. Thus, in all cases, (11,3) continues to be satisfied.

Suppose $e_j$ is unlabelled, but would be labelled except for the fact that $(A^T u + B^T v)_j > w_j$ (i.e. $e_j \notin E^*$). Then there is some labelled element $e_{j'} \in I$ such that $e_j \in sp^1(U_{i(j')}) - sp^1(U_{i(j')} - e_j)$, where $U_{i(j')}$ denotes the smallest set $U_i$ containing $e_{j'}$. It follows that case $E_u^-$ applies, and $(A_T u + B_T v)_j$ is either decreased or remains the same. (Actually, if $(A_T u + B_T v)_j$ is unchanged, it is because $E_v^+$ applies, in which case there is nothing to be gained by decreasing $(A_T u + B_T v)_j$, because all the elements of I which could be labelled as a result of labelling $e_j$ are already labelled.)

Finally, we note that the only variables which are decreased in value are those which are identified with sets $U_i$ and $V_k$ existing before the "interpolation" of new sets, as described above. Thus, if all of the dual variables (except possibly $u_p$ and $v_q$) identified with sets in $\mathcal{U}$ and in $\mathcal{V}$ are nonzero at the time the labelling computation is carried out, then there is a strictly positive value which can be assigned to $\delta$. Let $\Delta^-$ denote the set of dual variables, other than

$U_E$, which are to be decreased in value by the rules described above. The set of elements for which $(A^Tu + B^Tv)_j$ will be decreased is precisely the set $E_u^- - E_v^+ = E_u^- \cap E_v^0$. The largest value of $\delta$ we are free to choose is

$$\delta = \min\{u_E, \delta_{u,v}, \delta_e\},$$

where

$$\delta_{u,v} = \min \Delta^-$$

and

$$\delta_e = \min\{(A^Tu + B^Tv)_j - w_j | e_j \in E_u^- - E_v^0\}.$$

Any sets $U_i$, $V_k$ whose dual variables $u_i$, $v_k$ are reduced to zero are removed from the families $\mathcal{U}, \mathcal{V}$ before the labelling procedure is applied again at the next stage. (This does not apply to $U_p$ and $V_q$ for which $u_p$ and $v_q$ may remain at zero.) The elimination of these sets may enable additional elements to be labelled, according to the modified rules of the labelling procedure.

## 15. Primal-Dual Algorithm for Weighted Intersection Problem

Let $\bar{w}_j = (A^Tu + B^Tv)_j - w_j$

Step 0 (Start)

Set $I = \emptyset$,

$$u_E = \max_j\{w_j\},$$

$$U = \{U_o\} = \{\emptyset\},$$

$V = \{V_o\} = \{\emptyset\}$,

$\bar{w}_j = u_E - w_j$, $j = 1, 2, \ldots, n.$

$E^* = \{e_j | \bar{w}_j = 0\}$

<u>Step 1</u> (Labelling)

1.0 Give each element in $E^* - S^{(1)}(I)$ the label $\phi^+$.

1.1 If there are no unscanned labels, go to Step 3. Otherwise, find an element $e_i$ with an unscanned label. If the label is a "+" label go to Step 1.2; if it is a "-" label go to Step 1.3.

1.2 Scan the "+" label as follows. If $I + e_i$ is independent in $M_2$, go to Step 2. Otherwise, find the smallest set $V_k$ such that $e_i \in sp^2(V_k)$, identify the unique $M_2$-circuit C in $I + e_i$ and give each unlabelled element in $C - V_{k-1}$ the label $i^-$. Return to Step 1.1.

1.3 Scan the "-" label as follows. Find the smallest set $V_{i'}$, of which $e_i$ is a member, and give each element of $E^*$ in $sp^1(V_{i'}) - sp^1(V_{i'} - e_i)$ the label $j^+$. Return to Step 1.1.

<u>Step 2</u> (Augmentation Revision of Primal Solution)

2.1 An augmenting sequence S has been described, of which $e_i$ (found in Step 1.2) is the last element. The elements in S are identified by "backtracking." Augment I by adding to I all elements in the sequence with "+" labels and removing from I all elements with "-" labels.

2.2 Suppose, without lose of generality, the augmenting sequence S = $(e_1, e_2, \ldots, e_m)$. Revise the families $\mathcal{U}$ and $\mathcal{V}$ as follows: For j = 3, 5, ..., m, replace $e_{j-1}$ by $e_j$ in each of the subsets $U_i$ in which $e_{j-1}$ is contained. For j = 1, 3, ..., m-2, $U_j$ replaces $e_{j+1}$ by $e_j$ in each of the subsets $V_k$ in which $e_{j+1}$ is contained. If $u_p > 0$, increment p by one. If $v_q > 0$, increment q by one. Set $U_q = V_q = I$, and $u_q = v_q = 0$. Remove all labels from elements. Return to Step 1.0.

<u>Step 3</u> (Hungarian Labelling/Revision of Dual Solution)

3.1 Let $I_L$ and $I_U$ denote the subsets of labelled and unlabelled elements of I. If for i = 1, 2, ..., p, some, but not all, of the elements in $U_i - U_{i-1}$ are labelled, add one to the index of each of the sets $U_i$, $U_{i+1}$, ..., $U_p$ and then set $U_i = U_{i-1} \cup (U_i \cap I_u)$ and $u_i = 0$. If for k = 1, 2, ..., q, some but not all, of the elements in $V_k - V_{k-1}$ are labelled, add one to the index of each of the sets $V_k$, $V_{k+1}$, ..., $V_q$, and then set $V_k = V_{k-1} \cup (V_k \cap I_L)$ and $v_k = 0$.

3.2 Form the sets $\Delta^+$, $\Delta^-$ as follows. If the elements in $U_q - U_{q-1}$ are unlabelled, $u_q \in \Delta^+$. If for i = 1, 2, ..., q-1, the elements of $U_i - U_{i-1}$ are labelled (unlabelled) and those of $U_{i+1} - U_i$ are unlabelled (labelled) then $u_i \in \Delta^-$ $(u_i \in \Delta^+)$. If the elements of $v_q - v_{q-1}$ are labelled, $v_q \in \Delta^+$. If for k = 1, 2, ..., t-1, the elements of $V_k - V_{k-1}$ are labelled (unlabelled) and those of $V_{k+1} - V_k$ are unlabelled (labelled), $v_k \in \Delta^+$ $(v_k \in \Delta^-)$. Otherwise no dual variable $u_i$ or $v_k$ belongs to either $\Delta^+$ or $\Delta^-$.

3.3 Form the set $E_u^-$ as follows. For $j = 1, 2, \ldots, n$: If $e_j \in E - sp^1(I)$, then $e_j \in E_u^-$. Otherwise, find the smallest subset $U_{i(j)}$ in $\mathcal{U}$ such that $e_j \in sp^1(U_{i(j)})$. If the elements in $U_{i(j)} - U_{i(j)-1}$ are labelled, then $e_j \in E_u^-$. Form the set $E_v^+$ as follows. For $j = 1, 2, \ldots, n$: If $e_j \in sp^2(I)$, then find the smallest subset $V_{k(j)}$ in $\mathcal{V}$ such that $e_j \in sp^2(V_{k(j)})$. If the elements in $V_{k(j)} - V_{k(j)-1}$ are labelled, then $e_j \in E_v^+$.

3.4 Set $\delta = \min\{u_E, \delta_{u,v}, \delta_e\}$, where

$$\delta_{u,v} = \min \Delta^-,$$

$$\delta_e = \min\{\bar{w}_j \mid e_j \in E_u^- - E_v^+\} .$$

Set $u_i = u_i - \delta$, $v_k = v_k - \delta$, for all $u_i, v_k \in \Delta^-$, and $u_i = u_i + \delta$, $v_k = v_k + \delta$, for all $u_i, v_k \in \Delta^+$. Set $\bar{w}_j = \bar{w}_j - \delta$, for all $e_j \in E_u^- - E_v^+$ and $\bar{w}_j = \bar{w}_j + \delta$ for all $e_j \in E_v^+ - E_u^-$. Set $u_E = u_E - \delta$. If $u_E = 0$, halt (the primal and dual solutions are optimal). Otherwise, remove from $\mathcal{U}$ and $\mathcal{V}$ any sets $U_i$ and $V_k$, except $U_p$ and $V_q$, for which $u_i$ or $v_k$ is zero and renumber the sets in $\mathcal{U}$ and $\mathcal{V}$ accordingly. Set $E^* = \{e_j \mid \bar{w}_j = 0\}$. Return to Step 1.0.

## 16. Computational Complexity

There can be at most n augmentations. Between each augmentation, there may be many revisions of the dual variables, and we must estimate the number of such revisions.

Each time the values of the dual variables are revised, either an unlabelled element goes into the set $E^*$ and becomes labelled or at least one dual variable is decreased to zero. The former can occur at most n times before either augmentation or termination must occur. The latter can occur at most n times in succession, by the following reasoning. There are at most 2n nonzero dual variables, at most n of which are decreased in value with each revision of the dual variables. If no new elements are labelled, the same dual variables are decreased in value the next time a revision is made. This can be repeated at most n times in succession before $u_E$ goes to zero and termination occurs.

The conclusion is that at most $n^2$ revisions of the dual variables can occur between augmentations. Each such revision occasions a relabelling and rescanning of all the elements, which requires $O(n^2\, c(n))$ steps. (Perhaps the algorithm could be modified to avoid some duplicated effort.) The arithmetic and bookkeeping operations are dominated by the labelling and scanning operations. Accordingly, each augmentation may require as many as $O(n^4\, c(n))$ steps.

Since there are at most n augmentations and $O(n^4\, c(n))$ steps per augmentation, we conclude that the overall complexity is $O(n^5\, c(n))$.

## IV. PRIMAL ALGORITHM FOR WEIGHTED INTERSECTIONS

### 17. Weighted Augmenting Sequences

The "primal" algorithm is analogous to an algorithm of Busacker and Gowan [1] and of Jewell [6] for computing minimum-cost network flows. Their procedure is based on the following theorem (which we state without repeating definitions).

Theorem 17.1

Let $X_v$ be a minimum-cost flow of value v. Then the augmentation of $X_v$ by value $\varepsilon$ along a minimum cost augmenting path yields a minimum-cost flow of value $v + \varepsilon$.

The matroid algorithm proceeds by computing maximum-weight intersections containing successively larger numbers of elements. Having obtained $I_k$, a maximum-weight intersection with k elements, $I_{k+1}$ is obtained from $I_k$ by constructing a "maximum-weight augmenting sequence". This represents a generalization of the network flow algorithm in exactly the same sense that the matroid intersection problem is itself a generalization of the minimum-cost network flow problem (or, more precisely, the maximum-weight bipartite matching problem).

It is believed that this algorithm is conceptually simpler, and possibly more efficient, than the primal-dual algorithm presented above.

For any subset $E' \subseteq E$ we let $w(E')$ denote the sum of the weights of the elements in E. I.e.

$$w(E') = \sum_{e_j \in E'} w_j .$$

Given an augmenting sequence S with respect to an intersection I, we define the <u>weight of S</u> to be

$$\Delta(S) = w(S-I) - w(S \cap I).$$

Clearly,

$$w(I \oplus S) = w(I) + \Delta(S).$$

The matroid generalization of Theorem 17.1 is as follows:

Theorem 17.2

Let $I_k$ be a maximum-weight intersection with k elements and let S be a maximum-weight augmenting sequence with respect to $I_k$. Then $I_k \oplus S$ is a maximum-weight intersection with k+1 elements.

Proof:

At any given point in the primal-dual computation, the intersection $I_k$ existing at that point is of maximum weight with respect all intersections containing $|I_k| = k$ elements. $I_k$ is augmented by means of an augmenting sequence S to obtain a maximum-weight intersection with k+1 elements. Such a sequence S must be a maximum weight augmenting sequence, or else $I_k \oplus S$ would not be optimal. It follows that any maximum-weight sequence yields an optimal intersection with k+1 elements.

## 18. Concavity Property

Clearly, it is possible to start with the empty set and apply the augmenting sequence algorithm to obtain $I_1$, $I_2$, $I_3$, ..., maximum-weight intersections with 1, 2, 3, ... elements respectively, stopping when no further augmentation is possible. One can then compare the weights of these various intersections so as to determine an intersection which has maximum weight without restriction on the number of elements.

However, "the maximum weight of intersections is concave in k", just as "the minimum cost of flows is convex in the value of the flow." This means that if one seeks to compute a maximum-weight intersection without restriction on the number of elements, such a set is given by

$I_k$, where k is the smallest number of elements such that $w(I_k) \geq w(I_{k+1})$.

Theorem 18.1

Let $I_1$, $I_2$, $I_3$, ... be maximum-weight intersections with 1, 2, 3, ... elements respectively. Then

$$w(I_{k+1}) - w(I_k) \leq w(I_k) - w(I_{k-1}),$$

for k = 1, 2, 3,.

Proof:

Maximal-weight intersections for $I_{k-1}$ and $I_{k+1}$ are feasible solutions of the linear programming problem formulated in Section 10. Any convex combination of these two solutions is also a feasible solution, and is dominated by an optimal solution at an extreme point of the polyhedron corresponding to an intersection with k elements.

## 19. Primal Algorithm

Let $\Delta(S)$ denote the weight of the weightiest augmenting sequence S discovered at any given point in the procedure, and let m be the index of its last element. Let $\Delta(e_j)$ denote the weight of the weightiest "partial" sequence terminating in the element $e_j$. Labels are removed and new labels are applied to any given element $e_j$ whenever the value of $\Delta(e_j)$ can be increased.

Algorithm for Constructing Maximum-Weight Augmenting Sequence

Step 0 (Start)

Let I be the null set. No elements are labelled. Set $\Delta(S) = -\infty$,

$m = 0$, and $\Delta(e_j) = -\infty$, for all $e_j$.

Step 1 (Labelling)

1.0 Give each element $e_i$ in $E - sp^1(I)$ the label $\phi^+$ and set $\Delta(e_i) = w_i$.

1.1 If there are no unscanned labels, go to Step 3. Otherwise, from among the elements whose labels are unscanned, select that element $e_i$ whose label was the first to be applied. If the label is a "+" label (i.e. $e_i \notin I$) go to Step 1.2; if it is a "-" label go to Step 1.3.

1.2 Scan the "+" label as follows. If $I + e_i$ is independent in $M_2$ go to Step 2. Otherwise, identify the unique $M_2$-circuit C in $I + e_i$. Give each unlabelled element $e_j$ in C the label $i^-$ and set $\Delta(e_j) = \Delta(e_i) - w_j$. For each labelled element $e_j$ in C, compare $\Delta(e_j)$ with $\Delta(e_i) - w_j$. If $\Delta(e_j) < \Delta(e_i) - w_j$, set $\Delta(e_j) = \Delta(e_i) - w_j$, remove the existing label and apply the new (unscanned) label $i^-$ to $e_j$. Retuen to Step 1.1.

1.3 Scan the "-" label as follows. Give each unlabelled element $e_j$ in $sp^1(I) - sp^1(I-e_i)$ the label $i^+$. For each labelled element $e_j$ in $sp^1(I) - sp^1(I-e_i)$, compare $\Delta(e_j)$ with $\Delta(e_i) + w_j$. If $\Delta(e_j) < \Delta(e_i) + w_j$, set $\Delta(e_j) = \Delta(e_i) + w_j$, remove the existing label and apply the new (unscanned) label $i^+$ to $e_j$. Return to Step 1.1.

Step 2 (Construction of Maximum-weight Augmenting Sequence)

2.0 If $\Delta(S) < \Delta(e_i)$, set $\Delta(S) = \Delta(e_i)$ and set $m = i$. Return to Step 1.1.

2.1 A maximum-weight augmenting sequence S can be constructed, of which $e_m$ is the last element. The elements in S are identified by "backtracking". I.e. if the label of $e_m$ is $\ell^+$, the second-to-last element is $e_\ell$. If the label of $e_\ell$ is $k^-$, the third-to-last element is $e_k$, etc. The first element in the sequence has the label $\phi^+$.

2.2 Augment I by adding to I all elements in the sequence with "+" labels and removing from I all elements with "-" labels. Remove all labels from elements. Return to Step 1.0.

<u>Step 3</u> (Hungarian Labelling)

If $\Delta(S) \geq 0$, go to Step 2.1. If $-\infty < \Delta(S) < 0$, then I is of maximum weight, but not maximum cardinality. If $\Delta(S) = -\infty$, no augmenting sequence exists, the labelling is "Hungarian", and I is of both maximum weight and maximum cardinality. Halt.

20. <u>Convergence</u>

It is essential that the labelling procedure applies labels in such a way that no element $e_j$ can be reached by backtracking from its own label. Suppose this were to occur, and let A be the set of elements so reached by backtracking from $e_j$ to $e_j$. Then it can be shown that $I \oplus A$ would be an intersection and $|I \oplus A| = |I|$, $w(I \oplus A) > w(I)$, contrary to the assumption that I is of maximum weight, relative to all intersections with $|I|$ elements. (This assertion is exactly analogous to the proposition that a network flow is of minimum cost if and only if there exists no circulation with respect to that flow which has negative cost.)

The above observation suggests that there are only a finite number of sequences of elements which can be reached by backtracking from any given element $e_j$. And since each such sequence uniquely determines a value of $\Delta(e_j)$, there are only a finite number of such values and a finite number of labels that can be applied to $e_j$. This is the essential aspect of the argument for the finiteness of the labelling procedure and of the overall algorithm.

The augmenting sequences found by the algorithm are, in fact, maximum-weight augmenting sequences. For suppose S is a sequence found by the algorithm, and there exists some sequence $S' = (e_1, e_2, \ldots, e_m)$ such that $\Delta(S') > \Delta(S)$. Let $\Delta(e_i)$ be as computed by the algorithm, and let

$$\Delta'(e_i) = w_1 - w_2 + \ldots \pm w_i,$$

where the last sign is "+" or "-" depending upon whether i is odd or even. It is not possible for $\Delta'(e_1) > \Delta(e_1)$, so there must be some element $e_i$ such that $\Delta'(e_i) = \Delta(e_i)$ and $\Delta'(e_{i+1}) > \Delta(e_{i+1})$. But this implies that the last time a label of $e_i$ was scanned by the algorithm, $\Delta(e_i)$ was strictly less than its final value; otherwise $\Delta(e_{i+1})$ would have been at least as great at $\Delta'(e_{i+1})$ at the end of the computation. But if $\Delta(e_i)$ was increased in value after the last time its label was scanned, $e_i$ would have been given a new label, and this label would have been scanned before the end of the computation. This is a contradiction. Hence S must be a maximum-weight sequence.

## 21. Computational Complexity

We have observed that the primal algorithm terminates in a finite number of steps. In order to establish an upper bound on the length of the computation, we must first bound the number of scanning operations.

Suppose, for a given label at any given point in the computation, we backtrack from that label until an element with the label $\phi^+$ is reached. Let the depth of the label be taken to be equal to the number of elements so reached. (The depth of the label $\phi^+$ is unity.)

Step 1.1 requires that labels be scanned in the order in which they are applied. Thus all labels with depth two are scanned first. The scanning of these labels causes labels with depth two to be applied and these are scanned next. Thus labels with depth 1, 2, 3, 4, ... are successively scanned. This means that each time the label of any given element is scanned, its depth is at least two greater than the time before. But the depth of a label cannot exceed n. It follows that the label of any given element cannot be scanned more than $\frac{n}{2}$ times, and that there are at most $\frac{n}{2}$ scanning operations per augmentation.

An estimate of the complexity of the algorithm is obtained by reasoning similar to that used in Section 9 for the cardinality intersection algorithm. The principal difference is that in the primal algorithm each element may be scanned as many as $\frac{n}{2}$ times, rather than at most once. The arithmetic operations of the algorithm are dominated by the scanning operations. Accordingly, the overall complexity is $O(n^4\ c(n))$.

This estimate of complexity is lower by a factor of n than the estimate for the primal-dual method, but it is not clear whether the primal method will actually be more efficient in practice, since both estimates are only upper bounds.


## ACKNOWLEDGEMENT

The author must acknowledge the paramount contributions of Jack Edmonds. In 1965, the author was first exposed to matroids and matroid polyhedra by Edmonds' lectures at the University of California - NATO Conference on network flows and Integer Polyhedra. At that time Edmonds discussed the matroid polyhedral intersection theorem, for which the primal-dual algorithm provides a constructive proof. The author first reported the primal-dual algorithm at the 1969 Calgary International Conference on Combinatorial Structures and Their Applications.

The author must also acknowledge the very helpful advice of Richard Karp in the early stages of this work. Arnon Rosenthal provided a careful reading of the manuscript and helped correct a number of errors.

## REFERENCES

[1] R. G. Busacker and P. J. Gowan, "A procedure for determining a family of minimal-cost network flow patterns," ORO Tech. Report 15, Operations Research Office, Johns Hopkins University (1961).

[2] J. Edmonds, "Matroid partition," Math. of the Decision Sciences, American Mathematical Society Lectures in Applied Mathematics, 11 (1968), pp. 335-345.

[3] J. Edmonds, "Optimum branchings," Journal of Research of the National Bureau of Standards, 71B (1967), pp. 233-240; reprinted with [2], pp. 346-361.

[4] J. Edmonds, "Matroids and the greedy algorithm," Mathematical Programming, 1 (1971), pp. 127-136.

[5] J. Edmonds, "Submodular functions, matroids and certain polyhedra," Combinatorial Structures and Their Applications, Proceedings of the Calgary International Conference, Gordon and Breach, N. Y. (1970), pp. 69-87.

[6] J. Edmonds and D. R. Fulkerson, "Transversals and matroid partition," Journal of Research of the National Bureau of Standards, 69B (1965), pp. 147-153.

[7] W. S. Jewell, "Optimal flow through a network with gains," Proc. Second International Conference on Operations Research, Aix-en Provence, France (1960).

[8] S. Kundu and E. L. Lawler, "A matroid generalization of a theorem of Mendelsohn and Dulmage," to appear in Discrete Mathematics.

[9] E. L. Lawler, "Optimal matroid intersections," Combinatorial Structures and Their Applications, Proceedings of the Calgary

*International Conference*, Gordon and Breach, N. Y. (1970), abstract only, p. 233.

[10] E. L. Lawler, "Matroids with parity conditions; a new class of combinatorial optimization problems," submitted for publication.