ON THE SIMPLEX ALGORITHM FOR NETWORKS AND GENERALIZED NETWORKS

by

James B. Orlin

Sloan School of Management
MIT

Working Paper #146783

On The Simplex Algorithm for Networks and Generalized Networks

by James B. Orlin

Sloan School of Management

MIT

Abstract

We consider the simplex algorithm as applied to minimum cost network flows on a directed graph $G = (V, E)$. First we consider the strongly convergent pivot rule of Elam, Glover, and Klingman as applied to generalized networks. We show that this pivot rule is equivalent to lexicography in its choice of the variable to leave the basis. We also show the following monotonicity property that is satisfied by each basis B of a generalized network flow problem. If $b' \le b \le b^*$ and if $\ell \le B^{-1}b'$, $B^{-1}b^* \le u$, then $\ell \le B^{-1}b \le u$; i.e., if a basis is feasible for b' and b^* then it is feasible for b. Next we consider Dantzig's pivot rule of selecting the entering variable whose reduced cost is minimum and using lexicography to avoid cycling. We show that the maximum number of pivots using Dantzig's pivot rule is $O(|V|^2|E| \log|V|)$ when applied to either the assignment problem or the shortest path problem. Moreover, the maximum number of consecutive degenerate pivots for the minimum cost network flow problem is $O(|V|^2|E| \log|V|)$.

1. Introduction

In this paper we consider the following linear program.

$$\begin{array}{ll} \text{Minimize} & cx \\ \text{Subject to} & Ax = b \\ & \ell \le x \le u\ , \end{array} \qquad (1)$$

where A is an $m \times n$ integer-valued matrix with full row rank, b is an integral m-vector, and c, p, and u are integral n-vectors.

The linear program (1) is called a <u>generalized network flow problem</u> if each column of A has at most one positive entry and at most one negative entry. If (1) is a generalized network flow problem and, in addition, all entries are 0, 1, or -1 then it is called an <u>(ordinary) network flow problem</u>.

It has been established (see, for example, Elam et. al. (1979) and McBride (1981) that simplex-type procedures are very efficient in practice for solving generalized and ordinary network flows. In this paper we analyse the worst case behavior of the simplex algorithm as applied to ordinary network flows.

The variant of the simplex algorithm that we consider is Dantzig's pivot rule of selecting the entering variable whose reduced cost is minimum and using lexicography to avoid cycling.

In Section 2 of this paper, we show that the "strongly convergent" pivot rule developed by Elam et. al. for the generalized network flow problem is equivalent to lexicography. This former rule is a generalization of the "strongly feasible" pivot rule for ordinary network flows as developed by Cunningham (1976) and independently by Barr et. al. (1977).

We also show that if a basis B for a generalized network flow problem is feasible for (1) for right hand sides b' and b^* with $b' \le b^*$, then B is also feasible for all right hand sides b satisfying $b' \le b \le b^*$.

In Section 3 of this paper, we consider Dantzig's pivot rule as applied to ordinary network flows. Let $u^* = \max(u_j - \ell_j : j - 1, \ldots, n)$ and suppose that w^* is an upper bound on the difference in objective values between any two feasible solutions. We show that the maximum number of consecutive degenerate pivots is $O(m\ n \log w^*)$, and the maximum number of pivots is $O(m\ n\ u^* \log w^*)$. In particular, the simplex algorithm with Dantzig's pivot rule leads to a polynomially bounded number of pivots for both the shortest path problem and the optimal assignment problem. Moreover, in the case that w^* is large, we can replace $\log w^*$ by $n \log n$. In fact, we show that the number of pivots for either the shortest path problem or the optimal assignment problem is $O(m^2 n \log m)$.

These latter results contrast with those of Cunningham (1979) who showed that Bland's pivot rule and a different pivot rule using lexicography could lead to an exponential number of degenerate pivots for the shortest path problem. (His example was a modification of an example by Edmonds (1970)). Cunningham also provided an alternative pivot rule that prevented "stalling" , which is the occurrence of an exponential sequence of degenerate pivots.

Other researchers have shown that special cases of the simplex algorithm run in polynomial time. Zadeh (1979) showed that Dantzig's pivot rule for the shortest path problem starting from an artificial basis leads to Dijkstra's algorithm. Thus the number of pivots is $O(m^2)$ if all costs are non-negative. More recently, Hung (1983) developed a simplex method for the assignment problem in which the number of pivots is $O(m^3 \log w^*)$.

2. On Strongly Feasible Bases

Preliminaries

Let us consider an instance of the generalized network flow problem (1) in which the constraint matrix A has columns $A_1, \ldots, A_n$. We associate with A a directed graph $G = G(A)$, where $G = (V, E)$ with edge multipliers d, where $V = \{0, 1, \ldots, m\}$ and $E = \{e_1, \ldots, e_n\}$ with edge multipliers $d_1, \ldots, d_n$ defined as follows.

(1) If A_k has a positive entry in row i and a negative entry in row j, then $e_k = (i, j)$ and $d_k = |a_{jk}/a_{ik}|$.

(2) If A_k has a positive entry in row i and no negative entry, then $e_k = (i, 0)$ and $d_k = 1/a_{ik}$.

(3) If A_k has a negative entry in row j and no positive entry, then $e_k = (0, j)$ and $d_k = -a_{jk}$.

We have not used the usual convention of associating loops with columns that have only one non-zero entry. With our convention the bases for ordinary network flow problems correspond to spanning trees.

We illustrate the construction of the graph $G(A)$ in Figure 1, which corresponds to the matrix A of Table 1.

$$\begin{array}{cc} 1 & \\ 2 & \\ 3 & \\ 4 & \end{array}\begin{bmatrix} 1 & 2 & 0 & -1 & 0 & 0 \\ -1 & 0 & 1 & 0 & 2 & 0 \\ 0 & -3 & 0 & 0 & 0 & 2 \\ 0 & 0 & -2 & 0 & 0 & -1 \end{bmatrix}$$

Table 1. A constraint matrix for a generalized network.

[Figure 1]

We assume that each circuit C of G has an associated orientation. The edges of C that are oriented in the same direction as C are called forward edges of C, and the other edges of C are called backward edges. The flow multiplier of C is the product of the multipliers of the forward edges of C divided by the product of the multipliers of the backward edges of C, and we denote it as $d(C)$. For example, if C is the circuit 1, 2, 4, 3, 1 of Figure 1, then the forward edges are (1, 2) and (2, 4) and $d(C) = 8/3$.

By a basis of A we mean a partition of the columns of A into a triple (B, N_1, N_2) such that B is non-singular. The corresponding basic solution is the unique solution x such that $Ax = b$ and $x_j = \ell_j$ for $A_j \in N_1$ and $x_j = u_j$ for $A_j \in N_2$.

It is well known (see for example Elam et. al.) that a necessary and sufficient condition for a submatrix B to be a non-singular submatrix of A is that the corresponding graph G_B satisfies the properties (2.1) and (2.2) below.

(2.1) The connected component of G_B containing vertex 0 has no cycle.

(2.2) Each of the components of G_B not containing vertex 0 has exactly one circuit and the flow multiplier of this circuit is not 1.

This condition generalizes the spanning tree property for ordinary networks since the flow multiplier for every circuit in an ordinary network is 1.

In order to simplify some of the definitions, we consider negations of variables. Thus we allow the replacement of variable x_j by $x_j' = -x_j$.

(This substitution replaces the upper and lower bound constraints on x_j by the constraints $-u_j \le x_j' \le -\ell_j$, and the resulting edge multiplier is $1/d_j$. This substitution corresonds to reorienting the edge e_j in the graph G defined above.) We say that a basis (B, N_1, N_2 is canonically oriented if (3.1) – (3.4) are satisfied.

(3.1) $N_2 = \emptyset$.

(3.2) If vertex $i \neq 0$ is in the same component as vertex 0, then the path in G_B from 0 to i is a directed path (i.e., all edges are forward edges.)

(3.3) If C is a circuit in G_B, then C is a directed circuit oriented so that its flow multiplier is greater than 1.

(3.4) If i is a vertex in the same component as a circuit C, then there is a directed path from a vertex j of C to the vertex i.

In Figure 2, we illustrate a basis B that is canonically oriented. In Figure 3a and 3b we illustrate bases that are not canonically oriented.

[Figure 2]

[Figure 3]

A basis (B, N_1, N_2) is feasible if the corresponding basic solution is feasible. A basis is called strongly feasible as per Cunningham (1976) if it is feasible and if, in addition, when canonically oriented no variable is at its upper bound. (Elam et. al. call such bases "strongly convergent.") Below we show that a basis is strongly feasible if and only if it is lexicographically positive. This equivalent is a c corollary of Lemma 1.

We write b <.0) to mean that each component of b is negative (resp., positive.)

LEMMA 1. Suppose that (B, N_1, N_2) is a canonically oriented basis for a generalized network flow problem. Then for any m-vector b such that $b \le 0$ it follows that $B^{-1}b \ge 0$. Moreover, if $b < 0$ then 13^{-1} $b > 0$.

PROOF. Let $x = B^{-1}b$. Then x is the unique solution to $Bx = b$. We first show that the flow x_j for any non-circuit edge e_j is strictly positive. If $e_j = (i, k)$ and if vertex k has degree 1, then $x_j = -b_k/d_j$. In this case we can replace b_i by $b_i' = b_i - x_j$ and iterate. We eventually obtain a sub-basis such that no vertex has degree 1. Thus this sub-basis is the union of disjoint circuits.

Suppose that C is a circuit of G_B. Suppose further that we relable the vertices and edges of G_B so that $C = (1, e_1, 2, e_2, \ldots, e_k, 1)$. By a unit flow around C starting at vertex 1, we mean the flow in which $x_1 = (d(C) - 1)^{-1}$ and $x_i = d_i x_{i-1}$ for $2 \le i \le k$. Thus the flow balances at each vertex of C except that there is a gain in flow of one unit at vertex 1. Thus to satisfy the demand of $-b_i$ units at node i, it suffices to send a flow of $-b_i$ units around C starting at vertex i. By sending such a flow for all vertices i in circuits, we see that the resulting solution is strictly positive. □

COROLLARY 1. Suppose that (1) is a generalized network flow problem and that (B, N_1, N_2) is any basis of A. Then each row vector of B^{-1} is either non-positive or non-negative.

PROOF. If B is canonically oriented, then each row vector of B^{-1} is non-positive, by Lemma 1. If B is not canonically oriented, then there is a non-singular diagonal matrix D such that BD is canonically oriented. Then each row of $(BD)^{-1} = D^{-1}B^{-1}$ is non-positive, and hence each row of B^{-1} is either non-positive or non-negative. □

The proof of Corollary 1 relies only on some elementary concepts in Network Flow Theory. One can also construct an alternative proof that relies on concepts from the linear algebra of Leontief and pre-Leontief systems. A pre leontief matrix is a matrix with at most one positive entry per column. Generalized networks have the especially nice property that they remain pre-leontief after negations of variables. We have relied on concepts from network flow theory so as to make the exposition self-contained and so as to make the connection between strongly feasible bases and lexico-feasible bases more explicit. For more details on leontief and pre-leontief systems, see Veinott (1968).

For each vector-valued function $g(\cdot)$ we define the parametric linear program P_g as follows.

$$\begin{array}{lll} \text{Minimize} & cx & \\ \text{Subject to} & Ax = b - g(\Theta) & \qquad\qquad P_g(\Theta) \\ & \ell \le x \le u \;. & \end{array}$$

THEOREM 1. Let g be a continuous, vector-valued function such that $g(0) = 0$, $g(\Theta) > 0$ for all $\Theta > 0$, and for $\Theta_1 < \Theta_2$ it follows that $g(\Theta_1) \le g(\Theta_2)$. Then the following are true.

(i) A basis (B, N_1, N_2) is strongly feasible for (1) if and only if it is feasible for $P_g(\Theta)$ for all sufficiently small positive Θ.

(ii) If a basis (B, N_1, N_2) is feasible (resp. strongly feasible) for $P_g(\Theta_1)$ and $P_g(\Theta_2)$ then it is feasible (resp., strongly feasible) for $P_g(\Theta')$ for all Θ' with $\Theta_1 \le \Theta' \le \Theta_2$.

PROOF. Without loss of generality, let us assume that the basis (B, N_1, N_2) is canonically oriented. Let $b' = b - \Sigma(A_j\ell_j : A_j \in N_1)$.

Let us first consider property (ii). Assume that the basis (B, N_1, N_2) is feasible for $P_g(\Theta_1)$ and $P_g(\Theta_2)$. Then

$$\ell_B \le B^{-1}[b' - g(\theta_1)] \;,\; B^{-1}[b' - g(\theta_2)] \le u_B. \tag{2}$$

Moreover, by Lemma 1, it follows that for all θ' with $\theta_1 < \theta' \le \theta_2$

$$B^{-1}[b' - g(\theta_1)] \le B^{-1}[b' - g(\theta')] \le B^{-1}[b' - g(\theta_2)]. \tag{3}$$

By (2) and (3) it follows that (B, N_1, N_2) is feasible for $P(\theta')$.

Suppose now that (B, N_1, N_2) is strongly feasible $P_g(0)$. Then similarly to (2) and (3),

$$\ell_B \le B^{-1}(b' - g(0)) < u_B \tag{4}$$

and because $g(\theta) > 0$ for all $\theta > 0$, we obtain

$$\ell_B < B^{-1}(b; - g(\theta)). \tag{5}$$

It follows from (4) and (5) and the continuity of θ that (B, N_1, N_2) is strongly feasible for $P_g(\theta)$ for all sufficiently small positive θ.

Suppose now that (B, N_1, N_2) is feasible for $P_g(\theta)$ for all sufficiently small positive θ. Then we can choose a $\theta' > 0$ so that

$$\ell_B < B^{-1}(b' - g(\theta) < B^{-1}(b' - g(\theta')) < u_B \quad \text{for all} \quad 0 < \theta < \theta'$$

and thus by the continuity of g, (B, N_1, N_2) is strongly feasible for (1). □

In particular, we can let $g(\theta)$ be the vector whose j^{th} component is θ^j. Then for sufficiently small positive θ , the problem $P_g(\theta)$ corresponds to (1) with the negative of the usual perturbation.

COROLLARY. The canonically oriented basis (B, N_1, N_2) is strongly feasible if and only if the corresponding vector x'_B of basic variable is such that $(x'_B - \ell_B, -B^{-1})$ is lexico-positive. □

We refer the reader to Dantzig (1963) for more details on the equivalence of lexicography and perturbations.

An interesting special case of $P(\Theta)$ is the case in which $g(\Theta) = \Theta 1$, i.e., the j-th component is Θ for all j. Cunningham (1976) showed that a basis for an ordinary network flow problem is strongly feasible if it is feasible for $P_g(\Theta)$ for all sufficiently small Θ for this special case.

Since the strongly convergent pivot rule of Elam et. al. selects the exiting variable so as to maintain strongly feasible bases, we have also shown the following.

COROLLARY. The strongly convergent pivoting rule is equivalent to lexicography in the way that if selects the variable to leave the basis. □

The property (ii) of Theorem 1 is not typical of parametric linear programs. Indeed, the property does not appear to be generalizable beyond the class of generalized network flow problems.

A Concluding Remark

It appears in retrospect that the contributions of Cunningham and Elam et. al. in the development of their strongly feasible and strongly convergent pivot rules was not -- as they originally believed -- in the development of a novel new pivot rule. Instead, their contributions may be viewed as the novel treatment of a very old pivot rule. In particular, their pivot rules may be viewed as a (network) topological interpretation of the usual perturbation method for linear programming, an interpretation that lends itself well to an efficient implementation of the network simplex code.

3. On the Number of Simplex Pivots for Network Flows.

In this section we consider the simplex method as applied to ordinary network flows. The entering variable will be selected according to the "$\bar{c}_{min}$" rule of selecting the variable whose reduced cost is minimum (or one whose reduced cost is maximum in the case of a variable at its upper bound.) The existing variable is selected using lexicography so as to keep the basis strongly feasible. This pivot rule is called "Dantzig's pivot rule" since he is the first to propose the "$\overline{c}_{min}$" rule.

We show below that Dantzig's pivot rule as applied to the shortest path problem and the optimal assignment problem runs in $O(m^2 \log w^*)$ pivots, where w^* is an upper bound on the difference in objective values for any two basic feasible solutions. Moreover, in the case that w^* is more than exponentially large, we can improve the bound to $O(m^2 n \log n)$ pivots. If we run Dantzig's pivot rule on minimum cost network flows, then the number of pivots may be exponentially large, as demonstrated by Zadeh (1973). In this case, we show that the number of pivots is $O(m^2 u^* \log w^*)$, and the number of consecutive degenerate pivots is $O(m^2 \log w^*)$ or $O(m^2 n \log n)$, whichever is smaller.

To prove the convergence results, we first define the concept "equivalence" of network flow problems. Let $P = \{\min cx : Ax = b, \ell \le x \le u\}$, and let $P' = \{\min c'x : Ax = b', \ell \le x \le u\}$. We say that the linear programs P and P' are <u>equivalent</u> if the following are true:

(1) A basis (B, N_1, N_2) is strongly feasible for P if and only if it is strongly feasible for P',

(2) If (B, N_1, N_2) is any strongly feasible basis for P reoriented so that it is canonically oriented, then $\{j : \overline{c}_j = \overline{c}_{min}\} = \{j : \overline{c}'_j = \overline{c}'_{min}\}$, i.e., the variables that may enter the basis according to Dantzig's rule are the same for P' and P.

REMARK. If P is equivalent to P' and if f is an upper bound on the number of pivots for P using Dantzig's rule, then f is also an upper bound on the number of pivots for P'. □

Although the above remark is obvious, we note that the number of pivots for P and P' may not be the same. It is possible that ties for the entering variable would be resolved differently for P and P' under some implementations of Dantzig's pivot rule.

In order to apply the above remark, we state and prove an elementary lemma on linear convergence.

LEMMA 2. Suppose that z^k is the objective value of the basic feasible solution for the network flow problem (1) subsequent to the k-th pivot. Suppose further that there is a real number α with $0 < \alpha < 1$ and $z^{k+1} \le z^k - \alpha\,(z^k - z^*)$ for all $k \ge 1$, where z^* is the minimum objective value. Then the number of pivots is $O(\alpha^{-1} \log w^*)$.

PROOF. Let us assume the hypothesis of the Lemma, i.e., $(z^{k+1} - z^*) \le (1 - \alpha)(z^k - z^*)$. Inductively, it follows that $(z^{k+1} - z^*) \le (1 - \alpha)^k(z^1 - z^*) \le (1 - \alpha)^k\, x^*$. Moreover, by the integrality of z^k and z^*, if $z^{k+1} - z^* < 1$, then $z^{k+1} = z^*$. Hence, the number of pivots is at most $\log w^*/(-\log(1 - \alpha))$, which is $O(\alpha \log w^*)$. □

THEOREM 3. Suppose that the simplex method using Dantzig's pivot rule is applied to the minimum cost network flow problem (1). Then the number of pivots is $O(mnu^* \log w^*)$.

PROOF. We assume that we start with a strongly feasible basis in Phase 2. The result for Phase 1 is a special case since the Phase 1 problem is also a minimum cost network flow problem.

Let $g_i(\Theta) = \Theta$ for $i = 1, \ldots, m$ and let $P(\Theta)$ be the parametric program $\min(cx : Ax = b - g(\Theta),\ \ell \le x \le u)$. We first claim that the original problem $P(0)$ is equivalent to the problem $P((m + 1)^{-1})$. To see this we first note that by Theorem 1, $P(0)$ is equivalent to $P(\Theta)$ for all sufficiently positive Θ. We next note that by the unimodularity of each basis B,

$$0 \le || B^{-1} g(\Theta)|| < m \Theta ,$$

where $||\cdot||$ denotes the sup norm. By the above and the integrality of ℓ, u and b, we may choose $\Theta = 1/(m + 1)$.

Without loss of generality, assume that the basis prior to the $(k + 1)$-st pivot is canonically oriented. Then the entering variable increases its value by at least $(m + 1)^{-1}$, since no basic solution for $P((m + 1)^{-1})$ is degenerate and B is unimodular. Thus

$$z^{k+1} - z^k \le (m + 1)^{-1} \bar{c}_{min} . \tag{6}$$

Moreover, by relaxing the constraints "Ax = b", we see that

$$z^* - z^k \ge \sum_{j=1}^{n} (u_j - \ell_j) \bar{c}_j \ge n u^* \bar{c}_{min}. \tag{7}$$

Combining (6) and (7), we obtain the inequality

$$z^{k+1} - z^k \le ((m + 1) n u^*)^{-1} (z^* - z^k),$$

and the result then follows from Lemma 2. □

As a corollary, the number of pivots for the shortest path problem and for the optimal assignment problem are both bounded by a polynomial in the data. Similarly, the number of consecutive degenerate pivots is polynomially bounded since we can restrict attention to problems for which $u^* \le 1$, as described below. In order to show that the number of pivots is polynomially bounded in m and n independent of c, we show how we may restrict attention to problems in which the costs are "small".

LEMMA 3. Let P be the minimum cost network flow problem $\min(cx : Ax = b, \ell \le x \le u)$. Then there is a vector c' with $||c'|| \le 4^n (n!)^2$ such that the network flow problem $\min(c'x : Ax = b, \ell \le x \le u)$ is equivalent to P.

PROOF. Let (B, N_1, N_2) be a feasible basis that is canonically oriented, and let T be the corresponding tree. For each $e \in E - T$, let $C(T, e)$ be the circuit created upon adding edge e to T oriented in the same direction as e, and let $c(T, e)$ be the cost of the circuit. It is well known that the reduced cost of edge e with respect to the spanning tree solution induced by T is $c(T, e)$. Suppose that $S = \{C_1, C_2, \ldots, C_t\}$ is the set of all circuits in G that have a non-negative cost. (Thus every circuit of G is in S or its reversal is in S, or both are in S if their cost are 0). Suppose further that the C's are arranged in non-decreasing order of cost. Then a sufficient condition for the network flow problem $\min(c'x : Ax = b, \ell \le x \le u)$ to be equivalent to P is that:

$$\text{If } c(C_1) = 0 \text{ then } c'(C_1) = 0 . \tag{8}$$

$$\text{If } c(C_1) > 1 \text{ then } c'(C_1) \ge 1. \tag{9}$$

and for $j = 2, \ldots, t$ the following are true

$$\text{If } c(C_j) = c(C_{j-1}) \text{ then } c'(C_j) = c'(C_{j-1}). \tag{10}$$

$$\text{If } c(C_j) > c(C_{j-1}) \text{ then } c'(C_j) \ge c'(C_{j-1}) + 1. \tag{11}$$

If we consider $c'_1, \ldots, c'_n$ as linear variables, then the constraints (8)-(11) may be written as a system of linear equalities and inequalities in which the constraint matrix coefficients are all 0, 1, -1, 2, or -2 and where the right hand side coefficients are 0 or 1.

Suppose first that the feasible region induced by (8)-(11) has at least one corner point. Then any such feasible corner point solution may be expressed as $c'' = D^{-1}d$, where D is a submatrix of the constraint matrix and d is a subvector of the right-hand-side vector. Using Kramer's rule and the fact that the determinant of any submatrix of D is at most $2^n\ n!$, the numerator of c'' and the common denominator of the components of c'' are both bounded by $2^n\ n!$ Multiplying through by the common denominator, we obtain an integral vector c' satisfying (8)-(11) such that $||c'|| \le 4^n\ (n!)^2$.

In the case that the feasible region has no corner points, we may add a constraint of the form $c'_j = 0$ so as to maintain feasibility. If we iterate in this way we eventually obtain a system with corner points, and the argument then reduces to the above.□

THEOREM 4. Let P be the minimum cost network flow problem $\min(cx : Ax = b,\ \ell \le x \le u)$. Suppose further that Dantzig's pivot rule is applied to P. Then the maximum number of consecutive degenerate pivots is $O(m^2 n \log n)$ [also, $O(m^2 \log w^*)$]. Moreover, if the network flow problem is either the optimal assignment problem or the shortest path problem then the number of pivots is $O(m^2 n \log n)$ [also $O(m^2 \log w^*)$].

PROOF. Let us first consider the maximum number of consecutive degenerate pivots. Let (B, N_1, N_2) be the current strongly feasible basis, and without loss of generality let us assume that the basis is canonically oriented. Suppose that x^* is the current basic solution. Let us consider a related problem $P' = \min(cx : Ax = b,\ \ell' \le x \le u')$ where ℓ' and u' are defined as follows: $\ell'_j = x^*_j$ and $u'_j = x^*_j + 1$. ($u'_j \le u_j$ because the basis is canonically oriented). Then any non-degenerate pivot for P is also a non-degenerate pivot for P'. Thus we have shown that we can replace u^* by 1.

Consider the problem $P'(\Theta)$ where $g_j(\Theta) = \Theta$ as in the proof of Theorem 3, and let z^* denote the minimum objective value that can be obtained such that no variable x_j is at its upper bound. (Such a variable could only have resulted from a non-degenerate pivot). Thus there are at most m basic variables, and

$$z^* - z^k \geq m \, \overline{c}_{min}. \tag{12}$$

Combining (6) and (12), we obtain the inequality

$$z^{k+1} - z^k \leq [(m + 1)m]^{-1} \, (z^* - z^k),$$

and by Lemma 2 the number of consecutive degenerate pivots is $O(m^2 \log w^*)$. Moreover, we may replace w^* by $(\Sigma|c_i'| : 1 \leq i \leq n)$ for the c' described in Lemma 3. Thus the number of pivots is $O(m^2 n \log n)$.

Next we consider the shortest path problem. In this case we assume that the problem is written as: $(\min cx : Ax = 1, \; 0 \leq x_j \leq m + 1$ for $1 \leq j \leq n)$. In this case, every feasible basis is strongly feasible so that

$$z^{k+1} - z^k \leq \overline{c}_{min}. \tag{13}$$

In addition, $u^* = (m + 1)$ and no variable is ever at its upper bound unless there is a negative cost circuit. Thus each non-basic variable has value 0, and

$$z^* \quad z^k \geq \; m \, u^* \, \overline{c}_{min} = (m^2 + m) \, \overline{c}_{min} \tag{14}$$

Combining (13) and (14) and applying Lemma 2, we obtain that the number of pivots is $O(m^2 \log w^*)$. Then by Lemma 3, the number of pivots is $O(m^2 n \log n)$.

Finally we consider the optimal assignment problem. Here, $u^* = 1$, and each non-basic variable is at its lower bound. Thus

$$z^* - z^k \geq m \, \overline{c}_{min}. \tag{15}$$

Combining (6) and (15) and applying Lemma 2, we obtain that the number of pivots is $O(m^2 \log w^*)$. By applying Lemma 3, we see that the number of pivots is $O(m^2 n \log n)$.□

Some Concluding Remarks

The proof of Theorem 3 relies on (1) the equivalence of the original problem to the perturbed problem and (2) each non-degenerate pivot with Dantzig's pivot rule leads to a geometric improvement towards the optimal solution. Hung (1983) developed a different polynomial time pivoting procedure for the optimal assignment problem by exploiting property (2) above. His procedure performs as many consecutive degenerate pivots as possible followed by a non-degenerate $\bar{c}_{min}$ pivot.

Professor G.B. Dantzig (1983) independently proved that his simplex rule in conjunction with Cunningham's strongly feasible pivot rule converges geometrically to the optimum solution. His argument is essentially the same as the one given here for the proof of Theorem 3.

Although the above worst-case analyses are based on the use of Dantzig's pivot rule, we may relax this restriction substantially. For example, the same bound is valid if we consider an alternative pivot rule in whi the $\bar{c}_{min}$ - variable is selected at least once every k pivots for some fixed k. Also, the same bound is valid if the entering variable x_j is such that $k|\bar{c}_j| < |\bar{c}_{min}|$ for some fixed k.

Finally we note that the worst case analysis is unduly pessimistic and in no way reflects the average performance of the simplex method. In a recent very elegant paper, Haimovich (1983) gave a detailed probabilistic analysis of a variant of the simplex algorithm and proved that under suitable probabilistic assumptions the average number of pivots is $O(n)$.

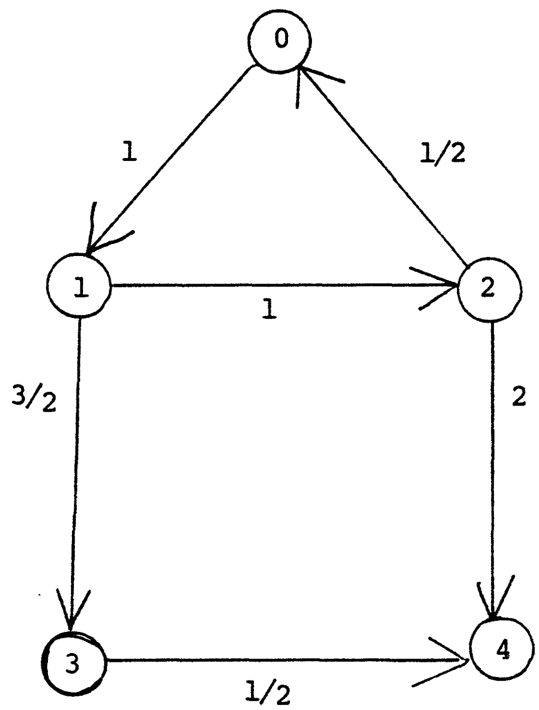

Figure 1. The graph G(A) induced by matrix A of Table 1.

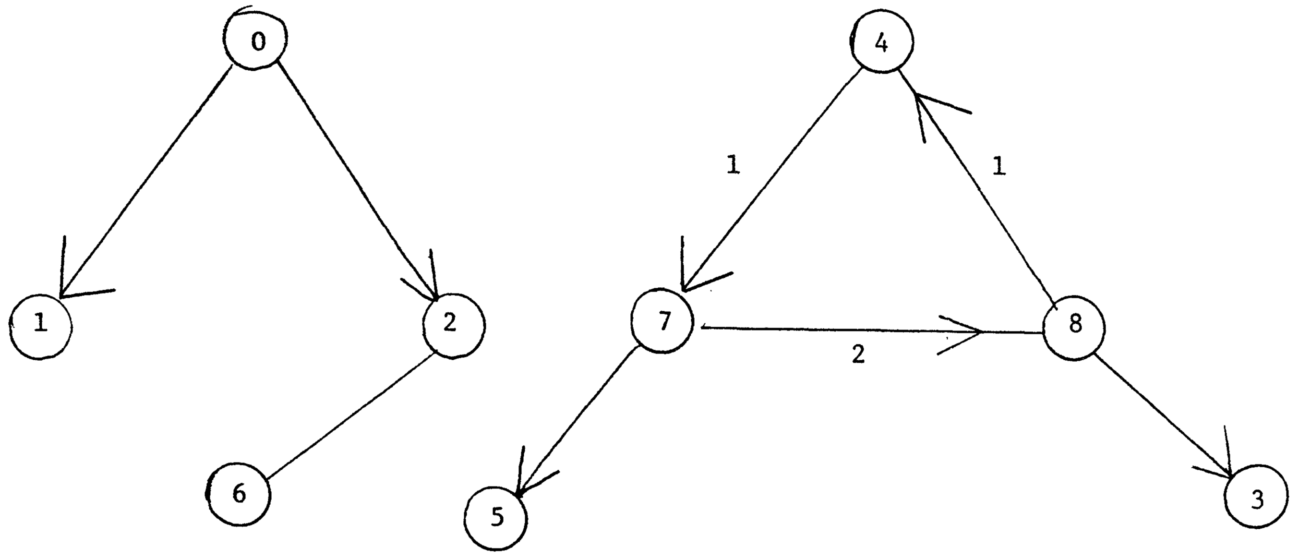

Figure 2. A canonically oriented basis.

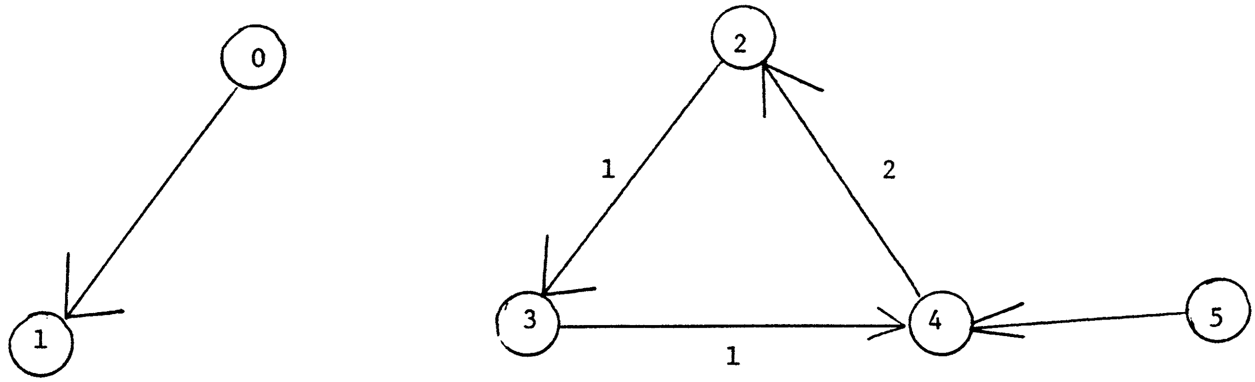

Figure 3a

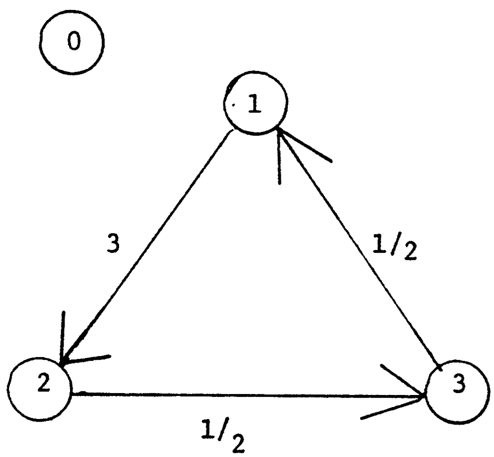

Figure 3b

Figures 3a and 3b. Bases that are not cononically oriented. In 3a, edge (5,4) is oriented incorrectly. In 3b the cycle has a flow multiplier that is less than 1.

ACKNOWLEDGMENTS

I wish to thank John VandeVate for his suggestions, which led to improvement in the exposition.

REFERENCES

Barr, R., F. Glover, and D. Klingman (1977). The alternating basis algorithm for assignment problems. Mathematical Programming 13, 1-13.

Cunningham, W. H., (1976). A network simplex method. Mathematical Programming 11, 105-116.

Cunningham, W.H., (1979). Theoretical properties of the network simplex method. Mathematics of Operations Research 4, 1976-208.

Dantzig, G. B. (1963). Linear Programming and Extensions, Princeton University Press, Princeton, N.J.

Dantzig, G. B. (1983). Personal communication.

Edmonds, J. (1970). Unpublished manuscript.

Elam, J., F. Glover and D. Klingman (1979). A strongly convergent primal simplex algorithm for generalized networks. Mathematics of Operations Research 4, 39-59.

Haimovich, M. (1983). The simplex algorithm is very good! - On the expected number of pivot steps and related properties of random linear programs. Unpublished manuscript.

Hung, M. (1983). A polynomial simplex method for the assignment problem. Operations Research 31, 595-600.

McBride, R. (1981). Efficient solution of generalized network problems. Unpublished manuscript.

Orlin, J. (1983). A polynomial time parametric simplex algorithm for the minimum cost network flow problem. Unpublished manuscript.

Veinott, A.F., Jr. (1968). Extreme points of Leontief Substitution Systems. Linear Algebra and its Applications 1, 181-194.

Zadeh, N. (1973). A bad network flow problem for the simplex method and other minimum cost flow algorithms. Mathematical Programming 5, 255-266.

Zadeh, N. (1979). Near-Equivalence of network flow algorithms. Technical Report 26, Department of Operations Research, Stanford University, Stanford, CA.