ON THE SIMPLEX METHOD USING AN ARTIFICIAL BASIS

Markku Kallio

April 1979
WP-79-25

INTERNATIONAL INSTITUTE FOR APPLIED SYSTEMS ANALYSIS
A-2361 Laxenburg, Austria

ACKNOWLEDGEMENT

The author is grateful to Dr. Anatoli Propoi for helpful discussions and valuable suggestions.

SUMMARY

The use of an artificial basis for the simplex method was suggested in an early paper by Dantzig. The idea is based on an observation that certain bases, which differ only in a relatively few columns from the true basis, may be easily inverted. Such artificial bases can then be exploited when carrying out simplex iterations. This idea was originally suggested for solving structured linear programming problems, and several approaches, such as Beale's method of pseudo-basic variables, have indeed been presented in the literature.

In this paper, we shall not consider the structure explicitly; rather its exploitation in our case is expected to result directly from the choice of an artificial basis. We shall consider this basis to remain unchanged over a number of simplex iterations. In particular, this basis may be chosen as the true basis which has been most recently reinverted. In such a case our approach yields an interpretation for a basis representation recently proposed by Bisschop and Meeraus who point out very favorable properties regarding the build-up of nonzero elements in the basis representation.

Our approach utilizes an auxiliary basis, which is small relative to the true basis, and whose dimension may change from

one iteration to another. We shall finally develop an updating scheme for a product form representation of the inverse of such an auxiliary basis.

ON THE SIMPLEX METHOD USING AN ARTIFICIAL BASIS

Markku Kallio

1. Introduction

For solving certain structured linear programming problems via the simplex method, Dantzig [4] suggests the use of an *artificial basis* and a *true basis*. The power of using an artificial basis results from its choice so that its inverse is easy to obtain compared to that of the true basis, and so that these two bases differ only in a relatively few columns. In particular, it is suggested that this approach is appropriate when the constraint matrix of a linear program is block triangular, such as in the case of a dynamic linear program, for instance. The artificial basis may then be chosen as a square block triangular matrix. Beale [1] has adopted Dantzig's suggestion for the block-angular linear program. In this case, the artificial basis can be chosen to be a square block diagonal matrix.

In this paper we shall consider a linear program which may or may not have a special structure. Our main goal is to show how the computations may be carried out, in general, when an artificial basis is being used. Although we shall not consider any special structure explicitly, the exploitation of a structure results directly from an appropriate choice for the artificial basis. The artificial basis is assumed to remain unchanged during a number of consequent iterations. Several strategies may be adopted for changing the artificial basis. In particular, the most recently reinverted true basis may play the role of an artificial basis. In this case, our approach yields the basis representation which Bisschop and Meeraus [2] derive from a matrix augmentation and partitioning approach.

Let k be the number of columns in which the artificial basis differs from the true basis in our approach. Following Beale [1], we shall call pseudo-basic the k variables in the artificial basis which are not in the true basis. When choosing an artificial basis, it is desirable that k is small. Thereafter, until the subsequent choice, the number of pseudo-basic variables cannot increase by more than one at each simplex iteration.

For carrying out simplex iterations, we shall introduce yet an auxiliary basis, which has as many columns (and rows) as there are pseudo-basic variables. Thus, the auxiliary basis is likely to be small compared with the true basis. As the number of pseudo-basic variables may also decrease, the dimension of an auxiliary basis may increase, decrease or remain the same. In particular, at each iteration (until redefinition of the artificial basis) the auxiliary basis changes so that either a row or a column changes, a row and column is deleted, or a row and a column is appended to the auxiliary basis. In the final part of this paper we shall show how the inverse of this basis can be updated in a product form framework.

2. Preliminaries

Consider the linear programming problem (LP):

find $x \in R^n$ to

(LP1) maximize cx,

(LP2) subject to $Ax = b$,

(LP3) $x \geq 0$,

where $c = (c_j) \in R^n$, $A = (a^j) \in R^{m\times n}$, and $b \in R^m$. Here c_j and a^j are the j^{th} element of vector c and column of matrix A, respectively. We assume that A is of full row rank. For the sake of simplicity, we assume also that (LP) is nondegenerate: for any basis B, all elements of $B^{-1}b$ are nonzero.

We shall consider the revised simplex method [3] for (LP). A main difficulty then arises in how the basis inverse should be represented and updated along the iterations. In the following we develop a basis representation, where, instead of the inverse of the true basis, the inverse of a small auxiliary basis is updated at each simplex iteration. The computations needed for an iteration are then carried out using the inverse of the auxiliary basis as well as the inverse of an artificial basis. The latter one remains unchanged from one iteration to another (until a new artificial basis is chosen).

For expository purposes, we shall adopt the following point of view from reference [5] on the simplex method:

A system solution satisfies (LP2), a homogeneous solution satisfies $Ax = 0$ and a feasible solution satisfies (LP2) and (LP3). If x is a feasible solution and z is a homogeneous solution, then $x + \theta z$ is feasible as long as it is non-negative, for $\theta \in R$. As θ increases, the objective function increases if and only if $cz > 0$. The simplex method chooses as z one of the vectors corresponding to (changing the value of) a nonbasic variable, and $cz > 0$ is the reduced cost for that variable. The new feasible solution $x + \bar{\theta} z$ is found by increasing θ (and the objective function) as much as possible before violating the nonnegativity constraint. Provided that $\bar{\theta}$ is bounded, the new feasible solution is a basic solution.

We shall now consider how the nonbasic variables are priced out and how such a homogeneous solution z is computed using an artificial basis and an auxiliary basis.

3. A Simplex Iteration

Consider an iteration of the simplex method. Let B be the current artificial basis and β the set of the basic variables corresponding to B. We shall not distinguish between a variable and its index. Thus β as well refers to the set of indices of basic variables. Let γ be the set of current basic variables, G the current true basis, and x the corresponding basic feasible solution. Denote by $\bar{\beta}$ and $\bar{\gamma}$ the complements of β and γ respectively. We have $x_i > 0$ for $i \in \gamma$ and $(B^{-1}b)_i > 0$ for $i \in \beta$ (by nondegeneracy) and $x_i = 0$ for $i \in \bar{\gamma}$. In particular $x_i = 0$ for $i \in \beta \cap \bar{\gamma}$, representing the set of pseudo-basic variables. Similarly, $\gamma \cap \bar{\beta}$ is the set of variables which are in the true basis but not in the artificial basis. Let k be the number of elements in this set. Also the number of elements in the set of pseudo-basic variables $\beta \cap \bar{\gamma}$ is k. For practical applications we may assume that $0 \leq k < m$. This will be guaranteed by an appropriate strategy of choosing the artificial basis.

For convenience (without loss of generality), we may assume that

$$\bar{\gamma} \cap \beta = \{1,2,\ldots,k\} \quad ,$$
$$\gamma \cap \beta = \{k+1,\ k+2,\ldots,m\} \quad ,$$
$$\gamma \cap \bar{\beta} = \{m+1,\ m+2,\ldots,m+k\} \quad , \text{ and}$$
$$\bar{\gamma} \cap \bar{\beta} = \{m+k+1,\ldots n\} \quad .$$

The situation may then be depicted as in Figure 1, which shows the constraint matrix A multiplied by the inverse of B. For convenience, we shall now first consider the computation of the direction vector and thereafter the pricing operation needed in a simplex iteration.

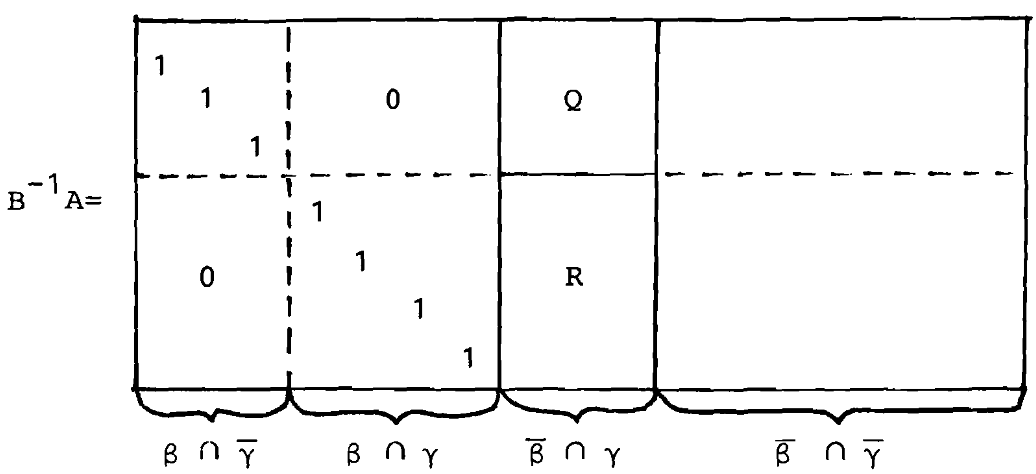

Figure 1

Computation of Direction z

For $j \in \bar{\beta}$, define $\alpha^j \equiv B^{-1}a^j$. Thus, α^j is the alfa-column of variable j corresponding to the artificial basis B. Define an n-vector z^j componentwise as follows:

$$z_i^j = \begin{cases} -\alpha_i^j & , \quad \text{of } i \in \beta \ , \\ 1 & , \quad \text{if } i = j \ , \\ 0 & , \quad \text{otherwise} \ . \end{cases} \tag{1}$$

Then $Az^j = B(-B^{-1}a^j) + a^j = 0$; i.e. z^j is a homogeneous solution. In particular, for the simplex method, z^j is the direction of change in the solution which corresponds to the basis B and an increase in the nonbasic variable $j \in \bar{\beta}$. In this case, $d^j = cz^j$ is the reduced cost for variable j corresponding to the artificial basis.

Let $e \in \bar{\gamma}$ be the variable to be increased during the current iteration. The pricing operation needed for determining e will be discussed below. For the current solution x, we define z as the direction of change as follows:

$$z = z^e + \sum_{j \in \bar{\beta} \cap \gamma} w_j z^j \ , \tag{2}$$

where $w = (w_j)$ is a k-vector of weights. Notice, that e may be in $\beta \cap \bar{\gamma}$. In this case we define $z^e \equiv 0$. Clearly, $A(x + \theta z) = b$, i.e. $x + \theta z$ is a system solution, for any $\theta > 0$. However, $x + \theta z$ may not be a feasible solution for any $\theta > 0$. Thus, further restrictions on w are needed to guarantee that z is a feasible direction. (For such a discussion, see reference [5]). In particular, the simplex method requires that (besides e) only currently basic variables $i \in \gamma$ may be changed when moving in direction z. Thus, we require that $z_i = 0$ for current nonbasic variables $i \in \bar{\gamma}$ such that $i \neq e$ (the nonbasic variable to be increased). Notice that this requirement is already satisfied for $i \in \bar{\beta} \cap \bar{\gamma}$ by definition of z in (1) and (2).

As illustrated in Figure 1, we define a k×k-matrix Q as follows:

$$Q = (\alpha_i^j : i \in \beta \cap \overline{\gamma}, j \in \overline{\beta} \cap \gamma) \quad . \tag{3}$$

Then, we have the following result:

Lemma 1. Q is nonsingular.

Proof: According to Figure 1, define

$$R = (\alpha_i^j : i \in \beta \cap \gamma, j \in \overline{\beta} \cap \gamma) \quad .$$

Then

$$B^{-1}G = \begin{pmatrix} O & Q \\ I & R \end{pmatrix} \quad ,$$

is nonsingular because B and G are nonsingular. Thus, Q is nonsingular. ||

We may call Q an auxiliary basis and define

$$w = Q^{-1}\overline{\alpha}^e \quad , \tag{4}$$

where the k-vector $\overline{\alpha}^e \equiv (\alpha_i^e : i \in \beta \cap \overline{\gamma})$. In this notation we have

Lemma 2. If $w = (w_j)$ is defined by (4), then z defined by (2) is a homogeneous solution for which $z_e = 1$ and $z_i = 0$ for all $i \in \overline{\gamma}$ such that $i \neq e$; i.e. z is the direction of change in the solution which corresponds to an increase in the nonbasic variable $e \in \overline{\gamma}$ during the simplex iteration.

Proof: For $i \in \beta \cap \overline{\gamma}$, we have

$$z_i = \overline{z} + QQ^{-1}\overline{\alpha}^e = \begin{cases} 0 \quad , & \text{if } i \neq e \quad , \\ 1 \quad , & \text{if } i = e \quad , \end{cases} \tag{5}$$

where $\overline{z}$ comprises the k first components of z^e. For $i \in \overline{\beta} \cap \overline{\gamma}$, we have by definition $z_i = 0$ for $i \neq e$, and $z_i = 1$ if $i = e$. Thus z is a homogeneous solution, for which the nonbasic component e is equal to 1 and all the other nonbasic components z_i, $i \in \overline{\gamma}$, are equal to zero. The result then follows from the uniqueness of such a homogeneous solution. ||

We define $E = (a^j : j \in \bar{\beta} \cap \gamma)$ as the matrix of columns in A corresponding to the basic variables which are not in the artificial basis, and we carry out the computations needed to obtain z as follows:
If $e \in \bar{\beta} \cap \bar{\gamma}$, compute $\alpha^e = B^{-1}a^e$ using the pseudo-basis B and the entering column a^e. For $e \in \beta \cap \bar{\gamma}$, $\alpha^e = B^{-1}a^e$ is a unit vector needing no computation. Let $\bar{\alpha}^e$ consist of the k first elements of α^e, as above, and compute $w = Q^{-1}\bar{\alpha}^e$, using the auxiliary basis Q. Carry out multiplication Ew. Thereafter, the direction of change in basic variables $\beta \cap \gamma$ can be obtained from $-\alpha^e + B^{-1}(Ew)$ as the last m-k components of this vector. The vector w indicates the change for variables i, for $i \in \bar{\beta} \cap \gamma$.

Pricing Out Nonbasic Variables

We shall next consider the computation of the reduced cost for a nonbasic variable $j \in \bar{\gamma}$. As noted before, $d_j = cz^j = c_j - \pi a^j$ is the reduced cost of j corresponding to the artificial basis. According to (2), when moving in the direction z, for $e = j \in \bar{\gamma}$, the rate of change in the objective function is $cz = d_j + \sum_{i \in \bar{\beta} \cap \gamma} w_i d_i$. If w is given by (4), we have, by Lemma 2, for the reduced cost of $j \in \bar{\gamma}$ corresponding to the true basis G,

$$cz = d_j - dQ^{-1}\bar{\alpha}^j \quad , \tag{6}$$

where the k-vector $d \equiv (d_i : i \in \bar{\beta} \cap \gamma)$ is the vector of reduced costs (corresponding to B) for $\bar{\beta} \cap \gamma$. We denote

$$\mu = dQ^{-1} \quad . \tag{7}$$

Thus (6) becomes

$$cz = dj - \mu\bar{\alpha}^j = d_j - (\mu, 0)B^{-1}a^j = d_j - \pi' a^j \quad , \tag{8}$$

where 0 is an (m-k)-vector of zeros and

$$\pi' = (\mu, 0) B^{-1} \quad . \tag{9}$$

Here π' may be interpreted as the correction to be made in the price vector π corresponding to B to obtain the price vector corresponding to G.

Thus, the computations needed for pricing out the nonbasic variables $j \in \bar{\gamma}$ are as follows:

Compute μ according to (7). For $j \in \beta \cap \bar{\gamma}$, notice that $\bar{\alpha}^j$ is the j^{th} unit vector and $d_j = 0$. Thus by (8), $cz = -\mu_j$ in this case. For $j \in \bar{\beta} \cap \bar{\gamma}$, we first compute π' according to (9) to obtain the current price vector $\pi + \pi'$ corresponding to G. As usual, we then compute $cz - c_j = (\pi + \pi')a^j$.

4. A Product Form of Inverse for the Auxiliary Basis

At the end of the simplex iteration, the entering variable $e \in \bar{\gamma}$ replaces the leaving variable ℓ in γ. For updating the basis representation, we merely have to update the inverse of the auxiliary basis Q. Bisschop and Meeraus [2] give updating formulas when Q^{-1} is stored explicitly. Indeed, if the number of pseudo-basic variables remains truly small, an explicit representation can be appropriate. In the following, however, we shall develop a product form representation for Q^{-1}. For this purpose, we assume that at the current iteration we have a $k' \times k'$-matrix D, for some $k' \geq k$, given in a product form such that

$$D = \begin{bmatrix} Q^{-1} & 0 \\ 0 & I \end{bmatrix} \quad , \tag{10}$$

where I is a $(k'-k) \times (k'-k)$ identity matrix. Using the product form representation of D, the computation of w and μ (according to (4) and (7)) can be done as follows:

$$(w^T, 0) = ((\bar{\alpha}^e)^T, 0) D^T \quad \text{and} \quad (\mu, 0) = (d, 0) D \quad . \tag{11}$$

In one iteration the dimension of the auxiliary basis Q may increase, remain the same or decrease. However, we do not let the dimension of D decrease; it either remains unchanged or it increases (if the dimension of Q increases). We shall now consider separately the four possible cases for updating the inverse representation (10) for Q^{-1}. We refer by $\bar{Q}$ and $\bar{D}$ to the updated auxiliary basis and its representation, respectively.

$e \in \bar{\beta}, \ \ell \in \bar{\beta}$. In this case, the column $\bar{\alpha}^e$ replaces column $\bar{\alpha}^\ell$ in Q. Let $\bar{\alpha}^\ell$ be the r^{th} column of Q. As usual (see e.g. [3]), D may now be updated through premultiplication by an elementary (column) matrix E, whose r^{th} column η is given componentwise by

$$\eta_i = \begin{cases} -w_i/w_r \ , & \text{for} \quad 1 \leq i \leq k \quad \text{and} \quad i \neq e \ , \\ 1/w_r \ , & \text{for} \quad i = e \ , \\ 0 \ , & \text{otherwise} \ . \end{cases} \tag{12}$$

For $\bar{D}$ and $\bar{Q}$, the updated matrices D and Q, respectively, we have

$$\bar{D} = ED = \begin{bmatrix} \bar{Q}^{-1} & 0 \\ 0 & I \end{bmatrix} ,$$

where I again is a $(k'-k)\times(k'-k)$ identity matrix. Dimensions k and k' remain unchanged in this case.

$e \in \beta, \ \ell \in \beta$. In this case, row e of Q is replaced by $\bar{\rho}$, the ℓ^{th} row of $\binom{Q}{R}$. First we have to compute $\rho = \bar{\rho}Q^{-1}$. In practice, this may be done using the following formula:

$$\rho = I_\ell B^{-1} E Q^{-1} \ , \tag{13}$$

where I_ℓ is the ℓ^{th} unit (row) vector. D is now updated through post-multiplication by an elementary (row) matrix, whose e^{th} row η is given componentwise as follows:

$$\eta_i = \begin{cases} -\rho_i/\rho_e \ , & \text{for} \quad 1 \leq i \leq k \quad \text{and} \quad i \neq e \ , \\ 1/\rho_e \ , & \text{for} \quad i = e \ , \\ 0 \ , & \text{otherwise} \ . \end{cases} \tag{14}$$

Again, if $\bar{D}$ and $\bar{Q}$ are updated matrices, then

$$\bar{D} = DE = \begin{bmatrix} \bar{Q}^{-1} & 0 \\ 0 & I \end{bmatrix} ,$$

and the dimensions k and k' remain unchanged.

$e \in \bar{\beta}$, $\ell \in \beta$. One column and row is now appended to both D and Q. For the purpose of simplifying notation, we shall assume such row and column permutations that the appended row and column are the $(k+1)^{th}$ row and column, respectively, in the updated matrices $\bar{D}$ and $\bar{Q}$. Accordingly, to bring D to the same dimension as $\bar{D}$, we assume that the $(k+1)^{th}$ unit row and column vector is (implicitly) appended to these positions in all elementary matrices of the current product form of D. $\bar{D}$ results now from multiplying D by two elementary matrices E_1 and E_2. For defining these matrices, let ρ be given as in (13), let $\bar{\delta} = \bar{\alpha}^e_\ell$, and define a row vector η^1 and a column vector $\bar{\eta}^2$ componentwise as follows:

$$\eta^1_i = \begin{cases} -\bar{\rho}_i , & \text{for } 1 \leq i \leq k , \\ 1 , & \text{for } i = k+1 , \\ 0 , & \text{otherwise} , \end{cases} \tag{15}$$

$$\eta^2_i = \begin{cases} -w_i/\delta , & \text{for } 1 \leq i \leq k , \\ 1/\delta , & \text{for } i = k+1 , \\ 0 , & \text{otherwise} , \end{cases} \tag{16}$$

where $\delta = \bar{\delta} - \bar{\rho}w$.

For determining $\bar{D}$, the updated representation (10) for $\bar{Q}^{-1}$, we have the following result:

Lemma 3. Let E_1 and E_2 elementary matrices so that η^1 in (15) is the $(k+1)^{th}$ row of E_1 and η_2 in (16) is the $(k+1)^{th}$ column of E_2. Then

$$\bar{D} = E_2E_1D = \begin{bmatrix} \bar{Q}^{-1} & 0 \\ 0 & I \end{bmatrix} ,$$

where I is the $(k'-k)\times(k'-k)$ unit matrix.

Proof: By our notation, the updated auxiliary basis is

$$\bar{Q} = \begin{bmatrix} Q & \bar{\alpha}^e \\ \bar{\rho} & \bar{\delta} \end{bmatrix} . \tag{17}$$

Thus, we have

$$E_2E_1D \left[\begin{array}{cc|c|c|c} Q & \bar{\alpha}^e & 0 & I & 0 \\ \bar{\rho} & \bar{\delta} & & & \\ \hline \multicolumn{2}{c|}{0} & I & 0 & I \end{array}\right] = E_2E_1 \left[\begin{array}{cc|c|cc|c} I & w & 0 & Q^{-1} & 0 & 0 \\ \bar{\rho} & \bar{\delta} & & & 1 & \\ \hline \multicolumn{2}{c|}{0} & I & \multicolumn{2}{c|}{0} & I \end{array}\right]$$

$$= E_2 \left[\begin{array}{cc|c|cc|c} I & w & 0 & Q^{-1} & 0 & 0 \\ 0 & \delta & & -\bar{\rho}Q^{-1} & 1 & \\ \hline \multicolumn{2}{c|}{0} & I & \multicolumn{2}{c|}{0} & I \end{array}\right] = \left[\begin{array}{cc|c|cc|c} I & 0 & 0 & (I+w\bar{\rho}/\delta)Q^{-1} & -w/\delta & 0 \\ 0 & 1 & & -\bar{\rho}Q^{-1}/\delta & 1/\delta & \\ \hline \multicolumn{2}{c|}{0} & I & \multicolumn{2}{c|}{0} & I \end{array}\right]$$

$$= \left[\begin{array}{c|c|c|c} I & 0 & \bar{Q}^{-1} & 0 \\ \hline 0 & I & 0 & I \end{array}\right] = (I,\bar{D}) , \tag{18}$$

(where again $\delta = \bar{\delta} - \bar{\rho}w$). Thus, $E_2E_1D = \bar{D}$. ||

<u>$e \in \beta,\ \ell \in \bar{\beta}$</u>. In this final case, the e^{th} row of Q and the column corresponding to the leaving column ℓ has to be deleted from Q. Although the dimension of Q now decreases, we shall leave the dimension of D unchanged. Again, for the purpose of simplifying notation, we assume that $e = k$ and ℓ is the last column of Q. We define an elementary (row) matrix T_1 and an elementary (column) matrix T_2 for which the k^{th} row η^1 and column η^2, respectively, are defined componentwise as follows:

$$\eta_i^1 = \begin{cases} \bar{\rho}_i , & \text{for } 1 \le i \le k , \\ 1 , & \text{for } i = k , \\ 0 , & \text{otherwise} , \end{cases} \tag{19}$$

$$\eta_i^2 = \begin{cases} -w_i/w_k , & \text{for } 1 \leq i \leq k , \\ 1/w_k , & \text{for } i = k , \\ 0 , & \text{otherwise} . \end{cases} \tag{20}$$

The following result completes the discussion of updating the product form representation of an auxiliary basis:

Lemma 4. Let T_1 and T_2 be elementary matrices so that η^1 in (19) is the k^{th} row of T^1 and η^2 in (20) is the k^{th} column of T_2. Then for the updated representation

$$\bar{D} = T_1 T_2 D = \begin{bmatrix} \bar{Q}^{-1} & 0 \\ 0 & I \end{bmatrix} ,$$

where I is a $(k'-k+1) \times (k'-k+1)$ identity matrix.

Proof: For the proof, we may refer to (17) and (18) where Q, $\bar{Q}$, D, $\bar{D}$ and k now play the role of $\bar{Q}$, Q, $\bar{D}$, D and $k-1$, respectively. Because $\bar{\alpha}^e$, in this case, is a unit vector $w = Q^{-1}\bar{\alpha}^e$ is the k^{th} column of Q^{-1} corresponding to the vector $\begin{bmatrix} -w/\delta \\ 1/\delta \end{bmatrix}$ in (18). Thus, T_2 is the unique elementary (column) matrix E such that the k^{th} column of ED is the k^{th} unit vector. Thereafter, T_1 $(=E_1^{-1}$ in (18)) is the unique elementary (row) matrix E such that the k^{th} row of ET_2D is the k^{th} unit vector. Thus $T_1T_2D = \bar{D}$. ||

REFERENCES

[1] E.M.L. Beale (1962) The Simplex Method Using Pseudo-basic Variables for Structured Linear Programming Problems. Recent Advances in Mathematical Programming, edited by R.L. Graves and P. Wolfe. McGraw-Hill, New York, 133-148.

[2] J. Bisschop and A. Meeraus (1977) Matrix Augmentation and Partitioning in the Updating of the Basic Inverse. Mathematical Programming 13:241-254.

[3] G.B. Dantzig (1963) Linear Programming and Extensions. Princeton University Press, Prenceton.

[4] G.B. Dantzig (1955) Upper Bounds, Secondary Constraints and Block Triangularity in Linear Programming. Econometrica 23:174-183.

[5] M. Kallio and E. Porteus (1978) A Class of Methods for Linear Programming. Mathematical Programming 14:161-169.