JOURNAL OF APPROXIMATION THEORY **20**, 57–69 (1977)

Minimum L_∞ Solution Of Underdetermined Systems Of Linear Equations

NABIH N. ABDELMALEK

National Research Council, Ottawa, Ontario, Canada

Communicated by E. W. Cheney

Received August 20, 1975

The problem of obtaining a minimum L_∞ solution of an underdetermined system of consistent linear equations is reduced to a linear programming problem. A modified simplex algorithm is then described. In this algorithm no conditions are imposed on the coefficient matrix, minimum computer storage is required and no artificial variables are needed. The algorithm is a simple and fast one. Numerical results are given.

1. INTRODUCTION

Consider the underdetermined system of linear equations

$$Ca = f, \tag{1}$$

where $C = (c_{ij})$ is a given real $n \times m$ matrix of rank $k \leqslant n \leqslant m$, and $f = (f_i)$ is a given real n-vector. It is required to calculate the solution m-vector a^* for this system.

It is known that system (1) has a solution if and only if $\text{rank}(C \mid f) = \text{rank}(C)$. If $\text{rank}(C \mid f) > \text{rank}(C)$, the system is inconsistent and it has no solution. Also since the number of equations is less than the number of unknowns, then if system (1) has a solution it has an infinite number of solutions. We shall assume throughout the present work that system (1) is consistent and thus an infinite number of solutions exist.

However, in the present problem, from these infinite solutions, we seek the solution vector a^* whose L_∞ or Chebyshev norm

$$\| a \|_\infty = \max[| a_1 |, ..., | a_m |], \tag{2}$$

is as small as possible. Such a problem arises in many engineering applications, particularly in control theory applications. These are known as the

ISSN 0021-9045

minimum norm control systems problems. See the references cited in [4, 5, 8].

Using some basic concepts from functional analysis, Cadzow [4] described an efficient algorithm for solving this problem. In his algorithm, a column exchange method is used and this necessitates that matrix C in (1) should satisfy the Haar condition. Using the same concepts from functional analysis, Cadzow [5] described another algorithm which handles the non-Haar cases, but necessitates that matrix C be of full rank.

In the present work, this problem is reduced to a linear programming problem. A modified simplex algorithm is then described. In this algorithm no conditions are imposed on matrix C, such as the Haar condition or the full rank condition. The present method is based on a method applied recently to a related problem [1]. In [5], Cadzow outlined a linear programming scheme for the present problem, but it is completely different from ours.

The present algorithm is in two parts. In part 1, an initial basic feasible solution for the linear programming problem is obtained without needing any artificial variables. The objective function z is then calculated. If $z < 0$, it is made positive. This requires the least effort. The marginal costs are then calculated. Part 2 consists of a slightly modified simplex method which suits our problem. The algorithm needs the minimum computer storage.

If $Ca = f$ is inconsistent, this will be detected in part 2 of the algorithm. This is indicated by the existence of an unbounded solution to the linear programming problem. In this case the calculation is terminated. If rank $(C \mid f) = \text{rank}(C) = k < n$, that is one or more equations in (1) is redundant, such equations will be found out at the end of the program.

The elements of the solution vector a^* to the given problem (1), are calculated from the objective function and the marginal costs in the final tableau of the linear programming problem.

The present algorithm is a simple one, and thus can be easily implemented. Numerical results show that the algorithm is a fast one.

2. The Linear Programming Presentation

Let in (2), $\| a \|_\infty = h \geq 0$. Then the present problem may be reduced to a linear programming problem as follows.

$$\text{minimize } h \tag{3a}$$

subject to

$$-h \leq a_i \leq h, \qquad i = 1,..., m \tag{3b}$$

and

$$Ca = f.$$

The last set of constraints may be replaced by

$$f \leqslant Ca \leqslant f. \tag{3c}$$

After rearranging the constraints, this problem is conventionally formulated as follows.

$$\min Z = e_{m+1}^{\mathrm{T}} \begin{pmatrix} a \\ h \end{pmatrix} \tag{4a}$$

subject to

$$\begin{aligned} Ca &\geqslant f \\ a + he &\geqslant 0 \\ -Ca &\geqslant -f \\ -a + he &\geqslant 0 \end{aligned} \tag{4b}$$

and

$$h \geqslant 0, \tag{4c}$$

where e_{m+1} is an $(m+1)$-vector each element of which is zero except the $(m+1)$th element which is 1. Also e is an m-vector each element of which is 1. The T here and later refers to the transpose.

It is more efficient to use the dual of problem (4). For a related case see [7, p. 174]. The dual formulation is

$$\max z = [f^{\mathrm{T}}, 0^{\mathrm{T}}, -f^{\mathrm{T}}, 0^{\mathrm{T}}]\, b = g^{\mathrm{T}} b \tag{5a}$$

subject to

$$\begin{pmatrix} C^{\mathrm{T}} & I & -C^{\mathrm{T}} & -I \\ 0^{\mathrm{T}} & e^{\mathrm{T}} & 0^{\mathrm{T}} & e^{\mathrm{T}} \end{pmatrix} b \; \begin{matrix} = \\ \leqslant \end{matrix} \; \begin{pmatrix} 0 \\ 1 \end{pmatrix} \tag{5b}$$

and

$$b_i \geqslant 0, \qquad i = 1, 2, \ldots, 2(n+m). \tag{5c}$$

On the r.h.s. of (5b), the 0 is an m-zero column and on the l.h.s. each I is an m-unit matrix and each 0^{T} is an n-zero row vector.

We may now add one slack variable, b_s say, in order to be able to replace the inequality in (5b) by an equality. Thus (5b) reduces to

$$\begin{pmatrix} C^{\mathrm{T}} & I & -C^{\mathrm{T}} & -I & 0 \\ 0^{\mathrm{T}} & e^{\mathrm{T}} & 0^{\mathrm{T}} & e^{\mathrm{T}} & 1 \end{pmatrix} \begin{pmatrix} b \\ b_s \end{pmatrix} = e_{m+1}. \tag{5b$'$}$$

However, we show in Lemma 1 below that we should take $b_s = 0$ and the inequality in (5b) becomes an equality. Hence (5b) is replaced instead by

$$\begin{pmatrix} C^T & I & -C^T & -I \\ 0^T & e^T & 0^T & e^T \end{pmatrix} b = e_{m+1}.$$

For convenience, we write the above equation in the form

$$Db = e_{m+1}. \tag{5d}$$

As usual, a simplex tableau for $(m+1)$ constraints in $2(n+m)$ variables is to be constructed for problem (5). We call this the large tableau, in order to identify it from the condensed tableau which will be described in Section 3. Later we show that we need to calculate only n columns of the condensed tableaux.

Let the basis matrix, at any stage of the computation be denoted by B. B is of order $(m+1)$. For any column j in the simplex tableau, the vector y_j is given by

$$y_j = B^{-1}D_j, \qquad j = 1, 2,..., 2(n+m); \tag{6}$$

where D_j is the jth column of matrix D of (5d).

Let the elements of the $2(n+m)$ vector $[f^T, 0^T, -f^T, 0^T]^T$ of (5a), associated with the basic variables be the $(m+1)$ vector g_B. Then for the marginal costs, denoted by $\{z_j - g_j\}$, we have

$$z_j = g_B{}^T y_j, \qquad j = 1,..., 2(n+m). \tag{7}$$

The basic solution, denoted by b_B is given by

$$b_B = B^{-1}e_{m+1}, \tag{8}$$

and the objective function z is given by

$$z = g_B{}^T b_B. \tag{9}$$

DEFINITION. Consider the matrix of constraints D in (5d). Because of the kind of symmetry this matrix has, we define any column i, $1 \leqslant i \leqslant (n+m)$, and the column $j = (i + (n+m))$ in this matrix as two corresponding columns.

By using a similar argument to that given by Osborne and Watson [7, Lemma 4.3] for a related problem, we can show that any two corresponding columns should not appear together in any basis.

Let z^* be the optimum objective function for programming problem (5) and let B^* be the basis matrix associated with the optimum solution. Then the optimum Chebyshev deviation $\|a^*\|_\infty = z^* = Z^*$. Z^* is the optimum solution of (4).

Again, if in the dual problem (5), a column is in the basis for the optimal solution, its corresponding inequality in the primal (4b) is an equality [6, p. 239]. Hence from (4b), $\binom{a^*}{z^*}$ is the solution of the system

$$(B^*)^{\mathrm{T}} \binom{a^*}{z^*} = g_{B^*}, \tag{10a}$$

where the elements of g_{B^*} are those associated with the basic variables for the optimum solution of (5).

Consider an example of obtaining the minimum L_∞ solution of system (1) for two equations of rank 2 in five unknowns. In this case (10a) consists of six equations in six unknowns, the five elements of a and z. These equations would be the first two and a suitable four of the remaining five of the following system.

$$\begin{bmatrix} r_1c_{11} & r_1c_{12} & r_1c_{13} & r_1c_{14} & r_1c_{15} & 0 \\ r_2c_{21} & r_2c_{22} & r_2c_{23} & r_2c_{24} & r_2c_{25} & 0 \\ r_3 & 0 & 0 & 0 & 0 & 1 \\ 0 & r_4 & 0 & 0 & 0 & 1 \\ 0 & 0 & r_5 & 0 & 0 & 1 \\ 0 & 0 & 0 & r_6 & 0 & 1 \\ 0 & 0 & 0 & 0 & r_7 & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ z \end{bmatrix} \begin{matrix} = r_1f_1 \\ = r_2f_2 \\ = 0 \\ = 0 \\ = 0 \\ = 0 \\ = 0 \end{matrix} \tag{10b}$$

where $r_i = +1$ or -1. If column i, $i = 1,..., (n + m)$ of the matrix of constraints in (5d), is in the basis, $r_i = +1$, and $r_i = -1$ if instead its corresponding column is in the basis.

In this example, the given two equations are in system (10a) and thus system (1) is satisfied. Also since four out of the five last equations in (10b) are in (10a), four out of the five elements of a^*, each equals $+z^*$ or $-z^*$.

It is clear from this example that in general, $(m + 1 - n)$ elements of a^*, each equals $+z^*$ or $-z^*$ and therefore the remaining $(n - 1)$ elements of a^* in absolute value, each $\leq z^*$. See also Lemmas 6 and 7 below.

For further use, we write (10a) in the form

$$(a^{*\mathrm{T}} z^*) = g_{B^*}^{\mathrm{T}}(B^*)^{-1}. \tag{10c}$$

LEMMA 1. *Assume that we have obtained an optimal basic feasible solution to linear programming problem* (5). *Then if b_s is in the basis*, $z^* = 0$.

Proof. Let B^* be the basis matrix for the optimum basic solution. Then from (8) and (9), z^* is given by

$$z^* = g_{B^*}^{\mathrm{T}}(B^*)^{-1} e_{m+1}.$$

Also since b_s is assumed in the basis, the marginal cost for $b_s = 0$. Hence from (6), (7), and (5b′),

$$(z_s - g_s) = z_s - g_{B^*}^{\mathrm{T}}(B^*)^{-1} e_{m+1} = 0,$$

which implies that $z^* = 0$. This completes the proof of the lemma.

Remark 1. If $z^* = 0$ then $|a^*|_r = 0$ and hence in (1) $a^* = 0$, implying that $f = 0$, which is a contradiction. This justifies taking $b_s \neq 0$ in (5b′).

In the following, Lemmas 2, 3, and 4 are analogous to Lemmas 1, 3, and 4 in [1], respectively. The proofs of these lemmas follow those of the corresponding lemmas in [1].

LEMMA 2. *Let b_B be any basic solution, feasible or not, and z be the objective function for programming problem* (5). *Let also column j be the corresponding column to column i, $1 \leqslant i \leqslant (n + m)$, in the matrix of constraints in* (5d). *Then*

$$y_i + y_j = 0, \qquad i = 1, \ldots, n, \tag{11a}$$

and

$$y_i + y_j = 2b_B, \qquad i = (n + 1), \ldots, (n + m).$$

Also for the marginal costs

$$(z_i - f_i) + (z_j - f_j) = 0, \qquad i = 1, \ldots, n, \tag{12a}$$

and

$$(z_i - f_i) + (z_j - f_j) = 2z, \qquad i = (n + 1), \ldots, (n + m). \tag{12b}$$

LEMMA 3. *Let us assume that we have obtained an initial basic solution to problem* (5). *Then*: (1) *For each basic solution, feasible or not, there correspond two bases, $B_{(1)}$ and $B_{(2)}$ say, each determines the same basic solution. Every column in one of the bases has its corresponding column in the other basis, arranged in the same order.* (2) *However, the two values of z are equal in magnitude but opposite in sign.*

LEMMA 4. *Consider the two bases $B_{(1)}$ and $B_{(2)}$ defined in Lemma* 3, *and let us use* (7)–(8) *to construct two simplex tableaux $T_{(1)}$ and $T_{(2)}$ which correspond, respectively, to $B_{(1)}$ and $B_{(2)}$. Let also i be the corresponding column to column j, where $1 \leqslant j \leqslant 2(n + m)$. Then we have y_i in $T_{(1)} = y_j$ in $T_{(2)}$.*

Consider also the following lemmas.

LEMMA 5. *At any stage of the computation, the $(m + 1)$th column of B^{-1} equals the basic solution b_B.*

Assume that we have obtained an optimum basic feasible solution to linear programming problem (5). Then it is concluded earlier that $(m + 1 - n)$ elements of a^* each equals $+z^*$ or $-z^*$, and the remaining $(n - 1)$ elements of a^*, in absolute value, each $\leqslant z^*$.

LEMMA 6. *If column j, $(n + 1) \leqslant j \leqslant (n + m)$, of the matrix of constraints of* (5d) *is in the final basis, then $a_{j-n}^* = -z^*$. But if instead, the corresponding column of j is in the final basis, $a_{j-n}^* = z^*$.*

Proof. The proof of the lemma may be established by considering the simple structure of equation j, $j = (n + 1), \ldots, (n + m)$, of (10b).

LEMMA 7. *The $(n - 1)$ elements of a^* whose absolute value, each $\leqslant z^*$, each is calculated from the marginal cost of a nonbasic column in the final condensed simplex tableau. These nonbasic columns are not corresponding columns of any column in the basis. Let column j, $(n + 1) \leqslant j \leqslant (n + m)$, of the matrix of constraints be such nonbasic column. Then*

$$a_{j-n}^* = (z_j - g_j) - z^*, \qquad (n + 1) \leqslant j \leqslant (n + m), \tag{13a}$$

or

$$a_{j-2n-m}^* = z^* - (z_j - g_j), \qquad (2n + m + 1) \leqslant j \leqslant 2(n + m). \tag{13b}$$

Proof. Again, the nonbasic columns which are not corresponding columns to any column in the basis are $(n - 1)$ columns j, where $(n + 1) \leqslant j \leqslant (n + m)$ and their corresponding columns.

Consider the case $(n + 1) \leqslant j \leqslant (n + m)$. From (5d), (6), and (7),

$$z_j - g_j = z_j = g_{B^*}^{\mathrm{T}} (B^*)^{-1} \begin{pmatrix} u_{j-n} \\ 1 \end{pmatrix}, \tag{14}$$

where u_j is the jth column in an m-unit matrix. Substituting in (14), $\begin{bmatrix} u_{j-n} \\ 1 \end{bmatrix} = \begin{bmatrix} u_{j-n} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, we get

$$z_j - g_j = g_{B^*}^{\mathrm{T}} [(B^*)_{j-n}^{-1} + (B^*)_{m+1}^{-1}],$$

where $(B^*)_{j-1}^{-1}$ and $(B^*)_{m+1}^{-1}$ are, respectively, the $(j - n)$ th and the $(m + 1)$ th column of $(B^*)^{-1}$. Hence by using (10c), Lemma 5, and (9), we get (13a). In the same way (13b) is proved.

LEMMA 8. *If the set of equations $Ca = f$ in* (1) *is inconsistent, i.e.,* rank $(C \mid f) >$ rank(C), *then the solution of linear programming problem* (5), *would be unbounded.*

Proof. It is sufficient to illustrate this case by an example. Consider the solved example of Section 4 below. Let in the set of Eqs. (15), $f_3 = -20$

instead. In this case $\text{rank}(C \mid f) = 3$ while $\text{rank}(C) = 2$ and the system would be inconsistent. Hence in tableau 2, $(z_{11} - g_{11})$ would be the most negative marginal cost and thus D_{11} replaces D_9 in the basis. This gives tableau 3. In tableau 3, we find that D_1 has the most negative marginal cost. yet every element of y_1 is $\leqslant 0$.

3. The Description of the New Method

From Lemma 2, if column y_i, $1 \leqslant i \leqslant (n + m)$, in the simplex tableau is known, its corresponding column y_j, $j = i + (n + m)$ is readily known and vise versa. The same is true for the marginal costs. Hence we start by constructing a simplex tableau for problem (5), for $(m + 1)$ constraints in only $(n + m)$ variables. Let these be the first $(n + m)$ elements of the $2(n + m)$-vector b. We call these the condensed tableaux.

The algorithm is in two parts. In part 1, we obtain an initial basic feasible solution. We also calculate the initial objective function z ($\geqslant 0$) and the initial marginal costs $\{z_i - g_i\}$.

We take advantage of the existence of the m-unit submatrices in matrix D in (5d) in obtaining an initial basic feasible solution without the need of artificial variables.

The m columns $(n + 1),..., (n + m)$, in matrix D, each is a column in an m-unit matrix augmented by a 1 as the $(m + 1)$ th element. We chose these m columns, or their corresponding columns, to form the first m columns in the initial basis matrix B. This is simply done by performing m Gauss–Jordan eliminations. For each of these columns, a Gauss–Jordan elimination consists of one step only. This is the step needed to eliminate the 1 in the $(m + 1)$th position of this column. The choice between column $(n + i)$ or its corresponding column is given in the next paragraph.

Consider any one of the first n columns in matrix D. Denote this column by X. Consider element i, $i = 1,..., m$, in succession of column X. If in X, element $i \leqslant 0$, we chose column $(n + i)$ in matrix D to form the ith column of B. If element i in $X > 0$, we chose instead, the corresponding column to column $(n + i)$ to form the ith column of B. When all these m columns enter the basis, the first m elements of X, each would keep its value, with a negative sign and the $(m + 1)$th element of X would be >0. In fact, this $(m + 1)$th element would equal the sum of the absolute values of the first m elements in X.

Column X will now be chosen to be the $(m + 1)$ th column of B. The process described in the previous paragraph guarantees that when column X enters the basis as the $(m + 1)$ th column of B, the initial basic solution would be feasible. That is each element of $b_B \geqslant 0$.

The objective function z is then calculated from (9). If $z < 0$, we make use of Lemmas 3 and 4 and replace the basis matrix by its corresponding one.

We also replace the columns of the condensed simplex tableau by their corresponding columns. In effect, we keep the simplex tableau unchanged, except for the f values and z. Such parameters have their signs reversed. See also [1]. The marginal costs $\{z_j - g_j\}$ are then calculated from (7). This ends part 1 of the algorithm, as we now have an initial basic feasible solution and $z \geqslant 0$.

Part 2 of the algorithm, is the ordinary simplex algorithm. The only difference is in the choice of the nonbasic column which enters the basis. The column to enter the basis is that which has the most negative marginal cost among the nonbasic columns in the current tableau and their corresponding columns. Relation (12) is used for calculating the marginal costs of the corresponding columns.

Finally, the elements of the solution vector a^* to the given problem (1) are calculated from Lemmas 6 and 7.

4. Numerical Results

The above described steps are now explained by a solved example.

Example 1. Obtain a minimum L_∞ solution of the following underdetermined system of linear equations.

$$\begin{aligned} 7a_1 - 4a_2 + 5a_3 + 3a_4 + a_5 &= -30, \\ -2a_1 + a_2 + 5a_3 + 4a_4 + a_5 &= 15, \\ 5a_1 - 3a_2 + 10a_2 + 7a_4 + 2a_5 &= -15. \end{aligned} \tag{15}$$

In (15) C is an 3×5 matrix of rank 2 and the system is consistent. The third equation is the sum of the other two.

Shown are the initial data for linear programming problem (5) and the condensed tableaux for the algorithm described in Section 3. Again, D_j, $1 \leqslant j \leqslant 2(n+m)$, is the jth column of the matrix of constraints in (5d). The pivot elements are bracketed.

Tableau 1 is obtained by having columns 12, 5, 14, 15, 16, and 1 of the matrix of constraints to form, respectively, columns 1,..., 6 of the initial basis matrix B.

In tableau 1, the initial basic solution is feasible but $z < 0$. We therefore make use of Lemmas 3 and 4 and write down tableau 1*. Tableau 1* is itself tableau 1 except for the f values and z. Such parameters have their signs reversed. The columns of tableau 1* are the corresponding columns of tableau 1. In tableau 1*, D_2 which has the most negative marginal cost replaces its corresponding column D_{10}. From (11a), $y_2 = -y_{10}$ and from

Initial Data

g	−30	15	−15	0	0	0	0	0
b_B	D_1	D_2	D_3	D_4	D_5	D_6	D_7	D_8
0	7	−2	5	1	0	0	0	0
0	−4	1	−3	0	1	0	0	0
0	5	5	10	0	0	1	0	0
0	3	4	7	0	0	0	1	0
0	1	1	2	0	0	0	0	1
1	0	0	0	1	1	1	1	1

Tableau 1 (part 1)

g	−30	15	−15	0	0	0	0	0
b_B	D_1	D_2	D_3	D_{12}	D_5	D_{14}	D_{15}	D_{16}
0.35	0	4.45	4.45	1	0	0	0	0
0.20	0	2.40	2.40	0	1	0	0	0
0.25	0	−3.25	−3.25	0	0	1	0	0
0.15	0	−2.95	−2.95	0	0	0	1	0
0.05	0	−0.65	−0.65	0	0	0	0	1
0.05	1	0.35	1.35	0	0	0	0	0

$z = -1.5$

Tableau 1*

g	30	−15	15	0	0	0	0	0
b_B	D_9	D_{10}	D_{11}	D_4	D_{13}	D_6	D_7	D_8
0.35	0	4.45	4.45	1	0	0	0	0
0.20	0	2.40	2.40	0	1	0	0	0
0.25	0	−3.25	−3.25	0	0	1	0	0
0.15	0	−2.95	−2.95	0	0	0	1	0
0.05	0	−0.65	−0.65	0	0	0	0	1
0.05	1	0.35	1.35	0	0	0	0	0
z = 1.5	0	25.5	25.5	0	0	0	0	0

Tableau 1**

g	30	15	15	0	0	0	0	0
b_B	D_9	D_2	D_{11}	D_4	D_{13}	D_6	D_7	D_8
0.35	0	−4.45	4.45	1	0	0	0	0
0.20	0	−2.40	2.40	0	1	0	0	0
0.25	0	3.25	3.25	0	0	1	0	0
0.15	0	(2.95)	−2.95	0	0	0	1	0
0.05	0	0.65	−0.65	0	0	0	0	1
0.05	1	0.35	1.35	0	0	0	0	0
z = 1.5	0	25.5	25.5	0	0	0	0	0

Tableau 2 (part 2)

g	30	15	15	0	0	0	0	0
b_B	D_9	D_2	D_{11}	D_1	D_{13}	D_6	D_7	D_8
0.576	0	0	0	1	0	0	1.509	0
0.322	0	0	0	0	1	0	0.814	0
0.085	0	0	0	0	0	1	−1.102	0
0.051	0	1	−1	0	0	0	0.339	0
0.017	0	0	0	0	0	0	−0.220	1
0.068	1	0	1	0	0	0	0.119	0
z = 2.797	0	0	0	0	0	0	8.644	0

Tableau 2*

g	30	15	15	0	0	0	0	0
b_B	D_9	D_2	D_{11}	D_1	D_{13}	D_6	D_{15}	D_8
0.576	0	0	0	1	0	0	−0.356	0
0.322	0	0	0	0	1	0	−0.170	0
0.085	0	0	0	0	0	1	1.271	0
0.051	0	1	−1	0	0	0	−0.237	0
0.017	0	0	0	0	0	0	(0.254)	1
0.068	1	0	1	0	0	0	0.017	0
z = 2.797	0	0	0	0	0	0	−3.051	0

Tableau 3

g	30	15	15	0	0	0	0	0
b_B	D_9	D_2	D_{11}	D_1	D_{13}	D_6	D_{15}	D_8
0.600	0	0	0	1	0	0	0	1.40
0.333	0	0	0	0	1	0	0	0.667
0.000	0	0	0	0	0	1	0	−5.000
0.067	0	1	−1	0	0	0	0	0.933
0.067	0	0	0	0	0	0	1	3.933
0.067	1	0	1	0	0	0	0	0.067
z 3	0	0	0	0	0	0	0	12

Tableau 4

g	30	15	15	0	0	0	0	0
b_B	D_9	D_2	D_{11}	D_1	D_{13}	D_6	D_{15}	D_{16}
0.600	0	0	0	1	0	0.04	0	0
0.333	0	0	0	0	1	0.00	0	0
0.000	0	0	0	0	0	0.20	0	1
0.067	0	1	−1	0	0	0.16	0	0
0.067	0	0	0	0	0	0.76	1	0
0.067	1	0	1	0	0	0.04	0	0
z 3	0	0	0	0	0	1.2	0	0

(12a), $(z_2 - g_2) = -(z_{10} - g_{10})$. This gives tableau 1**. In tableau 1**, D_2 replaces D_7 in the basis. This gives tableau 2.

In tableau 2, D_{15} which has the most negative marginal cost replaces its corresponding column D_7. From (11b), $y_{15} = 2b_B - y_7$ and from (12b), $(z_{15} - g_{15}) = 2z - (z_7 - g_7)$. This gives tableau 2*.

In tableau 2*, D_{15} replaces D_8 in the basis. This gives tableau 3. In tableau 3, D_{16} replaces its corresponding column D_8. This gives tableau 3* (not shown). In tableau 3*, D_{16} replaces D_6 in the basis. This gives tableau 4. Tableau 4 gives the optimum solution which is $z^* = 3$.

From Lemmas 6 and 7, we get the minimum L_∞ solution of system (15), which is $a^* = (-3, 3, -1.8, 3, 3)^T$.

We notice in tableaux 2 to 4, that $(z_3 - g_3) = (z_{11} - g_{11}) = 0$. This indicates that the third equation in (15) is redundant and thus $\text{rank}(C \vdots f) = \text{rank}(C) = 2$.

We also notice that the optimal basic feasible solution is degenerate as it contains one zero element. This indicates that the optimum solution is probably not unique.

By giving a careful look to the above simplex tableaux, we notice that six out of eight, i.e., $(m - 1)$ out of $(n + m)$ columns in the condensed tableaux are actually the six columns of a six-unit matrix. Such columns need only be accounted for and need not be written down. This can easily be done and the condensed simplex tableau may be condensed more. We denote such tableaux as the reduced tableaux. In the reduced tableaux, we calculate only $(n + 1)$ columns and their marginal cost, as well as b_B and z. These $(n + 1)$ columns are the $(n + 1)$ nonbasic columns in the condensed tableaux.

A computer program for the present algorithm which calculates the reduced tableaux only is coded in Fortran IV [2]. It is used to solve the above example in single precision calculation on the IBM 360/67 computer. The execution time, that is the CPU time is about 0.007 seconds.

The numerical example solved by Cadzow [4, p. 616], is solved by our program and the execution time is 0.016 seconds. This is $\frac{1}{2}$ the time given by Cadzow who used his method on a faster computer. Eventually, the last element of the solution vector of this example should read -0.5371.

Several test problems have been solved by the present method and the numerical results show that the present algorithm is a fast one.

5. Comments and Conclusion

The present method may be identified by the following features.

All the calculations are made in the reduced simplex tableaux. An initial basic feasible solution for the linear programming problem as well as the initial objective function z (≥ 0) are obtained with a minimum effort. We do not need to calculate the marginal costs until the end of part 1 of the algorithm. The inverse of the basis matrix, B^{-1}, is never calculated. The elements of the solution vector a^* of the given system (1) are calculated from z^* and the marginal costs of the final reduced tableau. The rank of C is known at the end of the solution.

Using our notation, we may compare the number of arithmetic operations of the method of Cadzow [5] to that of ours. The number of multiplications per iteration in [5] is $>(3nm + m)$. However, in the present method, the number of multiplications required per iteration, i.e., required to change a simplex tableau is $n(m + 3)$. This is about $\frac{1}{3}$ that of Cadzow. Numerical evidence of the test cases show that the present method converges in n to $2n$ iterations.

Finally, we mention that the present method would be most efficient if certain intermediate iterations in the simplex method could be skipped. For this point see Barrodale and Roberts [3].

References

1. N. N. Abdelmalek, Chebyshev solution of overdetermined systems of linear equations, *BIT* **15** (1975), 117–129.
2. N. N. Abdelmalek, "A FORTRAN Program for the Minimum L_∞ Solution of Underdetermined Systems of Linear Equations," NRC Technical Report, 1977.
3. I. Barrodale and F. D. K. Roberts, An improved algorithm for the discrete l_1 linear approximation, *SIAM J. Numer. Anal.* **10** (1973), 839–848.
4. J. A. Cadzow, A finite algorithm for the minimum l_∞ solution to a system of consistent linear equations, *SIAM J. Numer. Anal.* **10** (1973), 607–617.
5. J. A. Cadzow, An efficient algorithmic procedure for obtaining a minimum l_∞-norm solution to a system of consistent linear equations, *SIAM J. Numer. Anal.* **11** (1974), 1151–1165.
6. G. Hadley, "Linear Programming," Addison-Wesley, Reading, Mass., 1962.
7. M. R. Osborne and G. A. Watson, On the best linear Chebyshev approximation, *Comput. J.* **10** (1967), 172–177.
8. P. E. Sarachik, Functional analysis in automatic control, *IEEE Int. Conv. Rec.* **6** (1965), 96–107.