JOURNAL OF APPROXIMATION THEORY 11, 38–53 (1974)

On the Discrete Linear L_1 Approximation and L_1 Solutions of Overdetermined Linear Equations

NABIH N. ABDELMALEK

National Research Council, Ottawa, Ontario, Canada

Communicated by E. W. Cheney

Usow's algorithm for solving the discrete linear L_1 approximation problem is generalized so that it can also solve an overdetermined system of linear equations in the L_1 norm. It is then shown that this algorithm is completely equivalent to a dual simplex algorithm applied to a linear programming problem in nonnegative bounded variables. However, one iteration in the former is equivalent to one or more iterations in the latter.

A dual simplex algorithm is described which seems to be the most efficient and capable method for solving these two problems. Its efficiency is due to the absence of artificial variables and to its simplicity. Its capability is due to the fact that the Haar condition associated with Usow's method is completely relaxed. Numerical results are given.

1. INTRODUCTION

Consider the following two problems, assuming all functions are real valued.

(a) Let $f(x)$ be a given function defined on a finite subset $X = \{x_1, x_2, ..., x_n\}$ of an interval I on the real line. Let also linearly independent continuous functions $\phi_1(x), \phi_2(x), ..., \phi_m(x)$, where $m < n$, be defined on I. We consider the "polynomial"

$$L(a_1, ..., a_m, x) = a_1\phi_1(x) + \cdots + a_m\phi_m(x), \tag{1}$$

in short, $L(A, x)$, where A denotes the parameter vector $(a_1, ..., a_m)$ in m-space E_m. The L_1 approximation problem for $f(x)$ on X is to determine A^* which minimizes the function

$$R(A) = \sum_{i=1}^{n} | r(x_i)|, \tag{2}$$

where the residuals $r(x_i)$ are defined by the equalities

$$r(x_i) = L(A, x_i) - f(x_i), \qquad i = 1, ..., n. \tag{3}$$

(b) Consider the overdetermined system of linear equations

$$
\begin{aligned}
c_{11}a_1 + \cdots + c_{1m}a_m &= f_1 ,\\
\cdot \qquad\qquad & \quad \cdot \\
c_{n1}a_1 + \cdots + c_{nm}a_m &= f_n ,
\end{aligned}
\tag{4}
$$

where (c_{ij}) is an $n \times m$ constant matrix of rank $m < n$, and (f_i) and (a_i) are n and m vectors in E_n and E_m, respectively. The L_1 solution to (4) is an A^* which minimizes the function

$$R(A) = \sum_{i=1}^{n} | r_i |, \tag{5}$$

where

$$r_i = c_{i1}a_1 + \cdots + c_{im}a_m - f_i , \qquad i = 1,..., n. \tag{6}$$

The symbols used for problem (b) are chosen in a way to match those of problem (a). In (4), $f = \{f_i\}$ and $A = \{a_i\}$ correspond to $\{f(x_i)\}$ and $\{a_i\}$ of problem (a), respectively. Also matrix (c_{ij}) corresponds to $(\phi_j(x_i))$. Consequently, $\{r_i\}$ of (6) corresponds to $\{r(x_i)\}$ of (3).

It is clear that problem (a) is equivalent to problem (b). This is shown by writing down the n equations

$$
\begin{aligned}
a_1\phi_1(x_1) + \cdots + a_m\phi_m(x_1) &= f(x_1),\\
\cdot \qquad\qquad & \quad \cdot \\
a_1\phi_1(x_n) + \cdots + a_m\phi_m(x_n) &= f(x_n),
\end{aligned}
\tag{7}
$$

and examining (7) and (2) in view of (4) and (5).

For the important case when the approximating set of functions $\{\phi_j(x)\}$ is a Tchebycheff set, Usow [10] treated problem (a) by solving the geometrical problem equivalent to minimizing (2).

Wagner [12] reduced problem (b) to a linear programming problem in both the primal and the dual forms.

In Section 2, Usow's algorithm is generalized to handle problem (b) as well. In Section 3, the equivalent linear programming problems are presented. It is then shown in Sections 4 and 5 that Usow's algorithm is completely equivalent to a dual simplex algorithm applied to a linear programming problem in nonnegative bounded variables. Except that one iteration in the former algorithm is equivalent to one or more iterations in the latter. Also in Section 5, a suitable dual simplex algorithm for solving the above two problems is described and a known theorem for the discrete linear L_1 approximation is restated. In Section 6 numerical results are given. Finally it is concluded in Section 7, that compared to other existing methods, the presented algorithm seems to be the most efficient and capable one.

We mention here that the dual properties of the discrete linear L_1 and L_∞ approximations are emphasized once more. While Usow's algorithm is the analog of Cheney and Goldstein's [4], for the L_∞ approximation, the present work is the analog of Osborne and Watson's [6]. See also Valentine and Van Dine [11] and Stiefel [9].

2. Usow's Algorithm

Usow's approach is based on the following theorem, which is a characterization of the solution set [7, p. 114].

THEOREM 1. *Let the set of functions $\{\phi_j(x)\}$ be a Tchebycheff set, i.e., the matrix $(\phi_j(x_i))$ satisfies the Haar condition. Then the best L_1 approximation, $L(A^*, x)$ to $f(x)$ on X is a closed convex set which is the convex hull of best L_1 approximations for which $L(A^*, x)$ interpolates $f(x)$ in at least m of the given n points X.*

The equivalent geometrical problem is the following: Let the set K be

$$K = \{(A, d) \mid (A, d) \in E_{m+1}, R(A) \leqslant d\}.$$

Then K is a convex polytope, the vertices of which occur only when the function $L(A, x) - f(x)$ is zero at m (or more) points of X. A vertex $(A_i, d_i = R(A_i))$ on K is said to have abscissa A_i and ordinate d_i.

The algorithm is to descend on K from vertex to vertex along connecting edges of the polytope in such a way that certain intermediate vertices are by-passed. This descent continues until the lowest vertex (A^*, d^*) is reached. It is sufficient to describe one cycle in the algorithm.

Assume that we are at the vertex (A_k, d_k) on K. Let the polynomial $L(A_k, x)$ interpolate the m points of X denoted by $U_k = \{u_k^1, \ldots, u_k^m\}$ and call this the reference point set. In Lagrangian form,

$$L(F_k, x) = \sum_{i=1}^{m} f(u_k^i)\, \pi_i(x), \tag{8}$$

where F_k denotes the parameter vector $(f(u_k^1), \ldots, f(u_k^m))$. In terms of $\{\phi_j(x)\}$, $j = 1, \ldots, m$,

$$\pi_i(x) = \sum_{j=1}^{m} b_j^i \phi_j(x), \qquad i = 1, \ldots, m. \tag{9}$$

The m coefficients b_j^i are calculated as follows.

Let the functions $\phi_j(u_k^i)$, $i, j = 1,..., m$, form the matrix $(\phi_j(u_k^i))$. Let also for any point $x \in X$, $\Pi(x)$ and $\Phi(x)$ be the two m vectors whose elements are $\{\pi_1(x),..., \pi_m(x)\}$ and $\{\phi_1(x),..., \phi_m(x)\}$, respectively. Hence it is easy to verify that

$$\Pi(x) = [(\phi_j(u_k^i))^T]^{-1} \Phi(x). \tag{10}$$

For $i = 1,..., m$, the m coefficients b_j^i, $j = 1,..., m$, are the elements of the ith row of the matrix $[(\phi_j(u_k^i))^T]^{-1}$ in (10). It is also easy to verify that (10) satisfies $\pi_i(u_k^j) = \delta_{ij}$.

Let e_i be the ith column in an m unit matrix. Then if for some δ, $R(F_k - \delta e_i) < R(F_k)$, there is a T_j such that $T_j\delta > 0$ and $R(F_k - T_j e_i) < R(F_k)$. Also

$$R(F_k - T_j e_i) = \min_t\{R(F_k - te_i)\} \tag{11}$$

and $((F_k - T_j e_i), R(F_k - T_j e_i))$ is a vertex [10, pp. 238–239].

In other words a point $u_k^i \in U_k$ may be replaced by a point $u_{k+1}^i \in X - U_k$ such that the polynomial $L(A_{ki}, x)$ interpolating $U_{ki} = \{u_k^1,..., u_{k+1}^i,..., u_k^m\}$ has its norm $R(A_{ki}) < R(A_k)$. Also as indicated by (11), $R(A_{ki})$ is the minimum of all norms obtained if u_k^i were replaced by the different points of the set $\{X - U_k\}$.

We mention that in going from $(A_k, R(A_k))$ to $(A_{ki}, R(A_{ki}))$, one or more vertices on K might have been by-passed. The nearest vertex to $(F_k, R(F_k))$ and below it on the edge parallel to the ith parameter space coordinate axis, say the vertex $((F_k - t_r e_i), R(F_k - t_r e_i))$, is obtained from

$$| t_r | = \min_s\{| L(F_k, x_s) - f(x_s)|/| \pi_i(x_s)|\}, \qquad x_s \in X - U_k. \tag{12}$$

This point x_r is characterized by

$$\operatorname{sgn}[L(F_k, x_l) - f(x_l)] = \operatorname{sgn}[L(F_{ki}, x_l) - f(x_l)], \qquad x_l \in X - U_k - x_r. \tag{13}$$

Again, if there is no δ such that $R(F_k - \delta e_i) < R(F_k)$, then the norm $R(F_k)$ could not be reduced by moving on K along the edge parallel to the ith parameter space coordinate axis. In other words u_k^i should not be replaced by another point from the set $\{X - U_k\}$.

This iteration is repeated m times, once for each point in U_k in succession. The whole cycle is then repeated a finite number of times until the solution (A^*, d^*) is reached.

However, in order to handle problem (b) also, the above algorithm should be stated in a way which does not involve the point set X, but Eqs. (4) or (7) instead. This is indeed possible.

By examining the set of Eqs. (7), we see that each equation corresponds to a particular point in the set X. The coefficients in $L(A, x)$ interpolating any m points of X, might be calculated by solving the m equations in (7) which correspond to such points. We consider Eqs. (7) if we deal with problem (a) and Eqs. (4) if we deal with problem (b). Let us demonstrate on the former. Let the m equations in (7) which correspond to the m points U_k be denoted by the reference equation set.

The ith iteration in one cycle then is to attempt to replace the ith equation in the reference equation set by an equation not in the set, such that a minimum norm is obtained. This is done for each equation in the reference set in succession. The whole cycle may be repeated for a finite number of times until the solution (A^*, d^*) is reached.

It is to be mentioned that the polytope K might have a flat bottom and, consequently, has many (corners) vertices at the bottom. In this case, the point (A^*, d^*) will be one of such vertices and the solution is not unique. Any point on the flat bottom including such vertices is a best L_1 approximation. This might happen also in the presence of the Haar condition. The Haar condition guarantees that any m equations in (7) have a unique solution.

3. The Linear Programming Problem

It is seen in Section 1 that problem (a) is equivalent to problem (b). Thus, we may demonstrate on the latter one. It is shown by Wagner [12] that this problem may be reduced to a linear programming problem. The primal form is

$$\min Z = \sum_{i=1}^{n} \epsilon_{1i} + \sum_{i=1}^{n} \epsilon_{2i}, \tag{14a}$$

subject to the constraints

$$(C\ I_n\ -I_n) \begin{pmatrix} A \\ \epsilon_1 \\ \epsilon_2 \end{pmatrix} = f, \tag{14b}$$

a_i unrestricted, $i = 1,\ldots, m$,

$$\epsilon_{1i} \geqslant 0, \qquad \epsilon_{2i} \geqslant 0, \qquad i = 1,\ldots, n. \tag{14c}$$

Here C, A and f are, respectively, matrix (c_{ij}) and column vectors (a_i) and (f_i) of (4). The column vectors $\epsilon_1 = (\epsilon_{1i})$ and $\epsilon_2 = (\epsilon_{2i})$. I_n is an n unit matrix. A program for solving (14) using the simplex algorithm is given in [2].

Yet, by going over to the dual, we have the problem

$$\max z = \sum_{i=1}^{n} f_i w_i , \tag{15a}$$

subject to the constraints

$$C^T w = 0, \tag{15b}$$

$$-1 \leqslant w_i \leqslant 1, \qquad i = 1,..., n, \tag{15c}$$

where the vector $w = (w_i)$. An algorithm using interval programming techniques for solving (15) is given in [8].

However, by defining $b_i = w_i + 1$ and denoting $b = (b_i)$, we get the formulation

$$\max z = \sum_{i=1}^{n} f_i(b_i - 1), \tag{16a}$$

subject to the constraints

$$C^T b = \begin{pmatrix} \sum_{i=1}^{n} c_{i1} \\ \vdots \\ \sum_{i=1}^{n} c_{im} \end{pmatrix}, \tag{16b}$$

$$0 \leqslant b_i \leqslant 2, \qquad i = 1,..., n, \tag{16c}$$

This is a programming problem in nonnegative bounded variables. It may be solved by the simplex algorithm, as a problem with $(m + n)$ constraints instead of the m constraints (16b). However, it is shown by Hadley [5, pp. 387–394], that if a simple set of rules is observed, the same problem may be solved without adding the extra n constraints. Let us call this the special simplex method. The solution is based on the following theorem.

THEOREM 2. *A necessary and sufficient condition for a nonzero program for system* (16) *to be optimal is that* $(n - m)$ *elements of* b, *each has the value zero* (*lower bound*) *or* 2 (*upper bound*), *and that the other* m *elements are basic variables.*

In this algorithm, we construct a simplex tableau for problem (16) as if the elements of b were unbounded from above. Let $k_i(C^T)$, $i = 1,..., n$, be the ith column of matrix C^T. Let m of such columns form the basis matrix B and let b_B be the basic solution. Let us define a basis indicator set for b as the index set $I(b) \subset \{1, 2,..., n\}$ with the property that the vectors $\{k_i(C^T) \mid i \in I(b)\}$ are a basis for E_m. Let us also have the index given by

$b_B = \{b_{B_i}\}$, $i = 1,..., m$. Let the index sets $L(b)$ and $U(b)$ be indicators for the nonbasic variables b_i which are respectively at their lower and upper bounds. That is

$$L(b) = \{i \in \{1, 2,..., n\} \mid b_i = 0, i \notin I(b)\}$$

and

$$U(b) = \{i \in \{1, 2,..., n\} \mid b_i = 2, i \notin I(b)\}.$$

Then as usual, for any $k_i(C^T)$, $i \notin I(b)$,

$$y_i = B^{-1}k_i(C^T), \tag{17}$$

and

$$z_i = f_B{}^T y_i , \tag{18}$$

where the elements of f_B are f_i , $i \in I(b)$. Hence

$$z_i = f_B{}^T B^{-1}k_i(C^T). \tag{19}$$

Again, since some of the nonbasic variables will be at their upper bound (=2), from (16b),

$$b_B = B^{-1} \begin{pmatrix} \sum_{i=1}^{n} c_{i1} \\ \vdots \\ \sum_{i=1}^{n} c_{im} \end{pmatrix} - \sum_{i \in U(b)} 2B^{-1}k_i(C^T). \tag{20}$$

By denoting the first term on the right by b_{B_0} and by (17),

$$b_B = b_{B_0} - 2 \sum_{i \in U(b)} y_i . \tag{21}$$

Also the function z in (16a) is given by

$$z = \sum_{i \in I(b)} f_i(b_i - 1) - \sum_{i \in L(b)} f_i + \sum_{i \in U(b)} f_i . \tag{22}$$

The algorithm is summarized by the following: A nonbasic column may replace one of the columns in the basis, may go from its zero bound to its upper bound or may go from its upper bound to zero. The optimal solution is characterized by this theorem.

THEOREM 3. *A basic feasible solution is maximal, if the parameters* $\{z_i - f_i\}$, $i \notin I(b)$ *satisfy the relations*

$$z_i - f_i \geqslant 0, \qquad i \in L(b), \tag{23a}$$

and

$$z_i - f_i \leqslant 0, \qquad i \in U(b). \tag{23b}$$

In the next two sections, the relation between Usow's algorithm and the dual simplex algorithm to the special simplex method for solving (16) is established.

4. Linear Programming and Usow's Algorithm

Let us consider the following lemmas.

Lemma 1. *The optimal solution of* (16) *is bounded and is given by*

$$0 \leqslant z = Z \leqslant \sum_{i=1}^{n} |f_i|.$$

Proof. The second inequality is immediate from (15a) since $|w_i| \leqslant 1$ and the first inequality follows from (14a) since $\epsilon_1 \geqslant 0$ and $\epsilon_2 \geqslant 0$. However if $Z = z = 0$, then $\epsilon_1 = \epsilon_2 = 0$ and this implies the existence of an exact solution to the original set of Eqs. (4). We exclude this case from our consideration.

Lemma 2. *Consider a basic solution (feasible or not) to the special simplex method to problem* (16). *The parameters z_i of* (18) *and the residuals r_i of* (6) *for the corresponding reference are related by*

$$z_i - f_i = r_i . \tag{24}$$

Proof. For any $k_i(C^T)$, $i \notin I(b)$, from (19),

$$z_i - f_i = f_B{}^T B^{-1} k_i(C^T) - f_i = k_i{}^T(C^T)\, B^{-T} f_B - f_i ,$$

where $B^{-T} = (B^{-1})^T$ and where the transpose of $f_B{}^T B^{-1} k_i(C^T)$ equals itself. However, as by a corresponding reference, we mean that the columns of B are the same m rows of C in the reference equation set,

$$B^{-T} f_B = (a_1 ,..., a_m)^T. \tag{25}$$

Therefore, from (6)

$$z_i - f_i = \sum_{j=1}^{m} c_{ij} a_j - f_i = r_i . \tag{26}$$

Again, for a basic column $k_i(C^T)$, that is which corresponds to the reference equation set, $z_i - f_i = 0 = r_i$.

Finally, since $\{z_i - f_i\}$ is completely independent of the requirements vector (the right-side in (16b)), (24) is valid whether the basic solution is feasible or not.

LEMMA 3. *In the special simplex method to problem* (16), *the value of z for a basic solution* (*feasible or not*) *is given by*

$$z = \sum_{i\in L(b)} (z_i - f_i) - \sum_{i\in U(b)} (z_i - f_i). \tag{27}$$

Proof. In the first term on the right side of (20), the m summation signs may be replaced by one summation sign, as

$$b_B = \sum_{i=1}^{n} B^{-1}k_i(C^T) - 2 \sum_{i\in U(b)} B^{-1}k_i(C^T). \tag{28}$$

Also the first summation in (28) may be written as

$$\sum_{i=1}^{n} B^{-1}k_i(C^T) = \sum_{i\in I(b)} B^{-1}k_i(C^T) + \sum_{i\in L(b)} B^{-1}k_i(C^T) + \sum_{i\in U(b)} B^{-1}k_i(C^T). \tag{29}$$

Yet, $\sum_{i\in I(b)} B^{-1}k_i(C^T) = \sum_{i=1}^{m} e_i = e$, where e is an m vector with unit elements. Hence, by substituting this and (29) into (28), we get

$$b_B = e + \sum_{i\in L(b)} B^{-1}k_i(C^T) - \sum_{i\in U(b)} B^{-1}k_i(C^T).$$

Then by substituting this into (22) and also since $f_B{}^T e - \sum_{i\in I(b)} f_i = 0$, we get

$$z = \sum_{i\in L(b)} f_B{}^T B^{-1}k_i(C^T) - \sum_{i\in U(b)} f_B{}^T B^{-1}k_i(C^T) - \sum_{i\in L(b)} f_i + \sum_{i\in U(b)} f_i\,. \tag{30}$$

Finally, by taking the transpose of each term in the first two summations in (30), using (25) and rearranging the terms, we get (27).

LEMMA 4. *For every basic solution in the dual simplex algorithm for the programming problem* (16), *z is given by*

$$z = \sum_{i=1}^{n} | z_i - f_i | = \sum_{i=1}^{n} | r_i |, \tag{31}$$

where $\sum_{i=1}^{n} | r_i |$ *is the norm* (5) *for the corresponding reference.*

Proof. Let us suppose that we apply the dual simplex algorithm [5, pp. 242–247] to the enlarged system of problem (16). Then the algorithm starts with a nonfeasible primal solution but feasible dual solution. That is with one or more basic variable $b_{B_i} < 0$ such that $z_i - f_i \geqslant 0$ for all i. The basis is then changed, one column at a time, keeping all $z_i - f_i \geqslant 0$, until an optimal solution is reached.

However, for the nonenlarged system, this is equivalent to starting the solution with one or more b_{B_l} violating the condition $0 \leqslant b_{B_l} \leqslant 2$, such that $z_i - f_i \geqslant 0$, $i \in L(b)$ and $z_i - f_i \leqslant 0$, $i \in U(b)$. One then moves from one basic solution to another preserving this criterion all the time until an optimal solution is reached. Therefore, by doing so, every term in the first summation in (27) is positive and every term in the second summation is negative. Also since $\sum_{i \in I(b)} (z_i - f_i) = 0$, by (24) we get (31).

LEMMA 5. *In the dual simplex algorithm to problem* (16), *the objective function z equals the function Z of* (14).

Proof. This is a direct consequence of applying the dual simplex algorithm. It is also seen that $\sum_{i=1}^{n} | r_i |$ in (31) equals Z of (14).

As a result, z is expected to decrease after each iteration in the dual simplex algorithm to problem (16). This algorithm is described in the following section.

5. THE DUAL SIMPLEX ALGORITHM

We start solving (16) by choosing any m linearly independent columns of C^T to form the basis B. The simplex tableau is then formed by calculating from (17)–(19), the vectors $\{y_i\}$ and the set $\{z_i - f_i\}$. The boundedness of the solution is guaranteed by Lemma 1 and if any degeneracy occurs, it will not cause much difficulty.

The following steps constitute none other than a dual simplex algorithm for the method described by Hadley [5, pp. 387–394]. The choice of the vectors which leave the basis is first done in accordance with Usow's method. However, improvement for faster convergence is later presented. Obviously, at the start, any nonbasic variable b_i is given the value zero.

(1) For every $z_i - f_i < 0$, $i \in L(b)$, let b_i take the value 2 and indicate that by putting a mark above the corresponding column. Calculate b_B from (21) and go to step (2).

(2) If all b_{B_l} satisfy $0 \leqslant b_{B_l} \leqslant 2$, an optimal solution is reached. If not, go to step (3).

(3) Scan b_{B_l} for $l = 1, 2, \ldots$. The first one, which is either <0 or >2, is considered. Let such variable be b_{B_i}. The corresponding column in the basis is to be replaced by a nonbasic column according to one of the steps (3.1)–(3.4) below. If the new b_{B_i} still violates $0 \leqslant b_{B_i} \leqslant 2$, this iteration is repeated again until it satisfies this condition. In the next iteration the scanning proceeds from $b_{B_{i+1}}$ and back again from b_{B_1}. Let at any iteration,

$k_j(C^T)$ be associated with b_{B_i}, and let $k_r(C^T)$ replace $k_j(C^T)$ in the basis. We consider one of two cases.

Case 1. If $b_{B_i} < 0$, $k_r(C^T)$ is determined from

$$\theta_r = \max(\theta_1, \theta_2) < 0, \tag{32}$$

where

$$\theta_1 = (z_r - f_r)/y_{ir} = \max_s\{(z_s - f_s)/y_{is}\}, \qquad y_{is} < 0, \quad s \in L(b), \tag{33a}$$

and

$$\theta_2 = (z_r - f_r)/y_{ir} = \max_s\{(z_s - f_s)/y_{is}\}, \qquad y_{is} > 0, \quad s \in U(b). \tag{33b}$$

(3.1) If $\theta_r = \theta_1$, transform the simplex tableau in the usual manner and go to step (2).

(3.2) If $\theta_r = \theta_2$, transform the tableau as usual, then add 2 to the new b_{B_i}. Remove the mark from column r indicating that b_r is no longer at its upper bound. Go to step (2).

Case 2. If $b_{B_i} > 2$, $k_r(C^T)$ is determined from

$$\tau_r = \min(\tau_1, \tau_2) > 0, \tag{34}$$

where

$$\tau_1 = (z_r - f_r)/y_{ir} = \min_s\{(z_s - f_s)/y_{is}\}, \qquad y_{is} > 0, \quad s \in L(b), \tag{35a}$$

and

$$\tau_2 = (z_r - f_r)/y_{ir} = \min_s\{(z_s - f_s)/y_{is}\}, \qquad y_{is} < 0, \quad s \in U(b). \tag{35b}$$

(3.3) If $\tau_r = \tau_1$, transform the tableau as usual. Mark column $k_j(C^T)$ to indicate that b_j is at its upper bound. Subtract $2y_j$ from b_B and go to step (2).

(3.4) If $\tau_r = \tau_2$, transform the tableau as usual, remove the mark from column $k_r(C^T)$ and place a mark on column $k_j(C^T)$. Add 2 to the new b_{B_i} and subtract $2y_j$ from b_B. Go to step (2).

The value of z in any stage of the calculation may be calculated from (31) or (22).

We here mention that in performing step (3) above, the decrease Δz in z, for cases 1 and 2, respectively, are

$$\Delta z = b_{B_i}\theta_r, \tag{36a}$$

and

$$\Delta z = (b_{B_i} - 2)\,\tau_r. \tag{36b}$$

LEMMA 6. *In the present algorithm, if for a* b_{B_i}, $0 \leqslant b_{B_i} \leqslant 2$, *the replacement of the corresponding column in the basis will not result in a decrease in the objective function z. This corresponds to the case in Usow's method when the* ith *equation in the corresponding reference equation set is not to be replaced by any equation not in the set.*

Proof. For $0 \leqslant b_{B_i} \leqslant 2$, this corresponds to $b_{B_i} \geqslant 0$, in the enlarged basis system to problem (16). The first part of the lemma follows from an elementary property of the dual simplex algorithm when applied to the enlarged system. The second part of the lemma follows from Lemma 4, bearing in mind that for the corresponding reference, the columns of the basis are the same rows of C in the reference equation set.

LEMMA 7. *The last vector which enters the basis in step* (3), *in the* ith *place, i.e., associated with* b_{B_i} *which violated* $0 \leqslant b_{B_i} \leqslant 2$, *corresponds in Usow's method to the same row in* (4) *which replaces the* ith *equation in the reference equation set, for the corresponding reference.*

Proof. First, we mention that the vectors which leave the basis, in the present algorithm, leave in succession, in accordance with Usow's method. We then discuss the vectos which enter the basis.

By examining (10), one sees that the matrix $(\phi_j(u_k^i))^T$ is itself the basis matrix B for the corresponding reference. Accordingly the vector $\Pi(x_s)$, $x_s \in X - U_k$, is the vector y_s of (17). Hence, in view of this and (26), t_r of (12) is itself θ_r or τ_r as given by (33) or (35), respectively. Again, characteristic (13) in Usow's method is equivalent to keeping all $(z_i - f_i) \geqslant 0$ in the consecutive tableaux in the enlarged system to problem (16). This is the feasibility condition for the dual simplex algorithm.

Hence, if b_{B_i} in step (3) above satisfied the condition $0 \leqslant b_{B_i} \leqslant 2$ after one iteration, the new basis corresponds to the nearest vertex to $(F_k, R(F_k))$ which is below it on the edge parallel to the ith parameter space coordinate axis.

However, if two or more iterations were needed for b_{B_i} to satisfy the condition $0 \leqslant b_{B_i} \leqslant 2$, the final basis corresponds to the vertex on K which by-passed one or more vertices from $(F_k, R(F_k))$. Then, since each iteration results in a decrease in z, the lemma is established by Lemma 6.

The next lemma and theorem follow from Lemmas 4–7.

LEMMA 8. *The value z given by an optimal basic feasible solution is equal to the optimal norm* (5) *and corresponds to the same reference.*

THEOREM 4. *Usow's algorithm is completely equivalent to a dual simplex algorithm applied to a linear programming problem in nonnegative bounded* *variables. However, one iteration in the former is equivalent to one or more iterations in the latter.*

Now, in view of (36), the choice of the columns which leave the basis, i.e., in succession, may not be the most economical one. If a maximum decrease in z is desired after every iteration, the vector to leave the basis may be chosen from

$$\max_{i,j}\{b_{B_i}\theta_{r_i}, (b_{B_j} - 2)\,\tau_{r_j}\}, \qquad b_{B_i} < 0, \quad b_{B_j} > 2. \tag{37}$$

Nevertheless, for problems with large m and n, different rules might be more convenient [5, p. 246]. Yet it is also reported by Hadley [5, p. 111], that for such problems, small differences in the number of iterations were observed when such different rules were used.

The following theorems follow as a result of the fact that the present algorithm is a linear programming one. They are restatements of Theorem 1 in Section 2.

THEOREM 5. *Let matrix C of* (4) (*or its equivalent* $(\phi_j(x_i))$ *of* (7)) *be of rank m. Then there exists an L_1 solution to* (4). *Further, there is a reference of m equations of* (4), *for which the residuals are zeros.*

THEOREM 6. *If the rank of C is less than m, there exists an L_1 solution to* (4), *such that there is a reference of fewer than m equations for which the residuals are zeros.*

Theorem 5 indicates that the Haar condition may be replaced by the requirement that C or $(\phi_j(x_i))$ is of rank m. While Theorem 6 indicates that even the latter condition may also be relaxed.

6. NUMERICAL RESULTS

In each of the following three examples, matrix C is of rank 2 and the first two columns of C^T are chosen to form the initial basis B. The matrix in the first example violates the Haar condition, while in the other two the matrices satisfy this condition.

The first example was solved by a limiting approach in [1]. Given the system of equations

$$\begin{aligned} a_1 + a_2 &= 3, \\ a_1 - a_2 &= 1, \\ a_1 + 2a_2 &= 7, \\ 2a_1 + 4a_2 &= 11.1, \\ 2a_1 + a_2 &= 6.9, \\ 3a_1 + a_2 &= 7.2, \end{aligned}$$

it is required to determine $a_1{}^*$ and $a_2{}^*$ which minimize the L_1 norm of the residuals.

The following are the initial data for the programming problem and the simplex tableaus for the algorithm described in the present work. The pivot in each tableau is bracketed and also by k_i we mean $k_i(C^T)$.

Initial Data

	f	3	1	7	11.1	6.9	7.2
B^{-1}	b_B	k_1	k_2	k_3	k_4	k_5	k_6
$\begin{pmatrix} 0.5 & 0.5 \\ 0.5 & -0.5 \end{pmatrix}$	10 8	1 1	1 −1	1 2	2 4	2 1	3 1

Tableau 1

					×	×	×	×
		f	3	1	7	11.1	6.9	7.2
f_B	B	b_B	k_1	k_2	k_3	k_4	k_5	k_6
3	k_1	$9 - 2(1.5 + 3 + 1.5 + 2) = -7$	1	0	1.5	3	1.5	(2)
1	k_2	$1 - 2(-0.5 - 1 + 0.5 + 1) = 1$	0	1	−0.5	−1	0.5	1
		$z = 8.2$	0	0	−3	−3.1	−1.9	−0.2

Tableau 2

					×	×	×	
		f	3	1	7	11.1	6.9	7.2
f_B	B	b_B	k_1	k_2	k_3	k_4	k_5	k_6
7.2	k_6	$-3.5 + 2 = -1.5$	0.5	0	0.75	1.5	0.75	1
1	k_2	4.5	−0.5	1	−1.25	(−2.5)	−0.25	0
		$z = 7.5$	0.1	0	−2.85	−2.8	−1.75	0

Tableau 3

				×	×		×	
		f	3	1	7	11.1	6.9	7.2
f_B	B	b_B	k_1	k_2	k_3	k_4	k_5	k_6
7.2	k_6	$1.2 - 2(0.6) = 0$	0.2	0.6	0	0	0.6	1
11.1	k_4	$-1.8 + 2 - 2(-0.4) = 1$	0.2	−0.4	0.5	1	0.1	0
		$z = 4.7$	0.66	−1.12	−1.45	0	−1.47	0

Tableau 1 is formed by multiplying the initial data by B^{-1}. However, since $z_i - f_i < 0$, $i = 3, 4, 5$, and 6, the corresponding b_i are given the value 2 and this is indicated by marks on such columns. The vector b_{B_0} is modified by subtracting $2(y_3 + y_4 + y_5 + y_6)$.

In Tableau 1, we can only replace $k_1(C^T)$ and this is done by observing step (3.2) of Section 5. The decrease in z is 0.7. Yet, in Tableau 2 we had the choice of replacing either $k_6(C^T)$ or $k_2(C^T)$ in the basis. The decrease Δz in each case is 2.8. We chose to replace $k_2(C^T)$, and this is done by observing step (3.4). Finally, in Tableau 3 the solution is found optimal, feasible, and degenerate with the sixth and fourth columns of C^T forming the final basis. Thus, by solving the sixth and fourth equation of this example, we get the vertex

$$a_1^* = 1.77, \qquad a_2^* = 1.89, \qquad \text{and} \qquad d^* = 4.7.$$

However, had we chosen to replace $k_6(C^T)$ in Tableau 2, the solution in Tableau 3 would have also been optimal and degenerate with the fourth and second columns of C^T forming the final basis. The corresponding vertex is

$$a_1^* = 2.5167, \qquad a_2^* = 1.5167, \qquad \text{and} \qquad d^* = 4.7.$$

The solution in [1] is $(a_1, a_2)^* = (2.0883, 1.7309)$. This point in fact lies on the horizontal line on the bottom of K joining the previous two vertices. Or

$$(2.0883, 1.7309) = \lambda(1.77, 1.89) + (1 - \lambda)(2.5167, 1.5167),$$

with $\lambda \simeq .426104$.

The second example was solved by the simplex method in [2, p. 298]. The same result was obtained by both Usow's and the present methods. In the latter method, the fourth and first columns of C^T formed the final basis. The solution is obtained after two iterations by Usow's method and after two iterations, i.e., from Tableau 3, by using (37), by the present method.

The third example was solved by an interval programming technique in [8, p. 328]. The solution by the present algorithm is found optimal feasible and degenerate from the first tableau.

7. Comments and Conclusion

Compared to existing known methods, the algorithm described in Section 5 seems to be the most efficient and capable one. Besides its simplicity, its efficiency is due to the advantage of using the dual simplex techniques.

No artificial variables are needed and as a result the computational effort is largely reduced. Obviously, in the case of rank deficiency of matrix C, a maximum of m artificial variables are needed to start the iteration. Also in this case the problem may be solved as a two phase problem [5, Chapter 5].

The capability of the present method is demonstrated by the fact that the Haar condition for matrix C may be completely relaxed, as is shown by Theorems 5 and 6. This advantage, however, is shared by other methods. Yet, these methods are less efficient than the present one.

In the method described by Hadley [5, pp. 393–394], the problem has always to be solved as a two phase one and in each iteration, one out of six possibilities arise. In [2], $(2n + 1)$ artificial variables are needed and the method in [8] is not as simple as the present one. Finally, the method in [1], though based on a simple principle [3, p. 233], converges very slowly.

References

1. N. N. Abdelmalek, Linear L_1 approximation for a discrete point set and L_1 solutions of overdetermined linear equations, *J. ACM* **18** (1971), 41–47.
2. I. Barrodale and A. Young, Algorithms for best L_1 and L_∞ linear approximations on a discrete set, *Numer. Math.* **8** (1966), 295–306.
3. E. W. Cheney, "Introduction to Approximation Theory," McGraw-Hill, New York, 1966.
4. W. Cheney and A. A. Goldstein, Note on a paper by Zuhovickii concerning the Tchebycheff problem for linear equations, *SIAM J. Numer. Anal.* **6** (1958), 233–239.
5. G. Hadley, "Linear Programming," Addison-Wesley, Reading, MA, 1962.
6. M. R. Osborne and G. A. Watson, On the best linear Chebyshev approximation, *Computer J.* **10** (1967), 172–177.
7. J. R. Rice, "The Approximation of Functions," Vol. 1, Addison-Wesley, MA, 1964.
8. P. D. Robers and A. Ben-Israel, An interval programming algorithm for discrete linear L_1 approximation problems, *J. Approximation Theory* **2** (1969), 323–336.
9. E. Stiefel, Note on Jordan elimination, linear programming and Tchebycheff approximation, *Numer. Math.* **2** (1960), 1–17.
10. K. H. Usow, On L_1 approximation. II. Computation for discrete functions and discretization effects. *SIAM J. Numer. Anal.* **4** (1967), 233–244.
11. C. W. Valentine and C. P. van Dine, An algorithm for minimax polynomial curve-fitting of discrete data. *J. ACM* **10** (1963), 283–290.
12. H. M. Wagner, Linear programming techniques for regression analysis, *J. Amer. Statist. Assoc.* **54** (1959), 206–212.