Some Comments on 'Linear' Programming with Absolute-Value Functionals

M. R. Rao

University of Rochester, Rochester, New York

(Received October 20, 1971)

This note gives simple and straightforward reasoning for obtaining the results of SHANNO AND WEIL [*Opns. Res.* 19, 120–124 (1971)] for linear programming with absolute-value functionals.

THE 'LINEAR' programming problem with absolute-value functionals, as stated by SHANNO AND WEIL,[5] is as follows: max $z=\sum c_j|x_j|$, subject to $Ax=b$, with some or all x_j unrestricted in sign.

As they pointed out, the transformation frequently used for each unrestricted variable (e.g., x_j) in the constraints is to replace it by two nonnegative variables (e.g., $x_j^+-x_j^-$) and to give each of the new variables the appropriate c_j in the objective function. For instance, this approach is utilized in absolute-value regression[1] and in linear programming under uncertainty.[2] A mathematical analysis is presented in reference 5 to show that the simplex method (with unrestricted basis entry) will converge to an absolute maximum only when $-c_j^-\geqq c_j^+$. The same result was obtained earlier by EL AGIZY[3] in connection with two-stage programming under uncertainty.

In order to see this result, one merely has to note that, if the above condition is not satisfied for some variable x_k, the transformed problem (i.e., replace x_k by $x_k^+-x_k^-$ and give the corresponding objective function coefficients as c_k^+ and c_k^-, but do not include the restriction $x_k^+\cdot x_k^-=0$) would have an unbounded solution whenever a feasible solution exists. This is because, once a feasible solution is obtained, we can then increase x_k^+ and x_k^- by an arbitrarily large but equal amount without violating the constraints. This, however, would make the objective function arbitrarily large, and consequently the solution is unbounded. The fact that the simplex method does not give a finite optimum when $c_k^+>-c_k^-$ should come as no surprise, since the transformed problem has an unbounded solution. In such a case, as is to be expected, the simplex method identifies an unbounded solution, as shown in reference 5.

An example is given in reference 5 to show that, if some c_j (corresponding to an unrestricted variable x_j) is positive, the linear programming problem with absolute-value functionals may have a local optimum that is not necessarily a global optimum. In order to see why this may happen, one merely has to note that, if any c_j is positive, the objective function (to be maximized) is not concave. If all c_j are nonnegative, the objective function becomes convex and, if the feasible region is strictly bounded, there exists an optimal solution that is an extreme point.[4] However, adjacent-extreme-point methods would fail in general because of local optima.

REFERENCES

1. A. Charnes and W. W. Cooper, *Management Models and Industrial Applications of Linear Programming*, Wiley, New York, 1961.
2. G. B. Dantzig, *Linear Programming and Extensions*, Princeton University Press, Princeton, New Jersey, 1963.
3. M. El Agizy, "Two-Stage Programming Under Uncertainty with Discrete Distribution Function," *Opns. Res.* **15**, 55–70 (1967).
4. G. Hadley, *Nonlinear and Dynamic Programming*, Addison-Wesley, Reading, Massachusetts, 1964.
5. D. F. Shanno and R. L. Weil, " 'Linear' Programming with Absolute-Value Functionals," *Opns. Res.* **19**, 120–124 (1971).

A Note on Reinverting the Dantzig-Wolfe Type Decomposed LP Basis

William J. Reich, Jr.

Falstaff Brewing Corporation, St. Louis, Missouri

(Received April 5, 1971)

This note reports an efficient routine that has been developed and successfully applied to the reinversion of decomposed LP bases of the Dantzig-Wolfe type. In sum, if r is the total number of subproblems in the original LP problem and k is the number of subproblem vectors in the decomposed basis, then only $k-r$ Gauss-Jordan iterations are needed to invert the decomposed basis. The larger r, the more efficient and accurate the routine. This reduction in the number of Gauss-Jordan iterations is accomplished by finding the inverse of the *wrong* basis rapidly, and then making the necessary corrections to get the inverse of the right basis.

THIS NOTE describes an efficient routine for reinverting decomposed LP bases of the Dantzig-Wolfe type. The routine is 'efficient' in the sense that it improves accuracy by eliminating much of the computation necessary for inversion.

Given the problem,

$$\begin{aligned} &\textstyle\sum_{j=1}^{j=r} A_j x_j + I_m x_s = b_0, \\ &B_j x_j = b_j, \qquad (j=1, \cdots, r) \\ &\textstyle\sum_{j=1}^{j=r} c_j x_j = \min z, \end{aligned}$$

where r is the number of subproblems; A_j is m by n_j for $j=1, 2, \cdots, r$; B_j is m_j by n_j; b_0 has m components; b_j has m_j components; I_m is an $m\times m$ identity matrix; x_s is a vector of m slack variables.

This 'decomposes' to,

$$\begin{aligned} &\textstyle\sum_{j=1}^{j=r} \sum_{k=1}^{k=h_j} P_{kj} A_j x_{kj}^* + I_m x_s = b_0, \\ &\textstyle\sum_{k=1}^{k=h_j} P_{kj} = 1, \qquad (j=1, 2, \cdots, r) \end{aligned}$$