J. Inst. Maths Applics (1973) **11**, *339-342*

On the Possibility of Replacing the Backward Transformation in the Modified Simplex Method by a Forward Transformation

M. MEZEI and T. SZONDY

United Hungarian Chemical Works, Budapest, Hungary

[Received 6 May, 19627]

It is shown that every backward transformation of the modified simplex method can be replaced by a forward transformation if both the original matrix and its transposed form are stored and transformed by two separate sets of inverse matrix factors. Replacing a backward transformation by a forward transformation may improve both numerical stability and speed.

1. Introduction

IN the usual form of the modified simplex algorithm (Orchard-Hays, 1969) two types of transformations are used for computing the transformed form of a row and a column; the backward and the forward transformations, respectively. However, both theoretical considerations and the numerical experience show that in general the backward transformation is numerically less stable and more time-consuming than the forward one.

The aim of the present paper is to show the possibility of using a forward transformation for computing a row (or a linear combination of rows) of the transformed matrix.

2. The Comparison of the Forward and Backward Transformations

The backward transformation can be regarded as the main weakness of the modified simplex algorithm for the following reasons:

(1) The backward transformation is generally numerically less stable, because the transformation begins with the last inverse factor, that is the factor most distorted by round-off errors, and consequently the round-off errors have better chances for propagation.

(2) The number of operations in the backward transformation is generally greater than that in the forward transformation for the following reasons:

(a) The number of operations needed in a backward respectively forward transformation can be estimated as follows. Let the matrix have n columns and m rows and let $p_o = N_o/(m \times n)$ and $p = N/(m \times n)$ where N_o and N denote the number of non-zero elements in the initial matrix and the transformed one, respectively. In this case a forward transformation produces a transformed column by roughly $2p^2km$ operations, a backward transformation produces a transformed row by roughly $2(pkm + p_omn)$ operations, where k is the number of inverse factors. The factor p^2 follows from the fact that if we have 0 in the pivotal row, the whole forward transformation step can be deleted.

(b) If the column to be transformed has already been in the basis, it is sufficient to begin the transformation with the step where the column finally left the basis. When we are transforming a row by the backward transformation, only those elements of the row can be computed more easily whose variable was at one time basic (Zoutendijk, 1960). This, however, results in complicated bookkeeping, which is not likely to pay off.

(3) If the backing store is magnetic tape and the magnetic tape station cannot work in reverse mode, the backward reading of the inverse matrix requires a high number of time consuming rewindings of the tape.

On the other hand, it is very advantageous that the backward transformation enables us to compute the transformed form of a linear combination of several rows without further computational effort.

3. The Modified Forward Transformation

Let us denote the pivotal row and pivotal column of the actual transformation by p and q respectively. The ordinary forward transformation is performed as follows:

$$\eta_p = 1/a_{pq}, \tag{1}$$

$$\eta_i = -a_{iq}/a_{pq} \qquad (i = 1, \ldots, m) \tag{2}$$

$$\overline{a}_{pj} = a_{pj} \times \eta_p, \qquad (j = 1, \ldots, n+m) \tag{3}$$

$$\overline{a}_{ij} = a_{ij} + a_{pj} \times \eta_i, \qquad (i \neq p) \tag{4}$$

where a_{ij} is an element of the extended matrix (that is a matrix which consists of the original matrix and an $m \times m$ unit matrix) before the transformation, $\overline{a}_{ij}$ is its transformed value, and η_i is an element of the inverse factor resulting from the present transformation.

On the basis of duality it can be shown that the above transformation is equivalent to the following one (see e.g. Beale, 1968):

$$
\begin{aligned}
\eta'_p &= 1/a'_{pq}, && && (1')\\
\eta'_i &= -a'_{iq}/a'_{pq} && (i = 1, \ldots, m) && (2')\\
\overline{a}'_{pj} &= a'_{pj} \times \eta'_p, && (j = 1, \ldots, n+m) && (3')\\
\overline{a}'_{ij} &= a'_{ij} + a'_{pj} \times \eta'_i, && (i \neq p). && (4')
\end{aligned}
$$

Here a'_{ij} is the element of a matrix that is composed from the transposed original matrix and a negative $n \times n$ unit matrix, q and p denoting the pivotal row and column respectively.

We can see at once that as the formulae (1)-(4) produce easily a column of the transformed matrix, the formulae (1')-(4') can produce a row in the same way.

If the matrix is sparse this kind of transformation possesses all the advantages of the forward transformation considered in Section 2.

4. The Case of the Objective Function Elements of the Slack Variables

If the slack variables have non-zero objective function elements and we start from a basis which involves the slack variables the matrix will not have the "orthodox" form as some of the basic variables will belong to non-unit vectors. The "orthodox" form of the matrix can, however, be restored by eliminating the non-zero objective function elements by normal simplex transformation steps before the actual calculation begins. The only effect of these pre-manipulatory transformations is that some rows are combined to the objective function row. This may be of interest in the following two cases:

(1) The objective function of the problem has the form discussed above. In this case we can compose the appropriate linear combination of the original rows and add it to the objective function.

(2) The artificial objective function. As it can change from iteration to iteration, it can be shown that we need a whole extra transformation whenever an infeasible basic variable becomes feasible but remains in the basis.

5. The Steps of the Suggested Algorithm

As the updating of the artificial objective function can require extra transformations, we suggest that a special dual algorithm which is, obviously, mainly composed of well-known steps be applied in the first phase. In this case no artificial objective function arises.

Assuming that the objective function has to be minimized the main steps will be as follows:

(1) The basic matrix, the right-hand side, the bounds of the variables, and the objective function are normalized in any reasonable way.

(2) The unit matrix of the slack and artificial variables is written after the basic matrix. Artificial variables have zero upper and lower bounds.

(3) The objective function is updated according to Section 4, if necessary.

(4) The basic matrix, the right-hand side and the objective function are manipulated in the following way:

(4.1) The columns multiplied by their lower bounds are subtracted from the right-hand side. All upper bounds of the variables are replaced by the respective differences of the upper and lower bounds and all lower bounds are set zero.

(4.2) The columns of all variables having negative objective function elements, and finite upper bounds are subtracted from the right-hand side multiplied by their upper bounds, then the columns are multiplied by -1 and it is registered that these variables are at upper bounds. (In columns with very high upper bounds this step may be omitted.)

(4.3) A "pseudo-objective function" is prepared by replacing all the remaining negative elements in the objective function by zero. These changes in the objective function are recorded. (This pseudo-objective function is dual feasible.)

(5) The basic matrix (without the unit matrix) is transposed and the negative unit matrix is written after the transposed matrix.

(6) A search for a primal feasible basis is carried out by the dual algorithm using the pseudo-objective function instead of the original one. It is done as follows:

(6.1) The pivotal row and pivotal column is prepared by the two forward transformations described in this paper.

(6.2) The right-hand side and the pseudo-objective function are transformed in every step by means of the pivotal column and row, respectively.

(6.3) If during the iterations any manipulated element (according to step (4)) of the pseudo-objective function becomes positive, the value of this element is lowered (if necessary, until zero) in order to reduce the effect of the manipulation.

(7) After having reached primal feasibility (or after feasibilizing an infea-

sible problem by force in any reasonable way) an intermediate first phase based on the primal algorithm is carried out in order to satisfy the remaining equalities with zero right-hand sides. (Such equalities may remain after the dual algorithm described above, and their elimination is straightforward.)

(8) The pseudo-objective function is replaced by the original one and a search for optimum is carried out by the second phase of the primal algorithm. Naturally, the transformations are carried out as in step (6.1) and (6.2).

(9) The second phase may result in either finding the optimum or finding that the objective function is unbounded.

6. Summary

It is suggested that a row of the transformed matrix be computed by a forward transformation. This can be advantageous for the following reasons:

(1) It can be expected that the round-off errors are smaller than in the case of using the backward transformation.

(2) If the matrix is sparse, the transformation time generally decreases.

(3) As a normal row is generally more sparse than the objective function, the time needed for its transformation is less than that of the objective function row, even if we use the backward transformation (Zoutendijk, 1960). The use of the forward transformation may increase this advantage.

(4) As the computation of a row requires the same effort as that of a column, the algorithm is well suited for applying dual and primal-dual algorithms.

(5) The use of magnetic tape stations can be made more efficient.

On the other hand its main drawbacks are the following:

(1) The transposed original matrix is needed. The transposing may take a significant amount of computer time, but it gives a possibility of seeing the matrix row-wise, which can, for example help to detect errors.

(2) We have to compute (and store) two series of inverse factors (the η and η' vectors). This means that we have to perform two forward transformations in every step. Consequently techniques such as the multiple pricing are losing their advantages.

The authors are indebted to Mr T. Revesz for inspiring discussions on the subject.

REFERENCES

BEALE, E. M. L. 1968 *Mathematical programming in practice.* Bath: Pitman.

ORCHARD-HAYS, W. 1969 *Advanced linear programming computing techniques.* New York: McGraw-Hill.

Z0UTENDIJK, G. 1960 *Methods of feasible directions.* Amsterdam: Elsevier.