RAIRO. RECHERCHE OPÉRATIONNELLE

JAMES K. HO

Pricing for sparsity in the revised simplex method

RAIRO. Recherche opérationnelle, tome 12, nº 3 (1978), p. 285-290

<http://www.numdam.org/item?id=RO_1978__12_3_285_0>

R.A.I.R.O. Recherche opérationnelle/Operations Research
(vol. 12, n° 3, août 1978, p. 285 à 290)

PRICING FOR SPARSITY IN THE REVISED SIMPLEX METHOD (*) (1)

by James K. Ho (2)

Abstract. — *This paper presents computational experience in using a column selection rule for the revised simplex method which tends to maintain sparsity of the basis. It is observed that in all cases tested, the average density of the basis inverse in product form is also reduced, and that in many cases an overall improvement in computational efficiency can be achieved.*

1. COLUMN SELECTION IN THE SIMPLEX METHOD

Let B be a primal feasible basis for the linear program to minimize

$$c_B x_B + c_N x_N$$

subject to

$$B x_B + N x_N = b,$$

$$x_B, \qquad x_N \geqq 0;$$

so that B^{-1} exists and $B^{-1} b \geqq 0$. The reduced costs in the revised simplex method [4] are given by

$$\bar{c}_N = c_N - \pi N \qquad \text{where} \quad \pi = c_B B^{-1}.$$

If $\bar{c}_N \geqq 0$, then $(x_B^*, x_N^*) \equiv (B^{-1} b, 0)$ is an optimal solution. Otherwise, any column N_j in N with $\bar{c}_{N_j} < 0$ can be introduced into the basis for an improved solution.

The standard column selection rule chooses N_j with the most negative $\bar{c}_{N_j}$. This corresponds to following that edge of the polyhedral feasible region which shows the greatest rate of decrease in the objective value. The virtue of the

(*) Received May 1977.

(1) Work performed under the auspices of the E.R.D.A. By acceptance of this article, the publisher and/or recipient acknowledges the U.S. Government's right to retain a nonexclusive, royalty-free license in and to any copyright covering this paper.

(2) Applied Mathematics Department, Brookhaven National Laboratory, Upton, N.Y.

standard rule, and hence its popularity in practice, is that given $\bar{c}_N$ no extra computation is required. Many other rules have been proposed and tested [2, 3, 5, 6, 7, 9]. In general, they are devised to enhance, at the cost of extra computation, the actual one-step decrease in the objective value so that the number of iterations to optimality may be reduced. However, none of them takes into account explicitly the sparsity of the basis.

2. SPARSITY AND THE REVISED SIMPLEX METHOD

As summarized in [1], sparsity is the main feature of linear programs in practice that has allowed the development of efficient variants of the simplex method. In particular, along with the revised simplex algorithm which keeps the basis inverse in some factorized form, reinversion techniques are available to seek compact representations of the inverse of a sparse basis [1, 8, 11].

Consider the revised simplex method with the product form of inverse (PFI). Here, the basis inverse is represented by what are commonly called ETA vectors stored compactly (i. e. nonzeroes only) in an ETA file [10]. With each change of basis, a new ETA vector is created to update the PFI. As a result, the size of the ETA file grows, implying more work for each subsequent iteration. To regulate this process, a reinversion is performed periodically to start a new, usually much more compact ETA file.

This suggests the following heuristics. By choosing a sparser sequence of bases, relative to, say, that implied by the standard rule, we may hope to:

(i) attain a more compact ETA file at each reinversion; and

(ii) reduce the growth of the ETA file between reinversions.

Then, less work per iteration may be expected and provided that the number of iterations required is not much more than the standard sequence, an overall improvement may be achieved. Moreover, (i) and (ii) imply a reduction of storage requirement for the ETA file as well as improved numerical accuracy.

3. A SPARSE COLUMN SELECTION RULE

To find a sparse sequence of bases we propose the following modification of the standard column selection rule. Let K_j be the number of non-zero coefficients in column N_j and

$$\bar{c}'_{N_j} = \bar{c}_{N_j} / K_j .$$

Then choose N_j with the most negative $\bar{c}'_{N_j}$. In other words, we use the standard rule with reduced costs weighted by a measure of the column sparsity.

Note that this modification is so simple that it requires only changing one instruction in any advanced LP code.

In general, any positive, non-increasing function in K_j can be used as a weighting factor for the usual reduced costs or any of their normalized forms given by other column selection rules [2, 3, 5, 6, 7, 9].

4. COMPUTATIONAL EXPERIENCE

For an empirical comparison of the sparse rule with the standard rule, a Fortran, in-core implementation [12] of the revised simplex method with product form of inverse was used. The reinversion routine is that described in [1] and [11]. Ten small to medium-size problems (averaging 340 constraints) from various real applications were solved using the two column selection rules. The dimensions as well as the models giving rise to the test problems are summarized in table I. The experiments with Problems 1, 4, 5, 7, 9 and 10 were performed on a CDC 7600 at Brookhaven National Laboratory, U.S.A., and the rest on an IBM 370/158 at the Catholic University of Louvain, Belgium.

TABLE I

The test problems

PROBLEM	NAME	APPLICATION	ROWS	COLUMNS	NONZEROS	% DENSITY
1	SCAGR7	Agricultural model	130	270	680	1.95
2	BESOM	U.S. energy model	166	498	5889	7.12
3	SC205	Economic growth model	206	409	758	0.90
4	SCTAP1	Traffic assignment	311	791	3683	1.50
5	SCFXM1	Production planning	331	788	2943	1.13
6	SCORPION	French energy model	389	747	2133	0.73
7	SCSD8	Optimal design in structural engineering	398	3148	11732	0.94
8	SCAGR25	Agricultural model	472	972	2501	0.55
9	SCRS8	U.S. Energy model	491	1660	4520	0.55
10	SCFXM2	Production planning	661	1575	5890	0.57

In each case, the maximum size of ETA allowed was 12,000 non-zero elements and the reinversion frequency was set at 30 iterations. The averages for the number of non-zero coefficients in the basis and ETA were taken over the total number of simplex iterations to optimality, starting from an all-logical basis. The relative performance of the sparse rule to the standard rule is presented in table II.

TABLE II

Performance of the sparse rule relative to the standard rule

RELATIVE STATISTICS / PROBLEM	AVERAGE NUMBER OF NONZEROS IN		ITERATIONS	CPU TIME	
	BASIS	ETA	I	T	T/I
1	0.94	0.76	1.05	0.97	0.92
2	0.70	0.69	1.09	1.09	0.99
3	0.99	0.85	1.00	0.94	0.94
4	0.85	0.95	0.92	0.87	0.95
5	0.91	0.89	1.04	1.03	0.98
6	0.90	0.81	1.00	1.00	1.00
7	0.94	0.85	0.73	0.70	0.96
8	0.84	0.76	0.72	0.60	0.83
9	0.98	0.89	0.82	0.77	0.94
10	0.94	0.93	1.07	1.06	0.99
MEAN	0.90	0.84	0.94	0.90	0.95
DEVIATION	0.08	0.08	0.14	0.16	0.05

Based on the results of our test problems, we make the following observations:

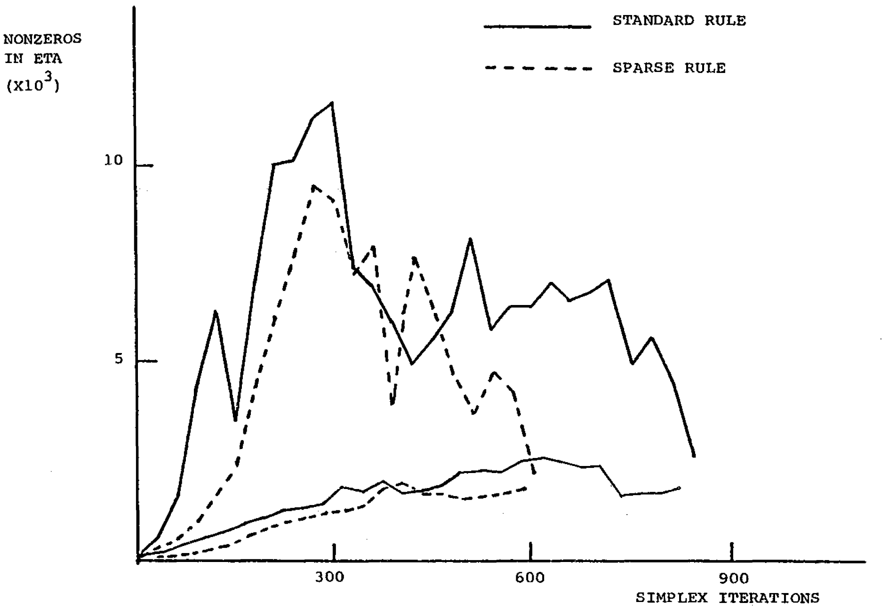
(*a*) the sparse rule tends to select a sparser simplex path. The average basis for the average case is reduced by 10%;

(*b*) the sparser path selected by the sparse rule has a smaller ETA file. The average ETA for the average case is reduced by 16%;

(*c*) the average time per iteration is reduced by 5% for the average case;

(*d*) in many cases, the sparser path is actually shorter than the standard path. When this occurs, an overall reduction in solution time can be expected (cf. *fig.*). Otherwise, it depends on the trade-off between the increase in iterations and the decrease in time per iteration;

(*e*) the sparse rule does not seem to do much harm even in the worst case encountered in our experiments. While the average case shows a 10% overall improvement, it is really the possibility of cases like Problems 7, 8 and 9 (30, 40 and 23 %, improvement respectively) that is of significance (*cf.* table II);



Growth of non-zeros in the ETA file for Problem 8.
In each case, the upper graph represents the situation right before a reinversion, the lower graph right after.

(*f*) since the sparse rule can be obtained from the standard rule by modifying only one instruction to the computer, it should be a useful option in a LP code.

ACKNOWLEDGEMENT

The author is grateful to the Center for Operations Research and Econometrics, Catholic University of Louvain, Belgium, for supporting part of this work during his visits. He wishes to thank Etienne Loute for many helpful discussions as well as assistance in the computer experiments, and John Tomlin for providing the LPM1 code.

REFERENCES

1. E. M. L. Beale, *Sparseness in Linear Programming, in Large Sparse Sets of Linear Equations*, J. K. Reid, éd., Academic Press, London, 1971, pp. 1-15.
2. H. Crowder and J. M. Hattingh, *Partially Normalized Pivot Selection in Linear Programming*, Mathematical Programming Study, Vol. 4, 1975, pp. 12-25.

3. L. CUTLER and P. WOLFE, *Experiments in Linear Programming*, in Recent advances in mathematical programming R. L. GRAVES and P. WOLFE, éd. McGraw-Hill, New York, 1963, pp. 177-200.
4. G. B. DANTZIG, *Linear Programming and Extensions*, Princeton University Press, Princeton, N.J., 1963.
5. J. C. DICKSON and F. P. FREDERICK, *A Decision Rule for Improved Efficiency in Solving Linear Programming Problems with the Simplex Algorithm*, Communications of the Association for Computing Machinery, Vol. 3, 1960.
6. D. GOLDFARB, *Using the Steepest-Edge Simplex Algorithm to Solve Sparse Linear Programs* in Sparse matrix computations, J. R. BUNCH and D. ROSE, ed., Academic Press, New York, 1976, pp. 227-240.
7. P. M. J. HARRIS, *Pivot Selection Methods of the Devex LP Code*, Mathematical Programming Study, Vol. 4, 1975, pp. 30-57.
8. E. HELLERMAN and D. RARICK, *Reinversion with the Preassigned Pivot Procedure*, Mathematical Programming, Vol. 1, 1971, pp. 195-216.
9. H. W. KUHN and R. E. QUANDT, *An Experimental Study of the Simplex Method*, Proceedings of symposia in applied mathematics, Vol. 15, Amer. Math. Soc., Providence, R.I., 1963.
10. W. ORCHARD-HAYS, *Advanced Linear Programming Computing Techniques*, McGraw-Hill, New York, 1968.
11. J. A. TOMLIN, *Pivoting for Size and Sparsity in Linear Programming Inversion Routines*, J. Inst. Math. and Appl., Vol. 10, 1972, pp. 289-295.
12. J. A. TOMLIN, *LPM1 user's Manual*, Systems Optimization Laboratory, Department of Operations Research, Stanford University, 1973.