NBER WORKING PAPER SERIES

FINDING A DUAL-FEASIBLE SOLUTION TO AN LP WITH m EQUALITIES IN (1 & m) DUAL ITERATIONS

by Vinay Dharmadhikari

Working Paper No. 100

National Bureau of Economic Research, Inc.
575 Technology Square
Cambridge, Massachusetts 02139

August 1975

Preliminary: Not for Quotation

Abstract

Lemke's dual-simplex method of linear programming is usually considered inferior to the primal simplex method for any general linear programming problems. One reason given is the difficulty of finding a starting dual-feasible basis. In this paper, a new starting technique is presented, which finds a dual-feasible basis in a single dual-simplex pivot for LP's with no equality constraints, and in $(1+m_3)$ pivots for LP's with m_3 equality constraints irrespective of the number of inequality constraints. The technique is illustrated on a small example problem. The performance, in terms of the number of pivots to optimality, of the dual-simplex with the new starting technique on 100 medium sized problems is reported and compared with that of the primal simplex. Finally, how the dual-simplex with the new starting technique can be efficiently implemented is briefly discussed.

Contents

Section 1: Introduction

Since 1954 when Lemke presented the dual-simplex method for solving linear programs, the method has been considered no better, and perhaps worse, than the primal simplex method. One of the reasons stated in the literature and held in the folklore is the difficulty of finding an initial dual-feasible basis so that the dual-simplex can be started. In this report we will present a new technique to obtain a dual-feasible basis. We will also discuss the suitability of the dual-simplex method combined with the new starting technique as a general purpose linear programming method.

Consider the general linear programming problem:

Problem I: Maximize $\bar{c}\bar{x}^t$

Subject to $A\bar{x}^t \rho\ \bar{b}^t$ Problem I

$\bar{x}^t \geq \bar{0}$

where $\bar{c} = (c_1, c_2, \ldots c_n)$ is the vector of objective coefficients, $A = ||a_{ij}||$ is an m by n constraint matrix, $\bar{x} = (x_1, x_2, \ldots x_n)$ is the vector of n variables, ρ is a vector composed of relations $\leq$, $\geq$ and $=$, and $\bar{b} = (b_1, b_2, \ldots b_m)$ is the vector of right-hand-sides. Let the first m_1 constraints be $\leq$ type, the next m_2 be $\geq$ type and the last m_3 be = type, where $m_1 + m_2 + m_3 = m$. The j-th column of A is denoted by A_j, the basis matrix by B and the r-th row of B^{-1} by β^r.

When $c_j \leq 0$ for all j, and each constraint has a surplus variable, the all logical-basis is automatically dual-feasible. When only $c_j \leq 0$ for all j, Dantzig [2] has given a technique that involves the direct use of the dual and a basis of size n. Lemke [6] suggested a technique that assumes that a set of m linearly independent columns of A is known. The Lemke technique first solves an identical problem with a different right-hand-side vector by revised simplex and uses the optimal basis as the dual-feasible starting basis for the original problem. The artificial constraint technique [1] [7] assumes knowledge of a basis $\bar{x}_B$, possibly infeasible. It involves pricing the nonbasic variables to determine the variables x_j, $j \varepsilon J$ with negative revised prices and the variable x_p with the smallest revised price. An artificial constraint $\sum_{j \varepsilon J} x_j + x_0 = M$ is added to the problem where M is a sufficiently large number. Substituting $x_p = M - x_0 - \sum_{\substack{j \varepsilon J \\ j \neq p}} x_j$ in all other constraints and the objective function yields a problem for which the basis $\bar{\bar{x}}_B = (\bar{x}_B, x_p)$ is dual-feasible. Wagner [8] has given a modified dual-simplex algorithm for upper bounded variables which, he states, can also be used as a starting technique by providing artificially large, non-restrictive upper bounds.

Section 2: The New Starting Technique

The new technique does not require any special conditions nor the knowledge of a set of m linearly independent columns of A. It starts by multiplying the m_2 constraints of $\geq$ type by -1. A logical variable is added to each of the m constraints with a zero objective coefficient. The last m_3 logical variables in the = type constraints are regarded as artificial variables. Let $c_p = \text{MAX}\ \{c_j,\ j = 1, 2, \ldots n\}$. An artificial, non-restrictive constraint $x_0 + \sum_{j=1}^{j=n} x_j = b_0$ is added to the problem where b_0 is sufficiently large. The objective coefficient c_0 of x_0 is set equal to c_p initially; it will be reset to zero after the first dual pivot. With these modifications, our problem becomes as follows where A_j now denotes the j-th column of the constraint matrix of the modified problem.

Problem II: Maximize $\sum_{j=1}^{j=n} c_j x_j + c_p x_0$

Subject to

$$\sum_{j=1}^{j=n} x_j + x_0 = b_0$$

$$\sum_{j=1}^{j=n} -a_{ij} x_j + x_{n+i} = b_i, \quad i = 1, 2, \ldots m_1$$

$$\sum_{j=1}^{j=n} a_{ij} x_j + x_{n+i} = -b_i, \quad i = m_1+1, \ldots m_1+m_2$$

$$\sum_{j=1}^{j=n} a_{ij} x_j + x_{n+i} = b_i, \quad i = m_1+m_2+1, \ldots m_1+m_2+m_3$$

$$x_j \geq 0, \quad j = 0, 1, \ldots n+m$$

We start with a basis consisting of all slack, surplus and artificial variables so that the basis matrix B equals identity matrix of size (m+1) by (m+1). The row vector of objective coefficients of the basic variables is $\bar{c}_B = (c_p, 0, 0, . ., 0)$. Since $B = I$, $B^{-1} = I$. The revised prices of the nonbasic columns are given by,

$$\begin{aligned} \pi_j &= \bar{c}_B B^{-1} A_j - c_j \\ &= \bar{c}_B A_j - c_j \\ &= (c_p, 0, 0, . . ., 0) A_j - c_j \\ &= c_p - c_j \end{aligned}$$

Therefore $\pi_j = c_p - c_j \geq 0$ for $j = 1, 2, . . . n+m$. Thus we have a dual feasible basis for problem II. We then apply the dual-simplex method to problem II, with following important modifications.

Section 3: Modifications of the Dual-Simplex

1. At the very first iteration, choose x_0 as the leaving variable and x_p as the entering variable. Since $c_0 = c_p$, $\bar{c}_B$ remains unchanged after the pivot and the new basis remains dual-feasible. Now that the variable x_0 is out of basis, we change its objective coefficient to zero so that the objective function becomes identical with that of the original problem I.

2. At all subsequent iterations, choose as the leaving variable one of the artificial variables $x_{n+m_1+m_2+1}, . . ., x_{n+m}$, if any artificial variable remains in the basis. Once the leaving variable is selected, determine the variable to enter the basis by the usual dual-simplex rules, except in the case where $y_{rj} = \beta^r A_j \geq 0$ for all j. Thus the rule can be stated as: choose x_k as the variable to enter the basis a) if $\exists\, y_{rj} < 0$ then k is given by

$$\begin{aligned} \frac{z_k - c_k}{y_{rk}} &= \max \frac{z_j - c_j}{y_{rj}}, \; y_{rj} < 0 \\ &= \min \left| \frac{z_j - c_j}{y_{rj}} \right|, \; y_{rj} < 0 \end{aligned}$$

b) if $y_{rj} \geq 0$ for all j, then k is given by[1]

$$\frac{z_k - c_k}{y_{rk}} = \min \frac{z_j - c_j}{y_{rj}}, \; y_{rj} > 0$$

$$= \min \left| \frac{z_j - c_j}{y_{rj}} \right|, \; y_{rj} > 0$$

We can see that the above modifications to the usual dual-simplex procedure preserve the dual-feasibility of the basis, and allow the m_3 artificial variables to be driven out of the basis in no more than m_3 pivot operations. Once an artificial variable is out of the basis, it can be eliminated from the problem and never considered as a candidate for an entering variable. If there is a negative variable x_r in the basis with $y_{rj} = \beta^r A_j \geq 0$ for all j, then the original problem is infeasible. If when all artificial variables are removed, x_0 is not in the basis, then the original problem is unbounded.

Section 4: Elimination of Equalities

Since the number of pivots to be performed in this technique in obtaining a dual feasible basis depends on the number m_3 of equality constraints in the original problem I, it is worthwhile to see if the number of equalities can be reduced. Given an equality $\sum_{j=1}^{j=n} a_{ij} x_j = b_i$, express one variable, say x_k, $a_{ik} \neq 0$ in terms of the other variables as $x_k = \frac{-1}{a_{ik}} \left(\sum_{\substack{j=1 \\ j \neq k}}^{j=n} x_j + b_i \right)$. This expression can be substituted wherever x_k occurs in the other constraints and the objective function, and the non-negativity of x_k can be preserved by replacing the i-th equality in the original problem by the inequality $-\frac{1}{a_{ik}} \left(\sum_{\substack{j=1 \\ j \neq k}}^{j=n} x_j + b_i \right) \geq 0$. This can be repeated for all other equalities, thus eliminating all equality constraints one by one. The above procedure can always be used to obtain an exactly equivalent problem with no equality constraints, and the new starting technique applied to this

[1] This was suggested by Professor Kennington.

Table 1. Description of Test Problems

Set No.	N	M1	M2	M3	Range of a_{ij}	Range of c_j
1	340	1	3	3	[511, 735]	[42, 88]
2	243	7	5	7	[342, 636]	[29, 48]
3	368	2	7	8	[-553, -429]	[65, 119]
4	85	8	8	2	[110, 164]	[-39, 34]
5	366	1	2	5	[213, 763]	[-67, 55]
6	157	7	7	4	[72, 725]	[-64, 62]
7	115	1	5	1	[961, 969]	[29, 36]
8	382	2	6	8	[234, 363]	[01, 62]
9	295	2	2	3	[711, 884]	[-95, 48]
10	349	4	4	5	[104, 277]	[14, 15]

Table 2. Total Number of Primal Pivots and Dual Pivots Required to Obtain an Optimal Solution

Prob. Set	Problem Number									
	1	2	3	4	5	6	7	8	9	10
1	28,17	26,9	23,17	33,18	35,8	39,17	24,15	26,23	32,15	26,20
2	92,52	83,43	80,50	76,34	99,58	83,67	97,38	100,32	81,40	73,49
3	53,46	68,42	59,41	59,52	79,43	66,40	63,26	68,59	55,42	60,52
4	46,21	61,15	63,17	50,17	53,21	44,13	68,27	55,15	75,20	70,17
5	38,20	37,16	21,15	29,12	37,16	29,15	29,20	26,18	37,21	36,14
6	82,25	81,14	89,23	70,26	62,29	85,26	75,36	54,26	76,25	82,27
7	20,4	21,5	24,6	23,4	25,5	27,4	22,5	22,11	23,5	26,5
8	103,31	87,26	72,46	80,29	70,40	70,27	100,37	77,35	77,46	71,52
9	21,15	18,9	28,7	18,9	21,9	20,7	25,9	27,9	25,10	25,9
10	94,46	84,54	79,61	74,51	89,68	69,56	63,44	86,53	69,51	79,53

The first entry in each pair is the total number of primal pivots and the second entry is the total number of dual pivots.

equivalent problem will yield a dual feasible basis in exactly one pivot operation. A significant advantage of this equality-elimination procedure is that, unlike the variable substitution in the artificial-constraint starting technique [7], our procedure can be carried out a priori -- in the problem generation phase and hence it need entail little additional effort.

Section 5: Comparison With Other Starting Techniques

1. Whereas the other starting techniques require presence of special conditions (e.g. Dantzig's) or advance knowledge of a set of m linearly independent columns of A (e.g. Lemke's), the new technique is completely general.
2. Unlike any other technique, the new technique guarantees a dual-feasible starting basis in $(1+m_3)$ dual-simplex pivots. And, if the equality-elimination procedure is used, a starting basis can be obtained in just one pivot operation.
3. The techniques such as Charnes' Big M, or Phase I - Phase II, for obtaining a primal-feasible starting basis involve introduction of artificial variables not only in the m_3 equality constraints but also in the m_2 $\geq$ type constraints. Moreover, since in these primal methods, there is no freedom of choosing the leaving variable, the number of pivot operations required to obtain a starting basis can be, and usually is, quite large. The new technique for obtaining a a dual-feasible basis, on the other hand, does not need any artificial variables in the $\geq$ type constraints.
4. Compared with other starting techniques, the new technique is probably the simplest to code and involves very few changes from the usual dual-simplex method.

Section 6: Conclusion

We presented a new technique for obtaining a dual-feasible basis for a general linear programming problem. We have shown how the number of pivots required to obtain the dual-feasible basis can be bounded from above by $(1+m_3)$ where m_3 is the number of strict equality constraints in the original linear program. We have given an equality-elimination procedure that can very easily be used to further reduce the number of pivots. Finally we have shown that the new technique compares favorably with other starting techniques. It is felt, in the light of the above that the dual-simplex method of Lemke needs to be reexamined for its suitability as a general purpose optimization method for solving linear programming problems. In addition to the very efficient starting technique for the dual-simplex, there is some evidence of the superiority of the dual-simplex over primal-simplex in two empirical investigations. The first by Hasegawa [4] was done in 1965 as a part of a master's thesis. Twenty-five problems with thirty constraints and seventy variables were randomly generated with five percent dense constraint matrices. A modified Lemke technique was used to obtain a starting dual-feasible basis for the dual-simplex, and the two-phase technique to obtain a starting dual-feasible basis for the primal method, revised-simplex. For each problem solved, the numbers of iterations of the dual-simplex were uniformly smaller than those for the revised-simplex. In fact, in many cases, the former required less than half as many iterations as the latter.

The second computational study by Kennington [5] was carried out primarily to confirm or contradict Hasegawa's results. Instead of the modified Lemke starting technique as used by Hasegawa, the new starting technique presented in this paper was used to get a starting basis for the dual-simplex, and the Charnes' big-M method was used to get a starting basis for the primal-simplex. One hundred problems with upto twenty nine constraints and 420 variables were randomly generated with completely dense constraint matrices. The sizes of the problems, the ranges of a_{ij} and c_j and other statistics for the problems are given in Table 1. The number of iterations taken by both methods to reach optimal solution are given in Table 2. Here again we see that the dual-simplex combined with the new starting technique took uniformly less iterations than did the primal simplex.

Thus we now have evidence from two independent investigators that for medium-scale linear programming problems of either low or high sparsity, the dual-simplex method worked uniformly better in terms of the number of iterations. Incidentally, the starting technique used by Hasegawa seems to take more iterations than those for the technique presented here.

It is felt, however, that the evidence presented above warrants a reevaluation of the dual-simplex with the new starting technique, as a linear programming method which may compete with other methods such as the revised-simplex, the primal-dual, the composite-simplex [1]. Of course, the iteration count by itself is not the most meaningful measure of a method's performance; the computational effort per iteration also needs to be taken into consideration. At the first glance a dual-simplex iteration seems to involve more effort because in addition to evaluation of all revised prices $\pi_j = \bar{c}_B B^{-1} A_j - c_j$ for all nonbasic j, it requires evaluation of $y_{rj} = \beta^r A_j$ for all nonbasic j, where β^r is the r-th row of B^{-1}. This situation, however, can be improved. As Kennington [5] has pointed out, the π_j need be computed only for those j for which $y_{rj} < 0$. Moreover, the r-th row of B^{-1} can be conveniently obtained by using the column-access, packed-matrix and the use of the product-form-of-inverse comprising of k elementary matrices at the k-th iteration since the last reinversion. If $B^{-1} = E_k E_{k-1} \ldots E_1$, then $\beta^r = e^r E_k E_{k-1} \ldots E_1$, where $e^r = (0, 0, \ldots 1, 0 \ldots 0)$ with a 1 in the r-th position. The multiplication of packed eta-vectors representing the elementary matrices E_j with a sparse row vector can be performed quite economically using clever schemes such as that of Winograd's [9]. When a nonbasic column k has $y_{rk} = \beta^r A_k < 0$ and $\pi_k = \bar{c}_B B^{-1} A_k - c_k = 0$, we have dual-degeneracy. In this case, one need not compute y_{rj} for all j because the variable x_k can be chosen as the entering variable while preserving dual-feasibility since the π_j will remain unaltered after the iteration. Thus the presence of dual-degeneracy actually can reduce the computation per iteration in determining the entering variable corresponding to a given leaving variable. It may also be pointed out that the dual-simplex method can use most of the efficient and storage saving techniques of programming used in the revised primal-simplex; in fact, many subroutines in a production LP code are directly transferrable, such as matrix packing, the product-form-of-inverse, reinversion, BTRAN and FTRAN operations, and so on. Thus we see that the dual-simplex, with the new starting technique not only shows great promise in terms of efficiency, but the change over from a primal LP code can be quite simple and inexpensive.

Many questions concerning the uniformly better iteration count performance of the dual-simplex as evidenced in the computational studies to date, yet remain to be theoretically as well as empirically investigated for root causes and explanations. But then there are many questions in the field of linear programming that are not fully answered yet; namely, for example, why most linear programs are solved in an extremely small fraction of the total number of iterations it could theoretically take for any Simplex procedure? Specifically, however, the questions of better iteration count performance, of estimating the actual computational effort per iteration, and of whether there are any special structures such as network problems, covering problems, decomposition problems that are especially amenable to the dual-simplex will be investigated and the findings will be reported in the future.

Appendix A: Illustrative Example

Problem I:

Maximize $3x_1 + 6x_2 + 8x_3$

Subject to

$$4x_1 + 8x_2 - 2x_3 \geq 8$$

$$2x_1 + 4x_2 + 4x_3 \leq 12$$

$$x_1 - 2x_2 + 4x_3 = 0$$

$$x_j \geq 0 \text{ for all } j$$

We add the new constraint $x_0 + x_1 + x_2 + x_3 = 100$. We evaluate the largest objective coefficient c_p,

$$c_p = \max_j \{3, 6, 8\}$$

$$c_0 = c_p = 8, \ p = 3.$$

Thus the modified problem is,

Problem III:

Maximize $8x_0 + 3x_1 + 6x_2 + 8x_3$

$$\begin{aligned} x_0 + x_1 + x_2 + x_3 &= 100 \\ -4x_1 - 8x_2 + 2x_3 + x_4 &= -8 \\ 2x_1 + 4x_2 + 4x_3 + x_5 &= 12 \\ x_1 - 2x_2 + 4x_3 + x_6 &= 0 \end{aligned}$$

$$x_j \geq 0 \text{ for all } j$$

Tableau 1

	$\bar{c}_B$	$\bar{x}_B$	$\bar{b}$	A_0	A_1	A_2	A_3	A_4	A_5	A_6
			$\bar{c}_B$	8	3	6	8			
←	8	x_0	100	1	1	1	[1]			
	0	x_4	-8		-4	-8	2	1		
	0	x_5	12		2	4	4		1	
	0	x_6	0		1	-2	4			1
			π	0	5	2				
							↑			

Tableau 2

	$\bar{c}_B$	$\bar{x}_B$	$\bar{b}$	A_0	A_1	A_2	A_3	A_4	A_5	A_6
			$\bar{c}_B$	0	3	6	8			
	8	x_3	100	1	1	1	1			
	0	x_4	-208	-2	-6	-10		1		
	0	x_5	-388	-4	-2				1	
←	0	x_6	-400	-4	-3	-6				1
			π	0	5	2				

Tableau 3

$\bar{c}_B$	$\bar{x}_B$	$\bar{b}$	A_0	A_1	A_2	A_3	A_4	A_5	A_6
8	x_3	100/3	1/3	1/2		1			
	x_4	1376/3	14/3	-1			1		
	x_5	-388	-4	-2				1	
6	x_2	400/6	2/3	1/2	1				-1/6
		π	20/3	4					2/6

Tableau 3 gives the dual-feasible starting solution to problem III, since the basis contains no artificial variables corresponding to the original equality constraints.

REFERENCES

1. Cooper, L. and Steinberg, D. I. Methods and Applications of Linear Programming, W. B. Saunders and Co., Philadelphia, Pa., 1974.

2. Dantzig, G. B., Ford, L. R., and Fulkerson, D. R. "A primal-dual algorithm for linear programming," Linear Inequalities and Related Systems, Kuhn and Tucker, eds., Princeton Univ. Press, Princeton, N. J., 1956.

3. Hadley, G. Linear Programming, Addison-Wesley Publishing Co., Reading, Mass., 1962.

4. Hasegawa, H. "A study of the dual simplex algorithm," M.S. Thesis, Washington Univ., St. Louis, Mo., 1965.

5. Kennington, Jeff L. "Computational Experience Using the Dual Simplex with Dharmadhikari's Starting Technique," Technical Report CP 74014, Southern Methodist University, Dallas, Texas, 1974.

6. Lemke, C. E. "The dual method of solving the linear programming problems," Naval Research Logistics Quarterly, 1, 48-54, 1954.

7. Simmonard, M. Linear Programming, Prentice-Hall, Englewood Cliffs, N. J., 1966.

8. Wagner, H. M. "The dual simplex algorithm for bounded variables," Naval Research Logistics Quarterly, 5, 257-261, 1958.

9. Winograd, S., "A new algorithm for inner-product," IEEE Transactions, Computers, 17, 693-694, 1968.