A Mutual Primal-Dual Simplex Method

Michel L. Balinski
Ralph E. Gomory

1. SIMPLEX METHODS

A pair of dual linear programs

Primal (Row) Program

Minimize

$$y_0 = a_{00} + a_{01} y_{m+1} + \cdots + a_{0n} y_{m+n}$$

constrained by

$$-y_1 = a_{10} + a_{11} y_{m+1} + \cdots + a_{1n} y_{m+n} \leq 0$$
$$\vdots$$
$$-y_m = a_{m0} + a_{m1} y_{m+1} + \cdots + a_{mn} y_{m+n} \leq 0$$
$$y_{m+1} \geq 0, \quad \ldots, \quad y_{m+n} \geq 0$$

Dual (Column) Program

Maximize

$$x_0 = a_{00} + x_1 a_{10} + \cdots + x_m a_{m0}$$

constrained by

$$x_1 \geq 0, \quad \ldots, \quad x_m \geq 0 \tag{1.1}$$
$$x_{m+1} = a_{01} + x_1 a_{11} + \cdots + x_m a_{m1} \geq 0$$
$$\vdots$$
$$x_{m+n} = a_{0n} + x_1 a_{1n} + \cdots + x_m a_{mn} \geq 0$$

is conveniently exhibited in the tableau

Primal Program

	1	y_{m+1}	$\cdots$	y_{m+n}	
1	a_{00}	a_{01}	$\cdots$	a_{0n}	$= y_0$
x_1	a_{10}	a_{11}	$\cdots$	a_{1n}	$= -y_1$
$\vdots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
x_m	a_{m0}	a_{m1}	$\cdots$	a_{mn}	$= -y_m$
	$= x_0$	$= x_{m+1}$	$\cdots$	$= x_{m+n}$	

(1.2)

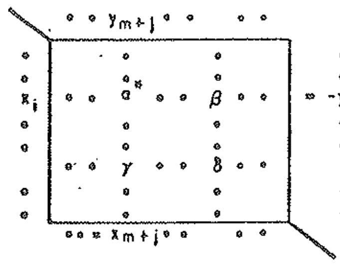
Dual Program

In the dual linear programs (1.1) or their tableau (1.2) $y_0, y_1, \ldots, y_m$, the basic variables of the primal program are expressed in terms of $y_{m+1}, \ldots, y_{m+n}$, the nonbasic variables; similarly, $x_0, x_{m+1}, \ldots, x_{m+n}$, the basic variables of the dual program, are expressed in terms of $x_1, \ldots, x_m$, the nonbasic variables.

A pivot step on (1.1) or (1.2) with pivot entry $a_{ij} \neq 0$ $(i, j \neq 0)$ is a

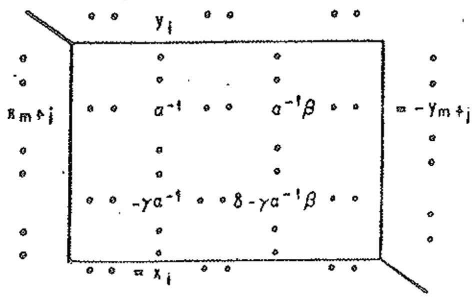
Proceedings of Mathematical Programming Symposium, 1963, pp. 17-26.

Gauss-Jordan or complete elimination step which simultaneously solves the i^{th} (row) equation of the primal for y_{m+j} and the j^{th} (column) equation of the dual for x_i, and uses these equations to eliminate y_{m+j} and x_i from the remaining row and column equations at the cost of introducing y_i and x_{m+j}. The y_{m+j} and x_i thereby become basic variables and y_i and x_{m+j} nonbasic variables. The pivot step with pivot entry $\alpha = a_{ij} \neq 0$ takes the tableau



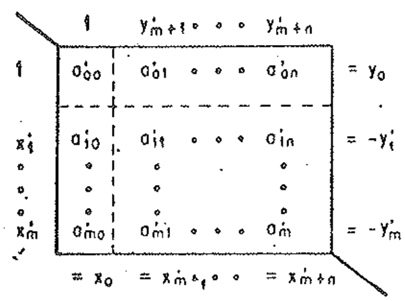
(1.3)

into the tableau



(1.4)

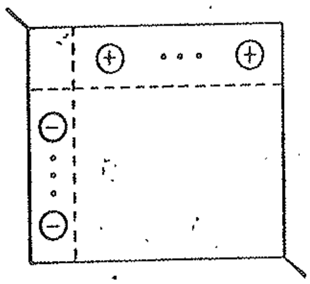
the other marginal variables and labels remaining in the same positions. Successive tableaus obtained by pivot steps simply reexpress the original pair of dual linear programs through different partitions into sets of basic and nonbasic variables. Any such tableau has the form



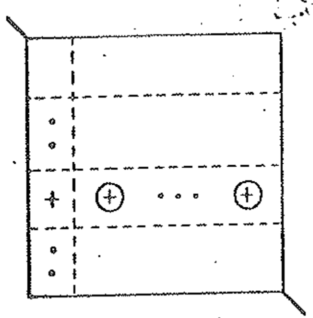
(1.5)

where the primed variables are a rearrangement of the original variables and the primed entries are determined by the succession of preceding pivot steps. Basic solutions to both programs are associated with any tableau (1.5); they obtain by setting the nonbasic variable equal to zero, thereby determining values for the basic variables $y_1' = -a_{10}', \ldots, y_m' = -a_{m0}'$, $y_0 = a_{00}' = x_0$, $x_{m+1}' = a_{01}', \ldots, x_{m+n}' = a_{0n}'$. If $-a_{10}' \geqq 0, \ldots, -a_{m0}' \geqq 0$ a basic feasible primal solution obtains; if $a_{01}' \geqq 0, \ldots, a_{0n}' \geqq 0$ a basic feasible dual solution obtains. If both primal and dual basic feasible solutions obtain, then they constitute optimal solutions to the programs.

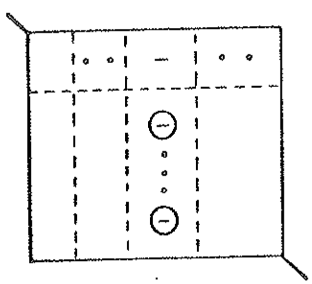
A simplex method for solving a pair of dual linear programs is a finite sequence of tableaus exhibiting equivalent pairs of dual linear programs obtained by successive pivot steps, with prescribed pivot entry choice rules, which obtain a tableau exhibiting optimal solutions to both programs, or the noncompatibility of the primal and/or dual constraints. Letting $\oplus$ denote nonnegative entries, and $\ominus$ nonpositive entries, these cases can be exhibited in tableau form:



(optimal solutions) (1.6)



(primal or row constraints noncompatible) (1.7)



(dual or column constraints noncompatible) (1.8)

A primal (dual) simplex method is a simplex method beginning with a tableau exhibiting a primal (dual) basic feasible solution with pivot steps which maintain primal (dual) feasibility in each succeeding tableau. A primal (dual) pivot choice rule is as follows:

If a tableau (1.5) does not exhibit optimal solutions to both programs there must exist a $a'_{0j} < 0$ for some j (a $a'_{i0} > 0$ for some i). Either (a) every entry in the column of $a'_{0j} < 0$ is nonpositive (every entry in the row of $a'_{i0} > 0$ is nonnegative) or (b) there exist positive (negative) entries. (a) The tableau exhibits the noncompatibility of the dual constrains (of the primal constraints).
(b) Choose as pivot entry $a'_{kj} > 0$ ($a'_{i\ell} < 0$) satisfying

$$\frac{a'_{k0}}{a'_{kj}} = \max_{a'_{sj} > 0} \frac{a'_{s0}}{a'_{sj}} \qquad \left(\frac{a'_{0\ell}}{a'_{i\ell}} = \max_{a'_{is} < 0} \frac{a'_{0s}}{a'_{is}}\right)$$

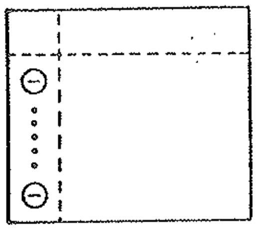
If an initial tableau does not exhibit a primal (dual) basic feasible solution some special device is introduced enabling consideration of an allied problem whose solution provides a primal (dual) basic feasible solution for the original problem. The original Dantzig method [1] is a primal simplex method; the Lemke paper [5] describes a dual simplex method.

The proofs for termination of a simplex method in a finite number of pivot steps use the fact that any tableau is uniquely determined by its associated nonbasic variables (of primal or of dual programs) and that there exist at most $\binom{n+m}{m} = \binom{n+m}{n}$ possible sets of nonbasic variables. Then any pivot steps assuring that no tableau is ever repeated guarantees finiteness. The finiteness proof for a primal (dual) simplex method in which no "degeneracies" occur, i.e., in which $a'_{i0} < 0$, $i \neq 0$, ($a'_{0j} > 0$, $j \neq 0$) is clear, for each pivot step strictly decreases (increases) the value of a'_{00} and thereby assigns an order to the sequence of tableaus. If, however, degeneracy occurs, some form of lexicographic order must be introduced to avoid the possibility of cycling.

2. A MUTUAL PRIMAL-DUAL SIMPLEX METHOD

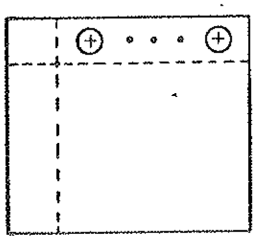
We describe here a simplex method for directly solving any pair of dual linear programs (1.1) or (1.2). The method specifies pivot choices for any tableau whether feasible or not, and degenerate or not, which lead to a tableau exhibiting a primal feasible solution (or, primal infeasibility) and then to a tableau exhibiting optimal solutions to both programs (or, primal objective unboundedness and dual infeasibility). This is accomplished by using a primal simplex pivot choice rule until primal degeneracies occur; then a dual simplex pivot choice rule is used on a subtableau corresponding to primal zero valued basic variables until the degeneracies are resolved. If "sub-dual degeneracies" are or come to be present in the subtableau, a primal simplex pivot choice rule is used on a sub-subtableau until these degeneracies are resolved, and so forth. The hierarchy of tableau, subtableau, sub-subtableau, etc., is used to establish a hierarchy of goals which are associated with every tableau. Every pivot step leads to a strict "improvement" in one of the goals, with goals higher in the hierarchy remaining unaffected. This serves to order the sequence of tableaus, thus assuring termination of the method in a finite number of pivot steps.

Every tableau (1.5) in the sequence of tableaus obtained by successive pivot steps has associated with it a hierarchy of numbered subtableaus each with a distinguished entry (and hence row and column). Odd numbered subtableaus k have the form



("primal or row feasible form") (2.1)

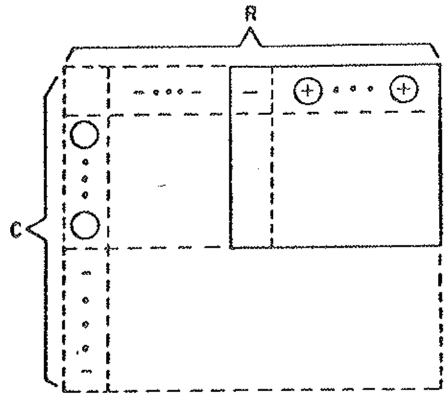
with $\alpha(k) \geqq 1$, the number of rows, and $-\beta(k)$, the value of the distinguished entry. Even numbered subtableaus k have the form



("dual or column feasible form") (2.2)

with $\alpha(k) \geqq 1$, the number of columns, and $\beta(k)$, the value of the distinguished entry.

The hierarchy of subtableaus associated with a tableau is initiated as follows. Subtableau 1 consists of all columns and all rows of (1.1) with $a_{i0} \leqq 0$, with distinguished entry some $a_{h0} > 0$ (if all $a_{i0} \leqq 0$ and the entire tableau is in primal feasible form, the entire tableau is taken as "subtableau 1"). Given any tableau suppose a subtableau k in primal (dual) feasible form has been defined with distinguished row R and column C. If C (if R) contains zeros and R is not all nonnegative (C is not all nonpositive), a subtableau k + 1 in dual (primal) feasible form is defined as consisting of rows (columns) corresponding to the zeros of C (of R), columns (rows) corresponding to the nonnegative entries of R (nonpositive entries of C), with distinguished entry some negative entry of R (some positive entry of C). Schematically,



where the whole diagram represents a primal feasible form subtableau k, and the subdiagram enclosed in solid lines a dual feasible form subtableau k + 1.

Associate with any tableau and its subtableaus a hierarchy of goals with goal k (k = 1, 2, ...) being to pivot to obtain a new tableau whose new subtableau k (if it exists) has $\alpha(k)$ larger, or, has $\alpha(k)$ unchanged but $\beta(k)$ larger; while $\alpha(i)$, $\beta(i)$ for $i < k$ remain unchanged.

Suppose, now, that we have reached the p^{th} tableau with entries a^p_{ij}, along with its well-defined hierarchy of subtableaus and their associated values $(\alpha_p(k), \beta_p(k))$. We describe the choice of pivot entry to obtain the $(p+1)^{-th}$ tableau and the hierarchy of subtableaus associated with the $(p+1)^{-th}$ tableau.

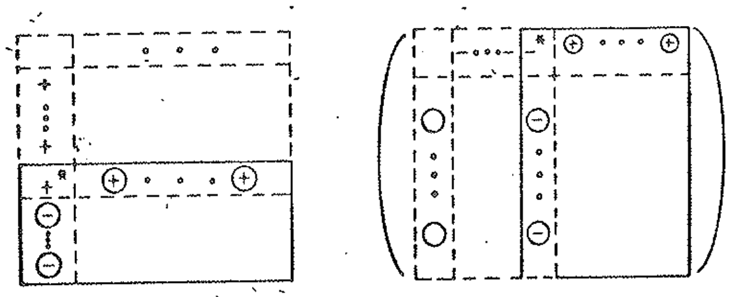
Rules

Suppose the subtableau with highest index K is in primal (dual) feasible form.

(a) Apply the primal (dual) simplex pivot choice rule (see above) to the subtableau K. Maintain same hierarchy of subtableaus, except K. Redefine K and any subsequent ones if possible.

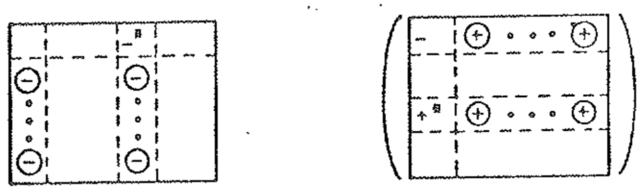
If rule (a) is not applicable then one of the two possibilities (b) or (c) must hold.

(b)



Choose as pivot entry the distinguished entry. Maintain same hierarchy of subtableaus, except K − 1 and K. Redefine K − 1 and any subsequent ones if possible.

(c)



Choose as pivot entry the negative (positive) entry in the distinguished row (column) whose column is nonpositive (whose row is nonnegative). Maintain same hierarchy of subtableaus except K. Redefine K and any subsequent ones if possible.

Finally, if ever the choice of pivot entry is an element of the first row (some a^p_{0j}) or of the first column (some a^p_{i0}), stop.

If the process is stopped because the choice of pivot entry is a^p_{00}, solutions to both programs obtain; if it is stopped because the choice of pivot entry is some a^p_{i0}, the primal or row program has no feasible solution; if it is stopped because the choice of pivot entry is some a^p_{0j}, the primal program is in feasible form but the dual or column program has no feasible solution.

If the pivot entry is chosen according to (a) either there is no K^{th} subtableau or $\alpha_{p+1}(K) \geqq \alpha_p(K)$ and if $\alpha_{p+1}(K) = \alpha_p(K)$ then $\beta_{p+1}(K) > \beta_p(K)$ (due to the absence of "K^{th} degeneracies"); while $\alpha_{p+1}(i) = \alpha_p(i)$ and $\beta_{p+1}(i) = \beta_p(i)$ for $i < K$ since the pivot entry has zeros in rows and columns that could have an effect on these values. If the pivot is chosen according to (b) either there is no $(K-1)^{th}$ subtableau or $\alpha_{p+1}(K-1) > \alpha_p(K-1)$ [see subtableau in (b)]; while again, and for the same reason, $\alpha_{p+1}(i) = \alpha_p(i)$ and $\beta_{p+1}(i) = \beta_p(i)$ for $i < K-1$. Finally, if the pivot is chosen according to (c), either there is no K^{th} subtableau or $\alpha_{p+1}(K) > \alpha_p(K)$ [see subtableau in (c)], while again $\alpha_{p+1}(i) = \alpha_p(i)$ and $\beta_{p+1}(i) = \beta_p(i)$ for $i < K$. Therefore, this choice of pivot entry and assignment of subtableaus always leads to strict improvement in some goal, thereby ordering the sequence of tableaus obtained in successive pivot steps. As noted above, this suffices to assure termination of the method in a finite number of pivot steps.

A. W. Tucker has pointed out that the inductive counterpart of this construction leads to a particularly simple and appealing proof of termination. The induction is made on the number $m+n$ of primal (or dual) variables.

3. SOME REMARKS

It is perhaps of interest to review the primal-dual algorithm of Dantzig, Ford, and Fulkerson [2] to enable comparison with the algorithm proposed here. By our definition the primal-dual algorithm is a simplex method applied not directly to the problem as stated but to an "extended" problem and with rather special pivot choice rules.

The problem to be solved and its dual as posed in [2] is

<u>Primal (Row) Program</u>

Minimize

$$y_0 = a_{01}y_{m+1} + \cdots + a_{0n}y_{m+n}$$

constrained by

$$a_{10} = a_{11}y_{m+1} + \cdots + a_{1n}y_{m+n} \geqq 0$$

$$\vdots$$

$$a_{m0} = a_{m1}y_{m+1} + \cdots + a_{mn}y_{m+n} \geqq 0$$

$$y_{m+1} \geqq 0, \ldots, y_{m+n} \geqq 0$$

Maximize

$$x_0 = x_1a_{10} + \cdots + x_ma_{m0}$$

constrained by

$$x_{m+1} = -a_{01} + x_1a_{11} + \cdots + x_ma_{ml}$$

$$\vdots$$

$$x_{m+n} = -a_{0n} + x_1a_{1n} + \cdots + x_ma_{mn}$$

$$x_{m+1} \geqq 0, \ldots, x_{m+n} \geqq 0$$

(3.1)

where it is assumed that the dual program has a feasible solution (if not, the extra variable $y_{m+n+1} \geqq 0$ and constraint $y_{m+1} + \cdots + y_{m+n} + y_{m+n+1} = a_{m+1,0}$ with $a_{m+1,0}$ arbitrarily large can be added to the primal program, thus assuring an easily found initial feasible solution to the new dual problem). An "extended" problem and its dual is then defined which can be exhibited in the tableau

Extended Primal Program

	1	y_{m+1}	$\circ\ \circ\ \circ$	y_{m+n}	
1	b_{00}	$-b_{01}$	$\circ\ \circ\ \circ$	$-b_{0n}$	$= z_0$
σ_1	$-a_{10}$	a_{11}	$\circ\ \circ\ \circ$	a_{1n}	$= -y_1$
$\vdots$	$\vdots$			$\vdots$	$\vdots$
σ_m	$-a_{m0}$	a_{m1}	$\circ\ \circ\ \circ$	a_{mn}	$= -y_m$
	$= \sigma_0$	$= \sigma_{m+1}$	$\circ\ \circ\ \circ$	$= \sigma_{m+n}$	

where $b_{0j} = \sum_{i=1}^{m} a_{ij}$ (3.2)

Dual to
Extended Primal Program

Here $z_0 = y_1 + \cdots + y_m$ is to be minimized subject to the row equations and $y_1 \geqq 0, \ldots, y_{m+n} \geqq 0$, and σ_0 is to be maximized subject to the column equations and $\sigma_1 \geqq 0, \ldots, \sigma_{m+n} \geqq 0$. Since $a_{i0} \geqq 0$, (3.2) is in primal feasible form. Notice that a feasible solution exists to the primal program (3.1) only if min $z_0 = 0$.

The primal-dual algorithm can be described as consisting of a finite sequence of tableaus, starting with (3.2), obtained by successive pivot steps. With every tableau is associated a (not necessarily basic) feasible solution $\{x_0, x_1, \ldots, x_{m+n}\}$ to the dual or column program (3.1). Then, given any tableau and its associated $\{x_0, x_1, \ldots, x_{m+n}\}$ a primal pivot choice rule is used on a subtableau consisting of all columns except for those corresponding to y_{m+j} for which $x_{m+j} > 0$ $(j = 1, \ldots, n)$. If a primal pivot choice cannot be made the subtableau can only be in ("optimal") form (1.6) and one of three cases for the complete tableau must hold:

(a) The distinguished entry has value 0
(b) The distinguished entry has positive value and the distinguished row some negative entry
(c) The distinguished entry has positive value and the distinguished row all nonnegative entries.

If (a) occurs the values exhibited for $y_{m+1}, \ldots, y_{m+n}$ in the tableau and the associated values for $\{x_0, x_1, \ldots, x_{m+n}\}$ constitute optimal solutions to the dual programs (3.1). This is easily established since these values are feasible and they make $x_0 = y_0$. If (b) occurs then a new feasible solution to the dual program (3.1) with values $\{\bar{x}_0, \bar{x}_1, \ldots, \bar{x}_{m+n}\}$ is associated with the whole tableau, with $\bar{x}_0 > x_0$. Namely,

$$\bar{x}_k = x_k + \Theta\sigma_k, \qquad \Theta = \min_{\substack{\sigma_{m+j} < 0 \\ (j = 1, \ldots, n)}} -\frac{x_{m+j}}{\sigma_{m+j}} \quad (> 0) \tag{3.3}$$

where σ_k is the value exhibited by the tableau of the variable σ_k. (This step can easily be described as a "partial pivot step" in which the values of the x_k are altered by the values of the σ_k). If (c) occurs then the minimum value of z_0 is attained but is positive, implying no feasible solution to the primal problem (3.1) exists.

In "geometric language" the primal-dual algorithm defines a sequence of successive neighboring vertices or extreme points of the convex polyhedron defined by the constraints of the extended primal program (3.2). It also defines a sequence of feasible points in the dual program (3.1) space which are not, in general, extreme. In fact the straight line joining two successive such dual feasible points [defined by (3.3)] usually lies in the interior of the dual convex polyhedral region (3.1), while the points themselves (except possibly the first) lie on some face of the polyhedron. In contrast, the mutual primal-dual simplex method defines a sequence of successive neighboring points (vertices after feasibility is achieved) of the convex polyhedron defined by the constraints of the original primal problem and visits only extreme points. Although there is no logical basis for comparison, intuition would seem to indicate that the computational advantage resides with "sticking to extreme points of the original problem."

Of course, the primary interest of these methods is in their application to highly "degenerate" problems, for example the assignment and transportation problems. The primal-dual algorithm applied to an assignment or transportation problem is the Hungarian method [4] (though it must be said that it was the ideas of the Hungarian method which led to the development of the primal-dual algorithm). Contrary to widely held beliefs, the Hungarian method (as described in [6]) can be described as a simplex method in much the same way as the primal-dual algorithm has been above. In fact, every operation as given in [6] has its simplex method counterpart. It is hoped (and expected) that the application of the idea of the mutual primal-dual simplex method to the assignment and transportation problems will lead to a new computational method which may better the efficiency of the Hungarian method, for in these problems the geometric considerations alluded to above appear to be important. Finally, the application of these ideas to the network flow algorithms, and particularly the "out of kilter" method of Fulkerson [3], should lead to further insight concerning the relationship between these specialized algorithms and simplex methods.

REFERENCES

1. Dantzig, G. B., "Maximization of a Linear Function of Variables Subject to Linear Inequalities," Chapter XXI of Activity Analysis of Production and Allocation (edited by T. C. Koopmans), Cowles Commission Monograph No. 13, Wiley, New York, 1951.
2. Dantzig, G. B., L. R. Ford, and D. R. Fulkerson, "A Primal-Dual Algorithm for Linear Programs," Paper 7 of Linear Inequalities and Related Systems (edited by H. W. Kuhn and A. W. Tucker), Annals of Mathematics Study No. 38, Princeton University Press, Princeton, 1956.
3. Fulkerson, D. R., "An Out of Kilter Method for Minimal Cost Flow Problems," RAND Corporation, P-1825, April 5, 1960.
4. Kuhn, H. W., "The Hungarian Method for the Assignment Problem," Naval Research Logistics Quarterly, Vol. 2, 1955, pp. 83-97.
5. Lemke, C. E., "The Dual Method of Solving the Linear Programming Problem," Naval Research Logistics Quarterly, Vol. 1, 1954, pp. 48-54.
6. Munkres, J., "Algorithms for the Assignment and Transportation Problems," Journal of the Society for Industrial and Applied Mathematics, Vol. 5, 1957, pp. 32-38.