RESEARCH ANNOUNCEMENTS

The purpose of this department is to provide early announcement of significant new results, with some indications of proof. Although ordinarily a research announcement should be a brief summary of a paper to be published in full elsewhere, papers giving complete proofs of results of exceptional interest are also solicited.

OUTLINE OF AN ALGORITHM FOR INTEGER SOLUTIONS TO LINEAR PROGRAMS

BY RALPH E. GOMORY[1]

Communicated by A. W. Tucker, May 3, 1958

The problem of obtaining the best integer solution to a linear program comes up in several contexts. The connection with combinatorial problems is given by Dantzig in [1], the connection with problems involving economies of scale is given by Markowitz and Manne [3] in a paper which also contains an interesting example of the effect of discrete variables on a scheduling problem. Also Dreyfus [4] has discussed the role played by the requirement of discreteness of variables in limiting the range of problems amenable to linear programming techniques.

It is the purpose of this note to outline a finite algorithm for obtaining integer solutions to linear programs. The algorithm has been programmed successfully on an E101 computer and used to run off the integer solution to small (seven or less variables) linear programs completely automatically.

The algorithm closely resembles the procedures already used by Dantzig, Fulkerson and Johnson [2], and Markowitz and Manne [3] to obtain solutions to discrete variable programming problems. Their procedure is essentially this. Given the linear program, first maximize the objective function using the simplex method, then examine the solution. If the solution is not in integers, ingenuity is used to formulate a new constraint that can be shown to be satisfied by the still unknown integer solution but not by the noninteger solution already attained. This additional constraint is added to the original ones, the solution already attained becomes nonfeasible, and a new maximum satisfying the new constraint is sought. This process is repeated until an integer maximum is obtained, or until some argument shows that a nearby integer point is optimal. What has been needed to transform this procedure into an algorithm is a systematic method for generating

[1] This work has been supported in part by the Princeton-IBM Mathematics Research Project.

the new constraints. A proof that the method will give the integer solution in a finite number of steps is also important. This note will describe an automatic method of generating new constraints. The proof of the finiteness of the process will be given separately.

Let us suppose that the original inequalities of the linear program have been replaced by equalities in nonnegative variables, so that the problem is to find nonnegative integers, $w, x_1, \cdots, x_m, t_1, \cdots, t_n$, satisfying

$$
\begin{aligned}
w &= a_{0,0} + a_{0,1}(-t_1) \cdots a_{0,n}(-t_n), \\
x_1 &= a_{1,0} + a_{1,1}(-t_1) \cdots a_{1,n}(-t_n), \\
&\vdots \\
x_m &= a_{m,0} + a_{m,1}(-t_1) \cdots a_{m,n}(-t_n)
\end{aligned} \tag{1}
$$

such that w is maximal. Using the method of pivot choice given by the simplex (or dual simplex) method, successive pivots result in leading the above array into the standard simplex form,

$$
\begin{aligned}
w &= a'_{0,0} + a'_{0,1}(-t'_1) \cdots a'_{0,n}(-t'_n), \\
x'_1 &= a'_{1,0} + \quad \cdots \quad \cdots a'_{1,n}(-t'_n), \\
&\vdots \\
x'_m &= a'_{m,0} + \quad \cdots \quad \cdots a'_{m,n}(-t'_n)
\end{aligned} \tag{2}
$$

where the primed variables are a rearrangement of the original variables and the $a'_{0,j}$ and $a'_{i,0}$ are nonnegative. From this array the simplex solution $t'_j = 0$, $x'_i = a'_{i,0}$ is read out.

An additional constraint can now be formulated. The constraint which will be generated is not unique, but is one of a large class that can be produced by a more systematic version of the following procedure.

If the $a'_{i,0}$ are not all integers, select some i_0 with $a'_{i_0,0}$ noninteger, and introduce the new variable

$$
s_1 = -f'_{i_0,0} - \sum_{j=1}^{j=n} f'_{i_0,j}(-t'_j) \tag{3}
$$

where $f'_{i_0,j} = a'_{i_0,j} - n'_{i_0,j}$, with $n'_{i_0,j}$ the largest integer $\leqq a'_{i_0,j}$. This new equation is added to the Equations (2), obtaining a new set which will be referred to as (2*). A feasible solution to (2*) is a vector, $w', x'_1, \cdots, x'_m, t'_1, \cdots, t'_n, s_1$ of nonnegative components. The values of $x'_1, \cdots, x'_m, t'_1, \cdots, t'_n$ determine the s_1 value through (3), so there is a natural correspondence between a solution

$x_1', \cdots, x_m', t_1', \cdots, t_n'$ of (2) and the (not necessarily feasible) solution that these values determine for (2*). Clearly any feasible solution to (2*) determines a feasible solution to the equations (2) simply by dropping the s_1.

It should be noted that if $f_{i_0,0}$ is $\neq 0$, then there is at least one $f_{i_0,j} \neq 0$, with $j \neq 0$, otherwise the equation

$$x_{i_0}' = a_{i_0,0}' + \sum_{j=1}^{j=n} a_{i_0,j}'(-t_j')$$

can have no solution in integers, and the program has no integer solution.

Since the simplex solution to (2), $t_j' = 0$, $x_i' = a_{i,0}'$ determines, through Equation (3), a negative value, $-f_{i_0,0}$ for s_1, the corresponding solution to (2*) is not feasible, i.e. the new restraint cuts off the old maximum. However, any nonnegative *integer* solution to (2) does give rise to a nonnegative integer solution to the equations (2*).

To see this suppose $w'', x_1'', \cdots, x_m'', t_1'', \cdots, t_n''$ is any solution in nonnegative integers to (2). The s_1'' determined is

$$\begin{aligned} s_1'' &= -f_{i_0,0}' - \sum_{j=1}^{j=n} f_{i_0,j}'(-t_j'') \\ &= n_{i_0,0}' + \sum_{j=1}^{j=n} n_{i_0,j}'(-t_j'') - a_{i_0,0}' - \sum_{j=1}^{n} a_{i_0,j}'(-t_j'') \end{aligned}$$

which using (2) becomes $s_1'' = n_{i_0,0}' + \sum_{j=1}^{n} n_{i_0,j}'(-t_j'') - x_{i_0}''$. Since the $n_{i_0,j}'$, the t_j'' and the x_{i_0}'' are all integers, the s_1'' determined is also an integer. Furthermore, since the $f_{i_0,j}'$ and the t_j'' are all nonnegative, (3) shows that s_1'' is $\geqq -f_{i_0,0}' > -1$. Since s_1'' is an integer, this shows it must be nonnegative.

This reasoning establishes a one-one correspondence between nonnegative integer solutions $w'', x_1'', \cdots, x_m'', t_1'', \cdots, t_n''$ to (2) and the corresponding nonnegative integer solutions $w'', x_1'', \cdots, x_m'', t_1'', \cdots, t_n'', s_1''$ to (2*). Since the w value is the same for both solutions, the problem of maximizing w over nonnegative integer solutions to (2) can be replaced by the problem of maximizing w over the nonnegative integer solutions to (2*). The solution to the original problem is obtained by dropping the s_1.

The procedure now is to maximize w over the solutions to (2*). This is done using the dual simplex method because all the $a_{0,j}'$ and $a_{i,0}'$ are already nonnegative, and $-f_{i_0,0}$ is the only negative entry in the zero column of the equations (2*). This fact usually makes remaximization quite rapid. The process is then repeated if the new simplex maximum is noninteger.

Of course the Equations (2*) involve one more equation than the Equations (2), and an equation is added after each remaximization. However, the total number need never exceed $m+n+2$. For if an s-variable, added earlier in the computation reappears among the variables on the left hand side of the equations after some remaximization, the equation involving it can simply be dropped, as the only equations that need be satisfied are the original ones. This limits the total number of s-variables to $n+1$ or less.

It should be noted that even the process just described involves an element of choice, any of the rows i of (2) with $a_{i,0}$ noninteger might be chosen to generate the new relation. Some choices are better than others. A good rule of thumb based on the idea of "cutting" as deeply as possible with the new relation, and borne out by limited computational experience, is to choose the row with the largest fractional part $f_{i,0}$ in the zero column.

The class of possible additional constraints is not limited to those produced by the method described here since it is easily seen that some simple operations on and between rows preserve the properties needed in the additional relations. These operations can be used to produce systematically a family of additional relations from which a particularly effective cut or cuts can be selected. A discussion of this class of possible additional constraints together with a rule of choice of row which can be shown to bring the process to an end in a finite number of steps—thus providing a finite algorithm—require some space and will be given as part of a more complete treatment in another place.

References

1. George B. Dantzig, *Discrete-variable extremum problems*, J. Operations Res. Soc. Amer. vol. 5, no. 2 (1957).

2. G. Dantzig, R. Fulkerson, and S. Johnson, *Solution of a large-scale traveling-salesman problem*, J. Operations Res. Soc. Amer. vol. 2, no. 4 (1954).

3. Harry M. Markowitz and Alan S. Manne, *On the solution of discrete programming problems*, Econometrica vol. 25, no. 1 (1957).

4. Stuart E. Dreyfus, *A comparison of linear programming and dynamic programming*, Rand Report P-885, June, 1956.

Princeton University