

U. S. AIR FORCE  
PROJECT RAND  
RESEARCH MEMORANDUM

Notes on Linear Programming: Part XLIII

A FEASIBILITY ALGORITHM FOR ONE-WAY  
SUBSTITUTION IN PROCESS ANALYSIS

Kenneth J. Arrow  
Selmer M. Johnson

RM-1976

ASTIA Document Number AD 144278

12 September 1957

Assigned to \_\_\_\_\_

This is a working paper. It may be expanded, modified, or withdrawn at any time. The views, conclusions, and recommendations expressed herein do not necessarily reflect the official views or policies of the United States Air Force.

---

The RAND Corporation

1700 MAIN ST. • SANTA MONICA • CALIFORNIA



SUMMARY

Suppose that we have a certain number of machines, not all of the same capability, with which to do a certain number of tasks, not all of the same difficulty. On the assumption that it is better to use a more capable machine for a more difficult task, an algorithm is given for the most efficient assignment of machines to tasks. It is shown that the algorithm solves an equivalent linear programming problem.



CONTENTS

SUMMARY .....	11
Section	
1. INTRODUCTION .....	1
2. THE ALGORITHM .....	2
3. MATHEMATICAL FORMULATION .....	3
4. THEOREM AND PROOF .....	4
REFERENCES .....	8
LIST OF RAND NOTES ON LINEAR PROGRAMMING .....	9



A FEASIBILITY ALGORITHM FOR ONE-WAY SUBSTITUTION IN  
PROCESS ANALYSIS\*

1. INTRODUCTION

We suppose, as is done in some simple forms of process analysis (see Refs. 1 and 2), that we have a certain number of machines of  $m$  different types with which to do a certain number of tasks of  $n$  different types. Each machine is capable of performing tasks of one or more types, and conversely each task can be done by machines of one or more types. We further assume that the machines can be numbered in decreasing order of capability, so that any machine can perform any task that can be performed by a later machine; similarly, it is assumed that the tasks can be numbered in decreasing order of difficulty, so that, if a given task can be performed by a given machine, all later tasks can be performed by that machine. Finally, we assume that, in a certain sense, it is more efficient to do a task by a machine that is as close as possible to being the first machine in the series. That is, if task 1 is more difficult than task 2, then it is better to use a more capable machine to do task 1 (if the machine is capable of performing task 1) and use a less capable machine to do task 2.

Suppose we are given a certain amount of work of each type to do and a certain quantity of machines of each type.

---

\* We are indebted to Harry Markowitz for cooperation in conjecturing the algorithm discussed here.

We wish to test whether or not the given task program is feasible.

## 2. THE ALGORITHM

The following algorithm is proposed: First, use the most capable machine for the most difficult task. If the whole desired quantity of the task can be done by the available quantity of this machine, apply the left-over amount of the most capable machine to the next most difficult task. Continue until the stock of machines of the most capable type is exhausted and then continue with machines of the second most capable type. This process is continued either until all the machine stocks capable of performing a certain task have been exhausted and some of this task remains, in which case the task program is infeasible, or until all the tasks have been assigned without using more of any machine than is available, in which case the program is feasible.

The second statement is obvious; the difficult point is to prove the criterion for infeasibility, since it might seem at first glance that other assignments of machines to tasks than the one considered might show a program to be feasible when the method of assigning the most difficult tasks to the most capable machines would not indicate this. In fact — as we shall see — under the assumptions made, this possibility cannot arise. Moreover, the algorithm will be shown to solve an equivalent linear programming problem.



### 3. MATHEMATICAL FORMULATION

Let  $a_{ij}$  be the amount of task  $j$  performed by a unit input of machine  $i$ . The ordering of machines and tasks by capability and difficulty, respectively, can be expressed as follows:

$$(1) \quad \text{if } a_{ij} > 0 \text{ and } i' < i, \text{ then } a_{i'j} > 0;$$

$$(2) \quad \text{if } a_{ij} > 0 \text{ and } j' > j, \text{ then } a_{ij'} > 0.$$

We are further assuming that if  $i < i'$  and  $j < j'$ , and if machine  $i'$  is capable of handling task  $j$  at all, then it is better to devote machine  $i$  to task  $j$  and use the amount of released stock of machine  $i'$  for task  $j'$ . That is, the output of task  $j'$  for a unit input of machine  $i$  is less than the output achievable by devoting the unit of machine  $i$  to task  $j$  and applying the amount of machine  $i'$  that would have been needed to perform the same amount of task  $j$  to the performance of task  $j'$ . By definition, it requires  $a_{ij}/a_{i'j}$  units of machine  $i'$  to perform the same amount of task  $j$  as one unit of machine  $i$ ; hence, the assumption can be expressed as follows:

$$(3) \quad \text{if } i < i', j < j', \text{ and } a_{i'j} > 0, \text{ then}$$

$$a_{ij'} < (a_{ij}/a_{i'j})a_{i'j'}; \text{ that is, } (a_{ij'}/a_{ij}) < (a_{i'j'}/a_{i'j}),$$

since by (1) the inequality  $a_{i'j} > 0$  implies  $a_{ij} > 0$ .

Let

$M_i$  = the number of units of the  $i$ -th type machine  
( $i = 1, 2, \dots, m$ ),

$T_j$  = the number of units of the  $j$ -th type task to be  
performed ( $j = 1, 2, \dots, n$ ),

$x_{ij}$  = the number of units of the  $i$ -th type machine  
assigned to the  $j$ -th type task,

$\bar{X}_r = (\bar{x}_{ij})_r$  = the set of  $x_{ij}$ 's defined according to  
the algorithm described in the introduction when  
applied to the first  $r$  tasks.

Thus

$$\bar{x}_{11} = \min \left( M_1, \frac{T_1}{a_{11}} \right).$$

If  $\bar{x}_{11} = M_1$ , then  $\bar{x}_{1j} = 0$  for  $j > 1$ , and

$$\bar{x}_{21} = \min \left( M_2, T_1 - \frac{a_{11}M_1}{a_{21}} \right).$$

If  $\bar{x}_{11} = \frac{T_1}{a_{11}}$ , then  $\bar{x}_{ij} = 0$  for  $i > 1$ , and

$$\bar{x}_{12} = \min \left( M_1 - \frac{T_1}{a_{11}}, \frac{T_2}{a_{12}} \right),$$

and so on.

#### 4. THEOREM AND PROOF

We shall establish the following result.

Theorem. If any allocation procedure satisfies a given program of machines and tasks, then the algorithm does also,

and in addition it maximizes the amount of work done on the last task type while meeting the other task requirements.

We present a proof by induction.

Let  $F_r$  be the set of  $x_{ij}$ 's satisfying the conditions

$$(4) \quad x_{ij} \geq 0,$$

$$(5) \quad \sum_{j=1}^n x_{ij} \leq M_i \quad (i = 1, \dots, m),$$

$$(6) \quad \sum_{i=1}^m a_{ij} x_{ij} \geq T_j \quad (j = 1, \dots, r).$$

We are seeking to prove that if  $F_n$  is nonnull, then it contains  $\bar{X}_n$ .

Let  $P_{r+1}$  be the linear programming problem of maximizing

$$(7) \quad \sum_{i=1}^m a_{i,r+1} x_{i,r+1}$$

among the elements of  $F_r$  — that is, maximizing the work done on the  $(r+1)$ -st type task while satisfying the requirements on the tasks of the first  $r$  types. We want to prove the following two statements:

$$(8) \quad \text{if } \bar{X}_r \text{ belongs to } F_r, \text{ then } \bar{X}_{r+1} \text{ solves } P_{r+1};$$

$$(9) \quad \text{if } \bar{X}_{r+1} \text{ solves } P_{r+1}, \text{ then it belongs to } F_{r+1}.$$

The two statements together imply that  $\bar{X}_{r+1}$  belongs to  $F_{r+1}$ .

if  $\bar{X}_r$  belongs to  $F_r$ . Now  $\bar{X}_0$  clearly belongs to  $F_0$ ; hence induction establishes the result that  $\bar{X}_r$  belongs to  $F_r$  for all  $r \leq n$ , and therefore  $\bar{X}_n$  belongs to  $F_n$ .

The proof of (9) is obvious. Since  $F_n$  is nonnull, there is a set of  $x_{ij}$ 's belonging to  $F_r$  for which (7) takes on a value at least as great as  $T_{r+1}$  (since an element of  $F_n$  belongs to  $F_r$  for all  $r$ ). From (8), then, (7) must take on at least as great a value when  $\bar{X}_{r+1}$  is substituted for the  $x_{ij}$ 's. Hence  $\bar{X}_{r+1}$  belongs to  $F_{r+1}$ .

To prove (8), we assume that  $\bar{X}_r$  is in  $F_r$ . Construct  $\bar{X}_{r+1}$  from  $\bar{X}_r$  by diverting any possible excess machine time from the  $r$ -th type task to the  $(r+1)$ -st type task according to the algorithm. It is easy to see that the set  $\bar{X}_{r+1}$  forms a basis for the linear programming problem  $P_{r+1}$ . By construction there are  $m+r$  basic variables  $> 0$  that form a "path" of steps to the right and downward through the  $(m$  by  $r+1)$  matrix of  $(x_{ij})$  in  $P_{r+1}$  starting at  $\bar{x}_{11}$  and ending at  $\bar{x}_{m,r+1}$ . (If an  $M_i$  and a  $T_j$  are exhausted simultaneously, we choose  $\bar{x}_{i+1,j} = 0$  for the basis.)

Let  $p_i$  and  $q_j$  be the dual variables associated with the inequalities (5) and (6), respectively. Then the duality conditions require that

$$(10) \quad a_{ij}q_j - p_i \leq 0 \quad (i = 1, \dots, m; j = 1, \dots, r),$$

$$(11) \quad a_{i,r+1} - p_i \leq 0 \quad (i = 1, \dots, m).$$

Inequalities (10) and (11) can be combined in the form

$$(12) \quad a_{ij}q_j - p_i \leq 0 \quad (i=1, \dots, m; j=1, \dots, r+1), \quad q_{r+1} = 1.$$

We define the  $q_j$ 's and  $p_i$ 's by requiring equality for the basic variables in (12); i.e., we require

$$(13) \quad a_{ij}q_j - p_i = 0 \quad \text{for } i, j \text{ corresponding to } \bar{x}_{ij} > 0.$$

We shall show that (12) holds also for every nonbasic variable such as  $x_{i_0j_0}$ , say; i.e., we shall show that

$$(14) \quad \frac{p_{i_0}/q_{j_0}}{a_{i_0,j_0}} \geq 1.$$

Now  $x_{i_0j_0}$  can be expressed in terms of a subset of the basic variables, say  $\bar{x}_{i_1j_0}, \bar{x}_{i_1j_1}, \bar{x}_{i_2j_1}, \bar{x}_{i_2j_2}, \dots, x_{i_0j_k}$ , where this set combined with  $x_{i_0j_0}$  forms the corners of a closed loop in the  $(x_{ij})$  matrix.

To simplify the notation, let

$$(u, v) = a_{i_u, j_v}.$$

Then from (13) we have

$$(u, u-1) = p_{i_u}/q_{j_{u-1}},$$

$$(u, u) = p_{i_u}/q_{j_u},$$

$$(0, k) = p_{i_0}q_{j_k}.$$

Therefore, we have

$$(u, u-1) / (u, u) = q_{j_u} / q_{j_{u-1}}.$$

Hence, the ratio in (14) can be written as

$$\begin{aligned} \frac{p_{1_0} / q_{1_0}}{(0, 0)} &= \frac{(0, k)}{(0, 0)} \prod_{u=1}^k \frac{(u, u-1)}{(u, u)} \\ &= \frac{(0, k)}{(0, 0)} \prod_{u=1}^k \frac{(u, u-1)}{(u, u)} \prod_{u=1}^k \frac{(u, 0)}{(u, 0)} \\ &= \left[ \frac{(0, k)}{(k, k)} \frac{(k, 0)}{(0, 0)} \right] \prod_{u=0}^{k-1} \left[ \frac{(u+1, u)(u, 0)}{(u, u)(u+1, 0)} \right] \geq 1, \end{aligned}$$

since by (3) the value of each heavy bracket group is  $\geq 1$ .

Thus, (12) is verified for every  $x_{ij}$ , and  $\bar{X}_{r+1}$  is an optimal linear programming solution to  $P_{r+1}$ . That is, (8) holds for each  $r$ , and the theorem is proved.

#### REFERENCES

1. Markowitz, H., Process Analysis of the Metal Working Industries, The RAND Corporation, Research Memorandum RM-1085, May 12, 1953.
2. \_\_\_\_\_, "Process Analysis of the U. S. Technology," Econometrica, Vol. 22 (1954), p. 515 (abstract).

LIST OF RAND NOTES ON LINEAR PROGRAMMING

- RM-1264 - Part I: The Generalized Simplex Method for Minimizing a Linear Form under Linear Inequality Restraints, by G. B. Dantzig, A. Orden, and P. Wolfe, April 5, 1954. Published in the Pacific Journal of Mathematics, Vol. 5, No. 2, June, 1955, pp. 183-195.
- RM-1265 - Part II: Duality Theorems, by G. B. Dantzig and A. Orden, October 30, 1953.
- RM-1266 - Part III: Computational Algorithm of the Simplex Method, by G. B. Dantzig, October 26, 1953.
- RM-1267-1 - Part IV: Constructive Proof of the Min-Max Theorem, by G. B. Dantzig, September 8, 1954. Published in the Pacific Journal of Mathematics, Vol. 6, No. 1, Spring, 1956, pp. 25-33.
- RM-1268 - Part V: Alternate Algorithm for the Revised Simplex Method Using Product Form for the Inverse, by G. B. Dantzig and W. Orchard-Hays, November 19, 1953.
- RM-1440 - Part VI: The RAND Code for the Simplex Method (SX4) (For the IBM 701 Electronic Computer), by William Orchard-Hays, February 7, 1955.
- RM-1270 - Part VII: The Dual Simplex Algorithm, by G. B. Dantzig, July 3, 1954.
- RM-1367 - Parts VIII, IX, and X: Upper Bounds, Secondary Constraints, and Block Triangularity in Linear Programming, by G. B. Dantzig, October 4, 1954. Published in Econometrica, Vol. 23, No. 2, April, 1955, pp. 174-183.
- RM-1274 - Part XI: Composite Simplex-Dual Simplex Algorithm—I, by G. B. Dantzig, April 26, 1954.
- RM-1275 - Part XII: A Composite Simplex Algorithm—II, by William Orchard-Hays, May 7, 1954.
- RM-1281 - Part XIII: Optimal Solution of a Dynamic Leontief Model with Substitution, by G. B. Dantzig, June 15, 1954. Published in Econometrica, Vol. 23, No. 3, July, 1955, pp. 295-302.
- RM-1290 - Part XIV: A Computational Procedure for a Scheduling Problem of Edie, by G. B. Dantzig, July 1, 1954. Published in the Journal of the Operations Research

Society of America, Vol. 2, No. 3, August, 1954,  
pp. 339-341.

- RM-1328 - Part XV: Minimizing the Number of Carriers to Meet a Fixed Schedule, by G. B. Dantzig and D. R. Fulkerson, August 24, 1954. Published in Naval Research Logistics Quarterly, Vol. 1, No. 3, September, 1954, pp. 217-222.
- RM-1369 - Part XVI: The Problem of Routing Aircraft - A Mathematical Solution, by A. R. Ferguson and G. B. Dantzig, September 1, 1954. Published in Aeronautical Engineering Review, Vol. 14, No. 4, April, 1955, pp. 51-55.
- RM-1374 - Part XVII: Linear Programming under Uncertainty, by G. B. Dantzig, November 16, 1954. Published in Management Science, Vol. 1, Nos. 3-4, April-July, 1955, pp. 197-206.
- RM-1375 - Part XVIII: Status of Solution of Large-scale Linear Programming Problems, by G. B. Dantzig, November 30, 1954.
- RM-1383 - Part XIX: The Fixed Charge Problem, by W. M. Hirsch and G. B. Dantzig, December 1, 1954.
- RM-1400 - Part XX: Maximal Flow through a Network, by L. R. Ford and D. R. Fulkerson, November 19, 1954. Published in Canadian Journal of Mathematics, Vol. 8, No. 3, 1956, pp. 399-404.
- RM-1418-1 - Part XXI: On the Min Cut Max Flow Theorem of Networks, by G. B. Dantzig and D. R. Fulkerson, April 15, 1955. Published in Linear Inequalities and Related Systems, Annals of Mathematics Study No. 38, edited by H. W. Kuhn and A. W. Tucker, Princeton University Press, 1956, pp. 215-221.
- RM-1475 - Part XXII: Recent Advances in Linear Programming, by G. B. Dantzig, April 12, 1955. Published in Management Science, Vol. 2, No. 2, January, 1956, pp. 131-144.
- RM-1432 - Part XXIII: A Production Smoothing Problem, by S. M. Johnson and G. B. Dantzig, January 6, 1955. Published in Proceedings of the Second Symposium in Linear Programming (Washington, D. C., January 27-29, 1955), Vol. 1, U. S. Department of Commerce, Washington, D. C., 1956, pp. 151-176.



- RM-1470 - Part XXIV: The Modification of the Right-hand Side of a Linear Programming Problem, by H. M. Markowitz, April 20, 1955.
- RM-1452 - Part XXV: The Elimination Form of the Inverse and Its Application to Linear Programming, by H. M. Markowitz, April 8, 1955.
- RM-1489 - Part XXVI: Computation of Maximal Flows in Networks, by D. R. Fulkerson and G. B. Dantzig, April 1, 1955. Published in Naval Research Logistics Quarterly, Vol. 2, No. 4, December, 1955, pp. 277-283.
- RM-1553 - Part XXVII: Dilworth's Theorem on Partially Ordered Sets, by A. J. Hoffman and G. B. Dantzig, August 26, 1955. Published in Linear Inequalities and Related Systems, Annals of Mathematics Study No. 38, edited by H. W. Kuhn and A. W. Tucker, Princeton University Press, 1956, pp. 207-214.
- RM-1560 - Part XXVIII: A Simple Linear Programming Problem Explicitly Solvable in Integers, by O. A. Gross, September 30, 1955.
- RM-1604 - Part XXIX: A Simple Algorithm for Finding Maximal Network Flows and an Application to the Hitchcock Problem, by L. R. Ford and D. R. Fulkerson, December 29, 1955. Published in Canadian Journal of Mathematics, Vol. 9, 1957, pp. 210-218.
- RM-1644 - Part XXX: A Class of Discrete-type Minimization Problems, by O. A. Gross, February 24, 1956.
- RM-1709 - Part XXXI: A Primal-Dual Algorithm, by G. B. Dantzig, L. R. Ford, and D. R. Fulkerson, May 9, 1956. Published in Linear Inequalities and Related Systems, Annals of Mathematics Study No. 38, edited by H. W. Kuhn and A. W. Tucker, Princeton University Press, 1956, pp. 171-181.
- RM-1736 - Part XXXII: Solving the Transportation Problem, by L. R. Ford and D. R. Fulkerson, June 20, 1956. Published in Management Science, Vol. 3, No. 1, October, 1956, pp. 24-32.
- RM-1737 - Part XXXIII: A Theorem on Flows in Networks, by David Gale, June 22, 1956. Published in Pacific Journal of Mathematics, Vol. 7, No. 2, 1957, pp. 1073-1082.

- RM-1798 - Part XXXIV: A Primal-Dual Algorithm for the Capacitated Hitchcock Problem, by L. R. Ford and D. R. Fulkerson, September 25, 1956. Published in Naval Research Logistics Quarterly, Vol. 4, No. 1, March, 1957, pp. 47-54.
- RM-1832 - Part XXXV: Discrete-variable Extremum Problems, by G. B. Dantzig, December 6, 1956.
- RM-1833 - Part XXXVI: The Allocation of Aircraft to Routes - An Example of Linear Programming under Uncertain Demand, by G. B. Dantzig, December 7, 1956.
- RM-1799 - Part XXXVII: Concerning Multicommodity Networks, by J. T. Robacker, September 26, 1956.
- RM-1864 - Part XXXVIII: Note on B. Klein's "Direct Use of Extremal Principles in Solving Certain Problems Involving Inequalities," by G. B. Dantzig, January 29, 1957. Published in The Journal of the Operations Research Society of America, April, 1956.



