J. Op. Res. Japan, Vol. 2, No. 1

A METHOD OF DISCRETE PROGRAMMING PROBLEM

SHINJI KATAOKA
Hitotsubashi University, Tokyo, Japan

(*Received July 14 1958*)

This paper proposes a method for finding a solution to the discrete variable extremum. Given a number of linear constraints, we aggregate them with proper weights and obtain one linear constraint which appears in the so-called "Knapsack Problem". Discussing relations between a solution of the initial problem and that of the Knapsack Problem, we get a sufficient condition for optimality and test if a given solution is optimal. When it is not, a method for improving the non-optimal solution is considered.

INTRODUCTION

Although a considerable progress has recently been made in the techniques of linear programming, it seems that there are only few theories on the discrete or integral variable programming problem. On the other hand, scheduling and allocation problems in metal or chemical industries sometimes require integral solutions.

In practice, fractional values obtained by linear programming are rounded off and regarded as the integral optimal solution. But it is not a satisfactory one.

Dantzig, Fulkerson and Johnson [1] solved the "travelling-salesman problem" by an ingenious method, and Dantzig [2] and Markowitz, Manne [3] presented somewhat more general procedures for computing the discrete problems. In any event, these two methods have following common features. We first obtain an optimal solution of a given problem by linear programming without the discrete condition. If it is not integral, we add to the old conditions a new linear one which rejects the non-integral optimal solution and should be satisfied by the integral optimal solution.

Above mentioned procedures are repeated until we attain our object. However, it is not yet proved that we are guaranteed to reach an integral optimal solution by their method.

In this paper, we wish to consider this problem from a slightly different point of view and to make a trial for further development.

The present object is therefore to construct a simple algorithm for the practical use, not to obtain a complete one.

In the algebraical expression, this is a problem of maximizing a linear function Z

$$Z=\sum_j v_j x_j \tag{Z}$$

subject to

$$\sum_{j=1}^{n} a_{ij} x_j = s_i \qquad (i=1,\ 2,\ \cdots,\ k) \tag{A}$$

$$x_j = 0 \quad \text{or} \quad 1 \tag{D}$$

Though the discrete condition (D) is not equivalent to " x_j=nonnegative integer" as Markowitz and Manne showed, any integral variable could be split into several zero-one variables. Therefore, we can consider (A) and (D) as general constraints for the discrete problem.

DEFINITIONS AND NOTATIONS

(Definition 1) Constraints.

Throughout this paper, there appear following four kinds of restrictions (A), (B), (C) and (D).

$$\sum a_{ij} x_j = s_i \qquad x_j \geqq 0 \tag{A}$$

$$(i=1,\ 2,\ \cdots,\ k, \qquad j=1,\ 2,\ \cdots,\ n)$$

Multiplying proper constant π_i with the i-th equation of (A) and aggregating into one constraint, we get,

$$\sum_i \pi_i \sum_j a_{ij} x_j = \sum \pi_i s_i$$

or

$$\sum q_j x_j = R \qquad x_j \geqq 0 \tag{B}$$

where $q_j=\sum_i \pi_i a_{ij}$, $R=\sum_i \pi_i s_i$.

$$0 \leqq x_j \leqq 1 \tag{C}$$

and discrete condition

$$x_j=0 \quad \text{or} \quad 1 \tag{D}$$

Sometimes we call (A) "initial constraints", (B) "aggregated constraints or Knapsack condition", (C) "continuous (zero-one) condition", (D) "discrete (zero-one) condition".

(Definition 2) Set of solution.

Let A denote a set of solution $X(x_1, x_2, \cdots, x_n)$ satisfying (A), and B, C and D are sets of X satifying (B), (C) and (D) respectively. A solution which belongs to D is called "zero-one solution".

(Definition 3) Value of Z.

When Y is a solution which belongs to a set P, then the value of the maximizing function Z is denoted as $Z(X \epsilon P)$ or simply $Z(P)$, and the maximum value, max $Z(P)$ to which the optimal solution is denoted as $\hat{X}$.

(Definition 4) Four problems.

As stated in Section 1, main purpose of this paper is to maximize

$$Z=\sum_j v_j x_j$$

subject to

$$\sum_{j=1}^{n} a_{ij}x_j=s_i\ ,$$

$$x_j=0 \quad \text{or} \quad 1.$$

Let this problem be denoted as "Problem (AD)", similary optimizing problems under (B) and (D), (A) and (C), (B) and (C) as "Problem (BD)", "Problem (AC)", and "Problem (BC) respectively. Especially, in later discussion, Problem (BD) proves to be the so-called "Knapsack Problem".

LEMMAS AND THEOREMS

(Lemma 1) Let P and Q be two sets of nonnegative solution X which satisfies some linear equalities (P) and (Q) respectively, and let $P \subset Q$ (all elements of P belong to Q). If $\hat{X}$ is an optimal solution of Problem (Q) and belongs to P, then $\hat{X}$ is also optimal in problem (P).
(Proof) Omitted.

Next, consider the above mentioned sets A, B, C and D.

$$A \subset B$$

meet D

$$AD \subset BD$$

and

$$D \subset C$$

meet B

$$BD \subset BC$$

then

$$AD \subset BD \subset BC \qquad (1)$$

From (1) and (Lemma 1) we get the following theorem,
(Theorem 1)
The sufficient condition for $\hat{X}$ to be optimal in Problem (AD) is that $\hat{X}$ is optimal in Problem (BC) and belongs to the set AD (that is $\hat{X}$ is a feasible solution of the linear constraints (AD)).

KNAPSACK PROBLEM

The maximizing problem under the constraint (B) and discrete condition (D) would be interpreted as the so-called "Knapsack Problem" which was treated by Dantzig and Bellman [2]. In this problem, a person is planning a hike and has decided not to carry more than R kilograms of different items of which j-th items is q_j kilograms weight and has value v_j. This is a problem how to select items in order to carry maximum total value in his sack. In mathematical term, maximizing

$$Z=\sum_j v_j x_j$$

under

$$\left.\begin{array}{l}\sum_j q_j x_j \leqq R \\ x_j=0 \quad \text{or} \quad 1\end{array}\right. \qquad (2)$$

Where $x_j=1$ means that the $j-$th item is selected, and $x_j=0$ means that it is not selected.

In order to obtain an optimal solution of this problem by linear programming, let only the equality (2) hold and all v's be positive not equal to zero. The reason why these assumptions are necessary will be explained in the later part of this section. Thus restriction (2) can be reduced to (B) and the above mentioned problem and Problem (BD) agree with each other.

Methods of getting the optimal solution of the Knapsack Problem was discussed by Dantzig and Bellman, the former solution is attained by successive use of linear-programming and the latter is a rigid one obtained by dynamic programming, but usually requires enormous labors for its success.

In practical cases, as intuition or experience could always search a considerably good solution, we are at first concerned with a criterion with which to decide whether a given feasible solution is optimal or not.

Replacing (C) instead of (D) in Problem (BD), get Problem (BC), which is a regular linear programming problem and can be easiy solved by a brief method.

(Theorem 2) Problem (BC)

to maximize

$$Z=\sum_j v_j x_j$$

subject to

$$\sum_j q_j x_j = R \ ,$$

$$0 \leqq x_j \leqq 1$$

has following optimal solution $X(x_1, x_2, \cdots, x_n)$ such as

$$x_1 = x_2 = \cdots = x_r = 1$$

$$x_{r+1} = \frac{R - \sum_{j=1}^{r} q_j}{q_{r+1}} \tag{3}$$

$$x_{r+2} = \cdots = x_n = 0$$

where

$$\max\left\{\frac{q_1}{v_1} \cdots \frac{q_r}{v_r}\right\} \leqq \min\left\{\frac{q_{r+1}}{v_{r+1}} \cdots \frac{q_n}{v_n}\right\} = \frac{q_{r+1}}{v_{r+1}}$$

and a number r is determined dy

$$\sum_{j=1}^{r} q_j \leqq R < \sum_{j=1}^{r+1} q_j$$

(Proof) Appendix 1

Theorem 2 can also be shown pictorially as stated in ref. (2).

Due to theorem 2 it proves that any item having zero v_j value can never be selected in the optimal solution of Problem (BC) *for $\frac{q_j}{v_j}$ becomes infinity.

Accordingly, even if a slack variable is introduced to (2), it never has positive value. This is the reason why we assumed (i) $v_j > 0$ and (ii) only equality does hold in (2).

Since

* It is to be noticed, however, that problem (BD) may have non-zero x_j of which $v_l = 0$ in an optimal solution.

$$BD \subset BC\ :$$

(Theorem 3) The sufficient condition for X to be optimal in Problem (BD) is that (i) X fulfils the constraints (B) and condition (D) and (ii) satisfies the optimality condition of Theorem 2.

DISCRETE PROGRAMMING PROBLEM

We now return to the initial discrete programming problem (AD).

Since (1) $$AD \subset BD \subset BC\ :$$

(Theorem 4) The sufficient conditon for $\hat{X}$ to be optimal in Problem (AD) is that

(i) $\hat{X}$ fulfils (A) and (D),

(ii) $\hat{X}$ is the optimal solution of Problem (BC) with properly selected constants.

As easily be seen, Theorem2–Theorem 4 only speak about the sufficient condition but not about necessary one, then even though a feasible solution does not satisfy the condition of Theorem 4, it is not always a non-optimal solution of Problem (AD). By choosing another set of π's or by adding other linear constraints proved to be the optimal solution, we may obtain another aggregated problem in which that feasible solution X can be optimal this time. Though the technique of selecting a suitable set of π's or constraints to be added is not yet described systematically, with respect to such a special problem, in which a combinatorial problem $a_{ij}=0$ or 1 for example, we can state it fairly well.

Before proceeding with the combinatorial problem, it is worthwhile to note that when a feasible zero-one solution X is given, a necessary condition for another feasible solution X' to be optimal in (AD) is $Z(X) < Z(X')$

A COMBINATORIAL PROBLEM

In order to put our methcd to test, we consider the following problem as an example.

Table I

$v_j \rightarrow$	4	4	5	2	4	9	7	9	10	8	7	11	13	13	14	9	5	
	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	x_{17}	S
I_1			1			1	1						1					1
I_2								1	1				1			1		1
I_3				1		1		1		1	1			1				1
I_4									1			1		1				1
I_5					1		1				1				1			1
I_6		1									1		1				1	1
I_7										1		1			1			1
I_8															1	1		1
I_9	1											1		1			1	1
$q_j^{(1)}$	1	1	1	1	1	2	2	2	2	2	3	3	3	3	3	2	2	
$q_j^{(1)}/v_j$	0.25	0.25	0.2	0.5	0.25	0.22	0.28	0.22	0.2	0.25	0.43	0.27	0.23	0.23	0.21	0.22	0.4	
	◎	◎				○			○						○			
			×					×					×	×		×		
$\Delta\pi_2$								0.5	0.5				0.5			0.5		
$q_j^{(2)}$								2.5	2.5				3.5			2.5		
$q_j^{(2)}/v_j$								0.28	0.25				0.27			0.28		
$\Delta\pi_1$			0.25			0.25	0.25						0.25					
$q_j^{(3)}$			1.25			2.25	2.25						3.75					
$q_j^{(2)}/v_j$			0.25			0.25	0.32						0.28					
													↟					

"We have items $I_1, I_2, \cdots, I_9$ and try to combine them into several combinations. When the value of each combination is given, for example (Table 1) combination $(I_1\ I_3)$ is 9 units valuable and $(I_2\ I_4)$ 10 units valuable and so on, what is the most valuable combining pattern of these nine items?"

The full set of possible combinations for this problem is shown in Table 1. Zeros are implied at all row-column intersections other than those with ones. If i-th item is involved in the j-th combination, then i, j element of this matrix a_{ij} is equal to one, and zero if not. The values $v's$ of these combinations are inserted into the second row of Table 1.

Now, let matrix of Table 1 except row v and column S be A, and column vector $(x_1\ x_2\ \cdots\ x_n)'$ * be X, then the concerned combinatorial problem can be described in mathematical term as follows.

Maximizing

$$Z=\sum_j v_j x_j$$

subject to

$$AX=S$$

$$\text{or} \quad \sum a_{ij}x_j=s_j$$

and

$$x_j=0 \quad \text{or} \quad 1$$

This is a typical discrete programming problem discussed before, and we shall deal with it by our aggregation method.

Computational Procedures

(Procedure 1) Let all nine $\pi's$ be equal to one and aggregate the nine linear constraints (A) into one equation (B_1)

$$\sum_j q_j^{(1)}x_j=R \qquad \cdots (B_1)$$

where

* A dash shows transpositon.

$$q_j^{(1)}=\sum_{i=1}^{9} a_{ij}$$

(Procedure 2) Compute $\epsilon_j^{(1)}=\frac{q_j^{(1)}}{v_j}$ for every combination.

(Procedure 3) By inspection (intuition or experience) choose a better solution $X^{(1)}$ which may be thought as a quasi-optimal solution. In Table I, selected combinations are 1-st, 2-nd 6-th, 9-th and 15-th, denoted by single circle, ○.

(Proc. 4) Find the maximum $\epsilon_j^{(1)}=\frac{q_j^{(1)}}{v_j}$ from among that of selected combinations. ($x_j=1$) In Table I

$$\max_{x_j=1} (\epsilon_j^{(1)})=\max (0.25,\ 0.25,\ 0.22,\ 0.2,\ 0.21)$$
$$=0.25=\epsilon_1^{(1)}=\epsilon_2^{(1)}$$

where the first and second combination shown by double circles, ◎, have maximum $\epsilon_j^{(1)}$ of the selected combinations.

(Proc. 5) Find combinations not selected in the feasible solution and has smaller $\epsilon_j^{(1)}$ value than the maximum found in Proc. 4. These are denoted by cross ×

(Proc. 6) Increase proper $\pi's$ from one until $\epsilon_j^{(1)}$ of single circle-combinations reach the maximum value, 0.25. Possible increments of $\pi's$ are denoted by $\Delta\pi's$ in Table 1.

First we consider the second constraint

$$x_8+x_9+x_{13}+x_{16}=1$$

multiplying π_2 before aggregation,

$$q_9^{(2)}=\pi_2+1$$

$$\frac{q_9^{(2)}}{v_9}=\frac{\pi_2+1}{10}=0.25$$

$$\therefore\quad \pi_2=1.5,\qquad \Delta\pi_2=0.5$$

where superscript (2) means that $q^{(2)}$'s are the coefficients of $x's$ in the second aggregation problem, $\pi_1=1$, $\pi_2=1.5$, $\pi_3=\cdots=\pi_9=1$. Since ϵ_j values except 8-th, 9-th, 13-th, 16-th do not change, only these four values are written in row $q_j^{(2)}$.

Similary in order to raise $q_6^{(2)}/v_6$ from 0.22 to 0.25, noticing the first restriction,

$$x_3+x_6+x_7+x_{13}=1$$

$$\frac{\Delta\pi_1+2}{9}=0.25$$

$$\therefore \quad \Delta\pi_1=0.25$$

then

$$q_3^{(3)}=q_3^{(2)}+0.25=q_3^{(1)}+0.25=1.25$$

$$q_6^{(3)}=q_3^{(2)}+0.25=2.25$$

$$q_7^{(3)}=2.25$$

$$q_{13}^{(3)}=q_{13}^{(2)}+0.25=3.5+0.25=3.75$$

At the third step 15-th combination is yet left with smaller value than 0.25, but because even if it is raised, the increasing effect does not affect the cross-combinations that have still smaller $\epsilon_j=\frac{q_j}{v_j}$ values than 0.25 such as 14-th combination, we need not increase it.

This procedure can also be simplified as follows:

put

$$a_{ij}'=\frac{a_{ij}}{v_j}$$

then

$$\epsilon_j=\sum_i \pi_i a_{ij}'$$

If in order to increase ϵ_j to $\epsilon_j+\Delta\epsilon_j$ we raise only i-th π, π_i to $\pi_i+\Delta\pi_i$, then

$$\Delta\pi_i a_{ij}' = \Delta\epsilon_j, \quad \Delta\pi_i = \frac{\Delta\epsilon}{a_{ij}'},$$

and $\Delta\epsilon_k$, the increment of ϵ_k other than ϵ_j, will be

$$\Delta\epsilon_k = \Delta\pi_i a_{ik}' = \frac{a_{ik}'}{a_{ij}'}\Delta\epsilon_j .$$

If $\epsilon_j^{(1)}$ is increased to ϵ_r by multiplying $\pi_i + \Delta\pi_i$ with i-th equation, then

$$\Delta\pi_i = \frac{\epsilon_r - \epsilon_j^{(1)}}{a_{ij}'}$$

and $\epsilon_k^{(1)}$ will be increased to $\epsilon_k^{(2)}$, $\frac{q_k^{(2)}}{v_k}$ in the next step.

$$\epsilon_k^{(2)} = \epsilon_k^{(1)} + \Delta\pi_i a_{ik}' = \epsilon_k^{(1)} + \frac{a_{ik}'}{a_{ij}'}(\epsilon_r - \epsilon_j^{(1)})$$

$$\therefore \quad \epsilon_r - \epsilon_k^{(2)} = (\epsilon_r - \epsilon_k^{(1)}) - \frac{a_{ik}'}{a_{ij}'}(\epsilon_r - \epsilon_j^{(1)})$$

Thus it turns out that $(\epsilon_r - \epsilon_k)$'s are transformed in the same way as the elimination method of the linear computation. Table II shows this step transformation.

(Proc. 7) At last, there may be left some cross-combinations which can not be equal to or larger than the maximum value by any means, such as 14-th.

Try if there exists such another feasible solution X' that has larger $Z(X')$ than $Z(X)$, and involves at least one left cross-combination.

(Theorem. 5) If there is not such a solution as in (Proc. 7), the initial feasible solution X is optimal in the given combinatorial problem.

(Proof) If the above condition is satisfied, we can be sure that left cross-combinations never come into the optimal solution and can be struck out from the initial set of combinations. On the other hand,

Table II $\left(a_{ij}'=\frac{a_{ij}}{v_j}\right)$

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	x_{17}
v_j	4	4	5	2	4	9	7	9	10	8	7	11	13	13	14	9	5
I_1			0.2			0.111	0.143						0.077				
I_2								0.111	0.1				0.077			0.111	
I_3				0.5		0.111		0.111		0.125	0.143			0.077			
I_4									0.1			0.091		0.077			
I_5					0.25		0.143				0.143				0.071		
I_6		0.25									0.143		0.077				0.2
I_7										0.125		0.091			0.071		
I_8															0.071	0.111	
I_9	0.25											0.091		0.077			0.2
$C_j^{(1)}$	0.25	0.25	0.2	0.5	0.25	0.222	0.286	0.222	0.2	0.25	0.429	0.273	0.231	0.231	0.213	0.222	0.4
	◎	◎				○			○						○		
			×					×					×	×		×	
$C_r-C_j^{(1)}$	0	0	0.05			0.03		0.03	0.05				0.02	0.02	0.04	0.03	
$C_r-C_j^{(2)}$	0	0	0.05			0.03		−0.03	0				−0.02	0.02	0.04	−0.03	
$C_r-C_j^{(3)}$	0	0	0			0		−0.07	0				−0.03	0.02	0.04	−0.03	
														↑↑			

through (Proc. 1)-(Proc. 6) by selecting suitable $\pi's$, the feasible solution X is proved to be optimal in the aggregated problem (BC) due to Theorem 3.

Accordingly we can be sure that X is also optimal in Problem (AD) from Theorem 4.

In Table I, there does not exist such a feasible solution X' that involves 14-th combination and $Z(X')>Z(X)$, therefore feasible solution ($x_1=x_2=x_6=x_9=x_{15}=1$ other $x_j=0$) is optimal in problem (AD).

(Proc. 8) If there exists such an X', repeat (Proc. 1)-(Proc. 7) replacing X' instead of X.

CONCLUSIONS

A method of computing the optimal solution of the discrete programming problem have been discussed. The present purpose was to give a criterion in the practical form, but not to obtain a complete theory. However it is of course desirable to be able to find out feasible solutions systematically by using, for example, the punch card machine. Then this aggregation method will help us in cutting off any unnecessary combinations. We may also regard this method as a kind of parametric linear programming.

APPENDIX I Proof to (Theorem 3)

maximize

$$Z=\sum_j v_j x_j \qquad \text{(a. 1)}$$

under

$$\sum_j q_j x_j = R, \qquad \text{(a. 2)}$$

$$x_j + y_j = 1 \qquad \text{(a. 3)}$$

Where y_j is nonnegative slack variable corresponding to x_j. Let $q's$ be arranged in certain order. Without loss of generality, we can suppose it as $q_1, q_2, \cdots q_n$.

Then a number r is defined by

$$\sum_{j=1}^{r} q_j \leqq R < \sum_{j=1}^{r+1} q_j \tag{a.4}$$

As (a. 2) and (a. 3) contain $n+1$ linear independent equations, we can solve next $n+1$ variables.

$$x_j = 1 - y_j \qquad j=1, 2, \cdots r. \tag{a.5}$$

$$x_{r+1} = \frac{R - \sum_{j=1}^{r} q_j}{q_{r+1}} - \sum_{j=r+2}^{n} \frac{q_j}{q_{r+1}} x_j + \sum_{j=1}^{r} \frac{q_j}{q_{r+1}} y_j \tag{a.6}$$

$$y_{r+1} = 1 - x_{r+1} \tag{a.7}$$

$$y_j = 1 - x_j \qquad j = r+2, \cdots, n \tag{a.8}$$

substituting (a. 5) (a. 6) into (a. 1),

$$Z = \sum_j v_j + \frac{v_{r+1}}{q_{r+1}} \left(R - \sum_{j=1}^{r} q_j \right)$$

$$+ \sum_{j=r+2}^{n} \left(v_j - \frac{v_{r+1}}{q_{r+1}} q_j \right) x_j + \sum_{j=1}^{r} \left(\frac{v_{r+1}}{q_{r+1}} q_j - v_j \right) y_j \tag{a.9}$$

For the sake of brevity, assuming $v_j > 0$, $q_j > 0$, and if

$$\max \left(\frac{q_1}{v_1}, \frac{q_2}{v_2}, \cdots, \frac{q_r}{v_r} \right) \leqq \min \left(\frac{q_{r+1}}{v_{r+1}}, \cdots, \frac{q_n}{v_n} \right) = \frac{q_{r-1}}{v_{r-1}} \tag{a.10}$$

then due to linearity of the function Z, the optimal solution can be obtained by putting values of the independent variables in (a. 5), (a. 6), (a. 7) and (a. 8) as zeros. Thus (Theorem 3) has been proved.

If we happen to have equality in (a. 4), and the condition (a. 10) are satisfied by certain order, then the zero-one solution $X(x_1 = \cdots = x_r = 1,\ x_{r+1} = \cdots = x_n = 0)$ would be optimal one to this Knapsack problem.

REFERENCES

1. G. B. DANTZIG, D. P. FULKERSON and S. M. JOHNSON; "Solution of Large Scale Traveling Salesman Problem", *Opns Res* 2 pp. 393~410 (1954)
2. G. B. DANTZIG; "Discrete-Variable Extremum Problems" Proc. of Symposium on Modern Techniques for Extremum Problems, Los Angeles (1955) pp. 266~277.
3. H. M. MARKOWITZ and A. S. MANNE; "On the Solution of Discrete Programming Problems" *Econometrica* 25 (1957) pp. 84~110.