

ON AN ITERATIVE METHOD FOR SOLVING INTEGER PROGRAMMING PROBLEMS

I. I. Pyatetskii-Shapiro, V. A. Volkonskii, L. V. Levina,

1966

SovietRxiv

View the original and related papers at <https://sovietrxiv.org/items/ru-196601.54715>

Source: Math-Net.Ru and CyberLeninka. Machine translation. Verify with the original.

Abstract

Full Text

UDC 519.95

**CYBERNETICS
AND CONTROL THEORY**

**I. I. Pyatetskii-Shapiro, V. A. Volkonskii, L. V. Levina,
A. Pomanskii**

**ON AN ITERATIVE METHOD FOR SOLVING
INTEGER PROGRAMMING PROBLEMS**

(Presented by Academician L. V. Kantorovich, December 7, 1965)

1°. As is well known, many practical problems of optimal control (in particular, of economic planning) reduce to problems of integer linear programming ⁽¹⁾. Existing methods of integer programming (Gomory's method and its modifications ⁽²⁾) make it possible to solve problems of relatively small size. For large-scale problems, the convergence rate of these methods, even when modern computers are used, proves insufficient. Therefore one has to introduce simplifying assumptions,* which often impoverish the solution itself. In this connection, it is of interest to study various heuristic devices that make it possible to obtain acceptable approximate solutions, in particular the modeling of real planning and control processes in the economy (see § 3°).

The method of solution set forth below is regarded by the authors only as the beginning of an investigation. It deliberately uses only the most general and simple devices. The method was suggested by ideas from the theory of games of automata ^(3,4).

By the proposed method, several variants of the problem of choosing optimal construction times and the distribution of capital investments in a branch producing equipment for oil extraction were computed. Mathematically, the problem reduces to an ordinary integer programming problem: it is required to maximize the linear functional

$$\sum_{j=1}^n c_j x_j \tag{1}$$

subject to the conditions

$$\sum_{j=1}^n a_{ij}x_j \leq b_i, \quad i = 1, \dots, m, \quad (2)$$

where the unknowns x_j ($j = 1, \dots, n$) take the values 0 or 1. All coefficients a_{ij}, c_j, b_i are nonnegative.

The solution is sought as follows.

2°. Fix b_0 and solve the system of $m + 1$ inequalities

$$\sum_{j=1}^n c_j x_j \geq b_0, \quad \sum_{j=1}^n a_{ij} x_j \leq b_i, \quad i = 1, \dots, m, \quad (3)$$

$$x_j = 0, 1 \quad (j = 1, \dots, n)$$

* For example, in the problem of choosing an optimal variant of capital investments, the only constraint usually adopted is the total volume of capital investments (in monetary form), without taking into account restrictions on available resources, the possibilities of performing particular kinds of work, the dynamics of these restrictions during the planning period, and the construction time.

by the iteration method that will be described below. Then, increasing b_0 , we obtain the solution of a new system, closer to the solution of the original problem. The process terminates when the system of inequalities ceases to be solved within a fixed number of iterations.

Let us proceed to the description of the iteration method. We choose the initial set x_j^0 arbitrarily. Suppose that at the k -th step the set $x_j^{(k)}$ ($j = 1, \dots, n$) has been obtained. We compute

$$\Delta_0 = \max \left(\left(b_0 - \sum_{j=1}^n c_j x_j^{(k)} \right) / b_0, 0 \right), \quad \Delta_i = \max \left(\left(\sum_{j=1}^n a_{ij} x_j^{(k)} - b_i \right) / b_i, 0 \right),$$

$$i = 1, \dots, m.$$

Then, using random choice, independently of one another we change the components of the vector x_j^k with one and the same probability p , equal to

$$p = \min \left(c, \max_{0 \leq i \leq m} \Delta_i \right).$$

In the computations carried out, $c = 1/2$ was adopted. In this way we obtain a new set $x^{(k+1)}$ ($j = 1, \dots, n$) and carry out the next iteration. When all Δ_i become 0, the solution has been found.

The described solution process may be represented as a random walk corresponding to a certain Markov chain. The states of this Markov chain are all possible sets (x_1, \dots, x_n) , $x_j = 0, 1$ ($j = 1, \dots, n$).

The states corresponding to solutions of system (3) are absorbing states. As is known, the probability of reaching, from any state of a finite irreducible Markov chain, an absorbing state in a finite number of steps is equal to one. This proves the theoretical convergence of the method used.

However, from the point of view of the laboriousness of the computational method, the principal interest is the number of steps (iterations) before reaching the solution region. The mathematical expectation of this quantity may naturally be regarded as an indicator of the efficiency of the algorithm. For example 3 (see item 4°), a table is given of the ratio of the number of steps under the method used to that under complete enumeration (equal to the average number of steps in a purely random walk: $p = 1/2$). A certain gain in laboriousness is apparently explained mainly by the “length of the iteration step” —the average number of components of the vector x changing during one iteration, which is set at different levels in different examples.

3°. The described algorithm was applied to the following problem. Suppose there are n objects of possible construction. The choice of a specific construction program is characterized by the vector $x = (x_1, \dots, x_n)$, where x_j takes the value 1 or 0 depending on whether the j -th object is included in the construction program in the given year. To each object there corresponds a vector of expenditures and output of various resources, or of one and the same resource in different years. Consider a matrix A of size $m \times n$, whose columns are these vectors. Then the system of inequalities can be written in the form

$$Ax \leq b, \quad (4)$$

where b is the vector limiting the expenditures of various resources and the requirements for the output produced. In addition, there is some formula for calculating the total loss from violating these constraints

$$\varphi = \varphi(\Delta_1, \Delta_2, \dots, \Delta_m).$$

One can imagine a more complex algorithm for solving such problems, using the idea of modeling the price-profit mechanism by analogy with iterative methods in ordinary convex programming [5]. For this purpose we introduce the vector $p = (p_1, \dots, p_n)$, where p_j is the probability of the event $x_j = 1$, and the vector of estimates of constraints (4) $v = (v_1, \dots, v_m)$.

The solution algorithm may consist of the following. Let p^t and v^t be the vectors p and v obtained at the t -th iteration. Using a random-selection mechanism, we find a realization of the vector x^t , assuming its components x_j^t to be independent and equal to one with probability p_j^t . Then, according to some rule, we correct the estimates v^t in accordance with the residuals $Ax^t - b$; for example, we increase the estimates corresponding to positive components of the residual vector and decrease the rest. We denote the resulting vector by v^{t+1} and use it to correct the probabilities p_j^t in accordance with the “profits” $(v^{t+1}A)_j$ and the total amount of loss φ^t . We obtain a new vector p^{t+1} . After this the next iteration begins. Experiments with such an algorithm have not yet been carried out.

Table 1

b	b_0	b_0	b_0	b_0	b_0	b_0
	$M - 5$	$M - 4$	$M - 3$	$M - 2$	$M - 1$	M
10	—	4	1, 4	3, 4	2, 2	—
9	1	3	—	13	19	51
8	2	12	32	260	8	55
7	6	13	194	33	7	—
6	—	—	7	—	—	—

Such heuristic methods can yield good results mainly for those practical problems whose specific features are taken into account by the modeled solution process. It is therefore not surprising if such an algorithm proves ineffective for a number of problems that are relatively easy to solve by other methods.

The results obtained in this work allow us to hope for sufficient effectiveness of computational methods that model economic processes of regulation.

4°. Examples of the following orders were computed:

	15	15	17	17	17	30	50
m	2	3	3	4	8	2	3
Number of examples	1	1	1	3	1	1	6

In three examples the solutions obtained can be compared with the optimal ones.

Example 1. $n = 17$, $m = 3$. The exact value of the maximum is 14.91.

b_0	13.88	14.76
Number of steps (iterations)	14	166

Example 2. $n = 30$, $m = 2$. The exact value of the maximum is 20.

b_0	15	16	17	18	19
Number of steps	3	253	1777	1335	2317

Example 3. To assess the effectiveness of the algorithm, an experiment was carried out with an example of the following simple structure ($N = 50$, $m = 3$):

$$\sum_{j=1}^{50} x_j \rightarrow \max, \quad \sum_{j=31}^{50} x_j \leq b, \quad \sum_{j=1}^{20} x_j \leq b, \quad \sum_{j=1}^{10} x_j + \sum_{j=41}^{50} x_j \leq b.$$

For this example it is not difficult to compute the total number of solutions of system (3) with fixed b_0 . As an indicator of the effectiveness of the method in comparison with complete enumeration one may use the quantity

$$R = 2^{50}/TN,$$

where N is the number of steps in which the solution of system (3) was obtained with the given value of b_0 ; T is the number of points in the solution region.

The values of R are given in Table 1 ($M = \max \sum_{l=1}^{50} x_j$).

The authors express their gratitude to A. D. Shapiro for participating in the construction of the examples and for discussing the results.

Central Economic-Mathematical Institute
Academy of Sciences of the USSR

Received
16 XI 1965

REFERENCES

1. D. B. Yudin, E. G. Golshtein, *Problems and Methods of Linear Programming*, Moscow, 1965.
2. R. E. Gomory, *Bull. Am. Math. Soc.*, 64, 5, 275 (1958).
3. M. L. Tsetlin, *UMN*, 18, 4 (1963).
4. I. M. Gelfand, M. L. Tsetlin, I. I. Pyatetskii-Shapiro, *DAN*, 152, No. 4, 845 (1963).
5. V. A. Volkonskii, *Economics and Mathematical Methods*, 1, issue 2, 1965.

Note: Figure translations are in progress. See original paper for figures.

Source: Math-Net.Ru and CyberLeninka. Machine translation. Verify with the original.