EFFECTIVE SOLUTION METHOD OF THE SUBSET SUM PROBLEM

Mahammad Aliyev

Cybernetics Institute of ANAS, Baku, Azerbaijan
mahammad.aliyev@yahoo.com

1. Introduction.

The knapsack problem can be formulated as a solution of the following linear integer programming formulation:

$$(KP) \qquad \text{maximize } \sum_{j=1}^{n} p_j x_j \tag{1}$$

$$\text{subject to } \sum_{j=1}^{n} w_j x_j \le c \tag{2}$$

$$x_j \in \{0;1\} \quad j = 1,\dots,n \tag{3}$$

when $p_j = w_j$ in (KP), the resulting optimization problem is known as the subset sum problem (SSP) because we are looking for a subset of the values w_i with the sum being as close as possible to, but not exceeding the given target value c.

$$(SSP) \qquad \text{maximize } \sum_{j=1}^{n} w_j x_i;$$

$$\text{subject to } \sum_{j=1}^{n} w_j x_i \le c$$

$$x_j \in \{0,1\}, \; j = 1,\dots,n$$

Although (SSP) is a special case of (KP) it is still NP-hard and is NP - complete [1].

The theory of NP - completeness gives us a framework for showing that it is very doubtful that a polynomial algorithm for solving e.g. the subset sum problem, then we would also be able to solve numerous famous optimization problems like the travelling salesman problem, general integer programming, and we would even be able to efficiently find mathematical proofs of theorems, as stated in Cook [1].

2. The construction of a sequence converging to the solution on a hyperarch.

Let's give the analytic expression of the problem on a hyperarch:

$$\left.\begin{array}{l} \sum_{j=1}^{n} p_j x_j \to \max \\ \sum_{j=1}^{n} w_j x_j \le c \\ \sum_{j=1}^{n} x_j = k \\ \sum_{j=1}^{n} (x_j - o_{nkj})^2 = r_{nk}^2 \\ x_j \in \{0,1\}, \; j = 1,\dots,n \end{array}\right\} \tag{4}$$

Assume that the point $X_1(n,k)$ is determined by the existence criterium and is located on the hyperarch K_{nk} [2]. We use this point and construct a sequence converging to the solution of problem (4). The terms of the sequence should satisfy the constraint and give strong monotonically increasing values to the functional.
1) Monotonity condition: take the first of the coordinates equal zero of the point $X_1(n,k)$ and compare it with the corresponding coordinates of the point $X_1(n,k)$ taking unit values p_i and p_j. $p_i - p_j > 0$.
Were $x_{1j} = 1;\ x_{1i} = 0$
2) Constraint satisfaction condition. Let's give constraint satisfaction condition for the indices i and j satisfying the monotonity condition. We calculate the difference $d = c - \sum_{j=1}^{n} w_j x_{1j}$; $d + w_j - w_i \geq 0$ is the constraint satisfaction condition.

$$\begin{cases} p_i - p_j > 0 \\ d + w_j - w_i \geq 0 \end{cases} \qquad (5)$$

If condition (5) is satisfied, we accept $x_{1j} = 0;\ x_{1i} = 1$. The point $X_2(n,k)$ found in such a way increases the functional and satisfies the constraint and is located on K_{nk}.

We continue this process until the relation (5) is not satisfied between the indices i and j.

We conduct the process $k = \overline{1, n-1}$ for all K_{nk} to $k = \overline{1, n-1}$ find the points $X^*_{q(n,k)}$ compare the values that they give to the functional and find the solution of problem (KP)

$$Z^* = \max_k \left(P, X_q^*(n,k)\right);\ k = \overline{1,n}; \qquad P(p_1, p_2, ..., p_n) \text{is object vector.}$$

The method given above gives more effective results for (SSP). Of exactly solved problems for (SSP) the taken examples is better than the results obtain for (KP) problems.

Example:

$W = (41,7,6,35,45,51,71,81,12,13,42,26,37,45,59,60,70,80,87,90,92,127,150,35,18,17,$
$60,64,63,77,66,27,21,83,70,77,140,141,38,39,5,23,29,35,47,49,57,68,77,83)$

c_i	1225	1500	2000	2500	2700
Exact solution	1225	1500	2000	2500	2700
Obtained result	1225	1500	2000	2500	2700

3. Investigation of difficulty order of the alqorithm.

To understand difficulty order we must consider the cycles contained in each other in the alqorithm:

$$k = \overline{1, n-1}; \quad j = \overline{1,n}; \quad i = \overline{1,n}; \quad kk = \overline{1,n};$$

The maximal number of embedded cycles is four.

Hence we proved polynomial property of the method and that its difficulty order are $O(n^4)$.

4. Results: In this work we obtained following results.

1. A method for reducing the solution of any linear integer programming problem to its solution on a hyperarch is given.
2. Absence of condition (5) between the coefficients p_i, p_j and w_j, w_i is necessary condition for the point to be an optimal solution on the K_{nk}.

3. The solution algorithm is polynomial.
4. The algorithm for (SSP) practically obtained only exact solution.

References

1. Hans Kellerer, Ulrich Pferschy, David Pisinger Knapsack Problems. Spinger-Verlag Berlin. Hidelberg 2004 525 p.
2. Aliyev M.M. One approach to the solution of the knapsack problem//Reports NAS of Azerb., 2006, №3, p.32-39.