

# Notes on the integer knapsack problem

Hiroshi Iida

e-mail: [oggi@res.otaru-uc.ac.jp](mailto:oggi@res.otaru-uc.ac.jp)

## Abstract

This paper includes a survey of recent studies for the integer knapsack problem, especially focusing on dominated terms that are not necessary on constructing an optimal solution. By discarding the terms we will gain a more small-sized and equivalent problem to the original. Some suggestions on identifying dominated terms in the integer knapsack problem are also included.

**Keywords:** combinatorial optimization; knapsack problem; dominance relation; subset-sum problem

## 1 Introduction

In the classical 0–1 knapsack problem (KP, also called binary knapsack problem),  $n$  items and a knapsack for carrying the items are given. Then, each item is of two properties, i.e. *profit* and *weight*, and the knapsack is of a capacity for the total weight of items carried. The KP is to find an optimal subset of the  $n$  items with maximum total profit without exceeding the capacity. It is well-known that the KP is  $\mathcal{NP}$ -hard. For more details on KP, see, e.g. Chapter 13 in Ibaraki and Fukushima [3] (in Japanese).

In this paper we study the integer knapsack problem (IKP, so-called unbounded knapsack problem). The problem is a generalization of KP so that each item is available in any number, i.e. enough to fill up the knapsack by only one type of item respectively. The IKP is formally stated as follows:

$$\text{maximize} \quad \sum_{j \in N} c_j x_j \quad (1)$$

$$\text{subject to} \quad \sum_{j \in N} a_j x_j \leq b \quad (2)$$

$$x_j \geq 0 \text{ (integer)}, j \in N, \quad (3)$$

where  $N := \{1, 2, \dots, n\}$  indicates the set of  $n$  items, and both the profit  $c_j$  and weight  $a_j$  associated with any  $j$ -th item and capacity  $b$  are positive integers. By replacing the formula (3) with  $x_j \in \{0, 1\}$  we have a well-known formulation for KP. Without loss of generality we will assume that  $a_j \leq b$  for any  $j \in N$  in order to exclude an unpromising item. Assuming that  $c_1/a_1 := \max_{j \in N} \{c_j/a_j\}$ , if  $b$  is divisible by the  $a_1$  then the problem is trivial. Throughout this paper, following the terminology in Zhu and Broughan [8], we call each  $j (\in N)$  *term* in place of *item*, while called *item type* in Martello and Toth [5].

We know that the IKP can be solved by dynamic programming scheme. The Bellman's recursion for IKP will be formally stated as follows:

$$v_j(\tilde{b}) = \begin{cases} v_{j-1}(\tilde{b}) & , a_j > \tilde{b} \\ \max \{v_{j-1}(\tilde{b}), v_j(\tilde{b} - a_j) + c_j\} & , a_j \leq \tilde{b} \end{cases}$$

where  $0 \leq \tilde{b} \leq b$ , and  $v_0(\tilde{b}) = 0$ . Then,  $v_n(b)$  gives an optimal value. By replacing the formula  $v_j(\tilde{b} - a_j) + c_j$  in the second row with  $v_{j-1}(\tilde{b} - a_j) + c_j$  we have a well-known recursion for KP (see, e.g. Dudziński and Walukiewicz [2]).

According to the formulation of the bounded knapsack problem (BKP) in Pisinger [7], we may assume that the IKP (1)–(3) is a BKP in which  $m_j := \lfloor b/a_j \rfloor$  for all  $j \in N$ . Indeed, each term  $j$  can only be taken up to  $m_j$  due to the capacity  $b$ . Also, a study on greedy heuristics for IKP is presented in Kohli and Krishnamurti [4].

The remainder of this paper is organized as follows: In the next section we will make a survey of dominance relations mentioned in

the literature, which play a central role to reduce a given IKP, i.e. to obtain a more small-sized and equivalent problem to the original. In Section 3 we will discuss other relations for identifying dominated terms.

## 2 Dominance relations

If  $a_j \leq a_k$  and  $c_j \geq c_k$  holds in an IKP then term  $k$  is not necessary to achieve the optimal value of the IKP, since replacing a term  $k$  with a term  $j$  in a solution including term  $k$  possibly improves without enlarging the total weight. Needless to say, which is peculiar to IKP, this is due to that each term is available in any number. Indeed, assuming the same case on KP we cannot discard item  $k$ , since both the items  $j$  and  $k$  will be included in an optimal solution.

**Definition:** following [5], a term is said to be dominated if the optimal value of IKP does not change when the term is removed from  $N$ . More precisely, following [8], if  $a_k \geq \sum_{j \neq k} l_j a_j$  and  $c_k \leq \sum_{j \neq k} l_j c_j$  holds then term  $k$  is dominated, where  $l_j$  is a nonnegative integer.

On identifying dominated terms in IKP, the following is proved in [5]:

**Theorem(Martello and Toth)** Given an instance of IKP and a term  $k$ , if there exists a term  $j$  such that

$$\lfloor a_k/a_j \rfloor c_j \geq c_k \quad (4)$$

then term  $k$  is dominated.

Following [8] we call this TH2, which obviously holds only if  $a_j \leq a_k$  and  $c_j/a_j \geq c_k/a_k$ . Roughly speaking, taking term  $j$ 's in  $\lfloor a_k/a_j \rfloor$  is better than a term  $k$  under the condition (4), due to the profit-to-weight ratio. In this case we also say that term  $j$  dominates term  $k$ . In

[5], the algorithm MTU2 is proposed to solve (especially large-scaled) IKP. After determining a core problem, the algorithm identifies dominated terms in the core by a procedure incorporating TH2. Also, the procedure is algorithmically improved by Dudziński [1].

When  $a_j = a_k$ , the dominance relation (4) is reduced to  $c_j \geq c_k$ . By this we may assume that there exist no two terms such that one has the same weight as the other's. In addition, when  $c_j/a_j = c_k/a_k$ , the (4) is reduced to  $\lfloor a_k/a_j \rfloor = a_k/a_j$ . Hence  $a_k$  is divisible by  $a_j$ , i.e.  $a_k \equiv 0(\text{mod } a_j)$ .

In general TH2 is very efficient to reduce IKP, however as mentioned in [5], it will be slightly affected by the minimum weight of a given problem. Namely, the larger the  $\min_j a_j$  is, the smaller the number of dominated terms identified is. For instance, the performance of MTU2 in weakly correlated cases is presented in [5], which is summarized in Table 1.

Table 1: average over 20 problems, performed on HP9000/840

$a_j$ : uniformly random in	running times in seconds	the number of undominated terms
[10, 1000]	0.45	4.4
[50, 1000]	1.23	15.2
[200, 1000]	1.65	25.3
$n$ : 10000		
$c_j$ : uniformly random in $[a_j - 100, a_j + 100]$		
$b$ : $10^5$		

As mentioned in [8], the performance of TH2 is weakened by a more strong correlation between the profits and weights. In a more particular case of SSP (Subset-Sum Problem), where the SSP indicates an IKP in which the profit is equal to the weight on all terms, i.e.  $c_j = a_j$  for all  $j \in N$ , the dominance relation (4) is reduced to  $a_k \equiv 0(\text{mod } a_j)$  due to  $c_j/a_j = c_k/a_k$ . On the unreasonable re-

sult of TH2 when applied to SSP in which the weights are randomly distributed in  $[10, u]$  where  $u = 1000, 10000$ , see Table 1 in [8]. Furthermore on SSP, when the produced weights are bounded, the larger the  $\min_j a_j$  is, the smaller the number of terms each weight of which is divisible by another's will be.

On the other hand, the following is proved in [8]:

**Theorem(Zhu and Broughan)** In IKP, if there exist two distinct terms  $j$  and  $k$  ( $j, k \in N \setminus \{1\}$ ) such that

$$a_j \equiv a_k \pmod{a_1}, \quad a_j \leq a_k \text{ and } p_j \leq p_k; \quad (5)$$

or

$$a_k \equiv 0 \pmod{a_1}, \quad (6)$$

where  $a_1 (> 1)$  satisfies  $c_1/a_1 = \max_{j \in N} \{c_j/a_j\}$ , and  $p_i := c_1 a_i - c_i a_1$  (for all  $i \in N \setminus \{1\}$ ), then term  $k$  is dominated.

Following [8] we call this TH4. Note that, in the proof of TH4, the assumption that the maximality of the profit-to-weight ratio to term 1 is only used for deriving (6). On the dominance relation (5), when  $a_j = a_k$ , only the third condition remains and is reduced to  $c_j \geq c_k$  as same as (4); otherwise ( $a_j < a_k$ ), the first condition can be written as  $a_k = a_j + l_1 a_1$ , where  $l_1$  is a positive integer. Then, the third condition  $p_j \leq p_k$  can be written as  $c_k \leq c_j + c_1(a_k - a_j)/a_1 = c_j + l_1 c_1$ . The (5) is thus corresponding to Definition.

On TH4, which is mentioned in [8], the number of dominated terms identified depends on the smallness of  $a_1$ , that is, the smaller the  $a_1$  is, the larger the number of dominated terms identified is. This implies that TH4 has the same drawback as TH2, that is, it is sensitive to the minimum weight of a given problem.

It should be pointed out that, in contrast to TH2, this theorem also favorably works to SSP (see Table 2 in [8]). Since  $p_i \equiv 0$  in SSP, the dominance relations (5) and (6) are reduced to those in Table 2 respectively. In this case,  $a_1$  is chosen so that  $a_1 \leq a_j$ , i.e. minimum weight, which seems to be one of essentials on the efficiency of TH4

to SSP.

Table 2: on SSP

dominance relations	
TH2	$a_k \equiv 0 \pmod{a_j}$
TH4	$a_j \equiv a_k \pmod{a_1}, a_j \leq a_k$ $a_k \equiv 0 \pmod{a_1}$

**Observation.** A term of secondary minimum weight will be helpful to improve TH4 on SSP when employed in place of  $a_1$ . In this case, we may exclude term 1 from candidates for dominated terms.

### 3 Suggestions

In this section we will outline three dominance relations, hoping that they will be helpful for the readers to devise a more efficient one. In what follows, in each case, term  $k$  is possibly dominated.

First, we present an extension of TH4.

**TH4':** If there exist three distinct terms  $i, j$  and  $k$  ( $i, j, k \in N \setminus \{1\}$ ) such that

$$a_i + a_j \equiv a_k \pmod{a_1}, a_i + a_j \leq a_k \text{ and } p_i + p_j \leq p_k,$$

where  $a_1$  and the definition of  $p_i$  are the same as those in TH4 respectively, then term  $k$  is dominated.

**Proof.** As same as the proof of TH4 we will obtain  $c_k \leq l_1 c_1 + c_i + c_j$  under an assumption that  $a_k = l_1 a_1 + a_i + a_j$ , where  $l_1$  is a nonnegative integer. ■

As anyone thinks it out, on implementing this, it will be helpful to

provide a table of all combinations of two terms' indices  $i, j$  associating with both  $p_i + p_j$  and  $a_i + a_j$ , the latter not exceeding maximum weight. In the case where  $n$  is even large, however, the size of the table grows too large. Needless to say our final aim is not to completely identify dominated terms. Therefore we should consider the tradeoff between the efficiency of TH4' and the computing time of making the table. In order to shrink the table provided, it will be preferable to provide an upper bound for whether the weight sum of two terms or the number of elements produced in the table. In addition TH4' can be easily extended to the case of the combination of three or more terms.

Second, similarly we can also consider an extension of TH2 as follows:

$$\lfloor a_k / (a_i + a_j) \rfloor (c_i + c_j) \geq c_k.$$

This holds only if  $(a_i + a_j) \leq a_k$  and  $(c_i + c_j) / (a_i + a_j) \geq c_k / a_k$ .

Last, while the dominance relations proposed hitherto in the literature are only concerned with terms, the following involves capacity  $b$ :

$$\forall i \in N, a_k + a_i > b \text{ and } \exists j \in N \text{ s.t. } c_j \geq c_k. \quad (7)$$

Under the condition (7), a feasible solution including term  $k$  is obviously constructed by just one term  $k$ . Therefore, if there exists term  $j$  of  $c_j$  greater than or equal to  $c_k$  then we have a possibly improved solution including term  $j$ . On (7), if there exists only one term of maximum profit  $\max_j c_j$  in an IKP, it remains; otherwise, at least one of the terms of maximum profit remains. In other words, it is not acceptable that all the terms of maximum profit are dominated. Also, the (7) will be interesting in a following sense:

**Example.** Let an instance of IKP in which  $c_1 = 3, a_1 = 8, c_2 = 2, a_2 = 6, c_3 = 1, a_3 = 5$ , and  $b = 10$ . By (7) term 1 dominates term 2, however, the weight of term 1 is greater than term 2's.

In addition the first part of (7) can be written as  $a_k + \min_{i \in N} a_i > b$ .

This implies that  $2a_k > b$ . Therefore, the (7) holds only if  $a_k \geq \lfloor b/2 \rfloor + 1$ . Assuming the case where  $\min_j a_j \geq \lfloor b/2 \rfloor + 1$ , only one term of  $\max_j c_j$  remains. Conversely no term can be dominated, provided  $\min_{i \in N} a_i + \max_{i \in N} a_i \leq b$ . Moreover the (7) is also applicable to KP, however, it implies that the (7) does not take advantage of the feature of IKP.

## 4 Comments

The dominance relations on the integer knapsack problem have been proposed in the literature in order to eliminate unnecessary terms, which contributes to efficiently solving the problem. In this paper we have focused on two efficient dominance relations, i.e. TH2 and TH4. We have also presented the extension of each of them, respectively. On the other hand we know that the two dominance relations have the same drawback. Then, the future research will turn to how to cope with the problem with even large minimum weight. It goes without saying that this will include to devise another dominance relation which will work against the hard case. Finally we would like to add that there will exist more room for consideration on a dominance relation which involves the capacity of a given problem, an actual example of which has already been presented in this paper.

## References

- [1] Dudziński, K. (1991) "A note on dominance relation in unbounded knapsack problems," *Opns. Res. Lett.* **10**(7) 417-419.
- [2] Dudziński, K. and S. Walukiewicz (1987) "Exact methods for the knapsack problem and its generalizations," *Euro. J. Opnl. Res.* **28**, 3-21.
- [3] 茨木俊秀, 福島雅夫『FORTRAN 77 最適化プログラミング』岩波書店 ISBN4-00-007700-7, Tokyo, Japan (First printing, April



- 1991).
- [4] Kohli, R. and R. Krishnamurti (1995) "Joint performance of greedy heuristics for the integer knapsack problem," *Discrete Appl. Math.* **56**, 37–48.
  - [5] Martello, S. and P. Toth (1990) "An exact algorithm for large unbounded knapsack problems," *Opns. Res. Lett.* **9**(1) 15–20.
  - [6] Martello, S. and P. Toth (1990) *Knapsack Problems: Algorithms and Computer Implementations*, Wiley, Chichester, England.
  - [7] Pisinger, D. (1995) "A Minimal Algorithm for the Bounded Knapsack Problem" in E. Balas and J. Clausen (eds.), *Integer Programming and Combinatorial Optimization, 4th IPCO conference, Lecture Notes in Computer Science* **920**, Springer, Berlin, 95–109.
  - [8] Zhu, N. and K. Broughan (1997) "On dominated terms in the general knapsack problem," *Opns. Res. Lett.* **21**(1) 31–37.