Core problems in the bi-criteria {0,1}-knapsack

Carlos Gomes da Silva(2,3,*), João Clímaco(1,3) and José Figueira(1,3,†)

(1) Faculdade de Economia da Universidade de Coimbra
Av. Dias da Silva, 165, 3004-512 Coimbra, Portugal
Phone: +351 239 760 500, Fax: +351 239 403 511
E-mail: figueira@fe.uc.pt

(2) Escola Superior de Tecnologia e Gestão de Leiria
Morro do Lena, Alto Vieiro, 2401-951 Leiria, Portugal
Phone: +351 244 820 300, Fax: +351 244 820 301
E-mail: cgsilva@estg.iplei.pt

(3) INESC-Coimbra
Rua Antero de Quental, 199
3000-033 Coimbra, Portugal
Phone: +351 239 851 040, Fax: +351 239 824 692
E-mail: jclimaco@inescc.pt

April, 2004

Abstract

The most efficient algorithms for solving single criterion {0,1}-knapsack problems are based on the core concept and the core problem. However, those concepts remain unnoticed in the multiple criteria case. In this paper we bring them to this topic.

A large amount of efficient solutions are considered for analysing the importance of the core itself and the core problem. The results show that the properties of the core in single criterion experiments are also observed in the solutions of the bi-criteria problem. These observations are used to derive a simple but effective method for solving the bi-criteria {0,1}-knapsack problem.

Key-words: Bi-criteria knapsack problem, Core problem, Exact methods, Combinatorial optimization

* Corresponding author

† Visiting Researcher (January-February, 2004) at the Operational Research Department, London School of Economics, Houghton Street, London WC2A 2AE, U.K.

1 Introduction

The {0,1}-knapsack problem is about selecting a set of items such that the sum of their values is maximized and the sum of their weights does not exceed the capacity of the knapsack. Dantzig (1957) showed that an optimal solution for the continuous {0,1}-knapsack problem can be obtained by sorting the items according to non-increasing *profit-to-weight ratios*, and including items until the knapsack is completed. At the end there is only one item which cannot be totally included. This item is called the *break* or *critical item*. With the items ordered this way, Balas and Zemel (1980) observed that for random instances the optimal solution for the {0,1}-knapsack problem is very similar to the continuous optimal solution. This similarity leads to the introduction of the core concept. The *core* is a subset of items around the critical item of the knapsack problem that must be considered to determine the exact solution, defining the so-called *core problem*. Results for large size instances showed that the size of the core is a very small proportion of the total number of items, and it increases very slowly with the latter (Balas and Zemel, 1980), which supports the existence of a small relevant problem.

All the variables corresponding to items outside the core are set to value 1 or 0, depending on their profit-to-weight ratio. The original problem is thus reduced, comprising only the items in the core. The exact core can only be determined after solving the original problem; an approximation of it is considered. In this case, considering only a small number of items - the ones in the core - is of great interest in the development of efficient algorithms. One of the reasons is that it avoids the complete sorting of the items required for deriving better upper and lower bounds. Balas and Zemel (1980) reported that this task absorbs a very significant part of the total computational time. It is also important because the solution of the core problem can further be used to provide improved lower bounds for the optimal solution of the original problem, thus allowing to fix the value of a significant number of variables at their optimal value.

The combination of these features gave rise to the most efficient algorithms: Fayard and Plateau (1982), Martello and Toth (1988), and Pisinger (1995).

Despite the importance of the core concept in the resolution of the single criterion {0,1}-knapsack, this concept remains unnoticed in the study of multiple criteria knapsack problems, *i.e.*, when several conflicting criteria are considered.

In this paper, we explore those concepts in the specific case of the bi-criteria problem, which can be formulated as follows:

$$
\begin{aligned}
&\max\ z_1(x_1,...,x_j,...,x_n) = \sum_{j=1}^{n} c_j^1 x_j \\
&\max\ z_2(x_1,...,x_j,...,x_n) = \sum_{j=1}^{n} c_j^2 x_j \\
&\quad s.t.: \\
&\quad \sum_{j=1}^{n} w_j x_j \leq W \\
&\quad x_j \in \{0,1\}, j = 1,...,n
\end{aligned}
\tag{1}
$$

where c_j^i represents the value of item j on criterion $i, i = 1, 2, x_j = 1$ if item j $(j = 1, ..., n)$ is included in the knapsack and $x_j = 0$ otherwise, w_j means the weight of item j and W is the overall knapsack capacity. We assume that c_j^1, c_j^2, W and w_j are positive integers and that

$w_j \leq W$ with $\sum_{j=1}^{n} w_j > W$. Constraints $\sum_{j=1}^{n} w_j x_j \leq W$ ($wx \leq W$, for short) and $x_j \in \{0,1\}, j = 1, ..., n$, define the feasible region in the *decision space*, and their image when using the criteria functions z_1 and z_2 define the feasible region in the *criteria space*. A feasible *solution*, x, is said to be *efficient* if and only if there is no feasible solution, y, such that $z_i(x) \leq z_i(y), i = 1, 2$ and $z_i(x) < z_i(y)$ for at least one i. The image of an efficient solution in the criteria space is called a *non-dominated solution*.

In our research, solving problem (1) consists of determining the set of all the efficient/non-dominated solutions. Certain efficient/non-dominated solutions can be obtained by maximizing weighted sums of the criteria, called supported efficient/non-dominated solutions, but there is a set of solutions, called *non-supported efficient/non-dominated* solutions, that cannot be obtained in this way, because despite being efficient/non-dominated, they are convex dominated by weighted sums of the criteria (Steuer, 1986). The non-supported non-dominated solutions are located in the dual gaps of consecutive supported non-dominated solutions.

The process of solving the bi-criteria problem can largely benefit from the developments proposed for solving the single criterion problem. In fact, solving (1) can be summarized as the computation of solutions which maximizes weighted sum functions (supported efficient solutions), *i.e.*, single criterion problems, and the computation of solutions that are in the way of those maximizations, just before reaching their optima, *i.e.*, approximate solutions of single criterion optimizations.

Despite the similarities between problems (1) and (2), below, the known algorithms for solving (1) (Visée *et al.*, 1998; Captivo *et al.*, 2003) are limited when compared with the ones proposed for solving the single criterion {0,1}-knapsack (Fayard and Plateau, 1982; Martello and Toth, 1988, and Pisinger, 1995), concerning the computational time and the size of instances which can be solved.

The rest of the paper is organized as follows: Section 2 presents the bi-criteria core problem and its influence in the criteria space. Section 3 concerns the bi-criteria core concept, the computational experiments on the size of the bi-criteria core, and the study of the worst cases found. Finally, Section 4 presents the main conclusions of this work.

2 Bi-criteria core problem

The core and the core problem were proposed by Balas and Zemel (1980) in the context of the single criterion {0,1}-knapsack problem, which is formulated as follows:

$$\begin{aligned} &\max\ p(x) = p(x_1, ..., x_j, ..., x_n) = \sum_{j=1}^{n} p_j x_j \\ &s.t.: \\ &\sum_{j=1}^{n} w_j x_j \leq W \\ &x_j \in \{0,1\}, j = 1, ..., n \end{aligned} \tag{2}$$

Consider the items ordered such that,

$$\frac{p_1}{w_1} \geq \frac{p_2}{w_2} \geq ... \geq \frac{p_n}{w_n} \tag{3}$$

Assuming that an optimal solution for the problem (2) is known, x^*, and the break item, b, given by $\sum_{j=1}^{b-1} w_j \leq W < \sum_{j=1}^{b} w_j$, then the exact *core*, C, showed in Figure 1, is defined as follows: $C = \{j_1, ..., j_2\}$, where $j_1 = \min\left\{j : x_j^* = 0, j = 1, ..., n\right\}$ and $j_2 = \max\left\{j : x_j^* = 1, j = 1, ..., n\right\}$. If $j_1 > j_2$ then $C = \{j_2, j_1\}$.

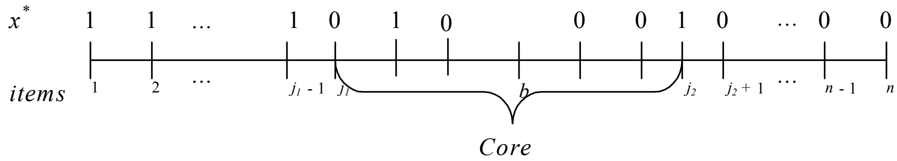

Figure 1: Exact core

Considering only the items in the core, C, a smaller knapsack problem is defined; it is called *core problem* and can be stated as follows,

$$\begin{array}{c} \max\ \widetilde{p} = \sum_{j \in C} p_j x_j \\ s.t.: \\ \sum_{j \in C} w_j x_j \leq \widetilde{W} \\ x_j \in \{0,1\}, j \in C \end{array} \tag{4}$$

where $\widetilde{W} = W - \sum_{j \in \{1,...,n\} \setminus C} w_j x_j$.

If the core is known in advance, then it will only be needed to solve problem (4) and setting the variables $x_1 = ... = x_{j_1 - 1} = 1$ and $x_{j_2+1} = ... = x_n = 0$, in order to obtain the optimal solution of problem (2). However, the exact core is not known until the problem is solved, thus if the procedure cannot prove the exactness of the core, the optimal solution may not be obtained in it.

In the bi-criteria case, given a core C, the core problem, $P(C)$, is:

$$\begin{array}{l} \max \widetilde{z}_1 = \sum_{j \in C} c_j^1 x_j \\ \max \widetilde{z}_2 = \sum_{j \in C} c_j^2 x_j \\ s.t.: \\ \sum_{j \in C} w_j x_j \leq \widetilde{W} \\ x_j \in \{0,1\}, j \in C \end{array} \tag{5}$$

where $\widetilde{W} = W - \sum_{j \in \{1,...,n\} \setminus C} w_j x_j$.

It is worthwhile to note that $P(C)$ spans a region in the criteria space which is precisely defined. Its boundaries can be easily computed considering the linear relaxation of problem (1) and the upper bounds on the values of criteria z_1 and z_2 in the core C.

Solving the linear relaxation of problem (1) consists of determining the entire set of extreme efficient solutions. This can be easily done by using an efficient pivoting process over a simplex tableau with bounded variables (see Gomes da Silva *et al.*, 2003, 2004a).

An upper frontier of the non-dominated solutions for problem (1) can be determined by linking all the extreme non-dominated solutions of the relaxed problem.

When computing the upper bounds on the values of criteria $z_i, i = 1, 2$, it can be used the one proposed by Dantzig (1957), which is the most simple to use.

Let,

- H, be a set of variables whose values are set equal to 1, that is, the items with profit-to-weight ratio greater than the ones of the core, C.

- b_{z_i}, be the break item in the core problem; and,

- $\overline{w}^i$, be the remaining capacity after including the items belonging to H and items from the core before item b_{z_i}.

It should be noticed that, the items in the core are ordered according to non-increasing values of the ratio $\dfrac{c_j^i}{w_j}$.

The Dantzig upper bound for z_i is then given by,

$$\overline{z}_i = \sum_{j \in H} c_j^i + \sum_{j \in C, j < b_{z_i}} c_j^i + \left\lfloor \overline{w}^i \frac{c_{b_{z_i}}^i}{w_{b_{z_i}}} \right\rfloor .$$

Figures 2 and 3 show the criteria space spanned by exactly solving two core problems in a problem with 100 items.

To define the sets H and C, the bi-critera problem (2) is converted into a single criterion one, by using a weighted sum function of the criteria, for example:

$$p(x) = 0.5z_1(x) + 0.5z_2(x)$$

According to this particular function, $p(x)$, the set C is composed of 21 items (11 items in the case of Figure 2) around the break item, and the set H is composed of the items located on the left of the core C.

To obtain the results displayed in Figure 2 and 3, we proceed as follows: 1) insert one by one the items of H (see dashed line up to the separation point in Figures 2 and 3); 2) sort the core items according to non-increasing values of the profit-to-weight ratio of z_1; 3) insert the items in the knapsack till its full capacity (see the line on the right from the separation point in Figures 2 and 3); 4) sort the core items according to non-increasing values of the profit-to-weight ratio of z_2; 5) insert the items in the knapsack till its full capacity (see the line on the left from the separation point in Figures 2 and 3); 6) solve the core problem using the method by Visée *et al.* (1998) which results in the solutions near the upper frontier and between the two lines above.

As it can be seen, sets H and C control the search of the criteria space: the searched area diminishes as variables change from C to H, reducing the size of C.

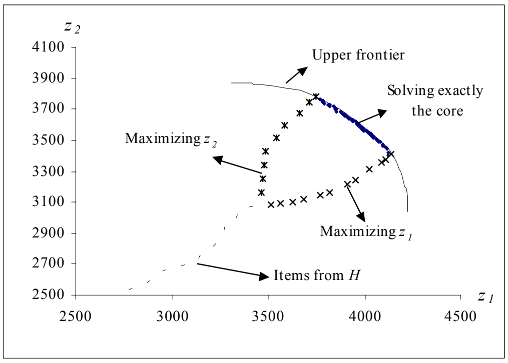

Figure 2: $p(x) = 0.5z_1 + 0.5z_2; |C| = 21$

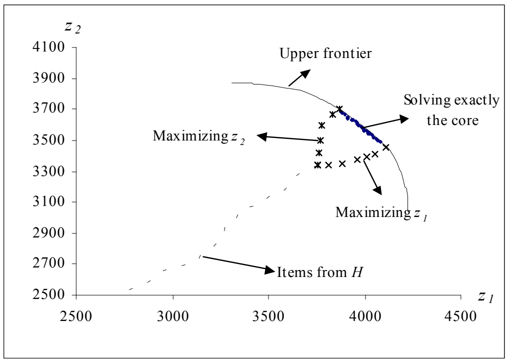

Figure 3: $p(x) = 0.5z_1 + 0.5z_2; |C| = 11$

3 Bi-criteria cores

The concept of core cannot be extended to multiple criteria problems in a straightforward manner. Some assumptions are required. To show this, let us consider the following instance of the bi-criteria problem with 10 items.

$$\begin{aligned}
&\max\ z_1 = 91x_1 + 38x_2 + 8x_3 + 77x_4 + 93x_5 + 13x_6 + 41x_7 + 81x_8 + 11x_9 + 93x_{10} \\
&\max\ z_2 = 35x_1 + 49x_2 + 86x_3 + 19x_4 + 50x_5 + 90x_6 + 21x_7 + 59x_8 + 63x_9 + 55x_{10} \\
&s.t.: \\
&33x_1 + 45x_2 + 23x_3 + 19x_4 + 70x_5 + 100x_6 + 63x_7 + 70x_8 + 55x_9 + 10x_{10} \leq 244 \\
&x_j \in \{0, 1\}, j = 1, ..., 10
\end{aligned}$$

This problem has five efficient solutions: $x^1 = (10011010011)$; $x^2 = (1011100101)$; $x^3 = (1110000111)$; $x^4 = (1110100011)$; $x^5 = (0111100101)$, with $z(x^1) = (293; 348)$, $z(x^2) = (443; 304)$, $z(x^3) = (320; 347)$, $z(x^4) = (332; 338)$, $z(x^5) = (388; 318)$.

Table 1 presents the exact core associated with each of the above five solutions, $x^t, t = 1, ..., 5$, and for three different functions, $p^k(x), k = 1, 2, 3$.

Let us now consider p solutions and q functions. The cores for each solution t and for each function k are defined as follows,

$$C^{k,t} = \left\{ j_1^{k,t}, ..., j_2^{k,t} \right\}$$

where $j_1^{k,t} = \min \left\{ j : x_j^t = 0, j = 1, ..., n \right\}$ and $j_2^{k,t} = \max \left\{ j : x_j^t = 1, j = 1, ..., n \right\}$.
Obviously, if $j_1^t > j_2^t$ then $C^t = \left\{ j_2^t, j_1^t \right\}$.

The last line of the Table 1 shows the core for each function considering the entire set of efficient solutions: $C^{k,*} = \left\{ j_1^{k,*}, ..., j_2^{k,*} \right\}$, where $j_1^{k,*} = \min \left\{ j : x_j^k = 0, t = 1, ..., p; j = 1, ..., n \right\}$ and $j_2^{k,*} = \max \left\{ j : x_j^k = 1,\ t = 1, ..., p; j = 1, ..., n \right\}$. Analogously, if $j_1^{k,*} > j_2^{k,*}$, then $C^{k,*} = \left\{ j_2^{k,*}, j_1^{k,*} \right\}$.

$p^1(x) = 0c^1 + 1c^2$	$p^2(x) = 1c^1 + 0c^2$	$p^3(x) = 0.5c^1 + 0.5c^2$
$C^{1,1} = \{2, 1, 4, 6\}$	$C^{2,1} = \{5, 8, 2, 7, 3, 9, 6\}$	$C^{3,1} = \{5, 8, 2, 9, 6\}$
$C^{1,2} = \{9, 2, 1, 4, 6, 8, 5\}$	$C^{2,2} = \{2, 7, 3\}$	$C^{3,2} = \{8, 2\}$
$C^{1,3} = \{4, 6, 8\}$	$C^{2,3} = \{4, 1, 5, 8, 2, 7, 3, 9, 6\}$	$C^{3,3} = \{4, 3, 1, 5, 8, 2, 9\}$
$C^{1,4} = \{4, 6, 8, 5\}$	$C^{2,4} = \{4, 1, 5, 8, 2, 7, 3, 9\}$	$C^{3,4} = \{4, 3, 1, 5, 8, 2, 9\}$
$C^{1,5} = \{9, 2, 1, 4, 6, 8, 5\}$	$C^{2,5} = \{1, 5, 8, 2, 7, 3\}$	$C^{3,5} = \{1, 5, 8, 2\}$
$C^{1,*} = \{9, 2, 1, 4, 6, 8, 5\}$	$C^{2,*} = \{4, 1, 5, 8, 2, 7, 3, 9, 6\}$	$C^{3,*} = \{4, 3, 1, 5, 8, 2, 9, 6\}$

Table 1: Exact cores for different $p(x)$ functions

From these results it can be concluded that:

1. For the same problem, several $p(x)$ functions can be built, which conditioned the composition of the core;

2. The exact core is also dependent on the efficient solution considered;

3. The core corresponding to the entire set of efficient solutions includes a large number of items;

4. The same solution is associated with different cores and different core sizes.

The example shows that when more than one criterion is considered, the profit-to-weight ratio is not well defined, once it can assume several values. It depends on how the weighted sum of the criteria, $p(x) = \lambda_1 z_1(x) + \lambda_2 z_2(x)$, is defined. Like in single criterion problems, we look for the smallest core of a given efficient solution. Thus, the following definition of bi-criteria core is proposed.

Definition 1 *Given a family of weighted sum functions, $\Im$, the exact bi-criteria core of an efficient solution of (1) is the smallest core when each function of $\Im$ is considered individually.*

In the example above, the exact bi-criteria core of solutions x^1, x^2, x^3, x^4, x^5 is $C^{1,1}, C^{3,2}, C^{1,3}$, $C^{1,4}$ and $C^{3,5}$, respectively.

According to Definition 1, determining the bi-criteria core of an efficient solution requires the analyzis of all functions of $\Im$. In order to obtain the smallest cores it is desirable that the most favourable function is available for determining the core of a given efficient solution. Since all the efficient solutions are considered, the weighted sum functions should span the locations of all of them.

A characterization of the entire non-dominated region is provided by the upper frontier generated from the linear relaxation of (1). This relaxation is useful to define $\Im$ since it consists of linking extreme non-dominated solutions, each of them obtained by maximizing an adequate weigthed sum function.

In the efficient pivoting process the criteria weights are normalized, *i.e.*, $\lambda_1 + \lambda_2 = 1, \lambda_1, \lambda_2 \geq 0$. With the normalization, it is only necessary to consider one weight, so the weighted sum function is $\lambda z_1(x) + (1 - \lambda) z_2(x)$, with $\lambda \in [0, 1]$.

If the linear relaxation has q distinct non-dominated solutions, this means that the interval $[0, 1]$ is divided into q sub-intervals, each of them representing the range of the weights that lead to the same non-dominated solution: $\left[\lambda^1, \lambda^2\right[, \left[\lambda^2, \lambda^3\right[, ..., \left[\lambda^q, 1\right], \lambda^1 > 0$. The computation of the upper limit of each interval is as follows.

Suppose that $\lambda = \lambda^k$ is the smallest value when the problem

$$\max \{\lambda z_1(x) + (1 - \lambda) z_2(x) : wx \leq W, x \in [0, 1]^n\}$$

is optimized, the solution x^k is obtained. Then, from the optimal conditions of the simplex method with bounded variables, the maximum value that λ can assume in order to keep obtainning x^k is given by: $\min \{v_1, v_2, 1\}$ where

$$v_1 = \min_{x_j^k = 1} \left\{ \frac{-\widetilde{c}_{j,f}^2}{\widetilde{c}_{j,f}^1 - \widetilde{c}_{j,f}^2} : \widetilde{c}_{j,f}^2 \geq 0, \widetilde{c}_{j,f}^1 - \widetilde{c}_{j,f}^2 < 0 \right\}$$

and,

$$v_2 = \min_{x_j^k = 0} \left\{ \frac{-\widetilde{c}_{j,f}^2}{\widetilde{c}_{j,f}^1 - \widetilde{c}_{j,f}^2} : \widetilde{c}_{j,f}^2 \leq 0, \widetilde{c}_{j,f}^1 - \widetilde{c}_{j,f}^2 > 0 \right\}$$

with,

x_f being the basic variable in the problem $\max \{\lambda z_1(x) + (1 - \lambda) z_2(x) : wx \leq W, x \in [0, 1]^n\}$;

$\widetilde{c}_{j,f}^1 = \dfrac{c_j^1}{w_j} - \dfrac{c_f^1}{w_f}$;

$\widetilde{c}_{j,f}^2 = \dfrac{c_j^2}{w_j} - \dfrac{c_f^2}{w_f}$.

Let $\lambda^{k+1} = \min \{v_1, v_2\}$. The λ^{k+1} is also the minimum value associated with the adjacent non-dominated solution. Consequently, in order to associate an interval with only one non-dominated solution, the open range $\left[\lambda^k, \lambda^{k+1}\right[$ is considered.

It should be noticed that despite the fact that the range $\left[\lambda^k, \lambda^{k+1}\right[$ is associtated with the same extreme non-dominated solution, the relation (3) is not stable in itself. This is due to the fact that relation (3) does not change if and only if

$$\lambda^{k+1} \leq \min_{j=1,\ldots,n-1} \left\{ \frac{-\widetilde{c}_{j,j+1}^2}{\widetilde{c}_{j,j+1}^1 - \widetilde{c}_{j,j+1}^2} : \widetilde{c}_{j,j+1}^2 \leq 0, \widetilde{c}_{j,j+1}^1 - \widetilde{c}_{j,j+1}^2 > 0 \right\}.$$

It is easy to see that this bound can never be higher than the bound obtained from the optimal condition of the simplex method, *i.e.*, $\min\{v_1, v_2\}$.

Hence, several $p(x)$ functions can be set in the interval $\left[\lambda^k, \lambda^{k+1}\right[$, perhaps giving rise to different bi-criteria cores. To overcome this drawback we consider the function $p^k(x) = \pi^k z_1(x) + \left(1 - \pi^k\right) z_2(x)$ with

		[1]	[2]	[3]					
	ESR	Av. core	Max Core	Min Core	STD	ES	$\frac{[1]}{n}100$	$\frac{[2]}{n}100$	$\frac{[3]}{n}100$
Average	48.30	9.47	30.27	2.00	5.50	124.93	9.46	30.27	2.00
Max	63.00	12.91	60.00	2.00	9.62	177.00	12.91	60.00	2.00
Min	39.00	7.56	16.00	2.00	3.22	73.00	7.56	16.00	2.00
STD	6.79	1.26	10.02	0.00	1.83	27.46	1.26	10.02	0.00

Table 2: $n = 100$ (30 instances; 3,748 efficient solutions)

$$\pi^k = \lambda^k + \frac{\lambda^{k+1} - \lambda^k}{2}, k = 1, ..., q \tag{6}$$

as the representative function of interval $\left[\lambda^k, \lambda^{k+1}\right[$. The family $\Im$ is thus defined as follows.

Definition 2 *A family $\Im$ is composed of functions $p^k(x) = \pi^k z_1(x) + \left(1 - \pi^k\right) z_2(x), k = 1, ..., q$, with q equal to the number of extreme non-dominated solutions of the linear relaxation of* (1) *and π^k defined as (6).*

3.1 Empirical experiments on the size of the bi-criteria core

In order to evaluate the size of the bi-criteria core of exact efficient solutions in the bi-criteria {0,1}-knapsack problem we proceed as follows: 1) generate the entire set of efficient solutions (using the exact method by Visée *et al.*, 1998); 2) generate the family of weighted sum functions, $\Im$; 3) the bi-criteria core is computed for each efficient solution, x^t.

The experiments comprises instances with $n = 100, 200, 300, 400, 500$, where the coefficients are randomly distributed in the range [0,100] and the capacity of the knapsack was set to 50% of the sum of the weights. The number of extreme non-dominated solutions of the linear relaxation problem (ESR), the average, maximum and minimum size of the core, and the number of efficient solutions (ES) were all computed for every instance. These descriptive statistics are also presented in percentage of the problem size. Tables 2, 3, 4, 5, 6 are referred to all the instances for each problem with the same size.

The obtained results show that on average the bi-criteria core is very small, increasing very slowly with the problem size (about 2 items per each increase of 100 items). However, it largely decreases in terms of relative size. These results are very similar to those obtained in single criterion problems. The standard deviation of the bi-criteria core is low, revealing the high "compactness" of the corresponding distribution.

		[1]	[2]	[3]					
	ESR	Av. core	Max Core	Min Core	STD	ES	$\frac{[1]}{n}100$	$\frac{[2]}{n}100$	$\frac{[3]}{n}100$
Average	98.27	12.25	52.57	2.00	7.35	410.33	6.12	26.28	1.00
Max	126.00	17.42	95.00	2.00	13.36	597.00	8.71	47.50	1.00
Min	81.00	9.57	28.00	2.00	4.13	285.00	4.78	14.00	1.00
STD	10.02	1.89	15.31	0.00	2.60	73.71	0.94	7.66	0.00

Table 3: $n = 200$ (30 instances; 12,316 efficient solutions)

		[1]	[2]	[3]					
	ESR	Av. core	Max Core	Min Core	STD	ES	$\frac{[1]}{n}100$	$\frac{[2]}{n}100$	$\frac{[3]}{n}100$
Average	148.17	13.60	81.33	2.00	9.31	767.07	4.53	27.11	0.67
Max	178.00	19.66	149.00	2.00	25.21	950.00	6.55	49.67	0.67
Min	118.00	11.11	31.00	2.00	4.39	499.00	3.70	10.33	0.67
STD	13.73	2.18	31.15	0.00	4.36	106.87	0.73	10.38	0.00

Table 4: $n = 300$ (30 instances; 22, 160 efficient solutions)

		[1]	[2]	[3]					
	ESR	Av. core	Max Core	Min Core	STD	ES	$\frac{[1]}{n}100$	$\frac{[2]}{n}100$	$\frac{[3]}{n}100$
Average	192.00	14.50	90.10	2.00	8.92	1207.90	3.63	22.53	0.50
Max	207.00	16.48	186.00	2.00	11.66	1409.00	4.12	46.50	0.50
Min	179.00	13.01	42.00	2.00	5.89	1022.00	3.25	10.50	0.50
STD	9.67	1.21	42.45	0.00	2.25	136.27	0.30	10.61	0.00

Table 5: $n = 400$ (10 instances; 12, 079 efficient solutions)

		[1]	[2]	[3]					
	ESR	Av. core	Max Core	Min Core	STD	ES	$\frac{[1]}{n}100$	$\frac{[2]}{n}100$	$\frac{[3]}{n}100$
Average	245.10	17.37	118.90	2.00	12.34	1754.60	3.47	23.78	0.40
Max	283.00	34.80	200.00	2.00	29.33	2208.00	6.96	40.00	0.40
Min	226.00	14.11	68.00	2.00	6.56	1425.00	2.82	13.60	0.40
STD	19.60	6.00	42.19	0.00	7.16	219.84	1.21	8.44	0.00

Table 6: $n = 500$ (10 instances; 17, 546 efficient solutions)

To examine the distribution of the percentage size of the core, all the efficient solutions for each problem with the same size were considered. The corresponding histograms are shown in Figures 4, 5, 6, 7, 8 (some statistics are also presented). As it can be seen, the configuration is always the same and shares other common factors: the median is very low and lower than the average, which is also small. About 88%($n = 100$), 96%($n = 200$), 98%($n = 300$), 98%($n = 400$), 97%($n = 500$) of the efficient solutions are associated with a bi-criteria core where the number of items corresponds only to 15% of the total number of items. The maximum percentage values for the core size are around 50%.

These results lead to an immediate conclusion, that is, for random instances, a very significant percentage of efficient solutions can be obtained by solving bi-criteria core problems of small size.

The most acute conclusion from these experiments concerns the consequences of the "compactness" of the bi-criteria core size which is a most promising path for the development of an effective exact method for solving efficiently the {0,1}-knapsack problem.

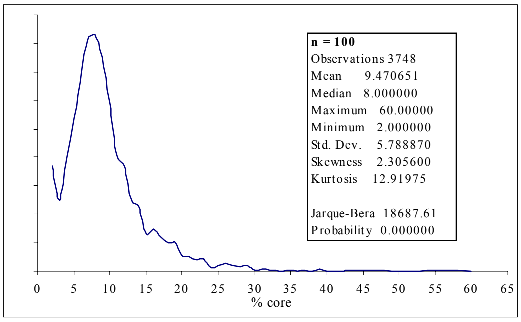

Figure 4: Percentual core for $n = 100$

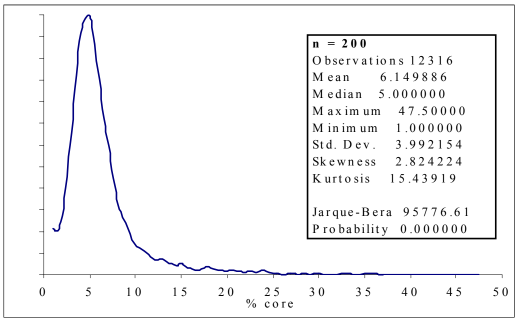

Figure 5: Percentual core for $n = 200$

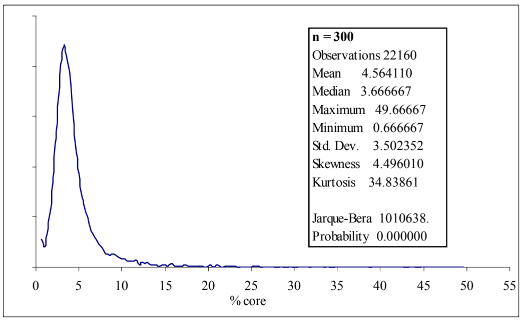

Figure 6: Percentual core for $n = 300$

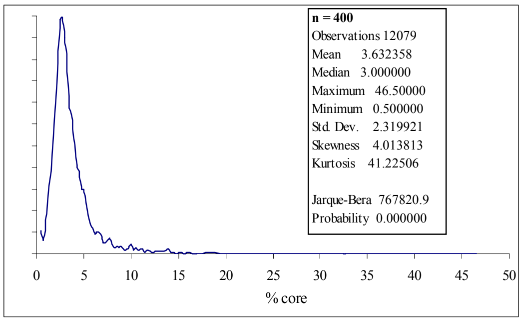

Figure 7: Percentual core for $n = 400$

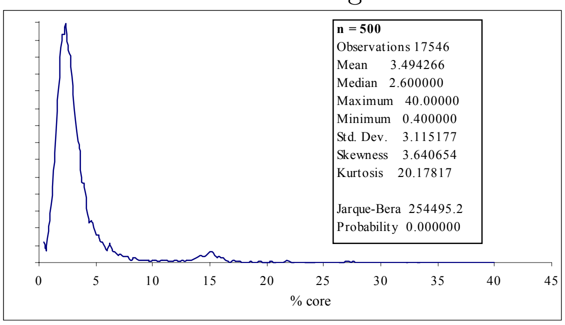

Figure 8: Percentual core for $n = 500$

n	$\|C\|$	$\sum_{j \in C} \|x_j - \widetilde{x}_j\|$	$\pi = \sum_{j \in C} \|x_j - \widetilde{x}_j\| / \|C\| \times 100$
100	46	9	19.56%
200	80	4	5.00%
300	147	4	2.72%
400	186	2	1.08%
500	200	4	2.00%

Table 7: Changed items in the bi-criteria core

3.1.1 Large size cores

In the previous experiments it was verified that some solutions (a very small number indeed) produce large exact bi-criteria cores with the functions of $\Im$. Although, when analyzing those cases in detail, a very surprising result can be seen. Despite the fact that the core is large, only a very small number of variables belonging to it assume a value different from the corresponding continuous solution (obtained by maximizing the weighted sum function, used to compute the exact bi-criteria core, in the space $\{x \in [0,1]^n, wx \leq W\}$). This can be observed in Table 7 where the size of the core is presented as well the number of variables with a different value from the continuous solution, and the percentage of items changed in the worst cases regarding the size of the bi-criteria core.

To understand why this happens, let $y_j = |x_j - \widetilde{x}_j|, j \in C$, where x and $\widetilde{x}$ are respectively an efficient solution and the associated continuous solution (the fractional variable is set to 0), and consider the product $w_j y_j$, which is only different from 0 in variables that are not concordant with the value in the associated continuous solution. Figure 9 shows an example of the behavior of this product, which is representative of what was observed in all instances in their worst cases concerning the core size.

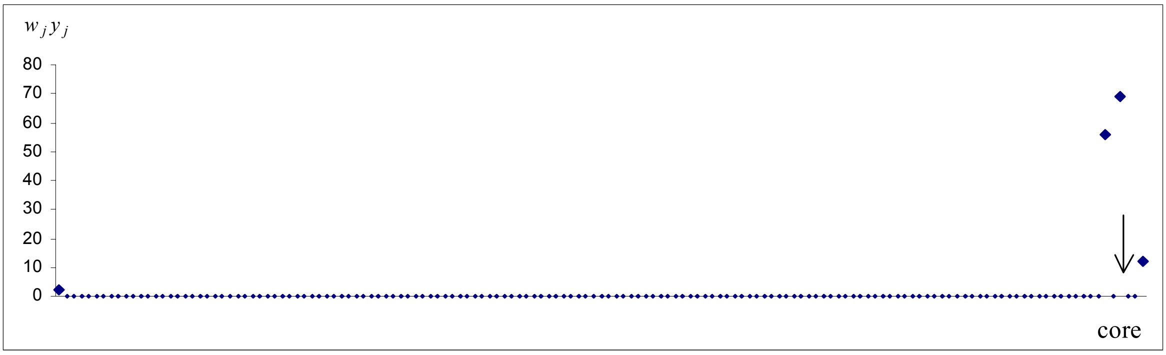

Figure 9: Changes in the core

Figure 9 shows that further changes in the value of variables are only made to accomodate significative changes in a very small neigboorhood of the break item, and that the weight of the item with the most extreme profit-to-weight ratio is also very small, meaning that the changes are derived to obtain a full knapsack. These observations are similar to the ones obtained by Pisinger (1999).

This observation has a great interest for enhancing the use of the core problem in the resolution of the bi-criteria {0,1}-knapsack problem, once it can be used as a guide for selecting variables.

4 A simple method for solving the bi-criteria {0,1}-knapsack

The "compactness" of the distribution of the bi-criteria core size can be explored to devise a simple method for solving the {0,1}-knapsack problem.

The method uses the linear relaxation of the problem for computing the family of representative weighted sum functions, $\Im$. Applying each function of $\Im$, the items are divided into three sets containing those with high, medium and low profit-to-weight ratios. The variables with the medium profit-to-weight ratios represent an approximation of the core (according to Section 3.1.1). The core is then extended with some items with the lowest weights from the other two sets. Variables belonging to the first set are fixed to 1, and the ones from the third group are set to 0. The core problem is solved exactly by the method of Visée *et al.* (1998) and the set of efficient solutions, $\widetilde{X}^{eff}$, is updated with the solutions obtained and completed with the values of the fixed variables. In order to avoid repetitions, the core problem is only solved if the partition (the devision made above) of the items was not previously obtained.

The method is presented below.

Method Bi-criteria_Core_Problem
BEGIN
    $\widetilde{X}^{eff} \leftarrow \phi$ // efficient solutions
    $Q \leftarrow \phi$ // analyzed structures
    c^{max} : size of the core (even number)
    α : number of additional elements to be added to the core (even number)
    Compute the family of functions $\Im = \{p^1(x), ..., p^k(x), ..., p^q(x)\}$
    FOR $k = 1$ TO q DO
    BEGIN
        Order the items according to non-increasing profit-to-weight ratios using $p^k(x)$
        $b \leftarrow \arg\min \left\{ t : \sum_{j=1}^{t-1} w_j \leq W < \sum_{j=1}^{t} w_j \right\}$ // break item
        $H^k \leftarrow \{1, ..., b - \frac{c^{\max}}{2} - 1\}$
        $C^k \leftarrow \{b - \frac{c^{\max}}{2}, ..., b + \frac{c^{\max}}{2}\}$
        $L^k \leftarrow \{b + \frac{c^{\max}}{2} + 1, ..., n\}$
        Remove the $\frac{\alpha}{2}$ lightest items from H^k and L^k, and add them to C^k
        IF $\left(Q \cap \left(H^k, C^k, L^k\right) = \phi\right)$ THEN
        BEGIN
            $Q \leftarrow Q \cup \left(H^k, C^k, L^k\right)$

$x_j \leftarrow 1, j \in H^k; x_j \leftarrow 0, j \in L^k$

$S \leftarrow \{\text{efficient solutions of } P(C^k)\}$ // solve $P\left(C^k\right)$

Update $\widetilde{X}^{eff}$ with S

END

END

END

To analyze the performance of this method, the instance used in the previous section, corresponding to the highest STD is considered. Based on the results from Section 3.1, the size of the core was set equal to 15, 17, 19, 21, 23 for $n = 100, 200, 300, 400, 500$, respectively. The number of additional members of the core was 10 for all the instances. The method is then applied and the solutions obtained, $\widetilde{Z}^{ref}$, are compared with the exact set of non-dominated solutions, Z^{ref}, and the percentage of the solutions found is taken into account. The experiments were performed on a Pentium 4 processor with 256 MB RAM and 40 GB hard disk at 1,495 Mhz. The method was implemented in Borland Delphi 4. The results obtained are presented in Table 8. As it can be seen the percentage of the non-dominated solutions found by the simple method is considerably high. It is also relevant to note that the performance of the method is not related with the number of items of the problem.

It should be noticed that the present state-of-the-art methods for solving the bi-criteria {0,1}-knapsack problem the above results are indeed very promising. In fact, a large amount of sofistication of the methods or a large amount of CPU time is needed in order to obtain similar outcomes (Gomes da Silva *et al.*, 2004b; Gandibleux *et al.*, 2001).

n	$\frac{\lvert Z^{ef} \cap \widetilde{Z}^{ef}\rvert}{\lvert Z^{ef}\rvert}$	$CPU(s)$
100	92.65%	8.020
200	87.15%	23.550
300	77.16%	75.140
400	85.41%	146.370
500	93.02%	605.990

Table 8: Performance of the method

5 Conclusions

In this paper, the core and the core problems were extended to the bi-criteria {0,1}-knapsack domain. The computational experiments conducted with random instances revealed that the characteristics found in the single criterion case are also found in the bi-criteria instances, *i.e.*, small sized cores, with low increase according to the dimension of the problem. This is certainly due to the hidden similarities when solving problems (1) and (2). It was also noticed that for the worst case of the size of the core, very few variables of the continuous solution were changed.

The distribution of the size of the bi-criteria exact core and the previous results suggest a very simple, although effective method for solving the bi-criteria problem.

Supported by these results the construction of an exact method based on the core itself and core problem is a promising stream of research for solving efficiently the bi-criteria {0,1}-knapsack problem.

Acknowledgements: The authors would like to acknowledge the support from MONET research project (POCTI/GES/37707/2001) and the third author also acknowledges the support of the grant SFRH/BDP/6800/2001 (Fundação para a Ciência e Tecnologia, Portugal).

References

[1] Balas, E., Zemel, E. (1980) "An algorithm for large zero-one knapsack problems", Operational Research, 28, pp. 1130-1154

[2] Captivo, M., Clímaco, J., Figueira, J., Martins, E., Santos, J.L. (2003) "Solving multiple criteria 0-1 knapsack problems using a labeling algorithm", Computers & Operations Research, 30, pp. 1865-1886

[3] Dantzig, G. (1957) "Discrete variable extremum problems", Operations Research, 5, pp. 226-277

[4] Fayard, D., Plateau, G. (1982) "An algorithm for the solution of the 0-1 knapsack problem", Computing, 28, pp. 269-287

[5] Gandibleux, X., Morita, H., Katoh, N. (2001) "The supported solutions used as a genetic information in a population heuristic", in E. Zitzler, K. Deb., L. Thiele, L., C.A. Coello Coello, (Eds), Proceedings of the first international conference on evolutionary multi-criterion optimization, Lecture Notes in Computer Science 1993, Springer, Berlin, pp. 429-442

[6] Gomes da Silva, C., Figueira, J., Clímaco, J. (2003) "An interactive procedure for the bi-criteria knapsack problems", Research Report, N^{o}4, INESC-Coimbra, Portugal (in Portuguese) http://www.inescc.pt/download/RR2003_04.pdf

[7] Gomes da Silva, C., Clímaco, J., Figueira, J. (2004a) "A scatter search method for the bi-criteria knapsack problems". To appear in European Journal of Operational Research (2004).

[8] Gomes da Silva, C., Figueira, J., Clímaco, J. (2004b) "Integrating partial optimization with scatter search for solving the bi-criteria {0, 1}-knapsack problem", Research Report, N^{o}7, INESC-Coimbra, Portugal. http://www.inescc.pt/download/RR2004_07.pdf (Submitted)

[9] Martello, S., Toth, P. (1990) "A new algorithm for the 0-1 knapsack Problems", Management Science, 34, pp. 633-645

[10] Martello, S., Toth, P. (1990) Knapsack Problems - Algorithms and Computer Implementations, John Wiley & Sons, New York

[11] Pisinger, D. (1995) "An expanding-core algorithm for the exact 0-1 knapsack problem", European Journal of Operational Research, 87, pp. 175-187

[12] Pisinger, D., Toth, P. (1998) "Knapsack problems" in Du, D-Z., Pardalos, P. (Eds), Vol 1, Handbook of Combinatorial Optimization, Kluwer Academic Publishers, Dordrecht.

[13] Pisinger, D. (1999) "Core Problems in knapsack algorithms", Operations Research, 47 (4), pp. 570-575

[14] Visée, M., Teghem, J., Ulungu, E.L. (1998) "Two-phases method and branch and bound procedures to solve the bi-objective knapsack problem", Journal of Global Optimization, 12, pp. 139-155

[15] Steuer, R. (1986) Multiple Criteria Optimization, Theory, Computation and Application, John Wiley & Sons, New York