

# The 0/1 Multidimensional Knapsack Problem and Its Variants: A Survey of Practical Models and Heuristic Approaches

Soukaina Laabadi<sup>1</sup>, Mohamed Naimi<sup>2\*</sup>, Hassan El Amri<sup>1</sup>, Boujemâa Achchab<sup>2</sup>

<sup>1</sup>ENS-Hassan II University, Laboratory of Mathematics and Applications, Casablanca, Morocco

<sup>2</sup>EST-Hassan I University, Laboratory of Analysis, Modeling Systems and Decision Support, Berrechid, Morocco

Email: soukainalaabadi@gmail.com, \*mohamed.naimi@uhp.ac.ma, hassanelamri@gmail.com, achchab@estb.ac.ma

**How to cite this paper:** Laabadi, S., Naimi, M., El Amri, H. and Achchab, B. (2018) The 0/1 Multidimensional Knapsack Problem and Its Variants: A Survey of Practical Models and Heuristic Approaches. *American Journal of Operations Research*, 8, 395-439. <https://doi.org/10.4236/ajor.2018.85023>

**Received:** July 20, 2018

**Accepted:** September 27, 2018

**Published:** September 30, 2018

Copyright © 2018 by authors and Scientific Research Publishing Inc. This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## Abstract

The 0/1 Multidimensional Knapsack Problem (0/1 MKP) is an interesting NP-hard combinatorial optimization problem that can model a number of challenging applications in logistics, finance, telecommunications and other fields. In the 0/1 MKP, a set of items is given, each with a size and value, which has to be placed into a knapsack that has a certain number of dimensions having each a limited capacity. The goal is to find a subset of items leading to the maximum total profit while respecting the capacity constraints. Even though the 0/1 MKP is well studied in the literature, we can just find a little number of recent review papers on this problem. Furthermore, the existing reviews focus particularly on some specific issues. This paper aims to give a general and comprehensive survey of the considered problem so that it can be useful for both researchers and practitioners. Indeed, we first describe the 0/1 MKP and its relevant variants. Then, we present the detailed models of some important real-world applications of this problem. Moreover, an important collection of recently published heuristics and metaheuristics is categorized and briefly reviewed. These approaches are then quantitatively compared through some indicative statistics. Finally, some synthetic remarks and research directions are highlighted in the conclusion.

## Keywords

0/1 Multidimensional Knapsack Problem, Survey, Real-World Applications, Heuristics, Metaheuristics

## 1. Introduction

The 0/1 MKP can be informally stated as the problem of packing items into a knapsack while staying within the limits of different constraints (dimensions).

They can be, for example, the maximum weight that can be carried, the maximum available volume, or/and the maximum amount that can be afforded for the items. Each item has a profit level assigned to it, and weight at each dimension. Moreover, the knapsack has a limited capacity on each dimension. The goal is to select a sub-set of items that maximizes the sum of their profits and keep the total weight on each dimension no more than the corresponding capacity. A detailed description can be found in [1].

A study of the Stony Brook University Algorithm Repository [2], carried out in 1998, stipulates that the knapsack problem (especially the MKP) was the 18th most popular and the 4th most needed problem among 75 other algorithmic problems. The popularity of MKP stems from the fact that it has attracted researchers from both camps: the theoreticians as well as the practitioners enjoy the fact that this problem is a special version of the general zero-one integer programming problem. On the other hand, due to its well-known NP-Hardness, many researchers choose the 0/1 MKP as a test problem for their new resolution approaches. Moreover, practitioners enjoy the fact that this simple structured problem can be used as a sub-problem to solve more complicated ones or can model many industrial problems like the loading problem [3] [4], cutting stock [5], task assignment and multiprocessor scheduling [6], as well as economic opportunities, such as project selection [7], capital budgeting [8] [9] [10], etc.

The MKP first appeared in the context of capital budgeting [9] [10]. A comprehensive overview of practical and theoretical results for the MKP can be found in the monograph on knapsack problems by [11]. A review of the MKP was given by [12]. An Elaborate literature on the MKP and its relations to different problems are published elsewhere [13] [14] [15]. Furthermore, a recent survey of the most popular algorithms that have been used for solving MKP, including exact and heuristic methods, can be found in [16]. We can also mention [17] for a recent survey of structures and algorithms of 0/1 MKP. On the other hand, [18] are interested in multi-objective MKP. They classify and briefly discuss the existing resolution approach on this topic, especially the metaheuristics. In the same way, [19] focused their research only on genetic algorithms. They review and compare 11 variants of this metaheuristic approach for solving MKP.

The 0/1 MKP is strongly NP-hard problem [14]. In other words, its exact resolution is very expensive in terms of computing time. Thus, heuristic and metaheuristic approaches have been proposed in order to achieve an approximate solution within a reasonable amount of time, but without ensuring the optimality. Other hybrid methods can be developed by combining a heuristic/metaheuristic with another heuristic/metaheuristic [20], or by combining an exact method with another exact method [21], or by combining a heuristic/metaheuristic with an exact method [22] [23].

## 2. Variants of MKP

The applicability of the 0/1 MKP in different areas has given rise to several variants, which may be modeled differently depending on the constraints of the

problem, types of variables, types of data, dimensions of the bag, goals of the problem, etc. According to the types of data, there are two versions of 0/1 MKP: Deterministic and non-deterministic 0/1 MKP. In the first version, all data are assumed to be known in advance. However, this is not the case in the second version.

## 2.1. Deterministic Variants of MKP

### 2.1.1. Standard 0/1 MKP

Given  $N$  items with profits  $p_i > 0$  and a knapsack with  $d$  dimensions. Each item  $i$  consumes an amount  $w_{ji} \geq 0$  from each dimension  $j$ . Knowing that each dimension has a capacity  $C_j > 0$ , the goal is to maximize the sum of profits of the items in the knapsack so that the sum of weights in each dimension  $j$  does not exceed  $C_j$ . Formally, the 0/1 MKP could be expressed with an integer programming model:

$$\begin{cases} \text{Max} \sum_{i=1}^N p_i x_i & (1) \\ \text{s.t} \sum_{i=1}^N w_{ji} x_i \leq C_j; \forall j = 1, \dots, d; & (2) \\ x_i \in \{0, 1\}; \forall i = 1, \dots, N; & (3) \end{cases}$$

Equation (2) is the capacity constraint of resources.

Equation (3) indicates that  $x_i$  is a binary decision variable, it equals to 1 if  $i$ -th item is selected, and 0 otherwise.

If  $d = 1$ , MKP reduces to the 0/1 Knapsack Problem:

$$\begin{cases} \text{Max} \sum_{i=1}^N p_i x_i & (4) \\ \text{s.t} \sum_{i=1}^N w_i x_i \leq C; & (5) \\ x_i \in \{0, 1\}; \forall i = 1, \dots, N; & (6) \end{cases}$$

### 2.1.2. The 0/1 Multiple MKP

The 0/1 Multiple MKP (0/1 MMKP) differs from the 0/1 MKP in the number of knapsacks. Instead of a single knapsack, we consider multiple knapsacks where each one, say  $k$ , has  $d$  dimensions with limited capacity  $C_j^k$ . Moreover, each item  $i$  consumes an amount  $w_{ij}^k$  from each dimension  $j$  of each knapsack  $k$ . The decision here is not only whether to select a single item but also to which knapsack it is packed. Similarly, we introduce a binary variable of  $x_{ik} = 1$  to represent that item  $i$  is selected and packed into knapsack  $k$  and  $x_{ik} = 0$  otherwise. Mathematically, the 0/1 MMKP is given by:

$$\begin{cases} \text{Max} \sum_{k=1}^M \sum_{i=1}^N p_i x_{ik} & (7) \\ \text{s.t} \sum_{i=1}^N w_{ij}^k x_{ik} \leq C_j^k; \forall j = 1, \dots, d; \forall k = 1, \dots, M; & (8) \\ \sum_{k=1}^M x_{ik} \leq 1; \forall i = 1, \dots, N; & (9) \\ x_{ik} \in \{0, 1\}; \forall i = 1, \dots, N; k = 1, \dots, M; & (10) \end{cases}$$

Equation (8) means that the total size of items may not exceed the dimension's capacity of each knapsack.

Equation (9) ensures that each item appears at most once in all knapsacks.

### 2.1.3. The 0/1 Multiple-Choice MKP

The 0/1 Multiple-Choice MKP (0/1 MCMKP) is a more complex variant of the 0/1 MKP. In fact, we are given a set of items  $N$  divided into  $n$  disjoint groups  $N = N_1 \cup N_2 \cup \dots \cup N_n$ , for all  $k, k' \in \{1, 2, \dots, n\}$  we have  $N_k \cap N_{k'} = \emptyset$ . Each item  $i, i \in N_k$ , has a profit  $p_{ik} > 0$ , and requires resources given by the weight vector  $w_{ik}$  ( $w_{ik}^1, w_{ik}^2, \dots, w_{ik}^d$ ) where  $d$  is the dimension of knapsack. The amounts of available resources are given by  $C_j$ . The aim of the 0/1 MCMKP is to pick exactly one item from each class in order to maximize the total knapsack's profit, without violating the capacity constraint of each dimension. In the same manner, we introduce a binary variable  $x_{ik}$  which equals to 1, if the item  $j$  of the  $k$ -th class is selected, and equals to 0 otherwise. Formally, the MCMKP can be stated as follows:

$$\left\{ \begin{array}{l} \text{Max} \sum_{k=1}^n \sum_{i \in N_k} p_{ik} x_{ik} \end{array} \right. \quad (11)$$

$$\left\{ \begin{array}{l} \text{s.t.} \sum_{k=1}^n \sum_{i \in N_k} w_{ik}^j x_{ik} \leq C_j, \forall j = 1, \dots, d; \end{array} \right. \quad (12)$$

$$\left\{ \begin{array}{l} \sum_{i \in N_k} x_{ik} = 1, \forall k = 1, \dots, n; \end{array} \right. \quad (13)$$

$$\left\{ \begin{array}{l} x_{ik} \in \{0, 1\}; \forall k = 1, \dots, n; i \in N_k; \end{array} \right. \quad (14)$$

Equation (12) means that the sum of selected items may not exceed each dimension's capacity of the knapsack.

Equation (13) ensures that only one item is selected from each class.

### 2.1.4. The 0/1 Multi-Objective MKP

The 0/1 Multi-Objective MKP (0/1 MOMKP) is a MKP with conflicting objectives (or criteria). In fact, we are given a knapsack with  $d$  dimensions, a set of  $N$  items, and a set of  $M$  objectives, where  $p_{ij}^k$  is the profit of item  $i$  relative to the objective  $k$  when it is selected for the  $j$ -th dimension of knapsack,  $w_{ij}$  is the weight of the item  $i$  when it is selected for the  $j$ -th dimension of knapsack, and  $C_j$  is the capacity of the  $j$ -th dimension of knapsack. The goal is to select a subset of the items so that the total profit of the selected items about each objective is a maximum, while respecting the capacity constraint of each knapsack dimension. So, the 0/1 MOMKP can be formulated as follows:

$$\left\{ \begin{array}{l} \text{Max} z_k = \sum_{i=1}^N \sum_{j=1}^d p_{ij}^k x_{ij}, \forall k = 1, \dots, M; \end{array} \right. \quad (15)$$

$$\left\{ \begin{array}{l} \text{s.t.} \sum_{i=1}^N w_{ij} x_{ij} \leq C_j, \forall j = 1, \dots, d; \end{array} \right. \quad (16)$$

$$\left\{ \begin{array}{l} x_{ij} \in \{0, 1\}; \forall i = 1, \dots, N; j = 1, \dots, d; \end{array} \right. \quad (17)$$

Equation (16) is the capacity constraint of knapsack dimensions.

Equation (17) means that  $x_{ij}$  is a binary decision variable, it equals to 1 if the item  $i$  is selected for the  $j$ -th dimension of knapsack, and 0 otherwise.

For more information about this problem, we invite readers to read the recent doctoral dissertation referenced in [24].

## 2.2. Non-Deterministic Variants of MKP

### 2.2.1. The 0/1 Stochastic MKP

In a stochastic version of the 0/1 MKP, we assume that the sizes of items are independent random variables that each size follows the same type of probability distribution, not necessarily with the same parameter. A joint probabilistic constraint is imposed on the capacity constraints and the objective function is the same as that of the deterministic problem. We denote the item sizes by  $\xi_{ji}$  (instead of  $w_{ji}$ ), and we formulate the problem as a probabilistic constrained stochastic programming, where capacity constraints of 0/1 MKP “Equation (2)” are replaced by the following joint probabilistic constraint:

$$\left\{ \begin{array}{l} \text{Max} \sum_{i=1}^N p_i x_i \end{array} \right. \quad (18)$$

$$\left\{ \begin{array}{l} \text{Pr} \left( \sum_{i=1}^N x_i \xi_{ji} \leq C_j \right) \geq q; \forall j = 1, \dots, d; \end{array} \right. \quad (19)$$

$$\left\{ \begin{array}{l} x_i \in \{0, 1\}; \forall i = 1, \dots, N; \end{array} \right. \quad (20)$$

where  $q \in [0, 1]$  is a fixed probability level (e.g. 0.9, 0.95). Assuming the random variables  $\xi_{ji}$  are independent, the joint probabilistic constraint “Equation (19)” can be written as follows:

$$\prod_{j=1}^d \text{Pr} \left( \sum_{i=1}^N x_i \xi_{ji} \leq C_j \right) \geq q. \quad (21)$$

Note that in some real-life applications, we can't assume that the random vectors are independent. In this case, we have the following inequality:

$$\text{Pr} \left( \sum_{i=1}^N x_i \xi_{ji} \leq C_j \text{ for } j = 1, \dots, d \right) \geq \prod_{j=1}^d \text{Pr} \left( \sum_{i=1}^N x_i \xi_{ji} \leq C_j \right) \geq q. \quad (22)$$

Then by replacing “Equation (19)” with “Equation (21)”, the problem becomes as following:

$$\left\{ \begin{array}{l} \text{Max} \sum_{i=1}^N p_i x_i \end{array} \right. \quad (23)$$

$$\left\{ \begin{array}{l} \prod_{j=1}^d \text{Pr} \left( \sum_{i=1}^N x_i \xi_{ji} \leq C_j \right) \geq q; \end{array} \right. \quad (24)$$

$$\left\{ \begin{array}{l} x_i \in \{0, 1\}; \forall i = 1, \dots, N; \end{array} \right. \quad (25)$$

For more details, you can refer to [25].

### 2.2.2. The 0/1 Fuzzy MKP

There are many knapsack-type problems that involve items whose weights or profits are not precisely known. One of the methods of dealing with imprecision is applying the fuzzy sets theory. Thus, the fuzzy numbers have been applied to

model the imprecise weights and profits in the 0/1 MKP. The goal of this variation of 0/1 MKP is to achieve a given accepted level of profit without exceeding a given capacity of each dimension of knapsack. Let  $\tilde{p}_i$  be a fuzzy interval which models the imprecise profit of item  $i$  and let  $\tilde{w}_{ji}$  be a fuzzy interval which models the imprecise weight of the item  $i$  in dimension  $j$ . So, the 0/1 Fuzzy MKP (0/1 FMKP) can be stated as follows:

$$\left\{ \begin{array}{l} \text{Max} \sum_{i=1}^N \tilde{p}_i x_i \end{array} \right. \quad (26)$$

$$\left\{ \begin{array}{l} \sum_{i=1}^N x_i \tilde{w}_{ji} \leq C_j; \forall i = 1, \dots, N; \forall j = 1, \dots, d; \end{array} \right. \quad (27)$$

$$\left\{ \begin{array}{l} x_i \in \{0, 1\}; \forall i = 1, \dots, N; \end{array} \right. \quad (28)$$

### 3. Real-World Applications

Many real-life problems can be modeled as the 0/1 MKP. In this section, we will give some examples issued from different fields: Logistics, informatics, telecommunications, finance and civil engineering.

#### 3.1. Multi-Unit Combinatorial Auctions

The Combinatorial Auctions (CAs), are a publicly held sale at which property or goods are sold to the highest bidder. Indeed, the auction is done in two main steps. Firstly, participants submit their bids. Secondly, the auctioneer is facing a problem of allocating prices to properties, so as to maximize its income, which is relative to the sum of all offers submitted by participants and accepted by the auctioneer. This problem is called Winner Determination Problem (WDP). A prominent variation of CAs, is Multi-Unit Combinatorial Auctions (MUCA). It differs from CAs or more precisely Single-Unit CAs in the number of copies of the same type of goods. However, in the Single-Unit CAs, the only one unit per good is available. [26] conducted their research on the MUCA, they formulated the WDP of this variant as a 0/1 MKP. The profit  $p_i$  used in the MDKP corresponds to the price of  $i$ -th bid.  $u_j$  represents the available amount of unit  $j$ , while the resource consumption  $r_{ij}$  is regarded as the number of units  $j$  requested in  $i$ -th bid. The decision variable  $x_i$  equals to 1 if the  $i$ -th bid is accepted by the auctioneer, and it equals to 0, otherwise. So, the formulation can be stated as follows:

$$\left\{ \begin{array}{l} \text{Max} \sum_i p_i x_i \end{array} \right. \quad (29)$$

$$\left\{ \begin{array}{l} \text{s.t} \sum_i r_{ij} x_i \leq u_j; \forall j; \end{array} \right. \quad (30)$$

$$\left\{ \begin{array}{l} x_i \in \{0, 1\}; \forall i; \end{array} \right. \quad (31)$$

Despite the close relationship between 0/1 MKP and WDP, the literature addresses both problems independently. [27] was the first to establish this relationship. Recently, [28] introduced a direct comparison between the WDP of the different types of auctions and their corresponding family of knapsack, proving the effectiveness of the developed algorithms for 0/1 MKP to solve the WDP.

### 3.2. Allocation Resources with Stochastic Demands

The well-known problem of allocating frequency that consists of assigning the electromagnetic spectrum to frequency bands becomes more and more complex due to the popularity and complexity of the most recent networked applications. They are supported by a variety of end system services in the cloud and different kind of networks. Resources in distributed computer systems include those in endpoint devices such as CPU, memory and disk, as well as those in networked system such as switches and routers. To serve multiple users running networked applications simultaneously, we need to satisfy their requests without violating the resource capacity. In other words, we consider the computing system as a unit that has limited resources and admits only one subset of users/tasks at the same time. We assign weight and profit to each user. Thus, the aim is to reply to maximum requests of users or tasks without exceeding the resource limitations. [29] treat the allocation resources when the demands may change over time and only when their statistics or distributions are known a priori. They have modeled this allocation problem as a stochastic 0/1 MKP, where the resources correspond to the dimensions of knapsack, while the users correspond to items to be stored in the knapsack. Each user  $j$  requests  $A_{ij}$  of resources  $i$ , where  $A_{ij}$  is a random matrix based on a certain distribution or having some known statistics, we assume that:  $A_{ij}$  et  $A_{ik}$  are independent whether  $j \neq k$ , if they have correlated demands, we consider them as one task and merge their demands. The profits used in the 0/1 MKP correspond to the satisfied demands  $c_j$  of each user  $j$ , the capacity  $b_i$  of each resource  $i$  represents the subset of users that can be admitted at the same time by the  $i$ -th resource. Let  $p$  denotes the overflow probability that indicates the maximum frequency in which admitted users/tasks may violate the capacity constraints. The binary decision variable  $x_j$  equals to 1 whether the demands of the user  $j$  are satisfied, and it equals to 0, otherwise. So, the problem is formulated as follows:

$$\left\{ \begin{array}{l} \text{Max} \sum_j c_j x_j \end{array} \right. \quad (32)$$

$$\left\{ \begin{array}{l} \text{s.t } Pr \left( \sum_j A_{ij} x_j > b_i \right) \leq p; \forall i; \end{array} \right. \quad (33)$$

$$\left\{ \begin{array}{l} x_i \in \{0,1\}; \forall j; \end{array} \right. \quad (34)$$

This problem can be solved by probabilistic methods using scenario approximation or simple average approximation (SAA) [30]. For some distributions such as Bernoulli, the calculation of the probability  $Pr \left( \sum_j A_{ij} x_j > b_i \right)$  is very difficult, which makes the problem strongly NP-hard.

### 3.3. Frequency Allocation in Cognitive Radio Networks

The well-known problem of allocating frequency that consists of assigning the electromagnetic spectrum to frequency bands becomes more and more complex due to the tremendously increasing demand of spectrum for new wireless devic-

es and applications. Consequently, this situation leads to an inefficiency in spectrum utilization. Hence comes the idea of changing the static spectrum allocation manner to dynamic manner so as to exploit more the unused spectrum. In order to address this issue, cognitive radio (CR) has been proposed as a prospective approach for achieving dynamic spectrum access [31].

The concept of CR is as follows: A so-called secondary user may at any time get access using the free frequency bands, *i.e.* not occupied by users having an open use type license (named primary users). In fact, the secondary user must use the frequency band once the demand of primary user is satisfied, or when the signal gets too weak. Generally, The CR uses the spectrum dynamically, which leads to the efficient use of radio frequencies without generating gaps in the spectrum.

Many authors have taken advantage of the optimization techniques to solve the spectrum allocation. For example, [32] formulated the allocation spectrum as a 0/1 MMKP. They consider CR having  $n$  cognitive users and a Centralized Coordinator Node (CCN) which collects the instantaneous reports from each cognitive user and decides the availability of the spectrum accordingly. They admit that there is a common channel for the communications between cognitive users and the CCN. Each cognitive user is a pair of transmitter and receiver. The CCN collects the transmission requests from all cognitive users and allocates the available spectrum optimally. Before formulating the problem mathematically, we introduce some notations:

$m$ : The number of primary users  $j$ .

$n$ : The number of cognitive users  $i$ .

$B_j$ : The primary bandwidth associated to the primary user  $j$ .

$K_j$ : The interference temperature at the primary band  $j$ .

$d_i$ : The bandwidth request from the  $i$ -th cognitive user to CCN for the current transmission.

$I_{i,j}$ : The level of interference generated by an accepted bandwidth request from the  $i$ -th cognitive user, to be assigned in primary band  $j$ .

$$\left\{ \begin{array}{l} \text{Max} \sum_{j=1}^m \sum_{i=1}^n h(d_i) x_{ij} \end{array} \right. \quad (35)$$

$$\left\{ \begin{array}{l} \text{s.t} \sum_{i=1}^n d_i x_{ij} \leq B_j; \forall j = 1, \dots, m; \end{array} \right. \quad (36)$$

$$\left\{ \begin{array}{l} \sum_{i=1}^n I_{i,j} x_{ij} \leq K_j; \forall j = 1, \dots, m; \end{array} \right. \quad (37)$$

$$\left\{ \begin{array}{l} \sum_{j=1}^m x_{ij} \leq 1; \forall i = 1, \dots, n; \end{array} \right. \quad (38)$$

$$\left\{ \begin{array}{l} x_{ij} \in \{0,1\}; \forall i = 1, \dots, n; j = 1, \dots, m; \end{array} \right. \quad (39)$$

Equation (35) indicates that the goal of this problem is to meet as many as possible with the cognitive user's requests, by considering that all cognitive users are equitable.

Equation (36) ensures that the overall bandwidth request from  $i$  cognitive users should respect the capacity of each primary band  $j$ .

Equation (37) indicates that the accumulated interference of  $i$  cognitive users



at  $j$ -th primary band is restricted by its interference temperature.

Equation (38) indicates that the bandwidth request from the  $i$ -th cognitive user is accepted by assigning a frequency segment in one and only one primary band  $j$ .

Equation (39) means that  $x_{ij}$  is a decision variable, which equals to 1 if the bandwidth request from the  $i$ -th cognitive user is accepted by assigning a frequency segment in the  $j$ -th primary bandwidth. Otherwise,  $x_{ij}$  equals to 0.

Last but not least, we notice that the problem can be bi-criteria whether we consider the least satisfied requests in the past, giving them the priority to be accepted by CCN.

### 3.4. MP-SoC Runtime Management Problem

The ever-increasing performance demands of modern embedded applications become more complex and have several constraints such as memory management, and time constraints. A popular and possible solution is the use of Multi-processor Computer System or MPSoC (MultiProcessor System On Chip). It allows the parallelism of tasks like uploading files and writing text at the same time, and it leads to high-powered computation regarding a uni-processor system.

The MPSoC runs simultaneously with some specifications to complete, for instance runtime, consumed energy, and available platform resources, etc. The literature has focused on runtime management, especially in heterogeneous MPSoC. To manage the runtime decision making and to avoid conservative worst-case assumptions, two-phases can be used. First, we consider a design time exploration per application as a set of possible operating points in a multi-dimensional search space. Finding the optimal solution means mapping the optimal application on the platform. The dimensions of the search space are: Costs (e.g. energy consumption), application constraints (e.g. performance) and platform resources (e.g. communication bandwidth). So, during the first phase called design time, we do a full exploration of the operating points. At this stage we don't know which applications will run simultaneously. Contrary to the second stage, called runtime, the platform resource usage as well as the application user requirements are known. Therefore, we can take the critical decisions in this stage.

The MPSoC runtime management can be modeled as 0/1 MCMKP [33]. The main goal is to minimize the total energy consumption of the platform by taking into account the available platform resources and respecting all application deadlines. The active applications are regarded as the items to be packed in the multidimensional knapsack. The value of items corresponds to the consumed energy per application, while the size of items is represented by the amount of the platform resources required to run an application. Thus, the knapsack corresponds to the platform of resources, and dimensions of knapsack correspond to the type of available platform resources. Before formulating the problem, we need the following notations:

- $s$  applications are active, hence  $s$  sets of operating points are available. Each set  $i$  contains  $N_i$  points.

- $m$  platform resource types are present: For instance, memory space, number of processors, and communication bandwidth.
- For each platform resource type  $k$ ,  $0 \leq k \leq m$ , the available amount is  $R_k$ .
- Each point  $j$  in the set  $i$  characterized by a combination of its energy consumption  $e_{ij}$ , its execution time  $t_{ij}$ , and the used amount  $r_{ijk}$  of the platform resource type  $k$  for  $0 \leq k \leq m$ .

Firstly, the problem has been formulated as a minimization problem, since the goal is to minimize the energy consumption of the platform:

$$\text{Min} \sum_{i=0}^s \sum_{j=0}^{N_i} x_{ij} e_{ij}. \quad (40)$$

where the binary variable  $x_{ij}$  denotes whether the point  $j$  in the set  $i$  is selected ( $x_{ij} = 1$ ) or not ( $x_{ij} = 0$ ).

Then the authors have transformed the minimization problem to maximization problem by using two main steps: Firstly, they remove from each set  $i$  the points whose execution time exceeds the application deadlines. Secondly, they consider each set as an ordered set relatively to the energy consumption axis, that is,  $j \leq j' \Rightarrow e_{ij} \geq e_{ij'}$ . Then, they replace  $e_{ij}$  by the value  $v_{ij} = e_{i0} - e_{ij}$  in the "Equation (40)". Here  $e_{i0}$  means the maximum energy consumption in the active application  $i$  (or set  $i$ ). So,  $v_{ij}$  is a non-negative value,  $v_{ij} \geq 0$ , for  $j = 0, \dots, N_i$ . Note that the minimization problem has transformed to a different form (see [34]).

Finally, the MP-Soc runtime management can be modeled as a maximization problem as following:

$$\left\{ \begin{array}{l} \text{Max} \sum_{i=0}^s \sum_{j=0}^{N_i} x_{ij} v_{ij} \end{array} \right. \quad (41)$$

$$\left\{ \begin{array}{l} \text{s.t.} \sum_{i=0}^s \sum_{j=0}^{N_i} x_{ij} r_{ijk} \leq R_k; \forall k = 0, \dots, m; \end{array} \right. \quad (42)$$

$$\left\{ \begin{array}{l} \sum_{j=0}^{N_i} x_{ij} = 1; \forall i = 0, \dots, s; \end{array} \right. \quad (43)$$

$$\left\{ \begin{array}{l} x_{ij} \in \{0, 1\}; \forall i = 0, \dots, s; j = 0, \dots, N_i; \end{array} \right. \quad (44)$$

Equation (42) indicates that the total used resources may not exceed the available platform resources.

Equation (43) ensures that only one point must be selected from each active set  $i$ .

### 3.5. Capital Budgeting Problem

The capital budgeting problem consists of selecting projects such as investing in R&D or opening a new branch, which are worth pursuing. This problem is a major topic of research and interest in project management. This purely financial problem was among the first applications of the 0/1 MKP in the literature [9]. There is a whole body of literature which makes the link between 0/1 MKP and capital budgeting in order to benefit from their robust approaches. [35] modeled the problem as a 0/1 FMKP, by considering the profit brought by each

project as uncertain and aiming at two key objectives: To minimize the investment cost, and to maximize the project performance. Thus, the selection problem can be translated as follows: The items are represented by the projects to be selected; they are defined by the number of required resources for projects implementation as well as the profit which is brought to the organization. Both features are regarded respectively as size and profit of items. That is, the knapsack dimensions correspond to the type of available resources (e.g. raw material), and the capacity of each dimension is determined by the amount of available resources of each type. We add an additional constraint relating to the overall capacity of knapsack, which is the budget devoted to the project portfolio. Before modeling this problem, we introduce some indices and parameters:

- $j$ : Number of projects,  $j = 1, 2, \dots, n$ .
- $i$ : Type of human resources,  $i = 1, 2, \dots, m$ .
- $k$ : Machine kind,  $k = 1, 2, \dots, s$ .
- $o$ : Type of raw material,  $o = 1, 2, \dots, z$ .
- $\tilde{H}_i$ : Available human resource of type  $i$ .
- $\tilde{h}_{ij}$ : Requirements of human resource  $i$  in project  $j$ .
- $\tilde{M}_k$ : Available machine—hour of type  $k$ .
- $\tilde{m}_{kj}$ : Requirements of machine—hour of type  $k$  in project  $j$ .
- $\tilde{R}_o$ : Maximum available raw material of type  $o$ .
- $\tilde{r}_{oj}$ : Requirements of raw material  $o$  in project  $j$ .
- $\tilde{B}_j$ : Maximum available budget for project  $j$ .
- $\tilde{C}_i$ : Cost of human resource per hour  $i$ .
- $\tilde{C}_k$ : Cost of machine type per hour  $k$ .
- $\tilde{C}_o$ : Unit cost of raw material  $o$ .
- $\tilde{p}_j$ : Total net profit of projects  $j$ .

$$\left\{ \begin{array}{l} \text{Max} \sum_{j=1}^n x_j \left[ p_j - \sum_{i=1}^m \tilde{h}_{ij} \tilde{C}_i - \sum_{k=1}^s \tilde{m}_{kj} \tilde{C}_k - \sum_{o=1}^z \tilde{r}_{oj} \tilde{C}_o \right] \quad (45) \\ \text{s.t} \sum_{i=1}^m \tilde{h}_{ij} x_j \leq \tilde{H}_i; \forall i = 1, \dots, m; \quad (46) \\ \sum_{k=1}^s \tilde{m}_{kj} x_j \leq \tilde{M}_k; \forall k = 1, \dots, s; \quad (47) \\ \sum_{o=1}^z \tilde{r}_{oj} x_j \leq \tilde{R}_o; \forall o = 1, \dots, z; \quad (48) \\ \left( \sum_{i=1}^m \tilde{h}_{ij} \tilde{C}_i + \sum_{k=1}^s \tilde{m}_{kj} \tilde{C}_k + \sum_{o=1}^z \tilde{r}_{oj} \tilde{C}_o \right) x_j \leq \tilde{B}_j; \forall j = 1, \dots, n; \quad (49) \\ \left( \sum_{i=1}^m \tilde{h}_{ij} \tilde{C}_i + \sum_{k=1}^s \tilde{m}_{kj} \tilde{C}_k + \sum_{o=1}^z \tilde{r}_{oj} \tilde{C}_o \right) x_j < \tilde{p}_j; \forall j = 1, \dots, n; \quad (50) \\ \sum_{j=1}^n x_j \geq 1; \quad (51) \\ x_j \in \{0, 1\}; \forall j = 1, 2, \dots, n; \quad (52) \end{array} \right.$$

Equation (45) means that the current problem has more than one objective, it aims to maximize profit of selected projects and minimize their cost in terms of human resources, machines and materials.

Equation (46) indicates that the required human resources for project work, should not exceed the available human resources.

Equations (47), (48) have the same description of “Equation (46)” but they are applied for machine-hour and raw materials, respectively.

Equations (49), (50) ensure that the total cost of a selected project  $j$  is less than its budget and its profit, respectively.

Equation (51) insures that at least one project should be selected.

Equation (52) indicates that  $x_j$  is a binary variable, which denotes whether the project  $j$  is selected ( $x_j = 1$ ) or not ( $x_j = 0$ ).

### 3.6. Real Estate Property Maintenance Problem

The real estate property maintenance problem (REPMP) deals with the problem of the maintenance of real estate property buildings’ components using a limited budget in a limited period to achieve multiple and often conflicting objectives like ensuring the quality of service, client satisfaction and regulatory compliance, and so on. Indeed, maintenance is a key activity for real estate property management. It ensures that building functions are preserved. We can distinguish two types of maintenance: Common maintenance that is dedicated to small operations usually performed according to predefined procedures such as what to do when the elevator is out of service. The second type is planned maintenance that is concerned with heavier operations such as facade renovating, change of heating system or boiler replacement. It is generally subject to specific multi-year action plans. So, the plan of actions is defined as a set of maintenance actions scheduled over several years. The goal of such action plan is to maintain buildings in a good working condition. The REPMP is a constrained complex decision problem. One of the main constraints must not exceed the predetermined budget that is always still insufficient to maintain all buildings in a good condition. The real estate property managers (or decision makers), must find an optimal plan to carry out in a given multiannual time limit in order to achieve the different goals of buildings maintenance. The complexity comes also from the large wide of actions and the need to plan the implementation of these actions over several years. Considering the complexity of REPMP, it has been modeled as a 0/1 MOMKP by [36]. The different years of the plan correspond to the dimension of knapsack, the actions correspond to the items, the cost to the weight of items, and the capacity to the allocated budget to maintain buildings. Before formulating the problem mathematically, we assume that:

- A real estate property is composed of a set of buildings and each building comprises a number of components (electricity, plumbing, etc.),
- An action can impact one or several components, and for each component, several actions can be proposed, and an action can be specific to one building or identical for several or all buildings.
- A score is assigned on each criterion for every component of the real estate property. A scale of four elements is used, from 1 for the worst situation to 4 for the best situation. In other words, low score corresponds to a situation in which a component has a major failure in a given criterion.

- Decision makers define for each type of component its required score to ensure its smooth-running condition.
- The objectives are to minimize the difference between the required score and the assessed score, knowing that this difference is considered null when the assessed score is greater or equal to the required score:

$$\text{Min} \sum_{j=1}^m \sum_{l=1}^p \text{Max} \left( 0; r_k^l - s_{j,l}^k \right)^2, \forall k = 1, \dots, p \quad (53)$$

where  $p$  is the total number of components of the real estate property,  $m$  is the duration in years of the maintenance of real estate property,  $r_k^l$  is the required score of the component  $l$  regarding the objective  $k$  and  $s_{j,l}^k$  is the score of the component  $l$  executed on  $j$ -th year regarding the objective  $k$ .

By modeling REPMP with 0/1 MOMKP, the minimization problem becomes a maximization problem that consists of the selection of the actions maximizing the global yearly profit throughout the duration of the real estate property maintenance while respecting the budget limitation constraints. Let the integer variable  $x_{ij}$ , denotes whether the action  $i$  is executed in the  $j$ -th year of the maintenance plan ( $x_{ij} = 1$ ) or not ( $x_{ij} = 0$ ). So, the problem can be formulated as follows:

$$\left\{ \begin{array}{l} \text{Max } z_k = \sum_{i=1}^n \sum_{j=1}^m c_{ij}^k x_{ij}; \forall k = 1, \dots, p; \end{array} \right. \quad (54)$$

$$\left\{ \begin{array}{l} \text{s.t } \sum_{i=1}^n x_{ij} w_{ij} \leq W_j; \forall j = 1, \dots, m; \end{array} \right. \quad (55)$$

$$\left\{ \begin{array}{l} \sum_{j=1}^m x_{ij} \leq 1; \forall i = 1, \dots, n; \end{array} \right. \quad (56)$$

$$\left\{ \begin{array}{l} x_{ij} \in \{0, 1\}; \forall i = 1, \dots, n; j = 1, \dots, m; \end{array} \right. \quad (57)$$

Equation (54) indicates that the goals are maximizing the profit  $c_{ij}^k$  of an action  $i$  executed in the  $j$ -th year regarding the objective  $k$ . The profit  $c_{ij}^k$  corresponds to the capacity of the action to reduce the difference between the required score and the actual score of a component.

Equation (55) means that the total cost  $w_{ij}$  of the actions  $i$  to execute every year  $j$  must not exceed the allocated yearly budget  $W_j$ .

Equation (56) ensures that each action  $i$ , during the maintenance plan, takes place only once.

#### 4. Heuristic Approaches

Heuristics are good techniques for solving the 0/1 MKP. In fact, two types of heuristics may be distinguished: Greedy heuristics and heuristics based on relaxation. The first class consists of methods based on resolution through steps. The second class consists of approximate procedures based on Linear Programming (LP). The goal in this approach is to produce good solutions in a reasonable amount of time by calculating some parameters (e.g. Lagrangian multipliers, surrogate multipliers).

## 4.1. Greedy Heuristics

Several researchers propose greedy heuristics to solve the 0/1 MKP. [37] uses primal greedy heuristics that sort firstly the items in descending order according to their profits. Starting from null solution, he adds elements into a current solution as long as the solution remains feasible. Also, [26] use a simple greedy heuristic for solving 0/1 MKP. For improving solution quality, they apply Weight-coded Evolutionary Algorithms (see [38]). [39] introduce a new heuristic called “dual greedy heuristic”. They set all the decision variables to ones and convert them to zeros to have a feasible solution. [40] use partial enumeration to solve the 0/1 MKP. They consider only some variables instead of enumerating all the variables. [41] introduce a new greedy heuristic where, at each step, instead of evaluating only one variable at a time they are evaluating a group of variables. The decision variables must be ordered according to an ordering strategy, this method is called “sliding enumeration” and gets good results using a good order. [42] develop a greedy heuristic for the general MKP problem. Their heuristic is adapted for the 0/1 MKP. On small sized 0/1 MKP problems, it reaches very good results in a very short time with problems with 50 - 100 variables and 10 - 200 constraints. However, when the number of variables or constraints increases, it loses its dominance. In general, the more the number of constraints is increasing, the worse heuristics get. But this method remains better than other compared heuristics when the number of constraints is increasing. In addition, [43] use genetic programming (GP) as a hyper-heuristic methodology to generate greedy heuristics to solve 0/1 MKP. Indeed, this hyper-heuristic operates on a search space of heuristics instead of a search space of solutions. It selects and applies a low-level heuristic at each stage of a search. The reached results over a set of standard benchmarks show that GP can be used successfully to generate greedy heuristics. [44] convert a Differential Evolution Algorithm (DEA) by using modified sigmoid function that belongs to binarization techniques [45] (*i.e.* the methods that develop the binary version of a continuous heuristic algorithm). They also combine the obtained DEA with hybrid mutation strategy to solve the 0/1 MKP. Moreover, a greedy strategy is adopted to repair infeasible solutions and select better trail individuals. The outcomes reveal that hybrid DEA is very effective in comparison with other existing methods.

## 4.2. Relaxation-Based Heuristics

### 4.2.1. Continuous Relaxation

[46] improved Soyster’s heuristic [47] that solves a series of small sub-problems generated by exploiting information obtained through a series of LP relaxations. By reducing the number of sub problems and adding constraints to these sub-problems, this improved heuristic yields efficient results for 0/1 MKP. [48] proposed new iterative relaxation-based heuristics for solving 0/1 MCMKP, which generates two sequence of lower and upper bounds. At each iteration, a relaxation is solved to produces an upper bound of the problem and to construct a reduced problem that can be used to obtain a lower bound. The problem is

enriched with pseudo-cuts that eliminate the solutions already visited during the search. Experimentally, this approach reaches rapidly to good lower bounds and visit best solutions in reasonable runtime, by comparing it to CPLEX solver and column generation-based algorithm. Besides, there are other more efficient relaxations: Lagrangian, surrogate and composite relaxations.

#### 4.2.2. Lagrangian Relaxation

In the Lagrangian relaxation, we divide constraints into two categories: Easy and difficult constraints. The idea is to integrate the difficult constraints in the objective function of the problem with a certain penalty to generate an easier problem. [49] introduced a new method based on Lagrangian relaxation. Knowing that not easy to find the optimal Lagrange multipliers, they used a Memetic Algorithm (*i.e.* GA combined with Local Search method) to find them. This approach was evaluated on 270 instances with 5, 10, and 30 constraints, with different numbers of objects and different tightness ratios. The results showed that the proposed method outperforms other constructive heuristics and the local improvement heuristics. [50] proposed a new heuristic algorithm that determines Lagrange multipliers for every constraint in order to reduce the problem to single dimension, then the obtained solutions are improved with iterative procedures. This hybrid algorithm was tested on small size instances of OR-Library Problems [51] and provided high-quality feasible solutions. [52] used the concept of Lagrangian capacity. Making a comparison with other Lagrangian methods, shows that the proposed method performs, especially on large scale data of literature. Moreover, [53] proposed a problem reduction heuristic for solving MKP. In fact, problem reduction involves removing some variables from the problem formulation or at least fixing those variables to some pre-determined values. So, the authors formulate the original problem by using Lagrangian method with dual variables from LP-relaxed problem as Lagrangian multipliers. Then, they exploit the information from this formulation to estimate the core problem so that the non-core variables are fixed to 0 or 1. Based on a benchmark of test problems from the MKP literature, the proposed method performs well in terms of solution quality when comparing with greedy heuristics and problem reduction approaches.

#### 4.2.3. Surrogate Relaxation

The surrogate constraint method translates a multidimensional problem into one dimensional problem using a suitable set of surrogate multipliers:

$$\sum_{i=1}^N \left( \sum_{j=1}^d \mu_j w_{ji} \right) x_i \leq \left( \sum_{j=1}^d \mu_j \right) C_j. \quad (58)$$

where  $\mu = \{\mu_1, \dots, \mu_d\}$  is the set of surrogate multipliers.

[54] proposed a new method for computing suitable surrogate constraints, that allows the user to adjust the quality of the obtained multipliers by means of a parameter  $\varepsilon$ . This method proves the effectiveness in comparing with other methods based on LP and that proposed by Chu and Beasley in [14]. Two heu-

istics based on surrogate constraint are introduced by [55]. In the first one, the relaxed problem is solved via an improved dynamic-programming algorithm. As for the second one, the previous approach is combined with a Branch & Cut procedure. Both approaches yielded better results than other heuristics, with a smaller gap between the best solution and optimal one. [56] was interested in 0/1 FMKP, he defuzzified the problem using triangular norm (t-norm) and t-conorm fuzzy relations. Then he developed the surrogate relaxation for this obtained problem. This methodology is applied to resolve multi-attribute project portfolio selection. [57] proposed a new heuristic for solving 0/1 MCMKP, it is based on oscillation strategy that explores both sides of the feasibility border, by using surrogate constraint information as choice rules.

#### **4.2.4. Composite Relaxation**

Composite relaxation was introduced by Greenberg and Pierskalla [58]. It combines the two previous relaxations that are considered as special cases of this last relaxation. To our knowledge, there is no article that covers the current relaxation in the last decade. However, in their article [59] published in 1994, the authors proved the limit improvement that the composite relaxation (as well as Lagrangian and surrogate relaxations) could bring to 0/1 MKP in comparison with continuous relaxation. [42] discussed the theoretical properties of the MKP, particularly; those relevant to surrogate and composite duality.

### **5. Metaheuristic Approaches**

Several metaheuristics, mostly based on analogies with natural phenomena, were developed and became quite popular for their effectiveness and efficiency to resolve the 0/1 MKP and its variants. We distinguish two categories of metaheuristics: Single-based metaheuristics that consist of modifying and improving a single candidate solution, and population-based metaheuristics that maintain and improve multiple candidate solutions.

#### **5.1. Single-Based Metaheuristics**

##### **5.1.1. Tabu Search**

[60] used the Tabu Search (TS) method for solving 0/1 MCMKP. First, a greedy heuristic is applied for generating a feasible neighbor. The search continues for a fixed number of iterations and TS structures are used to improve the search process. To test this new heuristic, 13 Khan's test problem instances are used [61]. The computational results indicate that the new TS resolves all problems, outperforms some heuristics and overalls others. [62] examined the 0/1 MKP with Generalized Upper Bound constraints (referred as the GUBMKP) using a new TS that adds an adaptive memory structure. Their approach was tested by using the three Li's test cases [63] and led to good outcomes, especially, when it's combined with Local Search (LS) method, it leads to even better results. Likewise, [64] are interested in GUBMKP, they developed a new TS that considers not only the change of objective function value but also the change of feasibility.



Comparing this approach with others, the results showed that it leads to a good compromise between intensification and diversification. [65] proposed a new hybrid tree search algorithm for solving 0/1 MKP that effectively combines TS with a dynamic and adaptive neighborhood search procedure. This heuristic was tested on wide set of benchmark problems and proved its competitiveness compared to other state-of-the-art heuristic methods. [66] used a variation of TS, called the Critical Event Tabu Search (CETS), for solving the General integer MKP (GMKP). A surrogate programming is embedded in CETS as choice rules to obtain high quality solutions.

### 5.1.2. Variable Neighborhood Search

The Variable Neighborhood Search (VNS) is clearly useful for approximate solution of many combinatorial and global optimization problems. But it seems to be slow to solve very large instances. [67] proposed a new heuristic for solving 0/1 MKP, based on the VNS principle. The set of neighborhoods is generated by exploiting information obtained from a series of relaxations. In each iteration, they add new constraints to the problem in order to produce a sequence of lower and upper bounds around the optimal value, with the goal to reduce the gap between them. The experiments yielded promising results, especially on large scale MKP instances, and the outcomes are compared with the state-of-the-art solving methods of 0/1 MKP. [68] proposed a hybrid approach, combining the strengths of VNS and TS. This method is used for solving the web service selection problem which is modeled as 0/1 MCMKP. The results compared with TS and VNS algorithms separately on the same test instances indicated that this heuristic can successfully be used for finding good solutions for relatively large size instances. [69] used a recent version of VNS: Relaxation Guided Variable Neighborhood Search (RGVNS), that follows the standard VNS scheme but uses a new VND algorithm [70]. In their article, they applied the RGVNS and other metaheuristics to approximate cores for 0/1 MKP, as well as in the original problem in order to evaluate the benefits of using a core-based approach. Also [71] used RGVNS and employed the LP techniques. They tested this approach on standard benchmark instances of 0/1 MKP. The computational experiments showed the advantage of the new RGVNS compared to VNS without guidance.

### 5.1.3. Simulated Annealing

[72] propose a new simulated annealing (SA) that incorporates fitness landscape parameters. Their basic idea is to ignore the association between the search space and fitness space as well as to focus only on the comparison between the current solution and the optimal solution. The proposed SA is tested on MKP instances and yields good quality solutions in optimal runtimes when comparing with other variants of SA. On the other hand, [73] present a hybrid simulated annealing method. Firstly, a constructive method based on fitness strategy is used to generate solutions, and another greedy heuristic to search better solutions. Then, SA is applied to improve the quality of solutions. This method is ad-

dressed to deal with rectangle knapsack problem with two-dimensions in which the objects to be packed in the knapsack has all a homogenous form (rectangles). The outcome results on 221 test instances show that the presented method performs better than some previous existing algorithms. Very recently, [74] incorporate the SA and a hybrid heuristic repair strategy in the Pollination Flower Algorithm (PFA) (see [75]) in order to maintain a balance between exploration and exploitation. The proposed algorithm was tested on MKP benchmark data. The results show that it is superior than the Quantum Genetic Algorithm [76], the Binary Particle Swarm Optimization Algorithm and the Binary Cuckoo Search Algorithm (see next sub-sections) in terms of accuracy and robustness.

## 5.2. Population-Based Metaheuristics

### 5.2.1. Genetic Algorithms

[77] are interested in gendered genetic algorithm, in which the gender of chromosomes is considered in the selection operation and only two chromosomes with opposite sex take place in crossover operation. The authors developed in their article some techniques for the sexual selection: The female chromosome is selected by the tournament selection while the male chromosome is selected based on the hamming distance from the selected female chromosome, fitness value or active genes. The results of computational experiment were compared with some well-known selection mechanisms for solving 0/1 MKP. Similarly, [78] aimed at the sexual genetic algorithm with fuzzy data. The authors proposed some techniques for controlling genetic parameters. Indeed, they proposed new sexual selection, crossover, mutation, and probabilities selection techniques by using fuzzy logic controller. Experimentally, the results were compared with other genetic operators, heuristics, and local search algorithms used for solving 0/1 MKP. In [15], the genetic algorithm incorporates greedy heuristics in the repair operator. They combined Chu's information of pseudo-utility ratios [14] and the Raidl's information of the optimal solution about LP of 0/1 MKP [79], in order to sorting variables decreasingly. Comparing to Chu's GA, computational experiments lead to better solutions over 270 standard test problem instances. Moreover, [80] integrate attribute reduction in Rough Set (RS) [81] into crossover operator. The authors apply RS to formulate a set of items. In this case, some redundant items can be generated. So, attribute reduction process can be applied to eliminate them while preserving the feasibility. The authors proposed two kinds of GAs based on attribute reduction in RS. The experiments tested on MKP benchmark data demonstrate that the methodology adopted is a good alternative to improve the performance of GA. In their improved GA, [82] have applied the pattern substitution that replaces the worst genes with excellent ones. Then, they sorted the obtained chromosomes with an improved greedy algorithm. Finally, they used the classical GA to find optimal solutions. This hybrid method yields better results than classical GA. GA combined with greedy algorithm produces robust solutions especially on large sized instances compared with a greedy algorithm only. [83] Introduced a novel algo-

rithm based on a greedy process, a Particle Swarm Optimization (PSO), and some genetic operators. In fact, they used a greedy solution in PSO initial population to make it better. Then, they added crossover and mutation operator to strengthen the diversification strategy. The hybrid algorithm is used to solve small sized instances of 0/1 MKP, and further comparison is needed on large scale instances.

Recently, [84] proposed two Memetic Algorithms (MAs) for solving 0/1 MKP. The first one is combined with a stochastic local search, while the second one is used with SA. Both proposed versions are tested on benchmark data. The experiments showed that they reached competitive results in comparison with well-known hybrid GA based approaches. Nevertheless, [85] analyzed the usefulness of different versions of the Linkage Tree Genetic Algorithm (LTGA) by using Chu's GA [14] as reference. For evaluating the performance of LTGA versions, the authors made a comparison using the 270 MKP instances provided by [14]. The results showed that Chu's GA outperforms all LTGA versions. [86] incorporated separately two methods based on rough sets in crossover operator of GA. The experiment results are compared with classical GAs, they show that both approaches can successfully be used for finding good solutions when the number of items is medium, but they lose their efficiency when the number of items increases.

Very recently, [87] introduced a new hybrid GA, named Guided GA (GGA) that uses information extracted with an efficiency-based method in order to generate the initial population as well as to evaluate the offspring by their fitness function. The GGA is applied to 0/1 MKP and provides competitive solutions in comparison with other optimization methods. In the same way, [88] present a new guided GA, called knowledge-based GA. In fact, the authors apply the primal greedy with the core concept decomposition to extract a useful information about the subset of important items. This information is used to drive the GA search process while generating the initial population and especially for measuring the fitness function. Computational experiments and comparisons reveal that the proposed GA gives competitive solutions. [89] have proposed a new crossover operator that is based on two-stage recombination scheme and generates only one offspring from two parents. In the first stage, the generated offspring inherits the similar genes that have the same position and the same value in both parents. In the second stage, the offspring chromosome is completed by non-similar genes having the same place but different values. The authors have suggested a specific version of their recombination operator for the 0/1 MOMKP. The computational experiments show that this new crossover operator yields promising results in comparison with three traditional crossovers using two well-known multi-objective evolutionary algorithms. Moreover, [90] propose a new repair operator in which the items are added in the increasing order of their values in the relaxed version of the problem. They also use the LP-relaxed solution for generating the new population. Thus, the obtained GA is

combined with a new Neural approach based on the principles of Neural Networks, called Neurogenetic approach. Testing these approaches on MKP instances, the Hybrid method provides better solutions in comparison with the new GA and the Neural approach separately. [91] adapt an approach inspired from multiple-population GAs (or island model GAs) on Cultural Algorithms (CAs) that allows the migration of the best individuals among sub-populations (or islands) and main population through the belief space structures. The experiments have shown that the CA with island model is able to find solutions, for MKP instances, either better or equal to the ones reached by two kinds of algorithms based on Distributed GAs [92]: Distributed canonical GA (DGA) and DGA Self-Reproduction with Mutation (DGA-SRM). Furthermore, [93] proposed a clustered GA that uses a roulette wheel selection with fuzzy technique as a selection strategy and adopt a special crossover operator which uses hierarchical clustering method to form two clusters of selected individuals. Based on 30 test problems from the MKP literature, the proposed GA performs better than classical GA.

### 5.2.2. Ant Colony Optimization

First applications of Ant Colony Optimization (ACO) have been concerned with solving ordering problems, like traveling salesman problem, etc. [94] introduced a new extended ACO to handle 0/1 MKP. The ACO worked very well on several instances and outperformed standard Evolutionary Algorithm on large-scale instances of the problem. [95] proposed an ACO-based algorithm for solving 0/1 MCMKP. The authors combined the traditional version of ACO with Unique Random Local Search (URLS). A comparison with state-of-the-art heuristic algorithms in solving 0/1 MCMKP, shows that the proposed method outperforms others and works successfully on harder instances. Similarly, [96] are interested in 0/1 MCMKP, they developed a hybrid method that combines ACO with Lagrangian relaxation. The obtained solution from Lagrangian relaxation in 0/1 MCMKP is used as heuristic factor in ACO. Moreover, they improved the ACO repair operator. The proposed methods are tested on 43 benchmark instances and compared with four existing algorithms. The experiments provided competitive solutions. Likewise, [97] combined ACO with Lagrangian relaxation, a comparison analysis with ACO and the Lagrangian heuristics separately, shows that the proposed algorithm performs better. [98] introduced a new ACO algorithm based on the Max Min Ant System (MMAS). They also proposed a method to choose the lower trail limit. Then, they combined this method with a local search procedure. They applied their algorithms to the 0/1 MKP. The results show that the proposed algorithms can resolve efficiently the 0/1 MKP.

Besides, [99] proposed a Binary Ant System (BAS) to deal with the 0/1 MKP. This algorithm is different from other ACO-based algorithms applied to 0/1 MKP. Indeed, the BAS uses a pheromone laying method specially designed for the binary solution structure, and it allows the infeasible solutions to take place

in the solution construction procedure. The computational results demonstrate the effectiveness of BAS among other ACO-based approaches for the test problems selected from OR-library. Nevertheless, [100] combined the BAS with Nested Partition (NP) and LP. The new algorithm uses the NP algorithm as global search strategies; uses the BAS to quickly generate good solutions and incorporates information obtained from solving a LP relaxation of 0/1 MKP. This hybrid method outperforms the state-of-the-art solution techniques in terms of quality and computational time. [101] used parallel ACO for solving 0/1 MKP by applying MapReduce parallel programming. The authors also changed some parameters like probability calculation of the runtime, crossover and mutation in order to reduce the ACO complexity. The proposed method is used to solve harder 0/1 MKP instances in cloud computing. Consequently, the results showed that the parallelism of ACO improves its defects of long search time. In the same direction, [102] are interested in parallel ACO, they proposed a parallel implementation under the GPGPU paradigm (General Purpose Graphics Processing Units) using CUDA (for more information we invite the reader to refer to [103]). The results obtained show that parallel ACO is an efficient approach to solve 0/1 MKP, especially for large instances, compared to the other three algorithms, namely; hybrid dynamic programming method with lower bounds computation (HDP + LBC), kernel search algorithm (KS) and NP. Besides, [104] have proposed a new ACO based on some binary quality indicators for guiding the search of artificial ants. Testing their algorithm on MOKP instances, the outcome results show that it is better when comparing with ACO and other evolutionary algorithms. [105] are interested in MCMKP, they present two variations of the ACO algorithm combined with a random local search algorithm for guiding ants to a better area of the search space. The two proposed algorithms improve the solutions generated by ACO and speed up the convergence to near-optimal solutions. The MCMKP's benchmark data from OR-Library are used for testing both algorithms.

### 5.2.3. Particular Swarm Optimization

[106] and [107] introduced a set-based PSO to solve discrete problems, namely; set-based optimization problems like 0/1 MKP. The authors developed a new presentation scheme of the classical PSO by updating the terms of position and velocity. The two proposed set-based PSO algorithms are tested on 0/1 MKP. The experimental results reveal that both algorithms are promising compared to other PSO-based approaches. [108] developed a new hybrid binary PSO, by combining some features of PSO and crossover operation of GA. This proposed algorithm is used to resolve 0/1 MKP. The obtained results show a good and promising solution quality comparing with another version of PSO and a new Cuckoo Search algorithm (CS). [109] propose a generic formulation of the set-based PSO that can be applied to any set-based optimization problem. As parameters tuning, they use the ones of [110]. In order to evaluate the used parameters tuning as well as the performance of the generic set-based PSO, the

authors use the 0/1 MKP as a test problem. The results reveal that the used parameters can be tuned effectively, and the proposed algorithm performs well when comparing with binary PSO and set-based PSO. [111] proposed two novel PSO algorithms to solve 0/1 MKP; the binary PSO with time-varying acceleration coefficients, and the chaotic binary PSO with time-varying acceleration coefficients. The two proposed algorithms are tested using 116 benchmark problems from OR-Library. The results show that both algorithms outperform two other existing PSO-algorithms. Similarly, [112] proposed binary PSO based on the surrogate information with proportional acceleration coefficients for solving 0/1 MKP. The new PSO was tested on 135 benchmark problems from the OR-Library to prove its effectiveness. [113] applied a quantum binary approach that belongs to binarization techniques to PSO. The obtained algorithm is combined with local search method to solve 0/1 MKP, as well as a heuristic based on repair operator that uses problem-specific knowledge. The outcomes of testing the hybrid method on a wide set of benchmark problems prove its competitiveness on other state-of-the-art heuristic methods. [114] introduced a self-adaptive check and repair operator (SACRO) combined with PSO for solving 0/1 MKP. The SACRO use more than one pseudo-utility ratio which changes as the PSO algorithm runs. The SACRO-based algorithm demonstrates its effectiveness in comparison with other PSO-based algorithms. In the same way, [115] proposes a PSO-based algorithm that uses a new SACRO based on three ratios: Surrogate relaxation ratio, profit/weight utility, and profit density. The obtained algorithm is combined with the hill-climbing local search scheme to escape from local optimal solutions. The proposed PSO-based algorithm is tested using OR-library problems. In comparison with the SACRO-based PSO [114], the three-ratio SACRO-based PSO with local search scheme seems to be more competitive and more robust. In the same way, [116] introduce a new self-adaptive repair operator concept with three pseudo-utility ratios that is later incorporated in PSO algorithm. The authors also use a local search scheme to improve the quality of generated solutions. The proposed PSO algorithm is compared with state-of-the-art PSO methods using 168 different widely used MKP benchmark data. However, [117] used the single GPU to accelerate the PSO in terms of computational time for solving 0/1 MKP. The experiments are tested on real test cases proposed by ([14], [118]) and yielded highly competitive results. Otherwise, [119] introduced a new initialization phase as well as the improvement procedure. In order to adapt PSO to 0/1 MKP, a new coding scheme is introduced. [120] developed a new approach that incorporates various features inspired from TS in recent discrete PSO versions called Essential PSO queen (EPSOq), in order to obtain another improved discrete PSO version. This approach leads to encouraging results, especially for large-scale strongly correlated 0/1 MKP instances.

### 5.3. Other Metaheuristic Approaches

[121] proposed a new artificial bee colony (ABC) algorithm for solving the 0/1 MKP. They have integrated ABC algorithm with problem specific heuristics of



0/1 MKP as well as LS. The results demonstrated that their ABC algorithm get nearest optimal solutions and converge rapidly in comparison with other swarm-based approaches. [122] introduced a new communication mechanism among bees, based on the updating and diffusion of pheromone produced by bees. The results obtained by this one produced better solutions in shorter time than the ABC algorithm without the communication mechanism. In the same way, [123] proposed a hybrid algorithm for solving MKP that integrates two models: A new ABC algorithm and a GA tournament selection. The first model consists of an ABC in which honey bees are directed with a multi-agent system. The agents collect solutions of individual bees before than a coordinator agent selects the best solution from all collected solutions. Corresponding to the computational experiments, the proposed algorithm proves that it can produce better solutions than GA, PSO and Glowworm Swarm Optimization (GSO). Other authors have taken advantage of algorithms designed for continuous optimization problems, after having converted to binary algorithms by applying binarization techniques. We cite [124] that have proposed discrete binary bat algorithm (BBA) for solving 0/1 MKP. It is based on the echolocation behavior of microbats. The obtained results are very promising compared to other bio-inspired algorithms. [125] proposed an artificial glowworm swarm optimization (GSO) algorithm to solve 0/1 MKP, combined with two strategies to select items. The results of the proposed algorithm are satisfying. [126] and [127] introduced a binary version of artificial fish swarm algorithm (AFSA or FSA) to efficiently solve the 0/1 MKP. In the same way, [128] proposed a simplified binary version of AFSA, called (S-bAFSA), that uses a random heuristic to obtain a feasible solution and incorporates a local search method to improve quality of solutions. Other strategies are also incorporated in fish behaviors. A comparison of S-bAFSA with other methods available in the literature has been carried out with MKP benchmark data. The authors conclude that the presented method is rather competitive for solving MKP large-scale instances.

[129] present an improved Migrating Birds Optimization (IMBO) to solve MKP. They combine meaningful initial solutions with randomly generated solutions to create a diversity in the swarm. Considering the structure of MKP, they introduce two new crossover operators based on a sharing scheme. The computational results and comparisons with other algorithms indicate that the IMBO is an alternative to solve the MKP, especially for large-scale problems. [130] proposed a binary artificial algae algorithm (AAA) for solving 0/1 MKP. This algorithm is combined with discrete process to achieve good discrete process results, and a repair operator based on pseudo-utility ratio, as well as local search method. In order to verify the robustness of the algorithm, the authors have compared its results with other population-based algorithms: Three versions of PSO-based algorithms, an improved GA, and two versions of FSA. The obtained results proved the superiority of the proposed algorithm. [131] developed an improved Firefly Algorithm (FA) for solving 0/1 MKP, as well as the dynamic

version of MKP (dynMKP) where the items arrive with respect to a certain probability distribution or other sources of dynamism. Standard FA, GA and EA have been chosen as comparative methods. According to the findings of the experiments, authors concluded that the improved FA achieves superior results on both MKP and dynMKP. In the same way, [132] proposed a Quantum-Inspired FA with PSO (QIFAPSO) based on concepts of quantum computing to solve discrete optimization problems. In their algorithm, they use discrete representation for fireflies and propose a variant of Hamming distance to compute the attractiveness between them, as well as other strategies for exploring the search space. The QIFAPSO has been tested on different MKP instances to evaluate its performance. The empirical results reveal that the proposed algorithm turns out to be better than other variants of Quantum Inspired algorithms, particularly when the number of constraints increases. Very recently, [133] introduced an Improved Fruit Fly Optimization Algorithm IFFOA, by using a parallel search to balance exploitation and exploration of search space. The obtained algorithm was combined with Modified Harmony Search algorithm MHS. In MHS, a novel scheme and random selection rule as well as vertical crossover are developed by considering specific characters of 0/1 MKP and FOA. So, The IFFOA demonstrates effectiveness in comparing with other state-of-the-art algorithms. Furthermore, [134] proposed a modified Pollination Flower Optimization (PFO) with GA crossover operator to get out the premature convergence and to increase the diversity of population. Sigmoid function is used to adapt PFO in binary search space. A penalty function is added to fitness function in order to assess the infeasible solutions. That is, a two-stage repair operator is applied to deal with infeasible solutions. The proposed algorithm has been compared with other similar algorithms recently stated in literature. The results show that the new PFO yields, in shorter runtime, better solutions than the comparative methods.

In addition to these bio-inspired metaheuristics, there are other kinds of population-based metaheuristics. In this context, we cite the work developed by [135] who proposed a binary coded version of Harmony Search (HS) metaheuristic for solving large-scale instances of the 0/1 MKP. This proposed HS incorporates an ingenious pitch adjustment scheme without parameter specification and a new repair operator that uses MKP-specific knowledge. More than that, they used the probability distribution instead of exact value of each decision variable. Other single-solution based metaheuristics are proposed for solving 0/1 MKP. That is, [136] developed a novel binary Differential Search Algorithm (DSA) incorporating two different solution strategies. The first strategy consists to find discrete solutions by integrating a Brownian motion-like random search with an integer rounding operation. The second strategy consists to maintain the feasibility of the obtained discrete solutions. The outcome results prove the capability of the proposed DSA to solve large-scale instances of the 0/1 MKP. Indeed, [137] proposed a binary Differential Evolution Algorithm (DEA) by using the sigmoid function that belongs to well-known binarization techniques, and



combine the binary obtained version with VNS to solve 0/1 MKP. Instead of employing check and repair operators, the authors employ some sophisticated constraint handling methods to enrich the population diversity by taking advantages of infeasible solutions within a predetermined threshold. The Computational results of testing the proposed DEA show its efficiency in solving benchmark instances from the OR-Library. Thereby, [138] propose a new binary DEA in which dichotomous mechanisms is fused into crossover and mutation operations. The dichotomous mechanism is inspired from dichotomous thinking of psychology which means the tendency of only seeing extremes, also known as “black and white thinking”. The Dichotomous Binary DEA (DBDEA) is tested on KP and MKP instances and proves its effectiveness and efficiency when comparing with two PSO variants and three DEA variants.

Other authors were inspired by human learning mechanisms, like [139] that introduced novel metaheuristic algorithm called Human Learning Optimization (HLO). In fact, the new algorithm uses the individual learning operator, social learning operator, random exploration learning operator and re-learning operator, in order to generate new solutions. According to experimental results, The HLO outperforms ABC [121], ACO [99], PSO ([106] [114]), as well as NP + BAS + LP [100], and others. However, as the algorithm is new, the authors recommend further research to improve it. [140] are interested in Swarm Electromagnetism-like Mechanism (SEM) based on charges of electrons. They thus proposed a modified Electromagnetism-like Mechanism for working in discrete spaces. For this reason, the GA is used by changing the vector calculations with GA operators. The proposed algorithm is tested on 0/1 MKP, and the experiments results show that it can find best solutions in a little computing time in comparison with stochastic population-based algorithms. [141] applied a modified Scatter Search (SS) for solving 0/1 MKP. In fact, they use a relaxed-based generator to obtain an elite population. According to the outcome results, this new generator clearly guides the SS algorithm to visit elite solutions more quickly. Also, the authors proposed to enhance the SS algorithm by integrating a search memory information. The empirical results show that the proposed SS outperforms other based-population algorithms including GAs. [142] adapt the Greedy Randomized Adaptive Search Procedure (GRASP) [143] to MOMKP by using a new strategy that allows to systematically explore different search directions. The search direction is characterized by the weights associated to each objective. By comparing the new GRASP with other well-known algorithms in multi-objective combinatorial optimization (MOCO), the results show that the proposed algorithm is robust in terms of solution quality and runtimes. Also, [18] have been interested in MOCO problems. They propose a method based on the notion of core that uses the two-Phase Pareto Local Search (2PPLS) [144]. In fact, the 2PPLS requires two elements to be adapted in MOCO problems: An initial population and a neighborhood. Thus, the authors use greedy heuristics to create initial population and Very-Large Scale Neighborhood (VLSN) [145]. The

proposed method is tested on bi-objective instances of MOMKP, and the results prove its effectiveness when comparing with other existing methods of literature, in particular MEMOTS [146]. Contrary, when it's tested on three-objective methods, the authors concluded that 2PPLS consumes a long runtime. So, the MEMOTS turns out to be more efficient when the number of objectives grows.

Moreover, [147] proposed a new SS for solving GUBMKP. They firstly applied GRASP to diversify initial solutions, and then use an algorithm based on the structure of generalized upper bound constraints for selecting the diversified solutions. The results reveal that the obtained SS are competitive with existing solution approaches of GUBMKP. Additionally, [148] developed the Teaching Learning Based-Optimization (TLBO) metaheuristic, based on the relationship between teachers and learners, by introducing a strategy called Teacher Training (TT) before the teaching phase of TLBO. The TT-TLBO is applied on 0/1 MCMKP. The obtained results show that the proposed algorithm outperforms the best published solution approaches for the 0/1 MCMKP. Nevertheless, [149] newly proposed the Estimation of Distribution Algorithm (EDA) to solve 0/1 MKP. It uses the mechanism of probability model based on specific-knowledge problem in order to speed up the CPU time. Then the proposed EDA are combined with LS to enhance the intensification strategy. Very recently, [150] introduce a new variant of Evolutionary Algorithm (EA) called Revised Weight-Coded Evolutionary Algorithm (RWCEA). In the context of MKP, the weight-coding consists to represent a candidate solution by a vector of real-valued weights. This type of solution-representation requires a decoding heuristic, but it allows avoiding the necessity of handling infeasible solutions that are frequently generated when using classical EA operators. The RWCEA is tested on MKP's benchmark and yields better or equal results than other state-of-the-art EAs.

On the other hand, other authors have proposed metaheuristics based on the paradigm of neuronal networks. We cite [151] who showed how domain-specific knowledge can be incorporated within the neural-network framework for solving this NP-Hard problem. Specifically, they developed a new approach named Augmented Neural Networks (AugNN) for solving 0/1 MKP. It combines the heuristic approach with the learning strategy. However, [152] combined this last neural method with GRASP for 0/1 MKP. The hybrid method is implemented in a parallel way using GPGPU with CUDA. The outcomes reveal that the GRASP with a basic operation of AugNN method add variability to the search process and improve the quality of solutions. Thereby, [153] develop an algorithm based on GRASP and Iterated Local Search (ILS) for MOMKP. The computational experiments show that the proposed algorithm outperforms, in shorter consuming time, three algorithms of literature, namely; MOTGA [154], MOGLS [155] and SPEA2 [156]. Furthermore, there are some articles that introduced the application of a promising metaheuristic approaches called Metaheuristics for Randomized Priority Search (Meta-RaPS) [157] to deal with 0/1 MKP. In fact, the Meta-RaPS is a strategy that uses both construction and improvement heuristics to generate high quality solutions. In [158], a modified version of Path Relinking is

incorporated in the improvement phase of Meta-RaPS. The proposed algorithm is tested using the 0/1 MKP and provides good results especially for large scale instances in less time compared to other state-of-the-art algorithms. That is, [159] propose the integration of two other methods into Meta-Raps: EDA and a machine learning algorithm known as Q-Learning. The two proposed algorithms are tested on MKP Benchmark data. The experiments reveal that Meta-Raps with EDA performs well when comparing with Q-Learning, and both algorithms perform better than other existing methods in MKP literature.

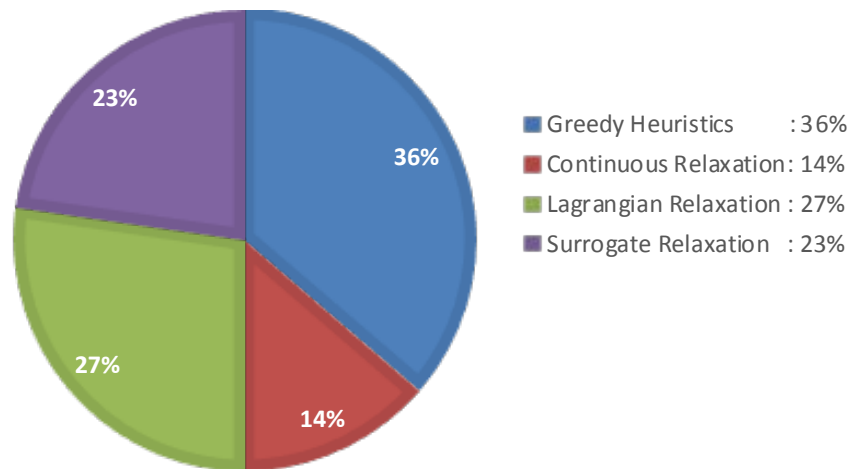
Lately, [160] adopted a multi-agent system for solving MCMKP. The initial problem is decomposed on several sub-problems associated with a coordinator agent and each sub-problem is assigned to an agent with a part of available resources. Each agent is responsible to resolve only one sub-problem. The merging of all feasible solutions giving by each agent provides feasible solution of initial MCMKP. Comparing with existing literature approaches for MCMKP, the empirical experiments show the effectiveness of the agent-based approach. [161] are also interested in MCMKP. So, they proposed a new fuzzy version of MCMKP in which each item may belongs to several groups according to a predefined fuzzy membership value. They modeled the proposed problem as a bi-objective problem by using the epsilon-constraint method [162]. Furthermore, they proposed an improved Partial-Bound Enumeration method (PBE) in which they insert a multi-start property to search the feasible bounded solutions in a parallel way. To compare both proposed methods, three simulated test problems are generated by using a uniform probability density function for small, moderate and large instances. Consequently, the authors conclude that the multi-start PBE method is less time consuming for reaching solutions that are equals, in terms of diversity and accuracy, to the ones obtained by the epsilon-constraint method. [163] introduced a two-step iterative procedure, called Iterative Pseudo Gap Enumeration approach (IPGE), to solve MKP. In the first step, a family of pseudo cuts/constraints is derived from the reduced cost constraints [164] according to a “pseudo-gap” values. Then, in the second step, the original problem with these pseudo cuts is solved by calling an ILP solver (CPLEX). The experiments show that the IPGE outperforms the so-called “reduce and solve” approach on 18 cases of 37 MKP benchmark data. On the other hand, [165] propose an adapted algorithm to solve Layout design problem modeled as a 0/1 MKP. The method uses Charge System Search Algorithm (CSS) classified as a multi-agent approach [166]. The proposed algorithm is compared with ACO algorithm and the results show that the obtained solutions are closely similar to those reached by ACO.

## 6. Some Indicative Statistics

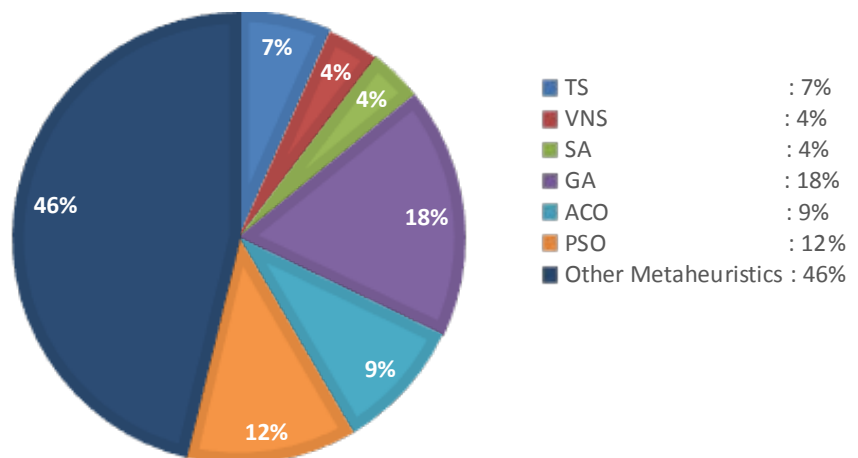
The consulted papers are collected from international scientific journals as well as international conference proceedings during the last decade (2007 to 2017). It is important to note that only papers devoted to 0/1 MKP have been retained. To manage the search process, we have used keywords like: “0/1 Multidimensional

Knapsack Problem”, “Heuristics” and “Metaheuristics”. Google Scholar and Google have been used as search engines. We have paid attention to take articles emanating from different databases like Science Direct, Springer, Scopus, DOAJ and ProQuest. As a result, 102 papers from various publishers are examined attempting to be as exhaustive as possible. The list includes works from these editors: Elsevier, Springer, Taylor & Francis, Wiley, Inderscience, INFORMS, IEEE Press, Emerald, Hindawi and Scientific Research. Some other databases like IEEE Xplore Digital Library and ACM Digital Library are especially used while searching indexed conference papers. However, we apologize for any unintended omission of other relevant articles.

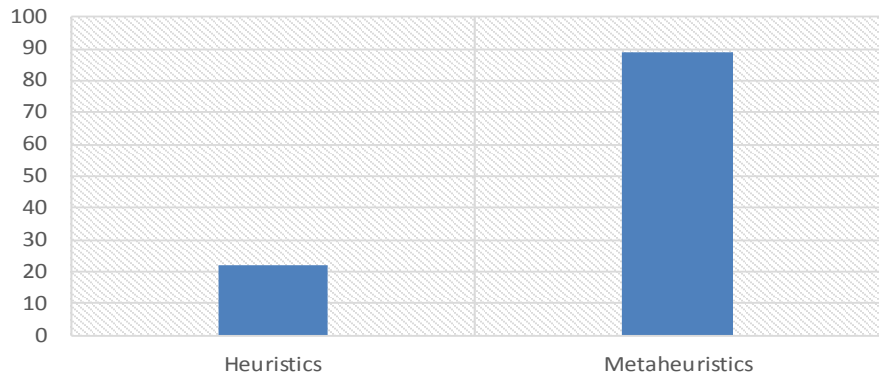
Selecting papers of 0/1 MKP solving methods are depicted in **Figure 1** and **Figure 2**. So, **Figure 1** is devoted to papers addressing heuristics. In **Figure 2**, we consider papers addressing metaheuristics and those that combine metaheuristics with other solving techniques. In **Figure 3**, all solving methods are present in one graph for making a comparison between both categories: Heuristics and metaheuristics.



**Figure 1.** Distribution of 0/1 MKP heuristics.



**Figure 2.** Distribution of 0/1 MKP metaheuristics.



**Figure 3.** Quantitative comparison between heuristics and metaheuristics for 0/1 MKP.

Considering the **Figure 1**, we can see clearly that the most common occurrence was greedy heuristics with eight references, comprising 36 % of all papers addressing heuristics (see **Table 1**), followed by heuristics based on surrogate relaxation with five references (23% of heuristic references). Likewise, heuristics based on Lagrangian relaxation present 27% of heuristic references. Then heuristics based on continuous relaxation with three references (14% of heuristic references). To the best of our knowledge, there is no reference concerning heuristics based on composite relaxation.

In **Figure 2** we have to mention that the most widely used metaheuristics are those based on population of solutions. We have GA with 19 references, comprising 18% of all references addressing metaheuristics to solve 0/1 MKP. Then, PSO represents 12% of metaheuristic references. It is followed by ACO which was used in 10 references (9% of metaheuristic references). In 15 references including articles addressing TS, VNS and SA (7% + 4% + 4% of metaheuristic references) were studied. 49 papers use other methods to deal with 0/1 MKP, comprising 46% of metaheuristic references. The most common were LS with eight references, followed by GRASP with three references, as well as ABC, FSA, Weight-coded EA, SS, and Meta-Raps with two references.

The following conclusions can be drawn from the **Figure 3**: The heuristics are studied in 22 papers, comprising almost 20% of all references given in the **Table 1**, contrary to the metaheuristics that remained the most common methods for solving 0/1 MKP, studied in 89 references (80% of the list). This is due to the fact that the 0/1 MKP is considered as a test problem in literature. So, if a metaheuristic has been developed for the 0/1 MKP, and it proved a high-quality solution by using a little CPU time, it will be promising for other different problems by applying minor modifications.

**Table 1** summarizes our literature review for resolution methods of 0/1 MKP. Papers addressing different algorithms of 0/1 MKP are listed. Two types of methods are exhibited in the first level: Heuristics and metaheuristics. Therefore, the sublevels of the heuristics level are greedy heuristics (GH), and Relaxation-based heuristics with sublevels, that are heuristics based on continuous or Lagrangian or surrogate relaxations. The metaheuristics level is divided into

**Table 1.** State-of-the-art heuristic and metaheuristic methods for solving 0/1 MKP.

References		Heuristics & Metaheuristics									
		Heuristics					Metaheuristics				
		Greedy Heuristics	Relaxed-based Heuristics			Single-based Metaheuristics		Population-based Metaheuristics			
	CR	LR	SR	TS	VNS	SA	GA	ACO	PSO	Other Metaheuristics	
2007	J. Pfeiffer & F. Rothlauf	x								Weight-Coded EA	
2007	A. Volgenant & I. Y. Zwiers	x									
2007	Y. Akcay <i>et al.</i>										
2007	M. J. Alves & M. M. Almeida						x				
2008	S. Hanafi & C. Wilbaut									SS	
2008	J. Puchinger <i>et al.</i>					x					
2008	M. Kong <i>et al.</i>							x			
2008	V. C. Li				x						
2008	J. L. Montaña & C. L. Alonso			x							
2009	B. Aghezzaf & M. Naimi						x				
2009	V. Boyer <i>et al.</i>			x						DP + B & C	
2009	S. Hanafi <i>et al.</i>					x					
2009	M. H. Akin	x									
2009	S. Hanafi & C. Wilbaut		x								
2010	H. H. Yang <i>et al.</i>						x				
2010	S. Iqbal <i>et al.</i>							x		Unique Random LS	
2010	L. Ke <i>et al.</i>							x		LS	
2010	S. Al-Shihabi & S. Ólafsson							x		NP + LP	
2010	S. Sundar <i>et al.</i>									ABC	
2010	W. N. Chen <i>et al.</i>								x		
2010	T. Lust & J. Teghem									Pareto LS	
2010	S. Iqbal <i>et al.</i>							x			
2011	J. Langeveld & A. P. Engelbrecht								x		
2011	Q. Gong <i>et al.</i>									GSO	
2012	M. J. Varnamkhasti & L. S. Lee							x			
2012	M. J. Varnamkhasti & L. S. Lee							x			
2012	C. Atilgan & U. Nuriyev			x							
2012	R. R. Hill <i>et al.</i>			x							
2012	Y. Yoon & Y. H. Kim			x							
2012	V. C. Li <i>et al.</i>				x						
2012	M. Khemakhem <i>et al.</i>				x						
2012	N. Turajlić & I. Dragović				x	x					
2012	Z. Ren <i>et al.</i>			x					x		

## Continued

2012	J. Langeveld & A. P. Engelbrecht					x	
2012	A. Gherboudj <i>et al.</i>				x	x	
2012	M. A. K. Azad <i>et al.</i>						FSA
2012	M. R. Bonyadi & X. Li						SEM
2012	V. C. Li <i>et al.</i>						SS + GRASP
2012	J. Deane & A. Agarwal						Aug NN
2012	E. Bas			x			
2012	I. Crévits <i>et al.</i>	x					
2012	L. Wang <i>et al.</i>						EDA + LS
2012	D. J. A. Da Silva <i>et al.</i>				x		CA
2012	S. Leung <i>et al.</i>					x	
2013	Y. Yoon & Y. H. Kim		x			x	LS
2013	S. Htiouech <i>et al.</i>			x			
2013	C. S. Hiremath & R. R. Hill	x			x		
2013	R. Ktari & H. Chabchoub			x			x
2013	R. T. Liu & X. J. Lv					x	
2013	J. Ji <i>et al.</i>						ABC
2013	S. Abusini & A. Andari	x				x	x
2013	J. Deane & A. Agarwal				x		Neural method
2013	D. S. Vianna & M. D. F. D. Vianna						LS
2013	H. Yang <i>et al.</i>					x	
2013	K. Khalili-Damghani <i>et al.</i>						PBE + Epsilon constraint method
2014	M. Chih						x
2014	A. Baykasoğlu & F. B. Ozsoydan						FA
2014	D. S. Vianna <i>et al.</i>						GRASP
2014	J. H. Drake <i>et al.</i>	x					GP
2014	J. P. Martins <i>et al.</i>					x	
2014	M. A. K. Azad <i>et al.</i>						FSA
2014	H. Fingler <i>et al.</i>					x	
2014	S. Sabba & S. Chikhi						BAA
2014	G. Lai <i>et al.</i>	x				x	
2014	X. Liu <i>et al.</i>	x				x	
2014	D. Zan & J. Jaros						x
2014	M. Chih <i>et al.</i>						x

**Continued**

2015	A. Rezoug <i>et al.</i>			x	x		LS
2015	M. Chih					x	SACRO
2015	X. Zhang <i>et al.</i>						HS
2015	M. F. Tasgetiren <i>et al.</i>			x			DEA
2015	W. Nakbi <i>et al.</i>	x				x	
2015	K. Kakryal & K. Jammu				x		
2015	L. Wang <i>et al.</i>						HLO
2015	M. A. K. Azad <i>et al.</i>						FSA
2015	I. B. Mansour & I Alaya					x	
2015	Z. Kenneth <i>et al.</i>						TT-TLBO
2016	Q. Yuan & Z. X. Yang						Weight-Coded EA
2016	B. De Almeida Dantas & E. N. Cáceres						Aug NN + GRASP
2016	A. Arin & G. Rabadi						Path Relinking + Meta-Raps
2016	B. Haddar <i>et al.</i>					x	
2016	X. Zhang <i>et al.</i>						AAA
2016	J. Liu <i>et al.</i>						DFA
2016	C. J. Lin <i>et al.</i>		x			x	
2016	D. Libao <i>et al.</i>	x					
2016	D. Zouache <i>et al.</i>						FA
2016	H. Peng <i>et al.</i>						DEA
2017	B. Alidaee <i>et al.</i>		x				
2017	T. Meng & Q. K. Pan						FFOA
2017	Z. Fang <i>et al.</i>				x		FPA
2017	A. V. Bole & R. Kumar						ABC
2017	A. Rezoug <i>et al.</i>					x	
2017	A. Rezoug <i>et al.</i>					x	
2017	I. K. Gupta <i>et al.</i>					x	
2017	T. Meng <i>et al.</i>						MBO
2017	M. Abdel-Basset <i>et al.</i>						FPA
2017	A. Arin & G. Rabadi						EDA + Q-Learning + Meta-Raps
2017	S. Htiouech & A. Alzaidi						System Multi-Agent (SMA)
2017	M. Chih					x	SACRO + Hill Climbing + LS
2017	C. Gao <i>et al.</i>						Enumeration method
2017	P. Sharafi <i>et al.</i>						CSS



two categories: Single-based algorithms and Population-based heuristics. The first sublevel includes TS, VNS and SA algorithms, while the second contains GA, ACO, PSO and other algorithms including hybrid methods. For an early literature before 2007, the reader is referred to [42].

## 7. Conclusions

Creating survey is an efficient and effective way of consolidating knowledge. It enables storage, sorting and statistical analysis. This article aims to build a relevant survey of the most useful variants of 0/1 MKP. It attempts to highlight some real-world applications encountered in the literature. Furthermore, it outlines a state-of-the-art of the most common heuristic methods, for solving 0/1 MKP. However, we cannot claim this survey to be exhaustive. The analysis carried out is based on a large number of literature references. From this analysis, we can see that articles related to 0/1 MKP continue to be published in a large number of research journals having a different scope. That is, on the one hand, our problem is considered as an open research issue and consequently as a test problem for a major metaheuristic. On the other hand, it can model a large wide of real-world problems and it can be considered also as a sub-problem of many other real-world problems.

We have noticed that the 0/1 MKP has several extensions in order to adapt it straightforwardly in different real problems. In our paper, we have cited six variants; four are extended versions of deterministic 0/1 MKP, and two belong to non-deterministic 0/1 MKP extensions. A number of challenging practical applications can be modeled as a variant of the 0/1 MKP. We have discussed the capital budgeting problem, the combinatorial auctions, allocation resources with stochastic demands, frequency allocation in cognitive radio networks, MP-SoC runtime management problem, and real estate property maintenance problem. These problems can benefit from the 0/1 MKP solving methods.

In terms of methods used to solve the 0/1 MKP variants, constructive heuristics based on specific-knowledge problem appear to be among the most popular approaches. But we notice that metaheuristics are more and more reported in recent articles since they track a trade-off between the solution quality and the CPU time, especially for large-size 0/1 MKP instances. The single based-population algorithms are clearly useful to find a good solution for the 0/1 MKP. However, they encounter some difficulties to solve very large instances. For this reason, they are seldom published. The based-population algorithms have attracted much attention from many researchers, in particular GA, PSO and ACO. The methods are also hybridized, either to provide high-quality solution (e.g. metaheuristics embedding greedy heuristics or combined with other metaheuristics) or to speed-up the computing time (e.g. combination of metaheuristics with parallelization strategies). However, we still need a deeper investigation in the hybrid metaheuristics field in order to have clear ideas about suitable and efficient hybridization techniques in the context of 0/1 MKP. Readers interested in this

promising topic can refer to [167] for a general and detailed dissertation in the context of combinatorial optimization. Very recently, most of authors try to adapt continuous metaheuristics to work in binary search spaces methods by using binarization techniques.

We believe this paper has helped to unify the rapidly expanding body of knowledge on the 0/1 MKP. We hope also that this work will encourage other researchers to pursue the study of this fascinating field of research.

## Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

## References

- [1] Martello, S. and Toth, P. (1990) Knapsack Problems: Algorithms and Computer Implementations. John Wiley & Sons Ltd., Chichester.
- [2] Skiena, S. (1999) Who Is Interested in Algorithms and Why? Lessons from the Stony Brook Algorithms Repository. *ACM SIGACT News*, **30**, 65-74. <https://doi.org/10.1145/333623.333627>
- [3] Eilon, S. and Christofides, N. (1971) The Loading Problem. *Management Science*, **17**, 259-268. <https://doi.org/10.1287/mnsc.17.5.259>
- [4] Hifi, M. (2009) Approximate Algorithms for the Container Loading Problem. *Operations Research*, **9**, 747-774.
- [5] Gilmore, P.C. and Gomory, R.E. (1966) The Theory and Computation of Knapsack Functions. *Operations Research*, **14**, 1045-1074. <https://doi.org/10.1287/opre.14.6.1045>
- [6] Labbé, M., Laporte, G. and Martello, S. (2003) Upper Bounds and Algorithms for the Maximum Cardinality Bin Packing Problem. *European Journal of Operational Research*, **149**, 490-498. [https://doi.org/10.1016/S0377-2217\(02\)00466-6](https://doi.org/10.1016/S0377-2217(02)00466-6)
- [7] Petersen, C.C. (1967) Computational Experience with Variants of the Balas Algorithm Applied to the Selection of R&D Projects. *Management Science*, **3**, 736-750. <https://doi.org/10.1287/mnsc.13.9.736>
- [8] Weingartner, H.M. (1966) Capital Budgeting of Interrelated Projects: Survey and Synthesis. *Management Science*, **12**, 485-516. <https://doi.org/10.1287/mnsc.12.7.485>
- [9] Lorie, J.H. and Savage, L.J. (1955) Three Problems in Rationing Capital. *The Journal of Business*, **28**, 229-239. <https://doi.org/10.1086/294081>
- [10] Markowitz, H.M. and Manne, A.S. (1957) On the Solution of Discrete Programming Problems. *Econometrica Journal of the Econometric Society*, **25**, 84-110. <https://doi.org/10.2307/1907744>
- [11] Kellerer, H., Pferschy, U. and Pisinger, D. (2004) Knapsack Problems. Springer, Berlin. <https://doi.org/10.1007/978-3-540-24777-7>
- [12] Freville, A. (2004) The Multidimensional 0-1 Knapsack Problem: An Overview. *European Journal of Operational Research*, **155**, 1-21. [https://doi.org/10.1016/S0377-2217\(03\)00274-1](https://doi.org/10.1016/S0377-2217(03)00274-1)
- [13] Gavish, B. and Pirkul, H. (1965) Efficient Algorithms for Solving Multiconstraint Zero-One Knapsack Problems to Optimality. *Mathematical Programming*, **31**, 78-105. <https://doi.org/10.1007/BF02591863>

- [14] Chu, P.C. and Beasley, J.E. (1998) A Genetic Algorithm for the Multidimensional Knapsack Problem. *Journal of heuristics*, **4**, 63-86. <https://doi.org/10.1023/A:1009642405419>
- [15] Lai, G., Yuan, D. and Yang, S. (2014) A New Hybrid Combinatorial Genetic Algorithm for Multidimensional Knapsack Problems. *Journal of Supercomputing*, **70**, 930-945. <https://doi.org/10.1007/s11227-014-1268-9>
- [16] Varnamkhasti, M.J. (2012) Overview of the Algorithms for Solving the Multidimensional Knapsack Problems. *Advanced Studies in Biology*, **4**, 37-47.
- [17] Puchinger, J., Raidl, G.R. and Pferschy, U. (2010) The Multidimensional Knapsack Problem: Structure and Algorithms. *INFORMS Journal on Computing*, **22**, 250-265. <https://doi.org/10.1287/ijoc.1090.0344>
- [18] Lust, T. and Teghem, J. (2012) The Multiobjective Multidimensional Knapsack Problem: A Survey and a New Approach. *International Transactions in Operational Research*, **19**, 495-520. <https://doi.org/10.1111/j.1475-3995.2011.00840.x>
- [19] Lienland, B. and Zeng, L.A. (2015) Review and Comparison of Genetic Algorithms for the 0-1 Multidimensional Knapsack Problem. *International Journal of Operations Research and Information Systems*, **6**, 21-31. <https://doi.org/10.4018/ijoris.2015040102>
- [20] Alonso, C.L., Caro, F. and Montaña, J.L. (2006) A Flipping Local Search Genetic Algorithm for the Multidimensional 0-1 Knapsack Problem. *Lecture Notes in Computer Science*, **4177**, 21-30. [https://doi.org/10.1007/11881216\\_3](https://doi.org/10.1007/11881216_3)
- [21] Dyer, M.E., Riha, W.O. and Walker, J. (1995) A Hybrid Dynamic Programming/Branch-and-Bound Algorithm for the Multiple-Choice Knapsack Problem. *Journal of Computational and Applied Mathematics*, **58**, 43-54. [https://doi.org/10.1016/0377-0427\(93\)E0264-M](https://doi.org/10.1016/0377-0427(93)E0264-M)
- [22] Gallardo, J. E., Cotta, C. and Fernández, A.J. (2005) Solving the Multidimensional Knapsack Problem Using an Evolutionary Algorithm Hybridized with Branch and Bound. *Lecture Notes in Computer Science*, **3562**, 21-30. [https://doi.org/10.1007/11499305\\_3](https://doi.org/10.1007/11499305_3)
- [23] Jourdan, L., Basseur, M. and Talbi, E.G. (2009) Hybridizing Exact Methods and Metaheuristics: A Taxonomy. *European Journal of Operational Research*, **199**, 620-629. <https://doi.org/10.1016/j.ejor.2007.07.035>
- [24] Schulze, B. (2017) New Perspectives on Multi-Objective Knapsack Problems. Ph.D. Thesis, University of Wuppertal, Germany. <http://elpub.bib.uni-wuppertal.de/servlets/DerivateServlet/Derivate-6701/dc1706.pdf>
- [25] Kunikazu, Y. and Prékopa, A. (2012) Convexity and Solutions of Stochastic Multidimensional Knapsack Problems with Probabilistic Constraints. Research report. <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.304.5298>
- [26] Pfeiffer, J. and Rothlauf, F. (2007) Analysis of Greedy Heuristics and Weight-Coded EAs for Multidimensional Knapsack Problems and Multi-Unit Combinatorial Auctions. *Proceedings of the 9th Annual Conference on Genetic and Evolutionary Computation (GECCO-2007)*, ACM, London, 7-11 July 2007, 1529-1529.
- [27] Holte, R.C. (2001) Combinatorial Auctions, Knapsack Problems, and Hill-Climbing Search. *Proceedings of the 14th Conference of the Canadian Society for Computational Studies of Intelligence (AI2001)*, Ottawa, 7-9 June 2001, 57-66.
- [28] Kelly, T. (2004) Generalized Knapsack Solvers for Multi-Unit Combinatorial Auctions: Analysis and Application to Computational Resource Allocation. *Proceedings of the International Workshop on Agent Mediated-Electronic Commerce VI: Theo-*

- ries for and Engineering of Distributed Mechanisms and Systems (AMEC 2004), New York, NY, 19 July 2004, 73-86.
- [29] Chen, F., La Porta, T. and Srivastava, M.B. (2012) Resource Allocation with Stochastic Demands. *Proceedings of the IEEE 8th International Conference on Distributed Computing in Sensor Systems (DCOSS 2012)*, Hangzhou, 16-18 May 2012, 257-264.
- [30] Luedtke, J. and Ahmed, S. (2008) A Sample Approximation Approach for Optimization with Probabilistic Constraints. *SIAM Journal on Optimization*, **19**, 674-699. <https://doi.org/10.1137/070702928>
- [31] Mitola, J. and Maguire, G. (1999) Cognitive Radio: Making Software Radios More Personal. *IEEE Personal Communications*, **6**, 13-18. <https://doi.org/10.1109/98.788210>
- [32] Song, Y., Zhang, C. and Fang, Y. (2008) Multiple Multidimensional Knapsack Problem and Its Applications in Cognitive Radio Networks. *Proceedings of the IEEE Military Communications Conference (MILCOM 2008)*, San Diego, 16-19 November 2008, 1-7.
- [33] Ykman-Couvreur, C., Nollet, V., Catthoor, F. and Corporaal, H. (2011) Fast Multi-dimension Multichoice Knapsack Heuristic for MP-SoC Runtime Management. *ACM Transactions on Embedded Computing Systems (TECS)*, **10**, Article No. 35. <https://doi.org/10.1145/1952522.1952528>
- [34] Mejia-Alvarez, E., Levner, P. and Mosse, D. (2002) Power-Optimized Scheduling Server for Real-Time Tasks. *Proceedings of the IEEE 8th Real-Time and Embedded Technology and Applications Symposium (RTAS 2002)*, San Jose, 25-27 September 2002, 239-250.
- [35] Khalili-Damghani, K. and Taghavifard, M. (2011) Solving a Bi-Objective Project Capital Budgeting Problem Using a Fuzzy Multi-Dimensional Knapsack. *Journal of Industrial Engineering International*, **7**, 67-73.
- [36] Taillandier, F., Fernandez, C. and Ndiaye, A. (2017) Real Estate Property Maintenance Optimization Based on Multiobjective Multidimensional Knapsack Problem. *Computer-Aided Civil and Infrastructure Engineering*, **32**, 227-251. <https://doi.org/10.1111/mice.12246>
- [37] Toyoda, Y. (1975) A Simplified Algorithm for Obtaining Approximate Solutions to Zero-One Programming Problems. *Management Science*, **21**, 1417-1427. <https://doi.org/10.1287/mnsc.21.12.1417>
- [38] Raidl, G.R. (1999) Weight-Codings in a Genetic Algorithm for the Multiconstraint Knapsack Problem. *Proceedings of the IEEE Congress on Evolutionary Computation (CEC 1999)*, Washington DC, 6-9 July 1999, 596-603.
- [39] Senju, S. and Toyoda, Y. (1968) An Approach to Linear Programming with 0-1 Variables. *Management Science*, **15**, 196-207. <https://doi.org/10.1287/mnsc.15.4.B196>
- [40] Volgenant, A. and Zwiers, I.Y. (2007) Partial Enumeration in Heuristics for Some Combinatorial Optimization Problems. *Journal of Operational Research Society*, **58**, 73-79. <https://doi.org/10.1057/palgrave.jors.2602102>
- [41] Akin, M.H. (2009) New Heuristics for the 0-1 Multi-Dimensional Knapsack Problems. Ph.D. Thesis, University of Central Florida, USA. <http://purl.fcla.edu/fcla/etd/CFE0002633>
- [42] Akcay, Y., Li, H. and Xu, S.H. (2007) Greedy Algorithm for the General Multidimensional Knapsack Problem. *Annals of Operations Research*, **150**, 17-29. <https://doi.org/10.1007/s10479-006-0150-4>

- [43] Drake, J.H., Hyde, M., Ibrahim, K. and Ozcan, E. (2014) A Genetic Programming Hyper-Heuristic for the Multidimensional Knapsack Problem. *Kybernetes*, **43**, 1500-1511. <https://doi.org/10.1108/K-09-2013-0201>
- [44] Libao, D., Sha, W., Chengyu, J. and Cong, H. (2016) A Hybrid Mutation Scheme-Based Discrete Differential Evolution Algorithm for Multidimensional Knapsack Problem. *Proceedings of the IEEE 6th International Conference on Instrumentation & Measurement, Computer, Communication and Control (IMCCC 2016)*, Harbin, 21-23 July 2016, 1009-1014.
- [45] Crawford, B., Soto, R., Astorga, G., García, J., Castro, C. and Paredes, F. (2017) Putting Continuous Metaheuristics to Work in Binary Search Spaces. *Complexity*, **2017**, 19. <https://doi.org/10.1155/2017/8404231>
- [46] Hanafi, S. and Wilbaut, C. (2009) Improved Convergent Heuristics for the 0-1 Multidimensional Knapsack Problem. *Annals of Operations Research*, **183**, 125-142. <https://doi.org/10.1007/s10479-009-0546-z>
- [47] Soyster, A.L., Lev, B. and Slivka, W. (1978) Zero-One Programming with Many Variables and Few Constraints. *European Journal of Operational Research*, **2**, 195-201. [https://doi.org/10.1016/0377-2217\(78\)90093-0](https://doi.org/10.1016/0377-2217(78)90093-0)
- [48] Crévits, I., Hanafi, S., Mansi, R. and Wilbaut, C. (2012) Iterative Semi-Continuous Relaxation Heuristics for the Multiple-Choice Multidimensional Knapsack Problem. *Computers & Operations Research*, **39**, 32-41. <https://doi.org/10.1016/j.cor.2010.12.016>
- [49] Yoon, Y. and Kim, Y.H. (2013) A Memetic Lagrangian Heuristic for the 0-1 Multidimensional Knapsack Problem. *Discrete Dynamics in Nature and Society*, **2013**, 10. <https://doi.org/10.1155/2013/474852>
- [50] Atilgan, C. and Nuriyev, U. (2012) Hybrid Heuristic Algorithm for the Multidimensional Knapsack Problem. *Proceedings of the IEEE 4th International Conference on Problems of Cybernetics and Informatics*, Baku, 12-14 September 2012, 1-4.
- [51] Beasley, J.E. (1990) OR-Library: Distributing Test Problems by Electronic Mail. *Journal of the Operational Research Society*, **41**, 1069-1072. <https://doi.org/10.1057/jors.1990.166>
- [52] Yoon, Y. and Kim, Y.H. (2012) A Theoretical and Empirical Investigation on the Lagrangian Capacities of the 0-1 Multidimensional Knapsack Problem. *European Journal of Operational Research*, **218**, 366-376. <https://doi.org/10.1016/j.ejor.2011.11.011>
- [53] Hill, R.R., Cho, Y.K. and Moore, J.T. (2012) Problem Reduction Heuristic for the 0-1 Multidimensional Knapsack Problem. *Computers & Operations Research*, **39**, 19-26. <https://doi.org/10.1016/j.cor.2010.06.009>
- [54] Montaña, J.L., Alonso, C.L., Cagnoni, S. and Callau, M. (2008) Computing Surrogate Constraints for Multidimensional Knapsack Problems Using Evolution Strategies. *Proceedings of the Workshops on Applications of Evolutionary Computation*, Naples, 26-28 March 2008, 555-564.
- [55] Boyer, V., Elkihel, M. and El Baz, D. (2009) Heuristics for the 0-1 Multidimensional Knapsack Problem. *European Journal of Operational Research*, **199**, 658-664. <https://doi.org/10.1016/j.ejor.2007.06.068>
- [56] Bas, E. (2012) Surrogate Relaxation of a Fuzzy Multidimensional 0-1 Knapsack Model by Surrogate Constraint Normalization Rules and a Methodology for Multi-Attribute Project Portfolio Selection. *Engineering Applications of Artificial Intelligence*, **25**, 958-970. <https://doi.org/10.1016/j.engappai.2011.09.015>
- [57] Htiouech, S., Bouamama, S. and Attia, R. (2013) Using Surrogate Information to

- Solve the Multidimensional Multi-Choice Knapsack Problem. *Proceedings of the IEEE Congress on Evolutionary Computation*, Cancun, 20-23 June 2013, 2102-2107.
- [58] Greenberg, H.J. and Pierskalla, W.P. (1970) Surrogate Mathematical Programming. *Operations Research*, **18**, 924-939. <https://doi.org/10.1287/opre.18.5.924>
- [59] Crama, Y. and Mazzola, J.B. (1994) On the Strength of Relaxations of Multidimensional Knapsack Problems. *INFOR: Information Systems and Operational Research*, **32**, 219-225. <https://doi.org/10.1080/03155986.1994.11732252>
- [60] Hiremath, C.S. and Hill, R.R. (2013) First-Level Tabu Search Approach for Solving the Multiple-Choice Multidimensional Knapsack Problem. *International Journal of Metaheuristics*, **2**, 174-199. <https://doi.org/10.1504/IJMHEUR.2013.054150>
- [61] Khan, S., Li, K.F., Manning, E.G. and Akbar, M. (2002) Solving the Knapsack Problem for Adaptive Multimedia Systems. *Studia Informatica*, **2**, 157-178.
- [62] Li, V.C., Liang, Y.C. and Chang, H.F. (2012) Solving the Multidimensional Knapsack Problems with Generalized Upper Bound Constraints by the Adaptive Memory Projection Method. *Computers & Operations Research*, **39**, 2111-2121. <https://doi.org/10.1016/j.cor.2011.10.016>
- [63] Li, V.C., Curry, G.L. and Boyd, E.A. (2004) Towards the Real Time Solution of Strike Force Asset Allocation Problems. *Computers & Operations Research*, **31**, 273-291. [https://doi.org/10.1016/S0305-0548\(02\)00192-2](https://doi.org/10.1016/S0305-0548(02)00192-2)
- [64] Li, V.C. (2008) A Perturbation-Guided Oscillation Strategy for the Multidimensional Knapsack Problem with Generalized Upper Bound Constraints. *Proceedings of the 7th International Symposium on Operations Research and Its Applications*, Lijiang, 31 October-3 November 2008, 369-376.
- [65] Khemakhem, M., Haddar, B., Chebil, K. and Hanafi, S. (2012) A Filter-and-Fan Metaheuristic for the 0-1 Multidimensional Knapsack Problem. *International Journal of Applied Metaheuristic Computing*, **3**, 43-63. <https://doi.org/10.4018/jamc.2012100103>
- [66] Alidaee, B., Ramalingam, V.P., Wang, H. and Kethley, B. (2017) Computational Experiment of Critical Event Tabu Search for the General Integer Multidimensional Knapsack Problem. *Annals of Operations Research*, **269**, 17.
- [67] Hanafi, S., Lazić, J., Mladenović, N. and Wilbaut, C. (2009) Variable Neighbourhood Decomposition Search with Bounding for Multidimensional Knapsack Problem. *IFAC Proceedings Volumes*, **42**, 2018-2022. <https://doi.org/10.3182/20090603-3-RU-2001.0502>
- [68] Turajlić, N. and Dragović, I. (2012) A Hybrid Metaheuristic Based on Variable Neighborhood Search and Tabu Search for the Web Service Selection Problem. *Electronic Notes in Discrete Mathematics*, **39**, 145-152. <https://doi.org/10.1016/j.endm.2012.10.020>
- [69] Puchinger, J., Raidl, G.R. and Pferschy, U. (2006) The Core Concept for the Multidimensional Knapsack Problem. *Proceedings of the 6th European Conference on Evolutionary Computation in Combinatorial Optimization*, Budapest, 10-12 April 2006, 195-208. [https://doi.org/10.1007/11730095\\_17](https://doi.org/10.1007/11730095_17)
- [70] Hansen, P. and Mladenovic, N. (2005) Variable Neighborhood Search. In: Burke E.K. and Kendall, G., Eds., *Search Methodologies*, Springer, Berlin, 211-238. [https://doi.org/10.1007/0-387-28356-0\\_8](https://doi.org/10.1007/0-387-28356-0_8)
- [71] Puchinger, J. and Raidl, G.R. (2008) Bringing Order into the Neighborhoods: Relaxation Guided Variable Neighborhood Search. *Journal of Heuristics*, **14**, 457-472. <https://doi.org/10.1007/s10732-007-9048-9>



- [72] Gupta, S. and Arora, S. (2015) Boosting Simulated Annealing with Fitness Landscape Parameters for Better Optimality. *International Journal of Computing*, **14**, 107-112.
- [73] Leung, S.C., Zhang, D., Zhou, C. and Wu, T. (2012) A Hybrid Simulated Annealing Metaheuristic Algorithm for the Two-Dimensional Knapsack Packing Problem. *Computers & Operations Research*, **39**, 64-73.  
<https://doi.org/10.1016/j.cor.2010.10.022>
- [74] Fang, Z., Gu, X. and Chen, J. (2017) Improved Heuristic Flower Pollination Algorithm for Solving Multi-Dimensional Knapsack Problems. *Proceedings of the IEEE 10th International Conference on Intelligent Computation Technology and Automation*, Changsha, 9-10 October 2017, 33-38.
- [75] Yang, X.S. (2012) Flower Pollination Algorithm for Global Optimization. *Proceedings of the 11th International Conference on Unconventional Computing and Natural Computation*, Orléans, 3-7 September 2012, 240-249.  
[https://doi.org/10.1007/978-3-642-32894-7\\_27](https://doi.org/10.1007/978-3-642-32894-7_27)
- [76] Malossini, A., Blanzieri, E. and Calarco, T. (2008) Quantum Genetic Optimization. *IEEE Transactions on Evolutionary Computation*, **12**, 231-241.  
<https://doi.org/10.1109/TEVC.2007.905006>
- [77] Varnamkhandi, M.J. and Lee, L.S. (2012) A Genetic Algorithm Based on Sexual Selection for the Multidimensional 0/1 Knapsack Problems. *International Journal of Modern Physics: Conference Series*, **9**, 422-431.
- [78] Varnamkhandi, M.J. and Lee, L.S. (2012) A Fuzzy Genetic Algorithm Based on Binary Encoding for Solving Multidimensional Knapsack Problems. *Journal of Applied Mathematics*, **2012**, 23.
- [79] Raidl, G.R. (1998) An Improved Genetic Algorithm for the Multiconstrained 0-1 Knapsack Problem. *Proceedings of the IEEE World Congress on Computational Intelligence*, Anchorage, 4-9 May 1998, 207-211.
- [80] Yang, H., Wang, M. and Yang, C. (2013) A Hybrid of Rough Sets and Genetic Algorithms for Solving the 0-1 Multidimensional Knapsack Problem. *International Journal of Innovative Computing, Information and Control*, **9**, 3537-3548.
- [81] Pawlak, Z. (1982) Rough Set. *International Journal of Information and Computer Science*, **11**, 341-356. <https://doi.org/10.1007/BF01001956>
- [82] Liu, X., Xiang, F. and Mao, J. (2014) An Improved Method for Solving the Large-Scale Multidimensional 0-1 Knapsack Problem. *Proceedings of the IEEE International Conference on Electronics and Communication Systems*, Coimbatore, 13-14 February 2014, 1-6.
- [83] Tisna, F.D., Abusini, S. and Andari, A. (2013) Hybrid Greedy-Particle Swarm Optimization-Genetic Algorithm and Its Convergence to Solve Multidimensional Knapsack Problem 0-1. *Journal of Theoretical & Applied Information Technology*, **58**, 522-529.
- [84] Rezoug, A., Boughaci, D. and Badr-El-Den, M. (2015) Memetic Algorithm for Solving the 0-1 Multidimensional Knapsack Problem. *Proceedings of the Portuguese Conference on Artificial Intelligence*, Coimbra, 8-11 September 2015, 298-304.
- [85] Martins, J.P., Fonseca, C.M. and Delbem, A.C. (2014) On the Performance of Linkage-Tree Genetic Algorithms for the Multidimensional Knapsack Problem. *Neurocomputing*, **146**, 17-29. <https://doi.org/10.1016/j.neucom.2014.04.069>
- [86] Yang, H.H., Wang, M.T., Chen, Y.J. and Kao, C.J. (2010) Crossover Based on Rough

- Sets: A Case of Multidimensional Knapsack Problem. *Proceedings of the IEEE International Conference on Industrial Engineering and Engineering Management*, Macao, 7-10 December 2010, 2411-2415.
- [87] Rezoug, A., Bader-El-Den, M. and Boughaci, D. (2017) Guided Genetic Algorithm for the Multidimensional Knapsack Problem. *Memetic Computing*, **10**, 14.
- [88] Rezoug, A., Bader-El-Den, M. and Boughaci, D. (2017) Knowledge-Based Genetic Algorithm for the 0-1 Multidimensional Knapsack Problem. *Proceedings of the IEEE Congress on Evolutionary Computation*, San Sebastian, 5-8 June 2017, 2030-2037.
- [89] Aghezzaf, B. and Naimi, M. (2009) The Two-Stage Recombination Operator and Its Application to the Multiobjective 0/1 Knapsack Problem: A Comparative Study. *Computers & Operations Research*, **36**, 3247-3262. <https://doi.org/10.1016/j.cor.2009.02.027>
- [90] Deane, J. and Agarwal, A. (2013) Neural, Genetic and Neurogenetic Approaches for Solving the 0-1 Multidimensional Knapsack Problem. *International Journal of Management & Information Systems*, **17**, 43-54.
- [91] da Silva, D.J.A., Teixeira, O.N. and de Oliveira, R.C.L. (2012) Performance Study of Cultural Algorithms Based on Genetic Algorithm with Single and Multi-Population for the MKP. In: Gao, S.C., Ed., *Bio-Inspired Computational Algorithms and Their Applications*, IntechOpen, 385-404.
- [92] Aguirre, H.E., Tanaka, K., Sugimara, T. and Oshita, S. (2000) Improved Distributed Genetic Algorithm with Cooperative-Competitive Genetic Operators. *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, Nashville, 8-11 October 2000, 3816-3822. <https://doi.org/10.1109/ICSMC.2000.886605>
- [93] Gupta, I.K., Choubey, A. and Choubey, S. (2017) Clustered Genetic Algorithm to Solve Multidimensional Knapsack Problem. *Proceedings of the 8th International Conference on Computing, Communication and Networking Technologies*, Delhi, 3-5 July 2017, 1-6.
- [94] Leguizamón, G. and Michalewicz, Z. (1999) A New Version of Ant System for Subset Problems. *Proceedings of the IEEE Congress on Evolutionary Computation*, Washington DC, 6-9 July 1999, 1459-1464.
- [95] Iqbal, S., Bari, M.F. and Rahman, M.S. (2010) Solving the Multi-Dimensional Multi-Choice Knapsack Problem with the Help of Ants. *Proceedings of the International Conference on Swarm Intelligence*, Beijing, 12-15 June 2010, 312-323. [https://doi.org/10.1007/978-3-642-15461-4\\_27](https://doi.org/10.1007/978-3-642-15461-4_27)
- [96] Ren, Z., Feng, Z. and Zhang, A. (2012) Fusing Ant Colony Optimization with Lagrangian Relaxation for the Multiple-Choice Multidimensional Knapsack Problem. *Information Sciences*, **182**, 15-29. <https://doi.org/10.1016/j.ins.2011.07.033>
- [97] Nakbi, W., Alaya, I. and Zouari, W. (2015) A Hybrid Lagrangian Search Ant Colony Optimization Algorithm for the Multidimensional Knapsack Problem. *Procedia Computer Science*, **60**, 1109-1119. <https://doi.org/10.1016/j.procs.2015.08.158>
- [98] Ke, L., Feng, Z., Ren, Z. and Wei, X. (2010) An Ant Colony Optimization Approach for the Multidimensional Knapsack Problem. *Journal of Heuristics*, **16**, 65-83. <https://doi.org/10.1007/s10732-008-9087-x>
- [99] Kong, M., Tian, P. and Kao, Y. (2008) A New Ant Colony Optimization Algorithm for the Multidimensional Knapsack Problem. *Computers & Operations Research*, **35**, 2672-2683. <https://doi.org/10.1016/j.cor.2006.12.029>
- [100] Al-Shihabi, S. and Ólafsson, S. (2010) A Hybrid of Nested Partition, Binary Ant



- System and Linear Programming for the Multidimensional Knapsack Problem. *Computers & Operations Research*, **37**, 247-255.  
<https://doi.org/10.1016/j.cor.2009.04.015>
- [101] Liu, R.T. and Lv, X.J. (2013) Map Reduce-Based Ant Colony Optimization Algorithm for Multi-Dimensional Knapsack Problem. *Applied Mechanics and Materials*, **380-384**, 4.
- [102] Fingler, H., Cáceres, E.N., Mongelli, H. and Song, S.W. (2014) A CUDA Based Solution to the Multidimensional Knapsack Problem Using the Ant Colony Optimization. *Procedia Computer Science*, **29**, 84-94.  
<https://doi.org/10.1016/j.procs.2014.05.008>
- [103] Solnon, C. and Bridge, D. (2006) An Ant Colony Optimization Meta-Heuristic for Subset Selection Problems. In: Nedjah, N. and Mourelle, L., Eds., *System Engineering Using Particle Swarm Optimization*, Nova Science, Hauppauge, 7-29.
- [104] Mansour, I.B. and Alaya, I. (2015) Indicator Based Ant Colony Optimization for Multi-Objective Knapsack Problem. *Procedia Computer Science*, **60**, 448-457.  
<https://doi.org/10.1016/j.procs.2015.08.165>
- [105] Iqbal, S., Bari, M.F. and Rahman, M.S. (2010) A Novel ACO Technique for Fast and near Optimal Solutions for the Multi-Dimensional Multi-Choice Knapsack Problem. *Proceedings of the 13th IEEE International Conference on Computer and Information Technology*, Dhaka, 23-25 December 2010, 33-38.
- [106] Chen, W.N., Zhang, J., Chung, H.S., Zhong, W.L., Wu, W.G. and Shi, Y.H. (2010) A Novel Set-Based Particle Swarm Optimization Method for Discrete Optimization Problems. *IEEE Transactions on Evolutionary Computation*, **14**, 278-300.  
<https://doi.org/10.1109/TEVC.2009.2030331>
- [107] Langeveld, J. and Engelbrecht, A.P. (2012) Set-Based Particle Swarm Optimization Applied to the Multidimensional Knapsack Problem. *Swarm Intelligence*, **6**, 297-342. <https://doi.org/10.1007/s11721-012-0073-4>
- [108] Gherboudj, A., Labeled, S. and Chikhi, S. (2012) A New Hybrid Binary Particle Swarm Optimization Algorithm for Multidimensional Knapsack Problem. In: Wyld, D., Zizka, J. and Nagamalai, D., Eds., *Advances in Computer Science, Engineering & Applications*, Springer, Berlin, 489-498.  
[https://doi.org/10.1007/978-3-642-30157-5\\_49](https://doi.org/10.1007/978-3-642-30157-5_49)
- [109] Langeveld, J. and Engelbrecht, A.P. (2011) A Generic Set-Based Particle Swarm Optimization Algorithm. *Proceedings of the International Conference on Swarm Intelligence*, Chongqing, 12-15 June 2011, 1-10.
- [110] Franken, N. (2009) Visual Exploration of Algorithm Parameter Space. *Proceedings of the IEEE Congress on Evolutionary Computation*, Trondheim, 18-21 May 2009, 389-398.
- [111] Chih, M., Lin, C.J., Chern, M.S. and Ou, T.Y. (2014) Particle Swarm Optimization with Time-Varying Acceleration Coefficients for the Multidimensional Knapsack Problem. *Applied Mathematical Modelling*, **38**, 1338-1350.  
<https://doi.org/10.1016/j.apm.2013.08.009>
- [112] Lin, C.J., Chern, M.S. and Chih, M. (2016) A Binary Particle Swarm Optimization Based on the Surrogate Information with Proportional Acceleration Coefficients for the 0-1 Multidimensional Knapsack Problem. *Journal of Industrial and Production Engineering*, **33**, 77-102. <https://doi.org/10.1080/21681015.2015.1111263>
- [113] Haddar, B., Khemakhem, M., Hanafi, S. and Wilbaut, C. (2016) A hybrid Quantum Particle Swarm Optimization for the Multidimensional Knapsack Problem. *Engi-*

neering Applications of Artificial Intelligence, **55**, 1-13.

<https://doi.org/10.1016/j.engappai.2016.05.006>

- [114] Chih, M. (2015) Self-Adaptive Check and Repair Operator-Based Particle Swarm Optimization for the Multidimensional Knapsack Problem. *Applied Soft Computing*, **26**, 378-389. <https://doi.org/10.1016/j.asoc.2014.10.030>
- [115] Chih, M. (2017) A Three-Ratio SACRO-Based Particle Swarm Optimization with Local Search Scheme for the Multidimensional Knapsack Problem. *Proceedings of the 18th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing*, Kanazawa, 26-28 June 2017, 17-22.
- [116] Chih, M. (2017) Three Pseudo-Utility Ratio-Inspired Particle Swarm Optimization with Local Search for Multidimensional Knapsack Problem. *Swarm and Evolutionary Computation*, **39**, 18.
- [117] Zan, D. and Jaros, J. (2014) Solving the Multidimensional Knapsack Problem Using a CUDA Accelerated PSO. *Proceedings of the IEEE Congress on Evolutionary Computation*, Beijing, 6-11 July 2014, 2933-2939.
- [118] Weingartner, H.M. and Ness, D.N. (1967) Methods for the Solution of the Multi-Dimensional 0/1 Knapsack Problem. *Operations Research*, **15**, 83-103. <https://doi.org/10.1287/opre.15.1.83>
- [119] Bonyadi, M.R. and Michalewicz, Z. (2012) A Fast Particle Swarm Optimization Algorithm for the Multidimensional Knapsack Problem. *Proceedings of the IEEE Congress on Evolutionary Computation*, Brisbane, 10-15 June 2012, 1-8.
- [120] Ktari, R. and Chabchoub, H. (2013) Essential Particle Swarm Optimization Queen with Tabu Search for MKP Resolution. *Computing*, **95**, 897-921. <https://doi.org/10.1007/s00607-013-0316-2>
- [121] Sundar, S., Singh, A. and Rossi, A. (2010) An Artificial Bee Colony Algorithm for the 0-1 Multidimensional Knapsack Problem. *Proceedings of the International Conference on Contemporary Computing*, Noida, 9-11 August 2010, 141-151. [https://doi.org/10.1007/978-3-642-14834-7\\_14](https://doi.org/10.1007/978-3-642-14834-7_14)
- [122] Ji, J., Wei, H., Liu, C. and Yin, B. (2013) Artificial Bee Colony Algorithm Merged with Pheromone Communication Mechanism for the 0-1 Multidimensional Knapsack Problem. *Mathematical Problems in Engineering*, **2013**, 13. <https://doi.org/10.1155/2013/676275>
- [123] Bole, A.V. and Kumar, R. (2017) Multidimensional 0-1 Knapsack Using Directed Bee Colony Algorithm. *Proceedings of the IEEE International Conference on Intelligent Techniques in Control, Optimization and Signal Processing*, Srivilliputhur, 23-25 March 2017, 1-10.
- [124] Sabba, S. and Chikhi, S. (2014) A Discrete Binary Version of Bat Algorithm for Multidimensional Knapsack Problem. *International Journal of Bio-Inspired Computation*, **6**, 140-152. <https://doi.org/10.1504/IJBIC.2014.060598>
- [125] Gong, Q., Zhou, Y. and Luo, Q. (2011) Hybrid Artificial Glowworm Swarm Optimization Algorithm for Solving Multidimensional Knapsack Problem. *Procedia Engineering*, **15**, 2880-2884. <https://doi.org/10.1016/j.proeng.2011.08.542>
- [126] Azad, M.A.K., Rocha, A.M.A. and Fernandes, E.M. (2014) Improved Binary Artificial Fish Swarm Algorithm for the 0-1 Multidimensional Knapsack Problems. *Swarm and Evolutionary Computation*, **14**, 66-75. <https://doi.org/10.1016/j.swevo.2013.09.002>
- [127] Azad, M.A.K., Rocha, A.M.A. and Fernandes, E.M. (2012) Solving Multidimensional 0-1 Knapsack Problem with an Artificial Fish Swarm Algorithm. *Proceedings of*

*the International Conference on Computational Science and Its Applications*, Salvador de Bahia, 18-21 June 2012, 72-86.

- [128] Azad, M.A.K., Rocha, A.M.A. and Fernandes, E.M. (2015) Solving Large 0-1 Multidimensional Knapsack Problems by a New Simplified Binary Artificial Fish Swarm Algorithm. *Journal of Mathematical Modelling and Algorithms in Operations Research*, **14**, 313-330. <https://doi.org/10.1007/s10852-015-9275-2>
- [129] Meng, T., Duan, J.H. and Pan, Q.K. (2017) An Improved Migrating Birds Optimization for Solving the Multidimensional Knapsack Problem. *Proceedings of the 29th IEEE Chinese Control and Decision Conference*, Chongqing, 28-30 May 2017, 4698-4703.
- [130] Zhang, X., Wu, C., Li, J., Wang, X., Yang, Z., Lee, J.M. and Jung, K.H. (2016) Binary Artificial Algae Algorithm for Multidimensional Knapsack Problems. *Applied Soft Computing*, **43**, 583-595. <https://doi.org/10.1016/j.asoc.2016.02.027>
- [131] Baykasoğlu, A. and Ozsoydan, F.B. (2014) An Improved Firefly Algorithm for Solving Dynamic Multidimensional Knapsack Problems. *Expert Systems with Applications*, **41**, 3712-3725. <https://doi.org/10.1016/j.eswa.2013.11.040>
- [132] Zouache, D., Nouioua, F. and Moussaoui, A. (2016) Quantum-Inspired Firefly Algorithm with Particle Swarm Optimization for Discrete Optimization Problems. *Soft Computing*, **20**, 2781-2799. <https://doi.org/10.1007/s00500-015-1681-x>
- [133] Meng, T. and Pan, Q.K. (2017) An Improved Fruit Fly Optimization Algorithm for Solving the Multidimensional Knapsack Problem. *Applied Soft Computing*, **50**, 79-93. <https://doi.org/10.1016/j.asoc.2016.11.023>
- [134] Abdel-Basset, M., El-Shahat, D., El-Henawy, I. and Sangaiah, A.K. (2017) A Modified Flower Pollination Algorithm for the Multidimensional Knapsack Problem: Human-Centric Decision Making. *Soft Computing*, **22**, 19.
- [135] Kong, X., Gao, L., Ouyang, H. and Li, S. (2015) Solving Large-Scale Multidimensional Knapsack Problems with a New Binary Harmony Search Algorithm. *Computers & Operations Research*, **63**, 7-22. <https://doi.org/10.1016/j.cor.2015.04.018>
- [136] Liu, J., Wu, C., Cao, J., Wang, X. and Teo, K.L. (2016) A Binary Differential Search Algorithm for the 0-1 Multidimensional Knapsack Problem. *Applied Mathematical Modelling*, **40**, 9788-9805. <https://doi.org/10.1016/j.apm.2016.06.002>
- [137] Tasgetiren, M.F., Pan, Q.K., Kizilay, D. and Suer, G. (2015) A Differential Evolution Algorithm with Variable Neighborhood Search for Multidimensional Knapsack Problem. *Proceedings of the IEEE Congress on Evolutionary Computation*, Sendai, 25-28 May 2015, 2797-2804.
- [138] Peng, H., Wu, Z., Shao, P. and Deng, C. (2016) Dichotomous Binary Differential Evolution for Knapsack Problems. *Mathematical Problems in Engineering*, **2016**, 12. <https://doi.org/10.1155/2016/5732489>
- [139] Wang, L., Yang, R., Ni, H., Ye, W., Fei, M. and Pardalos, P.M. (2015) A Human Learning Optimization Algorithm and Its Application to Multi-Dimensional Knapsack Problems. *Applied Soft Computing*, **34**, 736-743. <https://doi.org/10.1016/j.asoc.2015.06.004>
- [140] Bonyadi, M.R. and Li, X. (2012) A New Discrete Electromagnetism-Based Meta-Heuristic for Solving the Multidimensional Knapsack Problem Using Genetic Operators. *Operational Research*, **12**, 229-252. <https://doi.org/10.1007/s12351-010-0084-0>
- [141] Hanafi, S. and Wilbaut, C. (2008) Scatter Search for the 0-1 Multidimensional Knapsack Problem. *Journal of Mathematical Modelling and Algorithms*, **7**, 143-159.

- <https://doi.org/10.1007/s10852-008-9078-9>
- [142] Vianna, D.S., Montané, F.A.T., Meza, E.B.M. and Martins, C.B. (2014) A Memory-Based Grasp Heuristic for the Multiobjective Multidimensional Knapsack Problem. *International Journal of Latest Research in Science and Technology*, **3**, 186-194.
- [143] Feo, T.A., Resende, M.G. and Smith, S.H. (1994) A Greedy Randomized Adaptive Search Procedure for Maximum Independent set. *Operations Research*, **42**, 860-878. <https://doi.org/10.1287/opre.42.5.860>
- [144] Lust, T. and Teghem, J. (2010) Two-Phase Pareto Local Search for the Biobjective Traveling Salesman Problem. *Journal of Heuristics*, **16**, 475-510. <https://doi.org/10.1007/s10732-009-9103-9>
- [145] Ahuja, R.K., Ergun, O., Orlin, J.B. and Punnen, A.P. (2002) A Survey of Very Large-Scale Neighborhood Search Techniques. *Discrete Applied Mathematics*, **123**, 75-102. [https://doi.org/10.1016/S0166-218X\(01\)00338-9](https://doi.org/10.1016/S0166-218X(01)00338-9)
- [146] Lust, T. and Teghem, J. (2008) Memots: A Memetic Algorithm Integrating Tabu Search for Combinatorial Multiobjective Optimization. *RAIRO Operations Research*, **42**, 3-33. <https://doi.org/10.1051/ro:2008003>
- [147] Li, V.C., Liang, Y.C., Sun, Y.Y. and Chen, Y.S. (2012) A Scatter Search Method for the Multidimensional Knapsack Problem with Generalized Upper Bound Constraints. *Journal of the Chinese Institute of Industrial Engineers*, **29**, 559-571. <https://doi.org/10.1080/10170669.2012.743643>
- [148] Kenneth, Z., Lu, Y. and Vasko, F.J. (2015) Teacher Training Enhances the Teaching-Learning-Based Optimization Metaheuristic when Used to Solve Multiple-Choice Multidimensional Knapsack Problems. *International Journal of Metaheuristics*, **4**, 268-293.
- [149] Wang, L., Wang, S.Y. and Xu, Y. (2012) An Effective Hybrid EDA-Based Algorithm for Solving Multidimensional Knapsack Problem. *Expert Systems with Applications*, **39**, 5593-5599. <https://doi.org/10.1016/j.eswa.2011.11.058>
- [150] Yuan, Q. and Yang, Z.X. (2016) A Weight-Coded Evolutionary Algorithm for the Multidimensional Knapsack Problem. *Advances in Pure Mathematics*, **6**, 659-675. <https://doi.org/10.4236/apm.2016.610055>
- [151] Deane, J. and Agarwal, A. (2012) Neural Metaheuristics for the Multidimensional Knapsack Problem. Technical Report. [https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=2ahUKewjXqfeeks\\_dAhWSxoUKHcU\\_CvkQFjAAegQIARAC&url=https%3A%2F%2Fpdfs.semanticscholar.org%2F2f57%2F1ae946c865b4e98e0a980c31806a686b4621.pdf&usg=AOvVaw1vNuVhp9pjBPBFnlEkyHQc](https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=2ahUKewjXqfeeks_dAhWSxoUKHcU_CvkQFjAAegQIARAC&url=https%3A%2F%2Fpdfs.semanticscholar.org%2F2f57%2F1ae946c865b4e98e0a980c31806a686b4621.pdf&usg=AOvVaw1vNuVhp9pjBPBFnlEkyHQc)
- [152] De Almeida Dantas, B. and Cáceres, E.N. (2016) Sequential and Parallel Hybrid Approaches of Augmented Neural Networks and GRASP for the 0-1 Multidimensional Knapsack Problem. *Proceedings of the International Conference on Computational Science and Its Applications*, Beijing, 4-7 July 2016, 207-222.
- [153] Vianna, D.S. and Vianna, M.D.F.D. (2013) Local Search-Based Heuristics for the Multiobjective Multidimensional Knapsack Problem. *Production*, **23**, 478-487. <https://doi.org/10.1590/S0103-65132012005000081>
- [154] Alves, M.J. and Almeida, M.M. (2007) A Multiobjective Tchebycheff Based Genetic Algorithm for the Multidimensional Knapsack Problem. *Computers & Operations Research*, **34**, 3458-3470. <https://doi.org/10.1016/j.cor.2006.02.008>
- [155] Jaskiewicz, A. (2002) On the Performance of Multiple Objective Genetic Local

- Search on the 0/1 Knapsack Problem: A Comparative Experiment. *IEEE Transactions on Evolutionary Computation*, **6**, 402-412.  
<https://doi.org/10.1109/TEVC.2002.802873>
- [156] Zitzler, E., Laumanns, M. and Thiele, L. (2001) SPEA2: Improving the Strength Pareto Evolutionary Algorithm. Research Report.
- [157] Moraga, R.J., DePuy, G.W. and Whitehouse, G.E. (2004) Meta-RaPS Approach for the 0-1 Multidimensional Knapsack Problem. *Computers & Industrial Engineering*, **48**, 83-96. <https://doi.org/10.1016/j.cie.2004.02.008>
- [158] Arin, A. and Rabadi, G. (2016) Local Search versus Path Relinking in Metaheuristics: Redesigning Meta-RaPS with Application to the Multidimensional Knapsack Problem. *Applied Soft Computing*, **46**, 317-327.  
<https://doi.org/10.1016/j.asoc.2016.05.016>
- [159] Arin, A. and Rabadi, G. (2017) Integrating Estimation of Distribution Algorithms versus Q-Learning into Meta-RaPS for Solving the 0-1 Multidimensional Knapsack Problem. *Computers & Industrial Engineering*, **112**, 706-720.  
<https://doi.org/10.1016/j.cie.2016.10.022>
- [160] Htiouech, S. and Alzaidi, A. (2017) Smart Agents for the Multidimensional Multi-Choice Knapsack Problem. *International Journal of Computer Applications*, **174**, 5-9. <https://doi.org/10.5120/ijca2017915404>
- [161] Khalili-Damghani, K., Nojavan, M. and Tavana, M. (2013) Solving Fuzzy Multidimensional Multiple-Choice Knapsack Problems: The Multi-Start Partial Bound Enumeration Method versus the Efficient Epsilon-Constraint Method. *Applied Soft Computing*, **13**, 1627-1638. <https://doi.org/10.1016/j.asoc.2013.01.014>
- [162] Mavrotas, G. (2009) Effective Implementation of the Epsilon-Constraint Method in Multi-Objective Mathematical Programming problems. *Applied Mathematics and Computations*, **213**, 455-465. <https://doi.org/10.1016/j.amc.2009.03.037>
- [163] Gao, C., Lu, G., Yao, X. and Li, J. (2017) An Iterative Pseudo-Gap Enumeration Approach for the Multidimensional Multiple-Choice Knapsack Problem. *European Journal of Operational Research*, **260**, 1-11.  
<https://doi.org/10.1016/j.ejor.2016.11.042>
- [164] Boussier, S., Vasquez, M., Vimont, Y., Hanafi, S. and Michelon, P. (2010) A Multi-Level Search Strategy for the 0-1 Multidimensional Knapsack Problem. *Discrete Applied Mathematics*, **158**, 97-109. <https://doi.org/10.1016/j.dam.2009.08.007>
- [165] Sharafi, P., Askarian, M., Uz, M.E. and Abaci, H. (2017) Form Finding for Rectilinear Orthogonal Buildings through Charged System Search Algorithm. *International Journal of Optimization in Civil Engineering*, **7**, 129-142.
- [166] Kaveh, A. and Talatahari, S. (2010) A Novel Heuristic Optimization Method: Charged System Search. *ActaMechanica*, **213**, 267-289.  
<https://doi.org/10.1007/s00707-009-0270-4>
- [167] Urli, T. (2015) Hybrid Meta-Heuristics for Combinatorial Optimization. Ph.D. Thesis, University of Udine, Italy.  
[https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=2ahUKewiaoIf8mc\\_dAhUM1hoKHebcBcAQFjAAegQIChAC&url=https%3A%2F%2Fwww.a4cp.org%2Fsites%2Fdefault%2Ffiles%2Ftommaso\\_urli\\_-\\_hybr-id\\_meta-heuristics\\_for\\_combinatorial\\_optimisation.pdf&usq=AOvVaw3JThZYEPS1Op5oLRPZ2IzX](https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=2ahUKewiaoIf8mc_dAhUM1hoKHebcBcAQFjAAegQIChAC&url=https%3A%2F%2Fwww.a4cp.org%2Fsites%2Fdefault%2Ffiles%2Ftommaso_urli_-_hybr-id_meta-heuristics_for_combinatorial_optimisation.pdf&usq=AOvVaw3JThZYEPS1Op5oLRPZ2IzX)