Computers & Operations Research 37 (2010) 700–711

Contents lists available at ScienceDirect

Computers & Operations Research

journal homepage: www.elsevier.com/locate/cor

Labeling algorithms for multiple objective integer knapsack problems

José Rui Figueira[a,1], Gabriel Tavares[b], Margaret M. Wiecek[c,*]

[a]CEG-IST, Center for Management Studies - Instituto Superior Técnico, TagusPark, Av. Cavaco Silva, 2780-990 Porto Salvo, Portugal
[b]Rutcor, Rutgers University, 640 Bartholomew Road Piscataway, NJ 08854-8003, USA
[c]Department of Mathematical Sciences, Clemson University, Clemson, SC 29634-1907, USA

ARTICLE INFO

Available online 18 July 2009

Keywords:
Multiple objective integer knapsack problem
Labeling algorithms
Shortest path problem

ABSTRACT

The paper presents a generic labeling algorithm for finding all nondominated outcomes of the multiple objective integer knapsack problem (MOIKP). The algorithm is based on solving the multiple objective shortest path problem on an underlying network. Algorithms for constructing four network models, all representing the MOIKP, are also presented. Each network is composed of layers and each network algorithm, working forward layer by layer, identifies the set of all permanent nondominated labels for each layer. The effectiveness of the algorithms is supported with numerical results obtained for randomly generated problems for up to seven objectives while exact algorithms reported in the literature solve the multiple objective binary knapsack problem with up to three objectives. Extensions of the approach to other classes of problems including binary variables, bounded variables, multiple constraints, and time-dependent objective functions are possible.

Published by Elsevier Ltd.

1. Introduction

The multiple objective knapsack problem (MOKP) is a well known combinatorial optimization problem with a wide range of applications. Examples may be found in affordability analysis and capital budgeting where projects have to be chosen with respect to more than a single criterion (see e.g., [6,34,24]), in transportation investment planning [31], or in conservation biology [22].

In this paper we consider the multiple objective integer knapsack problem (MOIKP) formulated as

$$\begin{aligned} \max \quad & f(x) = Vx \\ \text{s.t.} \quad & wx \le W \\ & x_j \ge 0, \text{ integer}, \quad j = 1, \dots, l \end{aligned} \tag{1}$$

where V is an $r \times l$ matrix with nonnegative entries v_j^i, $i = 1, \dots, r$, $j = 1, \dots, l$. We denote the i th row of V by v^i and the j th column of V by v_j. Thus $f_i(x) = v^i x$, $i = 1, \dots, r$, represents the i th objective among the r conflicting objective functions. A special case of the above formulation is the case in which $r = 2$, i.e., the bi-objective case. The constraint $wx \le W$ is interpreted as a *capacity constraint* (*budget constraint*). The set of feasible solutions of (1) is denoted by $X = \{x : wx \le W, x_j \ge 0, \text{ integer}, j = 1, \dots, l\}$. We also consider the MOIKP with the right-hand-side coefficient equal to $h = 0, 1, 2, \dots, W$ and denote this problem by h-MOIKP.

Throughout the paper, we additionally assume that the weight coefficients w_j, $j = 1, \dots, l$, and the right-hand-side of the capacity constraint W are positive integers. In order to avoid trivial solutions let $0 < w_j \le W$, $j = 1, \dots, l$ and $\sum_{j=1}^{l} w_j > W$.

Solving (1) is understood as generating its efficient (Pareto) solutions. A feasible solution $\hat{x} \in X$ is said to be an efficient solution of (1) if there is no other feasible solution $x \in X$ such that $f(x) \geq f(\hat{x})$, i.e.:

$$\forall i \in \{1, \dots, r\}, \quad f_i(x) \ge f_i(\hat{x})$$

$$\text{and} \quad \exists i \in \{1, \dots, r\} \quad \text{such that } f_i(x) > f_i(\hat{x}) \tag{2}$$

Let $\mathcal{X}_e$ denote the set of efficient solutions (1) and let $\mathcal{Y}_e$ denote the image of $\mathcal{X}_e$ in the objective space, that is $\mathcal{Y}_e = f(\mathcal{X}_e)$, where $f = (f_1, \dots, f_r)$. The set $\mathcal{Y}_e$ is referred to as the set of nondominated outcomes of (1).

The MOKP is a difficult problem to solve since the binary single-criterion knapsack problem is already NP-complete. While many authors have proposed algorithms for finding all or some nondominated outcomes of the MOKP, a majority of the algorithms deal with the bi-objective knapsack problem [7,13,14,18,32].

* Corresponding author. Tel.: +1 864 656 5245; fax: +1 864 656 5230.
E-mail addresses: figueira@ist.utl.pt (J. Rui Figueira), gtavares@rutcor.rutgers.edu (G. Tavares), wmalgor@clemson.edu (M.M. Wiecek).
[1] Part of this work was accomplished when the first author was a visiting Researcher at DIMACS Center, Rutgers University, CoRE Building, 4th Floor, 96 Frelinghuysen Road, Piscataway, NJ 08854-8018, USA. (José R. Figueira is also Associate Researcher at LAMSADE, Université Paris-Dauphine) Tel.:+1 732 445 0075; fax: 1 732 445 5932.

0305-0548/$ - see front matter Published by Elsevier Ltd.
doi:10.1016/j.cor.2009.06.026

As the MOKP falls into the category of multiple objective integer programs, algorithms proposed for finding nondominated outcomes of the latter could be also applied to solve the former. In this review we focus on approaches developed specially for the knapsack problem with more than two objectives. In general, these approaches can be classified as exact procedures and metaheuristics. The former aims at finding all nondominated outcomes of the MOKP and includes branch-and-bound procedures [32] and dynamic-programming-based approaches solved in general with labeling algorithms.

Villarreal and Karwan [35] were perhaps the first ones who proposed dynamic programming (DP) approaches to the MOKP. They proposed four approaches: two basic ones, an embedded state approach, and a hybrid approach. Later in [36], they also extended DP recursive equations to the general multiple objective integer framework and presented them on the binary MOKP with multiple constraints. Klamroth and Wiecek [21] proposed a comprehensive DP methodology able to solve a broad class of knapsack problems including the binary and integer case and more complex extensions such as MOKP with multiple constraints, multiple periods, and time-dependent criterion functions. They presented DP recursive equations for five network models representing the MOIKP and showed how to apply the equations and networks for the original problem and its extensions.

An independent research effort was undertaken by Captivo et al. [7] who applied the concept of a labeling algorithm to the MOKP viewed as the multiple objective shortest path problem on an underlying network. The algorithm turned out to be very effective for some hard instances of the bi-objective binary case of the MOKP.

The combinatorial nature of MOKP motivated the development of meta-heuristic algorithms producing a subset or an approximation of the set of all nondominated outcomes. Arndt and Seelaender [2] outlined an approach based on the concept of ceiling points. Simulated annealing was extensively studied by Czyzak and Jaszkiewicz [9] and Ulungu et al. [33]. Hansen [18] and Gandibleux et al. [11] applied tabu-search principles to construct an approximation of the nondominated set. Combinations of tabu search and a genetic algorithm were developed by Ben Abdelaziz et al. [5]. A comparative study of the effectiveness of genetic algorithms was presented by Zitzler and Thiele [38].

In the current decade, researchers continued efforts on the development of genetic or hybrid algorithms, and also exact algorithms. Improved performance of genetic algorithms due to the use of approximate dominance was reported by Grosan and Oltsean [15] and Kumar and Banerjee [23]. Barichard and Hao [3] combined a genetic procedure with a tabu search operator, Guo et al. [17] proposed a hybrid memetic algorithm, and Zhang et al. [37] developed an immune system strength Pareto algorithm based on a clonal selection theory. Some authors also conducted more comparative studies including Laumanns et al. [26], Jaszkiewicz [20], and Colombo and Mumford [8].

In their recent articles, Laumanns et al. [25,27,28] present a method based on solving a sequence of constrained single-objective problems for the binary MOKP with three criteria. The same case is studied by Bazgan et al. [4] who propose an approach based on dynamic programming. These articles show that despite the success of metaheuristic algorithms, exact algorithms remain competitive for solving the MOKP.

In this paper, we focus on the MOIKP since to the best of our knowledge, a big majority of the literature deals with the binary case and a computational study of the MOIKP with more than two objectives has not been reported. We continue the development of labeling algorithms since they showed to be very promising in constructing an effective algorithm for solving the binary MOKP [7]. On the other hand, the dynamic-programming framework provided by Klamroth and Wiecek [21] for the MOIKP turned out to be a flexible tool for solving a variety of knapsack problems with multiple objectives. We therefore present a new framework featuring the computational effectiveness of labeling algorithms and the modeling flexibility of dynamic programming. This new framework is composed of a family of networks similar to those in [21] for which a generic labeling algorithm is designed. The algorithm with minor adjustments solves the multiple objective shortest path problem on every network and generates the set of all nondominated outcomes of the MOIKP.

In this paper, numerical results obtained with exact algorithms for the MOIKP with up to seven objective functions are reported for the first time. In Section 2, algorithms for constructing several network models for the MOIKP are presented. Although the models are collected from the literature, the algorithms reduce the original models and are implemented for the first time. The generic labeling algorithm, suitable for customization for the proposed network models, is developed in Section 3 while Section 4 describes computational experiments and contains detailed numerical results. The effectiveness of the algorithms is supported with numerical results obtained for randomly generated problems for up to seven objectives and 20 variables for all models, and three objectives and 40 variables for one of the models. One of the algorithms clearly outperforms the others when solving the MOIKP with up to seven objectives. Section 5 indicates further possibilities for the refinement of the algorithm and concludes the paper.

2. Algorithms for building network models

In this section we present four network models to be used for solving the MOIKP. These models are based on modeling approaches available in the literature and developed within the framework of dynamic programming. The new models are presented in the language of network flow programming [1] and their main feature is their reduced size. The new networks include a smaller number of vertices and arcs in comparison to the dynamic programming based networks.

Every network is defined as a directed and connected graph $\mathcal{G}=(\mathcal{V},\mathcal{A})$, where $\mathcal{V}$ is the set of vertices with $|\mathcal{V}|=n$ and $\mathcal{A}\subseteq\mathcal{V}\times\mathcal{V}\setminus\{(i,i)|i\in\mathcal{V}\}$ is the set of arcs with $|\mathcal{A}|=m$. The arc linking vertices i and j is denoted by (i,j), and the cost vector $c(i,j)=(c^1(i,j),\dots,c^r(i,j))$ is composed of r criterion values associated with the arc (i,j). These costs are determined by the negative of the entries of matrix V in (1). In the set $\mathcal{V}$, we distinguish a source vertex, a sink vertex and terminal vertices that are directly connected with the sink vertex. A path p from a source vertex s to a sink vertex t in $\mathcal{G}$ is a sequence of arcs and vertices from s to t, where the tail vertex of a given arc coincides with the head vertex of the next arc in the path.

Every network is composed of layers. A layer is a set of vertices. A layer g is called a successor of a layer g' if there is at least one arc from g' to g.

The networks have several common properties. All networks are acyclic. There may exist arcs from layer g to layers $g+1,g+2,\dots,G$. Within a layer, there may exist arcs linking vertices in this layer and there may exist terminal vertices. In the topological terms, all the arcs in a network are horizontal, vertical, or diagonal down right. This property allows us to set permanent labels from the top to the bottom of each layer.

Every vertex has a position in a layer. A vertex in position k, for $k=0,\dots,K$ in layer g is denoted by g^k. For example, 3^6 denotes a vertex in layer 3 in position 6.

Throughout the paper we use the following didactic example of the bi-objective case of MOIKP to illustrate all network models and