European Journal of Operational Research 000 (2016) 1–11

Contents lists available at ScienceDirect

European Journal of Operational Research

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

Discrete Optimization

A branch-and-price algorithm for the two-dimensional vector packing problem with piecewise linear cost function

Qian Hu[a,b], Wenbin Zhu[c,*], Hu Qin[d], Andrew Lim[b,a]

[a] International Center of Management Science and Engineering, School of Management and Engineering, Nanjing University, Nanjing 210093, China
[b] Department of Industrial & Systems Engineering, National University of Singapore, Singapore
[c] School of Business Administration, South China University of Technology, Guangzhou 510640, China
[d] School of Management, Huazhong University of Science and Technology, No. 1037, Luoyu Road, Wuhan, China

A R T I C L E I N F O

Article history:
Received 27 July 2015
Accepted 14 December 2016
Available online xxx

Keywords:
Packing
2D vector packing
Piecewise linear cost function
Branch-and-price
Column generation

A B S T R A C T

The two-dimensional vector packing problem with piecewise linear cost function (2DVPP-PLC) models a practical packing problem faced by many companies that use courier services. We propose a branch-and-price algorithm to solve the 2DVPP-PLC exactly. The column generation procedure is a key component that affects the performance of a branch-and-price algorithm. The pricing problem exhibits an interesting structure that allows us to decompose it into subproblems that form a lattice. We explore dominance relations on the lattice and design an efficient algorithm for the pricing problem. Experimental results show that our branch-and-price algorithm is capable of solving 2DVPP-PLC test instances effectively.

© 2016 Elsevier B.V. All rights reserved.

1. Introduction

The two-dimensional vector packing problem with piecewise linear cost function (2DVPP-PLC) is an important logistic planning problem commonly faced by companies that ship their products using a courier service. Hu, Lim, and Zhu (2015) motivated the problem based on a real application from an international manufacturer of children's apparel, such as shorts, jackets, jumpsuits and pajamas. The manufacturer operates several production bases and hundreds of retail stores located across the globe. In the transportation process, an express courier company is employed to distribute products from its production bases to retail stores. Since all items are flexible and non-fragile, only their weights and volumes need to be considered during the packing process. The express courier company charges the manufacturer for each carton of items based on its weight. The cost structure is approximated using a piecewise linear cost function of the weight.

Hu et al. (2015) investigated an integer linear programming (ILP) model for the 2DVPP-PLC. Solving the ILP model directly is viable only for small instances. They resorted to heuristics to solve large instances. To evaluate their two heuristics, Hu et al. (2015) generated a set of instances with known optimal solutions. However, neither heuristic was able to find an optimal solution for any of these instances. For random instances, although high-quality solutions are generated by these heuristics quickly, the optimality is hard to be proved. The above issues motivate us to design an exact algorithm for the problem.

We formally define the 2DVPP-PLC and introduce a set cover formation in Section 3. We then proceed to develop an exact algorithm based on the branch-and-price framework in Section 4. The key components that affect the performance of a branch-and-price algorithm are the branching scheme and the column generation procedure. We adapt a branching scheme from literature in Section 4.1. By introducing branching constraints at a node, the pricing problem exhibits an interesting structure (Section 5). It is a two-constraint knapsack problem (2KP) with a set of collectable bonus and penalty patterns. If a solution dominates a bonus (penalty) pattern, we collect the bonus (penalty). A dynamic programming algorithm is applied to produce feasible solutions of the 2KP quickly, if possible. When the dynamic programming algorithm fails, we resort to enumerating all feasible solutions, which can be done efficiently by decomposing the pricing problem into subproblems and exploring dominance relations. The feasible region of subproblems forms a lattice; therefore, we propose a lattice-based decomposition algorithm in Section 5.2.

Computational experiments are conducted to investigate the performance of our branch-and-price algorithm on different types of cost functions and instance sizes in Section 7. Results show that our algorithm is able to solve random 2DVPP-PLC instances. In addition, it verified optimal solutions for 78 out of 120 existing

* Corresponding author.
E-mail addresses: huqian@nju.edu.cn (Q. Hu), i@zhuwb.com (W. Zhu), tigerqin1980@gmail.com (H. Qin), alim.china@gmail.com (A. Lim).

http://dx.doi.org/10.1016/j.ejor.2016.12.021
0377-2217/© 2016 Elsevier B.V. All rights reserved.

instances within 3600 seconds; some of them involve as many as 1024 items. It also solved 321 out of 400 2DVPP instances to optimality within a time limit of 600 seconds.

2. Literature review

The 2DVPP-PLC is NP-hard since it is one of the many generalizations of the classical *one-dimensional bin packing problem* (1DBPP) (Coffman, Garey, & Johnson, 1997; Dyckhoff, 1990). Among the many variants of the 1DBPP in the literature, the problems that are closest to the 2DVPP-PLC are the *two-dimensional vector packing problem* (2DVPP) (Alves, De Carvalho, Clautiaux, & Rietz, 2014; Caprara & Toth, 2001; Spieksma, 1994) and the *bin packing problem with general cost structure* (BPP-GC) (Anily, Bramel, & Simchi-Levi, 1994; Epstein & Levin, 2012; Leung & Li, 2008; Li & Chen, 2006).

For a BPP-GC, the objective is to minimize the total cost instead of the number of bins used. Actually, the BPP-GC represents a group of problems, which differ in the types of their cost structures. We have found two predominant types of cost structures in the current literature: one is to define the cost of a bin as a concave and monotone function of the number of items packed into that bin (Anily et al., 1994; Epstein & Levin, 2012), and the other is to define the cost of a bin as a non-decreasing concave function of the utilization of the bin (Leung & Li, 2008; Li & Chen, 2006). The piecewise linear cost function used in the 2DVPP-PLC is another type of cost structure, which is not required to be convex nor concave.

Heuristics and exact methods are used to solve all kinds of bin packing problem and variants. *Constructive heuristics* pack items into bins one by one and do not change the assignments of the already packed items. Asymptotic analysis has been conducted for some simple constructive heuristics (Anily et al., 1994; Epstein & Levin, 2012; Leung & Li, 2008; Li & Chen, 2006). In the existing literature, there are some complex constructive heuristics that produce solutions without any theoretical quality guarantee; examples include the extreme point-based constructive heuristics for the three-dimensional bin packing problem (Crainic, Perboli, & Tadei, 2008) and the constructive bin-oriented heuristic for the two-dimensional bin packing problem with guillotine cuts (Charalambous & Fleszar, 2011). *Meta-heuristics* are able to produce high-quality solutions in reasonable computation time, but their worst case bounds on solution quality usually cannot be proven. Examples of the use of meta-heuristics include a guided local search heuristic and a tabu search heuristic for the three-dimensional bin packing problem (Crainic, Perboli, & Tadei, 2009; Faroe, Pisinger, & Zachariasen, 2003), a genetic algorithm for the variable sized bin-packing problem (Haouari & Serairi, 2009), and a weighted annealing heuristic for the 1DBPP (Loh, Golden, & Wasil, 2008). *Exact algorithms* are able to solve the problem to optimality. The exact algorithms for bin packing problem and its variants include the branch-and-bound algorithm (Caprara & Toth, 2001; Martello, Pisinger, & Vigo, 2000), the branch-and-price algorithm (Elhedhli, Li, Gzara, & Naoum-Sawaya, 2011; Vanderbeck, 1999) and the branch-and-price-and-cut algorithm (Belov & Scheithauer, 2006; Pisinger & Sigurd, 2007).

After thoroughly reviewing the prior research, we find that branch-and-price algorithms have successfully been applied to the 1DBPP and its variants that consider one-dimensional items. As early as 1994, Vance, Barnhart, Johnson, and Nemhauser (1994) proposed a branch-and-price algorithm to solve a special one-dimensional cutting stock problem, in which the demand for the roll of each length equals one; this problem is a special case of the 1DBPP. Later, Valério de Carvalho (1999), Vance (1998), and Vanderbeck (1999) proposed branch-and-price algorithms for the bin packing problem and cutting stock problem. In recent years, branch-and-price was applied to solve the variants of the 1DBPP, such as the bin packing problem with conflicts (Elhedhli et al., 2011; Fernandes Muritiba, Iori, Malaguti, & Toth, 2010; Sadykov & Vanderbeck, 2013), the ordered open-end bin packing problem (Ceselli & Righini, 2008) and the variable size bin packing problem with minimum filling constraint (Bettinelli, Ceselli, & Righini, 2010). To the best of our knowledge, no algorithm for the exact solutions of the BPP-GC has yet been published. Thus, this work initiates the study of exact algorithms for the 2DVPP-PLC.

3. Problem formulation

In the two-dimensional vector packing problem with piecewise linear cost function (2DVPP-PLC), the objective is to pack all given items into identical bins while minimizing the total cost of utilized bins. There are n types of items. For each item type, $i = 1, 2, \ldots, n$, there are d_i items, each with volume v_i and weight w_i. The volume and weight capacity of a bin are given by V and W, respectively. A set of items can be packed into a bin, if their total volume and weight do not exceed the capacities of the bin. The cost of a bin is a piecewise linear function of the total weight w of the items packed in a bin. The cost function consists of K pieces, with the kth piece given by $f_k(w) = \alpha_k w + \beta_k, w \in [e_{k-1}, e_k]$ and $0 = e_0 < e_1 < \cdots < e_K = W$. We further assume that v_i and w_i are integers and do not exceed V and W, respectively, the cost function is monotonic non-decreasing, and there is no limit on the number of available bins.

The 2DVPP-PLC can be naturally formulated as a set cover problem. If a set of items can be loaded into a bin, we call it a *packing pattern*. Let $\mathcal{P}$ be the index set of all possible packing patterns. A packing pattern $j \in \mathcal{P}$ can be represented as a column vector $\mathbf{A}^j = (a_{1,j}, a_{2,j}, \ldots, a_{n,j})^T$ where $a_{i,j}$ is the number of type i items in pattern j. Let c_j be the cost of a bin packed according to pattern j, and let decision variable x_j be the number of bins packed according to pattern j. The following set cover problem is solved to obtain the optimal solution to the 2DVPP-PLC:

$$\mathbf{SC}(\mathcal{P}): \quad \text{Minimize} \quad \sum_{j \in \mathcal{P}} c_j x_j \tag{1}$$

$$\text{Subject to} \quad \sum_{j \in \mathcal{P}} a_{i,j} x_j \geq d_i, \quad i = 1, 2, \ldots, n \tag{2}$$

$$x_j \geq 0, \quad \forall\ j \in \mathcal{P} \tag{3}$$

$$x_j \text{ integer}, \quad \forall\ j \in \mathcal{P} \tag{4}$$

where constraints (2) ensure that each item is packed into some bins.

4. A branch-and-price algorithm

There are two challenges in solving the set cover model. First, it is an integer program, which is much harder to solve than a linear program. Second, there is an exponential number of packing patterns; thus, it is impractical to generate all packing patterns in advance.

A branch-and-bound algorithm is employed to overcome the first challenge. The idea is to drop the integral constraint (4) and solve the linear relaxation. If there is no feasible solution to the linear relaxation, we conclude that there is no feasible solution to the original problem. If the solution is integral, it is also an optimal solution to the 2DVPP-PLC. Otherwise, we try to divide the linear program into two subproblems such that: (1) the fractional solution is not in the domain of either subproblem and (2) the domain of one of the subproblems contains an optimal integral solution to

Download English Version:

https://daneshyari.com/en/article/4959878

Download Persian Version:

https://daneshyari.com/article/4959878

Daneshyari.com