Computers & Operations Research 69 () 56–67

Contents lists available at ScienceDirect

Computers & Operations Research

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

Bin packing and related problems: General arc-flow formulation with graph compression

Filipe Brandão*, João Pedro Pedroso

INESC TEC and Faculdade de Ciências, Universidade do Porto, Portugal

ARTICLE INFO

Available online 27 November 2015

Keywords:
Bin packing
Cutting stock
Vector packing
Arc-flow formulation

ABSTRACT

We present an exact method, based on an arc-flow formulation with side constraints, for solving bin packing and cutting stock problems—including multi-constraint variants—by simply representing all the patterns in a very compact graph. Our method includes a graph compression algorithm that usually reduces the size of the underlying graph substantially without weakening the model.

Our formulation is equivalent to Gilmore and Gomory's, thus providing a very strong linear relaxation. However, instead of using column-generation in an iterative process, the method constructs a graph, where paths from the source to the target node represent every valid packing pattern.

The same method, without any problem-specific parameterization, was used to solve a large variety of instances from several different cutting and packing problems. In this paper, we deal with vector packing, bin packing, cutting stock, cardinality constrained bin packing, cutting stock with cutting knife limitation, bin packing with conflicts, and other problems. We report computational results obtained with many benchmark test datasets, some of them showing a large advantage of this formulation with respect to the traditional ones.

1. Introduction

The bin packing problem (BPP) is a combinatorial NP-hard problem (see, e.g., [10]) in which objects of different volumes must be packed into a finite number of bins, each with capacity W, in a way that minimizes the number of bins used.

The BPP can be seen as a special case of the cutting stock problem (CSP). In the CSP, items of equal size (which are usually ordered in large quantities) are grouped into orders with a required level of demand, while in the BPP the demand for a given size is usually close to one. According to Wäascher et al. [21], cutting stock problems are characterized by a weakly heterogeneous assortment of small items, in contrast with bin packing problems, which are characterized by a strongly heterogeneous assortment of small items.

The p-dimensional vector bin packing problem (pD-VBP), also called general assignment problem by some authors, is a generalization of bin packing with multiple constraints. In this problem, we are required to pack n items of m different types, represented by p-dimensional vectors, into as few bins as possible. In practice, this problem models, for example, static resource allocation problems where the minimum number of servers with known capacities is used to satisfy a set of services with known demands.

The method presented in this paper allows solving several cutting and packing problems through reductions to vector packing. The reductions are made by defining a matrix of weights, a vector of capacities and a vector of demands. Our method builds very strong integer programming models that can usually be easily solved using any state-of-the-art mixed integer programming (MIP) solver. Computational results obtained with many benchmark test datasets show a large advantage of our method with respect to traditional ones.

In this paper, instances for all the problems will be presented as follows: p is the number of dimensions; m is the number of different item types; b_i is the demand for items of type i; and, for each dimension d, w_i^d is the weight of item i and W^d is the bin capacity. For the sake of simplicity the dimension may be omitted in the one-dimensional case.

The remainder of this paper is organized as follows. Section 2 presents Valério de Carvalho's arc-flow formulation for bin packing and cutting stock, which provides the basis for our method. The general arc-flow formulation, which is a generalization of Valério de Carvalho's model, is presented in Section 3. Section 4

* Corresponding author.
E-mail addresses: fdabrandao@dcc.fc.up.pt (F. Brandão), jpp@fc.up.pt (J.P. Pedroso).

http://dx.doi.org/10.1016/j.cor.2015.11.009
0305-0548/

introduces the multidimensional graphs that can be used to model p-dimensional vector bin packing problems. Section 5 presents the graph compression algorithm, which is crucial for solving large-scale instances. Computational results are provided in Section 6 and Section 7 presents the conclusions.

2. Previous work on exact methods

Valério de Carvalho [19] provides an excellent survey on integer programming models for bin packing and cutting stock problems. In this section, we will introduce Valério de Carvalho's [18] arc-flow formulation, which is equivalent to Gilmore and Gomory's [12] formulation in terms of value of the linear relaxation. Gilmore and Gomory's model provides a very strong linear relaxation, but it is potentially exponential in the number of variables with respect to the input size; Valério de Carvalho's model is usually much smaller, being pseudo-polynomial in terms of decision variables and constraints.

2.1. *Valério de Carvalho's arc-flow formulation*

Among methods for solving BPP and CSP exactly, one of the most important is the arc-flow formulation with side constraints used by Valério de Carvalho [18]. This model has a set of flow conservation constraints and a set of demand constraints to ensure that the demand of every item is satisfied. The corresponding path-flow formulation is equivalent to the classical Gilmore and Gomory's [12] formulation.

Wolsey [22] proposed for the first time an arc-flow formulation for cutting and packing problems. Despite not presenting computational results, some properties that suggested computational advantages of such formulation were presented. For solving cutting and packing problems, computational experiments with arc-flow formulations were only performed much later by Valério de Carvalho [18], where an arc-flow formulation was used as a basis to produce a branch-and-price algorithm.

In the one-dimensional case, the problem of determining a valid solution to a single bin can be modeled as the problem of finding a path in a directed acyclic graph $G=(V,A)$ with $V=\{0,1,2,\ldots,W\}$ and $A=\{(i,j) \mid j-i=w_t,\ \text{for } t=1..m \text{ and } 0 \le i < j \le W\}$, meaning that there exists an arc between two vertices i and $j>i$ if there are items of size $w_t=j-i$. The number of vertices and arcs is bounded by $\mathcal{O}(W)$ and $\mathcal{O}(mW)$, respectively. Additional arcs $(k,k+1)$, for $k=w_{min},\ldots,W-1$, with w_{min} being the minimum item width, are included for representing unoccupied portions of the bin.

In order to reduce the symmetry of the solution space and the size of the model, Valério de Carvalho introduced some rules. The idea is to consider only a subset of arcs from A. If we search for a solution in which the items are ordered by decreasing values of weight, only paths in which items appear according to this order must be considered.

Example 1. Fig. 1 shows the graph associated with an instance with capacity $W=7$ and items of sizes 5, 3, 2 with demands 3, 1, 2, respectively.

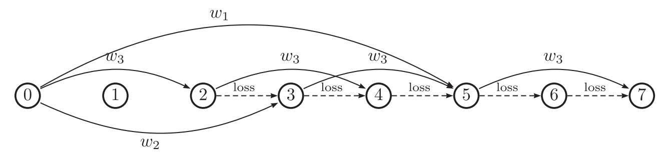

Fig. 1. Graph corresponding to Example 1.

The BPP and the CSP are thus equivalently formulated as that of determining the minimum flow between vertex 0 and vertex W, with additional constraints enforcing the sum of the flows in the arcs for each item type to be greater than or equal to the corresponding demand. Consider decision variables x_{ij} (associated with arcs (i,j) defined above) corresponding to the number of items of size $j-i$ placed in any bin at a distance of i units from the beginning of the bin. A variable z, representing the number of bins required, aggregates the flow in the graph, and can be seen as a feedback arc from vertex W to vertex 0. The model is as follows:

$$\text{minimize} \quad z \tag{1}$$

$$\text{subject to} \quad \sum_{(i,j)\in A: j=k} x_{ij} - \sum_{(i,j)\in A: i=k} x_{ij} = \begin{cases} -z & \text{if } k=0, \\ z & \text{if } k=W, \\ 0 & \text{for } k=1,\ldots,W-1, \end{cases} \tag{2}$$

$$\sum_{(i,j)\in A: j=i+w_k} x_{ij} \ge b_k, \quad i=k,\ldots,m, \tag{3}$$

$$x_{ij} \ge 0, \ \text{integer}, \quad \forall (i,j)\in A. \tag{4}$$

3. General arc-flow formulation

In this section, we propose a generalization of Valério de Carvalho's arc-flow formulation. The formulation is the following:

$$\text{minimize} \quad z \tag{5}$$

$$\text{subject to} \quad \sum_{(u,v,i)\in A: v=k} f_{uvi} - \sum_{(v,r,i)\in A: v=k} f_{vri} = \begin{cases} -z & \text{if } k=\textsc{s}, \\ z & \text{if } k=\textsc{t}, \\ 0 & \text{for } k\in V\setminus\{\textsc{s},\textsc{t}\}, \end{cases} \tag{6}$$

$$\sum_{(u,v,j)\in A: j=i} f_{uvj} \ge b_i, \quad i\in\{1,\ldots,m\}\setminus J, \tag{7}$$

$$\sum_{(u,v,j)\in A: j=i} f_{uvj} = b_i, \quad i\in J, \tag{8}$$

$$f_{uvi} \le b_i, \quad \forall (u,v,i)\in A, \ \text{if } i\ne 0, \tag{9}$$

$$f_{uvi} \ge 0, \text{integer}, \quad \forall (u,v,i)\in A, \tag{10}$$

where V is the set of vertices, $\textsc{s}$ is the source vertex and $\textsc{t}$ is the target; A is the set of arcs, where each arc has three components (u,v,i) corresponding to an arc between nodes u and v that contributes to the demand for items of type i; arcs (u,v,i) with $i=0$ are the loss arcs; f_{uvi} is the amount of flow along the arc (u,v,i); and $J\subseteq\{1,\ldots,m\}$ is a subset of items for which we decide that demands are required to be satisfied exactly, for efficiency purposes. For having tighter constraints, one may set $J=\{i=1,\ldots,m \mid b_i=1\}$ (we have done this in our experiments); but the optimum for $J=\varnothing$ is the same.

In Valério de Carvalho's model, a variable x_{ij} contributes to an item with weight $j-i$. In our model, a variable f_{uvi} contributes to items of type i; the label of u and v may have no direct relation to the item's weight. This new model is more general; Valério de Carvalho's model is a sub-case, where an arc between nodes u and v can only contribute to the demand of an item of weight $v-u$. As