Computers & Operations Research 76 (2016) 12–21

Contents lists available at ScienceDirect

Computers & Operations Research

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

Consistent neighborhood search for one-dimensional bin packing and two-dimensional vector packing

Mirsad Buljubašić *, Michel Vasquez

Ecole des Mines d'Alès, LGI2P Laboratory, Nîmes, France

ARTICLE INFO

Article history:
Received 24 April 2015
Received in revised form
9 June 2016
Accepted 12 June 2016
Available online 16 June 2016

Keywords:
Tabu search
Consistent neighborhood
Bin packing
Vector bin packing

ABSTRACT

We propose a consistent neighborhood search approach for solving the one-dimensional bin packing problem (BPP). The goal of this local search is to derive a feasible solution with a given number of bins, m, starting from $m = UB - 1$, where UB is an upper bound obtained by using a variant of the classical First Fit heuristic. To this end, the local search was performed on a partial solution with $m - 2$ bins, i.e. a solution containing a subset of items packed into $m - 2$ bins without capacity violations and a set of non-assigned items, with the objective of minimizing the total weight of non-assigned items and, ultimately, packing all the non-assigned items into two bins. A partial solution was constructed by deleting bins from the last complete solution. Local moves include rearranging the items assigned to a single bin along with non-assigned items, i.e. removing and adding items to the bin. A tabu search was performed with moves featuring a limited number of items to be added/dropped, plus a hill climbing/descent procedure with general (unlimited) add/drop moves, in order to minimize a given objective function. The very same procedure was used for all instances under consideration, with the same initial solution, same parameters, same order of neighborhood exploration, etc. Promising results were obtained for a wide range of benchmark instances; solutions equal to or better than the best known solutions found by heuristic methods were obtained for all the instances considered, successfully outperforming published results for the particular class of instances hard28, which appears to cause the greatest degree of difficulty for BPP algorithms. The method was also tested on the vector packing problem (VPP) and evaluated for classical two-dimensional VPP (2-DVPP) benchmarks, in all instances yielding optimal or best-known solutions.

1. Introduction

Given a set $I = \{1, 2, \ldots, n\}$ of items with associated weights w_i ($i = 1, \ldots, n$), the bin packing problem (BPP) consists in finding the minimum number of bins, of capacity C, required to pack all the items without violating any of the capacity constraints. In other words, the goal is to find a partition of items $\{I_1, I_2, \ldots, I_m\}$ such that

$$\sum_{i \in I_j} w_i \le C, \quad j = 1, \ldots, m$$

and m is minimum. The bin packing problem is known to be NP-hard [10]. It is one of the most extensively studied combinatorial problems and has a wide range of practical applications such as storage allocation, cutting stock, multiprocessor scheduling and the loading of flexible manufacturing systems, to name a few. The Vector Packing problem (VPP) is a generalization of BPP with multiple resources. Item weights w_i^r and bin capacities C_r are defined for each resource $r \in \{1, \ldots, R\}$, and the following constraint must be satisfied:

$$\sum_{i \in I_j} w_i^r \le C_r, \quad r = 1, \ldots, R, j = 1, \ldots, m.$$

Without loss of generality, we assume that capacities and weights are integer-valued.

This paper will present a new improvement heuristic based on a local search for solving BPP and VPP with two resources (2-DVPP). The method will first be described in detail for a BPP problem, followed by adaptations required to solve the 2-DVPP. The solution is iteratively improved by decreasing the number of bins being utilized. The procedure works as follows. First, the upper bound on the solution value, UB, is obtained by a variant of the First Fit heuristic. Next, an attempt is made to find a feasible solution with $UB - 1$ bins, and this process continues until the lower bound, time limit or maximum number of search iterations

* Corresponding author.
E-mail addresses: mirsad.buljubasic@mines-ales.fr (M. Buljubašić), michel.vasquez@mines-ales.fr (M. Vasquez).

http://dx.doi.org/10.1016/j.cor.2016.06.009
0305-0548/

is reached. Apart from the simple lower bound, $\left\lceil \frac{\sum_{i=1}^{n} w_i}{C} \right\rceil$, other lower bounds developed by Fekete and Schepers [7], Martello and Toth [13] (bound L_3) and Alvim et al. [1] are also used.

In order to find a feasible solution with a given number of bins, $m < UB$, a local search is employed. As opposed to the majority of papers published on BPP, the local search explores partial solutions that consist of a set of assigned items without any capacity violation and a set of non-assigned items. The moves rearrange the items assigned to a single bin along with non-assigned items, i.e. items are removed and added to the bin. The objective here is to minimize the total weight of non-assigned items. This local search on partial configurations is called the Consistent Neighborhood Search (since only valid partial packings are considered). It has been proven efficient on several combinatorial optimization problems [22,24]. Our approach will therefore be referred to as CNS_BP (Consistent Neighborhood Search for Bin Packing) in the remainder of the paper.

This search space of partial solutions is explored in two successive phases: (1) a tabu search with limited add/drop moves and (2) a descent with a general add/drop move. This sequence terminates when a complete solution is found or the running time limit or maximum number of iterations is reached. Additionally, the algorithm makes use of a simple reduction procedure that consists in fixing the assignments of all pairs of items that can fill an entire bin. More precisely, once a set of item pairs (i,j) such that $w_i + w_j = C$ is identified, the problem can be reduced by deleting those items (or setting their assignments). This same reduction has been used in most papers on BPP. It is important to mention that the reduction procedure does not have a significant influence on the final results (but can speed up the search) and that no reduction is possible for a large proportion of the instances considered.

This paper is organized as follows. Section 2 will describe relevant work. Then, our approach will be introduced in Section 3. The general framework will be presented first, followed by a description of all its algorithmic components. A number of critical remarks and parameter choices will be discussed in Section 4. Section 5 presents a summary of methodological adaptations to solve the 2-DVPP. The results of extensive computational experiments performed on the available set of instances, for both the BPP and 2-DVPP, will be provided in Section 6, followed by a conclusion.

2. Relevant work

2.1. BPP

There is a large body of literature concerning the one-dimensional bin packing problem. Both exact and heuristic methods have been applied for solving the problem. Martello and Toth [13] proposed a branch-and-bound procedure (MTP). Scholl et al. [17] developed a hybrid method (BISON) that combines a tabu search with a branch-and-bound procedure based on several bounds and a new branching scheme. Schwerin and Wäscher [18] offered a new lower bound for the BPP based on the cutting stock problem, then integrated this new bound into MTP and achieved high-quality results. Valerio de Carvalho [21] proposed an exact algorithm using column generation and branch-and-bound.

Gupta and Ho [11] introduced a minimum bin slack (MBS) constructive heuristic. At each step, a set of items that fits the bin capacity as tightly as possible is identified and packed into the new bin. Fleszar and Hindi [9] developed a hybrid algorithm that combines a modified version of the MBS and the Variable Neighborhood Search. Their hybrid algorithm performed well in computational experiments, by producing the optimal solution for 1329 out of the 1370 instances considered (the first two classes of instances to be discussed in Section 6.1).

Alvim et al. [1] presented a hybrid improvement heuristic (HI_BP) that uses tabu search to move the items between bins. In their algorithm, a complete yet infeasible configuration is to be repaired through a tabu search procedure. Simple “shift and swap” neighborhoods are explored, in addition to balancing/unbalancing the use of bin pairs by solving a Maximum Subset Sum problem. HI_BP performed very well, having obtained the optimal solution for 1582 out of the 1587 instances considered (the first four classes of instances to be discussed in Section 6.1).

In recent years, several competitive heuristics have been presented with results similar to those obtained by HI_BP. Singh and Gupta [19] proposed a compound heuristic (C_BP) which combines a hybrid steady-state grouping genetic algorithm with an improved version of Fleszar and Hindi's Perturbation MBS. Loh et al. [12] developed a weight annealing (WA) procedure, by relying on the concept of weight annealing to expand and accelerate the search by creating distortions in various parts of the search space. The proposed algorithm is simple and easy to implement; moreover, the authors reported high-level performances, exceeding those obtained by HI_BP.

Fleszar and Charalambous [8] offered a modification to the Perturbation-MBS method [9] where a new sufficient average weight (SAW) principle is introduced to control the average weight of items packed in each bin (referred to as Perturbation-SAWMBS). This heuristic outperformed the best state-of-the-art HI_BP, C_BP and WA algorithms. The authors also reported significantly lower quality results for the WA heuristic compared to those given in Loh et al. [12].

To the best of our knowledge, the most recent work in this area, is reported in Quiroz-Castellanos et al. [16]. It involves a grouping genetic algorithm (GGA-CGT) that outperforms all previous algorithms with regard to the number of optimal solutions found, particularly for the most difficult set of instances hard28. The authors propose a new set of grouping genetic operators to promote the transmission of the best genes in the chromosomes. A new reproduction technique that controls the exploration of the search space is also presented, as well as a variant of the First Fit procedure for producing a high-quality initial population.

Brandão and Pedroso [3] devised an exact approach for solving the bin packing and cutting stock problems based on an Arc-Flow Formulation of the problem which is then solved with the commercial Gurobi solver. They were able to optimally solve all standard bin packing instances within a reasonable computation times, including those instances that were not solved to optimality by any heuristic method.

2.2. VPP

With regard to the two-dimensional VPP, Spieksma [20] proposed a branch-and-bound algorithm, while Caprara and Toth [4] reported exact and heuristic approaches as well as a worst-case performance analysis. A heuristic approach using a set-covering formulation was presented by Monaci and Toth [15]. Masson et al. [14] proposed an iterative local search (ILS) algorithm for solving the Machine Reassignment Problem and VPP with two resources; they reported the best results for the classical VPP benchmark instances of Spieksma [20] and Caprara and Toth [4].

3. Proposed heuristic

This section will describe our improvement heuristic. The main part of the improvement procedure is illustrated in Fig. 1, while