A new framework to solve a numerous number of items packing by applying PSO

Group 16

Josip Koprcina s1062758

Shokrzad, Reza s1056369

Kulkarni, Pratik s1051211

April 2021

1 Problem

The bin packing problem (BPP) is a combinatorial NP-hard problem. A classic problem with remarkable applications in the industry. The objective would be to pack different-volume items into a minimum number of bins (containers and trucks in real-life applications). Moreover, deciding whether a set of items layouts into a restricted number of bins in the first place is an NP-complete problem. 2D packing, linear packing, packing by weight, packing by cost are some instances of this problem that have been worked on to find the optimal solution. There is a list of applications, such as filling up containers, loading trucks with weight capacity constraints, creating file backups in media, and technology mapping in field-programmable gate array semiconductor chip design[1]. We thought about an application of this problem in the primary level of goods logistics. For instance, carrying a huge number of loads from Amazon's main inventory to other inventories in different cities. In this way, the order of items is not critical. Otherwise, the objective would be to minimize the free space of each truck and as a consequence, possibly lowering the number of trucks needed. Since the problem's size is so large, it is not solvable by mathematics algorithms. Although by using PSO we cannot get access to a global optimum, having a sub-optimal answer concluded by PSO can be beneficial for enterprises because it should beat the current random packing method.

2 Methods

The problem is 3-dimensional. We need to consider it as a search problem to apply PSO such that its *fitness function* should be similar to equation [1] in which the term ($\#bins$) stands for the number of trucks, while the second part of the expression calculates the amount of free space.

$$Min\ (F = \lambda_1(\#bins) + \lambda_2(V_{bins} - \sum V_{boxes})) \tag{1}$$

To provide a *dataset*, there is a dataset mentioned in some papers named 'hard28', that is apparently not publicly available. We are going to send an email to the authors to see whether we can have access to it. Apart from that, we have an idea to generate an acceptable random dataset. We can consider the optimum answer, then create items from the extreme case by cutting them in some rules. Then, feed the model with shuffled items. In this way, we can *evaluate* our model by comparing the extreme case and the solution. In this work, we need to simplify the model to strengthen the algorithm for finding better solutions. Probably, we would reduce the scale by attaching same-size boxes, guessing the possible answers to eliminate some of them, and other similar actions.

3 Literature Review

Apart from dimension, the bin packing problem is an NP-hard problem, the reason why it would not solve in practice. Adam tried to address the one-dimensional bin packing problem by applying evolutionary-based heuristic algorithms [2]. Unlike other evolutionary heuristics used with optimization problems, their proposed algorithm uses a modified permutation with separators encoding scheme, a unique concept of separators' movements during mutation, and separators removal as a technique of problem size reduction. Furthermore, Eva has proposed a new approach to solving two-dimensional packing (Figure 1), especially in the textile and leather industry, by utilizing evolutionary algorithms and other meta-heuristic methods [3]. Afterward, Silviano and et al worked on a three-dimensional orthogonally packing a given set of rectangular-shaped items into the minimum number of three-dimensional (Figure 2) rectangular bins [4]. They used an exact branch-and-bound algorithm that incorporates original approximation algorithms. Additionally, there are many recent works that focused on expanding the application in the industry by considering more constraints than before. Ali and et al, studied the Bin Packing Problem with Conflicts and Item Fragmentation (BPPC-IF) which has applications in the delivery and storage of items that cannot be packed together [5]. In this practice, BPPC-IF's goal is to pack these items into a minimum number of fixed-capacity bins while not packing fragments of conflicting items into the same bin.

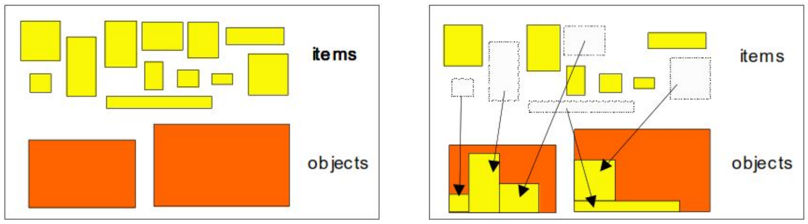

Figure 1: two dimensional packing task

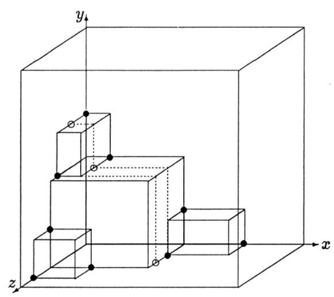

Figure 2: three dimensional packing task

References

[1] `https://en.wikipedia.org/wiki/Bin_packing_problem/`.

[2] Stawowy, A. (2008) 'Evolutionary based heuristic for bin packing problem', Computers and Industrial Engineering, 55(2), pp. 465–474.

[3] Hopper, E. (2000) 'Two-dimensional Packing utilising Evolutionary Algorithms and other Meta-Heuristic Methods A Thesis submitted to the University of Wales for the Degree of Doctor of Philosophy by', Test, (May).

[4] Martello, S., Pisinger, D. and Vigo, D. (2000) 'Three-dimensional bin packing problem', Operations Research, 48(2), pp. 256–267.

[5] Ekici, A. (2021) 'Bin Packing Problem with Conflicts and Item Fragmentation', Computers and Operations Research, 126, p. 105113.