Combining operation research with deep reinforcement learning methods to solve the 3D Bin packing problem

Farès CHOUAKI[1], Rebecca Siân JEFFERS[2],

[1] Sorbonne Université, CNRS, LIP6, F-75005 Paris, France
fares.chouaki@lip6.fr
[2] InstaDeep, 40 bis Rue du Faubourg Poissonnière, 75010 Paris
r.jeffers@instadeep.com

Mots-clés : *3D Bin Packing, Reinforcement learning, Dantzig-Wolfe Decomposition*

1 Introduction

The Three-dimensional Bin Packing Problem (3D-BPP), is the problem where we are interested in packing an assortment of heterogeneous cuboid items into an assortment of containers. This problem is strongly NP-hard [5], thus the existence of an algorithm that can solve large instances and guarantee optimality in a reasonable amount of time is unlikely. Many variants of this problem exist where more real-world constraints are added. This work is concerned with the variant where placed items need to be fully supported inside of the container and items are allowed to be rotated by 90 degrees along each axis of the container. Several exact formulations of this problem exist and can be used to solve small instances of said problem, whereas larger instances are solved using heuristic solvers. This paper aims to :

1. Suggest a way of providing lower bounds on the value of the optimal solution for instances of the 3D-BPP.
2. Combine deep reinforcement learning and exact optimization techniques to improve the quality of the solutions found by heuristic solvers.

2 Tight lower bounds using exact branch-&-bound and Dantzig-Wolfe Decomposition

To provide better lower bounds, the initial problem is decomposed using Dantzig-Wolfe decomposition and then solved using the **column generation** algorithm. The Dantzig-Wolfe decomposition of the 3D-BPP results in two sub-problems, a set partitioning problem called the restricted master problem (RMP) and a 3D knapsack problem called the subproblem.

To solve the restricted master problem the formulation proposed by [2] is used. The 3D knapsack problem is solved using the formulation proposed by [4].

3 Reinforcement learning as a heuristic solver for the 3D-BPP

To provide solutions to large instances of the 3D-BPP we still use Dantzig-Wolfe decomposition and column generation. However, instead of solving the 3D knapsack problem exactly, a pre-trained reinforcement learning agent is used. This agent is trained on an extended version of the BinPack environment of the Jumanji suite [1]. Once the agent is trained, it is integrated into the column generation algorithm in two places. It is first used to warm start the algorithm where the agent is used to provide an initial solution to the instance by having it pack containers one by one until all the items are packed. It is also used as a solver for the 3D knapsack problem at each iteration of the column generation algorithm.

4 Results

The solver and the lower bound generator are tested on the dataset proposed by [3] comprising 47 instances of the problem, and the results are compared to the best-known results for each instance.

The lower bounds found by our method prove that the best-known solutions to 39 instances are optimal, these lower bounds are on average 14.5% tighter than the liquid volume lower bound. The solver combining column generation and RL is able to find 25 of those optimal solutions and achieves state-of-the-art performances on 29 instances. Furthermore, this solver can find solutions that use 19.5% less containers than the solver that uses the RL agent sequentially.

5 Discussion

Based on the results found, it can be observed that combining column generation and RL is a way of improving the results obtained by an RL algorithm, integrating the RL agent in a column generation algorithm makes the algorithm faster and reduces the impact of the sequentiality on the solutions provided by the RL algorithm. Our approach also allows us to significantly reduce the episode length and the size of the action space of the agent compared to an agent that solves the full problem.

Although the results obtained with the solver proposed in this study were not as good as those found by the best-known heuristics, the advantage of such a solver is that it can be retrained to work on problems with different constraints. In fact, hand-crafted heuristics become useless as soon as a new constraint is added to the problem. The results of the solver can also be improved by allowing more training or inference time.

6 Perspectives

An interesting extension of this work is to compare the proposed solver against an agent trained to solve the 3D bin packing problem instead of an agent that solves a 3D knapsack problem repeatedly. The resulting patterns found by that agent could also be used to warm start the column generation solver. Another way of combining RL and combinatorial optimization would be to use a combination of MILP solver, constraints programming solver, and deep reinforcement learning as a solution for more constrained versions of the problem by restricting the set of items visible to the RL agent using constraint programming.

Références

[1] Bonnet, Clément, et al. "Jumanji : a Diverse Suite of Scalable Reinforcement Learning Environments in JAX." arXiv preprint arXiv :2306.09884 (2023).

[2] Eley, Michael. "A bottleneck assignment approach to the multiple container loading problem." OR Spectrum 25 (2003) : 45-60.

[3] Ivancic, Nancy J. An integer programming based heuristic approach to the three dimensional packing problem. Diss. Case Western Reserve University, 1988.

[4] Kurpel, Deidson Vitorio, et al. "The exact solutions of several types of container loading problems." European Journal of Operational Research 284.1 (2020) : 87-107.

[5] Martello, Silvano, David Pisinger, and Daniele Vigo. "The three-dimensional bin packing problem." Operations research 48.2 (2000) : 256-267.