1 The problem

This project is concerned with rectangle packing in general, and in particular with the TWO-DIMENSIONAL BIN PACKING problem (2BP), a fundamental topic of combinatorial optimization. 2BP calls for packing a given set of rectangles of different sizes (the *items*) into as less as possible large rectangles of identical size (the *bins*). Two versions of this problem are often considered, according to whether items can or cannot be rotated. If each item i is required in d_i units, one speaks of the CUTTING STOCK problem (2CS).

When solving 2BP or 2CS by column generation, each item is associated with a profit, and one wants to pack a single bin with items of maximum total profit (*pricing problem*). This is commonly known as TWO-DIMENSIONAL KNAPSACK (2K), and has various versions not only depending on item rotation (be it allowed or not), but also on how many copies of each item can be selected: one then has the INTEGER, the BOUNDED and the 0-1 2K. The special case of 2K with identical item profits is interesting by itself, and is referred to as TWO-DIMENSIONAL ORTHOGONAL PACKING (2OP). Both 2K and 2OP (and their generalizations to dimension $d > 2$) are NP-hard [Garey and Johnson, 1979] and have a wide range of applications in such fields as telecommunications, logistics, and manufacturing.

1.1 Lower and upper bounds

Several methods for computing lower and upper bounds have been proposed for 2K: the former are values of feasible solutions returned by primal heuristics, the latter of infeasible solutions computed by relaxations. Upper bounds by Langragian relaxation and subgradient optimization have been proposed by [Beasley, 1985] and [Christofides and Hadjiconstantinou, 1995]. The upper bound in [Belov et al., 2009] is obtained by relaxing item connectivity, so transforming each d-dimensional object ($d \geq 2$) into many 1-dimensional strips of various widths w_i. Packing strips becomes then a 1CS-like problem with limited bin availability. Bound is enforced by requiring contiguity of bins containing strips from the same item.

Lower and upper bounds of 2CS or 2BP are associated with relaxations and primal solutions, respectively. *Dual Feasible Functions* (DFF) [Fekete et al., 2006] are introduced to determine lower bounds to CS or BP more quickly than solving the well known Gilmore-Gomory LP [Gilmore and Gomory, 1961-63] by column generation. A DFF is a function $f : [0,1] \to [0,1]$ such that, for any finite set S of real numbers in $[0,1]$,

$$\sum_{y \in S} y \leq 1 \Rightarrow \sum_{y \in S} f(y) \leq 1$$

The role of DFF is explained by taking the dual LP of the Gilmore-Gomory CS formulation:

$$\max \sum_{i \in I} d_i \pi_i \qquad (1)$$

$$\sum_{i \in I} a_{ik} \pi_i \leq 1 \qquad \forall k \in K \qquad (2)$$

$$\pi_i \geq 0 \qquad \forall i \in I \qquad (3)$$

where K denotes the set of the (exponentially many) feasible patterns and I the set of items. Then any collection $C \subseteq I$ of items that fit a 1-dimensional normalized pattern k can be associated with the real numbers $y = w_i$, $i \in C$, that gives the item widths. Since k is a feasible pattern, $\sum_{i \in C} a_{ik} w_i \leq 1$, and so $\sum_{y \in S} y \leq 1$, where S contains a_{ik} copies of the i-th item width, $i \in C$. On the other hand, if π_i is dual feasible, then by (2) $\sum_{i \in C} a_{ik} \pi_i = \sum_{w_i \in S} \pi_i \leq 1$. Therefore $f(y) = \pi_i$ for $y = w_i$ is a DFF. In general, a DFF lower bound to CS cannot be stronger than the LP relaxation, but as we just saw there always exists a DFF (the dual optimum) that attains that bound.

To obtain the bounds [Carlier et al., 2007] used a discrete version of DFF and proposed a new class of Data-Dependent DFF.

2 Objective of research

A first objective of our study is to provide a comprehensive review of the literature and industry practices involving the described packing problems, and outline a conceptual framework for solution algorithms.

The long-term goal is to study and develop methodologies that can improve the efficiency of computing optimal values. Getting quick and good bounds to optimal values is a key issue to fasten optimality certification, so we plan to initially focus on upper bounds for 2K/2OP and lower bounds for 2BP/2CS. We hope that the result of our research will be valuable for industries as well as for other fields where packing problems find a relevant application.

2.1 Proposed work

The project is the continuation of my master thesis. In the thesis, I studied LP bounds for 2K by 1CS relaxation of topological constraints. This relaxation does not remove inequalities from a particular formulation, but rather weakens a geometrical property of items, that is connectivity. A further relaxation is done by removing the integrality constraint while solving the 1CS problem, so as to compute the upper bound by Dantzig-Wolfe decomposition of a linear program (column generation): the pricing problems solved to generate promising columns take the form of 01 or BOUNDED INTEGER 1K. Two directions of research can at this point be sketched, with the purpose of:

- strengthening the bound, and/or
- fastening bound computation.

The bound can be strengthened in various ways: by adding valid inequalities in order to cut away fractional LP optima; by adding inequalities that re-introduce relaxed topological constraints (such as pattern contiguity); by solving the integer program via *Branch-and-price* (B&P).

On the other hand, because upper bounds for 2K can be obtained by solving 1CS-like problems, we wish to investigate if and how DFF can play a role in fastening computation.

Branch-and-price. When the 1CS-like formulation (whose solution yields the upper bound) has exponentially many columns, optimal integer solutions can be found by branch-and-price. The performance of a B&P procedure inherently depends on the quality of the upper and lower bounds found at each node, and more importantly, relies on the use of computer resources (branching strategies, search strategies, column management etc.).

In general terms, B&P uses bounds from LP relaxations solved by column generation at branching nodes. While doing B&P, a non-basic column of the *Restricted Master Problem* (RMP) that appears in a branch is not necessarily non-basic in other branches. So, a well-thought-out problem-specific heuristic column management that makes use of knowledge from the original problem would arguably improve the algorithm performance. Another important aspect where the performance of B&P can be improved is the branching scheme. Intuitively, the B&P search tree is heuristically constructed via *node-picking* and *branching* rules. The node-picking rule decides where in the tree further branching or pruning should be done, and a bad rule can give rise to time-consuming branches. Branching rules have been investigated by [Achterberg et al., 2005], and hybrid ones by [Achterberg and Berthold, 2009]. Problem-specific branching and node-picking rules can be investigated and developed in the present context where we do not refer to a standard 1CS, but rather to a 1CS-like problem.

Also the use of computer resources can have a tremendous effect on algorithm performance. Nowadays, almost every processor is available with multiple cores. Hence, it is natural to thread up independent subproblems to multiple cores, so as to solve them in parallel. A more sophisticated approach would solve multiple B&P nodes in parallel, with a caveat that, although parallel computing of independent subproblems is intuitive, parallelization of B&P is quite involved: so, how to distribute local information to parallel processes can be the interesting direction of research.

Topological constraints. Topological constraints such as pattern contiguity are not easy to be encoded in a pattern-based 1CS-like model, but can be better introduced using an assignment model close to that commonly attributed to Kantorovich: decision variables x_i^t get value 1 if a strip from item i of the 2-dimensional problem is assigned to the t-th (1-dimensional) bin, and 0 otherwise. Assignment constraints require that each bin is packed within normalized unit capacity ($\sum_i w_i x_i^t \leq 1, \forall t$), and that the total area of strips from item i that are assigned to bins equals the item area ($\sum_t x_i^t = h_i x_i, \forall i$, where $x_i = 1$ iff item i is selected and 0 otherwise). It is then easy to express contiguity by

$$x_i^s - x_i^t + x_i^{t+1} \leq 1 \qquad \text{or} \qquad x_i^s - 2x_i^t + x_i^{t+1} \leq 0$$

for all items i and bins $s < t$. Both inequalities force in fact a strip from item i to be assigned to the t-th bin if one from the same item is assigned to the s-th and the $(t+1)$-st bin. A possible direction of research investigates polyhedral properties of the integer LP so constructed.

Dual Feasible Functions. Another interesting area of research can deal with the use of DFF to fasten bound computation. DFF [Alves et al., 2016] provide dual feasible solutions of models whose corresponding bounds are often very close to those provided by column generation. As one can design a DFF to be computed quickly, the computational burden can be very small compared to that obtained by column generation. Furthermore, for a given problem, it is often possible to derive not only a single DFF but several DFF, or even families of different DFF, providing possibly different bounds from which taking the best one.

References

[Achterberg and Berthold, 2009] Achterberg, T. and Berthold, T. Hybrid branching. In: van Hoeve, W.-J. and Hooker, J. N. (editors), *Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems*, Springer, Berlin Heidelberg (2009) 309-311

[Achterberg et al., 2005] Achterberg, T., Koch, T., and Martin, A. Branching rules revisited. *Oper. Res. Lett.*, **33**, 1 (2005) 42-54

[Alves et al., 2016] Alves, C., Clautiaux, F., Carvalho, J., and Rietz, J. Dual-feasible functions for integer programming and combinatorial optimization: basics, extensions and applications. @@@ INCOMPLETE REFERENCE (2016) @@@ PAGES

[Beasley, 1985] Beasley, J. An exact two-dimensional non-guillotine cutting tree search procedure. @@@ MISSING JOURNAL NAME **33** (1985) 49-64

[Belov et al., 2009] Belov, G., Kartak, V., MeiSSner, H., and Scheithauer, G. One dimensional relaxations and lp bounds for orthogonal packing. @@@ MISSING JOURNAL NAME **16** (2009) 745-766

[Carlier et al., 2007] Carlier, J., Clautiaux, F., and Moukrim, A. New reduction procedures and lower bounds for the two-dimensional bin packing problem with fixed orientation. @@@ MISSING JOURNAL NAME **34** (2007) 2223-2250

[Christofides and Hadjiconstantinou, 1995] Christofides, N. and Hadjiconstantinou, E. An exact algorithm for orthogonal 2-d cutting problems using guillotine cuts. @@@ MISSING JOURNAL NAME **83** (1995) 21-38

[Fekete et al., 2006] Fekete, S. P., Schepers, J., and van der Veen, J. An exact algorithm for higher-dimensional orthogonal packing. @@@ INVALID OR INCOMPLETE REFERENCE *CoRR*, abs/cs/0604045 (2006) @@@ PAGES

[Garey and Johnson, 1979] Garey, M. R. and Johnson, D. S. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA (1979)

[Gilmore and Gomory, 1961-63] Gilmore, P. and Gomory, R. A linear programming approach to the cutting stock problem. Parts I and II. *Operations Research* **9** (1961) 849-859 and **11** (1963) 863-888