A Value-Correction Construction Heuristic for the Two-Dimensional Cutting Stock Problem with Variable Sheet Size

Frederico Dusberger, Günther R. Raidl

Institute of Computer Graphics and Algorithms
Vienna Unviersity of Technology, Vienna, Austria
{dusberger|raidl}@ads.tuwien.ac.at

1 Introduction

Cutting and packing problems occur in many real-world applications such as industrial glass, paper or steel cutting, container loading and VLSI design [1]. In this work we consider in particular the two-dimensional cutting stock problem with variable sheet size (2CSV) in which we are given a set of n_E rectangular *element types* $E = \{1, \ldots, n_E\}$, each $i \in E$ specified by a height $h_i \in \mathbb{N}^+$, a width $w_i \in \mathbb{N}^+$ and a demand $d_i \in \mathbb{N}^+$. Furthermore, we have a set of n_T *stock sheet types* $T = \{1, \ldots, n_T\}$, each $t \in T$ specified by a height $H_t \in \mathbb{N}^+$, a width $W_t \in \mathbb{N}^+$, an available quantity $q_t \in \mathbb{N}^+$ and a cost factor $c_t \in \mathbb{N}^+$. Both elements and sheets can be rotated by 90°. The objective is to find a set of *cutting patterns* $\mathcal{P} = \{P_1, \ldots, P_n\}$, i.e. an arrangement of the elements specified by E on the available stock sheets specified by T without overlap and using only up to a given number K of stages of guillotine cuts, s.t. the sum over the cost factors of all used sheets is minimal.

A classical solution approach to the 2CSV has been proposed by Gilmore and Gomory [2] who employ column generation solving the pricing problem by dynamic programming (DP). Column generation as well as DP are still important components in many advanced algorithms for the 2CSV, see e.g. Cintra et al. [3]. More recently, several approaches using DP as their main framework have been proposed, however none of them considered sheets of variable size. In general, DP is not able to efficiently compute proven optimal patterns when demand constraints must be respected, and hence all these approaches are of heuristic nature. For example, Morabito and Pureza [4] compute a pattern for a single sheet by iteratively running a DP algorithm where after each iteration weights for the element types are updated. Similarly, Cui et al. [5] apply a value-correction heuristic to solve the cutting stock problem for a single sheet type and without imposing a stage limit.

2 A Value-Correction Construction Heuristic

Our approach solves the 2CSV by sequentially computing cutting patterns sheet by sheet. Initially, we assign values $v_i = h_i \, w_i \, \omega_i$, for $i = 1, \ldots, n_E$, to the element types in E, where ω_i is a weighting factor determined by the ratio of element size to average sheet size. Large elements that are difficult to place have a greater factor ω_i s.t. they are preferred and used earlier in the algorithm, since they are less likely to fit once other elements have been placed. Furthermore, these values are updated after each generated pattern in order to control the pattern generation process based on the residual demands and the total pattern value.

When a new sheet is started, greedily selecting the sheet type having e.g. the least cost-to-area ratio may not always be the best strategy. Hence, a subset $T_C \subseteq T$ with the most promising candidate sheet types is considered following the Pilot method proposed by Duin and Voß [6]. For each type $t \in T_C$ a number of cutting patterns is computed and after each generated pattern the element type values v_i are adjusted. From all generated patterns the one with the highest total value is added to the solution. Finally, the demands are updated according to the used elements. This process is iterated until all elements have been added.

Cutting patterns are generated by solving a two-dimensional knapsack problem by an approximate DP algorithm using the values v_i currently assigned to the element types. The algorithm is based on the one presented in [7] and heuristically tries to minimize the number of elements selected in the recursion that exceed the demand. Still remaining excess elements are removed in the end. In a postprocessing step a greedy randomized adaptive search procedure is used to add further elements to the free areas in the patterns to minimize the waste.

Experiments on benchmark instances from literature show that the approach yields solutions of very reasonable quality in short time. Moreover, we demonstrate its scalability on new large-scale instances from the cutting industry. Our approach constitutes a solid basis for the subsequent application of advanced metaheuristics improving the results at the cost of higher computing times.

References

1. Lodi, A., Martello, S., Vigo, D.: Recent advances on two-dimensional bin packing problems. Discrete Applied Mathematics **123**(13) (2002) 379 – 396
2. Gilmore, P.C., Gomory, R.E.: Multistage Cutting Stock Problems of Two and More Dimensions. Operations Research **13**(1) (1965) 94–120
3. Cintra, G., Miyazawa, F., Wakabayashi, Y., Xavier, E.: Algorithms for two-dimensional cutting stock and strip packing problems using dynamic programming and column generation. European Journal of Operational Research **191**(1) (2008) 61–85
4. Morabito, R., Pureza, V.: A heuristic approach based on dynamic programming and and/or-graph search for the constrained two-dimensional guillotine cutting problem. Annals of Operations Research **179**(1) (2010) 297–315
5. Cui, Y.P., Cui, Y., Tang, T.: Sequential heuristic for the two-dimensional bin-packing problem. European Journal of Operational Research **240**(1) (2015) 43–53
6. Duin, C., Voß, S.: Steiner Tree Heuristics – A Survey. In: Operations Research Proceedings 1993. Volume 1993 of Operations Research Proceedings. Springer (1994) 485–496
7. Dusberger, F., Raidl, G.R.: Solving the 3-Staged 2-Dimensional Cutting Stock Problem by Dynamic Programming and Variable Neighborhood Search. 3rd Int. Conference on Variable Neighborhood Search (to appear 2014)