Models for two-dimensional bin packing problems with customer order spread

Mateus Martin[a,*], Horacio Hideki Yanasse[b], Maristela O. Santos[c], Reinaldo Morabito[d]

[a]*Departamento de Engenharia de Produção, Universidade Federal de São Carlos, Rodovia João Leme dos Santos km 110, Sorocaba-SP, Brazil*

[b]*Instituto de Ciência e Tecnologia, Universidade Federal de São Paulo, Avenida Cesare Mansueto Giulio Lattes 1201, 12247-014, São José dos Campos-SP, Brazil*

[c]*Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, Avenida Trabalhador São carlense 400, 13566-590, São Carlos-SP, Brazil*

[d]*Departamento de Engenharia de Produção, Universidade Federal de São Carlos, Via Washington Luiz km 235, 13565-905, São Carlos-SP, Brazil*

Abstract

In this paper, we address an extension of the classical two-dimensional bin packing (2BPP) that considers the spread of customer orders (2BPP-OS). The 2BPP-OS addresses a set of rectangular items, required from different customer orders, to be cut from a set of rectangular bins. All the items of a customer order are dispatched together to the next stage of production or distribution after its completion. The objective is to minimize the number of bins used and the spread of customer orders over the cutting process. The 2BPP-OS gains relevance in manufacturing environments that seek minimum waste solutions with satisfactory levels of customer service. We propose integer linear programming (ILP) models for variants of the 2BPP-OS that consider non-guillotine, 2-stage, restricted 3-stage, and unrestricted 3-stage patterns. We are not aware of integrated approaches for the 2BPP-OS in the literature despite its relevance in practical settings. Using a general-purpose ILP solver, the results show that the 2BPP-OS takes more computational effort to solve than the 2BPP, as it has to consider several symmetries that are often disregarded by the traditional 2BPP approaches.

Keywords: Cutting & Packing, Mixed-integer linear programming, Non-guillotine pattern, 2-stage and 3-stage patterns, Order spread minimization

1. Introduction

The two-dimensional bin packing problem (2BPP) is a widely studied combinatorial optimization problem that considers a set of rectangular differently sized items to be cut out

*Corresponding author.

Email addresses: mpmartin@ufscar.br (Mateus Martin), horacio.yanasse@unifesp.br (Horacio Hideki Yanasse), mari@icmc.usp.br (Maristela O. Santos), morabito@dep.ufscar.br (Reinaldo Morabito)

of a minimum number of identical rectangular bins while minimizing the number of bins used. The problem is strongly NP-Hard, as it is an extension of the one-dimensional bin packing problem (Garey & Johnson, 1979). The 2BPP has many applications in manufacturing industries, such as in the cutting of glass panels, wooden boards, and steel sheets, and in logistical environments, such as truck loading and packaging design (Martin et al., 2022). There are several variations of the 2BPP, which are typically defined by the specific constraints related to the application field. Two of the main variations are the bin orientation, which allows the rotation of items by 90 degrees seeking to reach solutions with better material utilization rates, and the cutting profile, which is related to the cutting device and may consider non-guillotine or guillotine patterns (Lodi et al., 1999).

Solution approaches for the 2BPP are surveyed in the works of Lodi et al. (2002a), Scheithauer (2018), and Iori et al. (2021). These approaches can be categorized as: (i) exact algorithms that explore all possible solutions, such as branch and bound algorithms and models of integer linear programming (Lodi et al., 2004; Puchinger & Raidl, 2007); (ii) approximation algorithms with worst-case performance guarantees based on shelf allocation strategies such as first fit decreasing height and best fit decreasing height (Coffman et al., 1980; Lodi et al., 2002b); (iii) heuristic and metaheuristic algorithms to find satisfactory solutions to the problem in a reasonable amount of time (Lodi et al., 1999; Alvelos et al., 2009; Cui et al., 2015). There is also a strong field of studies about lower and upper bounds which are useful for the solution approaches (Boschetti & Mingozzi, 2003a,b).

In manufacturing environments, the items to be cut are related to customer orders. In this sense, Dyson & Gregory (1974) and Madsen (1979, 1988) seem to be the first to address cutting problems that deal with a sequencing decision seeking to contemplate the customer orders from a two-phase approach. They proposed to first solve the cutting problem with the column generation approach of Gilmore & Gomory (1965) and, then, to sequence the cutting patterns to reduce the number of discontinuities, which are the number of times that a customer order (e.g. all the copies of an item type) is re-initiated, from traveling salesman based approaches. They discussed that, in the glass industry, the shade of the glasses can be slightly different, which justifies that from aesthetic aspects the items of a customer order should be cut out of the minimum amount of objects. More recently, a few works addressed the 2BPP with due dates, which is a related problem that considers each item to be cut has a due date (Bennell et al., 2013; Arbib et al., 2021; Polyakovskiy & M'Hallah, 2021). This problem generally assumes that the processing of a bin lasts a constant interval of time regardless of the items to be cut out of the bin. The objective function is to minimize the number of bins used and a scheduling metric, such as the maximum lateness, which is computed in terms of bins.

In contrast to these problems, it is more appropriate to optimize the spread of customer orders during the cutting process when the items belonging to an order are dispatched together to the next stage of production or distribution after the order completion. In fact, the manufacturer generally follows an organizational policy of awaiting the consolidation of the demand of some orders before starting the cutting process to improve the use of raw materials. This policy tends to harm the delivery times and invoicing of orders if the sequence of bins cut generates items without considering they belong to different customer orders.

This decision can be addressed by the Minimization of Order Spread Problem (MORP), which is a combinatorial optimization problem that seeks to determine a processing sequence of tasks (Dyson & Gregory, 1974). Its objective function minimizes the largest spread among all orders (Madsen, 1983) or the sum of the spread of all orders (Foerster & Wäscher, 1998). These goals bring concepts of satisfactory levels of customer service. Indeed, a 2BPP solution that is optimal in terms of bin usage may be poor in terms of customer service if the orders remain in process for a long time during the cutting operation. The MORP is NP-Hard and also arises in the context of several other optimization problems, such as integrated circuit design and Graph Theory problems (Linhares & Yanasse, 2002).

In this paper, we propose mathematical models for the two-dimensional bin packing problem with customer order spread (2BPP-OS). The 2BPP-OS is an integrated problem with the decisions of the 2BPP and MORP. We highlight the importance of the 2BPP-OS in the context of low-scale production systems, given that they often compete not only on cost but also on the speed of delivery of products — see Melega et al. (2022) for a discussion about cutting and packing decisions that arise in low-scale production systems. The main contribution presented in this paper is a modeling approach based on Integer Linear Programming (ILP) to the 2BPP-OS for distinct cutting profiles. We present models for variants of the 2BPP-OS of non-guillotine, 2-stage, restricted 3-stage, and unrestricted 3-stage patterns – these patterns are explained in the next section. We are not aware of integrated approaches for the 2BPP-OS in the literature despite its evident relevance in practical settings. The proposed ILP formulations are derived from the modeling approaches of Padberg (2000), Lodi et al. (2004) and Puchinger & Raidl (2007) concerning the 2BPP. Using a general-purpose ILP solver, the results show that the 2BPP-OS takes more computational effort to solve than the 2BPP, as it has to consider several symmetries that are often disregarded by the traditional 2BPP approaches.

The paper is organized as follows. In Section 2, we describe the 2BPP-OS, the four addressed cutting profiles, and an illustrative example to highlight the notion of customer order spread. In Section 3, we present ILP formulations for the 2BPP-OS of non-guillotine, 2-stage, restricted 3-stage, and unrestricted 3-stage patterns. Using a general-purpose ILP solver and adapted benchmark instances from the literature, the computational performance of the proposed formulations is reported in Section 4. In Section 5, we conclude the study and discuss opportunities for future research.

2. Description of the problems

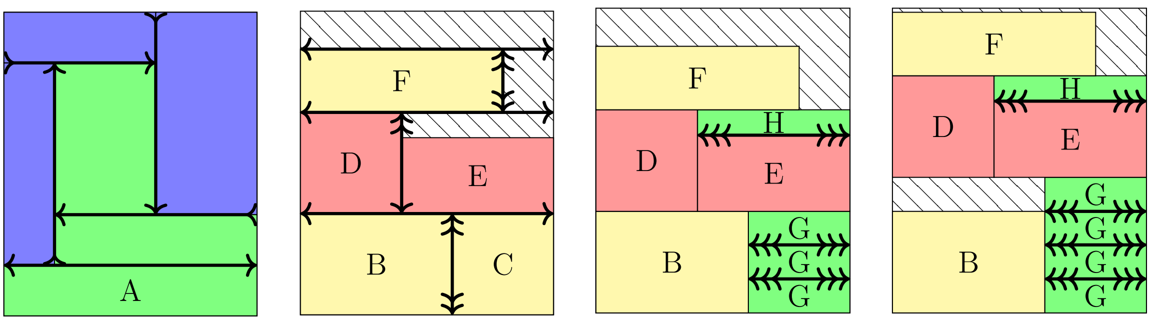
The 2BPP-OS addresses a set $I = \{1, \ldots, n\}$ of rectangular items to be cut out of a set of rectangular bins of length L and width W. Each item $i \in I$ is characterized by its length l_i, width w_i, and it is associated with a single customer order $c_i \in C = \{1, \ldots, m\}$. The problem considers identical items as distinct items. All the items of a customer order $c \in C$ are represented by set $I_c = \{i \in I \mid c_i = c\}$. Let us assume a set $S = \{1, \ldots, \overline{s}\}$ of bins to be ordinal and so its kth element represents the kth bin to be processed in the cutting process. Let us define a binary parameter π_i^s that is equal to 1 if item $i \in I$ is cut out of bin $s \in S$, and 0 otherwise. Therefore, the spread of customer order $c \in C$ is given by

$$OS^c = \max_{s \in S, i \in I_c} \{s\pi_i^s : \pi_i^s = 1\} - \min_{s \in S, i \in I_c} \{s\pi_i^s : \pi_i^s = 1\} + 1;$$

thus, the spread of the customer orders is computed in terms of bins; we assume the processing of each bin lasts a constant interval of time regardless of the items to be cut out of the bin. Therefore, in addition to the number of bins, one could minimize the largest spread among all the orders $(\max_{c \in C}\{OS^c\})$ and/or the sum of the spread of all orders $(\sum_{c \in C} OS^c)$. We state the problem in a lexicographical/hierarchical relation when the minimization of the number of bins is a primary objective (major contribution to the objective function) and the minimization of the customer order spread has a minor contribution to the objective function. Indeed, we consider the minimization of the largest spread among all orders as a secondary objective and the minimization of the sum of the spread of all orders as a tertiary objective. According to the typology of Wäscher et al. (2007) for cutting and packing problems, the 2BPP-OS can be categorized as a standard problem known as the Two-dimensional Rectangular Single-Stock Size Bin Packing Problem (2D-R-SSS-BPP), in which the spread of the customer orders is understood as an extension.

2.1. Cutting profiles

We address four variants of the 2BPP-OS that all include the following geometric constraints: the cuts are orthogonal and so the edges of the items must be parallel to the bins' edges, and any pair of items cut out of a bin must not overlap each other. As far as the technological constraint is concerned, we consider non-guillotine and guillotine patterns. In a guillotine pattern, all the cut items are obtained after a sequence of edge-to-edge cuts (the cutting of a larger rectangle produces two smaller rectangles); this limitation arises in manufacturing environments when they use cutting saws. The four cutting profiles are depicted in Fig. 1; for sake of clarity, we use arrows to highlight the relevant cuts of each pattern. These cutting profiles are described in what follows:



(a) non-guillotine. (b) 2-stage. (c) restricted 3-stage. (d) unrestricted 3-stage.

Figure 1: Examples of non-guillotine, 2-stage, unrestricted 3-stage, and unrestricted 3-stage patterns.

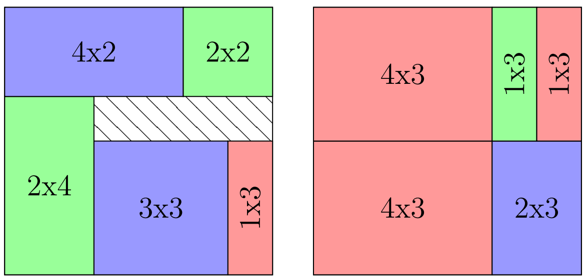
Table 1: Illustrative example of the 2BPP-OS with $L = W = 6$ and $n = 10$ items from $m = 3$ customer orders.

i	1	2	3	4	5	6	7	8	9	10
l_i	4	4	1	1	1	2	2	4	3	2
w_i	3	3	3	3	3	4	2	2	3	3
c_i	1	1	1	1	2	2	2	3	3	3

- Non-guillotine pattern: it is a more general type of cutting pattern that is not limited to edge-to-edge cuts only; the applications generally use laser beams or water jets as cutting devices with cuts in “L” and/or interrupted cuts;
- 2-stage pattern: there are two sequences of cuts in the same direction, that is, first-stage cuts generate strips out of a bin (as strips BC, DE, and F in Fig. 1b), and then second-stage cuts generate items out of the strips;
- Restricted 3-stage patterns: there are three sequences of cuts in the same direction, that is, first-stage cuts generate strips out of a bin (as strips BGGG, DEH, and F in Fig. 1c), second-stage cuts generate items or stacks out of the strips (as item D and stack EH out of strip DEH in Fig. 1c), and lastly, third-stage cuts generate items out of the stacks (as items E and H out of stack EH in Fig. 1c); this pattern limits the width of the strips to be equal to the width of an item (as strip BGGG that has the width of item B in Fig. 1c);
- Unrestricted 3-stage patterns: it is similar to the previous one; however, this pattern allows the strips to have a width that is a combination of the items' width. (as strip BGGGG that has the width of stack GGGG in Fig. 1d).

2.2. Illustrative example

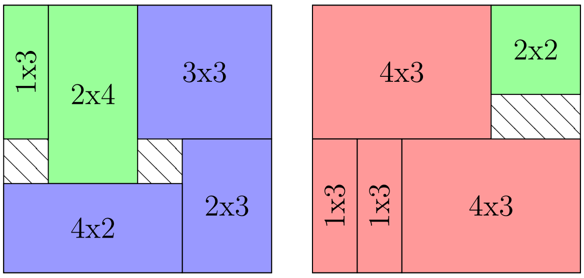
In Table 1, we present an illustrative example of the 2BPP-OS with bins of size $L = W = 6$ and $n = 10$ items from $m = 3$ customer orders (i.e., $C = \{1, 2, 3\}$). For this example,



(a) Bin $s = 1$. (b) Bin $s = 2$.

Figure 2: A solution for the illustrative example with two bins, largest spread of 2 units, and sum of the spread of 6 units.

the material bound is equal to $\lceil \sum_{i \in I} l_i w_i / (LW) \rceil = \lceil 1.889 \rceil$, that is, 2 bins is a lower bound on the optimal number of bins used (Scheithauer, 2018). In Fig. 2, we present a solution with two bins, largest spread of 2 units, and sum of the spread of 6 units. The items of customer order $c = 1$ are depicted in red, $c = 2$ in green, and $c = 3$ in blue; the hatched area is waste. Note that the processing of the three orders starts in the first bin and ends in the second bin; thus, each of them has a spread of 2 $(= 2 - 1 + 1)$ units. In contrast, in Fig. 3, we present another solution for the illustrative example with two bins, largest spread of 2 units, and sum of the spread of 4 units. For this solution, customer order $c = 1$ has an order spread of 1 $(= 2 - 2 + 1)$ unit; customer order $c = 2$ has an order spread of 2 $(= 2 - 1 + 1)$ units; and, customer order $c = 3$ has an order spread of 1 $(= 1 - 1 + 1)$ units.



(a) Bin $s = 1$. (b) Bin $s = 2$.

Figure 3: A solution for the illustrative example with two bins, largest spread of 2 units, and sum of the spread of 4 units.

The minimization of the largest spread among all the orders tends to contribute to the service level of the operation (scenario of the worst case), and the minimization of the sum of the spread of all orders tends to contribute to the flow of the operation (scenario of the medium case). Notice that issues such as time-consuming cutting, difficulties in managing the operation with greater volumes of work-in-process, and the difficulty of identifying and handling items tend to be reduced when the spread of customer orders is minimized. As the solutions in Figs. 2 and 3 present the same bin usage, the solution of Fig. 3 is preferable to the solution of Fig. 2 as the latter have a better metric of customer order spread.

3. Mathematical models

In this section, we propose ILP formulations for the 2BPP-OS of non-guillotine, 2-stage, restricted 3-stage, and unrestricted 3-stage patterns in Sections 3.1, 3.2, 3.3, and 3.4, respectively. For each formulation, we point out that the notation referring to its variables is within the scope of its section only, unless otherwise indicated. For instance, all these four formulations have the same approach for modeling the spread of customer orders which is explained in Section 3.1. For sake of clarity, we next recall the notation of the 2BPP-OS with the addition of two new definitions:

$L,\ W$	length and width of the bins, respectively;
$I = \{1, \dots, n\}$	set of items;
$l_i,\ w_i$	length and width of item $i \in I$, respectively;
$C = \{1, \dots, m\}$	set of customer orders;
$c_i \in C$	customer order of item $i \in I$;
$I_c = \{i \in I \mid c_i = c\}$	set of all items belonging to customer order $c \in C$;
$\underline{s},\ \overline{s}$	lower and upper bounds on the number of bins, respectively;
$S = \{1, \dots, \overline{s}\}$	set of bins;
O^c	lower bound on the number of bins required to fulfill customer order $c \in C$ only.

We assume, without loss of generality, the input data have positive integers. In particular, the data concerning the items (l_i, w_i, c_i) are sorted by non-increasing width, such that, $w_1 \geq w_2 \geq \ldots \geq w_n$. Notice that parameter $\underline{s}$ and parameters $\overline{s}$ and O^c can be computed from relaxations and heuristics for the 2BPP-OS, respectively. As mentioned before, the modeling approach is rooted in the works of Padberg (2000), Lodi et al. (2004), and Puchinger & Raidl (2007) concerning the 2BPP. The main differences concerning these works are discussed at the end of the corresponding sections.

3.1. Non-guillotine patterns

The non-guillotine patterns have to fulfill the geometric constraints only. There are nine families of decision variables in the formulation, of which six concern the cutting problem and the rest the spread of the customer orders. They are defined in what follows:

y^s	binary variable which equals 1, if bin $s \in S$ is cut, and 0 otherwise;
x_i^s	binary variable which equals 1, if item $i \in I$ is packed at bin $s \in S$, and 0 otherwise;
(α_i, β_i)	variables that represent the allocation point (lower-left corner) of item $i \in I$;
u_{ij}	binary variable which equals 1, if item $i \in I$ is packed at the left of item $j \in I$, and 0 otherwise;
v_{ij}	binary variable which equals 1, if item $i \in I$ is packed below item $j \in I$, and 0 otherwise;
b^c	start of the processing of customer order $c \in C$ in terms of bins;
e^c	end of the processing of customer order $c \in C$ in terms of bins;
s^c	spread of customer order $c \in C$ in terms of bins.

For the formulation, the non-overlapping of any pair of items $i, j \in I, i \neq j$, cut out of the same bin is guaranteed with two sets of binary variables u_{ij} and v_{ij}, and a set of constraints that ensure one out of four possible relative positions is at least fulfilled: item i is to the left of item j, item i is to the right of item j, item i is below item j, or item i is above item j (Padberg, 2000). An ILP formulation for the 2BPP-OS of non-guillotine patterns is given by model (1).

$$\textbf{Mininimize} \quad r_1 \sum_{s \in S} y^s + r_2 \max_{c \in C}\{s^c\} + r_3 \sum_{c \in C} s^c, \tag{1a}$$

s.t.

$$\sum_{s \in S} x_i^s = 1, \qquad i \in I, \tag{1b}$$

$$\sum_{i \in I} l_i w_i x_i^s \le LWy^s, \qquad s \in S, \tag{1c}$$

$$\alpha_i + l_i \le \alpha_j + L(1 - u_{ij}), \qquad i, j \in I, i \neq j, \tag{1d}$$

$$\beta_i + w_i \le \beta_j + W(1 - v_{ij}), \qquad i, j \in I, i \neq j, \tag{1e}$$

$$u_{ij} + u_{ji} + v_{ij} + v_{ji} \ge x_i^s + x_j^s - 1, \qquad s \in S, i, j \in I, i > j, \tag{1f}$$

$$b^c \le \sum_{s \in S} s x_i^s, \qquad c \in C, i \in I_c, \tag{1g}$$

$$e^c \ge \sum_{s \in S} s x_i^s, \qquad c \in C, i \in I_c, \tag{1h}$$

$$s^c = e^c - b^c + 1, \qquad c \in C, \tag{1i}$$

$$y^s = 1, \qquad s = 1, \dots, \underline{s}, \tag{1j}$$

$$y^s \ge y^{s+1}, \qquad s = \underline{s}, \dots, \overline{s} - 1, \tag{1k}$$

$$y^s \in \{0, 1\}, \qquad s \in S, \tag{1l}$$

$$x_i^s \in \{0, 1\}, \qquad s \in S, i \in I, \tag{1m}$$

$$0 \le \alpha_i \le L - l_i, \; 0 \le \beta_i \le W - w_i, \qquad i \in I, \tag{1n}$$

$$u_{ij}, v_{ij} \in \{0, 1\}, \qquad i, j \in I, i \neq j, \tag{1o}$$

$$1 \le b^c \le \overline{s} - O^c + 1, \qquad c \in C, \tag{1p}$$

$$O^c \le e^c \le \overline{s}, \qquad c \in C, \tag{1q}$$

$$O^c \le s^c \le \overline{s}, \qquad c \in C. \tag{1r}$$

The objective function (1a) minimizes the number of bins used, the largest spread among all orders, and the sum of the spread of all orders according to parameters r_1, r_2, and r_3 that work as weights to these three objectives. In the computational experiments of Section 4, we used $r_1 = m\overline{s}^2$, $r_2 = m\overline{s}$, and $r_3 = 1$ to establish a primary objective to the number of used bins, a secondary objective to the largest spread, and a tertiary objective to the sum of the spread. The term $\max_{c \in C}\{s^c\}$ can be written in a linear form if replaced with an auxiliary variable K and the addition of constraints $K \ge s^c$, $c \in C$.

Constraints (1b) ensure each item $i \in I$ is packed at a single bin. Constraints (1c) ensure that no item is packed at a bin ($x_i^s = 0$, $i \in I$) when the bin is not cut ($y^s = 0$). In addition, when $y^s = 1$, they ensure the sum of the items' area packed at bin $s \in S$ does not exceed the bin's area $L\,W$. Constraints (1d) and (1e) enforce the definition of variables u_{ij} and v_{ij} for any pair of items $i, j \in I$, $i \neq j$. Constraints (1f) ensure the fulfillment of at least one of the four possible relative positions when two items $i, j \in I$, $i \neq j$ are packed at the same bin $s \in S$.

Constraints (1g), (1h), and (1i) are responsible for modeling the spread of the customer orders. From the cutting variables x_i^s, constraints (1g) and (1h) enforce the definition of variables b^c and e^c, respectively, by considering each item $i \in I_c$ of customer order $c \in C$. They work as linking constraints between the cutting and order spread decisions of the 2BPP-OS. Constraints (1i) computes the spread of a customer order $c \in C$ in terms of bins, as defined in Section (2).

Expressions (1j) and (1k) work as fixing variables and valid inequalities, respectively. Constraints (1l) to (1r) define the domain of the variables. We note that considering parameter O^c in our computational experiments, that is, a lower bound on the number of bins required to fulfill the customer order $c \in C$ in the definition of variables b^c, e^c, and s^c was useful to provide better LP-relaxations. We also consider two additional expressions to eliminate those possibilities when two items $i, j \in I$ do not fit in a single bin in a horizontal and/or vertical direction, as given by expressions (2a) and 2b, respectively.

$$u_{ij} = u_{ji} = 0, \qquad i, j \in I, i \neq j, l_i + l_j > L, \quad (2a)$$
$$v_{ij} = v_{ji} = 0, \qquad i, j \in I, i \neq j, w_i + w_j > W. \quad (2b)$$

We note that Beasley (1985b) presented an alternative ILP modeling approach for non-guillotine patterns. In a preliminary phase of this study, we considered that approach to the 2BPP-OS of non-guillotine patterns, but discarded it after experiencing poor computational performance, mainly due to its pseudo-polynomial/great number of variables. In contrast, model (1) is based on the approach of Padberg (2000), which has a polynomial number of variables and constraints. The reader is referred to Scheithauer (2018) for a discussion about modeling strategies to reduce these numbers of variables in a Padberg-based formulation.

3.2. 2-stage patterns

The models of 2-stage patterns often assume that first-stage horizontal cuts generate strips out of the bins, and then second-stage vertical cuts generate items out of the strips. For this problem, without loss of optimality, the width of a strip is always equal to an item's width, which is known as its initializing item. In this sense, the other items packed in such a strip always have a smaller width. That is why we sorted the input data of the items by non-increasing width. Thus, we can characterize a strip by the index of its initializing item, as defined in set I, that is, strip $j \in I$ and the possible items to be packed at such a strip as items $i \in I$, where $j \leq i$. Moreover, we do not have to consider constraints for modeling the non-overlapping of pairs of items because the sequence of guillotine cuts already fulfills such a requirement.

There are seven families of decision variables in the formulation, of which four concern the cutting problem and the rest the spread of the customer orders. In addition to variables y^s, b^c, e^c, and s^c, which were defined in the previous section, the formulation has the following variables:

α_{ji} binary variable which equals 1, if item $i \in I$ is packed at strip $j \in I$, $j \leq i$, and 0 otherwise;

β_j^s binary variable which equals 1, if strip $j \in I$ is packed at bin $s \in S$, and 0 otherwise;

λ_{ji}^s binary variable which equals 1, if and only if $\beta_j^s = 1$ and $\alpha_{ji} = 1$, $s \in S, j, i \in I, j \leq i$, and 0 otherwise;

An ILP formulation for the 2BPP-OS of 2-stage patterns is given by model (3).

Mininimize (1a),

s.t.

$$(1i), (1j), (1k), (1p), (1q), (1r),$$

$$\sum_{j=1}^{i} \alpha_{ji} = 1, \qquad i \in I, \quad (3a)$$

$$\sum_{s \in S} \beta_j^s = \alpha_{jj}, \qquad j \in I, \quad (3b)$$

$$\sum_{i=j+1}^{n} l_i \alpha_{ji} \leq (L - l_j)\alpha_{jj}, \qquad j \in I, \quad (3c)$$

$$\sum_{j \in I} w_j \beta_j^s \leq W y^s, \qquad s \in S, \quad (3d)$$

$$\beta_j^s + \alpha_{ji} \leq 1 + \lambda_{ji}^s, \qquad s \in S, j, i \in I, j \leq i, \quad (3e)$$

$$\lambda_{ji}^s \leq (\beta_j^s + \alpha_{ji})/2, \qquad s \in S, j, i \in I, j \leq i, \quad (3f)$$

$$b^c \leq \sum_{s \in S} \sum_{j=1}^{i} s \lambda_{ji}^s, \qquad c \in C, i \in I_c, \quad (3g)$$

$$e^c \geq \sum_{s \in S} \sum_{j=1}^{i} s \lambda_{ji}^s, \qquad c \in C, i \in I_c, \quad (3h)$$

$$\alpha_{ji} \in \{0, 1\}, \qquad j, i \in I, j \leq i, \quad (3i)$$

$$\beta_j^s \in \{0, 1\}, \qquad s \in S, j \in I, \quad (3j)$$

$$\lambda_{ji}^s \in \{0, 1\}, \qquad s \in S, j, i \in I, j \leq i. \quad (3k)$$

Constraints (3a) ensure each item $i \in I$ is packed at a single strip $j \in I$, $j \leq i$. Constraints (3b) ensure each strip $j \in I$, if any, is packed at a single bin $s \in S$. Constraints (3c) guarantee the sum of the items' length packed at a strip $j \in I$ does not exceed the strips' length L. Constraints (3d) guarantee the sum of the strips' width packed at a bin $s \in S$ does not exceed the bin' width W. The previous two constraints also enforce: $\alpha_{ji} = 0$ if $\alpha_{jj} = 0$, $j, i \in I, j \leq i$; and, $\beta_j^s = 0$ if $y^s = 0$, $s \in S, j \in I$. Constraints (3e) and (3f) are responsible for generating the result $\lambda_{ji}^s = \beta_j^s \alpha_{ji}$, $s \in S$, $j, i \in I$, $j \leq i$. From the cutting variables λ_{ji}^s, the linking constraints (3g) and (3h) generate the definition of variables b^c and e^c by considering all the items $i \in I_c$ of the customer order $c \in C$. Constraints (3i) to (3k) define the domain of the variables. The other variables and constraints are as previously defined.

Similar to the previous section, we consider expressions to eliminate that possibility when two items $i, j \in I, j < i$ do not fit in a single strip in a horizontal direction, as given by expressions (4a) and 4b.

$$\alpha_{ji} = 0, \qquad j, i \in I, j < i, l_i + l_j > L, \quad \text{(4a)}$$
$$\lambda^s_{ji} = 0, \qquad s \in S, j, i \in I, j < i, l_i + l_j > L. \quad \text{(4b)}$$

The model of Lodi et al. (2004) for 2-stage patterns assumes, without loss of optimality, $\beta^s_j = 0$ if $s > j$, $s \in S, j \in I$. Indeed, this assumption is able to eliminate symmetries of the 2BPP by limiting the first item to be packed up to the first bin, the second item up to the second bin, and so on. Nevertheless, we do not consider this assumption, as it represents a virtual sequencing constraint that may lead to a loss of optimality when minimizing the spread of customer orders. For instance, one could solve a problem instance where the optimal solution of the 2BPP-OS has to pack the first item in the last bin due to the spread of a customer order. Moreover, in comparison to that model, we had to create the cutting variables λ^s_{ji} to associate the items packed at the same bin, as this information is required by the linking constraints (3g) and (3h).

3.3. Restricted 3-stage patterns

Puchinger & Raidl (2007) extended to 3-stage patterns the modeling approach of Lodi et al. (2004). The model assumes that first-stage horizontal cuts generate strips out of the bins; second-stage vertical cuts generate stacks out of the strips; and, then, third-stage horizontal cuts generate items out of the stacks. Similar to the previous section, the width of a strip is always equal to its initializing item's width. That is why they called this pattern restricted in opposition to the unrestricted 3-stage pattern that allows the strips to have more general widths. In addition, we now have stacks of items, and the first item in a stack is called its initializing item. All the items packed at the same stack must have the length of the stack's initializing item as the pattern is limited to three guillotine stages. Thus, we can characterize a stack by the index of its initializing item, as defined in set I, that is, stack $j \in I$ and the possible items to be packed at such a stack as items $i \in I$, where $j \leq i, l_j = l_i$. Notice that a stack is allowed to contain only its initializing item (stack of a single item).

There are eight families of decision variables in the formulation, of which five concern the cutting problem and the rest the spread of the customer orders. In addition to variables y^s, b^c, e^c, and s^c, which were defined in Section 3.1, the formulation has the following variables:

α_{ji} binary variable which equals 1, if item $i \in I$ is packed at stack $j \in I$, $j \leq i$, and 0 otherwise;

β_{kj} binary variable which equals 1, if stack $j \in I$ is packed at strip $k \in I$, $k \leq j$, 0 otherwise;

γ_k^s binary variable which equals 1, if strip $k \in I$ is packed at bin $s \in S$, and 0 otherwise;

λ_{kji}^s binary variable which equals 1, if and only if $\gamma_k^s = 1$, $\beta_{kj} = 1$, and $\alpha_{ji} = 1$, $s \in S, k \in I$, $j \in I, I \in I, k \leq j \leq i$, and 0 otherwise.

An ILP formulation for the 2BPP-OS of restricted 3-stage patterns is given by model (5).

Mininimize (1a),

s.t.

(1i), (1j), (1k), (1p), (1q), (1r),

$$\sum_{j=1}^{i} \alpha_{ji} = 1, \qquad i \in I, \tag{5a}$$

$$\sum_{k=1}^{j} \beta_{kj} = \alpha_{jj}, \qquad j \in I, \tag{5b}$$

$$\sum_{s \in S} \gamma_k^s = \beta_{kk}, \qquad k \in I, \tag{5c}$$

$$\sum_{i=j}^{n} w_i \alpha_{ji} \leq \sum_{k=1}^{j} w_k \beta_{kj}, \qquad j \in I, \tag{5d}$$

$$\sum_{j=k+1}^{n} l_j \beta_{kj} \leq (L - l_k)\beta_{kk}, \qquad k \in I, \tag{5e}$$

$$\sum_{k \in K} w_k \gamma_k^s \leq W y^s, \qquad s \in S, \tag{5f}$$

$$\gamma_k^s + \beta_{kj} + \alpha_{ji} \leq 2 + \lambda_{kji}^s, \qquad s \in S, k, j, i \in I, k \leq j \leq i, \tag{5g}$$

$$\lambda_{kji}^s \leq (\gamma_k^s + \beta_{kj} + \alpha_{ji})/3, \qquad s \in S, k, j, i \in I, k \leq j \leq i, \tag{5h}$$

$$b^c \leq \sum_{s \in S} \sum_{k \in I} \sum_{j=k}^{i} s \lambda_{kji}^s, \qquad c \in C, i \in I_c, \tag{5i}$$

$$e^c \geq \sum_{s \in S} \sum_{k \in I} \sum_{j=k}^{i} s \lambda_{kji}^s, \qquad c \in C, i \in I_c, \tag{5j}$$

$$\alpha_{ji} \in \{0, 1\}, \qquad j \in I, i \in I, j \leq i, \tag{5k}$$

$$\beta_{kj} \in \{0, 1\}, \qquad k \in I, j \in I, k \leq j, \tag{5l}$$

$$\gamma_k^s \in \{0, 1\}, \qquad s \in S, k \in I, \tag{5m}$$

$$\lambda_{kji}^s \in \{0, 1\}, \qquad s \in S, k, j, i \in I, k \leq j \leq i, \tag{5n}$$

Constraints (5a) ensure each item $i \in I$ is packed at a single stack $j \in I,\ j \leq i$. Constraints (5b) ensure each stack $j \in I$, if any, is packed at a single strip $k \in I,\ k \leq j$. Constraints (5c) ensure each strip $k \in I$, if any, is packed at a single bin $s \in S$. Constraints (5d) guarantee the sum of the items' width packed at stack $j \in I$ does not exceed the width w_k of its corresponding strip $k \in I,\ k \leq j$. Constraints (5e) guarantee the sum of the stacks' length packed at a strip $k \in I$ does not exceed the strips' length L. Constraints (5f) guarantee the sum of the strips' width packed at a bin $s \in S$ does not exceed the bin' width W. The previous two constraints also enforce: $\beta_{kj} = 0$ if $\beta_{kk} = 0,\ k, j \in I, k \leq j$; and, $\gamma_k^s = 0$ if $y^s = 0,\ s \in S, k \in I$. Constraints (5g) and (5h) are responsible for generating the result $\lambda_{kji}^s = \gamma_k^s \beta_{kj} \alpha_{ji},\ s \in S,\ k, j, i \in I,\ k \leq j \leq i$. From the cutting variables λ_{kji}^s, the linking constraints (5i) and (5j) generate the definition of variables b^c and e^c by considering all the items $i \in I_c$ of the customer order $c \in C$. Constraints (5k) to (5n) define the domain of the variables. The other variables and constraints are as previously defined.

Similar to the previous sections, we consider expressions to eliminate those possibilities when: two items $i, j \in I, j < i$ do not fit in a single stack in a horizontal or vertical direction, as given by expressions (6a); and, two items $k, j \in I, k < j$ do not fit in a single strip in a horizontal direction, as given by expressions (6b). Expressions (6c) present the counterpart of these ideas for variables λ_{kji}^s.

$$\alpha_{ji} = 0, \qquad j, i \in I, j < i, l_i \neq l_j \vee w_i + w_j > W, \quad \text{(6a)}$$
$$\beta_{kj} = 0, \qquad k, j \in I, k < j, l_k + l_j > L, \quad \text{(6b)}$$
$$\lambda_{kji}^s = 0, \qquad s \in S, k, j, i \in I, j < i, l_j \neq l_i \vee w_j + w_i > W. \quad \text{(6c)}$$

The model of Puchinger & Raidl (2007) for restricted 3-stage patterns assumes, without loss of optimality, $\gamma_k^s = 0$ if $s > k,\ s \in S, k \in I$. As in the previous section, these virtual sequencing constraints eliminate symmetries of the 2BPP; however, we do not consider them to avoid loss of optimality since we also minimize the spread of customer orders in the 2BPP-OS. Moreover, in comparison to that model, we had to create the cutting variables λ_{kji}^s to associate the items packed at the same bin, as this information is required by the linking constraints (5i) and (5j).

3.4. Unrestricted 3-stage patterns

The model of Puchinger & Raidl (2007) for unrestricted 3-stage patterns assumes the same definitions and sequence of cuts of the previous section. However, the unrestricted 3-stage patterns consider that the width of each stack $k \in I$ is equal to its initializing stack's width, which is defined by index $k \in I$. (In the previous section, the width of each stack $k \in I$ is the item's width w_k.) This new definition leads to another difference: a stack $j \in I$ can be packed at any strip $k \in I$, that is, not requiring the condition $k \leq j$ anymore as in variables β_{kj} of model (5). Therefore, the initializing stack $k \in I$ of a strip k may not be the first stack of such a strip.

There are nine families of decision variables in the formulation, of which six concern the cutting problem and the rest the spread of the customer orders. In addition to variables y^s, b^c, e^c, and s^c, which were defined in Section 3.1, the formulation has the following variables:

α_{ji} binary variable which equals 1, if item $i \in I$ is packed at stack $j \in I$, $j \leq i$, and 0 otherwise;

β_{kj} binary variable which equals 1, if stack $j \in I$ is packed at strip $k \in I$, and 0 otherwise;

γ_k^s binary variable which equals 1, if strip $k \in I$ is packed at bin $s \in S$, and 0 otherwise;

δ_{ji}^s binary variable which equals 1, if and only if $\gamma_j^s = 1$ and $\alpha_{ji} = 1$, $s \in S, j, i \in I, j < i$, and 0 otherwise. Thus, the variable assumes the value of 1 for all the items i packed a stack j (but its initializing item), which is then packed at a bin s;

λ_{kji}^s binary variable which equals 1, if and only if $\gamma_k^s = 1$, $\beta_{kj} = 1$, and $\alpha_{ji} = 1$, $s \in S, k \in I$, $j \in I, I \in I, k \leq j \leq i$, and 0 otherwise.

An ILP formulation for the 2BPP-OS of unrestricted 3-stage patterns is given by model (7).

Mininimize (1a),

s.t.

(1i), (1j), (1k), (1p), (1q), (1r),

$$\sum_{j=1}^{i} \alpha_{ji} = 1, \qquad i \in I, \tag{7a}$$

$$\sum_{i=j+1}^{n} \alpha_{ji} \leq (n-j)\alpha_{jj}, \qquad j \in I \backslash \{n\}, \tag{7b}$$

$$\sum_{k \in I} \beta_{kj} = \alpha_{jj}, \qquad j \in I, \tag{7c}$$

$$\sum_{s \in S} \gamma_k^s = \beta_{kk}, \qquad k \in I, \tag{7d}$$

$$\sum_{k \in I} \gamma_k^s \leq n y^s, \qquad s \in S, \tag{7e}$$

$$\sum_{i=j}^{n} w_i \alpha_{ji} \leq \sum_{i=k}^{n} w_i \alpha_{ki} + W(1 - \beta_{kj}), \qquad k, j \in I, k \neq j, \tag{7f}$$

$$\sum_{j=k+1}^{n} l_j \beta_{kj} \leq (L - l_k)\beta_{kk}, \qquad k \in I, \tag{7g}$$

$$\sum_{j \in I} w_j \gamma_j^s + \sum_{j \in I} \sum_{i=j+1}^{n} w_i \delta_{ji}^s \leq W y^s, \qquad s \in S, \tag{7h}$$

$$\gamma_j^s + \alpha_{ji} \leq 1 + \delta_{ji}^s, \qquad s \in S, j, i \in I, j < i, \tag{7i}$$

$$\delta_{ji}^s \leq (\gamma_j^s + \alpha_{ji})/2, \qquad s \in S, j, i \in I, j < i, \tag{7j}$$

$$\gamma_k^s + \beta_{kj} + \alpha_{ji} \leq 2 + \lambda_{kji}^s, \qquad s \in S, k, j, i \in I, j \leq i, \tag{7k}$$

$$\lambda_{kji}^s \leq (\gamma_k^s + \beta_{kj} + \alpha_{ji})/3, \qquad s \in S, k, j, i \in I, j \leq i, \tag{7l}$$

$$b^c \leq \sum_{s \in S} \sum_{k \in I} \sum_{j=1}^{i} s\lambda_{kji}^s, \qquad c \in C, i \in I_c, \quad (7m)$$

$$e^c \geq \sum_{s \in S} \sum_{k \in I} \sum_{j=1}^{i} s\lambda_{kji}^s, \qquad c \in C, i \in I_c, \quad (7n)$$

$$\alpha_{ji} \in \{0,1\}, \qquad j,i \in I, j \leq i, \quad (7o)$$

$$\beta_{kj} \in \{0,1\}, \qquad k,j \in I, k \leq j, \quad (7p)$$

$$\gamma_k^s \in \{0,1\}, \qquad s \in S, k \in I, \quad (7q)$$

$$\delta_{ji}^s \in \{0,1\}, \qquad s \in S, j,i \in I, j < i, \quad (7r)$$

$$\lambda_{kji}^s \in \{0,1\}, \qquad s \in S, k,j,i \in I, j \leq i. \quad (7s)$$

Constraints (7a) ensure each item $i \in I$ is packed at a single stack $j \in I$, $j \leq i$. Constraints (7b) enforce $\alpha_{ji} = 0$ if $\alpha_{jj} = 0$, $j,i \in I, j \leq i$. Constraints (7c) ensure each stack $j \in I$, if any, is packed at any strip $k \in I$. Constraints (7d) ensure each strip $k \in I$, if any, is packed at a single bin $s \in S$. Constraints (7e) enforce $\gamma_k^s = 0$ if $y^s = 0$, $s \in S, k \in I$.

Expressions (7f) are disjunctive constraints that guarantee the width of each stack $j \in I$ packed at strip $k \in I$ ($\beta_{kj} = 1$) does not exceed the width of the strip, which is given by $\sum_{i=k}^{n} w_i \alpha_{ki}$. Constraints (7g) guarantee the sum of the stacks' length packed at a strip $k \in I$ does not exceed the strips' length L. Constraints (7h) guarantee the sum of the strips' width packed at a bin $s \in S$ does not exceed the bin' width W. For a bin $s \in S$, the term $\sum_{j \in I} w_j \gamma_j^s$ gives the width of the initializing item of each stack packed at this bin, and the term $\sum_{j \in I} \sum_{i=j+1}^{n} w_i \delta_{ji}^s$ gives the width of the remaining items of these corresponding stacks. The previous two constraints also enforce: $\beta_{kj} = 0$ if $\beta_{kk} = 0$, $k,j \in I, k \leq j$; and, $\gamma_j^s = 0$ and $\delta_{ji}^s = 0$ if $y^s = 0$, $s \in S, j,i \in I, j < i$. Constraints (7i) and (7j) are responsible for generating the result $\delta_{ji}^s = \gamma_j^s \alpha_{ji}$, $s \in S$, $j,i \in I, j < i$. Constraints (7k) and (7l) are responsible for generating the result $\lambda_{kji}^s = \gamma_k^s \beta_{kj} \alpha_{ji}$, $s \in S$, $k,j,i \in I$, $j \leq i$. From the cutting variables λ_{kji}^s, the linking constraints (7m) and (7n) generate the definition of variables b^c and e^c by considering all the items $i \in I_c$ of the customer order $c \in C$. Constraints (7o) to (7s) define the domain of the variables. The other variables and constraints are as previously defined.

Similar to the previous sections, we consider expressions to eliminate those possibilities when: two items $i,j \in I, j < i$ do not fit in a single stack in a horizontal or vertical direction, as given by expressions (8a); and, two items $k,j \in I, k < j$ do not fit in a single strip in a horizontal direction, as given by expressions (8c). Expressions (8b), (8d), and (8e) are the counterpart of these ideas for variables δ_{ji}^s and λ_{kji}^s.

$$\alpha_{ji} = 0, \qquad j,i \in I, j < i, l_i \neq l_j \vee w_i + w_j > W, \quad (8a)$$

$$\delta_{ji}^s = 0, \qquad j,i \in I, j < i, l_i \neq l_j \vee w_i + w_j > W, \quad (8b)$$

$$\beta_{kj} = 0, \qquad k,j \in I, k \neq j, l_k + l_j > L, \quad (8c)$$

$$\lambda^s_{kji} = 0, \qquad s \in S, k, j, i \in I, j < i, l_j \neq l_i \vee w_j + w_i > W, \quad (8d)$$
$$\lambda^s_{kji} = 0, \qquad s \in S, k, j, i \in I, k \neq j, l_k + l_j > L. \quad (8e)$$

The model of Puchinger & Raidl (2007) for unrestricted 3-stage patterns assumes, without loss of optimality, $\gamma^s_k = 0$ if $s > k$, $s \in S, k \in I$. Again, we do not consider them to avoid loss of optimality since we also minimize the spread of customer orders. In comparison to that model, we had to create the cutting variables λ^s_{kji} to associate the items packed at the same bin, as this information is required by the linking constraints (7m) and (7n).

4. Computational experiments

We ran computational experiments to evaluate the computational performance of the proposed formulations. In what follows, we refer to model (1) for the 2BPP-OS of non-guillotine patterns as Model-NG. Likewise, the 2BPP-OS of 2-stage, restricted 3-stage, and unrestricted 3-stage patterns are referred to as Model-2S, Model-R3, and Model-U3 respectively. Since we are not aware of other integrated approaches for the 2BPP-OS, we compare our models with each other. The four models were coded in C++ using GUROBI v.10.0.0 as the general-purpose ILP solver. All the experiments were carried out on a PC with Intel Xeon E5-2680v2 (2.8 GHz), using 10 threads, 16 GB RAM, under a CentOS Linux 7.2.1511 Operating System. Each run of the solver was limited to 3,600 seconds. We next use letters "tl" in the tables to indicate when this time limit was reached for an instance or group of instances.

This section is divided into two parts. We comment on the benchmark instances used in the experiments at the beginning of these sections; we generated instances for the 2BPP-OS by adapting instances from the literature concerning the 2BPP. These adapted instances are available upon request to the authors. As a preprocessing phase prior to the models, we consider the two widely-known techniques of reducing the bins' size and enlarging the items' size, as discussed in Scheithauer (2018). In the experiments, from solutions with similar levels of bin usage, the goal is to analyze the models' performance concerning the quality of the solutions about the spread of customer orders in comparison with solutions from the approaches when these decisions are neglected during the search. In this sense, we report results from different sets of experiments. In all these experiments, the number of bins used is minimized ($\Phi_1 =$✓), except when $\underline{s} = \overline{s}$ as the optimal number of bins used is already known. In addition, we have sets of experiments with and without minimizing the largest spread among all customer orders ($\Phi_2 =$✓,✗) and the sum of the spread of all customer orders ($\Phi_3 =$✓,✗). Therefore, we are able to compare solutions with the same levels of bin usage when the decisions of the spread of customer orders are and are not considered.

Recall that parameters $\underline{s}$, $\overline{s}$, and O^c are required as input for the proposed formulations. For Model-2S (resp. Model-R3 and Model-U3), we presented the model of Lodi et al. (2004) (resp. restricted or unrestricted model of Puchinger & Raidl (2007)) to the solver to obtain valid bounds for parameters $\underline{s}$ and $\overline{s}$, considering up to 60 seconds for each run of the solver. Notice that $\underline{s} = \overline{s}$ when the optimality was proven during the 60 seconds; otherwise, after the end of the search, we rounded the dual bound's value up to obtain a value for the parameter $\underline{s}$, and the incumbent solution's value was used as the value of parameter $\overline{s}$. As far as the Model-NG is concerned, we considered solving a one-dimensional bin packing problem to provide a value for parameter $\underline{s}$ and the model of Lodi et al. (2004) to provide a value for parameter $\overline{s}$; we chose the model of Lodi et al. (2004) instead of the model of Padberg (2000), as the former provided better solutions within the 60 seconds. The solution obtained was provided to the solver as an initial solution for the integrated models. We highlight the integrated problems remain NP-Hard even when an optimal solution in terms of bins usage is provided to the solver (i.e., with $\underline{s} = \overline{s}$), as the MORP is NP-Hard and the cutting patterns of the initial solution are not fixed in the integrated models. Similarly, for each $c \in C$, we obtained the value of parameter O^c by solving these previous models considering only the items in set I_c.

4.1. Results for the set of instances #A

We generated instances for the 2BPP-OS by adapting the twelve gcut1-12 instances proposed in Beasley (1985a). The size $L \times W$ of the bins is 250×250 for gcut1-4 instances, 500×500 for gcut5-8 instances, and 1000×1000 for gcut9-12 instances. The number of items n is 10, 20, 30, or 50 (we considered the demand of one unit per item). The length l_i and width w_i of item $i \in I$ were sampled in the intervals $[L/4, 3L/4]$ and $[W/4, 3W/4]$, respectively. We arbitrarily aggregated the items to generate customer orders. For each instance, we only established the number of customers and a minimum number of items per customer. In this sense, the adapted instances: with $n = 10, 20$ items have $m = 3$ customer orders (minimum of 2 items per customer); with $n = 30$ items have $n = 3, 5$ customer orders (minimum of 3 items per customer); and, with $n = 50$ items have $m = 5, 7$ customer orders (minimum of 4 items per customers). Thus, the set of instances #A has a total of 18 instances. We refer to each instance as "name-#n-#m"; for example, instance gcut12-50-07 was generated from instance gcut12, and it has $n = 50$ items and $m = 7$ customers.

We report the results for Model-NG considering the set of instances #A in Table 2. We report the value of the number of customer orders m, number of items n, instance name, number of bins used ($\sum y^s$), largest spread among all the orders ($\max\{s^c\}$), sum of the spread of all orders ($\sum s^c$), value of objective function (OFV), linear relaxation (LR), lower bound at the end of the search (LB), optimality gap in percentage (gap[%]), and processing time in seconds (time[s]). The calculation of the processing time in the two rows of average results includes the case when the time limit was reached.

The results in Table 2 show that, for instances in set #A, the average optimality gap of the solver with Model-NG was 2.96% (1571.10 s) when $\Phi_2 = \Phi_3 =$✗ and 9.60% (2609.13 s) when $\Phi_2 = \Phi_3 =$✓. Moreover, they show a value of 9.61 bins, the largest spread of 9.50 units, and the sum of the spread of 37.89 units for the first case, and a value of 9.72 bins, the largest spread of 5.61 units, and the sum of the spread of 18.17 units for the second case. Considering the experiments with $\Phi_2 = \Phi_3 =$✓, the solver was able to find an optimal solution and prove its optimality in 4 out of 18 instances. Despite a low number of proven optimal solutions, we observe a reduction of almost 50% on the metrics of the spread of customer orders. For instance, instance gcut11-30-05 presented the largest spread of 9 units and the sum of the spread of 29 units when $\Phi_2 = \Phi_3 =$✗, and the largest spread of 4 units

Table 2: Results for the Model-NG with the set of instances #A.

Φ_2	Φ_3	m	n	Instance	$\sum y^s$	$\max\{s^c\}$	$\sum s^c$	OFV	LR	LB	gap[%]	time[s]
✗	✗	3	10	gcut01-10-03	5	4	10	375	253.71	375.00	0.00	0.06
				gcut05-10-03	3	3	8	144	121.00	144.00	0.00	0.04
				gcut09-10-03	3	3	6	81	61.19	81.00	0.00	0.01
			20	gcut02-20-03	6	6	16	648	510.00	648.00	0.00	0.19
				gcut06-20-03	7	7	18	1029	777.50	1029.00	0.00	18.19
				gcut10-20-03	7	7	18	1344	1114.94	1344.00	0.00	8.16
			30	gcut03-30-03	8	8	20	1536	1306.01	1536.00	0.00	83.95
				gcut07-30-03	11	11	29	4752	3669.97	4752.00	0.00	4.33
				gcut11-30-03	9	9	25	2187	1683.42	2187.00	0.00	2915.15
		5	30	gcut03-30-05	8	8	31	2560	2176.68	2560.00	0.00	43.45
				gcut07-30-05	11	11	47	7920	6116.61	7920.00	0.00	6.42
				gcut11-30-05	9	9	29	3645	2805.71	3240.00	11.11	tl
			50	gcut04-50-05	14	14	57	13720	11579.26	12740.00	7.14	tl
				gcut08-50-05	13	13	53	12740	11315.50	11760.00	7.69	tl
				gcut12-50-05	16	16	65	20480	16297.43	19200.00	6.25	tl
		7	50	gcut04-50-07	14	14	76	19208	16210.96	17836.00	7.14	tl
				gcut08-50-07	13	12	79	17836	15841.70	16464.00	7.69	tl
				gcut12-50-07	16	16	95	28672	22816.40	26880.00	6.25	tl
Average					9.61	9.50	37.89	7715.39	6369.89	7260.89	2.96	1571.10
✓	✓	3	10	gcut01-10-03	5	2	6	411	271.71	411.00	0.00	0.77
				gcut05-10-03	3	2	6	174	136.00	174.00	0.00	0.28
				gcut09-10-03	3	2	4	22	12.00	22.00	0.00	0.31
			20	gcut02-20-03	6	4	11	731	531.00	731.00	0.00	49.16
				gcut06-20-03	7	4	9	1122	801.50	1000.75	10.81	tl
				gcut10-20-03	7	5	13	1477	1141.94	1477.00	0.00	113.76
			30	gcut03-30-03	8	5	11	1667	1333.01	1448.01	13.14	tl
				gcut07-30-03	11	7	15	5019	3708.97	5015.50	0.07	tl
				gcut11-30-03	9	6	11	2360	1713.42	1878.44	20.40	tl
		5	30	gcut03-30-05	8	4	14	2734	2221.68	2369.17	13.34	tl
				gcut07-30-05	11	6	22	8302	6181.61	8109.28	2.32	tl
				gcut11-30-05	9	4	12	3837	2855.71	2979.68	22.34	tl
			50	gcut04-50-05	14	10	30	14450	11654.26	11906.00	17.61	tl
				gcut08-50-05	14	7	19	14229	11390.50	11835.00	16.82	tl
				gcut12-50-05	16	10	37	21317	16382.43	18395.98	13.70	tl
		7	50	gcut04-50-07	14	7	29	19923	16315.96	16667.26	16.34	tl
				gcut08-50-07	14	6	31	19827	15946.70	16569.00	16.43	tl
				gcut12-50-07	16	10	47	29839	22935.40	27021.64	9.44	tl
Average					*9.72*	*5.61*	*18.17*	*8191.17*	*6418.54*	*7111.71*	*9.60*	*2609.13*

and the sum of the spread of 12 units when $\Phi_2 = \Phi_3 =$✓. We highlight most of the reported solutions when $\Phi_2 = \Phi_3 =$✓ were found during the first 300 seconds of the search. As expected, the linear relaxation of Model-NG is weak since it is a Padberg-based model.

We report the results for Model-2S, Model-R3, and Model-U3 considering the set of instances #A in Table 3. We present four sets of experiments: [$\Phi_2 = \Phi_3 =$✗], [$\Phi_2 =$✗ and $\Phi_3 =$✓], [$\Phi_2 =$✓ and $\Phi_3 =$✗], and [$\Phi_2 = \Phi_3 =$✓]. For each model, the results are aggregated according to these experiments, and the numbers of customer orders m and items n. Each entry of the table is an average over three instances, except those in the last row and columns OPT. We present average values in the last row of the table, except in column OPT as it is the summation of the entries. The results in Table 3 show that, for instances in set #A, the average optimality gap of the solver with Model-2S, Model-R3, and Model-U3 were 5.07%, 5.64%, and 6.34%, with the average processing time of 1205.70 s, 1251.60 s and 1424.73 s, respectively. The solver was able to find an optimal solution and prove its optimality in 51 instances (out of $72 = 4 \times 18$) with Model-2S, in 50 instances with Model-R3, and in 48 instances with Model-U3. Despite the number of proven optimal solutions, these results clearly show that computational results get worse in terms of solution quality and processing time as the spread of customer orders is considered and the patterns become more general and complex. Although the patterns are different, the average number of used bins is 9.94 for the three models; this can be explained because the size of the items is relatively large in comparison with the bin's size in the gcut instances.

Table 3: Results for the Model-2S, Model-R3 and Model-U3 with the set of instances #A.

				Model-2S						Model-R3						Model-U3					
Φ_2	Φ_3	n	m	$\sum y^s$	$\max\{s^c\}$	$\sum s^c$	gap[%]	time[s]	OPT	$\sum y^s$	$\max\{s^c\}$	$\sum s^c$	gap[%]	time[s]	OPT	$\sum y^s$	$\max\{s^c\}$	$\sum s^c$	gap[%]	time[s]	OPT
✗	✗	10	3	4.00	3.67	8.00	0.00	0.01	3	4.00	4.00	10.00	0.00	0.01	3	4.00	4.00	9.33	0.00	0.01	3
		20	3	7.00	7.00	18.67	0.00	0.01	3	7.00	7.00	18.67	0.00	0.01	3	7.00	7.00	18.00	0.00	0.04	3
		30	3	9.67	9.67	25.67	0.00	0.01	3	9.67	9.00	23.00	0.00	0.04	3	9.67	9.67	24.00	0.00	0.09	3
			5	9.67	9.67	34.00	0.00	0.01	3	9.67	9.67	32.00	0.00	0.04	3	9.67	9.67	36.00	0.00	0.10	3
		50	5	14.67	14.67	60.33	0.00	0.42	3	14.67	14.67	55.33	0.00	0.20	3	14.67	14.67	52.67	0.00	0.70	3
			7	14.67	14.67	79.67	0.00	0.48	3	14.67	14.00	76.33	0.00	0.21	3	14.67	14.33	76.00	0.00	0.58	3
	✓	10	3	4.00	2.00	5.33	0.00	0.03	3	4.00	2.33	5.33	0.00	0.04	3	4.00	2.33	5.33	0.00	0.10	3
		20	3	7.00	4.67	11.00	0.00	13.97	3	7.00	5.00	11.00	0.00	21.77	3	7.00	5.00	10.67	0.00	71.95	3
		30	3	9.67	6.67	14.00	0.00	616.65	3	9.67	6.33	13.67	0.00	689.18	3	9.67	6.33	13.67	5.13	1483.50	2
			5	9.67	5.67	16.33	10.37	2499.03	1	9.67	5.67	16.33	10.37	2941.96	1	9.67	6.00	16.33	10.37	3412.39	1
		50	5	14.67	9.00	24.00	13.14	tl	0	14.67	9.67	24.67	19.91	tl	0	14.67	10.67	24.67	20.37	tl	0
			7	14.67	7.33	30.33	22.62	tl	0	14.67	7.00	30.33	31.83	tl	0	14.67	8.33	30.67	32.29	tl	0
✓	✗	10	3	4.00	2.00	6.00	0.00	0.03	3	4.00	2.00	5.67	0.00	0.03	3	4.00	2.00	5.67	0.00	0.04	3
		20	3	7.00	4.33	12.33	0.00	4.75	3	7.00	4.33	12.33	0.00	5.17	3	7.00	4.33	11.67	0.00	15.30	3
		30	3	9.67	6.33	18.33	0.00	54.05	3	9.67	6.33	17.67	0.00	253.30	3	9.67	6.33	18.00	0.00	783.38	3
			5	9.67	5.33	22.67	12.22	2483.01	1	9.67	5.33	22.00	12.22	2662.00	1	9.67	5.00	21.67	5.56	2749.35	2
		50	5	14.67	8.33	37.00	6.67	1864.25	2	14.67	8.33	35.00	6.67	1356.51	2	14.67	8.33	38.33	6.67	1667.48	2
			7	14.67	5.67	36.00	16.67	3373.18	1	14.67	5.33	36.33	12.22	2861.05	1	14.67	6.33	40.33	26.19	tl	0
	✓	10	3	4.00	2.00	5.33	0.00	0.08	3	4.00	2.00	5.33	0.00	0.07	3	4.00	2.00	5.33	0.00	0.14	3
		20	3	7.00	4.33	11.33	0.00	20.55	3	7.00	4.33	11.33	0.00	23.20	3	7.00	4.33	10.67	0.00	159.00	3
		30	3	9.67	6.33	14.00	0.00	932.71	3	9.67	6.33	13.67	0.38	1613.09	2	9.67	6.33	13.67	0.38	2249.53	2
			5	9.67	5.33	16.33	12.13	2673.67	1	9.67	5.33	16.33	12.13	3210.56	1	9.67	5.00	16.33	12.62	tl	0
		50	5	14.67	8.33	25.33	6.98	tl	0	14.67	8.33	25.67	7.32	tl	0	14.67	8.67	25.33	11.29	tl	0
			7	14.67	6.33	30.33	20.86	tl	0	14.67	6.00	31.00	22.30	tl	0	14.67	6.00	30.67	21.23	tl	0
Average/Sum				9.94	6.64	23.43	5.07	1205.70	51	9.94	6.60	22.88	5.64	1251.60	50	9.94	6.78	23.13	6.34	1424.73	48

4.2. Results for the set of instances #B

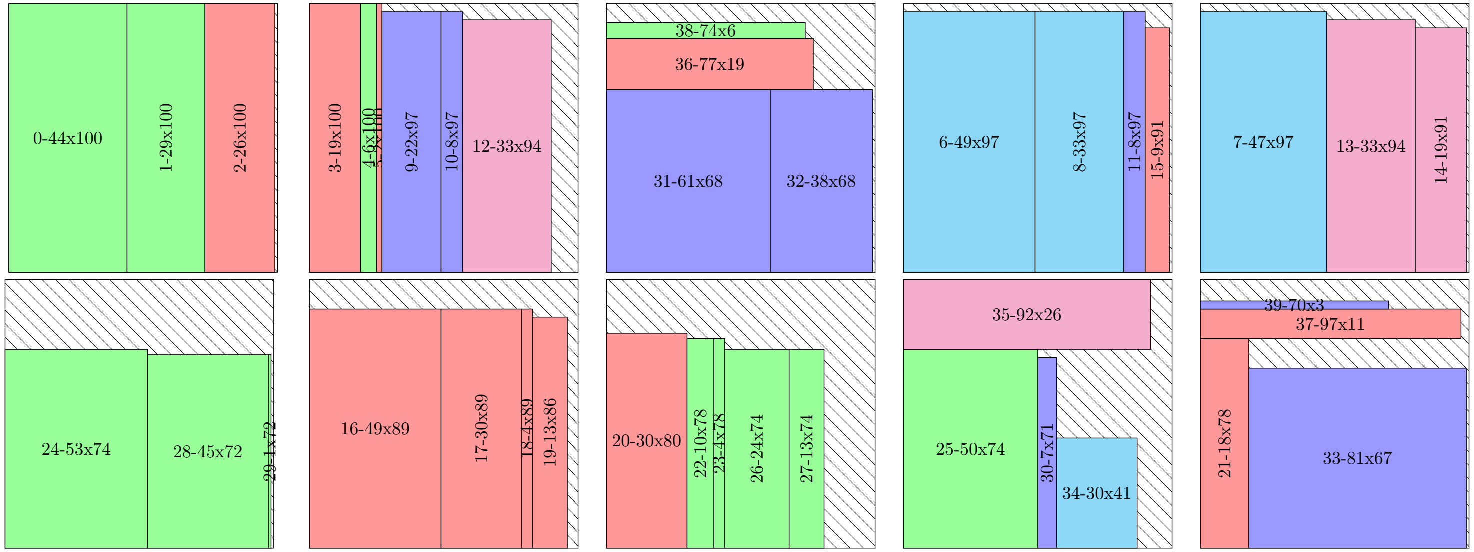
The set of instances #B is composed of 35 instances, based on the classical 2BPP problem instances proposed in Berkey & Wang (1987) and Lodi et al. (1999). These instances were randomly generated by these authors and have distinct characteristics, such as items with different shapes and items with small sizes in relation to the size of the bins – see Lodi et al. (1999) for a detailed description. Again, we arbitrarily aggregated the items to generate customer orders. Thus, the adapted instances: with $n = 20$ items have $m = 3$ customer orders (minimum of 2 items per customer); with $n = 40$ items have $n = 3, 5$ customer orders (minimum of 3 items per customer); and, with $n = 60$ items have $n = 5, 7$ customer orders (minimum of 4 items per customer).

We report the results for Model-2S, Model-R3, and Model-U3 considering the set of instances #B in Table 4. We report the results for Model-2S, Model-R3, and Model-U3 considering the set of instances #A in Table 3. For each model, the results are aggregated according to the four experiments with Φ_2 and Φ_3, and the numbers of customer orders m and items n. Each entry of the table is an average value over seven instances, except those in the last row and columns OPT. The results in Table 4 show that, for instances in set #B, the average optimality gap of the solver with Model-2S, Model-R3, and Model-U3 were 7.95%, 8.32%, and 10.07%, with the average processing time of 1551.76 s, 1544.83 s and 1815.35 s, respectively. The solver was able to find an optimal solution and prove its optimality in 83 instances (out of $140 = 4 \times 35$) with Model-2S, in 86 instances with Model-R3, and in 72 instances with Model-U3. Once again, these results clearly show that computational results get worse in terms of solution quality and processing time as the spread of customer orders is considered and the patterns become more general and complex. In contrast to the results of the previous section, as these instances have items with very different shapes, we observe a reduction in the number of bins used and/or the order spread metrics as the patterns become more complex.

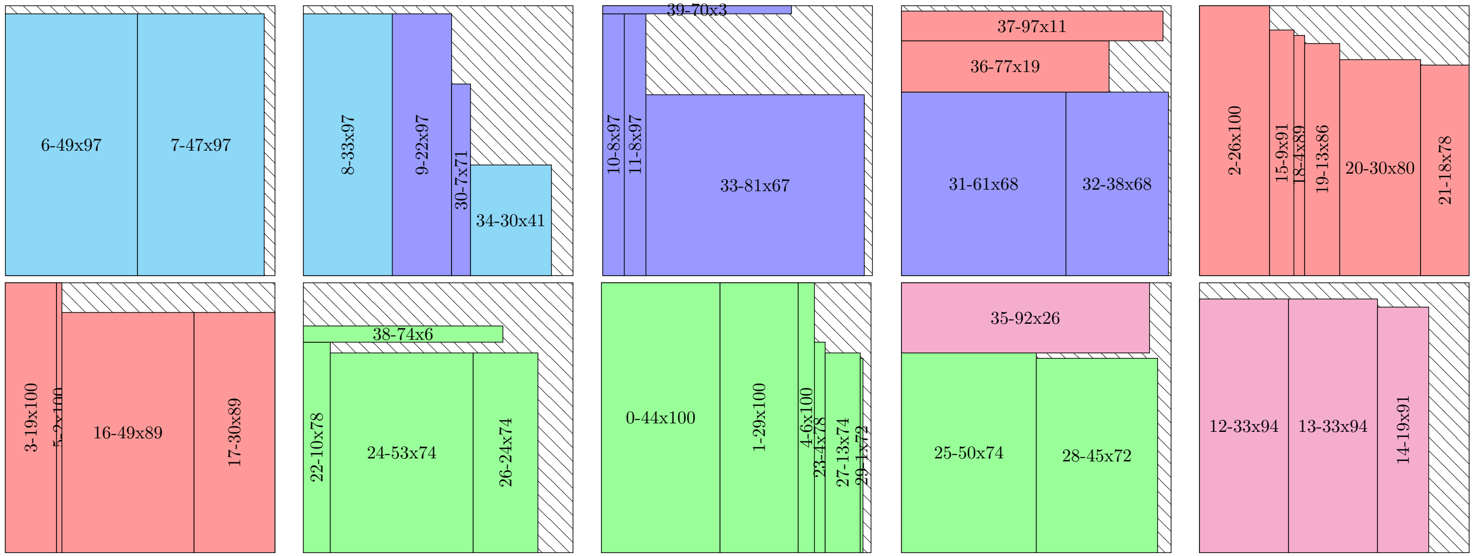
Alternatively stated, in the context of a branch-and-cut of a general-purpose ILP solver, the results show that the integration of the order spread to the 2BPP-OS did not make it easier to solve the models. In Fig. 4, we present two optimal solutions for a problem instance of set #B with $n = 40$ items and $c = 5$ customer orders when $\Phi_2 = \Phi_3 =$✗ and $\Phi_2 = \Phi_3 =$✓. The items of the same customer order are represented in the same color. It is easy to see in the figure that customer orders in the second case are processed quickly (i.e. they contribute to the service level and flow of the operation); in contrast, in the first case, there are items from different customer orders being cut from the same bin, which harms the order spread metrics.

Table 4: Results for the Model-2S, Model-R3 and Model-U3 with the set of instances #B.

				Model-2S						Model-R3						Model-U3					
Φ_2	Φ_3	n	m	$\sum y^s$	$\max\{s^c\}$	$\sum s^c$	gap[%]	time[s]	OPT	$\sum y^s$	$\max\{s^c\}$	$\sum s^c$	gap[%]	time[s]	OPT	$\sum y^s$	$\max\{s^c\}$	$\sum s^c$	gap[%]	time[s]	OPT
✗	✗	20	3	8.43	8.29	21.14	0.00	0.01	7	8.43	8.29	22.29	0.00	0.02	7	8.43	8.29	22.29	0.00	0.04	7
		40	3	11.71	11.71	32.71	0.00	1.64	7	11.71	11.43	32.43	0.00	0.09	7	11.57	11.43	32.00	1.43	514.55	6
			5	11.71	11.57	48.43	0.00	1.47	7	11.71	11.57	46.14	0.00	0.10	7	11.57	11.43	46.43	1.43	514.52	6
		60	5	21.86	21.43	94.14	0.00	4.28	7	21.86	21.71	91.14	0.00	0.53	7	21.86	21.71	90.43	1.24	515.54	6
			7	21.86	20.71	116.86	0.00	4.84	7	21.86	21.00	116.86	0.00	0.51	7	21.86	21.00	118.43	1.24	515.57	6
	✓	20	3	8.43	4.86	11.00	0.00	15.25	7	8.43	4.86	11.00	0.00	12.19	7	8.43	4.86	11.00	0.00	61.28	7
		40	3	11.71	5.43	14.29	3.83	1190.50	5	11.71	5.29	13.86	0.95	862.97	6	11.57	5.86	14.57	5.95	1895.51	4
			5	11.71	5.00	16.29	8.69	2929.60	2	11.71	4.86	16.00	5.68	2705.69	3	11.57	5.14	17.43	13.49	3267.00	1
		60	5	21.86	13.86	36.71	22.72	tl	0	21.86	13.00	36.71	28.64	tl	0	21.86	13.43	38.57	24.95	tl	0
			7	21.86	9.71	41.86	29.39	tl	0	21.86	9.43	40.00	30.87	tl	0	21.86	10.29	42.71	28.92	tl	0
✓	✗	20	3	8.43	4.71	13.43	0.00	0.84	7	8.43	4.71	13.57	0.00	1.33	7	8.43	4.71	12.86	0.00	2.86	7
		40	3	11.71	5.43	16.00	5.24	1190.04	5	11.71	5.29	15.86	2.38	908.07	6	11.57	6.00	17.57	13.27	1640.65	4
			5	11.71	4.86	22.43	2.86	756.55	6	11.71	4.86	23.29	2.86	606.25	6	11.57	5.00	22.29	4.35	1133.73	5
		60	5	21.86	10.29	47.29	19.74	3215.85	1	21.86	10.14	48.29	22.31	3210.78	1	21.86	10.57	49.86	23.47	tl	0
			7	21.86	7.71	50.29	17.12	2742.24	2	21.86	7.57	49.43	19.56	2946.10	3	21.86	8.57	55.00	22.39	2636.26	2
	✓	20	3	8.43	4.71	11.14	0.00	18.44	7	8.43	4.71	11.14	0.00	30.01	7	8.43	4.71	11.14	0.00	340.89	7
		40	3	11.71	5.43	14.29	5.13	1180.75	5	11.71	5.29	14.00	2.41	1611.94	5	11.57	5.86	14.71	9.69	1668.74	4
			5	11.71	4.86	16.29	4.79	3383.07	1	11.71	4.86	16.14	3.35	tl	0	11.57	5.29	18.00	8.83	tl	0
		60	5	21.86	10.29	36.71	19.94	tl	0	21.86	10.14	35.29	22.55	tl	0	21.86	10.00	36.43	19.18	tl	0
			7	21.86	8.14	41.86	19.54	tl	0	21.86	7.86	39.43	24.77	tl	0	21.86	8.43	41.86	21.59	tl	0
Average/Sum				15.11	8.95	35.16	7.95	1551.76	83	15.11	8.84	34.64	8.32	1544.83	86	15.06	9.13	35.68	10.07	1815.35	72



(a) $\Phi_2 = \Phi_3 =$✗: 10 bins, $\max\{s^c\}$ of 9 units, and $\sum s^c$ of 37 units.



(b) $\Phi_2 = \Phi_3 =$✓: 10 bins, $\max\{s^c\}$ of 5 units, and $\sum s^c$ of 16 units.

Figure 4: Two optimal solutions for Model-2S considering a problem instance of set #B with $n = 40$ items and $c = 5$ customer orders.

5. Conclusions

We addressed four variants of the two-dimensional bin packing problem with customer order spread. The problem arises in manufacturing industries looking for minimal waste solutions that are responsive in terms of quickly processing customer orders. Since the problem may appear in different environments, we proposed models considering different cutting profiles. We proposed models of non-guillotine, 2-stage, restricted 3-stage, and unrestricted 3-stage3 patterns. The results of the computational experiments showed it is possible to obtain satisfactory solutions in terms of the metrics of order spread, but that is also optimal in terms of bin usage. The obtained solutions may seem similar in terms of bin usage, but they are completely different from those solutions from approaches that do not consider the customer order spread.

A path for future research is to extend the pseudo-polynomial models of Silva et al. (2010) for the 2BPP to deal with the spread of customer orders. Note that our models are based on the allocation of items to bins; their performance gets worse when the number of items is large. Although such an extension does not seem straightforward, those models can deal with 2BPP's problem instances with a large number of items. One could also consider other practical requirements for cutting operations or production scheduling as open stacks, due dates, and other cutting profiles, as p-group patterns. Other research paths could address the cutting of three-dimensional objects and/or distinct relations of three terms in the objective function (e.g. when the minimization of the customer order spread is a primary objective).

Funding

The authors would like to thank the São Paulo Research Foundation (FAPESP-Brazil) [grant numbers 2020/00747-2, 2016/01860-1] and the National Council for Scientific and Technological Development (CNPq-Brazil) [grant number 304144/2019-3] for the financial support. Research was carried out using the computational resources of the Center for Mathematical Sciences Applied to Industry (CeMEAI) funded by FAPESP-Brazil [grant number 2013/07375-0].

Competing Interests

The authors have no relevant financial or non-financial interests to disclose.

Authors contributions

All authors contributed to the study's conception and design. Material preparation, data collection, and analysis were performed by Mateus Martin, Horacio Hideki Yanasse, Maristela O. Santos, and Reinaldo Morabito. The first draft of the manuscript was written by Mateus Martin and all authors commented on previous versions of the manuscript.

Data Availability

The datasets analyzed during the current study are available from the corresponding author upon reasonable request.

References

Alvelos, F., Chan, T., Vilaça, P., Gomes, T., Silva, E., & Valério de Carvalho, J. (2009). Sequence based heuristics for two-dimensional bin packing problems. *Engineering Optimization*, *41*, 773–791. doi:10.1080/03052150902835960.

Arbib, C., Marinelli, F., & Pizzuti, A. (2021). Number of bins and maximum lateness minimization in two-dimensional bin packing. *European Journal of Operational Research*, *291*, 101–113. doi:10.1016/j.ejor.2020.09.023.

Beasley, J. E. (1985a). Algorithms for Unconstrained Two-Dimensional Guillotine Cutting. *The Journal of the Operational Research Society*, *36*, 297. doi:10.2307/2582416.

Beasley, J. E. (1985b). An Exact Two-Dimensional Non-Guillotine Cutting Tree Search Procedure. *Operations Research*, *33*, 49–64. doi:10.1287/opre.33.1.49.

Bennell, J. A., Soon Lee, L., & Potts, C. N. (2013). A genetic algorithm for two-dimensional bin packing with due dates. *International Journal of Production Economics*, *145*, 547–560. doi:10.1016/j.ijpe.2013.04.040.

Berkey, J. O., & Wang, P. Y. (1987). Two-dimensional finite bin-packing algorithms. *Journal of the Operational Research Society*, *38*, 423–429. doi:10.1057/jors.1987.70.

Boschetti, M. A., & Mingozzi, A. (2003a). The two-dimensional finite bin packing problem - Part I: New lower bounds for the oriented case. *4OR*, *1*, 27–42. doi:10.1007/s10288-002-0005-z.

Boschetti, M. A., & Mingozzi, A. (2003b). The two-dimensional finite bin packing problem - Part II: New lower and upper bounds. *4OR*, *1*, 135–147. doi:10.1007/s10288-002-0006-y.

Coffman, E. G., Jr., Garey, M. R., Johnson, D. S., & Tarjan, R. E. (1980). Performance bounds for level-oriented two-dimensional packing algorithms. *SIAM Journal on Computing*, *9*, 808–826. doi:10.1137/0209062.

Cui, Y.-P., Cui, Y., & Tang, T. (2015). Sequential heuristic for the two-dimensional bin-packing problem. *European Journal of Operational Research*, *240*, 43–53. doi:10.1016/j.ejor.2014.06.032.

Dyson, R. G., & Gregory, A. S. (1974). The Cutting Stock Problem in the Flat Glass Industry. *Journal of the Operational Research Society*, *25*, 41–53. doi:10.1057/jors.1974.5.

Foerster, H., & Wäscher, G. (1998). Simulated annealing for order spread minimization in sequencing cutting patterns. *European Journal of Operational Research*, *110*, 272–281. doi:10.1016/S0377-2217(97)00257-9.

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

Gilmore, P. C., & Gomory, R. E. (1965). Multistage Cutting Stock Problems of Two and More Dimensions. *Operations Research*, *13*, 94–120. doi:10.1287/opre.13.1.94.

Iori, M., De Lima, V. L., Martello, S., Miyazawa, F. K., & Monaci, M. (2021). Exact solution techniques for two-dimensional cutting and packing. *European Journal of Operational Research*, *289*, 399–415. doi:10.1016/j.ejor.2020.06.050.

Linhares, A., & Yanasse, H. H. (2002). Connections between cutting-pattern sequencing, VLSI design, and flexible machines. *Computers and Operations Research*, *29*, 1759–1772. doi:10.1016/S0305-0548(01)00054-5.

Lodi, A., Martello, S., & Monaci, M. (2002a). Two-dimensional packing problems: A survey. *European Journal of Operational Research*, *141*, 241–252. doi:10.1016/S0377-2217(02)00123-6.

Lodi, A., Martello, S., & Vigo, D. (1999). Heuristic and metaheuristic approaches for a class of two-dimensional bin packing problems. *INFORMS Journal on Computing*, *11*, 345–357. doi:10.1287/ijoc.11.4.345.

Lodi, A., Martello, S., & Vigo, D. (2002b). Recent advances on two-dimensional bin packing problems. *Discrete Applied Mathematics*, *123*, 379–396. doi:10.1016/S0166-218X(01)00347-X.

Lodi, A., Martello, S., & Vigo, D. (2004). Models and Bounds for Two-Dimensional Level Packing Problems. *Journal of Combinatorial Optimization*, *8*, 363–379. doi:10.1023/B:JOCO.0000038915.62826.79.

Madsen, O. B. G. (1979). Glass cutting in a small firm. *Mathematical Programming*, *17*, 85–90. doi:10.1007/BF01588227.

Madsen, O. B. G. (1983). Production planning in a small firm in the industry glass. In B. Wilson, C. C. Berg, & D. French (Eds.), *Efficiency of Manufacturing Systems* (pp. 245–253). Boston, MA: Springer US. doi:10.1007/978-1-4684-4475-9.

Madsen, O. B. G. (1988). An application of travelling-salesman routines to solve pattern-allocation problems in the glass industry. *Journal of the Operational Research Society*, *39*, 249–256. doi:10.1057/jors.1988.42.

Martin, M., Yanasse, H. H., Santos, M. O., & Morabito, R. (2022). Models for two- and three-stage two-dimensional cutting stock problems with a limited number of open stacks. *International Journal of Production Research*, *0*, 1–22. doi:10.1080/00207543.2022.2070882.

Melega, G. M., De Araujo, S. A., Jans, R., & Morabito, R. (2022). Formulations and exact solution approaches for a coupled bin-packing and lot-sizing problem with sequence-dependent setups. *Flexible Services and Manufacturing Journal*, . doi:10.1007/s10696-022-09464-9.

Padberg, M. (2000). Packing small boxes into a big box. *Mathematical Methods of Operations Research*, *52*, 1–21. doi:10.1007/s001860000066.

Polyakovskiy, S., & M'Hallah, R. (2021). Just-in-time two-dimensional bin packing. *Omega*, *102*, 102311. doi:10.1016/j.omega.2020.102311.

Puchinger, J., & Raidl, G. R. (2007). Models and algorithms for three-stage two-dimensional bin packing. *European Journal of Operational Research*, *183*, 1304–1327. doi:10.1016/j.ejor.2005.11.064.

Scheithauer, G. (2018). *Introduction to cutting and packing optimization: problems, modeling approaches, solution methods*. International Series in Operations Research and Management Science. Springer. doi:10.1007/978-3-319-64143-0.

Silva, E., Alvelos, F., & Valério de Carvalho, J. M. (2010). An integer programming model for two- and three-stage two-dimensional cutting stock problems. *European Journal of Operational Research*, *205*, 699–708. doi:10.1016/j.ejor.2010.01.039.

Wäscher, G., Haußner, H., & Schumann, H. (2007). An improved typology of cutting and packing problems. *European Journal of Operational Research*, *183*, 1109–1130. doi:10.1016/j.ejor.2005.12.047.