



## Discrete Optimization

## Mixed-integer programming models for irregular strip packing based on vertical slices and feasibility cuts



Juan J. Lastra-Díaz\*, M. Teresa Ortuño

Department of Statistics and Operational Research, Institute of Interdisciplinary Mathematics, UCM Research Group HUMLOG, Complutense University of Madrid, Spain

## ARTICLE INFO

## Article history:

Received 31 May 2022

Accepted 4 August 2023

Available online 9 August 2023

## Keywords:

Packing

Integer programming

Irregular strip packing

Nesting

Cutting

## ABSTRACT

The irregular strip-packing problem, also known as nesting or marker making, is defined as the automatic computation of a non-overlapping placement of a set of non-convex polygons onto a rectangular strip of fixed width and unbounded length, such that the strip length is minimized. Nesting methods based on heuristics are a mature technology, and currently, the only practical solution to this problem. However, recent performance gains of the Mixed-Integer Programming (MIP) solvers, together with the known limitations of the heuristics methods, have encouraged the exploration of exact optimization models for nesting during the last decade. Despite the research effort, there is room to improve the efficiency of the current family of exact MIP models for nesting. In order to bridge this gap, this work introduces a new family of continuous MIP models based on a novel formulation of the NoFit-Polygon Covering Model (NFP-CM), called NFP-CM based on Vertical Slices (NFP-CM-VS). Our new family of MIP models is based on a new convex decomposition of the feasible space of relative placements between pieces into vertical slices, together with a new family of valid inequalities, symmetry breakings, and variable eliminations derived from the former convex decomposition. Our experiments show that our new NFP-CM-VS models outperform the current state-of-the-art MIP models. Ten instances are solved up to optimality within one hour for the first time, including one with 27 pieces. Finally, we provide a detailed reproducibility protocol and dataset as supplementary material to allow the exact replication of our models, experiments, and results.

© 2023 The Author(s). Published by Elsevier B.V.

This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

## 1. Introduction

Cutting and packing regular (convex) and irregular (non-convex) polygons onto a rectangular strip with unbounded length is a tedious and usual task in most manufacturing industries based on the cutting of any flat material. For instance, Milenkovic, Daniels, & Li (1991) study the nesting problem for the fashion and apparel industry, whilst Heistermann & Lengauer (1995) and Whelan & Batchelor (1993) do it for leather manufacturing, Elamvazuthi, Kamaruddin, & Azmi (2009) in furniture, Han, Bennell, Zhao, & Song (2013) in the glass industry, Alves, Brás, Valério de Carvalho, & Pinto (2012) in the automotive industry, and Cheok & Nee (1991) in shipbuilding. The irregular strip-packing methods aim to compute a non-overlapping placement of a set of

irregular polygons onto a fixed-width rectangular strip with unbounded length, called the *board*, whose length is the minimum among all feasible placements. Another closely related problem, called two-dimensional bin packing (Lodi, Martello, & Vigo, 2002), is defined as the computation of a non-overlapping placement of a set of polygons onto a larger closed polygon, called the *bin*, to minimize the number of bins required. Although most of bin packing problems are defined for rectangular bins and pieces (Iori, de Lima, Martello, Miyazawa, & Monaci, 2021), we also find many irregular bin packing problems in the aforementioned industries. Strip and bin packing problems, and all their variants concerning the geometry of the pieces or boards, belong to the broader family of Cutting and Packing (C&P) problems categorized by Dyckhoff (1990) and Wäscher, Haufner, & Schumann (2007), and extensively reviewed by Sweeney & Paternoster (1992), Dowsland & Dowsland (1992), Wang & Wäscher (2002), and Bennell, Oliveira, & Wäscher (2013).

Research on the irregular strip and bin packing problems can be traced back to the pioneering Linear Programming (LP) models

\* Corresponding author.

E-mail addresses: [jlastra@ucm.es](mailto:jlastra@ucm.es) (J.J. Lastra-Díaz), [mteresa@ucm.es](mailto:mteresa@ucm.es) (M.T. Ortuño).

for rectangular bin packing introduced by Gilmore & Gomory (1965), and the pioneering heuristic methods for irregular strip packing proposed by Art (1966), Adamowicz & Albano (1976a,b), and Albano & Sapuppo (1980) in the late nineteen sixties and seventies. These early works introduce many of the basic ideas subsequently exploited by all heuristics methods reported in the literature, such as the notion of a feasible non-overlapping region between pieces based on the No-Fit Polygon (NFP) representation (Bennell & Oliveira, 2008), and the sequential placement of pieces based on bottom-left heuristics. The boundary and outer region of the no-fit polygon  $NFP_{AB}$  set the feasible positions in which polygon  $B$  can be placed, without rotating, into a non-overlapping position with respect to polygon  $A$ . Thus, the no-fit polygon allows one to compute the feasible relative placements between polygons  $A$  and  $B$  a priori by checking whether their relative position vector  $\delta_{AB} = r_B - r_A$  belongs to either the boundary or the outer region of  $NFP_{AB}$ , where  $r_A$  and  $r_B$  denote their corresponding reference points. This property has converted the NFP into the most broadly adopted and effective geometric representation for nesting reported in the literature, both for the families of heuristics methods (Bennell & Oliveira, 2009) and for exact mathematical models (Leao, Toledo, Oliveira, Carravilla, & Álvarez-Valdés, 2020).

Fowler, Paterson, & Tanimoto (1981) show that the irregular strip-packing problem is NP-complete. For this reason, most practical solutions reported in the literature since the pioneering work of Art (1966) are based on sequential placement heuristics to build efficiently feasible solutions that are combined with meta-heuristics for exploring the space of feasible solutions, as shown in most of surveys on nesting (Bennell & Oliveira, 2009; Dowsland & Dowsland, 1995; Dowsland, Vaid, & Dowsland, 2002; Hopper & Turton, 2001; Riff, Bonnaire, & Neveu, 2009). Current heuristics-based methods can efficiently compute acceptable solutions for large problem instances, which have encouraged their early and extensive adoption in all industries mentioned above, especially in those industries with as significant and permanent diversity of products as the garment industry (Puri, 2013). Elkeran (2013) introduces the current state-of-the-art heuristic method for nesting, called Guided Cuckoo Search (GCS), which defines a two-stage method based on piece clustering and NFP-based bottom-left heuristics (Gomes & Oliveira, 2002) to build an initial feasible solution that is shrunk by solving an overlap minimization problem using a variant of the cuckoo search meta-heuristics. Although GCS was introduced almost a decade ago, subsequent works have been unable to outperform its results, as shown by Pinheiro, Amaro Júnior, & Saraiva (2016, Table 3), Sato, Martins, & Tsuzuki (2016a, Table 5), Cherri, Carravilla, & Toledo (2016a, Table 4), Mundim, Andretta, & de Queiroz (2017, Table 6), Amaro Júnior, Pinheiro, & Coelho (2017, Table 2), and Sato, Martins, Gomes, & Tsuzuki (2019, Tables 4–5). GCS (Elkeran, 2013) also significantly outperforms a commercial nesting system broadly adopted in the industry (Annamalai Vasantha, Jagadeesan, Corney, Lynn, & Agrawal, 2016, figure 3). However, the heuristics methods require up to one hour to improve their initial solutions (Mundim et al., 2017, Table 6), and they can neither provide an optimality proof nor a gap measure to the optimal solution.

Recent performance gains of the Mixed-Integer Programming (MIP) solvers, and the limitations of the heuristics methods mentioned above, have encouraged the exploration of exact mathematical models for nesting during the last decade, as shown by Leao et al. (2020). The mathematical models for nesting can be categorized into continuous, discrete, or semi-continuous models according to the type of decision variable used to represent the position of the pieces. Li (1994, §8) introduces the first continuous MIP model for nesting reported in the literature that is able to solve a limitation of the pioneering LP compaction model of Li & Milenkovic (1993, 1995). Although Li's model is not experi-

mentally evaluated, it sets the two main features of the family of continuous MIP models based on the NFP as follows: (1) the convex decomposition of the outer NFP feasible regions; and (2) the definition of mutually-exclusive binary variables to set the pairwise non-overlapping constraints between pieces defining the feasible regions for their relative placement. Thus, we can say that Li's model sets the basic continuous linear MIP model for nesting without rotations, from which all subsequent works, including this work, derive more tightened and refined formulations. For instance, Dean (2002, §5) introduces and evaluates for the first time a MIP model for nesting based on a refinement of the Daniels, Li, & Milenkovic (1994) bin packing model that is essentially identical to the Li's model. Subsequently, Fischetti & Luzzi (2009) (F&L) introduce a refinement of Li's model based on lifting the big-M formulation and a branching-priority algorithm to guide the Branch&Bound (B&B) exploration by fixing large sets of feasible relative placements among pieces; however, they neither mention nor provide a strategy and method to build a family of valid feasibility cuts among three or more pieces on any particular geometric decomposition, as done herein. Álvarez-Valdes, R., Martínez, A., & Tamarit, J. M. (2013) improve the F&L model by introducing a new MIP model called HS2, which is based on a detailed convex decomposition of the feasible regions into horizontal slices, a lifting for the bound constraints of the continuous variables, six new branching strategies, and the  $x$ -axis ordering of identical pieces to remove all symmetric solutions derived from their permutation.

More recently, Cherri et al. (2016b) introduce two continuous MIP models improving the HS2 model (Álvarez-Valdes, R., Martínez, A., & Tamarit, J. M. (2013)), together with the first MIP model integrating discrete rotations, which are based on the convex decomposition of the pieces and the definition of the non-overlapping constraints between pieces by using the convex no-fit polygons among their convex parts. The first model, called Direct Trigonometry Model (DTM), uses a direct trigonometry function encoding the separation lines defined by the edges of the polygons to build the non-overlapping constraints between pairs of convex parts from two pieces, whilst the second model, called NoFit-Polygon Covering Model (NFP-CM), uses the convex NFP between convex parts and several valid inequalities. DTM and NFP-CM use the same  $x$ -axis symmetry-breaking for identical pieces proposed by Álvarez-Valdes, Martínez, & Tamarit (2013). Subsequently, Rodrigues, Cherri, & Mundim (2017) improve the NFP-CM model by breaking the symmetries of the feasible space for the relative placements between pieces, setting the current state-of-the-art in terms of performance among the family of exact continuous mathematical models for irregular strip packing. However, the Improved NFP-CM model (Rodrigues et al., 2017) is only able to solve small problem instances with up to 17 convex pieces with simple geometry (Rodrigues et al., 2017, Table 2), and unlike the NFP-CM model (Cherri et al., 2016b, Table 3), it has not been evaluated on complex small and large problem instances yet. On the other hand, Santoro & Lemos (2015) introduce an exact MIP model to compute efficiently tighter upper bounds for the irregular nesting problem based on approximating the input pieces by parallel-chamfered  $n$ -gons with up to 8 sides, as shown in Santoro & Lemos (2015, fig. 2), and generalizing the disjunctive MIP model for rectangular strip packing proposed by Sawaya & Grossmann (2005). More recently, Kimms & Király (2022) introduce an extension of the DTM model (Cherri et al., 2016b) integrating some industry-related constraints, such as the ability to group pieces sharing the same orientation and the consideration of known cutting patterns for several pieces.

The main aim of this work is to introduce a new family of continuous MIP models for irregular strip packing without rotations capable of improving the performance of the current state-of-the-art NFP-CM models (Cherri et al., 2016b; Rodrigues et al., 2017).

Our main motivation is to propose and test the hypothesis that a new and more tightened formulation of the two aforementioned models might improve their performance. Thus, our new family of MIP models, called NFP-CM based on Vertical Slices (NFP-CM-VS), is a novel and more tightened formulation of the state-of-the-art NFP-CM models (Cherri et al., 2016b; Rodrigues et al., 2017) that is based on a new disjoint convex decomposition of the feasible regions between convex parts into vertical slices, together with a new family of valid inequalities, symmetry breakings, and variable reductions derived from the former geometric decomposition. A second aim of this work is to carry out a fair and reproducible comparison of our new models with the state-of-the-art aforementioned ones (Cherri et al., 2016b; Rodrigues et al., 2017) by replicating them and implementing our new MIP models into a single software platform integrating the latest version of the Gurobi solver, which is provided together with a detailed reproducibility protocol and dataset as supplementary material to allow the exact replication of all our models, experiments, and results. A third aim is to bridge the lack of reproducibility resources hampering the independent replication and confirmation of previously reported models and results by providing the aforementioned reproducibility resources. And finally, our fourth aim is a confirming evaluation for the first time of the state-of-the-art Improved NFP-CM model (Rodrigues et al., 2017) in the same problem instances reported for NFP-CM (Cherri et al., 2016b, Tables 1–4).

The main research problem tackled in this work is the definition and evaluation of new and more tightened MIP models for irregular strip packing which outperform the current state-of-the-art family of NFP-CM models (Cherri et al., 2016b; Rodrigues et al., 2017). Our main contribution is the introduction of a new family of MIP models, called NFP-CM-VS, which is based on a new convex decomposition of the feasible regions between convex parts into vertical slices, together with a new family of valid inequalities, symmetry breakings, and variable eliminations derived from the former geometric decomposition, including feasibility cuts among three pieces for the first time in the literature. Other contributions are the introduction of the first reproducible experimental survey in this line of research, and the first independent confirmation of the state-of-the-art MIP models introduced by Cherri et al. (2016b) and Rodrigues et al. (2017).

The rest of the paper is structured as follows. Section 2 reviews the literature on exact mathematical models for irregular strip packing. Section 3 introduces our new family of MIP models, whilst the Section 4 details our experimental setup and results, and Section 5 introduces our discussion of the results. The final section summarizes our main conclusions and future work. Subsequently, Appendix A introduces all raw output data generated for each MIP model evaluated herein not included in the results section because of lack of room. Finally, Appendix B introduces a detailed reproducibility protocol based on our supplementary dataset (Lastra-Díaz & Ortuño, 2023) to allow the exact replication of all our models, experiments, and results. Both aforementioned appendices are provided as supplementary material.

## 2. Related work on exact models for nesting

Our introduction has provided a review of the family of continuous MIP models for irregular strip packing based on the NFP, to which this work belongs. In addition, this section introduces a comprehensive categorization of the literature on exact mathematical models for nesting, together with an extended review of the linear MIP models based on the NFP. However, for a detailed and recent review of the family of mathematical models, we refer the reader to the survey of Leao et al. (2020) and the survey on geometric representations for nesting of Bennell & Oliveira (2008).

### 2.1. The irregular strip-packing problem

Although this work tackles the translation-only version, the general irregular strip-packing problem with rotations can be defined regardless of the geometric representation of the non-overlapping constraints as follows. Let  $\mathcal{P} = \{P_i\}_{1 \leq i \leq N}$  be the set of pieces to be placed onto the board  $\mathcal{B}$ ,  $T = \{t_i\}_{1 \leq i \leq N}$ , and  $R = \{r_i\}_{1 \leq i \leq N}$  the set of translation vectors and rotations defining a particular feasible solution,  $L$  the length of the board,  $\mathcal{O}$  the set of admissible rotations defining the domain for each  $r_i$  decision variable;  $P_i(r_i) \subset \mathbb{R}^2$  denotes an orientation of piece  $P_i$ , and  $\text{int}(A)$  denotes the interior set for any set  $A \subset \mathbb{R}^2$ . The minimization problem (1) defines the optimal solution to the irregular strip-packing problem, where Constraints (2) prevent the overlapping of pieces, and Constraints (3) force the pieces to be entirely contained in the board.

$$\text{minimize } L \quad (1)$$

$$\text{subject to } \text{int}(P_i(r_i) \oplus t_i) \cap \text{int}(P_j(r_j) \oplus t_j) = \emptyset, \quad 1 \leq i < j \leq N \quad (2)$$

$$(P_i(r_i) \oplus t_i) \subseteq \mathcal{B}, \quad 1 \leq i \leq N \quad (3)$$

$$t_i \in \mathbb{R}^2, r_i \in \mathcal{O}, L \in \mathbb{R}^+ \quad (4)$$

### 2.2. Categorization of exact mathematical models

Fig. 1 shows our categorization of the family of exact mathematical models for irregular strip packing reported in the literature, which can be divided into three large families as follows. First, the family of linear MIP models (A) based either on the NFP (A.1) or direct trigonometry (A.2) according to the geometric representation used for building the non-overlapping constraints, whose main features are the geometric decomposition of the feasible regions defined by the pairwise NFP between pieces and the use of mutually exclusive binary variables to select their relative placements, such as the family of continuous models pioneered by Li (1994, §8), Dean (2002, §5), and Fischetti & Luzzi (2009), which are subsequently refined by Alvarez-Valdes et al. (2013), Cherri et al. (2016b), and Rodrigues et al. (2017). Second, the family of Constraint Programming (CP) models based on the NFP (B), whose pioneering work is introduced by Ribeiro, Carravilla, & Oliveira (1999), and subsequently improved by Carravilla, Ribeiro, & Oliveira (2003), Ribeiro & Carravilla (2004), Carravilla & Ribeiro (2005), Ribeiro & Carravilla (2009), and Cherri, Carravilla, Ribeiro, & Toledo (2019). And third, the family of Non-Linear Programming (NLP) models based on alternative geometric representations (C), such as the family of models based on  $\Phi$ -functions (C.1), whose pioneering works are introduced by Stoyan, Novozhilova, & Kartashov (1996), Chernov, Stoyan, & Romanova (2010), Chernov, Stoyan, Romanova, & Pankratov (2012), Stoyan & Romanova (2013), Stoyan, Romanova, Pankratov, & Chugay (2015), Stoyan, Pankratov, & Romanova (2016a), Stoyan, Pankratov, & Romanova (2016b), Stoyan, Pankratov, & Romanova (2017), Romanova, Stoyan, Pankratov, Litvinchev, & Marmolejo (2020), Pankratov, Romanova, Shekhovtsov, Grebennik, & Pankratova (2020), and Stoyan, Zlotnik, & Chugay (2012); (C.2) other non-linear models based on direct trigonometry introduced by Kallrath (2009), Cherri, Cherri, & Soler (2018b), and Peralta, Andretta, & Oliveira (2018); and finally, (C.3) the family of models based on circle coverings allowing free rotations, whose pioneering work is introduced by Jones (2014) and subsequently refined by Rocha, Rodrigues, Gomes, Toledo, & Andretta (2014), Rocha, Gomes, Rodrigues, Toledo, & Andretta (2016), and Wang, Hanselman, & Gounaris (2018).

Likewise, the linear MIP models based on the NFP (A.1) can be divided into three subfamilies according to the nature of the

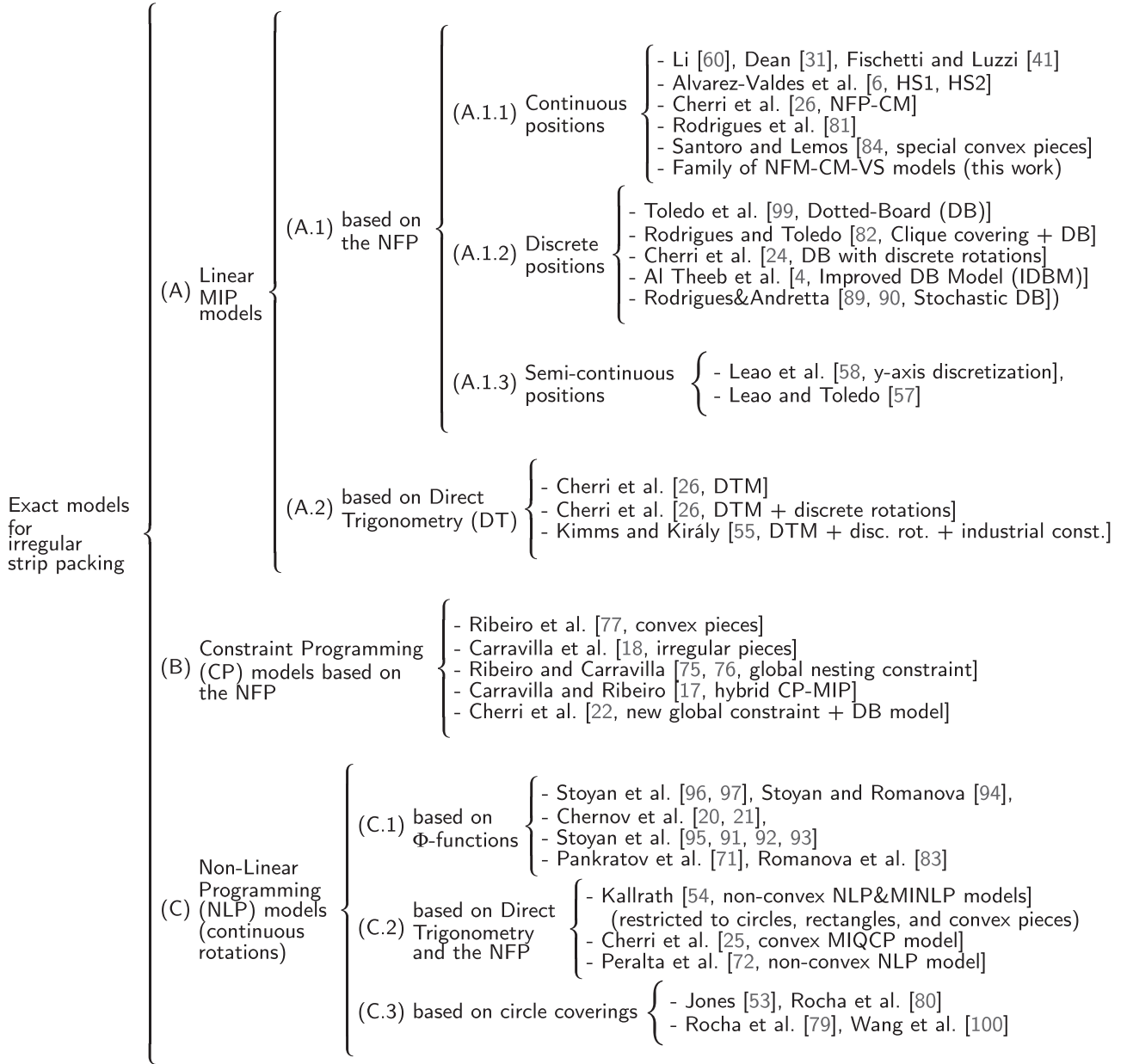


Fig. 1. Categorization of exact mathematical models for irregular strip packing.

decision variables encoding the  $(x,y)$  position of the pieces as follows: (A.1.1) continuous models, such as the former aforementioned ones (Alvarez-Valdes et al., 2013; Cherri et al., 2016b; Dean, 2002; Fischetti & Luzzi, 2009; Li, 1994; Rodrigues et al., 2017) and this work; (A.1.2) discrete models like the pioneering Dotted-Board model of Toledo, Carravilla, Ribeiro, Oliveira, & Gomes (2013), which subsequently refine (Rodrigues & Toledo, 2017) by using cliques to factorize multiple constraints; and (A.1.3) semi-continuous models such as the one proposed by Leao, Toledo, Oliveira, & Carravilla (2016) and subsequently improved by Leao & Toledo (2021).

### 2.3. Linear continuous MIP models

Regardless of the minor differences in the formulation of the continuous MIP models mentioned above, their main difference is the geometric decomposition of the feasible space of relative placements between pieces used to build the non-overlapping con-

straints, as shown in Fig. 2. Fischetti & Luzzi (2009) propose a convex decomposition of the feasible regions using the overall  $NFP_{ij}$  between pieces  $P_i$  and  $P_j$ , as shown in Fig. 2a. However, Fischetti & Luzzi do not provide a detailed definition of its geometry, which encourages Alvarez-Valdes et al. (2013) to propose a well-defined convex decomposition based on horizontal slices to build their HS2 model, as shown in Fig. 2b. Subsequently, Cherri et al. (2016b) propose a convex decomposition of the pieces and the definition of their non-overlapping constraints by using the convex no-fit polygons among their convex parts, as shown in Fig. 2c. The convex decomposition proposed by Cherri et al. for their NFP-CM model provides two key advantages on previous MIP models as follows: (1) it avoids the need explicitly to compute the overall NFP between irregular pieces by computing the NFP between convex parts, which is much easier than the former; and, (2) it allows non-overlapping constraints to be built between pieces using only one constraint per binary variable, unlike the previous MIP models that require at least three constraints per binary variable. However, one signif-

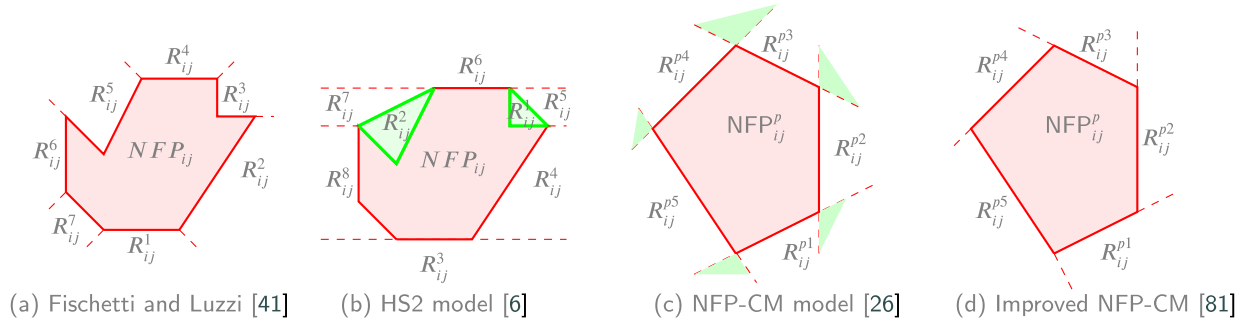


**Fig. 2.** Disjoint convex decomposition of the feasible relative placements between pieces defined by each model in the family of continuous MIP models for irregular strip packing. Every single convex feasible sub-region  $R_{ij}^k \subset \mathbb{R}^2$  in figures (a)–(d) above is enabled by a mutually-exclusive binary variable  $v_{ij}^k$ . The Fischetti & Luzzi (a) and HS2 (b) models set their non-overlapping constraints using the overall no-fit polygon  $NFP_{ij}$  between pieces  $P_i$  and  $P_j$ , whilst the NFP-CP models (c–d) set them using the convex NFP parts between each pair of convex parts of two pieces. Figure (c) shows in green the space of symmetric solutions generated by adjacent feasible sub-regions of the NFP-CM model (Cherri et al., 2016b), whilst figure (d) shows the symmetry-breaking of the former model proposed by the Improved NFP-CM model (Rodrigues et al., 2017). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

icant drawback of the NFP-CM model (Cherri et al., 2016b) is the existence of symmetric solutions induced by the overlapping of adjacent feasible sub-regions, as shown by regions in green in Fig. 2c, which encourages (Rodrigues et al., 2017) to break these symmetries by inserting one additional constraint per binary variable activating a separation line defined by the previous edge in the polygon's boundary, as shown in Fig. 2d, at the cost of doubling the number of non-overlapping constraints to break the symmetries in their geometric decomposition.

The main limitation of the family of current continuous MIP models based on the NFP approach is their inability to solve either large problem instances or instances including pieces with complex geometries, which we attribute to two difficulties derived from their structure as follows: (1) the large number of binary variables needed to build the non-overlapping constraints between pieces; and (2) a poor tightening of the LP relaxed model as a consequence of encoding the linearization of a min-max model in the objective function that does not directly depend on the binary variables. For instance, the number of binary variables in current continuous exact MIP models (Alvarez-Valdes et al., 2013; Cherri et al., 2016b; Fischetti & Luzzi, 2009; Rodrigues et al., 2017) is quadratic regarding the number of pieces  $n$  with scalability factor  $O(r^4 \frac{n}{2}(n-1))$ , whilst the number of non-overlapping constraints might be up to three times the latter factor, where  $r$  is the average number of edges per piece. The  $\frac{1}{2}n(n-1)$  factor is derived from the pairwise combinatory nature of the problem, whilst the  $r^4$  gives account of the geometric complexity of the pieces, which derives from the fact that the resulting NFP from two polygons with  $s$  and  $t$  edges might have  $O(s^2t^2)$  edges in the worst case (Agarwal, Flato, & Halperin, 2002, p.40), but it is at most  $O(s+t)$  for convex polygons. Thus, the complexity of current continuous MIP models grows rapidly with respect to both the number of pieces and their geometric complexity.

#### 2.4. Discrete and semi-continuous MIP models

Several authors have proposed discrete MIP models to solve large nesting instances efficiently to overcome the performance limitations of the family of continuous MIP models, as follows: (1) the discrete models based on a grid representation of the board introduced by Toledo et al. (2013), which is refined by Rodrigues & Toledo (2017) using a clique-based formulation, being subsequently extended by Rodrigues de Souza & Andretta (2020, 2022) to deal with the uncertainty in the demand of the pieces using a two-stage stochastic programming model; and finally, (2) the semi-continuous model based on the discretization of the  $y$ -axis introduced by Leao et al. (2016). The Dotted-Board model (DB) model

(Toledo et al., 2013) is able to solve problem instances with up to 56 pieces of two types within 2 hours (Toledo et al., 2013, BLAZP2P4\_28\_28, Table 3), whilst the semi-continuous MIP model of Leao et al. (2016) removes the resolution error in the  $x$ -axis direction and allows solving instances with up to 70 pieces (Leao et al., 2016, Table 1) and larger boards than the DB model. However, the discrete models above are approximations of the exact continuous problem. Thus, the quality of their solutions depends on the grid resolution, as argued by Toledo et al. (2013) and shown experimentally by Sato, de Sales Guerra Tsuzuki, de Castro Martins, & Gomes (2016b). Moreover, the computational cost of the discrete models also grows rapidly with the resolution of the grid and the number of different piece types, which encouraged several improvements to the DB model as follows. Rodrigues & Toledo (2017) propose a new clique-based formulation of the DB model to reduce the number of constraints, which achieves solving instances with up to 25 pieces of 22 piece types within 1 hour (Rodrigues & Toledo, 2017, Jakobs1, Table 3). Subsequently, Cherri et al. (2018a) propose two methods to build non-regular grids together with an efficient data structure to represent them. The grid representation, also called no-fit raster, is also used by recent state-of-the-art heuristics methods (Mundim, Andretta, Carravilla, & Oliveira, 2018; Mundim et al., 2017). More recently, Al Theeb, Hayajneh, & Jaradat (2021) introduce the Improved DB model, which requires significantly fewer variables than the original DB model formulation.

### 3. The new family of MIP models for nesting

This section introduces two new MIP models for irregular strip packing based on a new disjoint convex decomposition of the space of feasible placements between pieces defined by the pairwise convex NFP between their convex parts, together with a set of valid inequalities, symmetry breakings, and variable eliminations derived from the former decomposition. The use of the convex NFP between convex parts of the pieces to build the pairwise non-overlapping constraints is the core innovation of the family of NFP-CM models proposed by Cherri et al. (2016b) and subsequently improved by Rodrigues et al. (2017) to break the symmetries of the NFP-CM model, as shown in Fig. 2d. Thus, our new MIP models set a more tightened formulation than the NFP-CM model for the same feasible space of relative placements between pieces by defining a new set of constraints and two new binary variables based on our new disjoint convex decomposition shown in Fig. 3c. Our new NFP-CM-VS models include at most 2 binary variables more per convex NFP part than the NFP-CM model (Cherri et al., 2016b) to encode the left  $R_{ij}^{fgl}$  and right  $R_{ij}^{fgr}$  feasible sub-regions

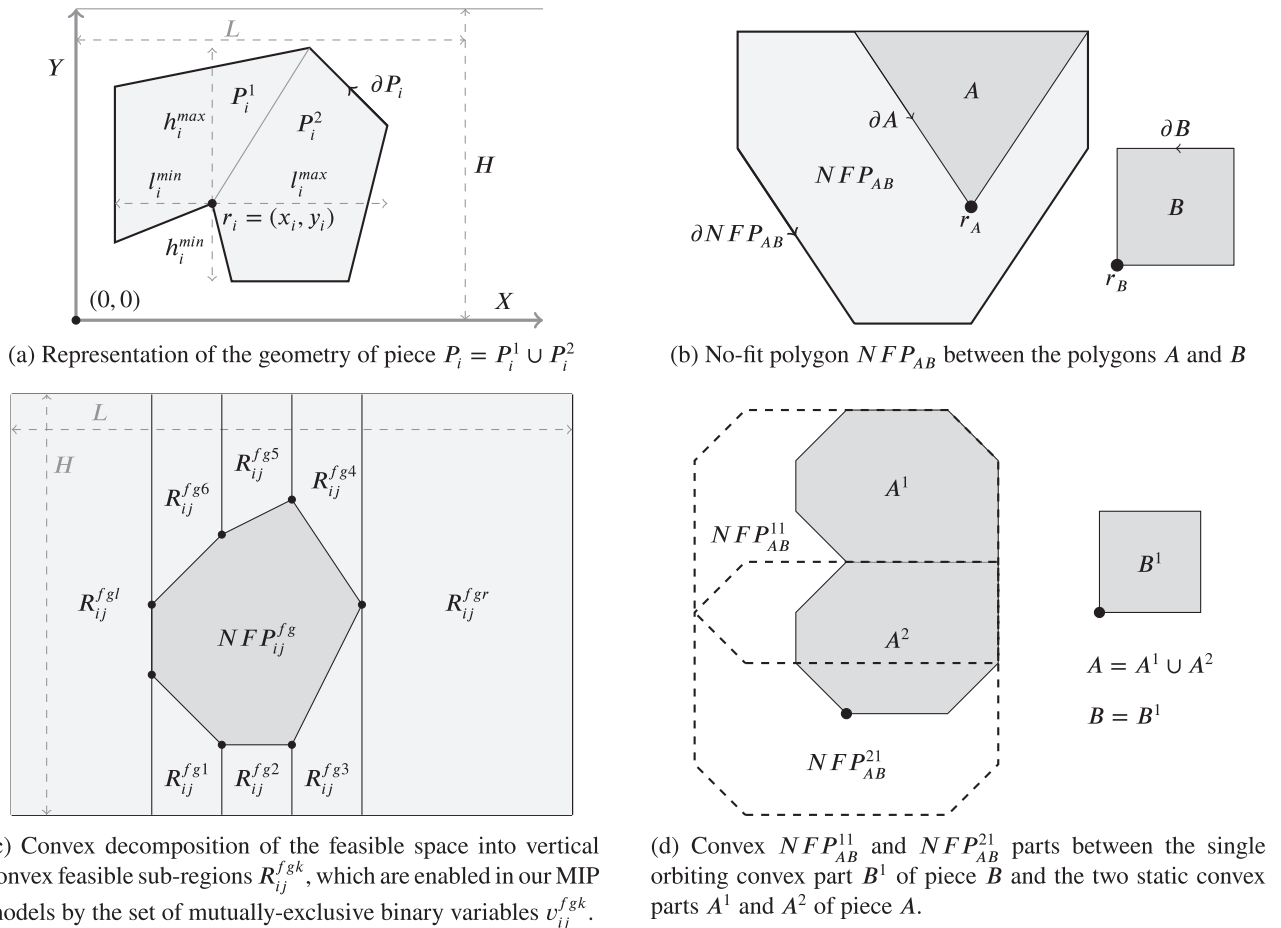


**Fig. 3.** Figure (a) shows the geometric representation of the pieces and the board, such that non-convex pieces are decomposed into convex parts. Figure (b) shows the no-fit polygon  $NFP_{AB}$  between the static piece  $A$  and the orbiting piece  $B$ . Figure (c) shows our new disjoint convex decomposition into vertical slices. Figure (d) shows the convex no-fit polygon parts  $NFP_{AB}^{11}$  and  $NFP_{AB}^{21}$  derived from the convex orbiting piece  $B$  and a non-convex static piece  $A$ .

shown in Fig. 3c in those cases in which the NFP's boundary does not include a vertical edge in some of their sides.

### 3.1. Rationale of our new geometric decomposition

The two aims of our new convex decomposition are as follows: (1) to break the symmetries of the feasible solutions; and (2) to infer a large number of infeasible relative placements among three pieces or two pieces of the same type, which can be hand-coded either as feasibility cuts or by removing some binary variables from the model, respectively. Our new feasibility cuts among three pieces can also be generalized to combinations of more than three pieces; however, we limit our evaluation here to piece triplets. First, we introduce the basic formulation of our new MIP model called No-Fit Polygon Covering Model based on Vertical Slices (NFP-CM-VS). Next, we introduce several families of valid inequalities and variable eliminations to break the symmetries and remove some infeasible solutions derived from the convex NFP parts between two pieces, as well as a family of feasibility cuts among three pieces. Finally, we introduce the NFP-CM-VS2 model integrating all cuts and variable eliminations of the full NFP-CM-VS model, but partially removes the big-M terms for the constraints encoding the vertical slices of our convex decomposition by factorizing all  $x$ -axis constraints into two single constraints per convex NFP part.

We have preferred to formulate our MIP models using a vertical decomposition, instead of adapting the HS2 horizontal decomposition (Alvarez-Valdes et al., 2013) to the convex NFP parts between pieces because, in many relevant industrial applications,

the length of the board  $L$  is commonly greater than its height  $H$ . Thus, we conjecture that the possibility of finding infeasible relative placements using our algorithms based on interval arithmetic among three or more pieces grows in the horizontal direction in the former scenario. However, our vertical slices are not the critical feature towards obtaining better results; rather than they are a tool together with the definition of convex NFP parts among pieces (Cherri et al., 2016b), to achieve it by deriving a new set of symmetry-breakings, variable reductions, valid cuts, and formulations which are the key features for improving the performance of our new models regarding the state-of-the-art models evaluated here. Because of the lack of space, we do not provide an equivalent formulation of our NFP-CM-VS models for those cases in which  $H \gg L$ ; but it might be done by using an equivalent horizontal decomposition and symmetrizing all constraints, as well as logic and geometric relationships defining the core contributions of our models.

### 3.2. Basic notation and geometric tools

**Representation of the pieces.** Fig. 3a details the meaning of the parameters used to represent the geometry of the pieces and the board. The board is represented by an open rectangle of fixed height  $H$  and variable length  $L$ , whose lower and upper bounds are denoted by  $L_{lb}$  and  $L_{ub}$ , respectively. We use a standard orthogonal  $x$ - $y$  reference frame setting the origin of the board in its bottom-leftmost corner. Each piece  $P_i$  can be either a regular polygon defined by a single convex part or an irregular polygon decom-

posed into a set of disjoint convex polygons, called convex parts, whose boundaries are represented by a counter-clockwise ordered list of points in the plane, called *vertices*, such that each pair of consecutive vertices defines a line segment, called an *edge*. We use the indexes  $f$  and  $g$  to denote the source convex parts generating the convex  $NFP_{ij}^{fg}$  part between the pieces  $P_i$  and  $P_j$ , such that  $P_i^f$  denotes the static convex part from  $P_i$  and  $P_j^g$  the orbiting convex part from  $P_j$  that generate the convex no-fit polygon  $NFP_{ij}^{fg}$ . The same index tuple  $(i, j, f, g)$  is extensively used to denote different types of objects defined by pairwise combinations of convex parts from pieces  $P_i$  and  $P_j$ , such as binary variables  $v_{ij}^{fgk}$  and their corresponding feasible geometric regions  $R_{ij}^{fgk}$ . For each piece  $P_i$ , either one arbitrary vertex from any of its convex parts or its barycenter (our preferred option) could be selected as a reference point, denoted by  $r_i = (x_i, y_i)$ , to set the continuous decision variables representing the position of  $P_i$  on the board. The parameters  $l_i^{\min}$  and  $l_i^{\max}$  denote the  $x$ -axis distance from the leftmost and rightmost points of  $P_i$  to its reference point  $r_i$ , respectively, whilst  $h_i^{\min}$  and  $h_i^{\max}$  denote the  $y$ -axis distance from the bottommost and topmost points of piece  $P_i$  to  $r_i$ , respectively. The set of indexes for the type of pieces is  $PT = \{1, \dots, m\}$ , and the type of each piece  $P_i$  is denoted by  $t_i \in PT$ , whilst the area of each type of piece  $t \in PT$  is denoted by  $\Delta_t$  and its demand by  $d_t$ .

**Convex decomposition of pieces.** To build the non-overlapping constraints of our family of NFP-CM-VS models, all non-convex pieces must be decomposed into convex polygons, as shown in Fig. 3a. Thus, each piece  $P_i$  is defined as the union of  $Q_i$  convex parts as follows  $P_i = \bigcup_{f=1}^{Q_i} P_i^f$ . We use Greene's convex decomposition algorithm (Greene (1983)) implemented by the CGAL library to decompose the pieces into convex parts as proposed by Cherri et al. (2016b), with the aim of replicating their NFP-CM models and experiments. Greene's algorithm efficiently approximates the optimal convex decomposition of a polygon without adding new vertices to its boundary. However, there are many other alternatives in the literature (Fernández, Cánovas, & Pelegrín (2000)) that could potentially generate fewer edges, and thus fewer binary variables, than Greene's algorithm. For instance, we found in many problem instances that our implementation of a variant of the Angle-Bisector (AB) convex decomposition method of Agarwal et al. (2002) generated models with fewer binary variables than those built using Greene's method. Thus, any practical implementation of our models should consider both the impact of the convex decomposition methods on the size of the resulting MIP models and their computational cost.

**Computation of convex NFP parts.** Fig. 3b shows in light grey the no-fit polygon  $NFP_{AB}$  between two convex polygons  $A$  and  $B$  with reference points  $r_A$  and  $r_B$ , respectively. The outer and boundary regions of  $NFP_{AB}$  define the feasible region in which polygon  $B$  can be placed without overlapping polygon  $A$ , whilst its inner region sets the non-feasible relative positions for placing  $B$ , and the boundary of  $NFP_{AB}$ , denoted by  $\partial NFP_{AB}$ , sets the positions in which both polygons are in contact. Given two convex polygons  $A$  and  $B$ , their no-fit polygon  $NFP_{AB} = A \oplus (-B(0, 0))$  is always convex (de Berg, Van Kreveld, Overmars, & Schwarzköpf, 1997, theorem 13.5) and it can be efficiently computed using any specialized algorithm for convex polygons, such as the orbiting method of Cuninghame-Green (1989), or a specialized version of Minkowski sums for convex polygons (de Berg et al., 1997, p.299). Although Cherri et al. (2016b) propose the use of Cuninghame-Green's algorithm to compute the NFP between convex polygons to build their NFP-CM model, our preferred option is to use convex Minkowski sums because once all convex parts of any orbiting piece  $B$  are translated to the origin by summing the  $-r_B$  vector, the relative

positions of all convex  $NFP_{AB}^{fg}$  parts are well defined with respect to the reference point  $r_A$  of the static piece, unlike the resulting NFP parts obtained with Cuninghame-Green's algorithm. Algorithm 1 introduces our detailed implementation of the MinkowskiSum algorithm (de Berg et al., 1997, p.299) for convex polygons used to compute the convex NFP between the convex parts of each pair of pieces.

Our preferred method to compute convex NFP parts. Algorithm 1 introduces the *ConvexMinkowskiSum* function used to compute the convex  $NFP_{AB}$  part between two convex parts from different pieces by exploiting the fact that  $NFP_{AB} = A \oplus (-B(0, 0))$ . *ConvexMinkowskiSum* function accepts two counter-clockwise oriented sequences of  $(x, y)$  coordinates encoding the boundary of the convex polygon  $\delta_a = \partial A$ , corresponding to the static convex part of piece  $A$  at its base position, and the boundary of the convex polygon  $\delta_b = -(\partial B \oplus -r_B)$ , corresponding to the reflection of the orbiting convex part of piece  $B$  translated to the origin  $(0, 0)$  by subtracting the position of the reference point  $r_B$ . *ConvexMinkowskiSum* function implicitly assumes that  $\delta_a$  and  $\delta_b$  define closed convex polygons. Thus, their first vertex should not be repeated at the end position. Both input sequences are sorted in steps 2–3 to place the bottommost vertex at the first position. *ConvexMinkowskiSum* function returns a counter-clockwise oriented  $(x, y)$  sequence representing the boundary of  $NFP_{AB}$  in the same Euclidean coordinates frame as the static polygon  $A$ .

**New convex decomposition based on vertical slices.** Fig. 3c shows our new disjoint convex decomposition of the feasible space of relative placements for the piece  $P_j$  with respect to the piece  $P_i$ , such that their corresponding convex parts  $P_j^g$  and  $P_i^f$  do not overlap. We recall that the non-overlapping constraints of our family of NFP-CM-VS models are defined by all pairwise combinations of convex parts between two distinct pieces.  $NFP_{ij}^{fg}$  denotes the convex no-fit polygon between the static convex part  $P_i^f$  of piece  $P_i$  and the orbiting convex part  $P_j^g$  of piece  $P_j$ . For example, Fig. 3d shows the  $NFP_{AB}^{11}$  and  $NFP_{AB}^{21}$  parts derived from the combinations of convex parts of pieces  $A$  and  $B$  as dashed lines. Likewise,  $\partial NFP_{ij}^{fg}$  denotes the boundary of  $NFP_{ij}^{fg}$  and it is defined by a counter-clockwise oriented polyline whose line segments are called *edges* and denoted by  $e_{ij}^{fgk} = (a_{ij}^{fgk}, b_{ij}^{fgk}) \in \mathbb{R}^2 \times \mathbb{R}^2$ , with end points  $a_{ij}^{fgk} = (a_{ij,x}^{fgk}, a_{ij,y}^{fgk})$  and  $b_{ij}^{fgk} = (b_{ij,x}^{fgk}, b_{ij,y}^{fgk})$ . Each convex feasible sub-region enabled in the solution of our MIP models by a binary variable  $v_{ij}^{fgk}$  is denoted by  $R_{ij}^{fgk} \subset \mathbb{R}^2$ .

The no-fit boundary  $\partial NFP_{ij}^{fg}$  is represented as the union of all its line segments, called edges, as defined in expression (5) below, being  $\kappa_{ij}^{fg}$  the overall number of boundary edges. To define the feasible sub-regions induced by  $NFP_{ij}^{fg}$ , we decompose the line segments  $e_{ij}^{fgk}$  of  $\partial NFP_{ij}^{fg}$  into three disjoint sets grouping the top, bottom, and side edges, as defined by the expressions (6)–(9) below.

$$\partial NFP_{ij}^{fg} = \bigcup_{k=1}^{\kappa_{ij}^{fg}} e_{ij}^{fgk}, \quad e_{ij}^{fgk} = (a_{ij}^{fgk}, b_{ij}^{fgk}), \quad a_{ij}^{fgk}, b_{ij}^{fgk} \in \mathbb{R}^2 \quad (5)$$

$$\partial NFP_{ij}^{fg} = \mathcal{T}_{ij}^{fg} \cup \mathcal{B}_{ij}^{fg} \cup \mathcal{S}_{ij}^{fg} \quad (\text{boundary's edge decomposition}) \quad (6)$$

$$\mathcal{T}_{ij}^{fg} = \{e_{ij}^{fgk} \in \partial NFP_{ij}^{fg} | a_{ij,x}^{fgk} > b_{ij,x}^{fgk}\} \quad (\text{top edges}) \quad (7)$$

$$\mathcal{B}_{ij}^{fg} = \{e_{ij}^{fgk} \in \partial NFP_{ij}^{fg} | a_{ij,x}^{fgk} < b_{ij,x}^{fgk}\} \quad (\text{bottom edges}) \quad (8)$$

**Algorithm 1** Our version of the MinkowskiSum( $\delta_a, \delta_b$ ) algorithm for convex polygons introduced by de Berg et al. (1997, p.299). ConvexMinkowskiSum function is used to compute the convex no-fit polygon  $NFP_{AB} = A \oplus (-B(0, 0))$  between convex parts  $A$  and  $B$  by calling the function below with parameters  $\delta_a = \partial A$  and  $\delta_b = -(\partial B \oplus -r_B)$ . The % operator in steps 10–12 below denotes the standard remainder operator available in any programming language.

**Input:** counter-clockwise oriented boundaries  $\delta_a \in \mathbb{R}^{2 \times n}, \delta_b \in \mathbb{R}^{2 \times m}$

**Output:** counter-clockwise boundary  $\delta_{NFP}$  of  $NFP_{AB}$

```

1: function CONVEXMINKOWSKI SUM( $\delta_a, \delta_b$ )
2:    $\delta'_a \leftarrow \text{sort}(\delta_a)$                                 ▷ sort function orders the (x,y) sequences such that
3:    $\delta'_b \leftarrow \text{sort}(\delta_b)$                                 ▷ its first vertex has the smallest y-axis coordinate
4:    $n \leftarrow \text{lengthof}(\delta'_a)$ 
5:    $m \leftarrow \text{lengthof}(\delta'_b)$ 
6:    $\delta_{NFP} \leftarrow \emptyset$                                     ▷  $\delta_{NFP}$  is an array of  $\mathbb{R}^2$  points
7:    $i \leftarrow 0$ 
8:    $j \leftarrow 0$ 
9:   while ( $i < n \vee j < m$ ) do
10:     $\delta_{NFP} \xleftarrow{\text{adds}} \delta'_a[i \% n] + \delta'_b[j \% m]$              ▷ adds a new vertex to  $\delta_{NFP}$ 
11:     $b \leftarrow \delta'_a[(i+1) \% n] - \delta'_a[i \% n]$ 
12:     $c \leftarrow \delta'_b[(j+1) \% m] - \delta'_b[j \% m]$ 
13:     $\theta \leftarrow b_x c_y - c_x b_y$ 
14:    if  $\theta \geq 0$  then
15:       $i \leftarrow i + 1$ 
16:    end if
17:    if  $\theta \leq 0$  then
18:       $j \leftarrow j + 1$ 
19:    end if
20:  end while
21:  return  $\delta_{NFP}$ 
22: end function

```

$$S_{ij}^{fg} = \{e_{ij}^{fgk} \in \partial NFP_{ij}^{fg} \mid a_{ij,x}^{fgk} = b_{ij,x}^{fgk}\} \quad (\text{side edges}) \quad (9)$$

For instance, the set of edges defining the top feasible regions  $\{R_{ij}^{fg4}, R_{ij}^{fg5}, R_{ij}^{fg6}\}$  of  $NFP_{ij}^{fg}$  shown in Fig. 3c is defined by  $T_{ij}^{fg} = \{e_{ij}^{fg4}, e_{ij}^{fg5}, e_{ij}^{fg6}\}$ , whilst the set of binary variables enabling them is  $\{v_{ij}^{fg4}, v_{ij}^{fg5}, v_{ij}^{fg6}\}$ . Likewise, the set of edges defining the bottom feasible regions  $\{R_{ij}^{fg1}, R_{ij}^{fg2}, R_{ij}^{fg3}\}$  of  $NFP_{ij}^{fg}$  shown in Fig. 3c is defined by  $B_{ij}^{fg} = \{e_{ij}^{fg1}, e_{ij}^{fg2}, e_{ij}^{fg3}\}$ , whilst the set of binary variables enabling them is  $\{v_{ij}^{fg1}, v_{ij}^{fg2}, v_{ij}^{fg3}\}$ . Finally, side edges  $S_{ij}^{fg}$  will not be considered in our model, because the feasible regions defined by vertical edges, either on the left or right sides of  $NFP_{ij}^{fg}$ , are enabled by the special binary variables  $v_{ij}^{fgl}$  and  $v_{ij}^{fgr}$ , as shown in Fig. 3c. Finally, we introduce a notation for the smallest and largest x-axis coordinates of each convex  $NFP_{ij}^{fg}$  part, as detailed in Eqs. (10) and (11) below.

$$\underline{x}_{ij}^{fg} = \min\{\partial NFP_{ij,x}^{fg}\} \quad (\text{smallest x-axis coordinate of the convex NFP part}) \quad (10)$$

$$\bar{x}_{ij}^{fg} = \max\{\partial NFP_{ij,x}^{fg}\} \quad (\text{largest x-axis coordinate of the convex NFP part}) \quad (11)$$

**Definition of index sets.** Before detailing our models, we define the index set  $I_{ij}^{fg}$  detailed below (12) to simplify our notation and the presentation of our models. Each tuple  $(i, j, f, g) \in I_{ij}^{fg}$  is used to denote the indexes involved in the definition of the binary variables or constraints concerning a convex  $NFP_{ij}^{fg}$  part obtained by evaluating the Algorithm 1 with the convex parts  $P_i^f$  and  $P_j^g$  of pieces  $P_i$  and  $P_j$  as input, where  $Q_i$  and  $Q_j$  are the number of convex parts of the former pieces, respectively. On the other hand,  $T_{ij}^{fg}$  and  $B_{ij}^{fg}$  denote the index set of the binary variables encoding

the top and bottom convex feasible sub-regions defined by  $NFP_{ij}^{fg}$ , whilst  $K_{ij}^{fg}$  denotes the index set of all binary variables encoding the convex feasible sub-regions shown in Fig. 3c, as detailed below.

$$I_{ij}^{fg} = \{\alpha \in \{1, \dots, N\} \times \{1, \dots, N\} \times \{1, \dots, Q_i\} \times \{1, \dots, Q_j\} \mid 1 \leq i < j \leq N\} \quad (12)$$

$$T_{ij}^{fg} = \{k \in \{1, \dots, K_{ij}^{fg}\} \mid e_{ij}^{fgk} \in T_{ij}^{fg}\} \quad (\text{indexes of variables encoding top feasible sub-regions}) \quad (13)$$

$$B_{ij}^{fg} = \{k \in \{1, \dots, K_{ij}^{fg}\} \mid e_{ij}^{fgk} \in B_{ij}^{fg}\} \quad (\text{indexes of variables encoding bottom feasible sub-regions}) \quad (14)$$

$$K_{ij}^{fg} = \{k \in T_{ij}^{fg} \cup B_{ij}^{fg} \cup \{v_{ij}^{fgl}, v_{ij}^{fgr}\}\} \quad (\text{indexes of variables encoding all feasible sub-regions}) \quad (15)$$

Most of the constraints of our models are based on one convex NFP part resulting from the combination of two convex parts from two different pieces, whose definition requires a tuple with five indexes from the Cartesian product of  $I_{ij}^{fg}$  with any of the index sets of binary variables in definitions (13)–(15). However, we also introduce here for the first time several constraints among three pieces that demand up to nine indexes for their definition. Thus, we define in (16) the index set  $II_{iju}^{fgh}$  for the convex parts  $\{P_i^f, P_j^g, P_u^h\}$  from three different pieces denoted by  $P_i, P_j$ , and  $P_u$ .

$$II_{iju}^{fgh} = \{\alpha \in \{1, \dots, N\} \times \{1, \dots, N\} \times \{1, \dots, N\} \times \{1, \dots, Q_i\} \times \{1, \dots, Q_j\} \times \{1, \dots, Q_u\} \mid 1 \leq i < j < u \leq N\} \quad (16)$$

### 3.3. The NFP-CM-VS models

The basic NFP-CM-VS model is defined by the objective function (17) and the Constraints (18)–(32), whilst the full NFP-CM-VS model is defined by the former objective function and the constraints, symmetry breakings, valid inequalities, and variable eliminations in expressions (18)–(41). The objective function (17) together with the Constraints (18)–(23) fit the definition of the basic NFP-CM model without cuts Cherri et al. (2016b), with the only exception of our two special binary variables  $v_{ij}^{fgl}$  and  $v_{ij}^{fgr}$  encoding the left and right feasible sub-regions shown in Fig. 3c, whilst the Constraints (24)–(29) encode our new convex decomposition based on vertical slices, which removes all symmetric solutions derived from any relative placement between pieces.  $N$  and  $m$  denote the number of pieces and types of pieces, respectively.

$$\min \quad z = L \quad (17)$$

$$\text{s.t.} \quad l_i^{\min} \leq x_i \leq L - l_i^{\max}, \quad 1 \leq i \leq N \quad (18)$$

$$h_i^{\min} \leq y_i \leq H - h_i^{\max}, \quad 1 \leq i \leq N \quad (19)$$

$$L \leq \sum_{t=1}^m d_t (l_t^{\min} + l_t^{\max}) = L_{ub} \quad (20)$$

$$L_{lb} = \max\{\max_{1 \leq t \leq m} \{l_t^{\min} + l_t^{\max}\}, \frac{1}{H} \sum_{t=1}^m d_t \Delta_t\} \leq L \quad (21)$$

$$(b_{ij,x}^{fgk} - a_{ij,x}^{fgk})(y_j - y_i) + (a_{ij,y}^{fgk} - b_{ij,y}^{fgk})(x_j - x_i) + C_{ij}^{fgk} \leq (1 - v_{ij}^{fgk})M_{ij}^{fgk} \quad \forall (i, j, f, g, k) \in \{I_{ij}^{fg} \times T_{ij}^{fg} \cup B_{ij}^{fg}\} \quad (22)$$

$$\begin{aligned} C_{ij}^{fgk} &= b_{ij,y}^{fgk} a_{ij,x}^{fgk} - b_{ij,x}^{fgk} a_{ij,y}^{fgk} \\ M_{ij}^{fgk} &= |b_{ij,x}^{fgk} - a_{ij,x}^{fgk}|H + |a_{ij,y}^{fgk} - b_{ij,y}^{fgk}|L_{ub} + C_{ij}^{fgk} \\ v_{ij}^{fgl} + v_{ij}^{fgr} + \sum_{k \in T_{ij}^{fg} \cup B_{ij}^{fg}} v_{ij}^{fgk} &= 1, \quad \forall (i, j, f, g) \in I_{ij}^{fg} \end{aligned} \quad (23)$$

$$b_{ij,x}^{fgk} + x_i - x_j \leq (1 - v_{ij}^{fgk})M_{ij}^{fgk}, \quad \forall (i, j, f, g, k) \in \{I_{ij}^{fg} \times T_{ij}^{fg}\} \quad (24)$$

$$\begin{aligned} M_{ij}^{fgk} &= b_{ij,x}^{fgk} + L_{ub} - l_i^{\max} - l_j^{\min} \\ x_j - x_i - a_{ij,x}^{fgk} &\leq (1 - v_{ij}^{fgk})M_{ij}^{fgk}, \quad \forall (i, j, f, g, k) \in \{I_{ij}^{fg} \times T_{ij}^{fg}\} \end{aligned} \quad (25)$$

$$\begin{aligned} \bar{M}_{ij}^{fgk} &= L_{ub} - l_j^{\max} - l_i^{\min} - a_{ij,x}^{fgk} \\ a_{ij,x}^{fgk} + x_i - x_j &\leq (1 - v_{ij}^{fgk})\bar{M}_{ij}^{fgk}, \quad \forall (i, j, f, g, k) \in \{I_{ij}^{fg} \times B_{ij}^{fg}\} \end{aligned} \quad (26)$$

$$\begin{aligned} \bar{M}_{ij}^{fgk} &= a_{ij,x}^{fgk} + L_{ub} - l_i^{\max} - l_j^{\min} \\ x_j - x_i - b_{ij,x}^{fgk} &\leq (1 - v_{ij}^{fgk})\bar{M}_{ij}^{fgk}, \quad \forall (i, j, f, g, k) \in \{I_{ij}^{fg} \times B_{ij}^{fg}\} \end{aligned} \quad (27)$$

$$\begin{aligned} \bar{M}_{ij}^{fgk} &= L_{ub} - l_j^{\max} - l_i^{\min} - b_{ij,x}^{fgk} \\ x_j - x_i - x_{ij}^{fg} &\leq (1 - v_{ij}^{fgl})M_{ij}^{fgl}, \quad \forall (i, j, f, g) \in I_{ij}^{fg} \end{aligned} \quad (28)$$

$$\begin{aligned} M_{ij}^{fgl} &= L_{ub} - l_j^{\max} - l_i^{\min} - x_{ij}^{fg} \\ x_j - x_i - x_{ij}^{fg} &\geq (1 - v_{ij}^{fgr})M_{ij}^{fgr}, \quad \forall (i, j, f, g) \in I_{ij}^{fg} \end{aligned} \quad (29)$$

$$\begin{aligned} M_{ij}^{fgr} &= l_j^{\min} + l_i^{\max} - L_{ub} - x_{ij}^{fg} \\ L &\in \mathbb{R}^+ \end{aligned} \quad (30)$$

$$(x_i, y_i) \in \mathbb{R}^2, \quad 1 \leq i \leq N \quad (31)$$

$$v_{ij}^{fgk} \in \{0, 1\}, \quad \forall (i, j, f, g, k) \in \{I_{ij}^{fg} \times K_{ij}^{fg}\} \quad (32)$$

Objective function (17) sets the minimization of the decision variable  $L$  encoding the length of the board. The  $L$  decision variable is defined by all current MIP models to linearize the objective function  $z = \min\{\max\{x_i : 1 \leq i \leq N\}\}$ . Constraints (18) and (19) set the lower and upper bounds for the reference point  $r_i = (x_i, y_i)$  of each piece to ensure that all pieces are inside the board. These two constraints encode the *Inner-Fit Polygon* (IFT) of each piece, which represents the feasible region of the board in which it can be placed. Constraint (20) sets the upper bound  $L_{ub}$  for  $L$  defined as the overall sum of the length of all pieces, whilst Constraint (21) sets the lower bound  $L_{lb}$  of  $L$ , which is defined as the largest value between the largest length of any piece and the overall sum of the areas of the pieces divided by the height ( $H$ ) of the board. The lower and upper bounds of  $L$  detailed above have also been used by most of the continuous MIP models reported in the literature (Alvarez-Valdes et al., 2013; Cherri et al., 2016b; Fischetti & Luzzi, 2009; Rodrigues et al., 2017). Although other upper and lower bounds for  $L$  have been studied, the bounds mentioned above have become the standard ones for the objective function of the irregular strip-packing problem because of the drawbacks of other alternatives. For instance, Alvarez-Valdes et al. (2013) propose a lower bound for  $L$  based on the approximation of the pieces by a set of inner rectangles and the solution of an integer program defining a 1-Contiguous Bin Packing Problem (1-CBPP) (Álvarez-Valdés et al., 2009). However, the authors finally discount the use of this lower bound because its computational cost is very high. Furthermore, Cherri et al. (2016b) point out, with regard to the possibility of using more tightened upper bounds, that “a tighter big-M formulation generally makes it hard to the solver to find good quality solutions at the beginning of the search”, a conclusion to which we also subscribe. Thus, this latter drawback and the computational cost of computing tighter lower bounds endorse using the lower and upper bounds introduced above as a practical solution.

Constraints (22) define the feasible region for any feasible relative placement of piece  $P_j$  regarding piece  $P_i$ , which is defined on the right side of each counter-clockwise oriented edge  $e_{ij}^{fgk} \in \partial NFP_{ij}^{fg}$  for all convex NFP parts between both former pieces. The definition of the Constraints (22) on the convex  $NFP_{ij}^{fg}$  parts between pieces is the core contribution of the NFP-CM model introduced by Cherri et al. (2016b), which has the advantage of requiring the minimum number of constraints to define the feasible space of relative placements between pieces among the current family of continuous MIP models. However, the feasible space generated by the Constraints (22) introduces a large number of symmetric solutions derived from the overlapping of feasible regions spanned by consecutive edges, as shown in Fig. 2c. To break these latter symmetries, Rodrigues et al. (2017) propose the convex decomposition of the feasible space shown in Fig. 2c, which doubles the number of non-overlapping constraints per binary variable of the NFP-CM model (Cherri et al., 2016b). Unlike the Im-

proved NFP-CM model of Rodrigues et al. (2017), our novel disjoint convex decomposition shown in Fig. 3c allows the NFP-CM-VS model to break the symmetries of the solution space at the expense of tripling the number of constraints per binary variable of the NFP-CM model (Cherri et al., 2016b). However, our new formulation allows the explicit derivation of many feasibility cuts and symmetry breaks linking the binary variables encoding the relative placements among multiple pieces, which provide a tighter formulation of the problem than the former state-of-the-art NFP-CM models (Cherri et al., 2016b; Rodrigues et al., 2017). On the other hand, the formulation of our NFP-CM-VS2 model introduced in Section 3.4 removes the aforementioned drawback on the number of constraints required by the symmetry-breaking of NFP-CM-VS, but it retains the remaining symmetry-breakings, variable reductions, and valid cuts defined by expressions (33)–(41).

Constraints (23) encode the selection in any feasible solution of a single feasible sub-region for the relative placement between two pieces, as defined by our convex decomposition of each convex  $NFP_{ij}^{fg}$  part shown in Fig. 3c. Constraints (24)–(29) encode the vertical lines delimiting our convex feasible sub-regions, as shown in Fig. 3c. Our model defines the same binary variables for the edges of  $\partial NFP_{ij}^{fg}$  than the NFP-CM (Cherri et al., 2016b) and Improved NFP-CM (Rodrigues et al., 2017) models, with the only exception of the two special binary variables  $v_{ij}^{fgl}$  and  $v_{ij}^{fgr}$  encoding the left and right sub-regions, as shown in Fig. 3c. Finally, Constraints (30)–(32) set the domains for the decision variables.

### 3.3.1. Symmetry-breaking and variable eliminations for identical pieces

*Symmetry-breaking of piece permutations.* It is a well-known fact that any permutation of identical pieces generates the same overall feasible space and optimal solutions, which leads the Branch and Bound (B&B) algorithm to obtain many symmetrical solutions. To break the aforementioned symmetries, we can impose that either  $x_i \leq x_j$  for all  $1 \leq i < j \leq N$  if pieces  $P_i$  and  $P_j$  are of the same type, as proposed by Alvarez-Valdes et al. (2013, §4.5) and Cherri et al. (2016b, §3.1), or  $y_i \leq y_j$  as proposed by Rodrigues et al. (2017, ineq.15). Our family of NFP-CM-VS models includes the y-axis symmetry-breaking proposed for the Improved NFP-CM model (Rodrigues et al., 2017), as defined by constraints (33) below. However, the satisfaction of the Constraints (33) joined to our new convex decomposition shown in Fig. 3c induces several a priori infeasible relative placements between identical pieces that can be coded into the models as a set of new variable eliminations and valid inequalities, as detailed by Constraints (34) and (35) below.

$$y_i \leq y_j, \quad 1 \leq i < j \leq N, j = \min\{k = i + 1, \dots, N \mid t_i = t_j\} \quad (33)$$

*Variable eliminations for identical pieces.* Because Constraints (33) must be satisfied for identical pieces, all binary variables  $v_{ij}^{fgk}$  enabling the bottom regions of any  $NFP_{ij}^{fg}$  between two identical pieces must be equal to 0 whenever pieces  $P_i$  and  $P_j$  are of the same type (i.e.  $t_i = t_j$ ) and the bottom region is below the reference point  $r_i$ . For this reason, all binary variables  $v_{ij}^{fgk}$  encoding bottom feasible sub-regions should be fixed to 0 and removed from any NFP-CM-VS model, as detailed by Constraints (34) below. Because the origin (0,0) of all convex  $NFP_{ij}^{fg}$  parts is defined at the reference point  $r_i$ , a bottom feasible region  $R_{ij}^{fgk}$  will be below  $r_i$  if the y-coordinates of the extreme points of its associate edge  $e_{ij}^{fgk}$  are negative, as detailed by equality (34).

$$v_{ij}^{fgk} = 0 \quad \forall (i, j, f, g, k) \in \{I_{ij}^{fg} \times B_{ij}^{fg} \mid t_i = t_j \wedge (a_{if,y}^{fg} < 0) \wedge (b_{ij,y}^{fg} < 0), 1 \leq i < j \leq N\} \quad (34)$$

*Valid inequality for identical pieces.* For the same reasons argued in the paragraph above to introduce the Constraints (34), given two orbiting identical pieces  $P_j$  and  $P_u$  and another static distinct piece  $P_i$  with index lower than the two former pieces,  $P_u$  cannot be placed below  $P_j$  if  $j < u$  and  $P_j$  is placed on top of  $P_i$  to satisfy Constraints (33). Thus, the Constraints (35) below must be satisfied by all feasible solutions of the NFP-CM-VS model.

$$\sum_{k \in I_{ij}^{fg}} v_{ij}^{fgk} + \sum_{k' \in B_{iu}^{fh}} v_{iu}^{fhk'} \leq 1, \quad \forall (i, j, u, f, g, h) \in \{I_{iju}^{fgh} \mid t_j = t_u, 1 \leq i < j < u \leq N\} \quad (35)$$

### 3.3.2. Valid cuts and variable reductions between two pieces

Given two non-convex pieces  $P_i$  and  $P_j$ , their overall  $NFP_{ij}$  is decomposed into a collection of convex  $NFP_{ij}^{fg}$  parts obtained from the combination of the convex parts from both pieces, as shown in Fig. 4b. For example, Fig. 4a shows a convex piece  $B$  and a non-convex piece  $A$  decomposed into three convex parts, whilst Fig. 4b shows the three pairwise  $NFP_{AB}^{fg}$  parts resulting from the combination of the convex parts of both former pieces, denoted by  $NFP_{AB}^{11}$ ,  $NFP_{AB}^{21}$ , and  $NFP_{AB}^{31}$ . Each convex  $NFP_{AB}^{fg}$  part in the example shown in Fig. 4 sets a collection of feasible sub-regions for the relative placement of piece  $B$  regarding piece  $A$ . However, any feasible relative placement of the orbiting piece  $B$  regarding piece  $A$  in our MIP models must be in a common feasible region for all  $NFP_{AB}^{fg}$  parts between both pieces. Thus, given two or more feasible sub-regions belonging to two or more different convex  $NFP_{AB}^{fg}$  parts of  $NFP_{AB}$ , we identify three a priori geometric relationships between them inducing the set of logic relationships enumerated below, which allow the formulation of our family of NFP-CM-VS models to be tightened.

- (1) *Identical feasible sub-regions.* If feasible sub-regions of different convex NFP parts between two pieces represent the same feasible region for their relative placement, then we can eliminate all redundant binary variables and represent all identical regions by a single binary variable in the model. According to our convex decomposition in Fig. 2, only the left and right feasible regions  $R_{ij}^{fgl}$  and  $R_{ij}^{fgr}$  encoded by the special binary variables  $v_{ij}^{fgl}$  and  $v_{ij}^{fgr}$  in different convex  $NFP_{ij}^{fg}$  parts can be identical. For example, Fig. 4c shows that  $v_{AB}^{11l}$ ,  $v_{AB}^{21l}$ , and  $v_{AB}^{31l}$  encode the same feasible region on the left, whilst  $v_{AB}^{11r}$ ,  $v_{AB}^{21r}$ , and  $v_{AB}^{31r}$  encode the same feasible region on the right. Thus, we can set a single variable to enable the left and right feasible sub-regions and remove the remaining ones during the building of the model. These variable eliminations are defined by the equalities (36) and (37) below.

$$v_{ij}^{fgl} = v_{ij}^{f'g'l}, \quad \forall (i, j, f, g, f', g') : (f, g) \neq (f', g') \wedge x_{ij}^{f'g'} = x_{ij}^{fg} \quad (36)$$

$$v_{ij}^{fgr} = v_{ij}^{f'g'r}, \quad \forall (i, j, f, g, f', g') : (f, g) \neq (f', g') \wedge \bar{x}_{ij}^{f'g'} = \bar{x}_{ij}^{fg} \quad (37)$$

- (2) *Subsumed feasible sub-regions.* Fig. 4d shows an example with two feasible sub-regions  $R_{AB}^{312}$  and  $R_{AB}^{313}$  respectively, which are subsumed by the feasible sub-region  $R_{AB}^{112}$  belonging to other convex NFP part between the same pieces, such that  $R_{AB}^{312} \subset R_{AB}^{112}$  and  $R_{AB}^{313} \subset R_{AB}^{112}$ . Thus, these subsumption relationships between feasible sub-regions set a logic implication between their corresponding binary variables, such that any feasible solution of the model that sets to 1 a binary variable enabling a subsumed sub-region, must also set to 1 the binary variables enabling their subsumer regions from

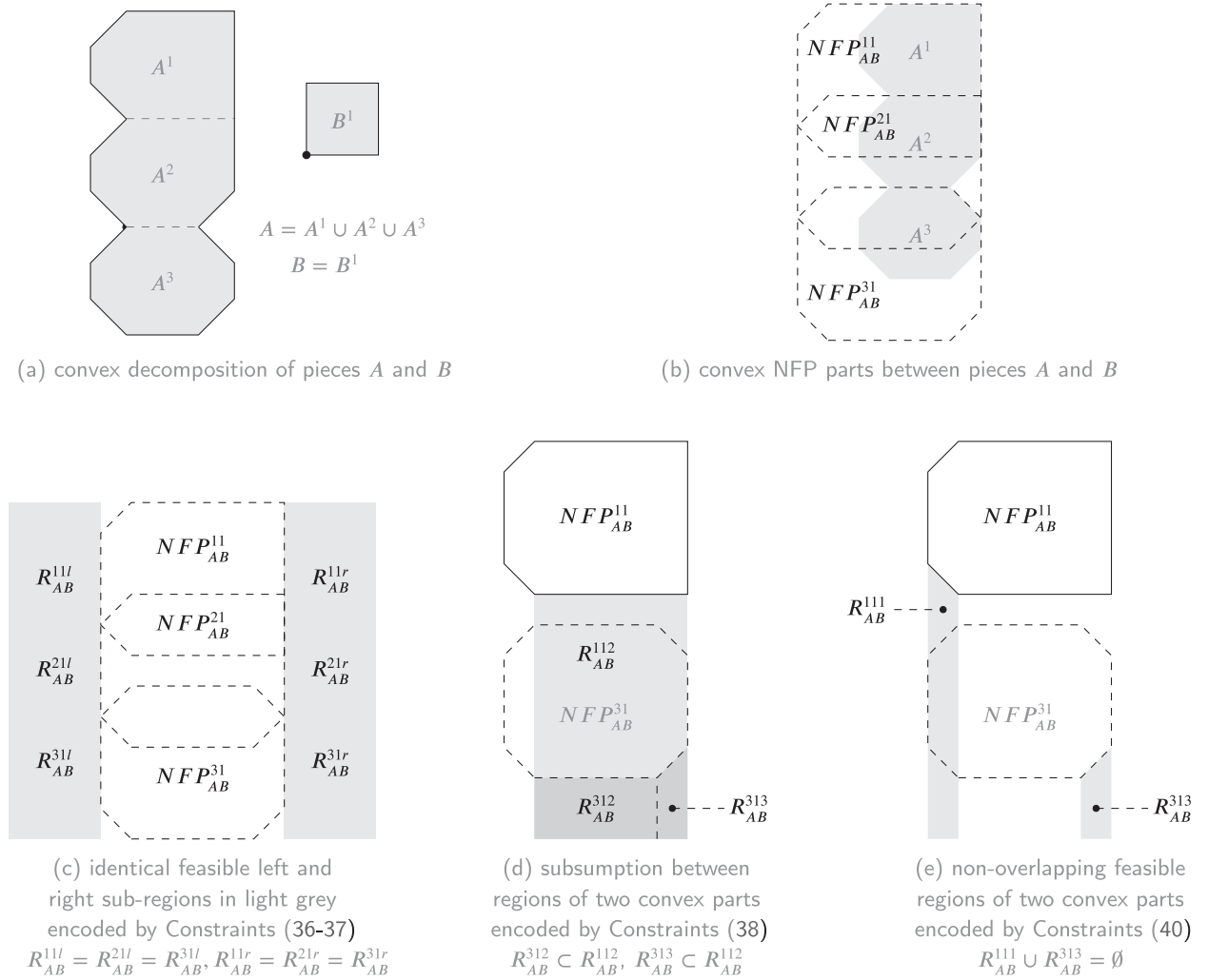


**Fig. 4.** This figure shows three geometric configurations between convex NFP parts of the pieces  $A$  and  $B$ , which induce three types of logic relationships encoded as valid cuts and variable eliminations into our NFP-CM-VS models. Every convex feasible sub-region  $R_{ij}^{fgk}$  in figures above is enabled by its corresponding binary variable  $v_{ij}^{fgk}$  in our MIP models.

other convex NFP parts between the same pieces. Thus, we can include the valid inequalities (38) in our models to force this type of logic relationship.

$$\sum_{k \in K_{ij}^{fg} \wedge (R_{ij}^{fgk} \subset R_{ij}^{f'g'k'})} v_{ij}^{fgk} \leq v_{ij}^{f'g'k'}, \quad \forall (i, j, f, g, f', g') : (f, g) \neq (f', g') \quad (38)$$

(3) *Non-overlapping regions.* If two feasible sub-regions belonging to two different convex NFP parts between two pieces do not overlap, then their corresponding binary variables cannot be enabled in a feasible solution at the same time because the resulting MIP model would be infeasible. For example, Fig. 4e shows that  $R_{AB}^{111} \cap R_{AB}^{313} = \emptyset$ , which implies that any feasible solution of the model must satisfy that  $v_{AB}^{111} + v_{AB}^{313} \leq 1$ . Thus, the former logic relationships can be represented in the model by including the valid inequalities (39).

$$v_{ij}^{fgk} + v_{ij}^{f'g'k'} \leq 1, \quad \forall (i, j, f, g, f', g', k, k') : R_{ij}^{fgk} \cap R_{ij}^{f'g'k'} = \emptyset \quad (39)$$

*Clique-based cuts.* The inequalities (39) above can be implemented in a much more efficient manner by defining them using the cliques of mutually-exclusive non-overlapping feasible sub-regions between convex NFP parts of two pieces, as defined by Constraints (40). The building of conflict graphs and clique cuts as proposed here is a well-known technique in integer programming (Atamtürk, Nemhauser, & Savelsbergh, 2000), which was successfully used by Rodrigues & Toledo (2017) to reformulate the DB model. Given an undirected graph  $G = (V, E)$ , a set of vertices  $C \subseteq V$  is a *clique* if  $\forall a, b \in C \Rightarrow \exists (a, b) \in E$ . The cliques allow factorizing many of the former constraints into a single one, which significantly reduces the number of constraints inserted into the model. Thus, we build an undirected graph  $G = (V, E)$  for representing the collection of Constraints (39), such that  $E$  contains an edge  $(v_{ij}^{fgk}, v_{ij}^{f'g'k'})$  for each pair of binary variables satisfying (39), and  $V$  contains all the binary variables that take part in at least one of the former constraints. Next, we compute the cliques of non-overlapping feasible sub-regions using the minimum Edge Clique Covering (ECC) algorithm (Conte, Grossi, & Marino, 2020) implemented by the ECC8 Java software library available at <https://github.com/Pronte/ECC>. Finally, the clique-based cuts (40) are inserted as SOS-type 1 constraints in the implemen-

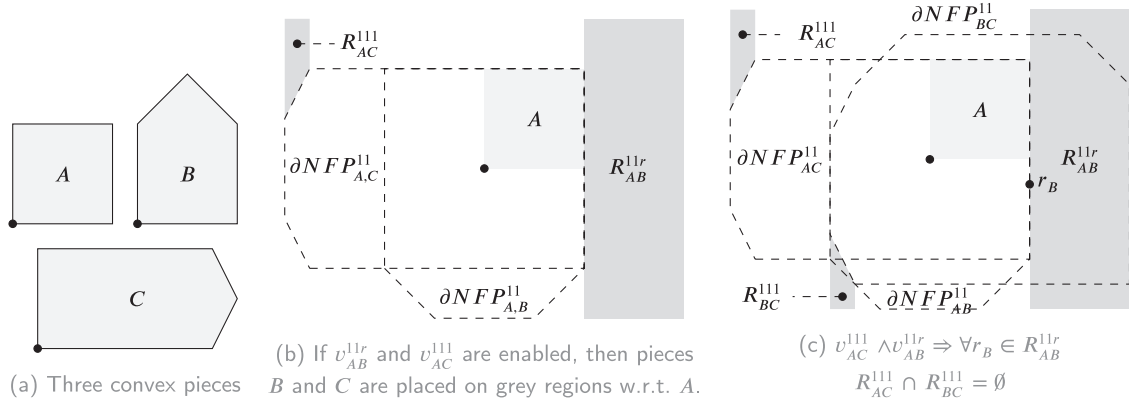


**Fig. 5.** Figure (a) shows an example of three convex pieces whose relative placements produce a large number of infeasible combinations of three binary variables when the regions  $R_{AC}^{111}$  and  $R_{AB}^{11r}$  shown in figure (b) are enabled in the solution. Figure (b) shows in grey the two feasible relative placements of pieces B and C regarding the static (pivot) piece A. Finally, figure (c) shows that if the binary variables  $v_{AC}^{11}$  and  $v_{AB}^{11r}$  are enabled, then  $\forall r_B \in R_{AB}^{11r}, R_{AC}^{111} \cap R_{BC}^{111} = \emptyset$ . Thus, the activation of the relative placements enabled by  $v_{AC}^{11}$  will produce an infeasible solution, as shown in figure (c). The joint activation of  $v_{AC}^{11}$  and  $v_{AB}^{11r}$  above causes that the all relative placements of piece C regarding piece B be infeasible, with the only exception of the left feasible sub-region enabled by  $R_{BC}^{111}$ . Every convex feasible sub-region  $R_{ij}^{fgk}$  in figure above is enabled by its corresponding binary variable  $v_{ij}^{fgk}$  in our MIP models. Algorithm 2 details how to compute all feasibility cuts derived from combining all convex parts from three pieces and encode them into our models with Constraints (41).

**Table 1**

Implementation details for all models evaluated herein. All models are built by ordering the pieces by non-increasing area to build the models, which are solved using Gurobi 9.5 with its default parameters on an UBUNTU 20.04 computer.

| MIP model                  | Reference               | Implementation details                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|----------------------------|-------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NFP-CM                     | Cherri et al. (2016b)   | Exact replication of the NFP-CM model (Cherri et al., 2016b) in which the non-overlapping feasibility cuts (Cherri et al., 2016b, Alg. 1, step 14) are inserted into the model as SOS-1 constraints for a fair comparison with our models.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| Improved NFP-CM (baseline) | Rodrigues et al. (2017) | Exact replication of the Improved NFP-CM model as detailed in Rodrigues et al. (2017, §3-4).                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| NFP-CM-VSnc                | this work               | Objective function (17) with Constraints (18)–(40). All constraints are inserted into the root node of the model, with the only exception of the Clique-based cuts (40) that are inserted as SOS-1 constraints.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| NFP-CM-VS                  | this work               | Same implementation than NFP-CM-VSnc plus the feasibility cuts among three pieces (41) inserted as user cuts (Lazy = -1) into the model. Although our preferred implementation is to insert the Constraints (41) as user cuts, any practitioner is free to integrate them in other ways. We recall that a user cut is only a valid inequality whose insertion into each B&B node is decided by the MIP solvers according to their heuristics, regardless of whether they are satisfied. Although Constraints (41) are enabled as user cuts in Gurobi by setting their “Lazy attribute” to -1, they should not be confused with lazy constraints, which are valid inequalities whose insertion into the model is delayed until their violation to reduce the complexity of the LP relaxation in each B&B node. |
| NFP-CM-VS2                 | this work               | Same implementation than NFP-CM-VS and substitution of the Constraints (24)–(29) by Constraints (48)–(49) inserted into the root node of the model.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |

tation of our models instead of Constraints (39).

$$\sum_{v_{ij}^{fgk} \in C_q} v_{ij}^{fgk} \leq 1, \quad \forall C_q \in \{\text{cliques covering constraints (39)}\} \quad (40)$$

### 3.3.3. Valid inequalities among three pieces

**Infeasible relative placements of three pieces.** One key advantage derived from our convex decomposition based on vertical slices is that it permits the disclosing of many infeasible combinations of binary variables encoding feasible relative placements among multiple pieces, which can be explicitly inserted into the model as feasibility cuts. For instance, given three convex pieces A, B, and C as shown in Fig. 5a, if binary variables  $v_{AB}^{11r}$  and  $v_{AC}^{11}$  are enabled in any solution of the NFP-CM-VS model, then pieces B and C will be placed in the grey sub-regions  $R_{AB}^{11r}$  and  $R_{AC}^{111}$  with respect to the piece A, as shown in Fig. 5b. However, Fig. 5c shows that if the three aforementioned pieces are placed in the former relative placements, then the piece C cannot be placed in a relative feasible sub-region with respect to piece B that is compatible (feasible) with its relative placement with respect to piece A defined by

$R_{AC}^{111}$ , with the only exception of the left feasible region enabled by  $v_{BC}^{111}$ . In this way, many infeasible relative placements among three pieces can be detected a priori and represented as valid inequalities in the model. Thus, we can identify many infeasible relative placements among three pieces  $\{P_i, P_j, P_u\}$  by testing if the relative placement of piece  $P_u$  regarding piece  $P_j$  is feasible in any solution of the problem given the relative placements of pieces  $P_j$  and  $P_u$  regarding  $P_i$ .

**Feasibility cuts among multiple pieces.** Although we focus here on representing the infeasible relative placements among three pieces, our ideas can be generalized to combinations of  $(p \geq 3)$  pieces by fixing the pairwise relative placements of the first  $(p - 1)$  pieces and testing the feasibility for the relative placements of the last piece regarding the former pieces, at the expense of exponentially increasing the number of valid inequalities inserted into the model.

**Computation of feasibility cuts among three pieces.** Algorithm 2 computes all feasibility cuts derived from activating any triplet of convex feasible sub-regions among three pieces by testing their feasibility with the *IsInfeasible* function defined by Algorithm 3, which partially evaluates the feasibility of  $v_{ju}^{ghk'}$  given that  $v_{ij}^{fgk}$  and  $v_{iu}^{fhk'}$  are enabled by testing if  $R_{ju}^{ghk'}$  and  $R_{iu}^{fhk'}$  regions over-

**Algorithm 2** *ComputeValidCuts* procedure computes all feasibility cuts derived from the activation of any triplet of convex feasible sub-regions among three pieces by testing their feasibility using [Algorithm 3](#).

**Input:** counter-clockwise oriented boundaries  $\delta_a \in \mathbb{R}^{2 \times n}$ ,  $\delta_b \in \mathbb{R}^{2 \times m}$

**Output:** C set of feasibility cuts among three pieces

```

1: procedure COMPUTEVALIDCUTS
2:    $C \leftarrow \emptyset$ 
3:   for all  $(i, j, u) \in \{N \times N \times N | i < j < u\}$  do
4:     for  $(1 \leq f \leq Q_i)$  do
5:       for  $(1 \leq g \leq Q_j)$  do
6:         for  $(1 \leq h \leq Q_u)$  do
7:            $\partial NFP_{ij}^{fg} \leftarrow \partial NFP_{ij}^{fg} \oplus -r_i$ 
8:            $\partial NFP_{iu}^{fh} \leftarrow \partial NFP_{iu}^{fh} \oplus -r_i$ 
9:            $\partial NFP_{ju}^{gh} \leftarrow \partial NFP_{ju}^{gh} \oplus -r_j$ 
10:          for all  $k \in K_{ij}^{fg}$  do
11:            for all  $k' \in K_{iu}^{fh}$  do
12:               $\Phi \leftarrow \emptyset$ 
13:              for all  $k'' \in K_{ju}^{gh}$  do
14:                if  $\text{IsInfeasible}(R_{ij}^{fgk}, R_{iu}^{fhk'}, R_{ju}^{ghk''})$  then
15:                   $\Phi \leftarrow \Phi \cup k''$ 
16:                end if
17:              end for
18:               $C \leftarrow C \cup (v_{ij}^{fgk} + v_{iu}^{fhk'} + \sum_{k'' \in \Phi} v_{ju}^{ghk''} \leq 2)$ 
19:            end for
20:          end for
21:        end for
22:      end for
23:    end for
24:  end procedure

```

▷ We initialize the output  
 ▷ We build the feasibility cuts for all piece triplets  
 ▷ tests all convex parts of  $P_i$   
 ▷ tests all convex parts of  $P_j$   
 ▷ tests all convex parts of  $P_u$   
 ▷ translation to the reference point of  $P_i$   
 ▷ translation to the reference point of  $P_i$   
 ▷ translation to the reference point of  $P_j$   
 ▷ tests every feasible sub-region of  $NFP_{ij}^{fg}$   
 ▷ tests every feasible sub-region of  $NFP_{iu}^{fh}$   
 ▷ initializes the set of infeasible  $k''$  indexes  
 ▷ tests every feasible sub-region of  $NFP_{ju}^{gh}$   
 ▷ feasibility test in Algorithm 3  
 ▷ Adds a feasibility Constraint (41)

**Algorithm 3** *IsInfeasible()* function uses interval arithmetic ([Moore et al., 2009, ch.2](#)) to test if three fixed relative placements among three pieces produce an infeasible solution by testing their potential overlapping along the horizontal direction. A closed interval of real numbers is denoted by  $[a, b] = \{x \in \mathbb{R} : a \leq x \leq b\}$ . Let be  $X = [\underline{X}, \bar{X}]$  and  $Y = [\underline{Y}, \bar{Y}]$  two intervals, we say that both are equal if they are the same subset of  $\mathbb{R}$ . The sum of two intervals is  $Z = X + Y = [\underline{X} + \underline{Y}, \bar{X} + \bar{Y}]$ . The intersection of two intervals is  $Z = X \cap Y = \{z \in \mathbb{R} : z \in X \wedge z \in Y\} = [\max\{\underline{X}, \underline{Y}\}, \min\{\bar{X}, \bar{Y}\}]$ , and  $X \cap Y = \emptyset$  if  $\bar{Y} < \underline{X}$  or  $\bar{X} < \underline{Y}$ .

**Input:**  $R_{ij}^{fgk}, R_{iu}^{fhk'}, R_{ju}^{ghk''} \subset \mathbb{R}^2$

**Output:** *true* if  $\exists r_j \in R_{ij}^{fgk} | R_{iu}^{fhk'} \cap R_{ju}^{ghk''} \neq \emptyset$ , or *false* otherwise

```

1: function ISINFEASIBLE( $R_{ij}^{fgk}, R_{iu}^{fhk'}, R_{ju}^{ghk''}$ )
2:    $x_{ij}^{fgk} \leftarrow [\min\{R_{ij,x}^{fgk}\}, \max\{R_{ij,x}^{fgk}\}] \subset \mathbb{R}$ 
3:    $x_{iu}^{fhk'} \leftarrow [\min\{R_{iu,x}^{fhk'}\}, \max\{R_{iu,x}^{fhk'}\}] \subset \mathbb{R}$ 
4:    $x_{ju}^{ghk''} \leftarrow [\min\{R_{ju,x}^{ghk''}\}, \max\{R_{ju,x}^{ghk''}\}] \subset \mathbb{R}$ 
5:    $\delta x_{iju} \leftarrow x_{ij}^{fgk} + x_{ju}^{ghk''} = [x_{ij}^{fgk} + x_{ju}^{ghk''}, \bar{x}_{ij}^{fgk} + \bar{x}_{ju}^{ghk''}]$ 
6:    $z \leftarrow \delta x_{iju} \cap x_{iu}^{fhk'} = [\max\{\delta x_{iju}, x_{iu}^{fhk'}\}, \min\{\bar{\delta x}_{iju}, \bar{x}_{iu}^{fhk'}\}]$ 
7:    $r \in \{\text{true}, \text{false}\}$ 
8:   if  $z = \emptyset$  then
9:      $r \leftarrow \text{true}$ 
10:  else
11:     $r \leftarrow \text{false}$ 
12:  end if
13:  return  $r$ 
14: end function

```

▷ sets the x-axis interval spanned by  $R_{ij}^{fgk}$   
 ▷ sets the x-axis interval spanned by  $R_{iu}^{fhk'}$   
 ▷ sets the x-axis interval spanned by  $R_{ju}^{ghk''}$   
 ▷ sum of x-axis intervals  
 ▷ gets feasible relative placement interval for  $u$   
 ▷ feasibility test

lap along the horizontal direction using interval arithmetic ([Moore, 1966](#); [Moore, Kearfott, & Cloud, 2009](#)). [Algorithm 2](#) traverses all valid combinations of three pieces in step 3 by implementing the classic algorithm of [Mifsud \(1963\)](#) for generating all combinations of three pieces in lexicographic order. Next, [Algorithm 2](#) explores all combinations of convex NFP parts among three pieces in steps

4–6 to test the feasibility of the relative placements defined by each combination of convex feasible sub-regions in steps 10–16. On the other hand, [Algorithm 3](#) tests whether the activation of the feasible sub-regions among three pieces induces an infeasible solution of our NFP-CM-VS model by testing their potential overlapping along the horizontal direction using interval arithmetic

(Moore et al., 2009, ch.2). *IsInfeasible* function receives three vertical region objects  $R_{ij}^{fgk} = \{a_{ij}^{fgk}, b_{ij}^{fgk}\} \subset \mathbb{R}^2$  defined by the extreme points of their corresponding  $e_{ij}^{fgk}$  edges, which encode the vertical slices of the three convex feasible sub-regions derived from the convex parts of the pieces, as shown in Fig. 5. Then, this function computes the  $x$ -axis intervals spanned by each region in steps 2–4, whilst step 5 computes the feasible  $x$ -axis interval for piece  $P_u$  that is induced by the  $x$ -axis relative placement of  $P_j$  regarding  $P_i$ , which is denoted by  $\delta_{iju}$ . Finally, step 6 tests if the relative placement of  $P_u$  induced by  $P_j$  regarding  $P_i$  is feasible by testing if the intersection of  $\delta_{iju}$  and  $x_{iu}^{fhk'}$  intervals is not empty. Then, all infeasible relative placements among three pieces can be inserted into our model using combinatorial Benders cuts (Codato & Fischetti, 2006), as defined by Constraints (41). Although our feasibility test only considers the overlapping along the horizontal direction, it is capable of detecting a large number of infeasible relative placements among three pieces. Because the number of valid inequalities (41) grows rapidly, raising to tens of millions for large problem instances, we insert this family of feasibility cuts into our NFP-CM-VS models as user cuts instead of inserting them into the master problem (17), as detailed in Table 1.

$$\begin{aligned} v_{ij}^{fgk} + v_{iu}^{fhk'} + \sum_{k'' \in K_{ju}^{gh}} v_{ju}^{ghk''} &\leq 2, \forall (i, j, u, f, g, h, k, k', k'') \\ : v_{ij}^{fgk} + v_{iu}^{fhk'} &= 2 \Rightarrow \forall r_B \in R_{ij}^{fgk}, R_{iu}^{fhk'} \cap R_{ju}^{ghk''} = \emptyset \end{aligned} \quad (41)$$

### 3.4. The NFP-CM-VS2 model

As mentioned above, our basic NFP-CM-VS model triples in average the number of constraints of the NFP-CM model (Cherri et al., 2016b) to break the symmetries of the feasible space, unlike the Improved NFP-CM model (Rodrigues et al., 2017) that only doubles them. To solve this drawback, we introduce a reformulation of our NFP-CM-VS model, called NFP-CM-VS2 and detailed below, which is derived from the full NFP-CM-VS model (17)–(41) by substituting the Constraints (24)–(29) by the Constraints (48)–(49). Then, the full NFP-CM-VS2 model is defined by the objective function (42) and the Constraints (43)–(52), together with all symmetry-breakings, valid cuts, and variable eliminations defined by expressions (33)–(41).

$$\min \quad z = L \quad (42)$$

$$\text{s.t.} \quad l_i^{\min} \leq x_i \leq L - l_i^{\max}, \quad 1 \leq i \leq N \quad (43)$$

$$h_i^{\min} \leq y_i \leq H - h_i^{\max}, \quad 1 \leq i \leq N \quad (44)$$

$$L_{lb} \leq L \leq L_{ub} \quad (45)$$

$$\begin{aligned} (b_{ij,x}^{fgk} - a_{ij,x}^{fgk})(y_j - y_i) + (a_{ij,y}^{fgk} - b_{ij,y}^{fgk})(x_j - x_i) + C_{ij}^{fgk} &\leq (1 - v_{ij}^{fgk})M_{ij}^{fgk} \\ \forall (i, j, f, g, k) &\in \{I_{ij}^{fg} \times T_{ij}^{fg} \cup B_{ij}^{fg}\} \\ M_{ij}^{fgk} &= |b_{ij,x}^{fgk} - a_{ij,x}^{fgk}|H + |a_{ij,y}^{fgk} - b_{ij,y}^{fgk}|L_{ub} + C_{ij}^{fgk} \end{aligned} \quad (46)$$

$$v_{ij}^{fgl} + v_{ij}^{fgr} + \sum_{k \in T_{ij}^{fg} \cup B_{ij}^{fg}} v_{ij}^{fgk} = 1, \quad \forall (i, j, f, g) \in I_{ij}^{fg} \quad (47)$$

$$\begin{aligned} x_j - x_i &\geq -(L_{ub} - (l_j^{\min} + l_i^{\max}))v_{ij}^{fgl} + \bar{x}_{ij}^{fg}v_{ij}^{fgr} \\ &+ \sum_{k \in T_{ij}^{fg}} b_{ij,x}^{fgk}v_{ij}^{fgk} + \sum_{k \in B_{ij}^{fg}} a_{ij,x}^{fgk}v_{ij}^{fgk}, \quad \forall (i, j, f, g) \in I_{ij}^{fg} \end{aligned} \quad (48)$$

$$\begin{aligned} x_j - x_i &\leq \bar{x}_{ij}^{fg}v_{ij}^{fgl} + (L_{ub} - (l_i^{\min} + l_j^{\max}))v_{ij}^{fgr} \\ &+ \sum_{k \in T_{ij}^{fg}} a_{ij,x}^{fgk}v_{ij}^{fgk} + \sum_{k \in B_{ij}^{fg}} b_{ij,x}^{fgk}v_{ij}^{fgk}, \quad \forall (i, j, f, g) \in I_{ij}^{fg} \end{aligned} \quad (49)$$

$$L \in \mathbb{R}^+ \quad (50)$$

$$(x_i, y_i) \in \mathbb{R}^2, \quad 1 \leq i \leq N \quad (51)$$

$$v_{ij}^{fgk} \in \{0, 1\}, \quad \forall (i, j, f, g, k) \in \{I_{ij}^{fg} \times K_{ij}^{fg}\} \quad (52)$$

*The most compact symmetry-breaking.* The Constraints (48) and (49) introduce a significant reduction in the complexity of the NFP-CM-VS model by factorizing all constraints encoding the vertical lines bounding the feasible sub-regions of each convex  $NFP_{ij}^{fg}$  part into only two constraints, whilst they hold the symmetry-breaking of the symmetric solutions for the space of feasible relative placements between pieces with much fewer constraints per convex  $NFP_{ij}^{fg}$  part than the state-of-the-art Improved NFP-CM model (Rodrigues et al., 2017). Note that the Improved NFP-CM model requires 2 constraints per binary variable of each convex  $NFP_{ij}^{fg}$  part to cover the feasible space as shown in Fig. 2d, whilst the NFP-CM-VS2 model only requires 1 constraint per binary variable (ineq. 22) plus two constraints (ineq. (48)–(49)) per convex  $NFP_{ij}^{fg}$  part. Thus, if the valid cuts among three pieces (41) are discarded from the full NFP-CM-VS2 model to define a weaker formulation called NFP-CM-VS2nc, then NFP-CM-VS2nc would provide the most compact symmetry-breaking of the feasible space of relative placements between pieces reported in the literature.

## 4. Evaluation

The goals of the experiments in this section are as follows: (1) to evaluate the performance of our two new MIP models, called NFP-CM-VS and NFP-CM-VS2; (2) to evaluate the impact of the family of feasibility cuts among three pieces defined by Constraints (41); (3) to carry out a fair comparison of the performance of our new models with the family of state-of-the-art continuous MIP models for irregular strip packing, which is based on the same hardware and software platform; (4) to replicate the state-of-the-art family of NFP-CM and Improved NFP-CM models from scratch; (5) to develop a reproducible benchmark of state-of-the-art MIP models based on our software implementation of all models evaluated herein into the same software library, which is provided as supplementary material (see Appendix B); (6) to evaluate the state-of-the-art Improved NFP-CM in a standard benchmark (Cherri et al., 2016b) including many unexplored problem instances not considered in its introductory paper (Rodrigues et al., 2017); (7) the independent confirmation of previous findings and results reported in the literature; and finally, (8) to elucidate the current state of the art on continuous MIP models for irregular strip packing in a sound and reproducible way.

### 4.1. Experimental setup

We reproduce the same experiments carried-out by Cherri et al. (2016b) to evaluate all models detailed in Table 1 by evaluating the same set of problem instances used by the former authors, plus

one additional large instance (Blaz2) for a total of 70 problem instances. Thus, our experiments include the same thirty-five small problem instances, twenty-four metal instances with holes, and ten large instances evaluated in (Cherri et al., 2016b, Tables 2–4), which include most of the problem instances evaluated by Alvarez-Valdes et al. (2013) that were also considered by Cherri et al. (2016b). In addition, the state-of-the-art Improved NFP-CM model, which sets our baseline for comparison, is evaluated here for the first time in most of the small instances shown in Table 2, and all metal and large problem instances shown in Tables 3 and 4.

Table 1 details the state-of-the-art MIP models for irregular strip packing evaluated herein and the details of our software implementation. To study the impact of our new family of feasibility cuts among three pieces (41), we have defined two versions of our new NFP-CM-VS model to be evaluated in our experiments, which are called NFP-CM-VSnc and full NFP-CM-VS respectively, as shown in Table 1. The only difference between these two later models is that NFP-CM-VSnc does not include our new family of feasibility cuts among three pieces (41).

All our experiments are based on our own software implementation of all MIP models evaluated herein, which uses the Java API of Gurobi 9.5 to solve the models. Our source code and a pre-compiled version of our software are publicly available in our reproducibility dataset (Lastra-Díaz & Ortuño, 2023). We have replicated the state-of-the-art family of NFP-CM models (Cherri et al., 2016b; Rodríguez et al., 2017) from scratch by integrating the three steps to evaluate all models as follows: (1) pre-processing to decompose the pieces into convex parts based on the CGAL implementation of the Greene's algorithm (Greene, 1983); (2) in-memory building of the MIP models by calling the Gurobi 9.5 API; and (3) resolution of the optimization models by calling the proper Gurobi functions with its default parameters. As we mentioned above, we use the ECC8 software library (Conte et al., 2020) to compute the cliques used to define the clique-based cuts (40) inserted as SOS-1 constraints into our NFP-CM-VS and NFP-CM-VS2 models. The version of the ECC algorithm (Conte et al., 2020) provided by ECC8 is not deterministic, which means that subsequent calls to ECC8 with the same data could return different cliques. For this reason, we save the cliques computed by ECC8 into plain text files to allow the exact replication of our experiments, and to study isolately the impact of the Valid cuts (41) in our two former models regarding the NFP-CM-VSnc model. All our experiments were implemented on a desktop UBUNTU 20.04 computer with an AMD Ryzen 7 5800X @3.8 GHz CPU (8 cores) and 64 Gb RAM.

Concerning the evaluation of large problem instances in Table 4, the results on large problem instances could be misleading because all MIP models evaluated here are far from being efficient for these sizes of problems. Thus, the feasible solutions heuristically found by Gurobi might be biased toward models with weaker formulations. However, despite this drawback, we consider these large problem instances in our experiments to reproduce the same experiments reported by Cherri et al. (2016b).

#### 4.2. Reproducing our benchmarks

All our experiments were generated by running the *RamNest-driver* program distributed with our RAMNEST V1R3 software library (Lastra-Díaz & Ortuño, 2023) with three reproducible benchmark files that defines all problem instances evaluated herein. All problem instances are defined in ESICUP XML file format and provided with our raw output data in our reproducibility dataset (Lastra-Díaz & Ortuño, 2023). The evaluation of each MIP model generates a Scalable Vector Graphics (SVG) file and another LaTeX TikZ picture text file providing an image of the solution, and a raw output file in (\*.csv) file format reporting the following data for each problem instance (see Appendix A): (1) name of the problem

instance; (2) number of pieces; (3) nesting efficiency; (4–5) lower and upper bounds of the solution; (6) MIP gap; (7) number of binary variables; (8) number of B&B nodes; (9) number of simplex iterations; (10) running time in seconds; and (11) the number of constraints. All the data reported here is automatically generated by running the *benchmark\_results* R-language script file on the collection of raw output files. Finally, all our models, experiments, and results can be exactly reproduced by following the instructions detailed in Appendix B, which also explains how to evaluate the MIP models detailed in Table 1 in other problem instances provided in the ESICUP XML file format. The aim of our reproducibility resources is not only to allow our results to easily be reproduced, but also to evaluate unseen instances provided in the ESICUP XML file format. Furthermore, we provide all MIP models evaluated here in the LP file format (Lastra-Díaz & Ortuño, 2023) to permit their direct evaluation with Gurobi 9.5, or another solver as CPLEX, instead of using our software.

#### 4.3. Evaluation metrics

To evaluate the performance of the MIP models detailed in Table 1, we compare the upper bound, solution gap ( $\frac{UB-LB}{UB}$ ), and computation time in seconds obtained by each MIP model in the evaluation of all problem instances. In order to provide a fair and unbiased comparison of the performance of all MIP models evaluated here, we adopt the same approach proposed by Cherri et al. (2016b), which uses performance profiles (Dolan & Moré, 2002) based on the ratios between the computation time of each model and the best computation time obtained by any model as a performance metric. Let be  $\Phi$  and  $M$  the sets of problem instances and MIP models evaluated in this study, respectively. Then the ratio of computation times  $t_{\phi,m}$  for each model  $m \in M$  is defined by  $r_{\phi,m}$  in formula (53), considering that the computation time is infinite whenever the models are unable to solve the problem instance up to optimality. We assume an arbitrary parameter  $r_M \geq r_{\phi,m} \forall (\phi, m) \in \Phi \times M$ , such that  $r_{\phi,m} = r_M$  if and only if the model  $m$  is unable to solve the problem  $\phi$ . Dolan & Moré (2002) define a *performance profile* of a solver or optimization model as the Cumulative Distribution Function (CDF) of the computation time ratios  $r_{\phi,m}$ , denoted by  $\rho_m(\tau)$ , and defined in formula (54) below. The performance profiles defined by the  $\rho_m(\tau)$  function set a well-founded and broadly adopted metric to compare optimization models by avoiding any bias derived from a particular set of problem instances and dealing with those cases in which the models are unable to solve the problem up to optimality.

$$r_{\phi,m} = \frac{t_{\phi,m}}{\min\{t_{\phi,i} : i \in M\}}, \quad \forall (\phi, m) \in \Phi \times M \quad (53)$$

$$\rho_m(\tau) = \frac{1}{n_\phi} \{ \phi \in \Phi : r_{\phi,m} \leq \tau \}, \quad \forall (m, \tau) \in M \times [1, r_M] \quad (54)$$

#### 4.4. Results obtained

Tables 2 and 3 report the Upper Bound (UB), terminating gap, and running time in seconds obtained by the MIP models in the evaluation of the thirty-five small problem instances and the twenty-four metal ones, respectively. Table 4 reports the Lower Bound (LB), UB, and terminating gap obtained in the evaluation of the eleven large problem instances. Figs. 6 and 7 show the performance profile curves comparing the performance ratio (54) obtained by all models in the evaluation of the small and metal instances, respectively. Table 5 reports the number of binary variables and constraints defined by each MIP model evaluated in all metal problem instances. Fig. 8 shows the performance profile comparing the NFP-CM-VSnc model with NFP-CM-VS to study

**Table 2**  
Upper Bound (UB), terminating gap ( $\frac{UB-LB}{UB}$ ), and running time in seconds obtained by the MIP models in the evaluation of small problem instances with # pieces. Termination MIPGap parameter was set to 0.002. Time Limit (TL) was set to 3600 seconds. Best results are shown in bold. Instances solved up to optimality within 1 hour for the first time in the literature are marked with an asterisk (\*).

| Instance                                    | #  | NFP-CM (Cherri et al., 2016b) |             |              | Improved NFP-CM (Rodrigues et al., 2017) |             |              | NFP-CM-VS (this work) |             |              | NFP-CM-VS2 (this work) |             |              |
|---------------------------------------------|----|-------------------------------|-------------|--------------|------------------------------------------|-------------|--------------|-----------------------|-------------|--------------|------------------------|-------------|--------------|
|                                             |    | UB                            | GAP         | Time (secs)  | UB                                       | GAP         | Time (secs)  | UB                    | GAP         | Time (secs)  | UB                     | GAP         | Time (secs)  |
| three (Alvarez-Valdes et al., 2013)         | 3  | 6                             | 0           | <b>0</b>     | 6                                        | 0           | 0.05         | 6                     | 0           | <b>0</b>     | 6                      | 0           | 0.02         |
| Shapes4 (Oliveira, Gomes, & Ferreira, 2000) | 4  | 24                            | 0           | <b>0.28</b>  | 24                                       | 0           | 0.31         | 24                    | 0           | 0.55         | 24                     | 0           | 0.61         |
| fu5 (Fujita, Akagi, & Hirokawa, 1993)       | 5  | 17.89                         | 0           | 0.03         | 17.89                                    | 0           | 0.05         | 17.89                 | 0           | <b>0.02</b>  | 17.89                  | 0           | <b>0.02</b>  |
| glass1 (Fischetti & Luzzi, 2009)            | 5  | 45                            | 0           | <b>0.02</b>  | 45                                       | 0           | 0.02         | 45                    | 0           | 0.06         | 45                     | 0           | 0.04         |
| fu6 (Fujita et al., 1993)                   | 6  | 23                            | 0           | <b>0.03</b>  | 23                                       | 0           | 0.04         | 23                    | 0           | 0.07         | 23                     | 0           | 0.04         |
| threep2 (Alvarez-Valdes et al., 2013)       | 6  | 9.33                          | 0           | <b>0.47</b>  | 9.33                                     | 0           | 0.53         | 9.33                  | 0           | 1.09         | 9.33                   | 0           | 0.91         |
| threep2w9 (Alvarez-Valdes et al., 2013)     | 6  | 8                             | 0           | 0.67         | 8                                        | 0           | 0.63         | 8                     | 0           | 0.71         | 8                      | 0           | <b>0.42</b>  |
| fu7 (Fujita et al., 1993)                   | 7  | 24                            | 0           | <b>0.09</b>  | 24                                       | 0           | 0.12         | 24                    | 0           | <b>0.09</b>  | 24                     | 0           | 0.13         |
| glass2 (Fischetti & Luzzi, 2009)            | 7  | 45                            | 0           | <b>0.06</b>  | 45                                       | 0           | 0.87         | 45                    | 0           | 0.74         | 45                     | 0           | 0.77         |
| fu8 (Fujita et al., 1993)                   | 8  | 24                            | 0           | 0.7          | 24                                       | 0           | 0.76         | 24                    | 0           | 0.73         | 24                     | 0           | <b>0.44</b>  |
| Shapes8 (Oliveira et al., 2000)             | 8  | 26                            | 0           | 920.3        | 26                                       | 0           | 738.2        | 26                    | 0           | 703.5        | 26                     | 0           | <b>450.6</b> |
| fu9 (Fujita et al., 1993)                   | 9  | 25                            | 0           | 0.78         | 25                                       | 0           | 1.45         | 25                    | 0           | 0.97         | 25                     | 0           | <b>0.74</b>  |
| threep3 (Alvarez-Valdes et al., 2013)       | 9  | 13.53                         | 0           | 193.1        | 13.53                                    | 0           | 274.3        | 13.53                 | 0           | 205.1        | 13.53                  | 0           | <b>99.23</b> |
| threep3w9 (Alvarez-Valdes et al., 2013)     | 9  | 11                            | 0.05        | TL           | 11                                       | 0           | 304.8        | 11                    | 0           | 661.9        | 11                     | 0           | <b>177.5</b> |
| glass3 (Fischetti & Luzzi, 2009)            | 9  | 100                           | 0           | <b>5.61</b>  | 100                                      | 0           | 315.6        | 100                   | 0           | 19.68        | 100                    | 0           | 12.78        |
| fu10 (Fujita et al., 1993)                  | 10 | 28.69                         | 0           | 121.5        | 28.69                                    | 0           | <b>17.67</b> | 28.69                 | 0           | 22.02        | 28.69                  | 0           | 27.32        |
| Dighe2 (Dighe & Jakiela, 1995)              | 10 | 100.1                         | 0           | <b>4.06</b>  | 100                                      | 0           | 52.76        | 100                   | 0           | 6.25         | 100                    | 0           | 7.56         |
| J1-10-20-0 (Alvarez-Valdes et al., 2013)    | 10 | 18                            | 0           | 4.8          | 18                                       | 0           | 11.13        | 18                    | 0           | 5.31         | 18                     | 0           | <b>3.22</b>  |
| J1-10-20-1 (Alvarez-Valdes et al., 2013)    | 10 | 17                            | 0           | 10.67        | 17                                       | 0           | 14.28        | 17                    | 0           | 7.3          | 17                     | 0           | <b>4.3</b>   |
| J1-10-20-2 (Alvarez-Valdes et al., 2013)    | 10 | 20                            | 0           | 4.9          | 20                                       | 0           | 7.84         | 20                    | 0           | <b>4.57</b>  | 20                     | 0           | 5.16         |
| J1-10-20-3* (Alvarez-Valdes et al., 2013)   | 10 | 20.75                         | 0           | 1373         | 20.75                                    | 0           | 826.2        | 20.75                 | 0           | 72.3         | 20.75                  | 0           | <b>39.99</b> |
| J1-10-20-4 (Alvarez-Valdes et al., 2013)    | 10 | 12.5                          | 0           | 49.13        | 12.5                                     | 0           | 189.2        | 12.5                  | 0           | <b>10.66</b> | 12.5                   | 0           | 21.68        |
| J1-12-20-0 (Alvarez-Valdes et al., 2013)    | 12 | 12                            | 0           | 75.92        | 12                                       | 0           | 314.3        | 12                    | 0           | 37.52        | 12                     | 0           | <b>19.1</b>  |
| J1-12-20-1 (Alvarez-Valdes et al., 2013)    | 12 | 10                            | 0           | 44.87        | 10                                       | 0           | 75.66        | 10                    | 0           | 21.72        | 10                     | 0           | <b>13.21</b> |
| J1-12-20-2 (Alvarez-Valdes et al., 2013)    | 12 | 12                            | 0           | <b>11.68</b> | 12                                       | 0           | 42.73        | 12                    | 0           | 23.47        | 12                     | 0           | 24.1         |
| J1-12-20-3 (Alvarez-Valdes et al., 2013)    | 12 | 8                             | 0           | 604.1        | 8                                        | 0           | 416.8        | 8                     | 0           | <b>115.7</b> | 8                      | 0           | 138.2        |
| J1-12-20-4 (Alvarez-Valdes et al., 2013)    | 12 | 13                            | 0           | 2437         | 13                                       | 0           | 934.6        | 13                    | 0           | 86.43        | 13                     | 0           | <b>40.68</b> |
| fu* (Fujita et al., 1993)                   | 12 | 33.14                         | 0           | 1135         | 33.14                                    | 0           | <b>250.8</b> | 33.14                 | 0           | 268.4        | 33.14                  | 0           | 582          |
| J1-14-20-0 (Alvarez-Valdes et al., 2013)    | 14 | 12                            | 0           | 317.1        | 12                                       | 0           | 841.2        | 12                    | 0           | 364.9        | 12                     | 0           | <b>218</b>   |
| J1-14-20-1 (Alvarez-Valdes et al., 2013)    | 14 | 12                            | 0.17        | TL           | 12                                       | 0.17        | TL           | 11.33                 | <b>0.03</b> | TL           | 11.33                  | <b>0.03</b> | TL           |
| J1-14-20-2 (Alvarez-Valdes et al., 2013)    | 14 | 14                            | <b>0.14</b> | TL           | 14                                       | <b>0.14</b> | TL           | 14                    | <b>0.14</b> | TL           | 14                     | <b>0.14</b> | TL           |
| J1-14-20-3 (Alvarez-Valdes et al., 2013)    | 14 | 10                            | 0           | 375.5        | 10                                       | 0           | 330          | 10                    | 0           | 170.8        | 10                     | 0           | <b>80.41</b> |
| J1-14-20-4 (Alvarez-Valdes et al., 2013)    | 14 | 14                            | <b>0.14</b> | TL           | 14                                       | <b>0.14</b> | TL           | 14                    | <b>0.14</b> | TL           | 14                     | <b>0.14</b> | TL           |
| Poly1a (Hopper, 2000)                       | 15 | 15.21                         | <b>0.15</b> | TL           | 16.17                                    | 0.2         | TL           | 15.68                 | 0.17        | TL           | 15.62                  | 0.17        | TL           |
| Dighe1* (Dighe & Jakiela, 1995)             | 16 | 100                           | 0           | 2950         | 128.5                                    | 0.22        | TL           | 124.2                 | 0.19        | TL           | 100                    | 0           | <b>2631</b>  |
| # Optimal                                   |    |                               |             | 30           |                                          |             | 30           |                       |             | 30           |                        |             | <b>31</b>    |
| # Best running time                         |    |                               |             | 10           |                                          |             | 2            |                       |             | 6            |                        |             | <b>16</b>    |

**Table 3**

Upper Bound (UB), terminating gap ( $\frac{UB-LB}{UB}$ ), and running time in seconds obtained by the MIP models in the evaluation of metal problem instances with # pieces. Termination MIPGap parameter was set to 0.002. Time Limit (TL) was set to 3600 seconds. The 'X' symbol denotes that no feasible solution was obtained. Best results are shown in bold. Instances solved up to optimality within 1 h for the first time in the literature are marked with an asterisk (\*).

| Instance                          | #  | NFP-CM (Cherri et al., 2016b) |             |              | Improved NFP-CM (Rodrigues et al., 2017) |      |             | NFP-CM-VS (this work) |             |             | NFP-CM-VS2 (this work) |             |              |
|-----------------------------------|----|-------------------------------|-------------|--------------|------------------------------------------|------|-------------|-----------------------|-------------|-------------|------------------------|-------------|--------------|
|                                   |    | UB                            | GAP         | Time (secs)  | UB                                       | GAP  | Time (secs) | UB                    | GAP         | Time (secs) | UB                     | GAP         | Time (secs)  |
| metal0-3 (Cherri et al., 2016b)   | 3  | 501                           | 0           | <b>0</b>     | 501                                      | 0    | 0.01        | 501                   | 0           | <b>0</b>    | 501                    | 0           | <b>0</b>     |
| metal0-4 (Cherri et al., 2016b)   | 4  | 501                           | 0           | <b>0</b>     | 501                                      | 0    | <b>0</b>    | 501                   | 0           | 0.02        | 501                    | 0           | 0.01         |
| metal0-5 (Cherri et al., 2016b)   | 5  | 501                           | 0           | <b>0</b>     | 501                                      | 0    | 0.01        | 501                   | 0           | <b>0</b>    | 501                    | 0           | <b>0</b>     |
| metal0-6 (Cherri et al., 2016b)   | 6  | 785                           | 0           | 0.05         | 785                                      | 0    | <b>0.03</b> | 785                   | 0           | 0.1         | 785                    | 0           | 0.07         |
| metal0-7 (Cherri et al., 2016b)   | 7  | 501                           | 0           | <b>0.1</b>   | 501                                      | 0    | 0.63        | 501                   | 0           | 1.3         | 501                    | 0           | 1.56         |
| metal0-8 (Cherri et al., 2016b)   | 8  | 529                           | 0           | <b>0.75</b>  | 529                                      | 0    | 1.66        | 529                   | 0           | 5.94        | 529                    | 0           | 3.72         |
| metal0-9 (Cherri et al., 2016b)   | 9  | 529                           | 0           | <b>0.76</b>  | 529                                      | 0    | 3.93        | 529                   | 0           | 8.98        | 529                    | 0           | 9.66         |
| metal0-10 (Cherri et al., 2016b)  | 10 | 356                           | 0           | <b>2.22</b>  | 356                                      | 0    | 8.18        | 356                   | 0           | 4.33        | 356                    | 0           | 5.5          |
| metal3-2* (Cherri et al., 2016b)  | 10 | 291.1                         | 0           | 3568         | 291.1                                    | 0    | 16.21       | 291.1                 | 0           | <b>5.59</b> | 291.1                  | 0           | 5.78         |
| metal0-11* (Cherri et al., 2016b) | 11 | 364.3                         | 0           | 13.66        | 364.3                                    | 0    | 8.15        | 364.3                 | 0           | 7.19        | 364.3                  | 0           | <b>5.67</b>  |
| metal0-12* (Cherri et al., 2016b) | 12 | 399                           | 0           | 92.73        | 399                                      | 0    | 74.77       | 399                   | 0           | 20.97       | 399                    | 0           | <b>12.25</b> |
| metal0-13* (Cherri et al., 2016b) | 13 | 399                           | 0           | 87.94        | 399                                      | 0    | 200.7       | 399                   | 0           | 24.67       | 399                    | 0           | <b>7.44</b>  |
| metal1-1* (Cherri et al., 2016b)  | 14 | 286                           | 0           | 2.26         | 286                                      | 0    | 4.79        | 286                   | 0           | 1.67        | 286                    | 0           | 4.74         |
| metal3-3* (Cherri et al., 2016b)  | 15 | 492.1                         | 0.35        | TL           | 490                                      | 0.09 | TL          | 490                   | 0           | 561.1       | 490                    | 0           | <b>352</b>   |
| metal2-2 (Cherri et al., 2016b)   | 18 | 300                           | 0.15        | TL           | 300                                      | 0.15 | TL          | 300                   | 0.13        | TL          | 300                    | <b>0.05</b> | TL           |
| metal3-4 (Cherri et al., 2016b)   | 20 | 592.9                         | 0.28        | TL           | 618.8                                    | 0.31 | TL          | 586                   | <b>0.27</b> | TL          | 604                    | 0.29        | TL           |
| metal3-5 (Cherri et al., 2016b)   | 25 | 768                           | <b>0.3</b>  | TL           | X                                        | X    | TL          | 768                   | <b>0.3</b>  | TL          | 770                    | 0.31        | TL           |
| metal2-3* (Cherri et al., 2016b)  | 27 | 490                           | 0           | <b>43.77</b> | 490                                      | 0    | 179.8       | 490                   | 0           | 111.2       | 490                    | 0           | 191.4        |
| metal1-2 (Cherri et al., 2016b)   | 28 | 545                           | <b>0.16</b> | TL           | 568                                      | 0.2  | TL          | 544                   | <b>0.16</b> | TL          | 568                    | 0.2         | TL           |
| metal2-4 (Cherri et al., 2016b)   | 36 | 563                           | <b>0.14</b> | TL           | 590                                      | 0.18 | TL          | 569                   | 0.15        | TL          | 590                    | 0.18        | TL           |
| metal1-3 (Cherri et al., 2016b)   | 42 | 910.6                         | 0.25        | TL           | X                                        | X    | TL          | 905                   | 0.24        | TL          | 889                    | <b>0.23</b> | TL           |
| metal2-5 (Cherri et al., 2016b)   | 45 | 701                           | <b>0.14</b> | TL           | X                                        | X    | TL          | 712                   | 0.15        | TL          | 735                    | 0.18        | TL           |
| metal1-4 (Cherri et al., 2016b)   | 56 | 1175                          | <b>0.22</b> | TL           | X                                        | X    | TL          | X                     | X           | TL          | 1282                   | 0.29        | TL           |
| metal1-5 (Cherri et al., 2016b)   | 70 | 1639                          | <b>0.3</b>  | TL           | X                                        | X    | TL          | X                     | X           | TL          | X                      | X           | TL           |
| # Optimal                         |    |                               |             | 14           |                                          |      | 14          |                       |             | <b>15</b>   |                        |             | <b>15</b>    |
| # Best running time               |    |                               |             | <b>8</b>     |                                          |      | 2           |                       |             | 3           |                        |             | 6            |

**Table 4**

Lower Bound (LB), Upper Bound (UB), and terminating gap ( $\frac{UB-LB}{UB}$ ) obtained by the MIP models in the evaluation of all large problem instances. Termination MIPGap parameter was set to 0.002. Time Limit (TL) was set to 3600 seconds, but no model was able to solve any instance up to optimality within this time limit, or raising the initial  $L_{lb}$  values (see bounds in LP files). The 'X' symbol denotes that no feasible solution was obtained. Best UB results are shown in bold.

| Instance               | #  | NFP-CM (Cherri et al., 2016b) |                 |      | Improved NFP-CM (Rodrigues et al., 2017) |             |      | NFP-CM-VS (this work) |               |      | NFP-CM-VS2 (this work) |               |      |
|------------------------|----|-------------------------------|-----------------|------|------------------------------------------|-------------|------|-----------------------|---------------|------|------------------------|---------------|------|
|                        |    | LB                            | UB              | GAP  | LB                                       | UB          | GAP  | LB                    | UB            | GAP  | LB                     | UB            | GAP  |
| Blaz2-16               | 16 | 15                            | <b>21.417</b>   | 0.3  | 15                                       | 21.747      | 0.31 | 15                    | 21.74         | 0.31 | 15                     | 21.5          | 0.3  |
| Blaz2                  | 20 | 19                            | 26.15           | 0.27 | 19                                       | <b>25.5</b> | 0.25 | 19                    | 26.505        | 0.28 | 19                     | 26.5          | 0.28 |
| Mao                    | 20 | 1474                          | <b>2009.109</b> | 0.27 | 1474                                     | X           | X    | 1474                  | 7411          | 0.8  | 1474                   | X             | X    |
| Albano                 | 24 | 8705                          | 11516           | 0.24 | 8705                                     | X           | X    | 8705                  | <b>11024</b>  | 0.21 | 8705                   | 13304.768     | 0.35 |
| Marques                | 24 | 69                            | 83.367          | 0.17 | 69                                       | X           | X    | 69                    | 100           | 0.31 | 69                     | <b>83</b>     | 0.17 |
| Jakobs1 (Jakobs, 1996) | 25 | 10                            | <b>12</b>       | 0.17 | 10                                       | <b>12</b>   | 0.17 | 10                    | 13            | 0.23 | 10                     | <b>12</b>     | 0.17 |
| Jakobs2 (Jakobs, 1996) | 25 | 19                            | <b>27</b>       | 0.3  | 19                                       | X           | X    | 19                    | 28            | 0.32 | 19                     | 28            | 0.32 |
| Blaz1                  | 28 | 22                            | 29.75           | 0.26 | 22                                       | X           | X    | 22                    | <b>28.178</b> | 0.22 | 22                     | 28.549        | 0.23 |
| Dagli                  | 30 | 51                            | 68.389          | 0.25 | 51                                       | X           | X    | 51                    | X             | X    | 51                     | <b>67.861</b> | 0.25 |
| Shapes0                | 48 | 40                            | X               | X    | 40                                       | X           | X    | 40                    | X             | X    | 40                     | X             | X    |
| Trousers               | 64 | 218                           | <b>294.962</b>  | 0.26 | 218                                      | X           | X    | 218                   | X             | X    | 218                    | X             | X    |
| # Best UB              |    |                               | <b>5</b>        |      |                                          | 2           |      |                       | 2             |      |                        | 3             |      |

**Table 5**

Number of binary variables and constraints defined by the MIP models in the metal problem instances.

| Instance  | #  | NFP-CM (Cherri et al., 2016b) |              | Improved NFP-CM (Rodrigues et al., 2017) |             | NFP-CM-VS (this work) |             | NFP-CM-VS2 (this work) |             |
|-----------|----|-------------------------------|--------------|------------------------------------------|-------------|-----------------------|-------------|------------------------|-------------|
|           |    | Binary                        | Constraints  | Binary                                   | Constraints | Binary                | Constraints | Binary                 | Constraints |
| metal0-3  | 3  | <b>24</b>                     | <b>36</b>    | 39                                       | 88          | 32                    | 156         | 32                     | 120         |
| metal0-4  | 4  | <b>42</b>                     | <b>61</b>    | 64                                       | 144         | 54                    | 317         | 54                     | 257         |
| metal0-5  | 5  | <b>92</b>                     | <b>119</b>   | 121                                      | 265         | 108                   | 879         | 108                    | 735         |
| metal0-6  | 6  | <b>286</b>                    | <b>370</b>   | 414                                      | 907         | 372                   | 6387        | 372                    | 5883        |
| metal0-7  | 7  | <b>340</b>                    | <b>440</b>   | 484                                      | 1062        | 437                   | 8046        | 437                    | 7464        |
| metal0-8  | 8  | <b>477</b>                    | <b>594</b>   | 635                                      | 1380        | 583                   | 12912       | 583                    | 12094       |
| metal0-9  | 9  | <b>544</b>                    | <b>679</b>   | 718                                      | 1563        | 661                   | 15590       | 661                    | 14676       |
| metal0-10 | 10 | <b>615</b>                    | <b>769</b>   | 805                                      | 1755        | 743                   | 18574       | 743                    | 17560       |
| metal3-2  | 10 | <b>852</b>                    | <b>1056</b>  | 1102                                     | 2404        | 949                   | 28043       | 949                    | 26761       |
| metal0-11 | 11 | <b>690</b>                    | <b>865</b>   | 896                                      | 1958        | 828                   | 21883       | 828                    | 20767       |
| metal0-12 | 12 | <b>769</b>                    | <b>966</b>   | 991                                      | 2170        | 917                   | 25514       | 917                    | 24292       |
| metal0-13 | 13 | <b>932</b>                    | <b>1152</b>  | 1170                                     | 2551        | 1088                  | 33838       | 1088                   | 32350       |
| metal1-1  | 14 | <b>1103</b>                   | <b>1347</b>  | 1357                                     | 2949        | 1264                  | 43595       | 1264                   | 41835       |
| metal3-3  | 15 | <b>2049</b>                   | <b>2539</b>  | 2682                                     | 5850        | 2262                  | 105489      | 2262                   | 102453      |
| metal2-2  | 18 | <b>1322</b>                   | <b>1617</b>  | 1566                                     | 3423        | 1391                  | 48357       | 1391                   | 46463       |
| metal3-4  | 20 | <b>3760</b>                   | <b>4657</b>  | 4948                                     | 10789       | 4134                  | 262543      | 4134                   | 257011      |
| metal3-5  | 25 | <b>5985</b>                   | <b>7410</b>  | 7900                                     | 17221       | 6565                  | 527200      | 6565                   | 518430      |
| metal2-3  | 27 | <b>3030</b>                   | <b>3691</b>  | 3591                                     | 7839        | 3174                  | 168062      | 3174                   | 163754      |
| metal1-2  | 28 | <b>4662</b>                   | <b>5689</b>  | 5788                                     | 12594       | 5284                  | 386490      | 5284                   | 379186      |
| metal2-4  | 36 | <b>5436</b>                   | <b>6607</b>  | 6444                                     | 14055       | 5682                  | 404047      | 5682                   | 396347      |
| metal1-3  | 42 | <b>10677</b>                  | <b>13016</b> | 13293                                    | 28916       | 12060                 | 1345847     | 12060                  | 1329215     |
| metal2-5  | 45 | <b>8540</b>                   | <b>10365</b> | 10125                                    | 22071       | 8915                  | 795760      | 8915                   | 783690      |
| metal1-4  | 56 | <b>19148</b>                  | <b>23328</b> | 23872                                    | 51915       | 21592                 | 3238838     | 21592                  | 3209094     |
| metal1-5  | 70 | <b>30075</b>                  | <b>36625</b> | 37525                                    | 81591       | 33880                 | 6382635     | 33880                  | 6335995     |

the impact of the valid cuts among three pieces (41). Thirty one of the thirty-five small instances and fifteen of the twenty-four instances are solved up to optimality within one hour, including one metal instance with 27 pieces which is the largest problem instance solved up to date in this line of research. Fig. 9 shows the optimal solutions for three small instances and seven metal instances solved up to optimality within one hour for the first time in the literature.

Because of the lack of room, Tables 2 and 3 omit the presentation of the lower bounds. However, this missing information and all our raw output data are provided in Appendix A as supplementary material. Figs. 1 to 3 in Appendix A show the images with the solutions for all problem instances evaluated herein. Table 1 in Appendix A reports the number of binary variables and constraints defined by each MIP model evaluated in all small and large problem instances, whilst Tables 2–13 in Appendix A report the raw data obtained for the evaluation of all MIP models in all problem instances.

## 5. Discussion

### 5.1. Comparison in small problem instances

The entire family of NFP-CM-VS models obtains a significantly higher performance ratio than the family of state-of-the-art NFP-CM models in the small problem instances, and significantly outperforms the baseline Improved NFP-CM model, as shown in Fig. 6. Likewise, the NFP-CM-VS2 model solves the largest number of small instances up to optimality, 31 instances, as shown in the second last row of Table 2.

Looking at the last row of Table 2, we find that among the small instances solved up to optimality, NFP-CM-VS2 is the fastest model in 16 instances, whilst NFP-CM is the fastest one in 10, NFP-CM-VS in 6, and Improved NFP-CM in 2. Fig. 9 shows three new small instances solved up to optimality within one hour for the first time in the literature. However, J1-10-20-3 and fu instances are solved by all models, whilst the Dighe1 instance is solved by NFP-CM and

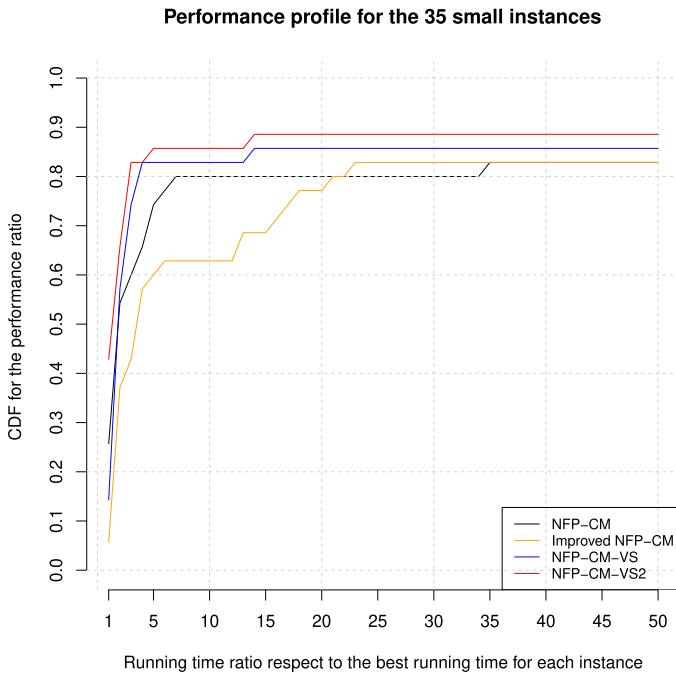


**Fig. 6.** Performance profile (Dolan & Moré, 2002) showing the Cumulative Distribution Function (CDF) for the performance ratio  $r_{\phi,m}$  comparing the running times of all MIP models in the evaluation of the small problem instances.

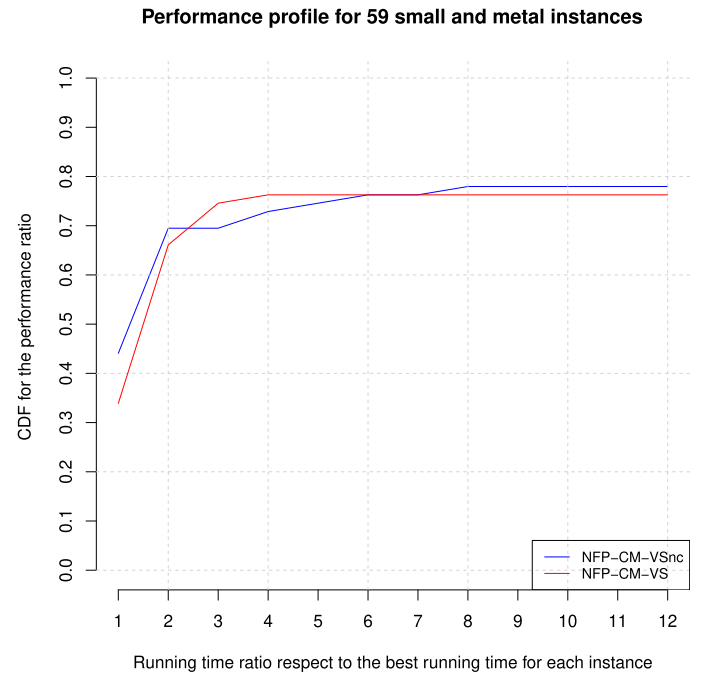


**Fig. 8.** Performance profile (Dolan & Moré, 2002) comparing the running times of the NFP-CM-VSnc and NFP-CM-VS MIP models in the evaluation of the small and metal instances.

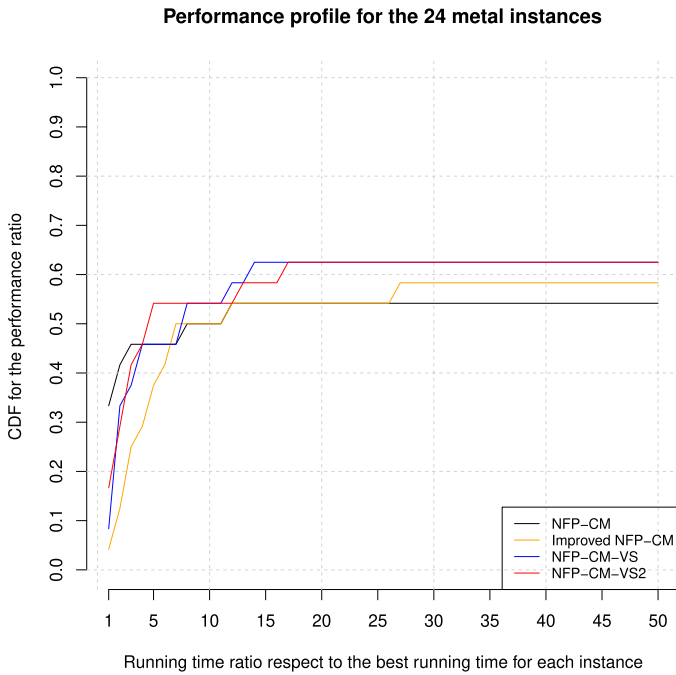


**Fig. 7.** Performance profile (Dolan & Moré, 2002) showing the Cumulative Distribution Function (CDF) for the performance ratio  $r_{\phi,m}$  comparing the running times of all MIP models in the evaluation of the metal problem instances.

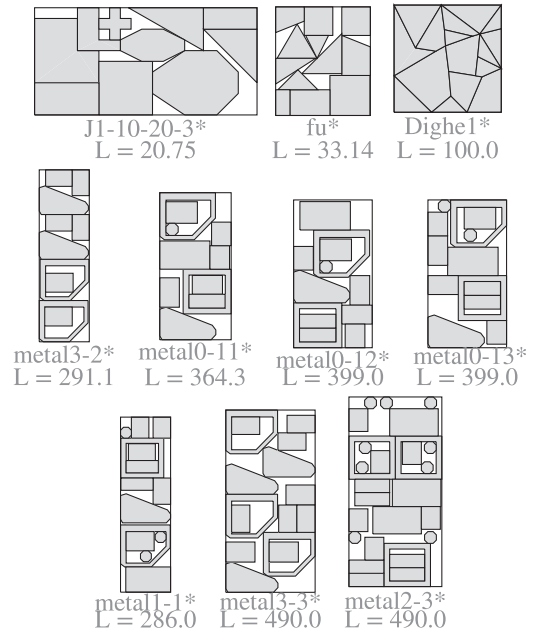


**Fig. 9.** Optimal solutions of small and metal instances solved within 1 hour for the first time (\*) in the literature. Fu and Dighe1 instances were solved within 2 and 5 hours respectively by Alvarez-Valdes et al. (2013, Table 7).

NFP-CM-VS2. Thus, this latter achievement should be attributed to the enhancements in the solver's performance.

The NFP-CM-VS2 model obtains a significantly higher performance ratio than NFP-CM-VS in the small instances, as shown in Fig. 6. Thus, the NFP-CM-VS2 formulation outperforms the NFP-CM-VS one in the small instances, which we attribute to the lower number of constraints of NFP-CM-VS2 regarding NFP-CM-VS.

NFP-CM and Improved NFP-CM models solve 30 small instances, as shown in the last row of Table 2. However, NFP-CM unexpectedly obtains a significantly higher performance ratio than Improved NFP-CM in the small instances, as shown in Fig. 6. We attribute this unexpected result to the fact that NFP-CM requires much fewer binary variables than Improved NFP-CM, as shown in Table 5 above and Table 1 in Appendix A.

## 5.2. Comparison in metal problem instances

The entire family of NFP-CM-VS models obtains a higher performance ratio than the family of state-of-the-art NFP-CM models in the metal problem instances, and significantly outperforms the baseline Improved NFP-CM model, as shown in Fig. 7. Likewise, the NFP-CM-VS and NFP-CM-VS2 models solve the largest number of metal instances up to optimality, 15 instances, as shown in the second last row of Table 3. Looking at Fig. 7, we find that Improved NFP-CM cannot significantly outperform NFP-CM despite its well-founded symmetry-breaking.

Looking at the last row of Table 3, we find that among the metal instances solved up to optimality, NFP-CM is the fastest model in 8 instances, whilst NFP-CM-VS2 is the fastest one in 6, NFP-CM-VS in 3, and Improved NFP-CM in 2. Fig. 9 shows seven new metal instances solved up to optimality within one hour for the first time in the literature, including the *metal2-3* instance with 27 pieces which is the largest one solved up to date in this line of research. However, six of seven instances are solved up to optimality by all models with the exception of *metal3-3*, which is only solved by our NFP-CM-VS and NFP-CM-VS2 models. Thus, this latter achievement should be mainly attributed to the enhancements in the solver's performance, as discussed in Section 5.5.

The NFP-CM-VS2 model obtains a slightly higher performance ratio than NFP-CM-VS in the metal instances, as shown in Fig. 7. Moreover, NFP-CM-VS2 obtains a feasible solution more than NFP-CM-VS, as shown in the results for *metal1-4* in Table 3. Despite its symmetry-breaking, Improved NFP-CM cannot significantly outperform NFP-CM in the metal instances, as shown in Fig. 7. Moreover, NFP-CM obtains feasible solutions for all metal instances.

The entire family of NFP-CM-VS models define a significantly lower number of binary variables than the baseline Improved NFP-CM model in all metal problem instances, as shown in Table 5. The number of binary variables defined by the family of NFP-CM-VS models is roughly 11% lower than those defined by Improved NFP-CM on average, and up to 18% for *metal0-3* instance. Thus, this latter conclusion confirms the contribution of our new variable reductions (34)–(37) to tighten the formulation of our new MIP models. However, NFP-CM model defines the lowest number of variables among all MIP models in all datasets, as shown in Table 5 above and Table 1 in Appendix A.

The NFP-CM-VS2 models require a significantly lower number of constraints than their corresponding NFP-CM-VS model to formulate the same problem, as shown in Table 5. On the other hand, Improved NFP-CM models double on average the number of constraints compared to their corresponding NFP-CM models, as expected from their symmetry-breaking, whilst our new NFP-CM-VS and NFP-CM-VS2 models require a significantly higher number of constraints derived from our new symmetry-breakings and valid cuts, especially the feasibility cuts among three pieces (41).

## 5.3. Comparison in large problem instances

Our new family of MIP models doubles the cases in which it can find feasible solutions compared to the baseline Improved NFP-CM model, as shown in Table 4. However, current state-of-the-art MIP models evaluated here are unable to either solve a large instance up to optimality or obtain feasible solutions for all of them, as shown in Table 4. Moreover, no MIP model raises the initial LB values (21) during its resolution within one hour, which confirms that these problem instances are too large to be solved efficiently by current MIP models and solvers.

The NFP-CM model obtains feasible solutions for 10 of 11 large problem instances, whilst NFP-CM-VS and NFP-CM-VS2 do it for 8, and Improved NFP-CM for only 3, as shown in Table 4.

The NFP-CM model obtains the lowest UB value in 5 large problem instances, whilst NFP-CM-VS2 does it in 3 cases, and NFP-CM-VS and Improved NFP-CM do it in 2, as shown in Table 4. As we noted above, the large problem instances might be biased toward weaker formulations as NFP-CM, which requires much fewer constraints than the rest of the models implementing costly symmetry-breakings and valid cuts. For instance, the stronger formulations provided by our family of NFP-CM-VS models require, on average, up to two orders of magnitude more constraints than the family of NFP-CM models, as shown in Table 5. Thus, the cost of solving LP problems in the B&B nodes is significantly higher for our NFP-CM-VS models than for weaker formulations such as NFP-CM or Improved NFP-CM.

NFP-CM significantly outperforms the baseline Improved NFP-CM model in the large problem instances, as shown in Table 4. Table 4 also shows that Improved NFP-CM fails in obtaining feasible solutions in most of instances and it only outperforms NFP-CM model in one large instance (*Blaz2*). Thus, the symmetry-breaking proposed by the Improved NFP-CM is unable to provide consistent performance results in the large problem instances in comparison with its results in the small ones. Although Improved NFP-CM only doubles the number of constraints of NFP-CM, we conjecture again that these contradictory results might be attributed to the significant impact of increasing the number of constraints in stronger formulations on the computational cost of solving large instances, as pointed out above.

## 5.4. Impact of valid cuts among three pieces

The valid cuts among three pieces implemented by our NFP-CM-VS model significantly improve the performance of the NFP-CM-VSnc model without them (see Table 1). This conclusion can be drawn by comparing the performance profiles shown in Fig. 8 for both former models in the evaluation of the small and metal problem instances. Looking at Fig. 8, you can see that the probability that NFP-CM-VS being at most six times slower than the best-performing model is always higher than the same probability for NFP-CM-VSnc, although this latter model solves up to optimality one small instance more than the former. Thus, these performance improvements can only be attributed to the single difference between both former models: the valid cuts among three pieces defined by inequalities (41).

## 5.5. Impact of technological advances and confirmation of previous results

Our replication of all MIP models evaluated herein allows us to confirm previous results and to draw some conclusions on the impact of the technological advances both in hardware and MIP solvers regarding the results reported by Cherri et al. (2016b) and Rodrigues et al. (2017), in spite of the fact that our experiments are based on Gurobi 9.5 (2022) instead of the CPLEX 12.6 solver used by the former authors.

The Improved NFP-CM baseline cannot significantly outperform the NFP-CM model in the evaluation of small and metal problem instances, as shown in Figs. 6 and 7. This latter conclusion unexpectedly contradicts the common belief on the outperformance of Improved NFP-CM on NFP-CM reported in the literature. However, we note that the Improved NFP-CM model introduced by Rodrigues et al. (2017) was not evaluated in the same set of problem instances as NFP-CM (Cherri et al., 2016b), as has been done here for the first time. On the contrary, Improved NFP-CM was evaluated in a different and more reduced set of instances, which limits the scope of previous conclusions. However, we bridge this minor gap here by independently confirming the achievements of the family of NFP-CM models (Cherri et al., 2016b; Rodrigues et al., 2017).

in a more sound way. Thus, we conjecture that the significant reduction of variables of NFP-CM with regards to Improved NFP-CM and the technological advances of current MIP solvers outweigh the symmetry-breaking of this latter model.

The advances in hardware and MIP solvers significantly contribute to improving the performance of the NFP-CM models. This conclusion can be drawn by looking the results reported in Tables 2–4, and comparing them with those reported by Cherri et al. (2016b). For instance, NFP-CM solves for the first time the *threep3*, *fu*, *J1-10-20-3*, and *Dighe1* instances. However, NFP-CM cannot solve the *threep3w9* instance up to optimality in our experiments, which we attribute to differences in B&C algorithms of the MIP solvers and the implementation of some valid inequalities using SOS-1 type constraints, as detailed in Table 1. This technological improvement is much more noticeable in the case of the metal and large problem instances, as follows. Firstly, NFP-CM solves six metal instances for the first time, as reported in Table 3. And secondly, NFP-CM significantly reduces the terminating gap in unsolved metal instances and all large instances, and it obtains for the first time a feasible solution for the *metal-4*, *metal1-5*, and *Trousers* instances.

The aforementioned technological advances significantly contribute to improving the performance of all MIP models evaluated herein. Ten instances are solved up to optimality within one hour for the first time in the literature, as shown in Tables 2 and 3. For instance, the *fu* instance is solved by all models. Likewise, seven metal instances are solved for the first time in this work, as shown in Table 3, including the *metal2-3* instance with 27 pieces solved by all models. These later conclusions can be drawn by comparing the results in Tables 2 and 3 with that reported in Cherri et al. (2016b).

### 5.6. The new state of the art

The stronger formulations of the problem proposed by our NFP-CM-VS and NFP-CM-VS2 models significantly outperform on average that proposed by the baseline Improved NFP-CM model in all problem instances, as shown by the performance profiles in Figs. 6 and 7, and the results reported for large problem instances in Table 4, which shows that NFP-CM-VS and NFP-CM-VS2 double the number of large instances in which Improved NFP-CM obtains a feasible solution. Our new MIP models obtain these later results despite requiring up to two orders of magnitude more constraints than Improved NFP-CM, as drawn from comparing the number of constraints reported in Table 5.

Our NFP-CM-VS2 model sets the new state-of-the-art among the family of continuous MIP models for nesting and provides a more consistent performance ratio than current state-of-the-art models, as shown in Figs. 6 and 7, and Tables 2–4. Our new family of MIP models outperform the state-of-the-art Improved NFP-CM model in most of the problem instances. However, they only obtain competitive results on large instances compared to NFP-CM.

Although NFP-CM-VS and NFP-CM-VS2 only obtain comparable results to that obtained by the NFP-CM model in terms of the terminating gap in the evaluation of large problem instances, in which NFP-CM has the advantage of having several orders of magnitude fewer constraints than our new NFP-CM-VS models, which significantly reduces the resolution of LP problems during the B&C exploration, the performance ratios reported in Figs. 6 and 7 show that all improvements proposed in the formulation of our NFP-CM-VS models improve the state of the problem. Moreover, NFP-CM-VS and NFP-CM-VS2 show consistent performance results in the full range of problem instances evaluated herein, unlike the Improved NFP-CM model whose performance significantly decreases in the large problem instances.

## 6. Conclusions and future work

We have introduced a new family of continuous MIP models for irregular strip packing with two different formulations, abbreviated to NFP-CM-VS and NFP-CM-VS2, which is based on a new convex decomposition of the feasible regions between convex parts into vertical slices, together with a new family of valid inequalities, symmetry-breakings, and variable eliminations derived from the former geometric decomposition.

Our new family of MIP models outperforms the state-of-the-art family of NFP-CM models (Cherri et al., 2016b; Rodrigues et al., 2017) in small and metal instances, whilst it is only competitive for large ones. Our new NFP-CM-VS models significantly and consistently set the new state of the art of the problem. We show that our new geometric decomposition based on vertical slices outperforms the symmetry-breaking proposed by the Improved NFP-CM model, despite our NFP-CM-VS models significantly increasing the number of constraints of NFP-CM instead of only doubling it as done by Improved NFP-CM. Likewise, we show that our new family of valid cuts among three pieces significantly contribute to the performance gain of our NFP-CM-VS models. Our experiments solve ten open instances up to optimality within one hour for the first time in the literature, including one with 27 pieces.

Another contributions are the introduction of the first confirmatory study on the MIP models and results introduced by Cherri et al. (2016b) and Rodrigues et al. (2017), as well as the first reproducible experimental survey in this line of research, which is based on our software implementation of all models evaluated herein into the same software library and a detailed reproducibility protocol to allow the exact replication of all our models, experiments, and results. Finally, our experiments contradict a prior belief on the outperformance of Improved NFP-CM on NFP-CM in a sound way by replicating all models from scratch and comparing them in the same benchmark reported by Cherri et al. (2016b). However, this latter finding does not contradict the results reported by Rodrigues et al. (2017) because this work does not replicate the same benchmarks of Cherri et al. (2016b), which prevents a direct and fair comparison of both models.

As forthcoming activities, we plan to study the proposal of a matheuristics to improve the performance of our new MIP models, and extend them to admit discrete rotations and industrial constraints. Finally, we also plan to study a formulation of our NFP-CM-VS models based on a horizontal decomposition of the feasible space among pieces.

## Acknowledgments

We are grateful to Luiz Cherri, Ramón Álvarez-Valdés, and Antonio Martínez-Sykora for providing us the problem instances used in their benchmarks, and to Alicia Lara-Clares for testing our reproducibility protocol. Finally, we thank David Pritchard for checking the use of English in our manuscript. This work has been partially supported by the LOG4D (PID2019-108679RB-I00) Project funded by the Spanish Ministry of Science, Innovation, and Universities.

## Appendix A. Raw output data for all models

This appendix introduces all raw output data generated by our experiments not reported herein by lack of room.

## Appendix B. The reproducible experiments on irregular strip-packing

This appendix introduces a detailed reproducibility protocol and dataset (Lastra-Díaz & Ortuño, 2023) providing our raw output data

together with a collection of software and data resources to allow the exact replication of all our experiments and results.

## Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.ejor.2023.08.009.

## References

- Adamowicz, M., & Albano, A. (1976). Nesting two-dimensional shapes in rectangular modules. *Computer-Aided Design and Applications*, 8, 27–33.
- Adamowicz, M., & Albano, A. (1976). A solution of the rectangular cutting-stock problem. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-6, 302–310.
- Agarwal, P. K., Flato, E., & Halperin, D. (2002). Polygon decomposition for efficient construction of Minkowski sums. *Computational Geometry: Theory and Applications*, 21, 39–61.
- Al Theeb, N. A., Hayajneh, M. T., & Jaradat, M. Y. (2021). New strategy to improve the dotted board model for solving two dimensional cutting and packing problems. *Computers & Industrial Engineering*, 159, 107467.
- Albano, A., & Sapuppo, G. (1980). Optimal allocation of two-Dimensional irregular shapes using heuristic search methods. *IEEE Transactions on Systems, Man and Cybernetics*, 10, 242–248.
- Alvarez-Valdés, R., Martínez, A., & Tamarit, J. M. (2013). A branch & bound algorithm for cutting and packing irregularly shaped pieces. *International Journal of Production Economics*, 145, 463–477.
- Álvarez-Valdés, R., Parreño, F., & Tamarit, J. M. (2009). A branch and bound algorithm for the strip packing problem. *OR Spectrum*, 31, 431–459.
- Alves, C., Brás, P., Valério de Carvalho, J., & Pinto, T. (2012). New constructive algorithms for leather nesting in the automotive industry. *Computers & Operations Research*, 39, 1487–1505.
- Amaro Júnior, B., Pinheiro, P. R., & Coelho, P. V. (2017). A parallel biased random-key genetic algorithm with multiple populations applied to irregular strip packing problems. *Mathematical Problems in Engineering*, 2017.
- Annamalai Vasantha, G. V., Jagadeesan, A. P., Corney, J. R., Lynn, A., & Agrawal, A. (2016). Crowdsourcing solutions to 2D irregular strip packing problems from internet workers. *International Journal of Production Research*, 54, 4104–4125.
- Art, R. C., Jr (1966). *An approach to the two dimensional irregular cutting stock problem*. Massachusetts Institute of Technology Ph.d. thesis.
- Atamtürk, A., Nemhauser, G. L., & Savelsbergh, M. (2000). Conflict graphs in solving integer programming problems. *European Journal of Operational Research*, 121(1), 40–55.
- Bennell, J. A., & Oliveira, J. F. (2008). The geometry of nesting problems: A tutorial. *European Journal of Operational Research*, 184, 397–415.
- Bennell, J. A., & Oliveira, J. F. (2009). A tutorial in irregular shape packing problems. *The Journal of the Operational Research Society*, S93–S105.
- Bennell, J. A., Oliveira, J. F., & Wäschler, G. (2013). Cutting and packing. *International Journal of Production Economics*, 145, 449–450.
- Carravilla, M. A., & Ribeiro, C. (2005). CP and MIP in the resolution of hard combinatorial problems: A case study with nesting problems. In *Proceedings of the CSCLP 2005 joint ERCIM/cologNET international workshop on constraint solving and constraint logic programming* (pp. 113–127). Uppsala, Sweden: Citeseer.
- Carravilla, M. A., Ribeiro, C., & Oliveira, J. F. (2003). Solving nesting problems with non-convex polygons by constraint logic programming. *International Transactions in Operational Research: A Journal of The International Federation of Operational Research Societies*, 10, 651–663.
- Cheok, B. T., & Nee, A. Y. C. (1991). Algorithms for nesting of ship/offshore structural plates. *American Society of Mechanical Engineers, Design Engineering Division (Publication) DE*, 32, 221–226.
- Chernov, N., Stoyan, Y., & Romanova, T. (2010). Mathematical model and efficient algorithms for object packing problem. *Computational Geometry: Theory and Applications*, 43, 535–553.
- Chernov, N., Stoyan, Y., Romanova, T., & Pankratov, A. (2012). Phi-functions for 2D objects formed by line segments and circular arcs. *Advances in Operations Research*, 2012.
- Cherri, L. H., Carravilla, M. A., Ribeiro, C., & Toledo, F. M. B. (2019). Optimality in nesting problems: New constraint programming models and a new global constraint for non-overlap. *Operations Research Perspectives*, 6, 100125.
- Cherri, L. H., Carravilla, M. A., & Toledo, F. M. B. (2016). A model-based heuristic for the irregular strip packing problem. *Pesquisa Operacional*, 36, 447–468.
- Cherri, L. H., Cherri, A. C., Carravilla, M. A., Oliveira, J. F., Toledo, F. M. B., & Vianna, A. C. G. (2018). An innovative data structure to handle the geometry of nesting problems. *International Journal of Production Research*, 56, 7085–7102.
- Cherri, L. H., Cherri, A. C., & Soler, E. M. (2018). Mixed integer quadratically-constrained programming model to solve the irregular strip packing problem with continuous rotations. *Journal of Global Optimization*, 72, 89–107.
- Cherri, L. H., Mundim, L. R., Andretta, M., Toledo, F. M. B., Oliveira, J. F., & Carravilla, M. A. (2016). Robust mixed-integer linear programming models for the irregular strip packing problem. *European Journal of Operational Research*, 253, 570–583.
- Codato, G., & Fischetti, M. (2006). Combinatorial benders' cuts for mixed-integer linear programming. *Operations Research*, 54, 756–766.
- Conte, A., Grossi, R., & Marino, A. (2020). Large-scale clique cover of real-world networks. *Information and Computation*, 270, 104464.
- Cuninghame-Green, R. (1989). Geometry, shoemaking and the milk tray problem. *New Scientist*, 123, 50–53.
- Daniels, K., Li, Z., & Milenkovic, V. (1994). Multiple containment methods. Technical report tr-12-94 Harvard Computer Science Group.
- Dean, H. T. (2002). *Minimizing waste in the 2-dimensional cutting stock problem*. Department of Mechanical Engineering, University of Canterbury Christchurch, New Zealand Ph.d. thesis.
- de Berg, M., Van Kreveld, M., Overmars, M., & Schwarzkopf, O. (1997). *Computational geometry: Algorithms and applications*. Springer-Verlag.
- Dighe, R., & Jakiela, M. J. (1995). Solving pattern nesting problems with genetic algorithms employing task decomposition and contact detection. *Evolutionary Computation*, 3, 239–266.
- Dolan, E. D., & Moré, J. J. (2002). Benchmarking optimization software with performance profiles. *Mathematical Programming. A Publication of the Mathematical Programming Society*, 91, 201–213.
- Dowsland, K. A., & Dowsland, W. B. (1992). Packing problems. *European Journal of Operational Research*, 56, 2–14.
- Dowsland, K. A., & Dowsland, W. B. (1995). Solution approaches to irregular nesting problems. *European Journal of Operational Research*, 84, 506–521.
- Dowsland, K. A., Vaid, S., & Dowsland, W. B. (2002). An algorithm for polygon placement using a bottom-left strategy. *European Journal of Operational Research*, 141, 371–381.
- Dyckhoff, H. (1990). A typology of cutting and packing problems. *European Journal of Operational Research*, 44, 145–159.
- Elamvazuthi, I., Kamaruddin, S., & Azmi, M. S. (2009). Automation of nesting and cutting processes of leather furniture production: A case study. *International Journal of Mechanical & Mechatronics Engineering IJMME*, 9.
- Elkeran, A. (2013). A new approach for sheet nesting problem using guided cuckoo search and pairwise clustering. *European Journal of Operational Research*, 231, 757–769.
- Fernández, J., Cánovas, L., & Pelegrín, B. (2000). Algorithms for the decomposition of a polygon into convex polygons. *European Journal of Operational Research*, 121, 330–342.
- Fischetti, M., & Luzzi, I. (2009). Mixed-integer programming models for nesting problems. *Journal of Heuristics*, 15, 201–226.
- Fowler, R. J., Paterson, M. S., & Tanimoto, S. L. (1981). Optimal packing and covering in the plane are NP-complete. *Information Processing Letters*, 12, 133–137.
- Fujita, K., Akagi, S., & Hirokawa, N. (1993). Hybrid approach for optimal nesting using a genetic algorithm and a local minimization algorithm. In *Proceedings of the 19th annual ASME design automation conference* (pp. 477–484). pdfs.semanticscholar.org.
- Gilmore, P. C., & Gomory, R. E. (1965). Multistage cutting stock problems of two and more dimensions. *Operations Research*, 13, 94–120.
- Gomes, A. M., & Oliveira, J. F. (2002). A 2-exchange heuristic for nesting problems. *European Journal of Operational Research*, 141, 359–370.
- Greene, D. H. (1983). The decomposition of polygons into convex parts. In F. Preparata (Ed.), *Computational geometry. In Adv. Comput. Res.: vol. 1* (pp. 235–259). Greenwich, Connecticut (USA): JAI Press.
- Han, W., Bennell, J. A., Zhao, X., & Song, X. (2013). Construction heuristics for two-dimensional irregular shape bin packing with guillotine constraints. *European Journal of Operational Research*, 230, 495–504.
- Heistermann, J., & Lengauer, T. (1995). The nesting problem in the leather manufacturing industry. *Annals of Operations Research*.
- Hopper, E. (2000). *Two-dimensional packing utilising evolutionary algorithms and other meta-heuristic methods*. Cardiff University Ph.d. thesis.
- Hopper, E., & Turton, B. C. H. (2001). A review of the application of meta-heuristic algorithms to 2D strip packing problems. *Artificial Intelligence Review*, 16, 257–300.
- 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.
- Jakobs, S. (1996). On genetic algorithms for the packing of polygons. *European Journal of Operational Research*, 88, 165–181.
- Jones, D. R. (2014). A fully general, exact algorithm for nesting irregular shapes. *Journal of Global Optimization*, 59, 367–404.
- Kallrath, J. (2009). Cutting circles and polygons from area-minimizing rectangles. *Journal of Global Optimization*, 43, 299–328.
- Kimms, A., & Király, H. (2022). An extended model formulation for the two-dimensional irregular strip packing problem considering general industry-relevant aspects. *European Journal of Operational Research*, 306(3), 1202–1218.
- Lastra-Díaz, J. J., & Ortuño, M. T. (2023). Reproducibility dataset for a new mixed-integer programming model for irregular strip packing based on vertical slices. *Mendeley Data*, V3, doi:10.17632/m8nzsk5c9v3.
- Leao, A. A. S., & Toledo, F. M. B. (2021). Enhanced solution for the irregular strip packing problem: Valid inequalities and branching priorities. In *LIII simpósio brasileiro de pesquisa operacional joao pessoa* (pp. 1–12).
- Leao, A. A. S., Toledo, F. M. B., Oliveira, J. F., & Carravilla, M. A. (2016). A semi-continuous MIP model for the irregular strip packing problem. *International Journal of Production Research*, 54, 712–721.
- Leao, A. A. S., Toledo, F. M. B., Oliveira, J. F., Carravilla, M. A., & Álvarez-Valdés, R. (2020). Irregular packing problems: A review of mathematical models. *European Journal of Operational Research*, 282, 803–822.

- Li, Z. (1994). *Compaction algorithms for non-convex polygons and their applications*. Harvard University Ph.D. thesis. <http://nrs.harvard.edu/urn-3:HUL.InstRepos:25619464>.
- Li, Z., & Milenkovic, V. (1993). A compaction algorithm for non-convex polygons and its application. In *Proceedings of the ninth annual symposium on computational geometry* (pp. 153–162). New York, NY, USA: ACM.
- Li, Z., & Milenkovic, V. (1995). Compaction and separation algorithms for non-convex polygons and their applications. *European Journal of Operational Research*, 84, 539–561.
- Lodi, A., Martello, S., & Vigo, D. (2002). Recent advances on two-dimensional bin packing problems. *Discrete Applied Mathematics*, 123, 379–396.
- Mifsud, C. J. (1963). Algorithm 154: Combination in lexicographical order. *Communications of the ACM*, 6, 103.
- Milenkovic, V., Daniels, K., & Li, Z. (1991). Automatic marker making. In *Proceedings of the third canadian conference on computational geometry* (pp. 243–246). nereida.deioc.ulles.
- Moore, R. E. (1966). *Interval analysis*. Prentice Hall.
- Moore, R. E., Kearfott, R. B., & Cloud, M. J. (2009). *Introduction to interval analysis: Vol. 2. Society for Industrial and Applied Mathematics (SIAM)*.
- Mundim, L. R., Andretta, M., Carravilla, M. A., & Oliveira, J. F. (2018). A general heuristic for two-dimensional nesting problems with limited-size containers. *International Journal of Production Research*, 56, 709–732.
- Mundim, L. R., Andretta, M., & de Queiroz, T. A. (2017). A biased random key genetic algorithm for open dimension nesting problems using no-fit raster. *Expert Systems with Applications*, 81, 358–371.
- Oliveira, J., Gomes, A., & Ferreira, J. (2000). TOPOS—A new constructive algorithm for nesting problems. *OR-Spektrum*, 22, 263–284.
- Pankratov, A., Romanova, T., Shekhovtsov, S., Grebennik, I., & Pankratova, J. (2020). Packing irregular polygons using quasi phi-functions. In *2020 10th International conference on advanced computer information technologies (ACIT)* (pp. 1–5).
- Peralta, J., Andretta, M., & Oliveira, J. F. (2018). Solving irregular strip packing problems with free rotations using separation lines. *Pesquisa Operacional*, 38, 195–214.
- Pinheiro, P. R., Amaro Júnior, B., & Saraiva, R. D. (2016). A random-key genetic algorithm for solving the nesting problem. *International Journal of Computer Integrated Manufacturing*, 29, 1159–1165.
- Puri, A. (2013). Efficacy of pattern making software in product development. *International Journal of Advanced Quality Management*, 1, 21–39.
- Ribeiro, C., & Carravilla, M. A. (2004). A global constraint for nesting problems. In *Integration of AI and OR techniques in constraint programming for combinatorial optimization problems* (pp. 256–270). Springer Berlin Heidelberg.
- Ribeiro, C., & Carravilla, M. A. (2009). A global constraint for nesting problems. *Artificial Intelligence Review*, 30, 99.
- Ribeiro, C., Carravilla, M. A., & Oliveira, J. F. (1999). Applying constraint logic programming to the resolution of nesting problems. *Pesquisa Operacional*, 19, 239–247.
- Riff, M. C., Bonnaire, X., & Neveu, B. (2009). A revision of recent approaches for two-dimensional strip-packing problems. *Engineering Applications of Artificial Intelligence*, 22, 823–827.
- Rocha, P., Gomes, A. M., Rodrigues, R., Toledo, F. M. B., & Andretta, M. (2016). Constraint aggregation in non-linear programming models for nesting problems. In *Computational management science* (pp. 175–180). Springer International Publishing.
- Rocha, P., Rodrigues, R., Gomes, A. M., Toledo, F. M. B., & Andretta, M. (2014). Circle covering representation for nesting problems with continuous rotations. *IFAC Proceedings Volumes. Proc. of the 19th IFAC World Congress*, 47, 5235–5240.
- Rodrigues, M. O., Cherri, L. H., & Mundim, L. R. (2017). MIP models for the irregular strip packing problem: New symmetry breaking constraints. *ITM Web of Conferences*, 14, 00005.
- Rodrigues, M. O., & Toledo, F. M. B. (2017). A clique covering MIP model for the irregular strip packing problem. *Computers & Operations Research*, 87, 221–234.
- Rodrigues de Souza, L., & Andretta, M. (2020). Modelo de programação estocástica para um problema de corte de itens irregulares. In *LII simpósio brasileiro de pesquisa operacional*.
- Rodrigues de Souza, L., & Andretta, M. (2022). A branchandcut algorithm for the irregular strip packing problem with uncertain demands. *International transactions in operational research: A Journal of The International Federation of Operational Research Societies*.
- Romanova, T., Stoyan, Y., Pankratov, A., Litvinchev, I., & Marmolejo, J. A. (2020). Decomposition algorithm for irregular placement problems. In *Intelligent computing and optimization* (pp. 214–221). Springer International Publishing.
- Santoro, M. C., & Lemos, F. K. (2015). Irregular packing: MILP model based on a polygonal enclosure. *Annals of Operations Research*, 235, 693–707.
- Sato, A. K., Martins, T. C., Gomes, A. M., & Tsuzuki, M. S. G. (2019). Raster penetration map applied to the irregular packing problem. *European Journal of Operational Research*, 279, 657–671.
- Sato, A. K., Martins, T. d. C., & Tsuzuki, M. d. S. G. (2016). A pairwise exact placement algorithm for the irregular nesting problem. *International Journal of Computer Integrated Manufacturing*, 29, 1177–1189.
- Sato, A. K., de Sales Guerra Tsuzuki, M., de Castro Martins, T., & Gomes, A. M. (2016). Study of the grid size impact on a raster based strip packing problem solution. *IFAC-PapersOnLine*, 49, 143–148.
- Sawaya, N. W., & Grossmann, I. E. (2005). A cutting plane method for solving linear generalized disjunctive programming problems. *Computers & Chemical Engineering*, 29, 1891–1913.
- Stoyan, Y., Pankratov, A., & Romanova, T. (2016). Cutting and packing problems for irregular objects with continuous rotations: Mathematical modelling and non-linear optimization. *The Journal of the Operational Research Society*, 67, 786–800.
- Stoyan, Y., Pankratov, A., & Romanova, T. (2016). Quasi-phi-functions and optimal packing of ellipses. *Journal of Global Optimization*, 65, 283–307.
- Stoyan, Y., Pankratov, A., & Romanova, T. (2017). Placement problems for irregular objects: Mathematical modeling, optimization and applications. In S. Butenko, P. M. Pardalos, & V. Shylo (Eds.), *Optimization methods and applications: In honor of Ivan V. Sergienko's 80th birthday* (pp. 521–559). Cham: Springer International Publishing.
- Stoyan, Y., & Romanova, T. (2013). Mathematical models of placement optimisation: Two- and three-dimensional problems and applications. In G. Fasano, & J. D. Pintér (Eds.), *Modeling and optimization in space engineering* (pp. 363–388). New York, NY: Springer New York.
- Stoyan, Y., Romanova, T., Pankratov, A., & Chugay, A. (2015). Optimized object packings using quasi-phi-functions. In G. Fasano, & J. D. Pintér (Eds.), *Optimized packings with applications* (pp. 265–293). Cham: Springer International Publishing.
- Stoyan, Y. G., Novozhilova, M. V., & Kartashov, A. V. (1996). Mathematical model and method of searching for a local extremum for the non-convex oriented polygons allocation problem. *European Journal of Operational Research*, 92, 193–210.
- Stoyan, Y. G., Zlotnik, M. V., & Chugay, A. M. (2012). Solving an optimization packing problem of circles and non-convex polygons with rotations into a multiply connected region. *The Journal of the Operational Research Society*, 63, 379–391.
- Sweeney, P. E., & Paternoster, E. R. (1992). Cutting and packing problems: A categorized, application-orientated research bibliography. *The Journal of the Operational Research Society*, 43, 691–706.
- Toledo, F. M. B., Carravilla, M. A., Ribeiro, C., Oliveira, J. F., & Gomes, A. M. (2013). The dotted-board model: A new MIP model for nesting irregular shapes. *International Journal of Production Economics*, 145, 478–487.
- Wang, A., Hanselman, C. L., & Gounaris, C. E. (2018). A customized branch-and-bound approach for irregular shape nesting. *J. Global Optimiz.*, 71, 935–955.
- Wang, P. Y., & Wäscher, G. (2002). Cutting and packing. *European Journal of Operational Research*, 141, 239–240.
- Wäscher, G., Haußner, H., & Schumann, H. (2007). An improved typology of cutting and packing problems. *European Journal of Operational Research*, 183, 1109–1130.
- Whelan, P. F., & Batchelor, B. G. (1993). Automated packing systems: Review of industrial implementations. *Machine Vision and Applications*.