

**Logic-based Benders reformulations for
integrated process configuration and
production planning problems**

K. Pérez Martínez,
Y. Adulyasak, R. Jans

G-2019-73

October 2019

La collection *Les Cahiers du GERAD* est constituée des travaux de recherche menés par nos membres. La plupart de ces documents de travail a été soumis à des revues avec comité de révision. Lorsqu'un document est accepté et publié, le pdf original est retiré si c'est nécessaire et un lien vers l'article publié est ajouté.

Citation suggérée : K. Pérez Martínez, Y. Adulyasak, R. Jans (Octobre 2019). Logic-based Benders reformulations for integrated process configuration and production planning problems, Rapport technique, Les Cahiers du GERAD G-2019-73, GERAD, HEC Montréal, Canada.

Avant de citer ce rapport technique, veuillez visiter notre site Web (<https://www.gerad.ca/fr/papers/G-2019-73>) afin de mettre à jour vos données de référence, s'il a été publié dans une revue scientifique.

The series *Les Cahiers du GERAD* consists of working papers carried out by our members. Most of these pre-prints have been submitted to peer-reviewed journals. When accepted and published, if necessary, the original pdf is removed and a link to the published article is added.

Suggested citation: K. Pérez Martínez, Y. Adulyasak, R. Jans (October 2019). Logic-based Benders reformulations for integrated process configuration and production planning problems, Technical report, Les Cahiers du GERAD G-2019-73, GERAD, HEC Montréal, Canada.

Before citing this technical report, please visit our website (<https://www.gerad.ca/en/papers/G-2019-73>) to update your reference data, if it has been published in a scientific journal.

La publication de ces rapports de recherche est rendue possible grâce au soutien de HEC Montréal, Polytechnique Montréal, Université McGill, Université du Québec à Montréal, ainsi que du Fonds de recherche du Québec – Nature et technologies.

Dépôt légal – Bibliothèque et Archives nationales du Québec, 2019
– Bibliothèque et Archives Canada, 2019

The publication of these research reports is made possible thanks to the support of HEC Montréal, Polytechnique Montréal, McGill University, Université du Québec à Montréal, as well as the Fonds de recherche du Québec – Nature et technologies.

Legal deposit – Bibliothèque et Archives nationales du Québec, 2019
– Library and Archives Canada, 2019

GERAD HEC Montréal
3000, chemin de la Côte-Sainte-Catherine
Montréal (Québec) Canada H3T 2A7

Tél. : 514 340-6053
Télec. : 514 340-5665
info@gerad.ca
www.gerad.ca

Logic-based Benders reformulations for integrated process configuration and production planning problems

Karim Pérez Martínez

Yossiri Adulyasak

Raf Jans

*GERAD & Department of Logistics and Operations
Management, HEC Montréal, Montréal (Québec),
Canada, H3T 2A7*

karim.perez-martinez@hec.ca

yossiri.adulyasak@hec.ca

raf.jans@hec.ca

October 2019

Les Cahiers du GERAD

G–2019–73

Copyright © 2019 GERAD, Pérez Martínez, Adulyasak, Jans

Les textes publiés dans la série des rapports de recherche *Les Cahiers du GERAD* n'engagent que la responsabilité de leurs auteurs. Les auteurs conservent leur droit d'auteur et leurs droits moraux sur leurs publications et les utilisateurs s'engagent à reconnaître et respecter les exigences légales associées à ces droits. Ainsi, les utilisateurs:

- Peuvent télécharger et imprimer une copie de toute publication du portail public aux fins d'étude ou de recherche privée;
- Ne peuvent pas distribuer le matériel ou l'utiliser pour une activité à but lucratif ou pour un gain commercial;
- Peuvent distribuer gratuitement l'URL identifiant la publication.

Si vous pensez que ce document enfreint le droit d'auteur, contactez-nous en fournissant des détails. Nous supprimerons immédiatement l'accès au travail et enquêterons sur votre demande.

The authors are exclusively responsible for the content of their research papers published in the series *Les Cahiers du GERAD*. Copyright and moral rights for the publications are retained by the authors and the users must commit themselves to recognize and abide the legal requirements associated with these rights. Thus, users:

- May download and print one copy of any publication from the public portal for the purpose of private study or research;
- May not further distribute the material or use it for any profit-making activity or commercial gain;
- May freely distribute the URL identifying the publication.

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Abstract: We propose a generalized decomposition approach for production planning problems with process configuration decisions. These problems appear in contexts where the machines are set up according to specific patterns, templates, or, in general, process configurations that allow to simultaneously produce products of different types. The problem consists of determining feasible process configurations for the machines and the production level for each used configuration to fulfill the given demand at the minimum total cost. The proposed approaches make use of logic-based Benders reformulations, which decompose the original problem into a master problem, where the configurations are determined, and a set of subproblems, where production planning decisions are determined. These reformulations can either be solved using a cutting plane algorithm or a branch-and-check algorithm. Reformulation enhancements through logic-based inequalities generated for subsets of products with common characteristics are proposed. Computational experiments on applications from the literature in the steel industry and the printing industry are carried out. Results show that the proposed methods find optimal solutions much faster than the benchmark approaches, and have a superior performance in terms of the number of instances optimally solved and solutions quality.

Keywords: Logic-based Benders decomposition, branch-and-check, integrated production planning

Acknowledgments: This research has been funded by GERAD, NSERC grants 2014-03849 and 2016-05822, and the Chair in Supply Chain Operations Planning at HEC Montréal. We also thank the authors of the papers which introduce the applications tested here for making their data sets available.

1 Introduction

Many manufacturing environments are designed to flexibly produce different product types simultaneously to achieve an efficient use of the production equipment. In these contexts, machines are set up according to process configurations which determine the specific combination of different products to produce at the same time. The produced amounts depend on the production level for each configuration. The setup operations, which consist of changing the configurations on the machines, can be costly and time-consuming. Production environments with these features appear, for instance, in the printing industry (Yiu et al., 2007; Tuytens and Vandaele, 2014; Baumann et al., 2015), where machines are set up using printing plates with different designs to be printed simultaneously; the apparel industry (Degraeve and Vandebroek, 1998; Degraeve et al., 2002; Martens, 2004), where templates with different stencils are used to cut different pieces of clothes at the same time; and the molded packaging industry (Martínez et al., 2019), where a set of different molds is attached to the machines to produce packages of different shapes simultaneously.

We address the integrated production planning problem that occurs in the described manufacturing contexts. The problem includes process configuration decisions, i.e., determining feasible configurations for the machines, and production planning decisions, i.e., deciding the production level of each configuration, in order to fulfill the demand at the minimum total cost. This class of problems is complex to solve due to the particularities of each decision level and the interaction between these two decisions. Firstly, various technical constraints are typically considered to ensure that the configurations are feasible in practice. Moreover, the number of possible configurations might increase exponentially with the number of product types and the size of the machines. Secondly, at the same time as the configurations are determined, the production level for each one of them must be decided (e.g., the number of times that each configuration is used) to obtain the produced quantities. The integration of these decisions typically leads to non-linear formulations (Hajizadeh and Lee, 2007; Degraeve and Vandebroek, 1998), as the produced amounts are determined as the multiplication of decision variables related to both the configurations used and their production levels.

The study of this problem in a general context is hence motivated by its complexity and wide applicability in different industries. Moreover, there is still a lack of efficient exact algorithms for these challenging problems. This study aims to fill the important gap in this area by proposing a solution framework based on the logic-based Benders decomposition (LBBD) technique which can be generalized to tackle various applications of this problem.

The main contributions of this paper are threefold. First, we propose exact solution methods to address a variety of production planning problems which integrate process configuration decisions. The proposed approaches use a general decomposition framework and two exact algorithms to find optimal solutions. Second, we propose reformulation enhancements to the framework through logic-based inequalities which take advantage of the possibility to group products according to common characteristics derived from the input data. Third, we present extensive computational experiments to analyze the performance of the decomposition framework, reformulation enhancements, and algorithms to solve the problems in different practical contexts. We apply our general framework on applications in the steel industry and the printing industry as described in the literature, and show that the proposed methods outperform the benchmark approaches for these problems.

The remainder of this paper is structured as follows. The next section reviews the literature regarding LBBD methods. Section 3 presents a general description of the problem, the decomposition framework, and the reformulation enhancements. Section 4 presents the applications of the proposed methods to the problems in the steel industry and the printing industry, and the computational results. Finally, Section 5 presents concluding remarks.

2 Literature review

Logic-based Benders decomposition is an extension of the Benders decomposition methods, where the formulation of the Benders cuts is not limited to solve the dual linear programs of the subproblems. Similar to classical Benders decomposition, LBBD assigns values to the complicating variables in the master problem and finds the best solution consistent with these values. Instead of solving the dual of the subproblems that remain when the complicating variables take fixed values, LBBD solves an inference dual, where a proof of optimality within an appropriated logical formalism is derived based on the fixed values of some of the variables and the constraints of the original problem. Logic-based Benders provides no standard scheme to generate Benders cuts so that they must be devised specifically for each problem class (Hooker and Ottosson, 2003). There are two common implementations of the LBBD: the original LBBD implementation, which can be seen as a cutting plane approach, and the branch-and-check implementation (B&Ch), where the cuts are generated and added during the branch-and-bound process.

The flexibility of LBBD provides an advantage, as it can be applied to any problem without requiring a specific partition for the integer and continuous variables to define the master problem and the subproblems. It also allows to use hybrid implementations which make use of different purpose solvers, such as Constraint Programming (CP), and Mixed Integer Programming (MIP) solvers. Many LBBD algorithms have been developed to solve a wide range of problems such as production planning problems (Hooker, 2007; Tran et al., 2016), location problems (Fazel-Zarandi and Beck, 2012; Fazel-Zarandi et al., 2013), transportation (Riedler and Raidl, 2018), and home health care operations planning (Riise et al., 2016; Roshanaei et al., 2017a,b). However, various challenges to achieve efficient implementations have been pointed out in the literature, such as the importance of having a tight relaxation of the subproblems in the master problem, the frequency of solving the subproblems, and the formulation of strong Benders' cuts (Hooker, 2007; Thorsteinsson, 2001; Roshanaei et al., 2017a).

Table 1 presents a review of related works which use LBBD methods to solve operations planning problems. This table specifies the related studies, the applications studied, the implementation approaches, the solvers used, and the enhancement strategies proposed in each work as an attempt to improve the performance of the developed LBBD methods.

Table 1: Studies in the literature regarding LBBD and B&Ch algorithms

Papers	Application(s)	Implementation		Solvers		Enhancements
		LBBD	B&Ch	MIP	CP	
Thorsteinsson (2001)	Two scheduling problems with dissimilar parallel machines and the capacitated vehicle routing problem with times windows.		✓	✓	✓	
Hooker (2007)	Parallel machine scheduling problems.	✓		✓	✓	A greedy algorithm to support the cuts formulation
Beck (2010)	Three parallel machine scheduling problems and a location-allocation problem.	✓	✓	✓	✓	
Hamdi and Loukil (2013)	A permutation flowshop scheduling problem with time lags.	✓		✓	✓	
Wheatley et al. (2015)	Inventory-location problem with service constraints.	✓		✓		A restrict-and-decompose algorithm (RDA)
Riise et al. (2016)	An outpatient scheduling problem.	✓		✓		Valid inequalities, multi-stage LBBD
Tran et al. (2016)	A parallel machine scheduling problem with setups.	✓	✓	✓		TSP specialized solver to solve the subproblems
Delorme et al. (2017)	Orthogonal stock cutting problems.		✓	✓	✓	
Martínez et al. (2019)	An integrated lot-sizing and scheduling problem in the molded pulp industry.		✓	✓		

In general, most of the research on LBBD methods target variants of the classical scheduling problems. In these decomposition frameworks, the master problem assigns tasks to resources, whereas the subproblem determines a feasible sequencing of the tasks given their allocation to resources. The master problem is solved using an MIP solver, whereas the subproblems are typically solved using CP techniques (Thorsteinsson, 2001; Beck, 2010).

The studies in Table 1 also present discussions about the challenges to achieve efficient LBBD algorithms with respect to the generation of stronger Benders cuts, including a tight relaxation of the subproblem in the master problem, the use of specialized techniques to solve the master problem and subproblems, and the timing to add the Benders cuts (i.e., the choice between LBBD and B&Ch implementations). For instance, Beck (2010) presents an empirical comparison between the LBBD and the B&Ch implementations, which shows that B&Ch algorithms can lead to a significant improvement over the LBBD when the master problem is significantly more difficult to solve than the subproblems. Comparisons between these two implementations presented in Tran et al. (2016) also highlight the superiority of B&Ch implementations over the LBBD on more complex problems. Strategies to solve more efficiently the LBBD reformulations are proposed by Tran et al. (2016), who use a TSP specialized solver for the subproblems, and Riise et al. (2016), who propose a 3-level framework where the master problem is solved using a LBBD algorithm.

Our methodological contributions with respect to this literature relate to the development of a general logic-based Benders decomposition framework that can be applied to a variety of practical problems which combine process configuration and production planning decisions. Our decomposition framework is applied to this class of problems which typically include non-linear constraints. Moreover, we propose reformulation enhancements to the general framework where logic-based inequalities are derived based on the possibility to group products with common characteristics in the input data. These inequalities are added dynamically to the problem in order to strengthen the relaxation of the subproblem considered in the master. We also present an empirical analysis of the performance of the LBBD and B&Ch implementations.

3 General description and notation

Section 3.1 provides the general notation for integrated process configuration and production planning problems. Sections 3.2 and 3.3 present the proposed logic-based Benders reformulations, and Section 3.4 shortly describes the solution algorithms.

3.1 The integrated process configuration and production planning problem

We consider integrated production planning problems that can be described using a general model (1)–(6). Variable vector $\mathbf{x}$ represents the configuration decisions which determine the configurations to be used in the machines, i.e., the combination of different products produced at the same time. In a more general case, these variables also determine the sequence in which the configurations should be scheduled. Variable vector $\mathbf{y}$ represents the production planning decisions, which determine the production levels of each configuration and other planning decisions, such as inventory levels and overproduction, among others. Variable vector $\mathbf{q}$ represents the production quantities for each product, which are determined in terms of the variables for the used configurations and their production levels.

$$\text{Objective function: } \min \mathbf{c}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} \quad (1)$$

$$\text{Process configuration constraints: } \mathbf{Ax} \geq \mathbf{a} \quad (2)$$

$$\text{Linking constraints: } \mathbf{Bx} + \mathbf{Cy} \geq \mathbf{b} \quad (3)$$

$$\text{Production quantities constraints: } \mathbf{q} = f(\mathbf{x}, \mathbf{y}) \quad (4)$$

$$\text{Production planning constraints: } \mathbf{Dy} + \mathbf{Eq} \geq \mathbf{e} \quad (5)$$

$$\text{Domain of the variables: } \mathbf{x} \in \mathbb{B}^{|\mathcal{A}|}; \mathbf{y} \in \mathbf{Y}; \mathbf{q} \geq \mathbf{0} \quad (6)$$

The objective function (1) minimizes the total cost of the configuration and production planning. The configuration costs include setup costs, whereas the total production planning costs can include costs associated to the production level for each configuration, overproduction, among others. Constraints (2) are related to the configuration decisions and ensure that the configurations used satisfy the technical conditions of the manufacturing environment. Constraints (3) link the production level of each configuration with the setup variables. Constraints (4) compute the production quantities. The function $f(\mathbf{x}, \mathbf{y})$ is typically non-linear, considering that the production amounts are computed as the product of the variables for the configuration decisions and the production level of each configuration. Some examples on how this function is defined are presented in Section 4 for the applications in the steel industry and the printing industry. Constraints (5) are related to the production planning decisions, which include demand fulfillment and other production constraints depending on the application. Constraints (6) define the domain of the variables as follows: $\mathbf{x}$ is a $|\mathcal{A}|$ -dimensional vector of binary variables; $\mathbf{y} \in \mathbf{Y}$, where $\mathbf{Y} = \{(\mathbf{y}_1, \mathbf{y}_2) \in \mathbb{R}_+^{|\mathcal{B}|} \times \mathbb{Z}_+^{|\mathcal{C}|}\}$, considering that the production planning decisions can involve both continuous and integer variables; and $\mathbf{q}$ are non-negative variables.

3.2 Logic-based Benders decomposition framework

This framework makes use of the logic-based Benders decomposition technique to reformulate problems of the form (1)–(6). It consists of a master problem (MP1), a set of subproblems (SPs), and the logic-based Benders cuts (BCs). MP1 is a relaxation of the original formulation (1)–(6) where the complicating non-linear constraints (4) which compute the production quantities are relaxed. The SPs provide bounding functions for the production quantities given fixed values for the configuration variables. The BCs are valid inequalities added to the master problem to impose the bounds derived in the SPs.

MP1 can be formulated as the problem of minimizing the objective function (1) subject to constraints (2)–(3) and (5)–(7). Constraints (7) are a relaxation of the SPs included in the master to approximate the production quantities. Note that an optimal solution of MP1 is a valid lower bound for the original problem.

$$\text{Production quantities approximation: } \mathbf{q} \geq \mathbf{F}\mathbf{x} + \mathbf{G}\mathbf{y} \quad (7)$$

The SPs use fixed values for the configurations variables $\bar{\mathbf{x}}$ to provide a linear approximation function $\beta_{\bar{\mathbf{x}}}(\mathbf{x}, \mathbf{y})$ for the production quantities. Function $\beta_{\bar{\mathbf{x}}}$ must provide valid bounds on the production quantities for any given value $(\mathbf{x}, \mathbf{y})$ which satisfies the feasible space of MP1 and, in particular, this function equals the correct value for the production quantities according the original problem when the configuration variables and production planning variables take the same value as $\bar{\mathbf{x}}$ and $\bar{\mathbf{y}}$, respectively (i.e., $\beta_{\bar{\mathbf{x}}}(\bar{\mathbf{x}}, \bar{\mathbf{y}}) = f(\bar{\mathbf{x}}, \bar{\mathbf{y}})$).

The BCs (8) impose the bounds derived in the SPs on the production quantity variables, where $\mathcal{H}$ (indexed by h) is the set of feasible configurations that can be used and $\bar{\mathbf{x}}^h$ represents the corresponding values for the configurations variables. These logic-based Benders cuts are added dynamically to MP1 according to the instructions of the solution algorithm in order to avoid the complete enumeration of all possible configurations $\mathcal{H}$.

$$\text{Logic-based Benders cuts (BCs): } \mathbf{q} \geq \beta_{\bar{\mathbf{x}}^h}(\mathbf{x}, \mathbf{y}) \quad \forall h \in \mathcal{H} \quad (8)$$

The way in which function $\beta_{\bar{\mathbf{x}}^h}$ is defined varies from one context to another. However, we present inequalities (9) and (10) as a more explicit form for the BCs (8), where $\theta_{\bar{\mathbf{x}}^h}$ is a linear function which computes the production quantity of each product using the fixed values $\bar{\mathbf{x}}^h$, and parameter M can be computed as an upper bound on the production quantities or as the difference on the production quantities w.r.t. function $\theta_{\bar{\mathbf{x}}^h}$ when variables x take values different from $\bar{\mathbf{x}}^h$. The logic of these cuts is such that (9) and (10) give an upper and lower bound on the production quantities, respectively, and both cuts give $\mathbf{q} = f(\bar{\mathbf{x}}^h, \bar{\mathbf{y}})$ when the configuration variables $\mathbf{x}$ are fixed to the values $\bar{\mathbf{x}}^h$ and the

production planning variables $\mathbf{y}$ are fixed to a given value $\bar{\mathbf{y}}$. Generally, only one of these sets of cuts is enough to ensure that variables $\mathbf{q}$ take the correct values according to the original problem.

$$\mathbf{q} \leq \theta_{\bar{\mathbf{x}}^h}(\mathbf{x}, \mathbf{y}) + M \sum_{\substack{j \in \mathcal{A}: \\ \bar{x}_j^h = 1}} (1 - x_j) + M \sum_{\substack{j \in \mathcal{A}: \\ \bar{x}_j^h = 0}} x_j \quad \forall h \in \mathcal{H} \quad (9)$$

$$\mathbf{q} \geq \theta_{\bar{\mathbf{x}}^h}(\mathbf{x}, \mathbf{y}) - M \sum_{\substack{j \in \mathcal{A}: \\ \bar{x}_j^h = 1}} (1 - x_j) - M \sum_{\substack{j \in \mathcal{A}: \\ \bar{x}_j^h = 0}} x_j \quad \forall h \in \mathcal{H} \quad (10)$$

3.3 Enhanced reformulation through logic-based subset inequalities

We introduce the logic-based subset inequalities (SIs) proposed for the general decomposition framework. These inequalities are specifically formulated to strengthen the bounds on the production quantities by considering a particular subset of products with common characteristics within a certain configuration, rather than the entire set of the products in the configuration at a time. In order to do this, the complete set of products is classified into groups according to common characteristics derived from input data (e.g., products of the same size or products with similar demand). We make use of the new logical variables $\mathbf{z} \in \mathbb{B}^{|\mathcal{D}|}$, which are linked to the configuration variables $\mathbf{x}$ by the following constraints.

$$\text{Linking constraints for the new variables: } \mathbf{H}\mathbf{x} + \mathbf{I}\mathbf{z} \geq \mathbf{g} \quad (11)$$

The master problem (MP2) in this enhanced reformulation is defined as MP1, constraints (11), and the domain of the new logical variables $\mathbf{z}$. Constraints (11) link the configuration variables in such a way that variables $\mathbf{z}$ equal 1 if a given group of products appears in a certain configuration, 0 otherwise. Note that, given a configuration $h \in \mathcal{H}$ which specifies the combination of products produced at the same time, the combination of different groups in such configuration $\bar{\mathbf{z}}^h$ can therefore be determined.

The SIs (12) are derived based on the fixed values $\bar{\mathbf{z}}^h$. The logic of these inequalities is such that, if a specific group of products appears in a certain configuration (i.e., a subset $\mathcal{D}' \subseteq \mathcal{D}$ of the new variables equals 1), then a stronger bound $\mathbf{q} \geq \alpha_{\bar{\mathbf{z}}^h}(\mathbf{x}, \mathbf{y})$ can be imposed on the production quantities, where $\alpha_{\bar{\mathbf{z}}^h}$ is a linear function which provides an approximation of the production quantities based on the values $\bar{\mathbf{z}}^h$. The SIs should provide stronger bounds than the subproblems relaxation (7) in order to achieve an improved performance. These inequalities are added dynamically to MP2 at the same time as the BCs.

$$\text{Logic-based subset inequalities (SIs): } \mathbf{q} \geq \alpha_{\bar{\mathbf{z}}^h}(\mathbf{x}, \mathbf{y}) - M \sum_{\substack{g \in \mathcal{D}': \\ \bar{z}_g^h = 1}} (1 - z_g) \quad \forall h \in \mathcal{H} \quad (12)$$

3.4 LBBD implementations

We present an LBBD and a branch-and-check algorithm to solve the proposed reformulations. Both algorithms converge to optimality after a finite number of steps if the Benders cuts provide a valid lower bound on the production quantities variables $\mathbf{q}$ for any feasible $(\mathbf{x}, \mathbf{y})$, and make $\mathbf{q} = f(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ when $\mathbf{x}$ and $\mathbf{y}$ are fixed to given values $\bar{\mathbf{x}}$ and $\bar{\mathbf{y}}$, respectively. We refer to the description of these implementations in Hooker and Ottosson (2003); Hooker (2007), and Thorsteinsson (2001) for further details.

The LBBD algorithm is implemented as a cutting plane approach which iterates between solving the MP and the SPs until a globally optimal solution is found and proved. At each iteration, the MP is solved to optimality, then the SPs compute the correct values for the production amounts $f(\bar{\mathbf{x}}, \bar{\mathbf{y}})$, where $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ are the values of variables $(\mathbf{x}, \mathbf{y})$ in the current optimal solution of the MP, derive the BCs, and the SIs. The BCs are added for each case where $\bar{\mathbf{q}} \neq f(\bar{\mathbf{x}}, \bar{\mathbf{y}})$, where $\bar{\mathbf{q}}$ are the values of variables $\mathbf{q}$ in the current optimal solution of the MP. A lower bound for the original problem, which corresponds

to the objective value of the MP optimal solution, is obtained after each iteration. An upper bound is also obtained by determining feasible values for the production variables given the configurations used in the MP solution. The algorithm finds an optimal solution for the original problem when the MP solution satisfies $\bar{\mathbf{q}} = f(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ for all the cases.

The B&Ch algorithm is implemented using the branch-and-bound callbacks of an MIP solver as follows. First, the linear problem of the MP is solved. If the solution of the MP is not integer, then branching starts. If an integer feasible solution is found, then the values $f(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ are calculated, where $(\bar{\mathbf{x}}, \bar{\mathbf{y}})$ are the values of variables $(\mathbf{x}, \mathbf{y})$ in the current MP solution. Next, the BCs and the SIs are derived. The BCs are added for each case where $\bar{\mathbf{q}} \neq f(\bar{\mathbf{x}}, \bar{\mathbf{y}})$, where $\bar{\mathbf{q}}$ are the values of variables $\mathbf{q}$ in the current integer solution of the MP, and then the branching continues. The algorithm finds an optimal solution when the lower bound equals the upper bound in the search tree.

4 Applications of the proposed solution frameworks

The proposed solution frameworks are applied to different problems in the literature where process configuration and production planning decisions are jointly made. Sections 4.1 and 4.2 present the problems and the computational results for applications in the steel tube industry and the printing industry, respectively.

4.1 Applications in the steel tube industry

We consider three variants of the continuous stock cutting problem (CSCP) with setups studied in Hajizadeh and Lee (2007) in the context of the steel tube industry: the open-ended CSCP, the closed-ended CSCP, and the CSCP with knife-dependent setups. These problems consider a continuous steel tube supplied by a production facility which has to be cut into smaller pieces using a cutting machine with limited length. In the open-ended CSCP, the cutting equipment is designed to produce one extra piece of any length for each stroke of cutting motion. In the closed-ended CSCP, no extra-piece is produced. Figure 1 illustrates these two cases. In the CSCP with knife-dependent setups, setting up a pattern entails a fixed time and a variable time proportional to the number of knives used in each pattern. A further description of these problems is presented in the online supplement and the original paper which introduces this application.

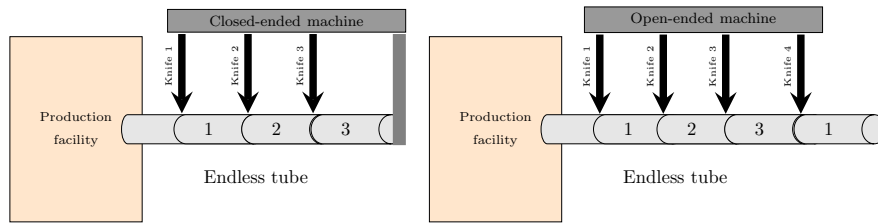


Figure 1: Closed-ended and open-ended configurations for cutting machines

We focus in this section on the open-ended CSCP, and mention how the solution approaches are extended for the other variants. The problem consists of determining: (1) the number of cutting patterns to be used; (2) the specific configuration of each pattern; and (3) the number of cutting strokes of each pattern. The objective is to minimize the total cutting time and setup time. The original formulation as defined by Hajizadeh and Lee (2007) is presented below. Table 2 presents the parameters and variables in this model.

$$\min \mathbf{c}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} : \quad \min C_s \sum_{j \in P} s_j + C_r \sum_{j \in P} z_j \quad (13)$$

$$\mathbf{A} \mathbf{x} \geq \mathbf{a} : \quad \sum_{i \in T} l_i a_{ij} \leq L s_j \quad \forall j \in P \quad (14)$$

Table 2: Parameters and variables for the open-ended CSCP

Sets	Parameters	Decision variables
T products (indexed by i);	C_r time for a single cutting stroke; C_s setup time for used patterns;	$\mathbf{x}$: s_j equals 1 if pattern j is used; 0, otherwise; a_{ij} number of pieces of product i in pattern j ;
P patterns (indexed by j).	d_i demand of product i ; l_i length of product i ; L width of the cutting machine; M large number defined as $\max_{i \in T} \{d_i\}$.	$\mathbf{y}$: z_j number of cutting strokes using pattern j ; x_i units of product i produced as extra pieces; $\mathbf{q}$: q_{ij} units of product i produced by pattern j .

$$\mathbf{B}\mathbf{x} + \mathbf{C}\mathbf{y} \geq \mathbf{b} : \quad z_j \leq Ms_j \quad \forall j \in P \quad (15)$$

$$\mathbf{q} = f(\mathbf{x}, \mathbf{y}) : \quad q_{ij} = a_{ij}z_j \quad \forall i \in T; j \in P \quad (16)$$

$$\mathbf{D}\mathbf{y} + \mathbf{E}\mathbf{q} \geq \mathbf{e} : \quad \sum_{j \in P} q_{ij} + x_i \geq d_i \quad \forall i \in T \quad (17)$$

$$\sum_{j \in P} z_j \geq \sum_{i \in T} x_i \quad (18)$$

$$z_{j-1} \leq z_j \quad \forall j \in P : j > 1 \quad (19)$$

$$\mathbf{x} \in \mathbb{B}^{|\mathcal{A}|}, \mathbf{y} \in \mathbf{Y}, \mathbf{q} \geq \mathbf{0} : \quad s_j \in \{0, 1\}; a_{ij}, z_j \in \mathbb{Z}_+; x_i, q_{ij} \geq 0 \quad \forall i \in T; j \in P \quad (20)$$

The objective function (13) minimizes the total setup and cutting time. Constraints (14) are the configuration constraints, which ensure that the sum of the length of the pieces in a used pattern do not exceed the machine length. Constraints (15) are the linking constraints to ensure that cutting strokes of a pattern occur only if such pattern is set up to the machine. Constraints (16) calculate the production quantities for each product obtained by each pattern. Note that these constraints are non-linear. Constraints (17)–(19) are production planning constraints. Constraints (17) ensure that the demand requirements are fulfilled. Constraints (18) ensure that the total number of extra pieces produced is less than or equal to the total number of strokes, as at most one extra piece can be produced per stroke. Inequalities (19) are symmetry-breaking constraints to order the used patterns according to the number of strokes. Finally, constraints (20) define the variables domain.

The formulation for the closed-ended CSCP is considered as a special case of the open-ended formulation presented above where no extra pieces are produced (i.e., $x_i = 0, \forall i \in T$). The CSCP with knife-dependent setups is an extension of the open-ended CSCP where additional setup variables are included to represent the variable time proportional to the number of knives used by each pattern. See Hajizadeh and Lee (2007) for further details.

4.1.1 Logic-based Benders reformulation

We proceed to define the MP1, SPs, and the BCs for the open-ended CSCP according to the general framework in Section 3.2. First, we make sure that the configuration decisions are represented by binary variables. In order to do that, we substitute variables a_{ij} by a sum of powers of two, as presented in equations (21). Let K_i (indexed by k) be the set of bins required to represent integer variables a_{ij} , i.e., $K_i = \{1, \dots, n_i\}$ where n_i is the minimum value such that $\sum_{k=1}^{n_i} 2^{k-1} \geq \lfloor \frac{L}{l_i} \rfloor$. Variable w_{ijk} equals 1 if k is used in writing a_{ij} , 0 otherwise. Note that alternative binary representations can be used to substitute variables a_{ij} . However, we implement this representation for the purposes of comparison with the benchmark approach in the literature (Hajizadeh and Lee, 2007), which also makes use of this variable substitution to linearize the original formulation.

$$a_{ij} = \sum_{k \in K_i} 2^{k-1} w_{ijk} = \sum_{k \in K_i} b_k w_{ijk} \quad \forall i \in T; j \in P \quad (21)$$

MP1 is defined by replacing the non-linear constraints (16) by an approximation for the production quantities, and by substituting the original variables a_{ij} as described above. The MP1 is therefore

formulated as follows: the objective function (13); the process configuration constraints (22), which substitute the original constraints (14); the linking constraints (15); the production planning constraints (17)–(19); constraints (23)–(24) as an approximation for the production quantities, where N_i is the maximum number of pieces of type i that can be included in a cutting pattern, i.e., $N_i = \left\lfloor \frac{L}{l_i} \right\rfloor$; and the domain constraints.

$$\mathbf{Ax} \geq \mathbf{a} : \quad \sum_{i \in T} l_i \sum_{k \in K_i} b_k w_{ijk} \leq L s_j \quad \forall j \in P \quad (22)$$

$$\mathbf{q} \geq \mathbf{Fx} + \mathbf{Gy} \quad q_{ij} \leq M_i \sum_{k \in K_i} w_{ijk} \quad \forall i \in T; j \in P \quad (23)$$

$$\sum_{l \in T: l \geq i} q_{lj} \leq N_i z_j \quad \forall i \in T; j \in P \quad (24)$$

Constraints (23) and (24) impose valid bounds on the total units of product i produced by pattern j . Constraints (23) ensure that $q_{ij} = 0$ if product i is not in pattern j (i.e., the original variables $a_{ij} = 0$ and therefore $\sum_{k \in K_i} w_{ijk} = 0$). In case product i is in pattern j , then the maximum produced amount is limited by $M_i = \max \{d_i; \max_{i' \in T: l_{i'} \leq L - l_i} \{d_{i'}\}\}$. Constraints (24) link the produced amounts with the number of strokes of each pattern, where products are assumed to be ordered from the smallest to the largest size, so that $N_i \leq N_{i-1}$, $\forall i \in T : i > 1$. These constraints impose that the total units of products $\{i, \dots, |T|\}$ produced by pattern j is at most the number of strokes of such pattern z_j , multiplied by N_i which is the maximum number of pieces that can be allocated to a pattern considering the corresponding subset of products. Note that this is a strengthened version of the following constraints: $q_{ij} \leq N_i z_j$, $\forall i \in T, j \in P$, which bound q_{ij} to the maximum number of pieces of type i that can fit in a pattern multiplied by the number of strokes of pattern j .

Given a solution h of MP1, the BCs (25) are derived based on the values for the configuration variables in the corresponding solution. In order to do that, the parameter $\bar{a}_{ij}^h$ which represents the number of pieces i in pattern j is computed as $\bar{a}_{ij}^h = \sum_{k \in K_i} b_k \bar{w}_{ijk}^h$, where $\bar{w}_{ijk}^h$ is the value of variable w_{ijk} in the given solution. The production quantities are checked to verify whether they satisfy the conditions of the original problem or not, i.e., check if $\bar{q}_{ij}^h = z_j^h \bar{a}_{ij}^h$, where $\bar{q}_{ij}^h$ and z_j^h are the values of variables q_{ij} and z_j in the given solution of MP1, respectively. Let $\mathcal{Q}_h = \{(i, j) : i \in T, j \in P, \bar{q}_{ij}^h \neq z_j^h \bar{a}_{ij}^h\}$ be the set of pairs (i, j) for which the variables q_{ij} do not satisfy the conditions of the original problem. The BCs are added for each $h \in \mathcal{H}'$ and $(i, j) \in \mathcal{Q}_h$, where $\mathcal{H}'$ corresponds to the set of solutions of MP1 considered so far.

$$\text{Logic-based Benders cuts (BCs): } q_{ij} \leq z_j \bar{a}_{ij}^h + M_i \sum_{\substack{k \in K_i: \\ \bar{w}_{ijk}^h = 0}} w_{ijk} \quad \forall h \in \mathcal{H}'; (i, j) \in \mathcal{Q}_h \quad (25)$$

Inequalities (25) impose that the units of i produced by pattern j equals $z_j \bar{a}_{ij}^h$, every time that the number of pieces of type i in pattern j equals $\bar{a}_{ij}^h$ (i.e., $\sum_{k \in K_i: \bar{w}_{ijk}^h = 0} w_{ijk} = 0$ and $\sum_{k \in K_i: \bar{w}_{ijk}^h = 1} (1 - w_{ijk}) = 0$). The equality between the terms q_{ij} and $z_j \bar{a}_{ij}^h$ is ensured since variables z_j are minimized by the objective function. Note that, besides ensuring $q_{ij} = z_j \bar{a}_{ij}^h$ when the number of pieces i in pattern j equals $\bar{a}_{ij}^h$, the BCs (25) also impose valid bounds for any other feasible solution of MP1. In particular, when the number of pieces i in pattern j is less than $\bar{a}_{ij}^h$ (i.e., $\sum_{k \in K_i: \bar{w}_{ijk}^h = 0} w_{ijk} = 0$ and $\sum_{k \in K_i: \bar{w}_{ijk}^h = 1} (1 - w_{ijk}) \geq 1$), the BCs ensure $q_{ij} \leq z_j \bar{a}_{ij}^h$. For any other value to the number of pieces i in pattern j which implies $\sum_{k \in K_i: \bar{w}_{ijk}^h = 0} w_{ijk} \geq 1$, the BCs are also valid, since M_i is computed as the maximum units of i that can be produced by any feasible pattern.

4.1.2 Enhanced reformulation through SIs

We present the enhanced reformulation with logic-based subset inequalities proposed for the problem in the steel industry. For this reformulation, we classify the products into a set of groups G (indexed by g), according to the maximum number of pieces that can be allocated to a cutting pattern, i.e., the values

of the input parameter $\left\lfloor \frac{L}{l_i} \right\rfloor$. Denote N_g as the maximum number of pieces of any product in group g that can fit in a cutting pattern and T_g as the set of products in group g , i.e., $T_g = \left\{ i \in T : \left\lfloor \frac{L}{l_i} \right\rfloor = N_g \right\}$. The new logical variables $y_{gj} \in \{0, 1\}$, $\forall g \in G, j \in P$, are included to the problem, where y_{gj} equals 1 if at least one piece of any product in set T_g is assigned to pattern j ; 0 otherwise. MP2 in this enhanced formulation can be formulated as MP1 presented in the previous section with the additional constraints (26), (27) and the domain of variables y_{gj} . Constraints (26) and (27) link the new variables with the configuration variables w_{ijk} , such that $y_{gj} = 1$ if and only if, at least one piece of any type $i : i \in T_g$ is assigned to pattern j . Parameter B_g is calculated as $B_g = \sum_{i \in T_g} |K_i|$.

$$\mathbf{H}\mathbf{x} + \mathbf{I}\mathbf{z} \geq \mathbf{g} : \sum_{i \in T_g} \sum_{k \in K_i} w_{ijk} \leq B_g y_{gj} \quad g \in G; j \in P \quad (26)$$

$$y_{gj} \leq \sum_{i \in T_g} \sum_{k \in K_i} w_{ijk} \quad g \in G; j \in P \quad (27)$$

Given a solution h of MP2, the SIs are devised with the aim of strengthening the bounds on the total production amounts. In order to do that, the used patterns in solution h and the particular group g to which the largest product allocated to these patterns belongs are identified. Let $\mathcal{Q}'_h = \{(j, g) : j \in P, \bar{s}_j^h = 1, g = \arg \min_{g' \in G: \bar{y}_{g'j}^h = 1} N_{g'}\}$ be the set which saves this information, where $\bar{s}_j^h$ and $\bar{y}_{gt}^h$ are the values of variables s_j and y_{gt} in the corresponding solution. We use this information obtained from the master solution to strengthen the bounds of the produced amounts of products that do not belong to this particular group g , i.e., for products $i : i \notin T_g$. For this reformulation it is also assumed that the products are ordered from the smallest to the largest size, so that $N_i \leq N_{i-1}, \forall i \in T$. Denote $i_g^{\min}$ and $i_g^{\max}$ as the smallest and the largest index i of the products in group g , respectively (i.e., the first and the last element in T_g).

Inequalities (28) and (29) are the logic-based subset inequalities implemented for the problems in this application. Inequalities (28) add a valid upper bound on the total units of products $i = 1, \dots, i_g^{\min} - 1$ produced by pattern j , where $(j, g) \in \mathcal{Q}'_h$. Parameter C_g^I corresponds to the maximum number of pieces of any type that can coexist with one piece of any product in group g , i.e., $C_g^I = \left\lfloor \frac{L - l_{i_g^{\min}}}{l_1} \right\rfloor$. Parameter D_g^I is a large number computed as $M_1 \left(\left\lfloor \frac{L}{l_1} \right\rfloor - C_g^I \right)$. Similarly, inequalities (29) impose an upper bound on the total units of products $i = i_g^{\max} + 1, \dots, |T|$ produced by pattern j , where $(j, g) \in \mathcal{Q}'_h$. Parameter C_g^{II} corresponds to the maximum number of pieces of any type $i : i > i_g^{\max}$ that can coexist with one piece of any product in group g , i.e., $C_g^{II} = \left\lfloor \frac{L - l_{i_g^{\max}+1}}{l_{(i_g^{\max}+1)}}.$

The parameter D_g^{II} is a large number computed as $M_{i_g^{\max}} \left(\left\lfloor \frac{L}{l_{i_g^{\max}+1}} \right\rfloor - C_g^{II} \right)$. The SIs are derived in addition to the BCs for each $h \in \mathcal{H}'$, where $\mathcal{H}'$ is the set of solutions of MP2 considered so far.

$$\text{SIs: } \sum_{\substack{i \in T: \\ i < i_g^{\min}}} q_{ij} \leq C_g^I z_j + D_g^I (1 - y_{gj}) \quad h \in \mathcal{H}'; \forall (j, g) \in \mathcal{Q}'_h \quad (28)$$

$$\sum_{\substack{i \in T: \\ i > i_g^{\max}}} q_{ij} \leq C_g^{II} z_j + D_g^{II} (1 - y_{gj}) \quad h \in \mathcal{H}'; \forall (j, g) \in \mathcal{Q}'_h \quad (29)$$

The reformulations presented in this section and Section 4.1.1 can be extended to the closed-ended CSCP and the CSCP with knife-dependent setups. The demand constraints, the configuration constraints, and the objective function in the master problems are adapted to include the particular conditions of each variant according to Hajizadeh and Lee (2007). The BCs and the SIs can be implemented as for the open-ended CSCP.

4.1.3 Computational results

This section presents the computational results for the application in the steel tube industry. All the computational tests were implemented using Python 3.7 and solver CPLEX 12.9.0 on a computer with processor Intel E5-2683/2.1GHz and 16GB of RAM. The data sets correspond to the 47 benchmark problems tested in the paper by Hajizadeh and Lee (2007) for the three variants considered in this application. We classify the instances into two data sets according to the results of the benchmark approach for each problem: Set A contains the instances for which an optimal solution is found; Set B contains instances which could not be solved to optimality within the time limit. Table 3 presents the number of instances in each data set (Inst.), the number of products $|T|$, the maximum number of patterns $|P|$ that can be used, and the number of groups $|G|$ obtained by classifying the products according to the maximum number of pieces than can fit in a cutting pattern. The computing time limit is 1800 seconds, as imposed for the benchmark approach in the original study.

Table 3: Data sets for the cutting problems in the steel tube industry

Data	Open-ended CSCP				Closed-ended CSCP				Knife-dependent CSCP			
	Inst.	$ T $	$ P $	$ G $	Inst.	$ T $	$ P $	$ G $	Inst.	$ T $	$ P $	$ G $
Set A	40	6-7	4-7	3-5	46	6-14	4-10	3-5	40	6-7	4-7	3-5
Set B	7	14	10	4-5	1	14	10	4-5	7	14	10	4-5

Table 4 presents the average results of these computational experiments. We present the findings of the benchmark approach proposed in Hajizadeh and Lee (2007), which consists of linearized models solved using CPLEX, the reformulation in Section 4.1.1 solved using the LBB and the B&Ch algorithms (i.e., LBB and B&Ch, respectively), and the enhanced reformulation in Section 4.1.2 solved using the LBB and B&Ch algorithms (i.e., LBB+SI and B&Ch+SI, respectively). To properly carry out comparisons, we reproduced the findings of the benchmark approach by using the same programming language and CPLEX version as for the proposed methods. The following statistics are reported: the number of optimal solutions (OS), the average upper bound (UB), the average lower bound (LB), the average gap (Gap), and the average computing time in seconds (Time).

Table 4: Average results for the cutting stock problems in the steel tube industry

Problem	Sol. method	Set A		Set B				
		UB	Time	OS	UB	LB	Gap	Time
Open-ended CSCP	Benchmark	36.63	9.52	0	121.71	80.67	33.49%	1800.00
	LBB	36.63	2.46	6	121.00	120.86	0.12%	607.73
	B&Ch	36.63	1.27	3	121.57	116.41	4.29%	1268.79
	LBB+SI	36.63	2.47	7	121.00	121.00	0.00%	133.87
	B&Ch+SI	36.63	1.44	6	121.00	120.73	0.23%	678.97
Closed-ended CSCP	Benchmark	81.33	43.97	0	219.00	200.20	8.58%	1800.00
	LBB	81.33	58.15	0	219.00	214.00	2.28%	1800.04
	B&Ch	81.33	40.16	0	219.00	215.00	1.83%	1800.00
	LBB+SI	81.33	25.38	0	219.00	215.00	1.83%	1800.08
	B&Ch+SI	81.33	32.10	0	219.00	205.46	6.18%	1800.02
Knife-dependent setups CSCP	Benchmark	37.13	41.01	0	120.64	71.68	40.42%	1800.00
	LBB	37.13	24.71	3	121.36	117.79	2.98%	1177.41
	B&Ch	37.13	17.49	2	121.79	112.96	7.27%	1333.01
	LBB+SI	37.13	12.48	5	119.71	119.36	0.31%	778.14
	B&Ch+SI	37.13	18.59	3	120.93	114.48	5.43%	1058.77

Results in Table 4 show that for all the cases in Set A, except for the LBBD in the closed-ended CSCP, the proposed solution methods find optimal solutions in shorter computing times. On average, the best performing algorithm solves these problem instances approximately 7.5 times faster than the benchmark approach for the open-ended CSCP, 1.7 times faster for the closed-ended CSCP, and 3.3 times faster for the knife-dependent setups CSCP. The enhanced reformulation, which make uses of the SIs, shows some advantage over the logic-based Benders reformulation when solved using the cutting plane implementation (LBBD+SIs). Note that for the two last problems, the computing times are reduced approximately by half using the LBBD+SIs compared to the LBBD.

The efficiency of the proposed frameworks, and in particular of the enhanced reformulation with the logic-based subset inequalities, becomes more evident on the results for Set B where the benchmark fails to find any optimal solution. For the open-ended CSCP, the best performing algorithm finds optimal solution for all the 7 instances in an average time of 133.87 seconds, whereas the benchmark approach only finds feasible solutions with an average optimality gap of 33.49%. For the closed-ended CSCP, an absolute reduction of 6.7% in the average gap is achieved by the best performing algorithm in comparison with the benchmark approach. Finally, for the knife-dependent setups CSCP, 5 out of the 7 instances in this set are solved to optimality by the best performing algorithm, so that an absolute reduction of 40.1% in the average gap and more than a 50% reduction in the average computing time are achieved in comparison with the benchmark approach. The best implementation for the problems in this industrial application is the enhanced reformulation solved using the LBBD implementation, which iterates between solving the master and the subproblems. The number of iterations of this implementation is 7.1 on average for the first problem, 8.6 on average for the second problem, and 7.8 on average for the third problem. Detailed results can be found in the online supplement.

4.2 Application in the printing industry

This application considers the production planning problem studied in Baumann et al. (2015) and Baumann and Trautmann (2014). The problem is inspired by a real-world offset printing process where customer-specific designs are imprinted on napkin pouches. The designs are allocated to printing-plate slots, as depicted in Figure 2, such that the customer demand for each design is fulfilled. Various technical constraints related to incompatibilities between the production equipment and the design features (e.g., standard or customer-specific designs, color code, and white border) are considered to ensure that the configurations of the printing plates are feasible in practice.

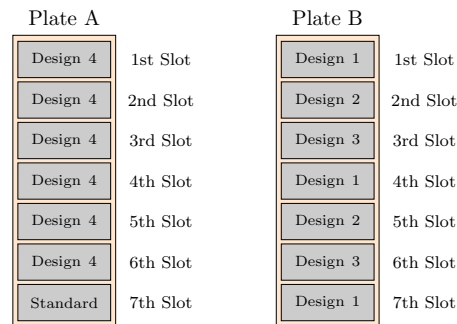


Figure 2: Examples of feasible configurations for a printing plate

The problem decisions aim to determine: (1) the number of printing plates to be used; (2) the allocation of designs to the slots of these plates; and (3) the number of rotations of each printing plate, i.e., the number of times that the printing plate is used. The objective is to minimize the total overproduction costs and setup costs. We focus on the benchmark approach presented in Baumann and Trautmann (2014), as this model was proved to be more efficient than the one in Baumann et al. (2015). The original formulation for this application is provided below. Table 5 presents the parameters and variables in this model.

Table 5: Parameters and variables for the printing problem

Sets		Parameters		Decision variables	
I	designs (indexed by i);	d_i	demand for design i ;	$\mathbf{x} :$	W_p equals 1 if plate p is used; 0, otherwise;
P	plates (indexed by p);	c^P	setup cost per used plate;	K_{inp}	equals 1 if design i is allocated to n slots
J	slots = $\{1, \dots, J \}$ (indexed by n);	c_i^O	overproduction cost per unit		on plate p ; 0 otherwise;
C	color codes (indexed by c);		of customer-specific design i ;	Z_{cp}	equals 1 if a design with color code c is on plate p ; 0 otherwise;
I^O	customer-specific designs;	c_i^S	overproduction cost per unit		rotations of plate p ;
I^S	standard designs;		of standard design i ;	$\mathbf{y} :$	r_p overproduced units of design i ;
I_c	designs with color code c ;	$\bar{c}$	maximum number of different	v_i	units of design i produced by plate p .
I_w	designs with white border;		color codes per plate;	$\mathbf{q} :$	q_{ip}

$$\min \mathbf{c}^T \mathbf{x} + \mathbf{d}^T \mathbf{y} : \quad \min \sum_{p \in P} c^P W_p + \sum_{i \in I^O} c_i^O v_i + \sum_{i \in I^S} c_i^S v_i \quad (30)$$

$$\mathbf{Ax} \geq \mathbf{a} : \quad \sum_{i \in I} \sum_{n \in J} n K_{inp} = |J| W_p \quad \forall p \in P \quad (31)$$

$$\sum_{i \in I^S} \sum_{n \in J} n K_{inp} \leq 1 \quad \forall p \in P \quad (32)$$

$$\sum_{i \in I_w} \sum_{n \in J} \frac{1}{2} n K_{inp} + \sum_{i \in I^S} \sum_{n \in J} K_{inp} \geq W_p \quad \forall p \in P \quad (33)$$

$$\sum_{c \in C} Z_{cp} \leq \bar{c} \quad \forall p \in P \quad (34)$$

$$\sum_{i \in I_c} \sum_{n \in J} K_{inp} \leq |J| Z_{cp} \quad \forall c \in C; p \in P \quad (35)$$

$$\sum_{p \in P} \sum_{n \in J} K_{inp} = 1 \quad \forall i \in I^O \quad (36)$$

$$W_{p-1} \geq W_p \quad \forall p \in P : p > 1 \quad (37)$$

$$\mathbf{q} = f(\mathbf{x}, \mathbf{y}) : \quad q_{ip} = r_p \sum_{n \in J} n K_{inp} \quad \forall i \in I; p \in P \quad (38)$$

$$\mathbf{Dy} + \mathbf{Eq} \geq \mathbf{e} : \quad \sum_{p \in P} q_{ip} = d_i + v_i \quad \forall i \in I \quad (39)$$

$$\mathbf{x} \in \mathbb{B}^{|\mathcal{A}|} : \quad W_p, K_{inp}, Z_{cp} \in \{0, 1\} \quad \forall i \in I; p \in P; n \in J; c \in C \quad (40)$$

$$\mathbf{y} \in \mathbf{Y}, \mathbf{q} \geq \mathbf{0} : \quad r_p, v_i, q_{ip} \geq 0 \quad \forall i \in I; p \in P \quad (41)$$

The objective function (30) minimizes the total setup costs and overproduction costs. Constraints (31)–(37) are the configuration constraints. Constraints (31) ensure that every slot of a used plate is occupied by one design. Constraints (32) guarantee that at most one slot per plate is occupied by a standard design. Constraints (33) ensure that if a plate is used, then either white-border designs are allocated to at least two slots or a standard design is allocated to one slot. Constraints (34) and (35) prevent that more than $\bar{c}$ different color-codes are allocated to the same plate. Constraints (36) prohibit that a customer-specific design is allocated to different plates, so that each demand order is always fulfilled by a single plate. Constraints (31)–(36) are all the technical and organizational constraints imposed on the configurations. For the details and justification of these constraints, we refer the reader to the online supplement and the paper by Baumann et al. (2015). Inequalities (37) are symmetry-breaking constraints which ensure that the plates with the smallest indices are used first. Constraints (38) compute the units of each design produced by each plate. Note that these constraints are non-linear. Constraints (39) are the production planning constraints that determine the overpro-

duced units while making sure that the demand is satisfied. Finally, constraints (40) and (41) define the domain of the variables.

4.2.1 Logic-based Benders reformulation

We discuss how the master problem and the logic-based Benders cuts are formulated for this application. MP1 is defined as follows: the objective function (30), constraints (31)–(37), (39)–(41), and an approximation of the production quantities imposed by constraints (42) and (43). Parameter M is calculated as $M = \max_{i \in I} \{d_i\}$. Note that variables r_p are not used in this reformulation and no variable substitution is needed to apply the decomposition framework to this problem, as the configuration variables are defined as binary in the original formulation.

$$\mathbf{q} \geq \mathbf{F}\mathbf{x} + \mathbf{G}\mathbf{y} : \quad q_{ip} \leq M \sum_{n \in J} n K_{inp} \quad \forall i \in I; p \in P \quad (42)$$

$$q_{ip} \geq \sum_{n \in J} d_i K_{inp} \quad \forall i \in I^O; p \in P \quad (43)$$

Constraints (42) ensure that $q_{ip} = 0$ if no design i is allocated to plate p . Otherwise, the total units of design i produced by plate p is limited by the number of slots for this design, i.e., $\sum_{n \in J} n K_{inp}$, multiplied by an upper bound on the number of rotations of any printing plate, i.e., $\max_{i \in I} \{d_i\}$. Constraints (43) impose a lower bound on the total units of each customer-specific design produced by plate p . These bounds are valid as constraints (36) ensure that the demand of any customer-specific design is fulfilled using a single plate.

Given a feasible solution h of MP1, the logic-based Benders cuts can be derived based on the values for the configuration variables K_{inp} in the corresponding solution. In order to do that, the number of rotations of each used plate in solution h , denoted $\bar{r}_p^h$, needs to be computed. Because of the technical constraints (36) which ensure that customer-specific designs are assigned to a single plate, $\bar{r}_p^h$ can be calculated as the number of rotations required to fulfill the complete demand of the customer-specific designs allocated to plate p in the given solution, i.e., $\bar{r}_p^h = \max_{i \in I^O: \sum_{n \in J} \bar{K}_{inp}^h \neq 0} \left\{ \frac{d_i}{\sum_{n \in J} n \bar{K}_{inp}^h} \right\}$, where $\bar{K}_{inp}^h$ are the values of variables K_{inp} in solution h . Note that for other printing problems where constraints (36) does not hold, i.e., the complete demand of a design can be fulfilled using several plates, computing $\bar{r}_p^h$ based on the fixed values $\bar{K}_{inp}^h$ could require solving a linear program.

Inequalities (44) are the BCs proposed for the printing problem. The logic of these cuts is such that, every time that design i' is allocated to n' slots of plate p (i.e., $K_{i'n'p} = 1$), the number of units of each design i produced by this plate equals the number of rotations $\bar{r}_p^h$ times the number of slots for the corresponding design $\sum_{n \in J} n K_{inp}$. The equality between q_{ip} and $\bar{r}_p^h \sum_{n \in J} n K_{inp}$ is ensured because the overproduction variables $v_i \geq 0$ are minimized in the objective function. For a given plate p , the pair (i', n') corresponds to the particular design i' and the corresponding number of slots n' which requires a larger number of rotations of such plate in order to fulfilled the total demand, i.e., $(i', n') = \arg \max_{i \in I^O, n \in J: \bar{K}_{inp}^h = 1} \left\{ \frac{d_i}{n \bar{K}_{inp}^h} \right\}$. That is, the pair (i', n') in plate p is such that design i' is allocated to this plate and $\frac{d_{i'}}{n'} = \bar{r}_p^h$. Let $\mathcal{C}_{hp}$ be the set which contains the pair (i', n') , i.e., $\mathcal{C}_{hp} = \left\{ (i, n) : \arg \max_{i \in I^O, n \in J: \bar{K}_{inp}^h = 1} \left\{ \frac{d_i}{n \bar{K}_{inp}^h} \right\} \right\}$. The parameter B_i^h is a large number calculated as $B_i^h = \bar{r}_p^h |J|, \forall i \in I^O$ and $B_i^h = \bar{r}_p^h, \forall i \in I^S$. These cuts are derived for each $h \in \mathcal{H}'$, where $\mathcal{H}'$ is the set of solutions of MP1 considered so far, and for each pair (i, p) for which the values of the production quantities in variables q_{ip} do not satisfy the conditions of the original problem, i.e., $(i, p) \in \mathcal{Q}_h$, where $\mathcal{Q}_h = \{(i, p) : i \in I, p \in P, \bar{q}_{ip}^h \neq \bar{r}_p^h \sum_{n \in J} n \bar{K}_{inp}^h\}$ and $\bar{q}_{ip}^h$ is the value of variable q_{ip} in solution h .

$$BCs: \quad q_{ip} \geq \bar{r}_p^h \sum_{n \in J} n K_{inp} - B_i^h (1 - K_{i'n'p}) \quad \forall h \in \mathcal{H}'; (i, p) \in \mathcal{Q}_h; (i', n') \in \mathcal{C}_{hp} \quad (44)$$

Note that the lower bounds on variables q_{ip} imposed by the BCs are derived in terms of the configuration variables $K_{i'n'p}$ and independent of the variables for other designs allocated to the same plate. This is valid because, as long as n' slots for the customer-specific design i' are allocated to plate p , the number of rotations of plate p is always greater than or equal to $\bar{r}_p^h$, as the complete demand of design i' must be fulfilled using only this plate. As an example, let's consider an instance where the demand of four designs (i.e., $d_1 = 120$, $d_2 = 100$, $d_3 = 120$, and $d_4 = 600$) must be fulfilled using two printing plates. A given solution of the master problem indicates the configurations shown in Figure 2 for Plate A and Plate B. The subproblem computes the number of rotations of these plates required to fulfill the total demand, i.e., $\bar{r}_A = 100$ and $\bar{r}_B = 50$. Note that, as long as two slots of Design 3 are allocated to Plate B, regardless the other designs allocated to the remaining slots, $\bar{r}_B \geq 50$ because the complete demand of this design must be fulfilled using this only plate. Accordingly, in this example the pair (i', n') determined for the BCs (44) is (4, 6) for Plate A and (3, 2) for Plate B. The BCs (44) are hence a strengthened version of the cuts that include the configuration variables of all the designs allocated to the same pattern: $q_{ip} \geq \bar{r}_p^h \sum_{n \in J} n K_{inp} - B_i^h \sum_{l \in I} \sum_{\substack{n \in J: \\ \bar{K}_{lnp}^h = 1}} (1 - K_{lnp}) - B_i^h \sum_{l \in I} \sum_{\substack{n \in J: \\ \bar{K}_{lnp}^h = 0}} K_{lnp}$.

4.2.2 Enhanced reformulation through SIs

We present the enhanced reformulation with logic-based subset inequalities for the printing problem. For this reformulation, we classify the designs into a set of groups G (indexed by g) according to the demand parameters such that designs in the same group have the same demand level. Denote d_g^G as the demand of any design in group g and $I_g = \{i \in I : d_i = d_g^G\}$ as the subset of designs in group g . The new logical variables $U_{gnp} \in \{0, 1\}, \forall g \in G, n \in J, p \in P$, are added to the problem, where U_{gnp} equals 1 if at least one design in group g is assigned to n slots in plate p , 0 otherwise. MP2 in this enhanced reformulation can be formulated as MP1 presented in the previous section in addition to constraints (45) and (46), which link the new logical variables U_{gnp} with the original configuration variables K_{inp} , and the corresponding domain constraints. Inequalities (45) and (46) ensure that $U_{gnp} = 1$ when at least one design in group g is allocated to n slots in plate p , and force that $U_{gnp} = 0$ otherwise. Parameter B_g is calculated as $B_g = \min\{|J|, |I_g|\}$.

$$\mathbf{H}\mathbf{x} + \mathbf{I}\mathbf{z} \geq \mathbf{g} : \quad \sum_{i \in I_g} K_{inp} \leq B_g U_{gnp} \quad \forall g \in G; n \in J; p \in P \quad (45)$$

$$\sum_{i \in I_g} K_{inp} \geq U_{gnp} \quad \forall g \in G; n \in J; p \in P \quad (46)$$

Given a solution h of MP2, the logic-based subset inequalities are derived in order to strengthen the bounds on the total produced amounts. For this purpose, for each used plate in solution h , the particular group g to which the design that requires a greater number of rotations to fulfill its demand belongs, is determined. Let $\mathcal{Q}'_h = \left\{ (g, n, p) : g \in G, n \in J, p \in P, \bar{U}_{gnp}^h = 1, \frac{d_g^G}{n} = \bar{r}_p^h \right\}$ be the set which saves this information, where $\bar{U}_{gnp}^h$ are the values of variables U_{gnp} in solution h . We use this information to strengthen the bounds on the total produced units of customer-specific designs and standard designs through inequalities (47) and (48), respectively. Inequalities (47) ensure that, as long as at least one design in group g is assigned to n slots of plate p (i.e., $U_{gnp} = 1$, where $(g, n, p) \in \mathcal{Q}'_h$), the total produced units of customer-specific designs is greater than $|J| \bar{r}_p^h$ if no standard designs are allocated to this plate, or greater than $(|J| - 1) \bar{r}_p^h$ otherwise. These inequalities are valid since the original constraints (31) ensure that all the plate slots are used and (32) ensure that at most one slot is assigned to standard designs in any printing plate. Inequalities (48) impose a lower bound on the total units of standard designs produced by each used plate when at least one design in group g is assigned to n slots of this plate (i.e., $U_{gnp} = 1$, where $(g, n, p) \in \mathcal{Q}'_h$).

$$\text{SIs:} \quad \sum_{i \in I^O} q_{ip} \geq \bar{r}_p^h |J| U_{gnp} - \bar{r}_p^h \sum_{i \in I^S} \sum_{n \in J} K_{inp} \quad \forall h \in \mathcal{H}'; (g, n, p) \in \mathcal{Q}'_h \quad (47)$$

$$\sum_{i \in I^S} q_{ip} \geq \bar{r}_p^h \sum_{i \in I^S} \sum_{n \in J} K_{inp} - \bar{r}_p^h (1 - U_{gnp}) \quad \forall h \in \mathcal{H}'; (g, n, p) \in \mathcal{Q}'_h \quad (48)$$

4.2.3 Computational results

Table 6 presents the characteristics of the data sets for this application, which consists of 72 problem instances proposed in the paper by Baumann and Trautmann (2014). We classify these instances according to the results of the benchmark approach: Set A contains the instances for which an optimal solution is found; Set B contains instances for which solutions with an optimality gap of less than 30% are found; Set C contains instances for which solutions with an optimality gap greater than or equal to 30% are found; and Set D contains instances for which no feasible solution is found within the time limit. Table 6 shows the number of instances in each data set (Inst.), the number of designs $|I|$, the maximum number of used plates $|P|$, and the number of groups $|G|$ obtained by classifying designs according their demand levels. The computing time limit is 1800 seconds, as imposed for the benchmark approach in the original studies.

Table 6: Data sets for the printing problem

Data	Inst.	$ I $	$ P $	$ G $
Set A	26	6–23	5–15	1–6
Set B	10	23–29	20–25	4–10
Set C	19	33–91	25–70	5–24
Set D	17	58–117	50–90	10–35

Table 7 presents the average results for Sets A, B, and C. Similar to the tests for the cutting stock problems, we reproduced the findings of the benchmark approach, i.e., solving the linear formulation in Baumann and Trautmann (2014) using an MIP solver, so that the model was implemented using the same programming language, solver, and computer as for the proposed algorithms. Results for Set A show that the proposed enhanced reformulation enables to find optimal solutions much faster than the benchmark approach. In particular, the B&Ch+SI algorithm solves these problem instances approximately 9.4 times faster compared with the benchmark solution method. The LBD and B&Ch algorithms fail to find an optimal solution for 2 and 1 out of the 26 instances in this data set, respectively. However, excluding these particular cases, the average solving time of the B&Ch algorithm is still approximately 60% of the average solving time of the original solution method.

Table 7: Average results for the printing problem

Sol. method	Set A		Set B					Set C		
	UB	Time	OS	UB	LB	Gap	Time	UB	LB	Gap
Benchmark	1384.89	68.99	0	2743.54	2192.42	19.48%	1800.01	6662.01	2993.43	45.60%
LBD	1386.07	410.14	0	3247.14	2247.81	29.72%	1800.04	6866.66	3385.69	48.36%
B&Ch	1384.89	106.64	0	2741.02	2310.01	15.54%	1800.02	5794.77	3277.60	39.83%
LBD+SI	1384.89	44.19	3	2742.13	2611.54	4.44%	1527.89	6231.98	3582.79	36.41%
B&Ch+SI	1384.89	7.35	5	2724.84	2622.90	3.49%	1171.20	5725.61	3390.53	34.87%

The solution methods which make use of the logic-based subset inequalities find optimal solutions for some instances in Set B, whereas the benchmark approach only finds solutions with an average gap of 19.48%. Note that the LBD+SI and B&Ch+SI algorithms solved 3 and 5 out of the 10 instances in Set B, respectively.

Finding good quality solutions for large instances of this problem is challenging according to the results for Sets C and D. None of the instances in these sets can be solved to optimality within 30 minutes by any of the tested methods. However, the algorithms to solve the enhanced reformulation, and in particular the B&Ch implementation, provide better solutions and lower bounds at the end of the time limit. These improvements allow an absolute reduction in the average gap of approximately 10.73% compared to the benchmark results for Set C. Results for Set D are not presented, as no feasible solutions are found within the time limit by the benchmark approach. However, the proposed LBD methods still provide improved results in terms of the total number of feasible solution found,

yet average optimality gaps are between 54.8% and 58.6% in these cases. Detailed results for this application can be found in the online supplement.

5 Concluding remarks

We propose a general solution framework based on the logic-based Benders decomposition technique to solve a variety of integrated production planning problems with process configuration decisions. We introduce a logic-based Bender reformulation and an enhanced reformulation through logic-based subset inequalities created based on common characteristics in the input parameters. These reformulations are presented as general frameworks that can be solved using either the LBB or the B&Ch implementation. We tested the proposed approaches on three variants of cutting stock problems which appear in the steel industry and an application in the printing industry from the literature.

In general, the enhanced reformulations with the logic-based subset inequalities speed up the convergence of the decomposition methods and our approaches significantly outperform the benchmark approach in the literature. These results are consistent for all the tested applications. The computational results also show that the differences in the performance of the LBB and B&Ch implementations may be related to the complexity of the decisions that remain in the master problem. For the cutting stock problems where a few technical constraints are imposed to the configurations decisions, the LBB algorithms outperform the B&Ch implementation for most instances. For the printing application, where many technical constraints are imposed to the configuration decisions, the performance of the LBB is generally inferior to that of the B&Ch implementation.

References

- P. Baumann and N. Trautmann. Efficient symmetry-breaking formulations for grouping customer orders in a printing shop. In 2014 IEEE International Conference on Industrial Engineering and Engineering Management, pages 506–510, Dec 2014.
- P. Baumann, S. Forrer, and N. Trautmann. Planning of a make-to-order production process in the printing industry. *Flexible Services and Manufacturing Journal*, 27(4):534–560, Dec 2015.
- J. C. Beck. Checking-up on branch-and-check. In David Cohen, editor, *Principles and Practice of Constraint Programming – CP 2010*, pages 84–98, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.
- Zeger Degraeve and Martina Vandebroek. A mixed integer programming model for solving a layout problem in the fashion industry. *Management Science*, 44(3):301–310, 1998.
- Zeger Degraeve, Willy Gochet, and Raf Jans. Alternative formulations for a layout problem in the fashion industry. *European Journal of Operational Research*, 143(1):80–93, 2002.
- Maxence Delorme, Manuel Iori, and Silvano Martello. Logic based Benders’ decomposition for orthogonal stock cutting problems. *Computers & Operations Research*, 78:290–298, 2017.
- Mohammad M Fazel-Zarandi and J Christopher Beck. Using logic-based Benders decomposition to solve the capacity-and distance-constrained plant location problem. *INFORMS Journal on Computing*, 24(3):387–398, 2012.
- Mohammad M Fazel-Zarandi, Oded Berman, and J Christopher Beck. Solving a stochastic facility location/fleet management problem with logic-based Benders’ decomposition. *IIE Transactions*, 45(8):896–911, 2013.
- I. Hajizadeh and C. Lee. Alternative configurations for cutting machines in a tube cutting mill. *European Journal of Operational Research*, 183(3):1385–1396, 2007.
- Imen Hamdi and Taïcir Loukil. Logic-based Benders decomposition to solve the permutation flowshop scheduling problem with time lags. In 5th International Conference on Modeling, Simulation and Applied Optimization (ICMSAO), pages 1–7, 2013.
- J. N. Hooker. Planning and scheduling by logic-based Benders decomposition. *Operations Research*, 55(3):588–602, 2007.

- J.N. Hooker and G. Ottosson. Logic-based Benders decomposition. *Mathematical Programming*, 96(1):33–60, Apr 2003. ISSN 1436-4646.
- J Martens. Two genetic algorithms to solve a layout problem in the fashion industry. *European Journal of Operational Research*, 154(1):304–322, 2004.
- Karim Pérez Martínez, Yossiri Adulyasak, Raf Jans, Reinaldo Morabito, and Eli Angela Vitor Toso. An exact optimization approach for an integrated process configuration, lot-sizing, and scheduling problem. *Computers & Operations Research*, 103:310–323, 2019.
- Martin Riedler and Günther Raidl. Solving a selective dial-a-ride problem with logic-based Benders decomposition. *Computers & Operations Research*, 96:30–54, 2018.
- Atle Riise, Carlo Mannino, and Leonardo Lamorgese. Recursive logic-based Benders’ decomposition for multi-mode outpatient scheduling. *European Journal of Operational Research*, 255(3):719–728, 2016.
- Vahid Roshanaei, Curtiss Luong, Dionne M Aleman, and David Urbach. Propagating logic-based Benders’ decomposition approaches for distributed operating room scheduling. *European Journal of Operational Research*, 257(2):439–455, 2017a.
- Vahid Roshanaei, Curtiss Luong, Dionne M Aleman, and David R Urbach. Collaborative operating room planning and scheduling. *INFORMS Journal on Computing*, 29(3):558–580, 2017b.
- E. S. Thorsteinsson. Branch-and-check: A hybrid framework integrating mixed integer programming and constraint logic programming. In Toby Walsh, editor, *Principles and Practice of Constraint Programming — CP 2001*, pages 16–30. Springer Berlin Heidelberg, 2001.
- T. T. Tran, A. Araujo, and J. C. Beck. Decomposition methods for the parallel machine scheduling problem with setups. *INFORMS Journal on Computing*, 28(1):83–95, 2016.
- Daniel Tuytens and Arnaud Vandaele. Towards an efficient resolution of printing problems. *Discrete Optimization*, 14:126–146, 2014.
- David Wheatley, Fatma Gzara, and Elizabeth Jewkes. Logic-based Benders decomposition for an inventory-location problem with service constraints. *Omega*, 55:10–23, 2015.
- Ka Fai Cedric Yiu, Kai-Ling Mak, and Henry YK Lau. A heuristic for the label printing problem. *Computers & Operations Research*, 34(9):2576–2588, 2007.