Logic-based Benders Decomposition for Integrated Process Configuration and Production Planning Problems

Karim Pérez Martínez, Yossiri Adulyasak, Raf Jans
HEC Montréal and GERAD, Montréal, QC, Canada.
karim.perez-martinez@hec.ca, yossiri.adulyasak@hec.ca, raf.jans@hec.ca

We propose a general logic-based Benders decomposition (LBBD) for production planning problems with process configuration decisions. This family of problems appears 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 requires determining feasible configurations for the machines and their corresponding production levels to fulfill the demand at the minimum total cost. The structure of this problem contains nonlinear constraints which link the number of units produced of each product with the used configurations and their production levels. We decompose the original problem into a master problem, where the configurations are determined, and a subproblem, where the production amounts are determined. This allows us to apply the LBBD technique to solve the problem using a standard LBBD implementation and a branch-and-check algorithm. LBBD enhancements through logic-based inequalities generated for subsets of products with common characteristics are proposed. Such inequalities represent a form of the subproblem relaxation added to the master problem during its resolution. In our computational experiments, we apply the proposed LBBD approaches to two different applications from the literature: cutting stock problems in the steel industry and a printing problem. Results show that the LBBD methods find optimal solutions much faster than the solution approaches in the literature, and have a superior performance with respect to the number of instances solved to optimality and the solution quality.

Key words: Logic-based Benders decomposition; branch-and-check; integrated production planning; cutting stock.

---

1. Introduction

Many manufacturing environments are designed to flexibly produce different types of products simultaneously to achieve an efficient use of the production equipment. In these contexts, machines are set up according to process configurations that determine the specific combination of different products to produce at the same time. The produced amounts depend on the production level for each used configuration. The setup operations, which change the configurations on the machines, can be costly and time-consuming. Manufacturing environments with these features appear, for instance, in the printing industry (Tuyttens 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), 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 simultaneously produce packages of different shapes.

We address the integrated production planning problem that occurs in the described manufacturing contexts. It 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 its structure, which leads to nonlinear formulations as the produced amounts are determined by the product of the variables related to the configuration decisions and the variables related to their production levels. Furthermore, determining the optimal configurations to be used in a certain production plan can be particularly difficult due to the large number of possibilities and the various technical constraints that ensure the feasibility of the configurations in practice.

In this paper, we present a general mathematical representation for production planning problems with configuration decisions, and develop a logic-based Benders decomposition (LBBD) method to tackle various applications of this class of problems. We decompose the original problem into a master problem, where the configuration decisions are made, and a subproblem, where the production quantities are determined given a set of fixed configurations. The master problem includes a subproblem relaxation which, as commonly implemented in the LBBD literature, consists of inequalities that provide valid bounds on the optimal value of the subproblem to reduce the feasible space of the master problem. In our case, the subproblem relaxation in the master problem is a set of inequalities which impose a valid lower bound on the produced amount of each product. Moreover, we develop enhancements to the LBBD through the implementation of logic-based inequalities, which are used as a further relaxation of the subproblem. These inequalities are derived based on the possibility of grouping products according to similarities in their parameters, and are dynamically added to the master problem during its resolution. Finally, we present extensive computational tests to assess the performance of the LBBD in different applications. We present how the LBBD is applied to solve integrated planning problems in the steel industry and the printing industry, and show that it outperforms existing approaches in the literature for such problems.

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

2. Literature Review

LBBD is an extension of the Benders decomposition (BD) method (Benders 1962) which decomposes a mixed integer optimization problem into a master problem (MP) and one or more subproblems (SP). Unlike in the classical BD, the SP in LBBD is not restricted to be a linear program. Similar to BD, LBBD assigns values to the complicating variables in the MP and finds the best solution given these fixed values in the SP. Instead of solving the dual of the SP to generate the cuts, LBBD solves an inference dual, which uses a logical formalism to deduce a bound on the optimal value of the SP from the fixed values of the complicating variables in the MP and the problem constraints. LBBD provides no standard scheme to generate LBBD cuts (BCs) and as such they must be devised specifically for each problem class (Hooker and Ottosson 2003). There are two common implementations for LBBD. The first one is the standard LBBD implementation, which solves the SP to generate BCs, which are added to the MP, and then resolves the MP to optimality at each iteration (Hooker 2007). The second one is the branch-and-check algorithm (B&Ch), which solves the SP for every integer solution found during the branch-and-bound process of the MP to generate the BCs, which are added to the branch-and-bound tree of the MP (Thorsteinsson 2001).

The flexibility of LBBD has been exploited to tackle a wide range of applications such as production planning problems (Harjunkoski and Grossmann 2001, Hooker 2007, Tran et al. 2016), crane scheduling (Emde et al. 2020, Sun et al. 2019), location problems (Fazel-Zarandi and Beck 2012, Fazel-Zarandi et al. 2013), transportation (Riedler and Raidl 2018), and health care operations planning (Riise et al. 2016, Roshanaei et al. 2017a,b, 2020a,b). Many LBBD implementations were successfully applied for problems which can naturally be decomposed into an assignment problem and a set of scheduling problems. Such implementations typically exploit constraint programming (CP) techniques to solve the SPs (Coban and Hooker 2013, Heching et al. 2019). LBBD presents significant advantages over other solution approaches in this context due to the suitability and efficiency of CP in solving the scheduling problems.

Various computational enhancements are also used to improve the efficiency of LBBD. We next discuss some of these enhancements which were used in the literature.

Using specialized algorithms which exploit the structure of the MP and the SP. Notable examples include: using specialized solvers such as CP optimizers and Concorde according to the structure of the SP (Fazel-Zarandi and Beck 2012, Tran et al. 2016); using highly efficient heuristics when the SP is a classical well-solved problem such as the Bin Packing problem (Fazel-Zarandi and Beck 2012, Roshanaei et al. 2017a); and implementing a multi-level LBBD, where a decomposition technique is applied to the MP or the SP of the LBBD (Wheatley et al. 2015, Riise et al. 2016, Roshanaei et al. 2020a,b).

Devising strong BCs. This type of enhancement focuses mainly on avoiding weak LBBD cuts. An example of this strategy is the derivation of feasibility BCs that remove a larger number of infeasible solutions. For the case where the MP is an assignment problem and the SPs are feasibility problems, this can be achieved by finding the smallest set of assigned tasks for which each SP remains infeasible. This set of tasks is found by using a greedy heuristic that requires solving the SPs repeatedly (Coban and Hooker 2013, Hooker 2007). Another example is the derivation of BCs that provide a nontrivial lower bound on the optimal value of the SP given different fixed values for the complicating variables in the master. This can be achieved by carefully tightening the big-M coefficients used in the BCs (Wheatley et al. 2015).

Devising a tight MP. Part of the SP can be included in the MP through valid inequalities to reduce the feasible space of the MP which allows to obtain solutions that are likely to be feasible with respect to the original problem. Ciré et al. (2016) found that this type of enhancement is the most effective one for improving the LBBD of problems with an assignment-scheduling structure. Other examples can be found in Heching et al. (2019) which explore three sets of valid inequalities (i.e., time windows, assignment, and multicommodity flow) as SP relaxations in the MP which resulted in an improved LBBD.

Developing an effective cut generation strategy. Beck (2010) presents a B&Ch variant (OPT15) where the SP is solved to generate the BCs only when the optimality gap of the MP solution is $\leq 15\%$. Other examples of this type of enhancement are the cut propagation strategies in Roshanaei et al. (2017a) which derive multiple BCs by solving of a single SP.

Different from other applications of LBBD, this paper studies production planning problems with a distinct structure, which includes nonlinear constraints and does not decompose into an assignment problem and scheduling problems. We also develop both standard LBBD and B&Ch algorithms for different problems in manufacturing for which LBBD has not been applied before. Moreover, our study contributes to the state-of-the-art of LBBD for nonlinear problems where, as far as we know, were explored only in the contexts of inventory management (Wheatley et al. 2015) and operating room scheduling (Roshanaei et al. 2020b). Finally, with respect to the common enhancements in LBBD, we introduce a general form of the BCs for the targeted problem class, which is highly challenging to solve due to its combinatorial and nonlinear structure, along with details on how these BCs can be strengthened for each application. We also present an additional SP relaxation through logic-based inequalities which are added to the MP during the solution process.

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 LBBD, and Section 3.4 briefly describes the solution algorithms.

3.1. The Process Configuration and Production Planning Problem

We consider integrated production planning problems with configuration decisions 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, i.e., the combination of different products produced at the same time, and the setup state for each configuration in the machine. These variables are binary and $\mathcal{A}$ denotes their index set. Variable vector $\mathbf{y}$ represents the production planning decisions that define the production level of each used configuration and other planning decisions, such as inventory levels and overproduction, among others. $\mathbf{y}$ can consist of continuous variables, whose index set is denoted by $\mathcal{B}$, and of integer variables, whose index set is denoted by $\mathcal{C}$. Variable vector $\mathbf{q}$ represents the production quantities of each product obtained by each configuration, which are determined in function of the variables for the used configurations and their production levels. These are nonnegative variables and $\mathcal{D}$ denotes their index set. Bold capital letters represent the coefficient matrices of the constraints and bold lowercase letters represent cost vectors, right hand side vectors, and decision variables. We refer to problems in the form (1)–(6) as the original problem (OP) henceforth.

$$\text{OP:} \quad \min \mathbf{c^T x} + \mathbf{d^T y} \tag{1}$$

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

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

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

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

$$\mathbf{x} \in \mathbb{B}^{|\mathcal{A}|};\ \mathbf{y} \in \mathbb{R}_+^{|\mathcal{B}|} \times \mathbb{Z}_+^{|\mathcal{C}|};\ \mathbf{q} \in \mathbb{R}_+^{|\mathcal{D}|} \quad \text{(Domain of the variables)} \tag{6}$$

The objective function (1) minimizes the total costs associated with the configuration and production planning decisions. The configuration costs include setup costs, whereas the total production planning costs can include costs associated with 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 in such a way that a given configuration can only be used if the machine is set up for it. Constraints (4) compute the production quantities. The function $f(\mathbf{x}, \mathbf{y})$ is typically nonlinear, considering that the produced amounts equal the product of the variables for the configuration decisions and their production levels. For instance, in the context of cutting stock problems, the total amount of item A produced by a given cutting pattern equals the number of pieces of type A in such pattern, multiplied by the number of times that the pattern is used. Constraints (5) are related to the production planning decisions, which include demand fulfillment constraints. Note that (4) and (5) could be merged into a single set of nonlinear constraints, yet we present them separately so that their structure is aligned with the structure of the LBBD and the BCs. Constraints (6) define the domain of the variables.

3.2. Standard LBBD Reformulation

We decompose the OP into an MP and an SP and apply the LBBD algorithm to solve it. The MP is a relaxation of the OP where the complicating nonlinear constraints (4) are replaced by linear constraints. In this standard LBBD, the master problem (MP-S) is modeled as follows.

MP-S: $\min \mathbf{c^T x} + \mathbf{d^T y}$

s.t. (2), (3), (5) and (6)

$$\mathbf{Fq} \geq \mathbf{Gx} + \mathbf{Hy} \qquad \text{(Production quantities approximation)} \qquad (7)$$

LBBD cuts (BCs) (8)

Constraints (7), which replace the expression (4) of the OP, impose valid bounds on the production quantities. These represent the SP relaxation, as they aim to approximate $\mathbf{q}$ to values close to $f(\mathbf{x}, \mathbf{y})$ in order to reduce the feasible space of the MP-S and to provide solutions which are likely to be feasible with respect to the OP. Constraints (8) are the BCs, which we discuss in detail later.

The SP is a feasibility problem which ensures that the production amounts are feasible with respect to the OP. More precisely, given an MP-S solution at iteration h, $(\bar{\mathbf{x}}^h, \bar{\mathbf{y}}^h, \bar{\mathbf{q}}^h)$, the SP computes $f(\bar{\mathbf{x}}^h, \bar{\mathbf{y}}^h)$ and checks if $\bar{\mathbf{q}}^h = f(\bar{\mathbf{x}}^h, \bar{\mathbf{y}}^h)$.

The BCs are the constraints added to the MP-S for every violation detected in the SP. These cuts ensure that variables $\mathbf{q}$ take feasible values with respect to the OP if the MP-S assigns the same values to the configuration variables as in the current solution (i.e., if $\mathbf{x} = \bar{\mathbf{x}}^h$). Let $\bar{\mathcal{A}}^h \subseteq \mathcal{A}$ denote the set of indices of variables $\mathbf{x}$ which are equal to 1 in the current solution of the MP-S, $\mathbf{M}$ denote a vector of coefficients which provides upper bounds on the optimal production quantities, and $\theta_{\bar{\mathbf{x}}^h}(\mathbf{x},\mathbf{y})$ denote a linear function which gives feasible values to the production quantities when $\mathbf{x}=\bar{\mathbf{x}}^h$. We introduce inequalities (9) and (10) as the general form of the BCs for the studied family of problems.

$$\text{BCs:} \quad \mathbf{q} \leq \theta_{\bar{\mathbf{x}}^h}(\mathbf{x},\mathbf{y}) + \mathbf{M}\sum_{j\in\bar{\mathcal{A}}^h}(1-x_j) + \mathbf{M}\sum_{j\in\mathcal{A}\setminus\bar{\mathcal{A}}^h} x_j \tag{9}$$

$$\mathbf{q} \geq \theta_{\bar{\mathbf{x}}^h}(\mathbf{x},\mathbf{y}) - \mathbf{M}\sum_{j\in\bar{\mathcal{A}}^h}(1-x_j) - \mathbf{M}\sum_{j\in\mathcal{A}\setminus\bar{\mathcal{A}}^h} x_j \tag{10}$$

The logic of the BCs is such that (9) and (10) give an upper and lower bound on the production quantities, respectively, and both BCs together give $\mathbf{q}=f(\bar{\mathbf{x}}^h,\bar{\mathbf{y}}^h)$ 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 the values $\bar{\mathbf{y}}^h$. Recall that $\mathbf{x}$ are binary variables, hence the second and third term on the right hand side of constraints (9) and (10) become equal to zero if $\mathbf{x}=\bar{\mathbf{x}}^h$. To ensure that the BCs enforces $\mathbf{q}$ to take feasible values, function $\theta_{\bar{\mathbf{x}}^h}$ is required to provide the same value as $f(\mathbf{x},\mathbf{y})$ when $\mathbf{x}=\bar{\mathbf{x}}^h$ and $\mathbf{y}=\bar{\mathbf{y}}^h$. Observe that the general form of the BCs (9) and (10) can be considered as "*standard no-good cuts*" in the sense that they only exclude the current MP solution which is not feasible for the original problem. However, using the function $\theta_{\bar{\mathbf{x}}^h}$ instead of the scalar value $f(\bar{\mathbf{x}}^h,\bar{\mathbf{y}}^h)$ as the first term in the right hand side of the BCs might allow to exclude other infeasible solutions of the MP which are different from the current one (e.g., MP solutions where $\mathbf{x}=\bar{\mathbf{x}}^h$ and $\mathbf{y}\in\mathbb{R}_+^{|\mathcal{B}|}\times\mathbb{Z}_+^{|\mathcal{C}|}$). Finally, the BCs (9) and (10) applied to a specific problem must be valid, i.e., they remove the current infeasible solution of the MP-S and do not remove any feasible solution of the OP (Chu and Xia 2004).

The general form of the BCs can be strengthened by two means: first, by tightening the $\mathbf{M}$ coefficients according to the MP solution at each iteration h ; and second, by identifying the smallest cardinality subsets of $\bar{\mathcal{A}}^h$ and $A\backslash\bar{\mathcal{A}}^h$ (e.g., $\bar{\mathcal{A}}^{h*} \subset \bar{\mathcal{A}}^h$ and $\bar{\mathcal{A}}^{h**} \subset \mathcal{A}\backslash\bar{\mathcal{A}}^h$, respectively) at each iteration h, for which the BCs can remain active without removing feasible solutions. This allows the BCs to provide nontrivial bounds on $\mathbf{q}$ for some cases where $\mathbf{x} \neq \bar{\mathbf{x}}^h$. This will be explained for the specific problems in Section 4.

3.3. Enhanced LBBD Reformulation

We introduce the logic-based subset inequalities (SIs) as an enhancement device for the LBBD. These inequalities benefit from the possibility of grouping products according to their common characteristics derived from the input data (e.g., products of similar size or demand). The SIs allow us to obtain a tighter bound on the total quantities produced by each used configuration based on the MP solution obtained at each iteration. In order to include the SIs, a preprocessing step is required to classify products into a set of groups denoted by $\mathcal{G}$. Moreover, a new set of binary variables, denoted by $\mathbf{z} \in \mathbb{B}^{|\mathcal{G}|}$, and linking constraints need to be added to the MP. Note that the size of $\mathbf{z}$ is known in advance because $\mathcal{G}$ is defined beforehand. The MP in this enhanced LBBD (MP-E) is as follows.

MP-E: $\min \mathbf{c}^{\mathbf{T}}\mathbf{x} + \mathbf{d}^{\mathbf{T}}\mathbf{y}$

s.t. (2)–(3), (5)–(7)

BCs in the form of (9) *and/or* (10)

$$\mathbf{I}\mathbf{x} + \mathbf{J}\mathbf{z} \geq \mathbf{g} \qquad \text{(Linking constraints for the new variables)} \tag{11}$$

Logic-based subset inequalities (SIs) (12)

$$\mathbf{z} \in \mathbb{B}^{|\mathcal{G}|} \qquad \text{(Domain of the new variables)} \tag{13}$$

Constraints (11) link the new variables $\mathbf{z}$ to the existing configuration variables $\mathbf{x}$ in such a way that a variable in $\mathbf{z}$ equals 1 if a given group of products appears in a certain configuration, 0 otherwise. More specifically, a given variable $z_g : g \in \mathcal{G}$ equals 1 if at least one product belonging to group g appears in a given configuration. Constraints (12) are the SIs and constraints (13) define the domain of the new variables accordingly.

The SP and the BCs remain the same as in the standard LBBD. The SIs are separated and added to the problem given an MP-E solution at iteration h, $(\bar{\mathbf{x}}^h, \bar{\mathbf{y}}^h, \bar{\mathbf{q}}^h, \bar{\mathbf{z}}^h)$. We present inequalities (14) as the general form of the SIs. Let $\bar{\mathcal{G}}^{h*} \subset \mathcal{G}$ denote a subset of the indices of variables $\mathbf{z}$ which take value 1 in the current MP-E solution, $\mathbf{M}_{\bar{\mathbf{z}}^h}$ denote a coefficient vector computed based on the values $\bar{\mathbf{z}}^h$, and $\alpha_{\bar{\mathbf{z}}^h}$ denote a linear function of the original variables $(\mathbf{x}, \mathbf{y})$ which gives valid bounds on the total quantities produced by each configuration when $\mathbf{z} = \bar{\mathbf{z}}^h$. The SIs can be written as follows.

$$\text{SIs:} \quad \mathbf{Kq} \geq \alpha_{\bar{\mathbf{z}}^h}(\mathbf{x}, \mathbf{y}) - \mathbf{M}_{\bar{\mathbf{z}}^h} \sum_{g \in \bar{\mathcal{G}}^{h*}} (1 - z_g) \tag{14}$$

The logic of the SIs is such that, if a specific group of products appears in a certain configuration (i.e., $z_g = 1, \forall g \in \bar{\mathcal{G}}^{h*}$), then a tighter bound in the form $\mathbf{Kq} \geq \alpha_{\bar{\mathbf{z}}^h}(\mathbf{x}, \mathbf{y})$ is imposed on the total quantities produced by such configuration. The SIs (14) are only added to the MP-E if they are not satisfied in the current solution, i.e., if $\mathbf{K}\bar{\mathbf{q}}^h < \alpha_{\bar{\mathbf{z}}^h}(\bar{\mathbf{x}}^h, \bar{\mathbf{y}}^h)$.

The SIs are a further SP relaxation that should give tighter bounds than the ones given by constraints (7), and must not remove feasible solutions in order to guarantee optimality.

3.4. LBBD Implementations

We solve the proposed LBBD reformulations using the standard LBBD and the B&Ch implementations. The standard LBBD implementation solves the corresponding MP (i.e., MP-S or MP-E) to optimality, next solves the SP to generate the BCs which are added to the master, and then resolves the MP at each iteration to optimality. When solving the enhanced reformulation, the SIs are also derived and added to the MP after each resolution. At each iteration h, a lower bound and an upper bound on the optimal solution are obtained. The lower bound corresponds to the optimal value of the MP, while an upper bound is obtained by determining feasible values for variables $\mathbf{q}$ which are consistent with the fixed values of the integer variables in the current MP solution. As for the B&Ch implementation, we use the branch-and-bound Lazy Constraints callback of the optimization solver CPLEX (CPLEX Optimization Studio 2019). The SP is solved to generate the BCs when an integer solution is found in the branch-and-bound tree of the MP, as well as the SIs if the enhanced reformulation is being solved. The algorithm terminates when the lower bound equals the upper bound in the search tree of the MP. Similar to the standard implementation, it is possible to obtain an upper bound at the integral nodes of the B&Ch by determining feasible values for variables $\mathbf{q}$ which are consistent with the fixed values of the current MP solution. With the LBBD cuts (9) and (10), CPLEX can directly obtain such upper bound once the cuts are added. In addition to this, one can also set the upper bound using the Heuristic callback (CPLEX Optimization Studio 2019). Nevertheless, this strategy is proven to be redundant in our experiments as CPLEX could identify the upper bound which is aligned with the incumbent solution determined during our solution process. In addition, the preliminary results on a subset of instances show that the solution time when using the Heuristic callback on top of the Lazy Constraints callbacks is slightly higher overall due to additional computing time to verify and pass along the solution to CPLEX in our case.

4. Applications of the LBBD Method

We applied the proposed LBBD to different problems in the literature where process configuration and production planning decisions are jointly made. Sections 4.1 and 4.2 describe 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): the open-ended CSCP, the closed-ended CSCP, and the CSCP with knife-dependent setups. These problems consider a continuous steel tube that has to be cut into smaller pieces using a cutting machine with a limited length. The machine is set up according to a cutting pattern, which defines a specific arrangement of the knives in the machine to cut a combination of pieces at the same time. The combination of pieces to be cut at the same time can be of the same type or different types, i.e., it is possible to obtain more than one piece of a certain type from a single pattern use. The main decisions consist of determining: (1) the number of distinct cutting patterns to be used; (2) the configuration of each pattern (i.e., the number of pieces of each type to be cut by each pattern); and (3) the number of times that each pattern is used (i.e., the number of pattern repetitions). The objective is to minimize the total cutting time and setup time. In the open-ended CSCP, the cutting machine has no ending barrier blocking the continuous tube, so that one extra piece of any length can be produced per repetition of any pattern. In the closed-ended CSCP, no extra piece is produced. Figure 1 illustrates the design of the machines for these two variants. In the CSCP with knife-dependent setups, setting up a pattern entails an extra variable time which is proportional to the number of used knives. For further details see Hajizadeh and Lee (2007) and Section 1 of the online supplement.

We focus this section on the open-ended CSCP, and mention how the LBBD is applied to the other variants. The OP as defined by Hajizadeh and Lee (2007) is presented below. Table 1 presents the parameters and variables in this formulation.

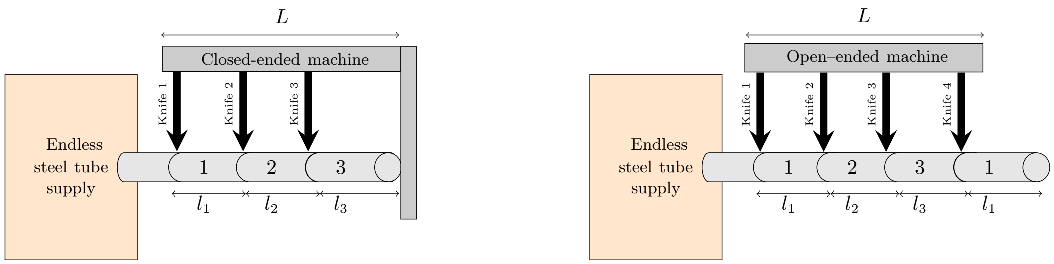

Figure 1 **Closed-ended and open-ended configurations for cutting machines**

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

Table 1 **Parameters and variables for the open-ended CSCP**

$$\text{OP (Open-ended CSCP)}: \quad \min C_s \sum_{j\in P} s_j + C_r \sum_{j\in P} z_j \tag{15}$$

$$\mathbf{Ax} \geq \mathbf{a}: \quad \sum_{i\in T} l_i a_{ij} \leq L s_j \qquad \forall j \in P \tag{16}$$

$$\mathbf{Bx}+\mathbf{Cy} \geq \mathbf{b}: \quad z_j \leq M s_j \qquad \forall j \in P \tag{17}$$

$$\mathbf{q} = f(\mathbf{x},\mathbf{y}): \quad q_{ij} = a_{ij} z_j \qquad \forall i \in T; j \in P \tag{18}$$

$$\mathbf{Dy}+\mathbf{Eq} \geq \mathbf{e}: \quad \sum_{j\in P} q_{ij} + x_i \geq d_i \qquad \forall i \in T \tag{19}$$

$$\sum_{j\in P} z_j \geq \sum_{i\in T} x_i \tag{20}$$

$$z_{j-1} \leq z_j \qquad \forall j \in P : j > 1 \tag{21}$$

$$\mathbf{x} \in \mathbb{B}^{|\mathcal{A}|}: \quad s_j \in \{0,1\},\ a_{ij} \in \mathbb{Z}_+ \qquad \forall i \in T; j \in P \tag{22}$$

$$\mathbf{y} \in \mathbb{R}_+^{|\mathcal{B}|} \times \mathbb{Z}_+^{|\mathcal{C}|};\ \mathbf{q} \in \mathbb{R}_+^{|\mathcal{D}|}: \quad z_j \in \mathbb{Z}_+,\ x_i \geq 0;\ q_{ij} \geq 0 \qquad \forall i \in T; j \in P \tag{23}$$

The objective function (15) minimizes the total setup and cutting time. Constraints (16) are the configuration constraints, which ensure that the sum of the length of the pieces in a used pattern does not exceed the machine length. Constraints (17) are the linking constraints to ensure that repetitions of a pattern occur only if such pattern is used, which corresponds to the pattern setup decision. Constraints (18) compute the amount of each product produced by each pattern. Note that these constraints are nonlinear. Constraints (19)–(21) are production planning constraints. Constraints (19) ensure that the demand is fulfilled by the total amounts produced by the patterns, and by the extra pieces. Constraints (20) ensure that the total number of extra pieces is less than or equal to the total number of pattern repetitions, as at most one extra piece can be produced per repetition of any pattern. Inequalities (21) are symmetry-breaking constraints to order the used patterns according to the number of repetitions. Finally, constraints (22) and (23) define the domain of the variables. Note that, some of the configuration variables are integer. This is not aligned with the general form of the OP, hence we use a binary substitution to represent these integer variables in order to apply the LBBD.

The OP for the closed-ended CSCP is defined as a special case of the OP for the open-ended CSCP 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 OP (15)–(23) which includes additional setup variables to represent the extra setup time linked to the number of used knives. See Hajizadeh and Lee (2007) and Section 1.1.2 of the online supplement for details.

4.1.1. Standard LBBD. We proceed to define the MP-S, SP, and the BCs for the open-ended CSCP according to the general LBBD in Section 3.2. First, we ensure that the OP is aligned with its general form, so that all the configuration variables are binary. We substitute the integer variables a_{ij} by a sum of powers of two as presented in (24). Let K_i (indexed by k) be the set of bins required to represent integer variables a_{ij}, i.e., $K_i = \{1, ..., n_i\}$ where n_i is the minimum value such that $\sum_{k=1}^{n_i} 2^{k-1} \geq \left\lfloor \frac{L}{l_i} \right\rfloor$, and parameter $b_k = 2^{k-1}$. Variable w_{ijk} equals 1 if k is used in writing a_{ij}, 0 otherwise.

$$a_{ij} = \sum_{k \in K_i} b_k w_{ijk} \qquad \forall i \in T;\, j \in P \tag{24}$$

The MP-S is modeled as the OP, where the nonlinear constraints (18) are replaced by an approximation for the production quantities, the variables a_{ij} are substituted as described above, and the BCs are added. Therefore, the MP-S consist of: the objective function, linking constraints and production planning constraints as in the OP; the configuration constraints (25) that replace the original constraints (16); constraints (26) and (27) as an approximation for the production quantities, where $N_i = \left\lfloor \frac{L}{l_i} \right\rfloor$ is the maximum number of pieces i that can be allocated to any pattern; the BCs; and the domain constraints (28). See Section 1.1 of the online supplement for a full presentation of the MPs in this application.

$$\text{MP-S (Open-ended CSCP):} \quad \min C_s \sum_{j \in P} s_j + C_r \sum_{j \in P} z_j$$

$$\mathbf{Ax} \geq \mathbf{a}: \quad \sum_{i \in T} l_i \sum_{k \in K_i} b_k w_{ijk} \leq L s_j \qquad \forall j \in P \tag{25}$$

$$\mathbf{Bx} + \mathbf{Cy} \geq \mathbf{b}: \quad (17)$$

$$\mathbf{Dy} + \mathbf{Eq} \geq \mathbf{e}: \quad (19), (20) \text{ and } (21)$$

$$\mathbf{Fq} \geq \mathbf{Gx} + \mathbf{Hy}: \quad q_{ij} \leq M_i \sum_{k \in K_i} w_{ijk} \qquad \forall i \in T;\, j \in P \tag{26}$$

$$\sum_{l \in T: l \geq i} q_{lj} \leq N_i z_j \qquad \forall i \in T; j \in P \tag{27}$$

BCs: The BCs for the open-ended CSCP

$$\mathbf{x} \in \mathbb{B}^{|\mathcal{A}|}: \quad s_j \in \{0,1\}; w_{ijk} \in \{0,1\} \qquad \forall i \in T; j \in P; k \in K_i \tag{28}$$

$$\mathbf{y} \in \mathbb{R}_+^{|\mathcal{B}|} \times \mathbb{Z}_+^{|\mathcal{C}|};\, \mathbf{q} \in \mathbb{R}_+^{|\mathcal{D}|}: \quad (23)$$

Constraints (26) and (27) are the SP relaxations which impose valid bounds on the amount of product i produced by pattern j. Constraints (26) ensure that $q_{ij} = 0$ if no piece of type i is allocated to pattern j. In case that product i is allocated to pattern j, q_{ij} is limited by M_i, which is the maximum value between the demand of product i

and the demand of any other product possibly allocated to the same pattern, i.e., $M_i = \max\left\{d_i; \max_{i' \in T: l_{i'} \leq L - l_i}\{d_{i'}\}\right\}$. Constraints (27) link the total amount produced by a pattern for a subset of products with the number of repetitions of such pattern, where products are assumed to be ordered from the smallest to the largest size (i.e., $N_i \leq N_{i-1}, \forall i \in T: i > 1$). These constraints impose that the total amount of products $\{i, \ldots, |T|\}$ produced by pattern j is at most the number of repetitions of such pattern z_j, multiplied by N_i, which is the maximum total number of pieces that can fit into any 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 i that can fit into pattern j multiplied by the number of repetitions of that pattern.

Given an MP-S solution at iteration h, $(\bar{s}_j^h, \bar{w}_{ijk}^h, \bar{z}_j^h, \bar{x}_i^h, \bar{q}_{ij}^h)$, the SP checks if the production amounts are feasible. This SP decomposes by product and by pattern, so that each SP computes $\bar{a}_{ij}^h = \sum_{k \in K_i} b_k \bar{w}_{ijk}^h$, i.e., the number of pieces i in pattern j, and checks if $\bar{q}_{ij}^h = \bar{z}_j^h \bar{a}_{ij}^h$.

Inequalities (29) are the BCs for the open-ended CSCP. Let $\bar{K}_{ij}^h = \{k \in K_i : \bar{w}_{ijk}^h = 1\}$ denote the set of indices of variables w_{ijk} which take the value 1 in the current solution of the MP-S, which implies $K_i \setminus \bar{K}_{ij}^h = \{k \in K_i : \bar{w}_{ijk}^h = 0\}$. The BCs are added for each violation detected by the SPs, i.e., for each element in set $\mathcal{Q}_h = \{(i,j) : i \in T, j \in P, \bar{q}_{ij}^h \neq \bar{z}_j^h \bar{a}_{ij}^h\}$, and $\forall\, h \in \mathcal{H}'$, where $\mathcal{H}'$ corresponds to the index set of the MP-S solutions considered so far.

$$\text{BCs (Open-ended CSCP):} \quad q_{ij} \leq z_j \bar{a}_{ij}^h + M_i \sum_{k \in K_i \setminus \bar{K}_{ij}^h} w_{ijk} \qquad \forall h \in \mathcal{H}'; (i,j) \in \mathcal{Q}_h \tag{29}$$

The BCs (29) are a strengthened version of the following BCs: $q_{ij} \leq z_j \bar{a}_{ij}^h + M_i \sum_{k \in \bar{K}_{ij}^h} (1 - w_{ijk}) + M_i \sum_{k \in K_i \setminus \bar{K}_{ij}^h} w_{ijk}$, $\forall h \in \mathcal{H}'; (i,j) \in \mathcal{Q}_h$, which are aligned with the general form presented in inequalities (9). The proposed BCs (29) impose that the amount of product i produced by pattern j never exceeds $z_j\bar{a}^h_{ij}$ if the number of pieces i in pattern j equals $\bar{a}^h_{ij}$ (i.e., $\sum_{k\in K_i\setminus\bar{K}^h_{ij}} w_{ijk}=0$ and $\sum_{k\in\bar{K}^h_{ij}}(1-w_{ijk})=0$). The BCs (29) together with constraints (19), which ensure that the demand of product i is fulfilled, enforce that the production amounts are feasible with respect to the OP. Note that, by omitting the term $M_i\sum_{k\in\bar{K}^h_{ij}}(1-w_{ijk})$ which appears in the general form of the BCs, the strengthened BCs (29) give a valid upper bound on q_{ij} when the number of pieces i in pattern j is different from $\bar{a}^h_{ij}$. In particular, for solutions where the number of pieces i in pattern j is less than $\bar{a}^h_{ij}$ (i.e., $\sum_{k\in K_i\setminus\bar{K}^h_{i}} w_{ijk}=0$ and $\sum_{k\in\bar{K}^h_{i}}(1-w_{ijk})\geq 1$), the BCs (29) ensure $q_{ij}\leq z_j\bar{a}^h_{ij}$. We prove the validity of the BCs (29) in Section 1.2 of the online supplement.

4.1.2. Enhanced LBBD. We present the MP-E and the SIs for the open-ended CSCP. As described in Section 3.3, the enhanced LBBD requires classifying products into groups according to similar characteristic in their input data. For this problem, we classify the products into a set of groups denoted by $\mathcal{G}$ (indexed by g), according to the maximum number of pieces that can fit into a pattern, i.e., products with the same value of $N_i=\left\lfloor\frac{L}{l_i}\right\rfloor$ are allocated to the same group. Let T_g denote the set of products allocated to group g and N_g denote the maximum number of pieces of any product in group g that can fit into any pattern, hence $T_g=\left\{i:i\in T,\left\lfloor\frac{L}{l_i}\right\rfloor=N_g\right\}$.

The MP-E is defined by adding to the MP-S a new set of binary variables y_{gj} that indicates the presence of group g in pattern j, a set of constraints that link these new variables to the existing configuration variables, and the SIs. The MP-E is therefore formulated as presented below. Constraints (30) and (31) link the new variables with the configuration variables w_{ijk}, such that $y_{gj}=1$ if at least one piece of type $i\in T_g$ is assigned to pattern

j and $y_{gj}=0$ otherwise. Parameter $B_g=\sum_{i\in T_g}|K_i|$. Constraints (32) represent the SIs, which we describe next, and constraints (33) define the domain of the new variables.

$$
\begin{aligned}
&\text{MP-E (Open-ended CSCP):} && \min C_s\sum_{j\in P}s_j + C_r\sum_{j\in P}z_j \\
&\mathbf{Ax}\geq\mathbf{a}: && (25) \\
&\mathbf{Bx}+\mathbf{Cy}\geq\mathbf{b}: && (17) \\
&\mathbf{Dy}+\mathbf{Eq}\geq\mathbf{e}: && (19),(20) \text{ and } (21) \\
&\mathbf{Fq}\geq\mathbf{Gx}+\mathbf{Hy}: && (26) \text{ and } (27) \\
&\textit{BCs:} && \text{BCs for the open-ended CSCP } (29) \\
&\mathbf{x}\in\mathbb{B}^{|\mathcal{A}|};\mathbf{y}\in\mathbb{R}_+^{|\mathcal{B}|}\times\mathbb{Z}_+^{|\mathcal{C}|};\mathbf{q}\in\mathbb{R}_+^{|\mathcal{D}|}: && (28) \text{ and } (23) \\
&\mathbf{Ix}+\mathbf{Jz}\geq\mathbf{g}: && \sum_{i\in T_g}\sum_{k\in K_i}w_{ijk}\leq B_g y_{gj} \qquad g\in\mathcal{G};j\in P \qquad (30) \\
& && y_{gj}\leq\sum_{i\in T_g}\sum_{k\in K_i}w_{ijk} \qquad \forall g\in\mathcal{G};j\in P \qquad (31) \\
&\textit{Logic-based SIs}: && \text{SIs for the open-ended CSCP} \qquad (32) \\
&\mathbf{z}\in\mathbb{B}^{|\mathcal{G}|}: && y_{gj}\in\{0,1\} \qquad \forall g\in\mathcal{G};j\in P \qquad (33)
\end{aligned}
$$

Given an MP-E solution at iteration h, $(\bar{s}^h_j,\bar{w}^h_{ijk},\bar{z}^h_j,\bar{x}^h_i,\bar{q}^h_{ij},\bar{y}^h_{gj})$, the SPs and the BCs remain the same as in the standard LBBD, while the SIs are derived as follows. Let $\bar{P}^h=\{j: j\in P,\, \bar{s}^h_j=1\}$ denote the set of used patterns in the current solution, and $\bar{\mathcal{G}}^h_j=\{g: g\in\mathcal{G},\, \bar{y}^h_{gj}=1\}, \forall j\in\bar{P}^h$ denote the set of groups allocated to pattern j. From set $\bar{\mathcal{G}}^h_j$, we separate the group with the smallest N_g to define the subset $\bar{\mathcal{G}}^{h*}_j=\left\{g: g\in \arg\min_{g'\in\bar{\mathcal{G}}^h_j}\{N_{g'}\}\right\}, \forall j\in\bar{P}^h$. This means that, set $\bar{\mathcal{G}}^{h*}_j$ contains the group to which the product with the largest size among the products allocated to pattern j belongs. Note that $\bar{\mathcal{G}}^{h*}_j\subseteq\bar{\mathcal{G}}^h_j$ and $|\bar{\mathcal{G}}^{h*}_j|=1$. We use this information to derive an upper bound on the total amount of products that do not belong to that group $g\in\bar{\mathcal{G}}^{h*}_j$ produced by pattern j.

The SIs are formulated as inequalities (34). These impose that, if in an MP-E solution at least one of the products in group $g: g\in\bar{\mathcal{G}}^{h*}_j$ is assigned to pattern j, then the total amount of products that do not belong to this group produced by pattern j (i.e., $\sum_{i\in T\setminus T_g} q_{ij}$) is limited by the pattern repetition multiplied by the maximum number of pieces of any type that can coexist with any product in $g \in \bar{\mathcal{G}}_j^{h*}$. Parameters C_g and D_g are computed as follows: $C_g = \left\lfloor \frac{L-\min_{i\in T_g}\{l_i\}}{l_1} \right\rfloor$ is the maximum number of pieces of any type that can coexist with one piece of any product in group g; and D_g is computed as $D_g = M\left(\left\lfloor \frac{L}{l_1} \right\rfloor - C_g\right)$. Recall that products are ranked by their size from the smallest to the largest, hence l_1 is the size of the smallest product, and $M = \max_{i\in T}\{d_i\}$ is an upper bound on variables z_j. As for the BCs, the SIs are derived $\forall h \in \mathcal{H}'$, where $\mathcal{H}'$ is the index set of the MP-E solutions considered so far, and added only if violated in the current solution, i.e., if $\sum_{i\in T\setminus T_g} \bar{q}_{ij}^h > C_g \bar{z}_j^h$. We prove the validity of the SIs (34) in Section 1.3 of the online supplement.

$$\text{SIs (Open-ended CSCP): } \sum_{i\in T\setminus T_g} q_{ij} \le C_g z_j + D_g(1-y_{gj}) \quad \forall h \in \mathcal{H}';\, j \in \bar{P}^h;\, g \in \bar{\mathcal{G}}_j^{h*} \quad (34)$$

We conclude this section by mentioning that the LBBD for the open-ended can be applied to the closed-ended CSCP by making $x_i = 0, \forall i \in T$. For the CSCP with knife-dependent setup, the MPs need to be modified according to the conditions of this variant as in Hajizadeh and Lee (2007). The BCs and the SIs can be implemented without change.

4.1.3. Computational Results. This section presents the computational results for the application in the steel tube industry. We used Python 3.7 and solver CPLEX 12.9 with the default setting and one thread, on a workstation Intel E5-2683/2.1GHz with 16GB of RAM. The data sets consist of the 47 benchmark instances proposed by Hajizadeh and Lee (2007) and 50 new instances generated based on the guidelines in that original paper. The problem data are available at `https://github.com/Karim-Perez/process-configuration-problems`. We classify these instances into four sets according to their size as presented in Table 2. The computing time limit is 1800 seconds, as for the solution method in the original study.

		Set A	Set B	Set C	Set D
Number of products	$(\|T\|)$	$\{6,7\}$	$\{10,14\}$	$\{18,20\}$	$\{25,30\}$
Number of patterns	$(\|P\|)$	$\{4,7\}$	10	$\{18,20\}$	$\{25,30\}$
Number of groups	$(\|\mathcal{G}\|)$	$[2,5]$	$[4,6]$	$[4,7]$	$[4,7]$
Number of instances		40	17	20	20

Table 2 Data sets for the cutting problems in the steel tube industry

Tables 3 and 4 present the average results of the solution approach presented in Hajizadeh and Lee (2007) (*H&L (2007)*), who solve a linearized version of the OP solved using CPLEX, the standard LBBD (*LBBD*) and the B&Ch (*B&Ch*) implementations of the standard LBBD method (*Stand. (BCs)*) and the enhanced LBBD method (*Enh. (BCs+SIs)*). To properly carry out comparisons, we reproduced the findings in H&L (2007) using Python 3.7 and CPLEX 12.9 as for the LBBD. Let lb be the lower bound and ub be the upper bound on the optimal solution obtained by a solution method for an instance. The following statistics are reported for each method and data set: the average normalized lower bound (nLB), which is computed as $\frac{lb}{ub^*}$ for each instance, where ub^* is the best ub of the instance among all the methods considered; the average optimality gap (Gap), which is computed as $\frac{(ub-lb)}{ub} \times 100\%$ for each instance; the average computing time in seconds ($Time$) over all instances for which an optimal or feasible solution is provided; the average total number of BCs (BCs); the average total number of SIs (SIs); the total number of instances solved to optimality (OS); and, only for the large data sets where some methods are unable to provide any solution, we report the total number of instances for which a feasible solution was found (FS). Bold numbers indicate the best $Time$ when all the instances in the set were solved to optimality, and the best *nLB* and best *Gap* for the sets where not all the instances were solved to optimality.

		Set A						Set B					
Solution Method		nLB	Gap	Time	BCs	SIs	OS	nLB	Gap	Time	BCs	SIs	OS
Open-ended CSCP													
H&L (2007)		1.000	0.0%	9.96	-	-	40/40	0.784	21.7%	1534.04	-	-	3/17
LBBD	Stand. (BCs)	1.000	0.0%	2.26	17.1	-	40/40	**1.000**	0.1%	332.22	22.5	-	16/17
	Enh. (BCs+SIs)	1.000	0.0%	2.44	17.3	1.9	40/40	**1.000**	**0.0%**	92.00	22.1	1.9	17/17
B&Ch	Stand. (BCs)	1.000	0.0%	**1.13**	24.3	-	40/40	0.984	1.7%	660.23	41.5	-	13/17
	Enh. (BCs+SIs)	1.000	0.0%	1.35	23.8	2.3	40/40	**1.000**	0.2%	388.56	40.0	3.4	16/17
Closed-ended CSCP													
H&L (2007)		1.000	0.0%	3.23	-	-	40/40	0.953	4.7%	684.79	-	-	14/17
LBBD	Stand. (BCs)	1.000	0.0%	4.97	25.7	-	40/40	**0.993**	1.3%	476.43	22.8	-	14/17
	Enh. (BCs+SIs)	1.000	0.0%	5.42	25.4	2.3	40/40	**0.993**	**1.1%**	386.50	23.4	1.3	14/17
B&Ch	Stand. (BCs)	1.000	0.0%	**2.06**	33.5	-	40/40	0.988	1.4%	450.37	47.1	-	14/17
	Enh. (BCs+SIs)	1.000	0.0%	2.13	31.4	2.3	40/40	0.986	1.5%	419.09	44.2	2.8	14/17
CSCP with knife-dependent setups													
H&L (2007)		1.000	0.0%	26.36	-	-	40/40	0.724	27.7%	1710.97	-	-	2/17
LBBD	Stand. (BCs)	1.000	0.0%	21.84	19.6	-	40/40	0.991	1.9%	828.53	23.7	-	12/17
	Enh. (BCs+SIs)	1.000	0.0%	**13.33**	19.3	2.1	40/40	**0.998**	**0.8%**	568.30	26.4	2.1	14/17
B&Ch	Stand. (BCs)	1.000	0.0%	16.21	27.7	-	40/40	0.965	4.1%	1117.59	50.2	-	8/17
	Enh. (BCs+SIs)	1.000	0.0%	18.07	26.1	2.4	40/40	0.975	2.9%	824.10	47.9	3.6	10/17

Table 3 Average results for the cutting stock problems in the steel tube industry (Sets A and B)

Results in Table 3 show that the instances in Set A can be solved to optimality in less than 27 seconds on average for the three problems. However, except for the standard LBBD implementation in the closed-ended CSCP, the LBBD methods find optimal solutions in shorter computing times. On average, the best performing LBBD solves these instances 8.8 times, 1.6 times, and 2 times faster than the approach in H&L (2007) for the open-ended CSCP, the closed-ended CSCP, and the CSCP with knife-dependent setups, respectively.

The results for the data sets B, C, and D show that, for all the three problem variants, the LBBD provides significant improvements with respect to the lower bounds obtained at the end of the execution. This can be seen by the differences in the nLB between the

Open-ended CSCP														
		Set C						Set D						
Solution Method		nLB	Gap	Time	BCs	SIs	OS	nLB	Gap	Time	BCs	SIs	OS	FS
H&L (2007)		0.460	54.2%	1800.00	-	-	0/20	0.366	63.7%	1800.00	-	-	0/20	20/20
LBBD	Stand. (BCs)	0.928	10.2%	1717.42	23.1	-	1/20	0.853	16.6%	1624.17	10.6	-	2/20	20/20
	Enh. (BCs+SIs)	**0.949**	**7.2%**	1523.99	34.8	4.3	5/20	**0.887**	**13.5%**	1623.67	12.4	1.4	2/20	20/20
B&Ch	Stand. (BCs)	0.868	14.8%	1800.00	81.6	-	0/20	0.824	21.5%	1628.17	159.7	-	2/20	20/20
	Enh. (BCs+SIs)	0.887	13.0%	1715.07	86.4	6.5	1/20	0.828	21.5%	1632.20	142.5	5.9	2/20	20/20
Closed-ended CSCP														
H&L (2007)		0.331	67.1%	1800.00	-	-	0/20	0.250	75.2%	1800.00	-	-	0/20	19/20
LBBD	Stand. (BCs)	0.905	13.0%	1560.75	11.7	-	3/20	0.868	**20.6%**	1581.94	8.9	-	3/20	20/20
	Enh. (BCs+SIs)	**0.936**	**10.4%**	1475.33	16.8	2.1	4/20	**0.876**	21.5%	1617.66	9.2	0.6	2/20	20/20
B&Ch	Stand. (BCs)	0.846	18.2%	1726.87	170.8	-	1/20	0.850	23.5%	1796.13	343.7	-	1/20	20/20
	Enh. (BCs+SIs)	0.853	19.3%	1733.80	136.5	12.2	2/20	0.849	26.6%	1792.15	320.4	9.1	1/20	18/20
CSCP with knife-dependent setups														
H&L (2007)		0.427	57.4%	1800.0	-	-	0/20	0.334	66.7%	1800.00	-	-	0/20	20/20
LBBD	Stand. (BCs)	0.906	14.1%	1800.23	16.0	-	0/20	0.849	18.2%	1626.58	9.8	-	2/20	20/20
	Enh. (BCs+SIs)	**0.935**	**9.4%**	1719.45	29.1	3.6	1/20	**0.878**	**15.8%**	1639.92	10.5	1.1	2/20	20/20
B&Ch	Stand. (BCs)	0.856	17.1%	1800.00	111.2	-	0/20	0.824	22.1%	1800.00	227.4	-	0/20	20/20
	Enh. (BCs+SIs)	0.875	14.9%	1732.33	106.1	9.7	1/20	0.829	22.8%	1727.21	227.4	8.2	1/20	20/20

Table 4 **Average results for the cutting stock problems in the steel tube industry (Sets C and D)**

LBBD and the approach in H&L (2007), where the closer the nLB is to 1, the better the lower bounds are with respect to the best found solutions. Unlike in the results for the small instances, the enhanced LBBD presents a clear advantage over the standard LBBD for most cases in Sets B, C and D. In particular, the LBBD Enh. (BCs+SIs) is able to optimally solve a larger number of instances and to substantially outperform the approach in H&L (2007). For the open-ended CSCP, it solves the instances in Set B within 92 seconds on average, whereas H&L (2007) finds solutions with an average gap of 21.7% and its corresponding CPU time is 1534 seconds on average. The LBBD could reduce the average gaps by approximately 47.0% and 50.2% for Sets C and D, respectively. For the closed-ended CSCP, its improvements with respect to the lower bounds could reduce the average gap by 3.6%, 56.7%, and 54.6% for the sets B, C and D, respectively. Finally, for the CSCP with knife-dependent setups, the enhanced LBBD could reduce the average gap by 26.9%, 48%, and 50.9% for Sets B, C, and D, respectively.

We consider that the standard LBBD implementation is superior to the B&Ch implementation in this application due to the fact that the structure of the MP does not have many technical constraints, which makes it easier to solve. Unlike in the OP for Sets A and B, the closed-ended CSCP is harder to solve than the open-ended CSCP by using the LBBD. This may be explained by the MP in the LBBD for the closed-ended problem which does not allow extra pieces to compose the total production quantities, hence it has less flexibility to provide solutions which are feasible with respect to the OP.

4.2. Application in the Printing Industry

This application considers the production planning problem studied in Baumann and Trautmann (2014) and Baumann et al. (2015). The problem is inspired by a real-world offset printing process that produces napkin pouches. Pouches with customer-specific designs (CSDs) are make-to-order products, while pouches with standard designs (SDs) are make-to-stock. Both design types are imprinted by offsetting the inked images from rotating printing plates to the surface of the paper. The printing plates used consist of seven slots, where an individual design must be allocated to each slot, as depicted in Figure 2. There is a specific demand for the CSDs, which must be satisfied, but it is possible to produce more than the required demand. For the SDs, no specific demand is assumed in this particular problem, but at most one slot not occupied by CSDs in each plate can be filled with an SD. Various technical constraints related to incompatibilities between the printing equipment and the design features (e.g., type of design, color code, and white border layout) are considered to ensure that the configurations of the printing plates are feasible in practice.

The main decisions aim to determine: (1) the number of distinct printing plates to be used; (2) the allocation of designs to the slots of these plates; and (3) the number

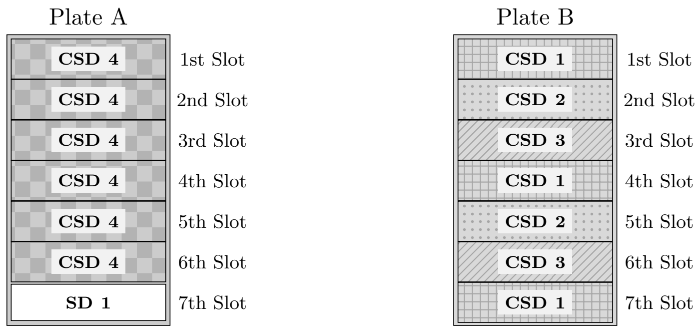

Figure 2 Examples of feasible configurations for a printing plate

of rotations of each printing plate, i.e., the number of times that the plate is used. The objective is to minimize the total overproduction costs and setup costs. We present the OP below as the nonlinear version of the formulation in Baumann and Trautmann (2014). Table 5 presents the parameters and variables of the OP.

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 $= \{1, \dots, \lvert J\rvert\}$ (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
I^O	customer-specific designs (CSDs);	c_i^S	overproduction cost per unit			on plate p; 0 otherwise;
I^S	standard designs (SDs);		of standard design i;	$\mathbf{y}:$	r_p	rotations of plate p;
I_c	designs with color code c;	$\bar{c}$	maximum number of different		v_i	number of overproduced units of design i;
I_w	designs with white border layout;		color codes per plate;	$\mathbf{q}:$	q_{ip}	number of units of design i produced by plate p.

Table 5 Parameters and variables for the printing problem

$$\text{OP (Printing)}: \quad \min \sum_{p \in P} c^P W_p + \sum_{i \in I^O} c_i^O v_i + \sum_{i \in I^S} c_i^S v_i \tag{35}$$

$$\mathbf{Ax} \geq \mathbf{a}: \quad \sum_{i \in I} \sum_{n \in J} n K_{inp} = |J| W_p \qquad \forall p \in P \tag{36}$$

$$\sum_{i \in I^S} \sum_{n \in J} n K_{inp} \leq 1 \qquad \forall p \in P \tag{37}$$

$$\sum_{i \in I_w} \sum_{n \in J} \frac{1}{2} n K_{inp} + \sum_{i \in I^S} \sum_{n \in J} K_{inp} \geq W_p \qquad \forall p \in P \tag{38}$$

$$\sum_{c \in C} Z_{cp} \leq \bar{c} \qquad \forall p \in P \tag{39}$$

$$\sum_{i \in I_c} \sum_{n \in J} K_{inp} \leq |J| Z_{cp} \qquad \forall c \in C;\, p \in P \tag{40}$$

$$\sum_{p \in P} \sum_{n \in J} K_{inp} = 1 \qquad \forall i \in I^O \tag{41}$$

$$W_{p-1} \geq W_p \qquad \forall p \in P : p > 1 \qquad (42)$$

$$\mathbf{q} = f(\mathbf{x}, \mathbf{y}): \quad q_{ip} = r_p \sum_{n \in J} n K_{inp} \qquad \forall i \in I; p \in P \qquad (43)$$

$$\mathbf{D}\mathbf{y} + \mathbf{E}\mathbf{q} \geq \mathbf{e}: \quad \sum_{p \in P} q_{ip} = d_i + v_i \qquad \forall i \in I \qquad (44)$$

$$\mathbf{x} \in \mathbb{B}^{|\mathcal{A}|}: \quad W_p,\, K_{inp},\, Z_{cp} \in \{0,1\} \qquad \forall i \in I;\, p \in P;\, n \in J;\, c \in C \qquad (45)$$

$$\mathbf{y} \in \mathbb{R}_+^{|\mathcal{B}|} \times \mathbb{Z}_+^{|\mathcal{C}|}: \quad r_p,\, v_i \geq 0 \qquad \forall i \in I;\, p \in P \qquad (46)$$

$$\mathbf{q} \in \mathbb{R}_+^{|\mathcal{D}|}: \quad q_{ip} \geq 0 \qquad \forall i \in I;\, p \in P \qquad (47)$$

The objective function (35) minimizes the total setup costs and overproduction costs. Constraints (36)–(42) are the configuration constraints. Constraints (36) ensure that every slot of a used plate is occupied by one design, i.e, no empty slots are allowed for a used plate. Constraints (37) guarantee that at most one slot per plate is occupied by an SD. Inequalities (38) enforce the white border layout technical constraints, which impose that each used plate must allocate at least two slots to the designs with white border layout and/or allocate an SD to one slot. Constraints (39) and (40) prevent that more than $\bar{c}$ different color codes are allocated to the same plate. Constraints (41) prohibit that a CSD is allocated to different plates, so that the total demand for each design of this type is always fulfilled by a single plate. Constraints (36)–(41) are all the technical and organizational constraints imposed on the configuration of the plates. For the details and justification of these constraints, we refer the reader to Section 2 of the online supplement and to Baumann et al. (2015). Inequalities (42) are symmetry-breaking constraints that ensure that the plates with the smallest indices are used first. Constraints (43) compute the amount of each design produced by each plate. Note that these constraints are nonlinear. Constraints (44) are the production planning constraints that determine the number of overproduced units for each design while making sure that the demand is satisfied. Finally, constraints (45)–(47) define the domain of the variables.

4.2.1. Standard LBBD. We discuss how the MP-S and the BCs are formulated for this application. To obtain the MP-S, the nonlinear constraints (43) in the OP are replaced by constraints (48) and (49), and the BCs are added. Parameter M is an upper bound on the number of rotations of a printing plate, computed as $M = \max_{i\in I}\{d_i\}$. Note that variables r_p are not used in the MP-S, leaving the number of rotations of each plate to be computed in the SP. The MP-S is presented below. A full version of the MP for this application is presented in Section 2.1 of the online supplement.

$$
\begin{aligned}
&\text{MP-S (Printing problem):} && \min \sum_{p\in P} c^P W_p + \sum_{i\in I^O} c_i^O v_i + \sum_{i\in I^S} c_i^S v_i && \\
&\mathbf{Ax} \geq \mathbf{a}: && (36) - (42) && \\
&\mathbf{Dy} + \mathbf{Eq} \geq \mathbf{e}: && (44) && \\
&\mathbf{Fq} \geq \mathbf{Gx} + \mathbf{Hy}: && q_{ip} \leq M \sum_{n\in J} n K_{inp} \qquad \forall i \in I; p \in P && (48) \\
& && q_{ip} \geq \sum_{n\in J} d_i K_{inp} \qquad \forall i \in I^O; p \in P && (49) \\
&\textit{BCs:} && \text{BCs for the printing problem} && \\
&\mathbf{x} \in \mathbb{B}^{|\mathcal{A}|}; \mathbf{y} \in \mathbb{R}_+^{|\mathcal{B}|} \times \mathbb{Z}_+^{|\mathcal{C}|}; \mathbf{q} \in \mathbb{R}_+^{|\mathcal{D}|}: && (45); v_i \geq 0; (47) \qquad \forall i \in I; p \in P &&
\end{aligned}
$$

Constraints (48) and (49) provide bounds on the amount of each design produced by each plate. Constraints (48) ensure that $q_{ip} = 0$ if no design of type i is allocated to plate p. Otherwise, q_{ip} is limited by the number of slots for design i in plate p, i.e., $\sum_{n\in J} nK_{inp}$, multiplied by an upper bound on the number of rotations of any plate. Constraints (49) enforce the amount of each CSD produced by plate p to be greater than or equal to its demand, if the plate contains that specific CSD. This is valid because constraints (41) impose that the demand of CSDs must be fulfilled using a single plate. Note that constraints (49) are not imposed for SDs since their demand is assumed to be zero.

Given an MP-S solution at iteration h, $(\bar{W}_p^h, \bar{K}_{inp}^h, \bar{Z}_{cp}^h, \bar{v}_i^h, \bar{q}_{ip}^h)$, the SP determines the number of rotations of each used plate, denoted by $\bar{r}_p^h$, and checks if the production quantities are feasible with respect to the OP, i.e., if $\bar{q}_{ip}^h = \bar{r}_p^h \sum_{n \in J} n\bar{K}_{inp}^h$. As the demand for SDs is null and each CSD can be assigned to only one plate, the optimal number of rotations for each used plate $\bar{r}_p^h$ can be computed as the number of rotations required to fulfill the total demand of the CSDs in plate p. Let $\bar{P}^h = \{p : p \in P, \bar{W}_p^h = 1\}$ denote the set of used plates, $\bar{I}_{hp}^O = \{i : i \in I^O, \sum_{n \in J} n\bar{K}_{inp}^h \geq 1\}, \forall p \in \bar{P}^h$ denote the set of CSDs assigned to plate p, and $\bar{a}_{ip}^h = \sum_{n \in J} n\bar{K}_{inp}^h$ denote the number of slots in plate p allocated to design i in the current solution. The rotations for each used plate is computed as $\bar{r}_p^h = \max_{i \in \bar{I}_{hp}^O} \left\{ \frac{d_i}{\bar{a}_{ip}^h} \right\}, \forall p \in \bar{P}^h$.

Inequalities (50) are the BCs for the printing problem. The BCs are added for each violation detected in the SP, i.e., for each element in set $\mathcal{Q}_h = \{(i,p) : i \in I, p \in \bar{P}^h, \bar{q}_{ip}^h \neq \bar{r}_p^h \bar{a}_{ij}^h\}$, and $\forall h \in \mathcal{H}'$, where $\mathcal{H}'$ is the index set of the MP-S solutions considered so far. Let $\bar{I}_{hp}^{O*}$ be the set that contains the design in plate p that requires the largest number of rotations to fulfill its total demand, i.e., $\bar{I}_{hp}^{O*} = \left\{ i' : i' \in \arg\max_{i \in \bar{I}_{hp}^O} \left\{ \frac{d_i}{\bar{a}_{ip}^h} \right\} \right\}$. Note that $\bar{I}_{hp}^{O*} \subseteq \bar{I}_{hp}^O$ and we limit $|\bar{I}_{hp}^{O*}| = 1$. The logic of the BCs is such that, if the design $i' : i' \in \bar{I}_{hp}^{O*}$ occupies the same number of slots as in the current solution (i.e., $K_{i'np} = 1 : n = \bar{a}_{i'p}^h$), the total amount of each design i produced by this plate is equal to the number of rotations $\bar{r}_p^h$ multiplied by the number of slots for the corresponding design $\sum_{n \in J} nK_{inp}$. Parameter B_i^h is calculated as $B_i^h = \bar{r}_p^h |J|, \forall i \in I^O$ and $B_i^h = \bar{r}_p^h, \forall i \in I^S$.

$$\text{BCs (Printing): } q_{ip} \geq \bar{r}_p^h \sum_{n \in J} nK_{inp} - B_i^h \sum_{i' \in \bar{I}_{hp}^{O*}} \sum_{\substack{n \in J: \\ n = \bar{a}_{i'p}^h}} (1 - K_{i'np}) \quad \forall h \in \mathcal{H}'; (i,p) \in \mathcal{Q}_h \quad (50)$$

The BCs (50) are a strengthened version of the following BCs: $q_{ip} \geq \bar{r}_p^h \sum_{n \in J} nK_{inp} - B_i^h \sum_{i' \in \bar{I}_{hp}^O} \sum_{\substack{n \in J: \\ n = \bar{a}_{i'p}^h}} (1 - K_{i'np}) - B_i^h \sum_{i' \in I \setminus \bar{I}_{hp}^O} \sum_{n \in J} K_{i'np}, \forall h \in \mathcal{H}'; (i,p) \in \mathcal{Q}_h$, which are aligned with the general form (10). Note that the BCs (50) are derived in terms of the configuration variables for the design $i' : i' \in \bar{I}^{O*}_{hp}$ and independent of the variables for other designs in the same plate, i.e., any design $i : i \in \bar{I}^{O}_{hp} \setminus \bar{I}^{O*}_{hp}$. This is valid because, as long as design i' occupies the same number of slots in plate p as in the current solution, the number of rotations of such plate is always greater than or equal to $\bar{r}^{h}_{p}$, as the complete demand of design i' must be fulfilled using only this plate. We prove the validity of the BCs (50) in Section 2.2 of the online supplement.

4.2.2. Enhanced LBBD. We present the MP-E and the SIs for the printing problem. For this application, we classify the designs into a set of groups denoted by $\mathcal{G}$ (indexed by g) according to the demand parameters, such that designs in the same group have the same demand level. Let $d^{\mathcal{G}}_{g}$ denote the demand of any design in group g and $I_g = \{i : i \in I,\, d_i = d^{\mathcal{G}}_{g}\}$ denote the set of designs in group g. The MP-E is defined as presented below, by adding to the MP-S a new set of binary variables U_{gnp}, a set of constraints that link these new variables with the original configuration variables K_{inp}, and the SIs. Constraints (51) and (52) 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 computed as $B_g = \min\{|J|, |I_g|\}$. Constraints (53) are the SIs, which we discuss in detail next, and constraints (54) define the domain of the new variables.

$$
\begin{aligned}
\text{MP-E (Printing problem):} \quad & \min \sum_{p\in P} c^P W_p + \sum_{i\in I^O} c^O_i v_i + \sum_{i\in I^S} c^S_i v_i \\
\mathbf{Ax} \geq \mathbf{a}: \quad & (36) - (42) \\
\mathbf{Dy} + \mathbf{Eq} \geq \mathbf{e}: \quad & (44) \\
\mathbf{Fq} \geq \mathbf{Gx} + \mathbf{Hy}: \quad & (48) \text{ and } (49) \\
\textit{BCs:} \quad & \text{BCs for the printing problem (50)}
\end{aligned}
$$

$$\mathbf{x}\in\mathbb{B}^{|\mathcal{A}|};\ \mathbf{y}\in\mathbb{R}_+^{|\mathcal{B}|}\times\mathbb{Z}_+^{|\mathcal{C}|};\ \mathbf{q}\in\mathbb{R}_+^{|\mathcal{D}|}:\quad (45);\ v_i\geq 0;\ (47) \qquad \forall i\in I;\ p\in P$$

$$\mathbf{Ix}+\mathbf{Jz}\geq\mathbf{g}:\quad \sum_{i\in I_g} K_{inp}\leq B_g U_{gnp} \qquad \forall g\in\mathcal{G};n\in J;p\in P \tag{51}$$

$$\sum_{i\in I_g} K_{inp}\geq U_{gnp} \qquad \forall g\in\mathcal{G};n\in J;p\in P \tag{52}$$

$$\textit{Logic-based SIs:}\quad \text{SIs for the printing problem} \tag{53}$$

$$\mathbf{z}\in\mathbb{B}^{|\mathcal{G}|}:\quad U_{gnp}\in\{0,1\} \qquad \forall g\in\mathcal{G};n\in J;p\in P \tag{54}$$

Given an MP-E solution at iteration h, $(\bar{W}_p^h, \bar{K}_{inp}^h, \bar{Z}_{cp}^h, \bar{v}_i^h, \bar{q}_{ip}^h, \bar{U}_{gnp}^h)$, the SP and the BCs remain the same as in the standard LBBD, while the SIs are derived as follows. Let $\bar{\mathcal{G}}_p^h=\{g: g\in\mathcal{G}, \sum_{n\in J} n\bar{U}_{gnp}^h\geq 1\}$, $\forall p\in\bar{P}^h$ be the set of groups allocated to each used plate p, and $\bar{b}_{gp}^h=\min_{n\in J}\{n\bar{U}_{gnp}^h\}$, $\forall p\in\bar{P}^h, g\in\bar{\mathcal{G}}_p^h$ be the minimum number of slots allocated to each group g assigned to plate p in the current solution. From set $\bar{\mathcal{G}}_p^h$, we separate the group that requires the largest number of rotations to fulfill the total demand of the designs that belong to it, and define the subset $\bar{\mathcal{G}}_p^{h*}=\left\{g' : g'\in\arg\max_{g\in\bar{\mathcal{G}}_p^h}\left\{\frac{d_g^{\mathcal{G}}}{\bar{b}_{gp}^h}\right\}\right\}$, $\forall p\in\bar{P}^h$. Note that $\bar{\mathcal{G}}_p^{h*}\subseteq\bar{\mathcal{G}}_p^h$ and we limit $|\bar{\mathcal{G}}_p^{h*}|=1$. We use this information to derive a lower bound on the total amounts of CSDs and SDs produced by each plate, which are imposed through the SIs (55) and (56), respectively.

SIs (Printing problem):

$$\sum_{i\in I^O} q_{ip}\geq\bar{r}_p^h\left(|J|-\sum_{i\in I^S}\sum_{n\in J}K_{inp}\right)-\bar{r}_p^h|J|\sum_{g\in\bar{\mathcal{G}}_p^{h*}}\sum_{\substack{n\in J:\\ n=\bar{b}_{gp}^h}}(1-U_{gnp}) \qquad \forall h\in\mathcal{H}';p\in\bar{P}^h \tag{55}$$

$$\sum_{i\in I^S} q_{ip}\geq\bar{r}_p^h\sum_{i\in I^S}\sum_{n\in J}K_{inp}-\bar{r}_p^h\sum_{g\in\bar{\mathcal{G}}_p^{h*}}\sum_{\substack{n\in J:\\ n=\bar{b}_{gp}^h}}(1-U_{gnp}) \qquad \forall h\in\mathcal{H}';p\in\bar{P}^h \tag{56}$$

The SIs (55) ensure that, as long as at least one design in group $g\in\bar{\mathcal{G}}_p^{h*}$ is assigned to a number of slots equal to $\bar{b}_{gp}^h$, which implies $U_{gnp}=1: g\in\bar{\mathcal{G}}_p^{h*}, n=\bar{b}_{gp}^h$, the total amount of CSDs produced by plate p is always greater than or equal to $|J|\bar{r}_p^h$ if no SD is assigned to this plate, or greater than or equal to $(|J|-1)\bar{r}_p^h$ if an SD is assigned to plate p. The lower bounds on the total amounts of CSDs imposed by the SIs (55) are valid because: (i) $U_{gnp}=1: g\in\bar{\mathcal{G}}_p^{h*}, n=\bar{b}_{gp}^h$ ensures that the total number of rotations of plate p is at least $\bar{r}_p^h$; and (ii) constraints (36) and (37) ensure that the total number of slots occupied by CSDs in any plate equals either $|J|-1$ or $|J|$. The SIs (56) enforce the total amount of SDs produced by plate p to be greater than or equal to the number of rotations $\bar{r}_p^h$, if any SD is allocated to this plate and $U_{gnp}=1: g\in\bar{\mathcal{G}}_p^{h*}, n=\bar{b}_{gp}^h$. Finally, the SIs (55) and (56) are added to the MP-E only if they are violated in the current solution, i.e., if $\sum_{i\in I^O}\bar{q}_{ip}^h<\bar{r}_p^h\left(|J|-\sum_{i\in I^S}\sum_{n\in J}n\bar{K}_{inp}^h\right)$ and $\sum_{i\in I^S}\bar{q}_{ip}^h<\bar{r}_p^h\sum_{i\in I^S}\sum_{n\in J}n\bar{K}_{inp}^h$, respectively. We prove the validity of SIs (55)–(56) in Section 2.3 of the online supplement.

4.2.3. Computational Results. Table 6 presents the data sets for this problem, which consists of the 72 benchmark instances in Baumann et al. (2015). As for the first application, the problem data are available at `https://github.com/Karim-Perez/process-configuration-problems`. We classify these instances into four sets according to the number of designs as presented below. The computing time limit is 1800 seconds for all these experiments.

		Set A	Set B	Set C	Set D
Number of designs	$(\|I\|)$	$[6,20]$	$[23,29]$	$[32,39]$	$[58,117]$
Number of plates	$(\|P\|)$	$[5,15]$	$[20,25]$	$[25,30]$	$[50,90]$
Number of groups	$(\|\mathcal{G}\|)$	$[1,6]$	$[4,10]$	$[5,12]$	$[10,35]$
Number of instances		24	12	12	24

Table 6 Data sets for the printing problem

Tables 7 and 8 present the average results for the data sets. We reproduced the findings in Baumann and Trautmann (2014) (*B&T (2014)*), who solve a linearized version of the OP using CPLEX, on our workstation and present the same statistics as for the application in the steel industry. For Sets C and D, the average computing time and the number of optimal solution are not presented, as all the solution methods reached the time limit without finding an optimal solution for these instances.

		Set A						Set B					
Solution Method		nLB	Gap	Time	BCs	SIs	OS	nLB	Gap	Time	BCs	SIs	OS
B&T (2014)		1.000	0.0%	90.15	-	-	24/24	0.839	16.4%	1522.67	-	-	2/12
LBBD	Stand. (BCs)	1.000	0.1%	358.70	214.5	-	23/24	0.856	24.7%	1670.04	468.6	-	1/12
	Enh. (BCs+SIs)	1.000	0.0%	37.05	42.6	16.1	24/24	0.966	3.8%	1293.03	149.9	41.4	5/12
B&Ch	Stand. (BCs)	1.000	0.0%	19.60	182.5	-	24/24	0.879	12.6%	1651.23	713.1	-	1/12
	Enh. (BCs+SIs)	1.000	0.0%	**5.91**	111.7	30.8	24/24	**0.972**	**2.8%**	968.29	403.3	78.2	7/12

Table 7 **Average results for the printing problem (Set A and Set B)**

		Set C				Set D				
Solution Method		nLB	Gap	BCs	SIs	nLB	Gap	BCs	SIs	FS
B&T (2014)		0.617	39.8%	-	-	0.512	57.9%	-	-	7/24
LBBD	Stand. (BCs)	0.725	45.0%	408.7	-	**0.562**	54.1%	612.7	-	16/24
	Enh. (BCs+SIs)	**0.804**	26.8%	138.9	38.2	**0.562**	50.5%	172.7	40.2	19/24
B&Ch	Stand. (BCs)	0.681	35.4%	1467.7	-	0.543	47.1%	2408.8	-	23/24
	Enh. (BCs+SIs)	0.743	**26.7%**	724.6	162.2	0.559	**45.8%**	1344.0	302.7	21/24

Table 8 **Average results for the printing problem (Set C and Set D)**

Overall, the enhanced LBBD solves most instances with up to 29 designs (Sets A and B) and provides improved results for larger instances. The standard LBBD solved using the standard implementation (*LBBD Stand. (BCs)*) performs poorly and has no advantage over the approach in B&T (2014). This can be explained by the structure of the MP, which includes many technical constraints and where finding optimal solutions at each iteration consumes a significant amount of time.

The enhanced LBBD presents a superior performance for the instances in Sets A and B. In particular, the B&Ch Enh. (BCs+SIs) solves the instances in Set A 15.3 times faster than the OP in B&T (2014). Regarding Set B, the B&Ch Enh. (BCs+SIs) could solve 5 more instances, and improve the average gap and computing time by approximately 13.6% and 36.4%, respectively, in comparison with the approach B&T (2014).

Finding good quality solutions for large instances is challenging according to the results for Sets C and D in Table 8. None of these instances could be solved within 30 minutes by any of the tested methods. However, the B&Ch implementation, and particularly the enhanced reformulation, allows to reduce the average gap by 13.1% in Set C and to provide feasible solutions for 16 more instances in Set D, in comparison to the approach in B&T (2014). The detailed results for the tested problems and further remarks on the LBBD can be found in Section 3 of the online supplement.

5. Concluding Remarks

We developed a general LBBD to solve production planning problems with configuration decisions. We introduced a standard LBBD and an enhanced LBBD, where the latter one implements logic-based inequalities as a further SP relaxation added to the MP during the solution process. We applied the LBBD on three variants of cutting stock problems in the steel industry and an application in the printing industry from the literature, and assessed the performance of the standard LBBD and the B&Ch implementations.

We conclude that generating and adding the SIs dynamically to the MP is an effective device to improve the LBBD, with the potential to be successfully applied to other problems. Including the SIs seems to speed up the convergence of the LBBD, which significantly outperforms the approaches in the literature, regardless the implementation used (i.e., the standard LBBD and the B&Ch). The results also show that the performance differences between the standard LBBD and the B&Ch implementations may be related to the structure of the MP, which varies among the applications in the domain of the studied problem due to the tailored technical constraints. Based on experiments not reported in this paper, we observed that adding the SIs at the same time as the BCs in the LBBD method is more efficient than an alternative implementation where the SIs are added first and the BCs are added as last resource. With respect to the linearized version of the OP, we also observed that neither applying the classical Benders decomposition via the automatic CPLEX Benders nor adding the SIs as lazy constraints improve the computational results of this approach. Finally, we consider that the proposed LBBD can be potentially enhanced through the development of efficient heuristics that determine improved solutions for the OP from an MP solution at each iteration, and through the development of efficient BC and SI generation strategies.

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 reviewers for their valuable comments that significantly improved the quality of this paper, and the authors of the papers which introduce the applications tested here for making their data sets available.

References

Baumann P, Forrer S, Trautmann N (2015) Planning of a make-to-order production process in the printing industry. *Flexible Services and Manufacturing Journal* 27(4):534–560.

Baumann P, Trautmann N (2014) Efficient symmetry-breaking formulations for grouping customer orders in a printing shop. *2014 IEEE International Conference on Industrial Engineering and Engineering Management*, 506–510.

Beck JC (2010) Checking-up on branch-and-check. Cohen D, ed., *Principles and Practice of Constraint Programming – CP 2010*, 84–98 (Springer Berlin Heidelberg).

Benders JF (1962) Partitioning procedures for solving mixed variables programming problems. *Numersiche Mathematik* (4):238–252.

Chu Y, Xia Q (2004) Generating Benders cuts for a general glass of integer programming problems. Régin JC, Rueher M, eds., *Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems*, 127–141 (Springer Berlin Heidelberg).

Ciré AA, Coban E, Hooker JN (2016) Logic-based Benders decomposition for planning and scheduling: A computational analysis. *The Knowledge Engineering Review* 31(5):440–451.

Coban E, Hooker JN (2013) Single-facility scheduling by logic-based Benders decomposition. *Annals of Operations Research* 210(1):245–272.

CPLEX Optimization Studio II (2019) V12. 9: User's manual for CPLEX. *International Business Machines Corporation* URL `https://www.ibm.com/support/knowledgecenter/SSSA5P_12.9.0/ilog.odms.studio.help/Optimization_Studio/topics/COS_home.html`.

Degraeve Z, Gochet W, Jans R (2002) Alternative formulations for a layout problem in the fashion industry. *European Journal of Operational Research* 143(1):80–93.

Degraeve Z, Vandebroek M (1998) A mixed integer programming model for solving a layout problem in the fashion industry. *Management Science* 44(3):301–310.

Emde S, Polten L, Gendreau M (2020) Logic-based Benders decomposition for scheduling a batching machine. *Computers & Operations Research* 113:104777.

Fazel-Zarandi MM, Beck JC (2012) Using logic-based Benders decomposition to solve the capacity- and distance-constrained plant location problem. *INFORMS Journal on Computing* 24(3):387–398.

Fazel-Zarandi MM, Berman O, Beck JC (2013) Solving a stochastic facility location/fleet management problem with logic-based Benders' decomposition. *IIE Transactions* 45(8):896–911.

Hajizadeh I, Lee C (2007) Alternative configurations for cutting machines in a tube cutting mill. *European Journal of Operational Research* 183(3):1385 – 1396.

Harjunkoski I, Grossmann IE (2001) A decomposition approach for the scheduling of a steel plant production. *Computers & Chemical Engineering* 25(11):1647 – 1660.

Heching A, Hooker JN, Kimura R (2019) A logic-based Benders approach to home healthcare delivery. *Transportation Science* 53(2):510–522.

Hooker JN (2007) Planning and scheduling by logic-based Benders decomposition. *Operations Research* 55(3):588–602.

Hooker JN, Ottosson G (2003) Logic-based Benders decomposition. *Mathematical Programming* 96(1):33–60.

Martínez KP, Adulyasak Y, Jans R, Morabito R, Toso EAV (2019) An exact optimization approach for an integrated process configuration, lot-sizing, and scheduling problem. *Computers & Operations Research* 103:310–323.

Riedler M, Raidl G (2018) Solving a selective dial-a-ride problem with logic-based Benders decomposition. *Computers & Operations Research* 96:30–54.

Riise A, Mannino C, Lamorgese L (2016) Recursive logic-based Benders' decomposition for multi-mode outpatient scheduling. *European Journal of Operational Research* 255(3):719–728.

Roshanaei V, Booth KE, Aleman DM, Urbach DR, Beck JC (2020a) Branch-and-check methods for multi-level operating room planning and scheduling. *International Journal of Production Economics* 220:107433.

Roshanaei V, Luong C, Aleman DM, Urbach D (2017a) Propagating logic-based Benders' decomposition approaches for distributed operating room scheduling. *European Journal of Operational Research* 257(2):439–455.

Roshanaei V, Luong C, Aleman DM, Urbach DR (2017b) Collaborative operating room planning and scheduling. *INFORMS Journal on Computing* 29(3):558–580.

Roshanaei V, Luong C, Aleman DM, Urbach DR (2020b) Reformulation, linearization, and decomposition techniques for balanced distributed operating room scheduling. *Omega* 93:102043.

Sun D, Tang L, Baldacci R (2019) A Benders decomposition-based framework for solving quay crane scheduling problems. *European Journal of Operational Research* 273(2):504–515.

Thorsteinsson ES (2001) Branch-and-check: A hybrid framework integrating mixed integer programming and constraint logic programming. Walsh T, ed., *Principles and Practice of Constraint Programming — CP 2001*, 16–30 (Springer Berlin Heidelberg).

Tran TT, Araujo A, Beck JC (2016) Decomposition methods for the parallel machine scheduling problem with setups. *INFORMS Journal on Computing* 28(1):83–95.

Tuyttens D, Vandaele A (2014) Towards an efficient resolution of printing problems. *Discrete Optimization* 14:126–146.

Wheatley D, Gzara F, Jewkes E (2015) Logic-based Benders decomposition for an inventory-location problem with service constraints. *Omega* 55:10 – 23.