Formulations and Theoretical Analysis of the One-Dimensional Multi-Period Cutting Stock Problem with Setup Cost

Eduardo M. Silva[a], Gislaine M. Melega[b], Kerem Akartunalı[c], Silvio A. de Araujo[a]

[a] *Universidade Estadual Paulista "Júlio de Mesquita Filho" (UNESP), São José do Rio Preto - Brazil*
[b] *Universidade Federal de São Carlos, São Carlos - Brazil*
[c] *Dept. of Management Science, University of Strathclyde, Glasgow, UK*

Abstract

In this paper, we study the one-dimensional multi-period cutting stock problem with setup costs on cutting patterns. We present pattern-based and pseudo-polynomial formulations for the problem. Reformulations are also proposed to improve the lower bounds. We then present a thorough theoretical analysis to establish the strength of the various proposed formulations in comparison to each other. Finally, a computational analysis is conducted to complement the theoretical analysis and provide further insights with respect to the complexity and strength of the formulations.

Keywords: Combinatorial Optimization, Cutting, Cutting Pattern Setups, Strong Reformulations.

1. Introduction

The cutting stock problem (*CSP*) was one of the problems identified by Kantorovich in his 1939 paper entitled "Mathematical methods of organizing and planning production" (later published in Kantorovich (1960)). Among the very first techniques to emerge from operational research to be applied in practice, the *CSP* is concerned with determining the best way of cutting a set of objects into smaller items, often with a large potential of economic savings. The *CSP* is encountered in a wide variety of industrial applications in the steel, wood, glass and paper industries (Ben Amor and Valério de Carvalho, 2005).

The cutting process in the *CSP* may be affected by various factors, particularly by the number of times one has to switch between different cutting patterns, e.g., changing the positions of the cutting knives (Wuttke and Heese, 2018). Such adjustments often interrupt production and/or may impose a setup cost every time a different cutting pattern is used. Therefore, it is often desirable to have a cutting plan composed of fewer cutting patterns. The problem that focuses only on minimizing of the number of different cutting patterns

Email addresses: `eduardo.msilva094@gmail.com` (Eduardo M. Silva), `gislainemelega@gmail.com` (Gislaine M. Melega), `kerem.akartunali@strath.ac.uk` (Kerem Akartunalı), `silvio.araujo@unesp.br` (Silvio A. de Araujo)

while satisfying demands is known in the literature as the Pattern Minimization Problem (*PMP*) (Vanderbeck, 2000). In the remainder of the paper, the *CSPs* will be used specifically to denote the single period problem with setups costs on the cutting patterns. We note that even this single period version of the problem is known to be $\mathcal{NP}$-hard (McDiarmid, 1999).

Considering the *CSPs* with multiple time periods (Trkman and Gradisar, 2007; Tomat and Gradisar, 2017), there is a strong relevance to the lot-sizing problem, which has been an area of very active research over the last six decades (Brahimi et al., 2017), offering significant cost savings to the manufacturing sector by generating the least costly production plan over a planning horizon with multiple periods. The lot-sizing problem deals with key decisions such as when and how much to produce or stock, while respecting limitations such as satisfying demands on time. The body of research devoted to the topic (and solution methodologies therein) is extensive, ranging from polyhedral methods such as extended formulations and valid inequalities (Doostmohammadi and Akartunalı, 2018; Gruson et al., 2019; Zhao and Zhang, 2020) to decomposition and relaxations (Van Vyve et al., 2014; de Araujo et al., 2015; Akartunalı et al., 2016), and heuristics designed for real-world problems (Fiorotto et al., 2017; Wu et al., 2018; Absi and van den Heuvel, 2019), as well as stochastic and robust approaches to tackle uncertainty in a broad range of settings (Alem et al., 2020; Attila et al., 2021; Quezada et al., 2020).

Integrated lot-sizing and cutting stock problems were recently classified in the extensive review of Melega et al. (2018), where the authors essentially identify three levels of production, with the first level associated to the purchase/manufacture of object(s), the second level to the cutting process of objects into pieces, and third level to production of final products from pieces. In their classification scheme, the case that considers exclusively the second level, with multiple time periods in a planning horizon, the inventory of cut pieces providing the link between different periods and cutting of objects planned for each period is called the Multi-Period Cutting Stock Problem (*MPCSP*). It is also worth noticing that when more than one level is considered, with multiple time periods in a planning horizon, Melega et al. (2018) classify the problem as the Integrated Lot-sizing and Cutting Stock Problem.

In this paper, we consider the *MPCSP* with setup costs on cutting patterns and a one-dimensional cutting process, which will hereafter refer to it as the *MPCSPs*. Although integrated decision problems have gained more attention over the last decade, there are only a small number of papers dealing with setups on cutting patterns with multiple periods in the literature. Moreover, most of this literature simply uses the multi-period adaptation of the well-known *CSP* formulations of Gilmore and Gomory (1961) and Kantorovich (1960). To the best of our knowledge, only recently, Ma et al. (2019) presented, for the first time, a multi-period adaptation (including setups) of the arc-flow model of Alves and Valério de Carvalho (2008a).

Our paper provides important contributions in this domain. First, we present three formulations to provide a rather complete picture of alternative formulations for the *MPCSPs*. To the best of our knowledge, two of these *MPCSPs* formulations are proposed here for the first time in the literature : the ones inspired by the *CSP* models of Johnston and Sadinlija (2004) and Delorme and Iori (2019). Secondly, we consider strengthening the formulations by using extended reformulations. More specifically, we use the facility location reformulation of Krarup and Bilde (1977). Although this is an effectively used method in the lot-sizing domain, its application to the *MPCSPs* is not trivial due to cutting patterns. Thirdly, we present a thorough theoretical analysis investigating the strength of various formulations given in the paper, providing a comparative ranking with respect to lower bounds to be expected from the formulations. Finally, to complement our theoretical analysis with an understanding of performance in practice, a computational analysis is provided.

We remark that although the concepts of the cutting-stock and lot-sizing formulations are used, the formulations proposed are not direct adaptations from the literature of these individual problems. Considering the point of view of the *CSP*, including setups on the single period case is already an area of extensive research (see Subsection 2.2) and some of the formulations proposed in the paper are presented for the first time in the literature, even considering their simplified single period version. From the lot sizing problem point of view, since we are considering the production of cutting patterns (which contains a set of items), the classical production variables of the Uncapacitated Lot Sizing (*ULS*) problem had to be modified and the theoretical results for the *ULS* are no longer valid. As the adaptations are not direct, an interesting problem arises, which is different from the classical *ULS* and has not been fully explored in the literature. Additionally, it is important to highlight that, to the best of our knowledge, the facility location reformulation has never been applied to such a problem, and it presents high quality lower bounds.

It is worth remarking that in the *MPCSPs* addressed in this study, the setup cost only reflects the direct or indirect costs related to a setup, for example, when changeovers imply an unavoidable loss of material (Arbib and Marinelli, 2007), or when several workers are needed to perform the setup, which implies high labor cost (Kolen and Spieksma, 2000), or when a setup involves a costly craftwork (Bonnevay et al., 2016). The considered setup costs do not include penalty costs for lost production capacity, since this should be taken into account via the introduction of setup times, which may impact time-related parameters or indicators (such as due dates, throughput, production capacity). In general, when considering practical applications, production capacity and its consequent constraints must be considered when integrating cutting stock and lot sizing problems. Since the research developed in this paper does not consider production capacity, it is limited to some exceptional practical applications, and it is also relevant as a relaxation of several real problems, where production capacity is apparent.

The remainder of the paper is organized as follows: a literature review about models and methods related to our problem, as well as a discussion regarding the impact of setup costs, setup times and production capacity, is presented in Section 2. The three formulations and their descriptions are presented in Section 3. Strengthening the formulations using the facility location reformulation is discussed in Section 4. Our main theoretical results evaluating the strengths of different formulations are presented in Section 5, followed by computational results and discussion in Section 6. Finally, in Section 7, we make our concluding remarks and discuss some potential directions for future research.

2. Literature review

In this section, we present a literature review related to studies that address the cutting stock problem with setup on the cutting patterns. The problem of minimizing the number of different cutting patterns in the final solution of a *CSP* has been considered since the 1970s. Most of the papers dealing with setups on cutting patterns are an extension of the Single Stock Size Cutting Stock Problem, defined in the typology presented in Wäscher et al. (2007). In this literature review, we firstly present a discussion regarding setup costs, setup times and production capacity. Afterwards, we discuss papers that present studies for the single period cutting stock problem with setups on cutting patterns (*CSPs*), and finally, we discuss papers regarding studies for the multiple period cutting stock problem with setups on cutting patterns and, in this case, there are papers on the *MPCSPs* as well as papers on the integrated lot-sizing and cutting stock problem (more than one level according to Melega et al. (2018)) that also consider setups on cutting patterns.

2.1. Setup costs, setup times and production capacity

Firstly, it is important to understand the complex trade-offs present in the objective function. In the context of the classical ULS problem, a setup cost indicates the fixed cost borne to start production, and it has a clear trade-off with inventory costs, since the larger the amount produced in a period to fulfill future demand, the smaller the incidence of fixed costs and the larger the inventory costs. In the context of this paper, the setups refer to the action of positioning cutting knives. A clear trade-off exists between solutions that diversify patterns (more setups) to cut fewer objects, and solutions with fewer patterns (less setups) that perhaps spend more in terms of objects cut. Additionally, a clear trade-off exists between solutions that bring forward production (increasing inventory) to have better combinations of items in cutting patterns, which might decrease the total use of raw material by cutting fewer objects (Vanzela et al., 2017). However, the trade-off between setups and inventory costs is not as clear as is for the classical ULS problem. The number of knives setups (number of setups) may only loosely, or not at all, be related to production volumes (setup run duration) and hence to inventory levels. This point is discussed in the paper of Diegel et al. (2006), where the authors attempt to avoid the problem of short setup runs in the cutting plan (which maps to avoiding small values for production amounts of each cutting pattern in each period in our problem context.) According to the authors, on average, fewer number of setups mean longer setup runs, which increase the inventory levels. However, individual setup runs may not change uniformly. As the number setups decreases, some setup runs become longer, but others may remain as is, or even become shorter. It is also worth mentioning that we are aware of only one paper that considers a multi-objective approach for a setting with multiple periods and setups on cutting patterns (Oliveira et al., 2021), albeit without an elaboration on the points we have discussed.

Secondly, it is important to distinguish the impact of setup time from the impact of setup cost. Setups on cutting patterns have two types of major impact on production: one is the time a setup requires, which may impact on time-related parameters or indicators (due dates, throughput, production capacity, etc.); another is a direct or indirect cost derived from setup operations. Regarding production capacity, when considering single-period problems, demand is not time-indexed and hence, it is natural in this case to minimize the production time, that is, to maximize throughput. For this reason, the absence of production capacity constraints is common in the single-period case. When considering multiple cutting stock problems, production capacity and its consequent constraints must be considered to model most of the practical applications. The resource involved in the capacity constraint is time, including time spent on the number of setups and setups run duration. It is worth mentioning that the inclusion of production capacity constraints makes the problem much more difficult to solve and in general, heuristic procedures are employed.

2.2. Single period cutting stock problem with setups on cutting patterns

Regarding some important achievements from the literature, we highlight the heuristic approaches of Haessler (1988) for the one-dimensional trim-loss problem for the paper and film industries, which minimizes waste and setup costs of changing cutting patterns. The first branch-and-price-and-cut algorithm for the *PMP* is presented in Vanderbeck (2000) which solves a compact formulation for the *CSPs*. The formulation minimizes the number of different cutting patterns, that can be seen as an analogy to the setup on cutting patterns. Umetani et al. (2003) presented a mathematical model to the *CSPs* in which the deviation of the cut items from the demand is minimized, while the total number of different cutting patterns is considered as a constraint in the model and equal to a specific value. The proposed model is solved by an iterated local search algorithm with adaptive pattern generation, within some practical constraints on the generation of cutting patterns. Later, Yanasse and Limeira (2006) proposed a bi-objective approach for the *CSPs* and Golfeto et al. (2009) introduced an symbiotic genetic algorithm. Aloisio et al. (2011a) addressed the *PMP* for special instances, where no more than two items fit in an object in stock. They explored two formulations for the problem, and derived various results concerning the existence of specific solutions. Aloisio et al. (2011b) discusses two formulations to the *PMP*, which consists of the formulation proposed by Vanderbeck (2000) and the reformulation obtained by applying the Dantzig-Wolfe decomposition. The authors show that the linear relaxation of the reformulation is stronger and as practical as the original formulation in terms of computational effort.

A literature review of mathematical models and solution approaches to the *CSPs* is presented in Henn and Wäscher (2013). Since then, more papers have been published. Cui et al. (2013) studied the two-dimensional *CSPs* where the main objective is input minimization, and pattern minimization is an auxiliary objective. The authors proposed a sequential grouping heuristic, which essentially selects the items that can be used to generate the next cutting pattern. The work of Song and Bennell (2014) focused on irregular shapes in the two-dimensional problem and employed both column generation and sequential heuristics as solution methods to exploit the problem structure. The model minimizes the number of stock sheets used, while meeting the demand of items and within a constraint that limits the number of possible cutting pattern to a given value, as in (Umetani et al., 2003). In de Araujo et al. (2014) and Aliano Filho et al. (2018), the authors proposed a bi-objective approach for the one-dimensional *CSPs*, minimizing both the number of stock objects and the number of cutting patterns used. The work of Cui et al. (2015) used a two-phase approach to solve the one-dimensional *CSPs*, where the first phase essentially generates promising cutting patterns based on a sequential grouping process and the second phase solves a mixed integer programming model.

Some papers consider setup costs based on practical cases, where the quantification of the setups costs is precisely supported, i.e., setups are (or can be) priced with good precision. Arbib and Marinelli (2007) consider a real cutting process in a glass factory, where changeovers imply an unavoidable loss of material, which can be evaluated with great accuracy since cuts are done while the glass float moves forward. In Kolen and Spieksma (2000), a practical case that arises in abrasive paper production is considered, where every setup involves a time-consuming operation that can be associated with labor cost. Bonnevay et al. (2016) study a paper printing application, where setup involves a costly craftwork (print composition).

2.3. Multiple period cutting stock problem with setups on cutting patterns

Studies of the *MPCSPs* in the literature are scarce. The paper by Aktin and Özdemir (2009) with an application in medicine proposes an extension of the *CSP* model of Gilmore and Gomory (1961, 1963), and involves a generalized cost function with material, setup, labor and delay costs. The authors propose a two-stage solution method, with the first stage estimating the total number of cutting patterns, as well as their generation, while the second stage simply solves the model. The second study consists of an application in the furniture industry (Gramani and França, 2006). There is a time limit capacity for the use of the cutting machine, where the time spent to cut one plate depends on the cutting pattern used in this plate. The authors also use the classical *CSP* model of Gilmore and Gomory (1961, 1963) with an objective to minimize trim-loss, along with inventory and setup costs. The solution approach uses a shortest path reformulation of the problem. Nonås and Thorstenson (2000, 2008) studied the one-dimensional cutting stock problem in a Norwegian company that produces off-road trucks. The mathematical model consists of a non-linear formulation with a continuous time horizon and a concave objective function, which minimizes holding costs and setup costs associated with cutting patterns. As solution methods, two global search procedures and three local search procedures are presented.

To the best of our knowledge, Ma et al. (2018) is the only paper that considers setup costs, albeit without capacity constraints, in multi-period settings. The authors propose a mathematical model based on Gilmore and Gomory (1961, 1963), and a dynamic programming-based heuristic. The approaches were applied to real data of a problem that arises in a company that produces extra-high-voltage and high-voltage switch equipment. According to the authors, the computational results have significance for the company because it enables them to reduce total cost and enhance competitive advantages. The authors highlight that the studied problem has applications in a range of industries. In a more recent paper, Ma et al. (2019) studied the *MPCSPs* with setup costs and production capacities incorporating setup duration. Two mathematical models, based on the classical *CSP* model of Gilmore and Gomory (1961, 1963) and the arc-flow formulation of Alves and Valério de Carvalho (2008a), are presented and two heuristics, based on column generation and dynamic programming are proposed. Their computational experiments highlight the efficiency of the formulation based on Gilmore and Gomory (1961, 1963) due to the fast increasing size of the arc-flow model. The authors also present a statistical analysis regarding the changes in the setup cost and the production capacity. Ma et al. (2021) extend the previous work of Ma et al. (2019) by considering the production replanning problem according to demand realization which might be different from the predicted demand.

As pointed out by Melega et al. (2018), in some industrial applications, delivering the orders on time can be far more important than reducing the resulting waste and the cost of cut objects. Models that consider due dates in the formulation better describe the need of the industry in such a case (Reinertsen and Vossen, 2010; Arbib and Marinelli, 2014; Braga et al., 2015). These papers consider production capacity limits and combine the standard objective of minimizing the number of rolls used with a scheduling term penalizing the tardiness of the cutting operations. This problem is referred to in the literature as the combined Cutting Stock and Scheduling Problem. The multi-period setting also appears in these papers, but a shorter planning horizon is considered. In general, they assume that it takes exactly one unit of time to cut a stock roll.

As mentioned earlier, the multi-period cutting stock problem studied in this paper addresses multiple periods in a finite planning horizon, the inventory of items which comprises the link between periods, and the cutting process of objects in each period. In this way, the decision-making of such processes might occur at different levels of the supply chain. For instance, the planning managers are usually responsible for the production planning of the items in order to meet the demand, whereas the machine manufacturers perform the optimization of the cuts in the cutting process. However, the literature have pointed out computational results demonstrating the benefits of an integrated approach instead of taking decisions separately (Hendry et al., 1996; Arbib and Marinelli, 2005; Gramani and França, 2006; Gramani et al., 2009; de Araujo et al., 2014; Vanzela et al., 2017). As follows, we next review those studies in which setup on cutting patterns appears in the integrated lot-sizing and cutting stock problem.

Ghidini et al. (2007) present a mathematical model for an application in the furniture industry that addresses setup cost on the cutting patterns and production capacities integrating setup duration, which is solved by a column generation procedure. Santos et al. (2011) approach a multi-period cutting stock problem with setup cost and time, in addition to the capacity of the cutting process, to model a real-world problem from the furniture industry. The model is solved by an optimization package considering real data from the factory and different sets of cutting patterns. In an application in the aluminum industry, Suliman (2012) presents a non-linear mathematical model, which considers setup costs for cutting patterns, as well as a capacity constraint in terms of the number of total cuts. There are also constraints to manage the inventory availability of objects with costs related to their inventories. To solve the problem, the authors present an algorithm based on a lot-sizing approach, where for each period, starting with the last one, the products and the quantities to be produced are established, as well as the cutting patterns to be used. Oliveira et al. (2021) analysed a multi-objective integrated lot-sizing and cutting stock problem and proposed a goal programming model that takes into account six different goals representing the interests of different stakeholders in the manufacturing process. The model is solved by a column generation based heuristic. Christofoletti et al. (2021) propose an integrated lot-sizing and three-dimensional cutting stock problem from the mattress industry. It considers setup cost and time, as well as limited production capacity. A mathematical model of mixed integer programming was proposed and solved with an optimisation package. Computational results based on real data are presented.

Two other papers also deal with setups related to the cutting patterns. However, they use setup variables to estimate the setup time in the capacity constraints (Alem and Morabito, 2013) and minimum object length to use a cutting pattern (Silva et al., 2015). Therefore, these studies do not address setup cost. Alem and Morabito (2013) propose a two-stage stochastic mixed optimization model to represent the production planning of a small-scale furniture industry. The authors consider stochastic demand and setup times and use robust optimization tools to solve the integrated problem. Computational tests were performed using real and simulated data. Silva et al. (2015) deal with the problem observed in a textile factory, in which the mathematical model comprises the *CSPs*, considering upper and lower bounds on demand for each piece, with a setup constraint for each cutting pattern to guarantee a minimum length to use that cutting pattern. The model is solved by a column generation procedure.

3. Problem definition and mathematical formulations

We make the following assumptions in defining and formulating to the *MPCSPs*. There is a finite planning horizon represented by a set T of periods, and a set P of item types, for which external demand is known. We consider the one-dimensional case, i.e., only one dimension is taken into account in the cutting process, and assume there is a single object type of length L, which is available in stock in an unlimited quantity. In each period $t \in T$, an item type $i \in P$ of given length l_i has to be cut from objects to meet demand d_{it}. W.l.o.g., we assume that $L \geq l_i$, L and l_i are all integral, and $l_i > 1$ holds for all i. A cutting pattern is defined as a way a stock object is cut to produce the demanded items. Every time a different cutting pattern is cut, a setup cost is incurred. We do not consider setup time and capacity constraints.

As the demand is known for all periods, the produced items can be brought forward to a certain period t at the cost of storing them. Backorder is not allowed. W.l.o.g., we also assume zero inventory for all items at the beginning of the horizon, otherwise, for each item with a positive initial inventory, this quantity can be removed from its respective demand and the value of the initial inventory can be set to zero. Therefore, the *MPCSPs* consists of determining the cutting patterns and the number of times each cutting pattern will be cut (i.e., the cutting pattern frequency) in each period of the planning horizon to satisfy customer demands, while minimizing the costs associated with cutting pattern setup, inventory of items, and objects consumed.

We also define the following notation before presenting the formulations.

Sets

T	set of periods (index t);
P	set of item types (index i);
J	set of feasible cutting patterns indices;
H_i	set of integer frequencies for item type i in object of length L.
M_t	set of cutting pattern frequencies needed to satisfy demands from period t to $\|T\|$.

Input Parameters

L	length of the objects;
l_i	length of item type i;
d_{it}	demand of item type i in period t;
c_t	setup cost of cutting patterns in period t;
h_{it}	unit holding cost of item type i at the end of period t;
c	material cost per object;
$\overline{a}_{ijt}$	number of items i obtained from cutting pattern j in period t.

For any period t, note that each cutting pattern j can be described by a vector $A_j = (\overline{a}_{1jt}, \overline{a}_{2jt}, ..., \overline{a}_{|P|jt})^T$. Obviously, a cutting pattern j is valid if $\overline{a}_{1jt}l_1 + \overline{a}_{2jt}l_2 + ... + \overline{a}_{|P|jt}l_{|P|} \leq L$ holds.

In this paper, we consider $|M_t| = \max_{i,j,\ \overline{a}_{ijt}>0} \left\{ \left\lceil \frac{\sum_{\tau=t}^{|T|} d_{i\tau}}{\overline{a}_{ijt}} \right\rceil \right\}$, representing an upper bound for the frequency of the cutting patterns in period t.

General Decision Variables

x_{jt}	number of objects cut according to cutting pattern j in period t;
y_{jt}	binary variable, 1 if cutting pattern j is used in period t, 0 otherwise;
s_{it}	amount of inventory of item type i at the end of period t.

The formulations proposed in this paper are presented in the following order: Section 3.1 shows a column generation-based formulation that consists of a multi-period adaptation of the classical *CSP* model of Gilmore and Gomory (1961) (denoted by *AGG*); Section 3.2 displays a knapsack-based formulation inspired by the *CSP* model of Johnston and Sadinlija (2004) (denoted by *AJS*); Section 3.3 presents an arc-flow-based formulation motivated by the reflect formulation of Delorme and Iori (2019) (denoted by *ARE*). To the best of our knowledge, two of these formulations (*AJS* and *ARE*) are proposed here for the first time in the literature. In Section 4, we will also propose facility location reformulations of these three formulations, which, to the best of our knowledge, has not been considered before for this problem setting. It is worth remarking that we have also implemented other formulations, such as a knapsack-based formulation motivated by Vanderbeck (2000) and an arc-flow-based formulation inspired by Alves and Valério de Carvalho (2008a). However, for the sake of keeping the exposition in this paper concise and focused, we did not include these formulations in this paper, due to their not promising computational results.

3.1. Adapted Gilmore and Gomory formulation (AGG)

This is a multi-period extension of the *CSP* formulation of Gilmore and Gomory (1961) with setups on cutting patterns. This formulation is that most often used in the literature for multiple periods extensions of the *CSP* and can be adapted to our problem as follows:

***AGG* model**

$$\textbf{Minimize} \sum_{t\in T}\sum_{i\in P} h_{it}s_{it} + \sum_{t\in T}\sum_{j\in J}(c_t y_{jt} + c x_{jt}) \tag{3.1}$$

subject to:

$$x_{jt} \leq |M_t| y_{jt} \qquad \forall j,\ \forall t \tag{3.2}$$

$$s_{i,t-1} + \sum_{j\in J} \overline{a}_{ijt} x_{jt} = d_{it} + s_{it} \qquad \forall i,\ \forall t \tag{3.3}$$

$$x_{jt} \in \mathbb{Z}^+,\ y_{jt} \in \{0,1\} \qquad \forall j,\ \forall t \tag{3.4}$$

$$s_{it} \geq 0 \qquad \forall i,\ \forall t \tag{3.5}$$

The objective function (3.1) is the minimization of the holding costs of the items, the setup costs for the cutting patterns and the material costs of objects. Constraints (3.2) force y_{jt} to be 1 if objects are cut according to the cutting pattern j in period t, i.e., $x_{jt} > 0$. Constraints (3.3) represent the inventory balance constraints of items. Constraints (3.4) and (3.5) indicate the domains of the decision variables.

3.2. Adapted Johnston and Sadinlija formulation (AJS)

Johnston and Sadinlija (2004) proposed a new compact formulation for the *CSP*. Extending their ideas to the *MPCSPs* and applying it to a formulation inspired by Vanderbeck (2000), we first define binary variables r_{ijht}, which are 1 when item type i appears in pattern j in period t with multiplicity h, and 0 otherwise, and then a new integer variable p_{ijht} that is equal to x_{jt} if $r_{ijht} = 1$ and 0 otherwise. We note that h is bounded by $|H_i|$. In other words, for $h \notin H_i$, where $H_i = \{1, 2, ..., |H_i|\}$, no feasible cutting patterns are generated, thus, the corresponding binary variable can be excluded from the model for all cutting patterns j. Additionally, the size of the set J needs to be defined in advance and in this paper, it will be defined according to the solution of the AGG formulation. The formulation is then as follows:

***AJS* model**

$$\textbf{Minimize} \sum_{t\in T}\sum_{i\in P} h_{it}s_{it} + \sum_{t\in T}\sum_{j\in J}(c_t y_{jt} + c x_{jt}) \tag{3.6}$$

subject to:

$$\sum_{i \in P} \sum_{h \in H_i} h r_{ijht} l_i \leq L y_{jt} \qquad \forall j,\ \forall t \qquad (3.7)$$

$$x_{jt} \leq |M_t| y_{jt} \qquad \forall j,\ \forall t \qquad (3.8)$$

$$s_{i,t-1} + \sum_{j \in J} \sum_{h \in H_i} h p_{ijht} = d_{it} + s_{it} \qquad \forall i,\ \forall t \qquad (3.9)$$

$$p_{ijht} - |M_t| r_{ijht} \leq 0 \qquad \forall i,\ \forall j,\ \forall h,\ \forall t \qquad (3.10)$$

$$\sum_{h \in H_i} r_{ijht} \leq y_{jt} \qquad \forall i,\ \forall j,\ \forall t \qquad (3.11)$$

$$\sum_{h \in H_i} p_{ijht} \leq x_{jt} \qquad \forall i,\ \forall j,\ \forall t \qquad (3.12)$$

$$\sum_{h \in H_i} p_{ijht} \geq x_{jt} - |M_t| \left(1 - \sum_{h \in H_i} r_{ijht} \right) \qquad \forall i,\ \forall j,\ \forall t \qquad (3.13)$$

$$p_{ijht} \in \mathbb{Z}^+, \quad r_{ijht} \in \{0, 1\} \qquad \forall i,\ \forall j,\ \forall h\ \forall t \qquad (3.14)$$

$$x_{jt} \in \mathbb{Z}^+, \quad y_{jt} \in \{0, 1\} \qquad \forall j,\ \forall t \qquad (3.15)$$

$$s_{it} \geq 0 \qquad \forall i,\ \forall t \qquad (3.16)$$

The objective function (3.6) and setup constraints (3.8) are identical to previous formulation. Constraints (3.7) ensure that the items cut in an object do not exceed its length, considering the multiplicity of each item in the cutting pattern. Constraints (3.9) represent the inventory balance, where the amount produced is represented by the sum of the multiplicity of each item type i over all cutting patterns j. Constraints (3.10) guarantee that only one r_{ijht} and its corresponding p_{ijht} can be selected for each (i, j, t) tuple because, at most, one r_{ijht} can be chosen due to constraints (3.11). Constraints (3.12) and (3.13) guarantee that if p_{ijht} is strictly positive, then it must be equal to x_{jt}. Finally, constraints (3.14), (3.15) and (3.16) represent the variable domains.

3.3. Adapted Reflect formulation (ARE)

In Delorme and Iori (2019) a computational enhancement of the *CSP* arc-flow model, first proposed by Wolsey (1977) and later explored by Valério de Carvalho (1999, 2002), called Reflect arc-flow formulation, is proposed. In our study, the Reflect formulation is extended to the multi-period problem and a cutting pattern setup environment, based on the *PMP* arc-flow formulation of Alves and Valério de Carvalho (2008b), is introduced.

As described in Ma et al. (2019), the multi-period *PMP* arc-flow problem is formulated over acyclic digraphs $G^{nt} = (V, A^{nt})$, $\forall n \in M_t$ and $\forall t \in T$, where $M_t = [1, 2, ..., |M_t|]$. A vertex in V corresponds to an integer position within the object, with vertex 0 representing the leftmost border of the object and L the rightmost (hence, $|V| = L + 1$). An arc (d, e) in A^{nt} represents an item of width $e - d$ placed at a position d of the leftmost border of the object. The unused portion of the objects are represented by arcs (d, e) with $e - d = 1$. Any path with tail in 0 and head in L created by arcs in set A^{nt} indicates that an object is cut n times in period t according to this path (cutting pattern).

In the Reflect formulation, only half of the object capacity is considered, i.e., just $L/2$ vertices. Two main features of the Reflect model are:

- it considers only vertices corresponding to normal cutting patterns with size smaller than $L/2$ (including 0), plus an additional node, called R, corresponding to size $L/2$;
- it considers the same arcs of the arc-flow formulation, but: (i) it "reflects" each arc (d, e) with $d < L/2$ and $e > L/2$ into an arc $(d, L - e)$; (ii) it removes all arcs (d, e), including loss arcs, having $d \geq L/2$; and (iii) it creates a last loss arc by connecting the rightmost node before R with R.

Intuitively, a path in the Valério de Carvalho (1999, 2002) formulation can be seen as a pair of colliding paths in the Reflect formulation, i.e., two paths both starting at node 0 and ending at the same node, but with only one of the two paths passing through R (the one that contains the reflected arc). Delorme and Iori (2019) proved that any feasible cutting pattern for the *CSP* can be represented by a pair of colliding paths, whose reflected arc (d, e) has $d \leq e$. Thus, reflected arcs (d, e) with $d > e$ only generate symmetric solutions and can be excluded from the model.

Figure 1 presents a multigraph and a reflect multigraph, respectively, for an example with object length 11 ($L = 11$) and four items of lengths $l_1 = 7, l_2 = 4, l_3 = 3$ and $l_4 = 2$. These arcs are responsible for at least two paths. In the reflect multigraph, all the arcs before the reflected node (R) are added to the reflect multigraph, whereas the arcs that go through the reflected node are reflected until the corresponding node. In the case of the arc with length 7, it is reflected to the node 4. Finally, those arcs present after the reflected node are removed from the reflect multigraph.

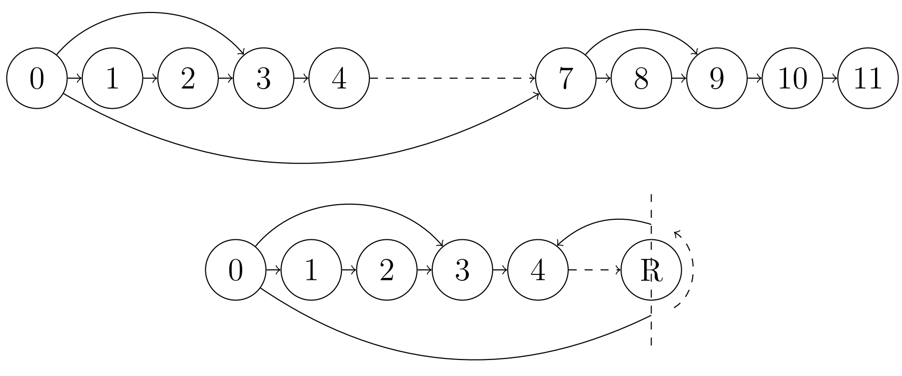


Figure 1: Multigraphs required for the standard arc-flow and the reflect flow, respectively.

We now extend this idea of the reflect multigraph to the multi-period and cutting pattern setup settings. Maintaining the general idea of the arc-flow problem, we define multigraphs $G^{nt} = (V, A^{nt})$, $\forall n \in M_t$ and $\forall t \in T$, where the set of vertices is $V = \{0\} \cup \{r \in \mathbb{N} : 0 < r < L/2\} \cup \{L/2\}$. The set of arcs A^{nt} is then partitioned into subsets A_s^{nt} and A_r^{nt}, where A_s^{nt}

represents the set of "standard" arcs and A_r^{nt} represents the set of reflected arcs, i.e., the arcs (d,e) from the arc-flow formulation that have been reflected to arc $(d, L-e)$. Each arc is defined by the quintuple (n,t,d,e,k), where $k=s$ indicates that it is a standard arc (i.e., in A_s^{nt}), $k=r$ indicates a reflected arc (i.e., in A_r^{nt}), and the remaining four indices are as defined previously. We also define A_i^{nt} as the subset of arcs associated with items i in period t, i.e., $A_i^{nt} = \{(d, d+l_i, s) \in A_s^{nt},\ \forall n,\ \forall t\} \cup \{(d, L-d-l_i, r) \in A_r^{nt},\ \forall n,\ \forall t\}$. Associating an integer decision variable f_{dek}^{nt} to each arc $(d,e,k) \in A^{nt}$, $\forall n \in M_t$ and $\forall t \in T$, we can model the problem as follows:

***ARE* model**

$$\textbf{Minimize} \sum_{t\in T}\sum_{i\in P} h_{it}s_{it} + \sum_{t\in T}\sum_{n\in M_t}\sum_{(d,e,r)\in A_r^{nt}} (c_t + cn) f_{der}^{nt} \tag{3.17}$$

subject to:

$$\sum_{(d,e,s)\in A_s^{nt}} f_{des}^{nt} = \sum_{(e,u,k)\in A^{nt}} f_{euk}^{nt} + \sum_{(d,e,r)\in A_r^{nt}} f_{der}^{nt} \qquad \forall e \in V\backslash\{0\},\ \forall t,\ \forall n \tag{3.18}$$

$$\sum_{(0,e,k)\in A^{nt}} f_{0ek}^{nt} = 2 \sum_{(d,e,r)\in A_r^{nt}} f_{der}^{nt} \qquad \forall t,\ \forall n \tag{3.19}$$

$$s_{it-1} + \sum_{n\in M_t}\sum_{(d,e,k)\in A_i^{nt}} n f_{dek}^{nt} = d_{it} + s_{it} \qquad \forall i,\ \forall t \tag{3.20}$$

$$f_{dek}^{nt} \in \mathbb{Z}^+ \qquad \forall n,\ \forall t,\ \forall (d,e,k) \in A^{nt} \tag{3.21}$$

$$s_{it} \geq 0 \qquad \forall i,\ \forall t \tag{3.22}$$

The objective function (3.17) is the minimization of the inventory holding cost of the items, the number of reflected arcs (which is equivalent to the number of different cutting patterns), and the number of multiplicities n of each reflected arc (which is equivalent to the material cost of objects). Constraints (3.18) ensure that the amount of flow from standard arcs entering a node e is equal to the amount of flow (for both standard and reflected arcs) emanating from e plus the amount of flow from reflected arcs entering e in all periods and multiplicities. Constraints (3.19) impose boundary conditions by enforcing the amount of flow emanating from node 0 to be twice the number of different cutting patterns used. Constraints (3.20) represent the inventory balance, and constraints (3.21) and (3.22) represent the variable domains.

Due to the reflection properties of the multigraph, the arc reduction proposed by Valério de Carvalho (1999), which was later extended to the multi-period case by Poldi and de Araujo (2016), can not be applied to the reflect arc-flow, except that arcs with length w_i are not placed before w_k if $w_i > w_k$, $i, k \in P$. Another criterion used was the one proposed by Côté and Iori (2018), which modifies the multigraph by removing the unit-width loss arcs $(d, d+1)$ and creating longer loss arcs that connect each vertex in V to its consecutive vertex in V (as shown in Figure 1).

4. Facility Location Reformulation

In the lot-sizing literature, extended reformulations are often employed to overcome the poor quality of the lower bounds obtained with the linear relaxation of the original formulations. The most classical reformulations consist of redefining the variables of the original problem according to facility location and shortest path problems (see Krarup and Bilde (1977); Eppen and Martin (1987), respectively). Such variable redefinition has been shown to describe the convex hull of the *ULS* (see Barany et al. (1984)). However, as mentioned before, since in the *MPCSPs* we are not considering individual items but cutting patterns which contain a set of items, some theoretical results proposed for the *ULS* are no longer valid, and a different and challenging problem arises that has not been fully explored in the literature. In this section, we intend to help fill this gap by presenting reformulations for the *MPCSPs*. For this, the three formulations presented in Section 3 are reformulated as a facility location problem which, though not describing the convex hull of the *MPCSPs*, obtains considerably improved lower bounds as shown by our theoretical and computational results. It is worth mentioning that we have not found any paper in the literature that employs the facility location reformulation for the *MPCSPs*.

An additional parameter is used in order to simplify the models based on the facility location reformulation, as follows:

$\hat{h}_{it\tau}$: holding cost of item type i from period t to period τ, i.e., $\hat{h}_{it\tau} = \sum_{q=t}^{\tau-1} h_{iq}$;

4.1. Adapted Gilmore and Gomory Facility Location reformulation (AGGFL)

This is a reformulation of the AGG model as a facility location problem. To present this new formulation, let us observe that a cutting pattern activated in period t contributes to fulfill the demand of item type i in different periods $\tau \geq t$. Let then:

- $\varphi_{ijt\tau}$: portion of the frequency of cutting pattern j cut in period t to satisfy part of the demand of item type i in period τ.

Observe that when reformulating the *MPCSPs* as a facility location problem, the relationship between the *MPCSPs* and facility location decision variables consists of considering a facility as a period-indexed cutting pattern that must be chosen in order to perform the cutting process, and satisfy the demand of period-indexed clients . In this way, the demand (cutting) of an item type i in a future time period τ can be spread over different cutting patterns j in previous time periods, from t to τ, in order to be met. Mathematically, the constraints connecting the production of the cutting patterns to the production of the facilities are given by:

$$x_{jt} = \sum_{\tau=t}^{|T|+1} \varphi_{ijt\tau} \qquad \bar{a}_{ijt} > 0, \ \forall i, \ \forall j, \ \forall t. \tag{4.1}$$

Note that defining the φ variables up to period $\tau = |T| + 1$ (rather than $\tau = |T|$) allows any excess production that would end up as inventory to be at the end of the horizon. In addition, the characterization of the cutting patterns ($\overline{a}_{ijt}$) in these constraints guarantee that the facility location decision variables ($\varphi_{ijt\tau}$) will have positive values for any item, only if this item belongs to the corresponding cutting pattern, i.e., $\overline{a}_{ijt} > 0$.

In order to demonstrate the relationship in constraints (4.1), we use the example presented in Figure 1, i.e., an object with length 11 and four items types of lengths $l_1 = 7, l_2 = 4, l_3 = 3$ and $l_4 = 2$. If the cutting pattern $A_{j_1} = (1, 0, 1, 0)^T$ represents a facility for any given $t \in T$, the production of facility indexed with $j_{1,t}$ is represented as a bipartite graph, where the upper node represents the period indexed cutting pattern and the first $4|T|$ lower nodes the period indexed clients, while the remaining ones represent the production excess over the end of the horizon. Figure 2 illustrates such scheme, where each arc represents the amount of item types cut from cutting pattern j_1 in period t to satisfy part of the demands for further periods, in other words, the arcs represent the product, $\overline{a}_{ij_1t}\varphi_{ij_1t\tau}$, for $i \in \{1, 2\}$ and $\tau \geq t$.

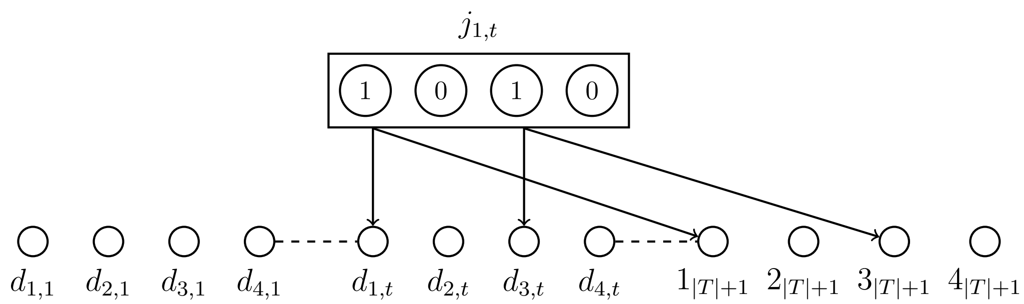


Figure 2: Cutting pattern production as facility production.

Therefore, the Facility Location reformulation for the *MPCSPs* considering the *AGG* formulation is given as follows:

AGGFL **model**

$$\textbf{Minimize} \sum_{t \in T} \sum_{i \in P} \sum_{j \in J} \sum_{\tau = t}^{|T|+1} \hat{h}_{it\tau} \overline{a}_{ijt} \varphi_{ijt\tau} + \sum_{t \in T} \sum_{j \in J} (c_t y_{jt} + c x_{jt}) \tag{4.2}$$

subject to:

(4.1)

$$\varphi_{ijt\tau} \leq d_{i\tau} y_{jt} \qquad \forall i, \ \forall j, \ \forall t, \ \tau \geq t \tag{4.3}$$

$$\sum_{j \in J} \sum_{\tau = 1}^{t} \overline{a}_{ijt} \varphi_{ij\tau t} = d_{it} \qquad \forall i, \ \forall t \tag{4.4}$$

$$x_{jt} \in \mathbb{Z}^+ \ y_{jt} \in \{0, 1\} \qquad \forall j, \ \forall t \tag{4.5}$$

$$\varphi_{ijt\tau} \in \mathbb{R}^+ \qquad \forall i, \ \forall j, \ \forall t, \ \forall \tau \geq t \tag{4.6}$$

The objective function (4.2) is the standard cost function. The linking constraints (4.1) are added to the reformulation in order to calculate the material costs of objects in the objective function. Constraints (4.3) link the facility location and setup variables ensuring that a setup is accounted each time a new cutting pattern is used. Constraints (4.4) guarantee the demand satisfaction and (4.5)-(4.6) define variable domains.

4.2. Adapted Johnston and Sadinlija Facility Location reformulation (AJSFL)

The *AJS* reformulation can be obtained in a similar fashion to the *AGGFL* described in Section 4.1. We introduce a set of φ variables with extra indexes h to obtain the link with the *AJS* production variables, i.e.:

$$p_{ijht} = \sum_{\tau=t}^{|T|+1} \varphi_{ijht\tau} \qquad \forall i,\ \forall j,\ \forall h,\ \forall t \tag{4.7}$$

The reformulation is then given by:

***AJSFL* model**

$$\textbf{Minimize} \sum_{t\in T}\sum_{i\in P}\sum_{j\in J}\sum_{\tau=t}^{|T|+1}\sum_{h\in H_i} \hat{h}_{it\tau} h \varphi_{ijht\tau} + \sum_{t\in T}\sum_{j\in J}(c_t y_{jt} + c x_{jt}) \tag{4.8}$$

subject to:

(4.7)

$$\sum_{i\in P}\sum_{h\in H_i} h r_{ijht} l_i \le L y_{jt} \qquad \forall j,\ \forall t \tag{4.9}$$

$$x_{jt} \le |M_t| y_{jt} \qquad \forall j,\ \forall t \tag{4.10}$$

$$\varphi_{ijht\tau} \le d_{i\tau} r_{ijht} \qquad \forall i,\ \forall j,\ \forall h,\ \forall t, \forall \tau \ge t \tag{4.11}$$

$$\sum_{j\in J}\sum_{\tau=1}^{t}\sum_{h\in H_i} h \varphi_{ijht\tau} = d_{it} \qquad \forall i,\ \forall t \tag{4.12}$$

$$\sum_{h\in H_i} r_{ijht} \le y_{jt} \qquad \forall i,\ \forall j,\ \forall t \tag{4.13}$$

$$\sum_{h\in H_i} p_{ijht} \le x_{jt} \qquad \forall i,\ \forall j,\ \forall t \tag{4.14}$$

$$\sum_{h\in H_i} p_{ijht} \ge x_{jt} - |M_t| \left(1 - \sum_{h\in H_i} r_{ijht}\right) \qquad \forall i,\ \forall j,\ \forall t \tag{4.15}$$

$$p_{ijht} \in \mathbb{Z}^+, \quad r_{ijht} \in \{0,1\} \qquad \forall i,\ \forall j,\ \forall h\ \forall t \tag{4.16}$$

$$\varphi_{ijht\tau} \in \mathbb{R}^+ \qquad \forall i,\ \forall j,\ \forall h,\ \forall t, \forall \tau \ge t \tag{4.17}$$

$$x_{jt} \in \mathbb{Z}^+, \quad y_{jt} \in \{0,1\} \qquad \forall j,\ \forall t \tag{4.18}$$

The objective function (4.8) is the standard cost function. Constraints (4.7) are responsible for linking the decision variables and are added to the reformulation in order to calculate the material costs of objects in the objective function. Constraints (4.9), (4.10), (4.13), (4.14) and (4.15) are the same as in the *AJS* model. Constraints (4.11) link the facility location and setup decision variables ensuring that a setup is accounted each time a new cutting pattern is used. Constraints (4.12) guarantee the demand satisfaction. Finally, constraints (4.16) - (4.18) define the variable domains.

4.3. Adapted Reflect Facility Location reformulation (AREFL)

Even though the cutting pattern setup is not explicitly expressed in the *ARE* model, it is still possible to obtain an extended reformulation from it. Consider the link between item type i production along period t for cutting patterns with frequency $n \in M_t$, given as follows:

$$\sum_{t=\tau}^{|T|+1} \overline{\varphi}_{int\tau} = \sum_{(d,e,k)\in A_i^{nt}} n f_{dek}^{nt} \qquad \forall i,\ \forall n,\ \forall t \tag{4.19}$$

where the $\overline{\varphi}$ variables represent the portion of production of item type i considering all cutting patterns $j \in J$ which were cut n times in period t to satisfy item type i demand of period τ.

The facility location reformulation of the *ARE* model is then given as follows:

***AREFL* model**

$$\textbf{Minimize} \sum_{t\in T}\sum_{i\in P}\sum_{n\in M_t}\sum_{\tau=t}^{|T|+1} \hat{h}_{it\tau}\overline{\varphi}_{int\tau} + \sum_{t\in T}\sum_{n\in M_t}\sum_{(d,e,r)\in A_r^{nt}} (c_t + cn) f_{der}^{nt} \tag{4.20}$$

subject to:

(4.19)

$$\sum_{(d,e,s)\in A_s^{nt}} f_{des}^{nt} = \sum_{(e,u,k)\in A^{nt}} f_{euk}^{nt} + \sum_{(d,e,r)\in A_r^{nt}} f_{der}^{nt} \qquad \forall e \in V\backslash\{0\},\ \forall t,\ \forall n \tag{4.21}$$

$$\sum_{(0,e,k)\in A^{nt}} f_{0ek}^{nt} = 2 \sum_{(d,e,r)\in A_r^{nt}} f_{der}^{nt} \qquad \forall t,\ \forall n \tag{4.22}$$

$$\overline{\varphi}_{int\tau} \le d_{i\tau} \sum_{(d,e,k)\in A_i^{nt}} f_{dek}^{nt} \qquad \forall i,\ \forall n,\ \forall t,\ \tau \ge t \tag{4.23}$$

$$\sum_{\tau=1}^{t}\sum_{n\in M_t} \overline{\varphi}_{in\tau t} = d_{it} \qquad \forall i,\ \forall t \tag{4.24}$$

$$f_{dek}^{nt} \in \mathbb{Z}^+ \qquad \forall n,\ \forall t,\ \forall (d,e,k) \in A^{nt} \tag{4.25}$$

$$\overline{\varphi}_{int\tau} \in \mathbb{R}^+ \qquad \forall i,\ \forall n,\ \forall t,\ \forall \tau \ge t \tag{4.26}$$

The objective function (4.20) is the standard cost function. The linking constraints (4.19) are added to the reformulation in order to calculate the material costs of objects in the objective function. Constraints (4.21) and (4.22) represent the reflect flow balance of the multigraphs. Constraints (4.23) link the facility location and setup variables ensuring that a setup is accounted each time a new cutting pattern is used for each n. Constraints (4.24) guarantee the demand satisfaction, and (4.25)-(4.26) define variable domains.

4.3.1. An alternative AREFL reformulation

The facility location variables of model (4.20)-(4.26) are $O(|T|^2|P|\sum_{n\in M_t} n)$. In this section, we present an alternative *AREFL* reformulation with facility location variable $O(|T|^2|P|)$, which obtains a weaker lower bound, but can be solved more efficiently. Consider the φ variables as the portion of the total production of item type i with respect period t, i.e.:

$$\varphi_{it\tau} = \sum_{n\in M_t} \overline{\varphi}_{int\tau} \qquad \forall i,\ \forall t,\ \forall \tau \geq t \tag{4.27}$$

This approach mimics the variables of the facility location reformulation of the uncapacited lot-sizing problem, since the φ indexes only relate items and periods. The link constraints (4.19) are replaced by:

$$\sum_{t=\tau}^{|T|+1} \varphi_{it\tau} = \sum_{n\in M_t} \sum_{(d,e,k)\in A_i^t} n f_{dek}^{nt} \qquad \forall i,\ \forall t \tag{4.28}$$

and the setup constraints (4.23) are then replaced by:

$$\varphi_{it\tau} \leq \sum_{n\in M_t} \sum_{(d,e,k)\in A_i^{nt}} f_{dek}^{nt} \qquad \forall i,\ \forall t \tag{4.29}$$

In the remainder of the paper, we will refer to this alternative formulation as *AREFL** model, in order to distinguish it from *AREFL*. We note that *AREFL** will be used in the computational analysis in Section 6 due to its computational efficacy, and although it presents a weaker bound than *AREFL*, the difference is rather minimal.

Table 1 summarizes some characteristics of the *MPCSPs* formulations and reformulations discussed in this paper. We mention the presence of knapsack and setup constraints and the number of binary, integer and continuous variables in each model.

5. Theoretical strengths of formulations

In this section, we intend to explore the relationship between the lower bounds of the formulations presented so far and, hence, theoretically establish the relative strengths of

Table 1: Formulations and Reformulations review.

Models	Knapsack constraints	Setup constraints	Number of Variables		
			Binary	Integer	Continuous
AGG	no	yes	$O(\|J\|\|T\|)$	$O(\|J\|\|T\|)$	$O(\|J\|\|T\|)$
AJS	yes	yes	$O(\|J\|\|T\|\|P\|\sum_i \|H_i\|)$	$O(\|J\|\|T\|\|P\|\sum_i \|H_i\|)$	$O(\|T\|\|P\|)$
ARE	no	no	—	$O(\|J\|\|T\|\|P\|\sum_t \|M_t\|)$	—
$AGGFL$	no	yes	$O(\|J\|\|T\|)$	$O(\|J\|\|T\|)$	$O(\|J\|\|T\|^2\|P\|)$
$AJSFL$	yes	yes	$O(\|J\|\|T\|\|P\|\sum_i \|H_i\|)$	$O(\|J\|\|T\|\|P\|\sum_i \|H_i\|)$	$O(\|J\|\|T\|^2\|P\|\sum_i \|H_i\|)$
$AREFL$*	no	no	—	$O(\|J\|\|T\|\|P\|\sum_t \|M_t\|)$	$O(\|T\|^2\|P\|)$

different formulations. Regarding the single period *CSP*, a few theoretical results can be found in the literature. Vanderbeck (2000) proved that the Dantzig-Wolfe decomposition of his formulation leads to a formulation at least as strong as the lower bound provided by the well-known Gilmore and Gomory (1961) formulation. In Valério de Carvalho (2002), the equivalence between the arc-flow formulation and the Gilmore and Gomory (1961) formulation is shown. Though we have not used it in this paper, it is also worth commenting that there are other single period *CSP* formulations in the literature, such as the One Cut and the Kantorovich-based formulations (Valério de Carvalho, 2002; Delorme and Iori, 2019).

We use the following notation in the remainder of the paper. $\mathbf{Z}_{LP}$ denotes the lower bound obtained from the linear problem (LP) relaxation of a formulation, e.g., $\mathbf{Z}_{LP}(AGG)$ denotes the solution obtained by the AGG formulation without the integrality constraints. Likewise, X_{LP} denotes the feasible region of the LP relaxation of a formulation, e.g., $X_{LP}(AGG)$ denotes the feasible region of the AGG formulation without integrality constraints.

For the sake of clarity, we provide the following definition (from Wolsey (1998)) which will be used throughout the Section.

Definition 5.1. *Given a polyhedron $Q \subset (\mathbb{R}^n \times \mathbb{R}^p)$, the projection of Q onto the subspace $\mathbf{R}^n$, denoted $proj_{(x)}(Q)$, is defined as:*

$$proj_{(x)}(Q) = \{x \in \mathbb{R}^n | (x, y) \in Q \text{ for some } y \in \mathbb{R}^p\}. \tag{5.1}$$

Proposition 5.1. *$\mathbf{Z}_{LP}(AGG) \geq \mathbf{Z}_{LP}(AJS)$, i.e., the lower bound obtained by the AGG formulation is at least as strong as the one obtained by the AJS formulation.*

The *AJS* model is a multi-period linearized version of the *PMP* model of Vanderbeck (2000). As proven in this paper, its Dantzig-Wolfe decomposition when letting the knapsack constraints define the subproblem naturally leads to the *CSP* formulation of Gilmore and Gomory (1961, 1963).

Proposition 5.2. *The ARE formulation models the MPCSPs.*

This directly follows from the fact that the reflect arc-flow formulation models the *CSP*, as proved by Delorme and Iori (2019). The authors also conclude the lower bound obtained by the reflect arc-flow, when demands are not considered as a limitation in the number of arcs, is the same as the one from the arc-flow formulation of Valério de Carvalho (1999).

Therefore, it is possible to conclude that the *ARE* formulation and the *PMP* arc-flow formulation of Ma et al. (2019) provide the same lower bound. We next discuss a decomposition technique of the *ARE* formulation.

5.1. Dantzig-Wolfe decomposition for the arc-flow MPCSPs formulation

In a similar fashion to Valério de Carvalho (2002), we can obtain an equivalent pattern-based formulation of the ARE model, applying the Dantzig-Wolfe decomposition to its linear relaxation, keeping the inventory balance constraints in the master problem and letting the flow conservation constraints define the subproblem.

The set of flow conservation constraints along with the non-negativity constraints and without the integrality requirements, define a polyhedral cone with equality constraints for each multiplicity $n \in M_t$ and period t, i.e, $P_{nt} = \{x \geq 0 : Ax = 0\}$, where A is the incidence matrix and x is a vector containing the arc-flow variables from the associated graph. From Minkowski's theorem, any point x of a non-empty polyhedron P can be expressed as a convex combination of the extreme points of P plus a non-negative linear combination of the extreme rays.

For each multiplicity n and period t, P_{nt} has just one extreme point, the origin, which corresponds to the solution with null flow. The circulation flows along each colliding path can not be expressed as a non-negative linear combination of other circulation flows, and are, therefore, the extreme rays of P_{nt}. The extreme flows are not bounded and each set of colliding path on the planning horizon will correspond to an extreme ray. Also, the master problem will not have a convexity constraint since its only extreme point is the null solution.

For the sake of simplicity, we consider the following constraints along with the flow balance constraints (3.18) and (3.19) to generate the subproblem:

$$y_{nt} = \sum_{(d,e,r) \in A_r^{nt}} f_{der}^{nt} \qquad \forall t,\ \forall n \tag{5.2}$$

where $y_{nt} \in \mathbb{Z}^+$.

The subproblem will only generate extreme rays to the master problem. Let Γ be the set of feasible colliding path indices. Note that for any period or multiplicity, the set Γ will be the same. Let μ_{nt}^j be the variables of the master problem, which take the value 1 if the corresponding cutting pattern j is included in period t with frequency n and 0 otherwise. Note $y_{nt} = \sum_{j \in \Gamma} \mu_{nt}^j$. For a given n, a column in the master problem can be defined by $(\underset{\sim}{a}_{nt}^j)$, where $\underset{\sim}{a}_{nt}^j = (a_{1nt}^j, a_{2nt}^j, ..., a_{|P|nt}^j)$ is the vector that defines the number of items in each period and cutting pattern multiplicity. The coefficients of these columns, a_{ijt}^n, are expressed in terms of the decision variables of the subproblem, f_{dek}^{jnt}, which correspond to the arcs (d, e) that take the value 1 when the arc is included in the path j with frequency n and 0 otherwise. The relation is explicitly given by:

$$a_{int}^j = \sum_{(d,d+l_i) \in A_i^{nt}} f_{d(d+l_i)k}^{jnt} \qquad \forall i,\ \forall j,\ \forall n,\ \forall t \tag{5.3}$$

Note we can simply set $a_{int}^j = \overline{a}_{ijt}$ for all i, j, t and n. The first result obtained from this decomposition extend a well known results from the CSP literature regarding the arc-flow formulation and the formulation of Gilmore and Gomory (1961, 1963).

Proposition 5.3. *$\boldsymbol{Z}_{LP}(AGG) = \boldsymbol{Z}_{LP}(ARE)$, i.e., the lower bound obtained by the AGG and ARE formulations are equal.*

Proof. The model (3.1)-(3.5) can be derived by taking the following variable substitution into the master problem:

$$x_{jt} = \sum_{n \in M_t} n\mu^j_{nt} \qquad \forall j,\ \forall t \tag{5.4}$$

$$y_{jt} = \sum_{n \in M_t} \mu^j_{nt} \qquad \forall j,\ \forall t \tag{5.5}$$

□

The next result correlates the facility location reformulations of the *ARE* and *AGG* formulations.

Theorem 5.4. *$\boldsymbol{Z}_{LP}(AGGFL) = \boldsymbol{Z}_{LP}(AREFL)$, i.e., the lower bound obtained by the AGGFL and AREFL formulations are equal.*

Proof. Using the Dantzing-Wolfe decomposition of the *AREFL* as above, we show $proj_{(A,x,y,s)} X_{LP}(M-AREFL) \subseteq X_{LP}(AGGFL)$ and $proj_{(A,x,y,s)} X_{LP}(AGGFL) \subseteq X_{LP}(M-AREFL)$, where *M-AREFL* is the master problem of the *AREFL* decomposition model.

⇒ Given a $X_{LP}(M - AREFL)$ point, we define φ variables according to each cutting pattern $j \in \Gamma$ as follows:

$$\sum_{\tau=t}^{|T|+1} \varphi^j_{int\tau} = n\overline{a}_{ijt}\mu^j_{nt} \qquad \forall i,\ \forall j,\ \forall t,\ \forall n \tag{5.6}$$

$$\varphi^j_{int\tau} \leq d_{i\tau}\overline{a}_{ijt}\mu^j_{nt} \qquad \forall i,\ \forall j,\ \forall t,\ \forall \tau \geq t,\ \forall n \tag{5.7}$$

Therefore, we can set $\varphi_{int\tau} = \sum_{j\in\Gamma} \varphi^j_{int\tau}$ without affecting the feasibility of constraints (4.19) and (4.23).

Now, using Equations (5.4) and (5.5), we take the *AGGFL* variables such that $\overline{a}_{ijt}\varphi_{ijt\tau} = \sum_{n\in M_t} \varphi^j_{int\tau}$, $\forall i,\ \forall j,\ \forall t,\ \forall \tau \geq t$. Hence, the feasibility constraints (4.1) and (4.3) are satisfied.

⇐ A similar argument is applied to obtain the other inclusion. For a given point in $X_{LP}(AGGFL)$, we use Equations (5.4) and (5.5) to define φ variables according to each $n \in M_t$ as follows:

$$\sum_{\tau=t}^{|T|+1} \varphi^n_{ijt\tau} = n\overline{a}_{ijt}\mu^j_{nt} \qquad \forall i,\ \forall j,\ \forall t,\ \forall n \tag{5.8}$$

$$\varphi^n_{ijt\tau} \leq d_{i\tau}\overline{a}_{ijt}\mu^j_{nt} \qquad \forall i,\ \forall j,\ \forall t,\ \forall \tau \geq t,\ \forall n \tag{5.9}$$

Now, we take $\varphi_{int\tau} = \sum_{j\in\Gamma} \varphi^n_{ijt\tau}$ to conclude the proof. □

Theorem 5.5. $\boldsymbol{Z}_{LP}(AJSFL) \geq \boldsymbol{Z}_{LP}(AGG)$*, i.e., the lower bound obtained by the AJSFL, i.e., the facility location reformulation of the AJS model, provides a lower bound at least as strong as the lower bound of the AGG model.*

Proof. It can be shown that $proj_{(A,x,y,s)}(X_{LP}(AJSFL)) \subset X_{LP}(AGG)$. For instance, choosing the AGG cutting plan containing only homogeneous cutting patterns, i.e., for each t we took $|P|$ cutting patterns such that $\overline{a}_{ijt} = 1$ if $j = i$ and 0 otherwise. Then, the extreme point $s_{it} = 0$ $x_{it} = d_{it}/\overline{a}_{iit}$, $y_{it} = d_{it}/|M_t|$ $\forall i, \forall t$ does not lie in $proj_{(A,x,y,s)}(X_{LP}(AJSFL))$.

Indeed, by noting the $\overline{a}$ variables of the AGG model are linked with the AJS model variables by:

$$\overline{a}_{ijt} = \sum_{h \in H_i} hr_{ijht} \qquad \forall i,\ \forall j,\ \forall t$$

the constraints (4.7), (4.14) and (4.11) implies that:

$$\varphi_{ijht\tau} \leq d_{i\tau} \underbrace{\frac{d_{it}}{|M_t|}}_{<1} < d_{i\tau} \qquad \forall i,\ \tau \in [t, |T|)$$

Therefore, a contradiction. □

Proposition 5.6. *The lower bound obtained by any of the extended formulations presented in this paper is at least as strong as the lower bound obtained by its respective formulation.*

It is obvious that the facility location reformulation naturally tightens the convex hull of the underlying lot-sizing formulation and therefore the result is trivial.

Theorem 5.7. $\boldsymbol{Z}_{LP}(AREFL) = \boldsymbol{Z}_{LP}(AGGFL) \geq \boldsymbol{Z}_{LP}(AJSFL) \geq \boldsymbol{Z}_{LP}(ARE) = \boldsymbol{Z}_{LP}(AGG) \geq \boldsymbol{Z}_{LP}(AJS)$.

This main result follows from all the previous results presented.

6. Computational study

The main objective of this section is to complement the theoretical results presented earlier by providing computational insights to address questions, such as "how difficult is a formulation to solve?" and "how much better is a formulation?" in a practical context with a range of classes of test problems. The formulations were tested with randomly generated instances in a PC with Intel Core i7 3.2 GHz CPU processor and 16 GB of RAM. The mathematical formulations were implemented in Visual Studio 2017 using the C callable library of IBM ILOG Cplex 12.9.0. In order to evaluate the impact of the proposed formulations and facility location reformulations of the *MPCSPs* we analysed two sets of instances, as presented in the next two subsections. In the first set, some easy instances are considered by varying item and period number, object size and inventory cost, while in the second set, more challenging instances are considered by varying item and period number and item length.

In Section 6.1, the pattern-based and the pseudo-polynomial formulations are exactly solved by a commercial optimization package with a *MIP* tolerance of 0.0001% and a time limit of 1800 seconds. In order to obtain an exact solution to this set of easy instances, the pattern-based formulations (*AGG* and *AGGFL*) consider all the possible columns, i.e., all possible cutting patterns are generated a priori and considered when solving the linear relaxation and the mixed-integer cutting stock problem. We also note that, due to its computational efficacy, *AREFL** is used in computational testing (rather than *AREFL*). In Section 6.2, difficult instances are considered, and the pattern based formulations *AGG* and *AGGFL* are solved by a column generation-based heuristic procedure called, respectively, *AGG-H* and *AGGFL-H*. The heuristics are quite straightforward. They consist in first solving the relaxed master problem by column generation, then considering the variables limited to those associated with the generated columns, the integer problem is solved. We next provide some details regarding the column generation procedure.

The restricted master problem starts with the columns related to the maximal homogeneous cutting patterns, i.e., columns of type: $(0, ..., a_{ii}, ..., 0)$, where $a_{ii} = |H_i|$, $\forall i$. The same columns are inserted for each period, $t \in T$. Then, during the pricing procedure, a knapsack problem is solved to find new cutting patterns for the restricted master problem. In the multi-period scenario, a pricing problem is solved for every period $t \in T$ and columns are included in the restricted master problem for period t until no more attractive columns are generated. The master problem, as well as the pricing subproblems are solved by the commercial optimization package.

We evaluate and compare the performance of the proposed models with respect to different factors, such as *LP* relaxation solution, Z_{LP}, time to solve the *LP* relaxation, T_{LP} (in seconds), mixed-integer problem (*MIP*) solution, Z_{IP}, time to solve the *MIP*, T_{IP} (in seconds), absolute *MIP* gap, GAP_A (in percentage), and relative *MIP* gap, GAP_R (in percentage). The absolute and relative gaps are calculated according to equations (6.1) and (6.2), respectively, where Z_{BestLB} is the best lower bound achieved when solving the integer problem. We consider the value GAP_R as the one provided by the optimization package, after solving the integer problem.

$$GAP_A = \frac{100(Z_{IP} - Z_{LP})}{Z_{IP}} \tag{6.1}$$

$$GAP_R = \frac{100(Z_{IP} - Z_{BestLB})}{Z_{IP}} \tag{6.2}$$

6.1. Computational results for the exact solution approaches

The first data set used to generate instances for the multi-period cutting stock problem with setups is based on data from the literature, more specifically, Poldi and de Araujo (2016) (number of periods, items and inventory cost), Cui et al. (2015) (setup cost and object length) and the CUTGEN1 generator proposed by Gau and Wäscher (1995) (length and demand of items), and with specifics as follows:

- number of periods: $|T| = \{3, 6\}$;
- number of items: $|P| = \{10, 20\}$;
- object length: $L = \begin{cases} 500, & \text{Small;} \\ 1000, & \text{Medium;} \\ 1500, & \text{High.} \end{cases}$
- setup costs: $c_t = 0.1L$;
- item length: $l_i \in [0.375L, 0.625L]$;
- inventory costs: $h_{it} = \begin{cases} 0.01l_i, & \text{Low;} \\ 0.05l_i, & \text{High.} \end{cases}$
- item demand: Generated by CUTGEN1[1]. with $d = 10$.

Using a full factorial design, the data set consists of 2 x 2 x 3 x 2= 24 problem classes. For each class, 20 instances are generated, resulting in a total of 480 problems. It is worth highlighting that these instances are easy to solve: requirements consist of less than 20 items of length greater than $L/3$, which implies less than 2 items per object. So basically, we do not need more than 400 positions in the object. Additionally, considering object lengths 500, 1000 or 1500 does not change the maximum number of possible subset sums.

Table 2 presents the average number of variables for classes of instances with medium object length (160 instances). The calculation is obtained as shown in Table 1 for each class and then the average value is taken. The *AGG* model presents the smaller number of variables for these classes, followed by its reformulation, *AGGFL*, with a 133% overall increment when compared to *AGG*. When the number of items is increased from 10 to 20 the *AGG* formulation presents an increase of 300%, followed by 636% of increase for the *ARE* model and 669% for the *AJS* model. We highlight that the *AREFL** formulation presents the lowest overall variable increment (only 0.4%) when compared to its original formulation *ARE*.

Table 3 presents the main results of this section by displaying, for each class and mathematical model, the average values for the *LP* relaxation and computational time to solve the *LP*. For all classes, the facility location reformulations present considerable improvements for the lower bound, which can reach up to 339% (Class 24 with the *AJSFL*). The *AJSFL* presents the highest improvements of the lower bound when compared to its original formulation (*AJS*), reaching on average 235.78% increase of the lower bound, followed by the *AGGFL* with 189% average improvement. Containing the largest problems in this

[1]CUTGEN1 is an instance generator for the one-dimensional cutting stock problem (Gau and Wäscher (1995))

Table 2: Average number of variables for classes of instances with medium object length.

Classes	Instances average size					
$(\|T\|/\|P\|)$	AGG	AJS	ARE	$AGGFL$	$AFJSFL$	$AREFL$*
Class (3/10)	900	11520	4986	1500	24930	5076
Class (3/20)	3600	88200	36246	6000	353800	36426
Class (6/10)	1800	41700	34206	4800	103800	34626
Class (6/20)	7200	322720	255324	19200	1320640	256044
Mean	**3375**	**115922**	**82690**	**7875**	**450792**	**83043**

data set, Classes 19 to 24 demonstrate the best lower bounds obtained by the reformulations when compared with their respective original formulations. It is worth to mention that it is known in the literature that AGG and ARE formulations provide the same lower bound quality, and we have shown in this study (Theorem 5.4) that the same behavior holds for the facility location reformulations $AGGFL$ and $AREFL$. However, since we are using the alternative $AREFL$* described in Section 4.3.1, the lower bound values are not equal (though the difference is minimal). For all classes, AJS consistently obtained the worst lower bounds, whereas $AGGFL$ provided the best values. The computational time required to solve the *LP* relaxation is relatively small (not more than 0.75 seconds on average), and the original formulations are naturally faster to solve than their respective facility location reformulations.

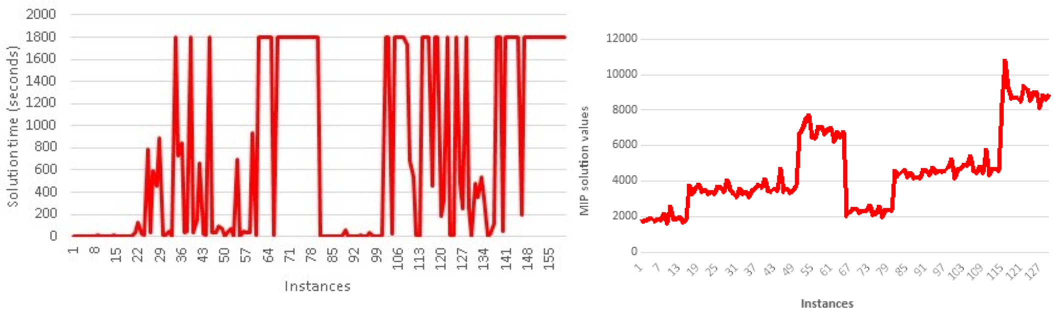
Table 4 displays, for each mathematical model and class, the objective function value found for the *MIP* and the time spend to solve the *MIP*. Note that AJS and $AJSFL$ are not able to solve some instances in Classes 19, 20, 21, 23 and 24, as remarked in the table, therefore, we are not able to compare these two formulations in these classes. The results show that, for the first set of classes (Class 1 to 6), AGG, ARE, $AGGFL$, and $AREFL$* present similar objective function values. However, as the problem size increases due to the number of period and items (Classes 7 to 24), the reformulations, on average, are able to find the best solutions in 13 classes, whereas the formulations found only in 5 classes. This fact reinforces the motivation for using reformulations when solving the problem. The $AREFL$* reformulation presents the highest improvements when compared to its original formulation ARE. The AJS presents the worse values to the problem, followed by the $AJSFL$. There is no clear behaviour of the formulations regarding the computational time, only the natural fact that as the problem enlarges, more computational time is spent to solve the mixed-integer problem. For most of the classes, the reformulations are faster than their respective original formulations. The smallest average computational time is presented by $AGGFL$ (932.94 seconds), followed by AGG (981.94 seconds) and $AREFL$* (1004.79 seconds).

In Table 5, we present the average values for the absolute and relative gaps obtained by each formulation in each class. As noted earlier, AJS and $AJSFL$ could not solve some instances in Classes 19, 20, 21, 23 and 24, and hence, we are not able to compare these two formulations in these classes. The absolute gap obtained by the reformulations are considerably smaller when compared to the original formulations, with $AGGFL$ achieving

Table 3: Evaluation of the lower bounds obtained by the proposed mathematical models

Classes	*LP* relaxation solution (Z_{LP})						Time for the *LP* relaxation (seconds) (T_{LP})					
($\|T\|/\|P\|/L/h_{it}$)	*AGG*	*AJS*	*ARE*	*AGGFL*	*AJSFL*	*AREFL**	*AGG*	*AJS*	*ARE*	*AGGFL*	*AJSFL*	*AREFL**
Class 1 (3/10/S/L)	690.18	497.48	690.18	1334.83	1080.84	1312.72	0.002	0.008	0.02	0.061	0.022	0.025
Class 2 (3/10/S/H)	691.49	497.47	691.49	1757.47	1340.96	1649.5	0.002	0.01	0.01	0.04	0.08	0.025
Class 3 (3/10/M/L)	825.22	551.14	825.22	1756.82	1325.57	1707.56	0.003	0.007	0.015	0.04	0.08	0.026
Class 4 (3/10/M/H)	825.22	551.14	825.22	2082.26	1502.31	1960.94	0.002	0.018	0.01	0.02	0.1	0.2
Class 5 (3/10/H/L)	661.61	474.23	661.61	1619.65	1268.46	1568.38	0.002	0.012	0.02	0.036	0.063	0.03
Class 6 (3/10/H/H)	661.61	474.22	661.61	1849.15	1408.52	1739.51	0.002	0.02	0.02	0.036	0.08	0.04
Mean (3/10)	**725.89**	**507.61**	**725.89**	**1733.36**	**1321.11**	**1656.44**	**0.002**	**0.0125**	**0.016**	**0.039**	**0.07**	**0.058**
Class 7 (3/20/S/L)	1177.77	862.74	1177.77	2485.17	2089.64	2456.82	0.008	0.053	0.037	0.45	0.17	0.08
Class 8 (3/20/S/H)	1179.16	862.76	1179.16	3264.98	2632.75	3081.01	0.005	0.08	0.03	0.26	0.32	0.067
Class 9 (3/20/M/L)	1386.66	923.69	1386.66	3333.74	2596.16	3227.72	0.07	0.075	0.036	0.173	0.256	0.07
Class 10 (3/20/M/H)	1386.66	923.70	1386.66	3982.45	2976.33	3726.69	0.003	0.09	0.03	0.15	0.41	0.06
Class 11 (3/20/H/L)	1241.33	895.12	1241.33	3152.64	2550.42	3003.4	0.009	0.074	0.05	0.3	0.263	0.9
Class 12 (3/20/H/H)	1241.33	895.12	1241.33	3558.46	2812.13	3277.7	0.003	0.1	0.05	0.022	0.4	0.1
Mean (3/20)	**1268.82**	**893.85**	**1268.82**	**3296.24**	**2609.57**	**3128.89**	**0.016**	**0.079**	**0.039**	**0.22**	**0.3**	**0.24**
Class 13 (6/10/S/L)	1140.65	780.11	1140.65	2590.17	2096.32	2535.93	0.003	0.048	0.05	0.18	0.21	0.143
Class 14 (6/10/S/H)	1144.34	780.1	1144.34	3556.75	2705.33	3332.15	0.005	0.11	0.04	0.09	0.43	0.09
Class 15 (6/10/M/L)	1263.59	810.28	1263.59	3411.78	2601.15	3291.75	0.04	0.06	0.057	0.085	0.27	0.114
Class 16 (6/10/M/H)	1263.59	810.28	1263.59	4215.42	3038.16	3897.88	0.003	0.1	0.05	0.07	0.55	0.12
Class 17 (6/10/H/L)	1084.15	748	1084.15	3183.22	2526.84	3053.8	0.004	0.111	0.08	0.13	0.335	0.12
Class 18 (6/10/H/H)	1084.15	748	1084.15	3706.36	2819.8	3426.62	0.004	0.14	0.07	0.09	0.59	0.13
Mean (6/10)	**1163.42**	**779.46**	**1163.42**	**3443.95**	**2631.27**	**3256.36**	**0.009**	**0.095**	**0.056**	**0.1**	**0.4**	**0.12**
Class 19 (6/20/S/L)	1783.48	1165.56	1783.48	4683.41	3914.86	4608.15	0.011	0.19	0.18	1.34	1.16	1.5
Class 20 (6/20/S/H)	1784.1	1190.5	1784.1	6375.16	5189.91	6071.41	0.014	0.43	0.18	0.9	2.48	0.85
Class 21 (6/20/M/L)	2334.79	1480.64	2334.79	6538.44	5071.75	6364.36	0.072	0.29	0.098	0.66	1.455	0.3
Class 22 (6/20/M/H)	2334.79	1437.58	2334.79	8006.94	5941.31	7532.2	0.007	0.47	0.09	0.43	3.16	0.23
Class 23 (6/20/H/L)	1901.43	1226	1901.43	6093.52	4958.93	5826.34	0.009	0.343	0.14	0.945	1.882	0.58
Class 24 (6/20/H/H)	1901.43	1272.47	1901.43	7028.42	5589.78	6527.01	0.011	0.7	0.17	0.61	3.21	0.57
Mean (6/20)	**2006.67**	**1295.46**	**2006.67**	**6434.53**	**5111.09**	**6154.91**	**0.02**	**0.4**	**0.143**	**0.81**	**2.22**	**0.51**
Mean	**1291.2**	**869.1**	**1291.2**	**3731.97**	**2918.26**	**3549.15**	**0.01**	**0.15**	**0.06**	**0.3**	**0.75**	**0.22**

the smallest average values (10.23%), followed by *AREFL** (14.64%). The impact of the problem size on the absolute gap can be seen more dramatically for formulations, while gaps for reformulations may be even observed to decrease. Considering relative gaps (i.e., the final gap value provided by the optimization package), *AGGFL* and *AREFL** reformulations are able to find the smallest gaps on all classes except for Classes 14 and 16, where *AGG* found better results.



(a) Medium Objects, Solution Time (b) Medium Objects, MIP Values

Figure 3: Solution times (left) and MIP values (right) with medium object length, using *AGGFL*

Next, we briefly discuss the impact of some key input parameters on the solution process in order to provide some insights with respect to parameter sensitivities. In Figure 3, we

Table 4: Evaluation of the upper bounds (feasible solutions) obtained by the proposed mathematical models.

Classes	*MIP* solution (Z_{IP})						Time for the *MIP* solution (seconds) (T_{IP})					
($\|T\|/\|P\|/L/h_{it}$)	*AGG*	*AJS*	*ARE*	*AGGFL*	*AJSFL*	*AREFL**	*AGG*	*AJS*	*ARE*	*AGGFL*	*AJSFL*	*AREFL**
Class 1 (3/10/S/L)	1437.24	1438.1	2677.53	1439.49	1441.32	1439.49	10.93	1313.81	359.44	6.07	1551	161.62
Class 2 (3/10/S/H)	1985.66	2007.77	1989.5	1985.66	2008.05	1985.66	26.1	1661.7	586.75	11.97	1793	224.8
Class 3 (3/10/M/L)	1874.08	1875.93	1874.84	1874.86	1877.02	1874.86	4.11	1473.51	453.51	2.5	1706	82.35
Class 4 (3/10/M/H)	2359.36	2375.12	2359.36	2359.36	2368	2359.36	5	1800	35.77	7	1712	18.4
Class 5 (3/10/H/L)	1795.53	1807.34	1796.1	1796.21	1802.8	1796.21	26.55	1625.74	668.23	15.1	1789	186.89
Class 6 (3/10/H/H)	2233.85	2301.18	2233.9	2233.9	2286.53	2233.9	269.66	1800	301.64	282.43	1792	118.16
Mean (3/10)	**1947.62**	**1967.57**	**2155.2**	**1948.55**	**1963.95**	**1948.25**	**56.95**	**1612.46**	**400.79**	**54.18**	**1723.33**	**132.03**
Class 7 (3/20/S/L)	2645.96	2805.84	2677.53	2642.06	2712.83	2648.13	1482.88	1800	1322.8	1001.87	1800	1023.841
Class 8 (3/20/S/H)	3628.53	4275.3	3668.1	3626.94	3789.4	3619.4	1666.36	1800	1481.31	1618.07	1800	1308.8
Class 9 (3/20/M/L)	3519.58	3761.69	3544.2	3516.96	3597.01	3518.04	798.73	1800	1570.74	417.89	1800	470.93
Class 10 (3/20/M/H)	4419.78	5115.2	4403.85	4417.93	4582.19	4402.81	1188.14	1793	547.27	1171.12	1796	524.83
Class 11 (3/20/H/L)	3400.57	3854.08	3452.39	3397.06	3488.2	3400.97	1387.615	1800	1391.23	1303.2	1800	782.69
Class 12 (3/20/H/H)	4181.97	5124.53	4120.76	4168.01	4415.48	4115.91	1762.23	1800	985.77	1795	1800	882.65
Mean (3/20)	**3692.73**	**4156.11**	**3644.7**	**3628.16**	**3764.18**	**3617.54**	**1380.2**	**1800**	**1216.52**	**1217.86**	**1800**	**832.29**
Class 13 (6/10/S/L)	2751.34	2891.51	2850.63	2759.1	2784.95	2787.51	667.8	1800	1736.44	379.12	1800	1453.279
Class 14 (6/10/S/H)	3966.1	4346.57	4063.34	3968.88	4064.04	3996.17	996.38	1800	1756.18	1056.38	1800	1545.53
Class 15 (6/10/M/L)	3638.02	3844.86	3728.78	3639.83	3672.543	3658.28	289.05	1800	1707.31	238.24	1715	1347.28
Class 16 (6/10/M/H)	4761.7	5169.94	4775.92	4761.68	4902.62	4772.42	416.44	1800	1225.53	610.87	1714	987.1
Class 17 (6/10/H/L)	3520.79	3804.11	3694.85	3519.35	3617.17	3570.02	966.97	1800	1685.3	910.42	1800	1427.35
Class 18 (6/10/H/H)	4497.37	5139.88	4543.85	4510.18	4724.46	4510.54	1398.83	1800	1357.4	1448.75	1800	1247.2
Mean (6/10)	**3855.89**	**4199.48**	**3942.9**	**3859.84**	**3960.96**	**3882.49**	**789.24**	**1800**	**1578.03**	**773.96**	**1771.5**	**1334.62**
Class 19 (6/20/S/L)	5114.07	36046.3 [1]	6058.94	5062.43	5462.07	5217.38	1800	1800	1800	1800	1800	1800
Class 20 (6/20/S/H)	7266.47	7.27×10^6 [2]	8490.57	7220.55	8260.91	7453.32	1800	1800	1800	1800	1800	1767.26
Class 21 (6/20/M/L)	6916.9	8258.1 [1]	7642.46	6886.58	7270	6978.96	1711.3	1800	1793.58	1711.22	1800	1722.58
Class 22 (6/20/M/H)	9006.55	3×10^5	9018.74	9017.12	10066.8	8913.74	1708.3	1800	1766.17	1719.2	1800	1707.34
Class 23 (6/20/H/L)	6777.98	9731.54 [3]	7862.44	6653.63	7303.25	6840.87	1800	1800	1800	1800	1800	1800
Class 24 (6/20/H/H)	8545.51	4×10^7 [2]	8767.3	8703.31	10630.6 [3]	8312.26	1800	1800	1800	1800	1800	1766.42
Mean (6/20)	**7271.25**	**7.9×10^6**	**7973.41**	**7257.27**	**8165.60**	**7286.09**	**1769.93**	**1800**	**1793.29**	**1771.74**	**1800**	**1760.6**
Mean	**4162.94**	**2.1×10^6**	**4424**	**4158.73**	**4452.34**	**4169.38**	**981.94**	**1750.77**	**1242.39**	**932.94**	**1773**	**1004.79**

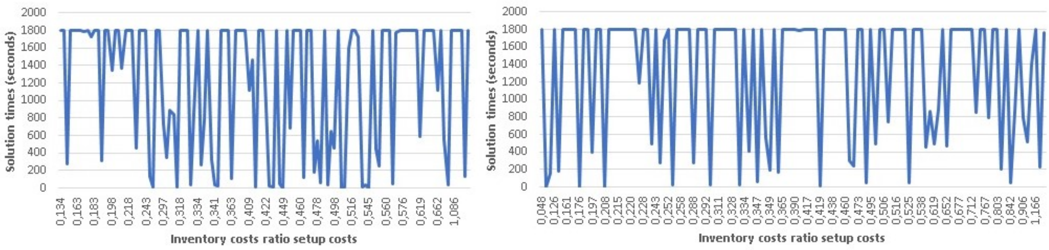
[1]15 instances were solved; [2]9 instances were solved; [3]17 instances were solved.

Table 5: Evaluation of the absolute and relative Gaps obtained by the proposed mathematical models.

Classes	Absolute Gap (Gap_A) (%)						Relative Gap (Gap_R) (%)					
($\|T\|/\|P\|/L/h_{it}$)	*AGG*	*AJS*	*ARE*	*AGGFL*	*AJSFL*	*AREFL**	*AGG*	*AJS*	*ARE*	*AGGFL*	*AJSFL*	*AREFL**
Class 1 (3/10/S/L)	52.28	65.33	52.32	7.51	24.8	17.24	0.006	2.86	0.43	0.005	6.9	0.05
Class 2 (3/10/S/H)	65.9	75.22	65.46	12.31	33.1	9.03	0.008	10.7	0.67	0.007	16.1	0.006
Class 3 (3/10/M/L)	56.18	70.41	56.18	6.48	28.91	9.1	0.006	4.32	0.08	0.005	8.37	0.004
Class 4 (3/10/M/H)	65.23	76.89	65.24	12	36.29	17.18	0.005	12.01	0.005	0.005	14.86	0.005
Class 5 (3/10/H/L)	63.49	73.8	63.5	10.1	29.41	12.87	0.007	7.66	0.51	0.006	12.22	0.03
Class 6 (3/10/H/H)	70.6	79.45	70.6	17.54	38.3	22.4	0.14	19.6	0.2	0.23	21.31	0.007
Mean (3/10)	**62.28**	**73.5**	**62.22**	**10.99**	**31.8**	**14.64**	**0.029**	**9.52**	**0.31**	**0.043**	**13.29**	**0.017**
Class 7 (3/20/S/L)	55.57	69.22	56.05	6	22.9	15.03	2	18.17	2.6	0.85	15.04	1.62
Class 8 (3/20/S/H)	67.5	79.79	67.86	9.97	30.44	7.28	3.4	31.42	3.5	2.63	19.34	0.86
Class 9 (3/20/M/L)	60.81	75.42	60.88	5.26	27.73	8.37	0.57	20.46	0.87	0.037	13.92	0.26
Class 10 (3/20/M/H)	68.74	81.73	68.62	9.94	35	15.52	1.67	29.35	0.27	1.6	18.65	0.13
Class 11 (3/20/H/L)	63.6	76.78	64.13	7.29	26.86	11.77	2.18	26.92	3.23	1.15	15.8	1
Class 12 (3/20/H/H)	70.39	82.48	69.87	14.7	36.25	20.44	5.5	37.41	1.1	5.3	25.1	0.71
Mean (3/20)	**64.43**	**77.57**	**64.57**	**8.86**	**29.86**	**13.1**	**2.55**	**27.29**	**1.92**	**1.93**	**17.98**	**0.74**
Class 13 (6/10/S/L)	58.65	72.97	60.06	6.23	24.6	9.17	0.39	16.82	5.94	0.072	9.46	2.91
Class 14 (6/10/S/H)	71.22	82.02	71.9	10.58	33.34	17	0.9	26.4	4.87	0.97	17.83	2.53
Class 15 (6/10/M/L)	65.42	78.88	60.07	6.38	28.86	10.26	0.089	19.22	3.75	0.056	8.7	1.53
Class 16 (6/10/M/H)	73.56	84.23	73.64	11.65	37.91	18.61	0.25	26.38	1.16	0.38	18.73	0.86
Class 17 (6/10/H/L)	69.4	80.31	70.8	9.73	30	14.72	0.64	22.82	7.25	0.16	13.91	2.95
Class 18 (6/10/H/H)	76.01	85.43	76.14	18.1	40.31	24.26	2.66	34.73	3.27	3.94	26.4	2.12
Mean (6/10)	**69.04**	**80.64**	**68.77**	**10.44**	**32.5**	**15.67**	**0.82**	**24.39**	**4.37**	**0.92**	**15.84**	**2.27**
Class 19 (6/20/S/L)	65.46	81.2 [1]	70.22	7.58	28.25	11.81	6.56	33.8	20.1	4.1	20.45	6.93
Class 20 (6/20/S/H)	75.53	92.34 [2]	78.8	11.85	37.13	18.43	7.24	62.64	19.13	6.4	30.64	7.59
Class 21 (6/20/M/L)	66.32	81.98 [1]	68.82	5.16	30.1	8.95	3.1	27.63	9.98	2.33	18.46	2.94
Class 22 (6/20/M/H)	74.13	88.19	74.16	11.38	40.92	15.71	5.1	42.19	3.38	5.63	29.86	2.1
Class 23 (6/20/H/L)	72.1	85.59 [3]	75.72	8.56	32.1	14.82	6.47	43.17	18.6	4.39	24.8	5.88
Class 24 (6/20/H/H)	77.79	95.22 [2]	78.31	19.31	47.25 [3]	21.51	9.97	77.47	8.48	2.35	41.84	3.86
Mean (6/20)	**71.89**	**87.42**	**74.33**	**10.64**	**35.96**	**15.2**	**6.41**	**47.82**	**13.28**	**4.2**	**27.68**	**4.88**
Mean	**66.91**	**79.79**	**67.47**	**10.23**	**32.53**	**14.64**	**2.45**	**27.26**	**4.97**	**1.77**	**18.7**	**1.97**

[1]15 instances were solved; [2]9 instances were solved; [3]17 instances were solved.

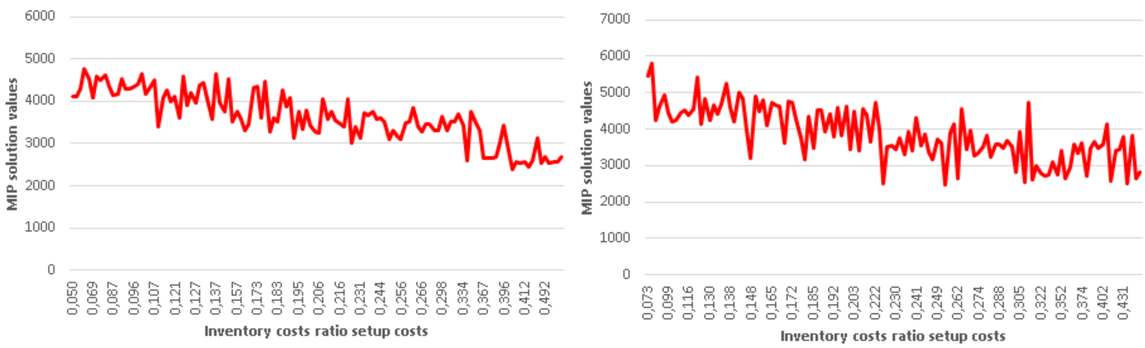
present the solution times and *MIP* values for *AGGFL* (the overall most effective solution method) when the object length is set as medium (similar results were obtained for small and large objects). Although we cannot reach a conclusive statement with respect to how object length impacts solution process of the problem, it is noteworthy to remark that its impact is rather limited, and other specifics of the test instance, in particular its problem size, have more significant impact on the solution process. On the other hand, when we look into *MIP* values, the profile is almost identical for high and small object lengths, albeit with increased values for medium size objects. To conclude, we highlight that similar patterns are observed when *AREFL** and the others solution methods are evaluated in a similar fashion (this pattern is not specific to *AGGFL* and is representative of the general behaviour).



(a) *AREFL** - Instances with $|T| = 3$, $|P| = 20$ (b) *AREFL** - Instances with $|T| = 6$, $|P| = 10$

Figure 4: *AREFL** solution times vs. the ratio between inventory and setup costs

Next, in Figure 4, we demonstrate two plots for the relationship between the solution times and the ratio between setup and inventory costs, when *AREFL** is the solution method. As we have already observed the problem size as the most significant factor for the solution process, in each of these figures, we included instances of the same number of periods and items, and ranked them from the smallest ratio to the highest. As the random nature of these two plots indicate, there is no impact of the ratio between setup and inventory costs on the solution process. We remark that we have made the same observation when different problems sizes or solution methods are considered.



(a) Instances with $|T| = 3$, $|P| = 20$ (b) Instances with $|T| = 6$, $|P| = 10$

Figure 5: *AGGFL MIP* values vs. the ratio between inventory and setup costs

Finally, in Figure 5, we demonstrate two plots for the relationship between the *MIP* values and the ratio between setup and inventory costs, when $AGGFL$ is used as the solution method (as before, solution method or different problem sizes did not impact the behaviour observed). It is interesting to observe that as the ratio goes higher (i.e., the inventory cost going higher with respect to setup cost), the overall cost goes lower. However, it is important to recall that there are still at times significant absolute gaps in these instances.

6.2. Computational results for the exact and heuristic solution approaches

In order to better evaluate the formulations proposed in this paper, the best mathematical models obtained from the analysis in Subsection 6.1 (i.e, *ARE*, AGG and reformulation) were tested on difficult instances. To this aim, in the second data set used to generate instances, we only vary the item length (based on Gau and Wäscher (1995)) while keeping the object size (medium), the inventory costs (low) and the setup cost as constant. Hence, the parameter specifics are as follows:

- number of periods: $|T| = \{3, 6\}$;
- number of items: $|P| = \{10, 20\}$;
- item length: $l_i \in \begin{cases} [0.01, 0.2]L, & \text{Small;} \\ [0.01, 0.8]L & \text{Medium;} \\ [0.2, 0.8]L, & \text{High.} \end{cases}$

Considering the difficulty encountered in problem solution, we split the presentation of the computational results into two parts. Firstly, we will present the results for classes 1 to 6, which contain 3 periods. Then, we will present the results for classes 7 to 12 with 6 periods. Throughout the section, the heuristic approach for AGG and $AGGFL$ are denoted as *AGG-H* and *AGGFL-H*, respectively.

Table 6 presents the average values for the *LP* relaxation and computational time to solve the *LP* in the same fashion as in Table 3. The proposed instances presented convergence issues during the column generation procedure, i.e., tailing off effect. Hence, in order to obtain valid lower bounds for the *AGG-H* and *AGGFL-H* models, before convergence, we use the corresponding Lagrangian bound (see, e.g., Degraeve and Jans (2007) and Vanderbeck and Wolsey (2010)). We can still observe the good performance of the facility location reformulations regarding the lower bound improvements, when compared to the original formulations. For all classes, the improvements of the facility location reformulations range from 126.6% (class 3 with *AREFL**) up to 208.9% (class 4 with *AGGFL-H*). On overall average, the improvements are about 159% and 163.9% for the *AREFL** and *AGGFL-H* formulations, respectively. The computational time required for solving the *LP* reached its limit for class 4 while using the *AREFL** model. The classes 1 and 4 were the most difficult ones to solve when using the *AREFL** model, their solution time represents 97.9% of the total time for solving classes 1 to 3, and 99.4% of the total time for solving classes 4 to 6, respectively. An explanation to this behavior can be seen by the features of such classes, in which the length of the items are very small when compared to the objects, hence, the

Table 6: Evaluation of the lower bounds obtained by the proposed mathematical models from class 1 to 6.

Classes	*LP* relaxation solution (Z_{LP})				Time for *LP* relaxation (seconds) (T_{LP})			
($\|T\|/\|P\|/l_i$)	*ARE*	*AGG-H*	*AREFL**	*AGGFL-H*	*ARE*	*AGG-H*	*AREFL**	*AGGFL-H*
Class 1 (3/10/S)	132.15	123.6	323.62	305.59	14	0	73.15	0
Class 2 (3/10/M)	511.43	499.44	1387.21	1388.72	0.1	0	1.21	0
Class 3 (3/10/H)	781.39	772.75	1770.93	1772.24	0.03	0	0.39	0
Mean (3/10)	**474.99**	**465.26**	**1160.59**	**1155.5**	**4.71**	**0**	**24.9**	**0**
Class 4 (3/20/S)	188.76	186.2	573.65	575.18	1200	0.01	1800	0.02
Class 5 (3/20/M)	1055.22	1045.64	2653.41	2662.26	4.67	0	9.51	0.02
Class 6 (3/20/H)	1195.25	1189.72	3277.7	3343.32	0.1	0.01	1.8	0.01
Mean (3/20)	**813.1**	**807.18**	**2168.25**	**2193.58**	**401.6**	**0**	**603.6**	**0.01**
Mean	**644**	**636.22**	**1664.42**	**1674.53**	**203.15**	**0**	**314.33**	**0**

Table 7: Evaluation of the upper bounds (feasible solutions) obtained by the proposed mathematical models from class 1 to 6.

Class	*MIP* solution (Z_{LP})				Time for *MIP* solution (seconds) (T_{IP})			
($\|T\|/\|P\|/l_i$)	*ARE*	*AGG-H*	*AREFL**	*AGGFL-H*	*ARE*	*AGG-H*	*AREFL**	*AGGFL-H*
Class 1 (3/10/S)	635.74	1078.8	638.13	1087.71	1800	6.44	1800	4.89
Class 2 (3/10/M)	1502.97	1698.26	1515.15	1705.38	1800	0.34	1800	0.66
Class 3 (3/10/H)	1900.75	2074.84	1916.56	2083.39	1476	0.1	1800	0.35
Mean (3/10)	**1346.48**	**1617.3**	**1356.13**	**1625.5**	**1692**	**2.26**	**1800**	**1.96**
Class 4 (3/20/S)	3674.57	2217.36	2086.61 [1]	2224.91	1800	1213.43	1800	672.79
Class 5 (3/20/M)	2891.82	3194	2948.42	3209.96	1800	96	1800	89.8
Class 6 (3/20/H)	3455.71	3793.75	3514.51	3798	1800	5.52	1800	11.46
Mean (3/20)	**3340.77**	**3068.37**	**2849.85**	**3077.39**	**1800**	**438.31**	**1800**	**258**
Mean	**2343.62**	**2342.83**	**2013**	**2351.44**	**1746**	**220.28**	**1800**	**130**

[1] Feasible solutions were found for 17 instances;

number of possible paths is considerably high, which makes it even difficult to solve the linear relaxations.

Table 7 displays, for each formulation and class, the objective function value found for the *MIP* and the time spent to solve the *MIP* in a similar fashion to Table 4. The *ARE* formulation is able to obtain the best average solution for all classes except class 4, where *AGG-H* reached a better average. On overall average, the formulations present slightly little differences, except that *AREFL** could not solve all the instances. As for the solution time, *ARE* and *AREFL** formulations attained the time limit for all classes except for class 3, which are solved with an average of 1476 seconds. The *AGGFL-H* obtained the better time average, with 15% faster then *AGG-H* for classes 1-3 and 69.8% faster for classes 4-6. In both cases, the most difficult instances were the ones with smaller size items compared to the object.

In Table 8, we present the average values for the absolute and relative gaps obtained by each formulation in each class in the same fashion as in Table 5. However, the relative gap of the *AGG-H* and *AGGFL-H* is obtained via formula (6.2), where Z_{BestLB} is now the best lower bound of the ARE formulation for each respective instance. It is no surprise that the facility location reformulation obtained a better absolute gap when compared to its original

Table 8: Evaluation of the mean absolute and relative gaps obtained by the proposed mathematical models, classes 1 to 6.

Classes	Absolute Gap (Gap_A) (%)				Relative Gap (Gap_R) (%)			
($\|T\|/\|P\|/l_i$)	*ARE*	*AGG-H*	*AREFL**	*AGGFL-H*	*ARE*	*AGG-H*	*FLARE**	*AGGFL-H*
Class 1 (3/10/S)	79.13	88.85	49.15	71.9	45.68	68.32	47.5	68.57
Class 2 (3/10/M)	66.7	70.6	8.4	18.5	1.5	12.91	3.5	13.27
Class 3 (3/10/H)	58.9	62.75	7.6	14.95	0.33	8.8	1.9	9
Mean (3/10)	**68.26**	**74.1**	**21.76**	**35.12**	**15.83**	**30**	**17.63**	**30.28**
Class 4 (3/20/S)	92.43	91.61	72.5 [1]	78.14	85.92	78.87	56.38	74.1
Class 5 (3/20/M)	63.51	67.28	10	27.07	5.7	27.88	8.62	17.98
Class 6 (3/20/H)	65.41	68.64	6.73	11.97	1.18	10	4.29	10
Mean (3/20)	**74.59**	**75.84**	**29.74**	**39**	**31**	**38.91**	**28.28**	**34**
Mean	**71.42**	**74.97**	**25.75**	**37**	**23.41**	**34.45**	**22.95**	**32.14**

[1] Feasible solutions were found for 17 instances;

Table 9: Evaluation of the *LP* relaxation and number of feasible solutions obtained by the proposed mathematical models for instances from class 7 to 12.

Class	*LP* Relaxation				Number of integer feasible solutions			
($\|T\|/\|P\|/l_i$)	*ARE*	*AGG-H*	*AREFL**	*AGGFL-H*	*ARE*	*AGG-H*	*AREFL**	*AGGFL-H*
Class 7 (6/10/S)	196.3	182.53	610.48	595.55	20	20	6	20
Class 8 (6/10/M)	865.54	852.44	2844.38	2811.18	20	20	20	20
Class 9 (6/10/H)	1149	1035.64	3280	3299.73	20	20	20	20
Mean (6/10)	**736.94**	**690.2**	**2249.95**	**2235.47**	**20**	**20**	**15.3**	**20**
Class 10 (6/20/S)	224.25	352.53	727.69	1229.1	0	20	0	20
Class 11 (6/20/M)	1436.79	1422.3	5108.58	5039.21	20	20	20	20
Class 12 (6/20/H)	1855.96	1836.53	6380	6404.46	20	20	20	20
Mean (6/20)	**1172.33**	**1203.77**	**4072.09**	**4224.1**	**13**	**20**	**13**	**20**
Mean	**954.64**	**947**	**3161.02**	**3229.79**	**16.5**	**20**	**14.15**	**20**

formulation. For the relative gaps, the *ARE* and *AGG-H* formulations obtained a better performance than *AREFL** and *AGGFL-H* only for classes 1-3. We note that increasing the size of instances implies in greater absolute and relative gaps when varying the items' length.

As the instance size increases, the solver struggles to solve the small item class instances within the time limit. The analysis of the remaining classes (7-12) is presented in Table 9. Even though the *ARE* model obtained feasible solutions for all instances of class 7, either *ARE* and *AREFL** models could not solve any of the instances of class 10. The upper bound solution time reached it limit of 1800 seconds as well for classes 7 to 12 using *ARE* and *AREFL** models, while the *AGG-H* and *AGGFL-H* models reached the time limit only for classes 7,8,10 and 11. In a overall way the *AGGFL-H* model was 18% faster when obtaining integer solution than the *AGG-H* model for classes 9 as 12.

Even though the overall aspects presented in this section differ from the analysis presented in Section 6.1, where the use of the facility location reformulation resulted in a better performance in aspects such as *MIP* values, absolute and relative gaps, the results obtained from the item length analysis are still encouraging to justify the use of the extended reformulations in cutting stock problems, both in terms of obtaining much stronger lower bounds and also directing the search effectively to high quality solutions of the problem with good computational performance.

7. Conclusion

This paper considers the multi-period cutting stock problem with setup costs for the cutting patterns. One formulation was adapted from the literature, while five others are new formulations proposed for the problem, three of which are reformulations based on the facility location problem with stronger lower bounds. A thorough theoretical study regarding lower bound strength was conducted in order to establish a comparative understanding among these formulations. In addition, a computational study was performed based on randomly generated instances to evaluate the formulations in term of computational performance and in a practical context so that theoretical relationships can be better understood.

The computational experiments were performed over two sets of instances. In the first set, we fix the item length and vary the holding costs and the object length while in the second we fix the holding costs and the object length and vary the item length, resulting in more difficult instances. Both sets of instances have shown that the proposed facility location reformulations significantly improve the quality of the lower bounds: on average, these improvements vary from 174% up to 235% for the first set of instances, and from 126.6% up to 208.9% for the second set of instances. Regarding the upper bound achievements of the proposed formulations when using the first data set, the $AGGFL$ and $AREFL*$ are able to find the smallest relative gaps on 22 out of 24 classes, with 1.77% and 1.97% on average, respectively. In addition, the $AGGFL$ and $AREFL*$ are on average always faster than their respective original formulations. As for the second set of instances, the formulations present slight differences on overall averages, however, as the instances size increase, the arc-flow formulation did not find feasible solution for the small item class with 3 and 6 periods where the $AREFL*$ found the smaller number of feasible solutions. To sum up, we can state that the facility location reformulations proposed for the multi-period cutting stock problem with setup costs not only substantially boost the lower bound values, but also result in an effective search for high quality feasible solutions to the problem, showing the merits of this research.

For future research, efficient solution methods can be developed in order to solve bigger instances of the problem. Algorithmic work that takes advantage of the structure of the model can be developed, such as local search algorithms or other metaheuristics such as genetic algorithms. Another interesting subject for future research is to extend the theoretical insights and reformulations proposed in this paper to cutting stock problems considering different practical aspects, such as: several machines, capacity constraints, sequence-dependent cut setups (Arbib and Marinelli, 2007; Wuttke and Heese, 2018), among others. In addition, other strength strategies, such as shortest path reformulation and (ℓ, S) inequalities can be extended to enhance formulations based on cutting stock problems. Considering the capacitated case, applying Lagrangian relaxation to the capacity constraint will result in a sub-problem similar to the one studied in this paper. A multi-objective approach can also be applied to analyse the complex trade-offs present in the objective function.

We also would like to highlight that the feature considered in this paper, where a cutting pattern produces several items, can be extended to several process industries, where the products are obtained by processes that can produce several types of products simultaneously. These processes can be a specific mode of configuration of a production system that can produce several different items simultaneously and in varied quantities. A recent general discussion about it can be found in Villas Boas et al. (2021). Examples of such process industries are: refineries (Göthe-Lundgren et al., 2002; Shi et al., 2014), molded pulp (Martínez et al., 2018, 2019), electrofused grains (Luche et al., 2009), foundry (de Araujo et al., 2008), offset printing industry (Baumann et al., 2015) industries, among others. In practice, industries define a list of alternative processes as input data, and the decision is related to the selection of the processes to be used in each period of the planning horizon. However, according to Martínez et al. (2019), for some production environments, the number of configurations might be large and hence the complete enumeration not possible while considering only a subset of them may lead to sub-optimal solutions. Hence, an integrated approach that considers the process configuration together with other decisions is also an interesting avenue for future research.

Acknowledgments

This research was funded by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) (Process numbers 305261/2018-5 and 406335/2018-4) and by Fundação de Amparo a Pesquisa do Estado de São Paulo (FAPESP) (Process number 2013/07375-0, 2016/01860-1 and 2018/19893-9). The work of the third author is supported by the Air Force Office of Scientific Research under award number FA9550-18-1-7003. We also thank four anonymous reviewers for their valuable suggestions that helped improve the content and presentation of the paper.

References

Absi, N., van den Heuvel, W., 2019. Worst case analysis of relax and fix heuristics for lot-sizing problems. European Journal of Operational Research 279, 449 – 458.

Akartunalı, K., Fragkos, I., Miller, A., Wu, T., 2016. Local cuts and two-period convex hull closures for big-bucket lot-sizing problems. INFORMS Journal on Computing 28, 766–780.

Aktin, T., Özdemir, R.G., 2009. An integrated approach to the one-dimensional cutting stock problem in coronary stent manufacturing. European Journal of Operational Research 196, 737 – 743.

Alem, D., Morabito, R., 2013. Risk-averse two-stage stochastic programs in furniture plants. OR Spectrum 35, 773–806.

Alem, D., Oliveira, F., Peinado, M., 2020. A practical assessment of risk-averse approaches in production lot-sizing problems. International Journal of Production Research 58, 2581 – 2603.

Aliano Filho, A., Moretti, A.C., Pato, M.V., 2018. A comparative study of exact methods for the bi-objective integer one-dimensional cutting stock problem. Journal of the Operational Research Society 69, 91–107. URL: https://doi.org/10.1057/s41274-017-0214-7, doi:10.1057/s41274-017-0214-7, arXiv:https://doi.org/10.1057/s41274-017-0214-7.

Aloisio, A., Arbib, C., Marinelli, F., 2011a. Cutting stock with no three parts per pattern: Work-in-process and pattern minimization. Discrete Optimization 8, 315 – 332.

Aloisio, A., Arbib, C., Marinelli, F., 2011b. On lp relaxations for the pattern minimization problem. Networks 57, 247 – 253.

Alves, C., Valério de Carvalho, J.M., 2008a. A branch-and-price-and-cut algorithm for the pattern minimization problem. RAIRO- Operations Research 42, 435–453.

Alves, C., Valério de Carvalho, J.M., 2008b. A stabilized branch-and-price-and-cut algorithm for the multiple length cutting stock problem. Computers & Operations Research 35, 1315 – 1328.

de Araujo, S.A., Poldi, K.C., Smith, J., 2014. A genetic algorithm for the one-dimensional cutting stock problem with setups. Pesquisa Operacional 34, 165 – 187.

de Araujo, S.A., de Reyck, B., Degraeve, Z., Fragkos, I., Jans, R., 2015. Period decompositions for the capacitated lot sizing problem with setup times. INFORMS Journal on Computing 27, 431–448.

Arbib, C., Marinelli, F., 2005. Integrating process optimization and inventory planning in cutting-stock with skiving option: An optimization model and its application. European Journal of Operational Research 163, 617 – 630.

Arbib, C., Marinelli, F., 2007. An optimization model for trim loss minimization in an automotive glass plant. European Journal of Operational Research 183, 1421–1432. URL: https://www.sciencedirect.com/science/article/pii/S0377221706003146, doi:https://doi.org/10.1016/j.ejor.2005.07.034.

Arbib, C., Marinelli, F., 2014. On cutting stock with due dates. Omega 46, 11 – 20.

Attila, Ö.N., Agra, A., Akartunalı, K., Arulselvan, A., 2021. Robust formulations for economic lot-sizing problem with remanufacturing. European Journal of Operational Research 288, 496 – 510.

Barany, I., Van Roy, T., Wolsey, L., 1984. Strong formulations for multi-item capacitated lot-sizing. Manag. Sci. 30, 1255–1261.

Baumann, P., Forrer, S., Trautmann, N., 2015. Planning of a make-to-order production process in the printing industry. Flex. Serv. Manuf. J. 27, 534—-560.

Ben Amor, H., Valério de Carvalho, J., 2005. Cutting stock problems, in: Desaulniers, G., Desrosiers, J., Solomon, M.M. (Eds.), Column Generation. Springer US, pp. 131–161.

Bonnevay, S., Gavin, G., Aubertin, P., 2016. Comparison of two metaheuristics to solve a 2-d cutting stock problem with set-up cost in the paper industry. Int. J. Metaheuristics 5, 31–50. URL: https://doi.org/10.1504/IJMHEUR.2016.079107, doi:10.1504/IJMHEUR.2016.079107.

Braga, N., Alvez, C., Macedo, R., de Carvalho, J.V., 2015. A model-based heuristic for the combined cutting stock and scheduling problem. Lecture Notes in Computer Science 9156, 409 – 505.

Brahimi, N., Absi, N., Dauzère-Pérès, S., Nordli, A., 2017. Single-item dynamic lot-sizing problems: An updated survey. European Journal of Operational Research 263, 838–863.

Christofoletti, M.M., de Araujo, S.A., Cherri, A.C., 2021. Integrated lot-sizing and cutting stock problem applied to the mattress industry. Journal of the Operational Research Society 72, 1279–1293. URL: https://doi.org/10.1080/01605682.2020.1718013, doi:10.1080/01605682.2020.1718013, arXiv:https://doi.org/10.1080/01605682.2020.1718013.

Cui, Y., Yang, L., Zhao, Z., Tang, T., Yin, M., 2013. Sequential grouping heuristic for the two-dimensional cutting stock problem with pattern reduction. International Journal of Production Economics 144, 432–439.

Cui, Y., Zhong, C., Yao, Y., 2015. Pattern-set generation algorithm for the one-dimensional cutting stock problem with setup cost. European Journal of Operational Research 243, 540 – 546.

Côté, J.F., Iori, M., 2018. The meet-in-the-middle principle for cutting and packing problems. INFORMS Journal on Computing 30, 646–661.

de Araujo, S.A., Arenales, M.N., Clark, A.R., 2008. Lot sizing and furnace scheduling in small foundries. Computers Operations Research 35, 916–932. URL: https://www.sciencedirect.com/science/article/pii/S0305054806001390, doi:https://doi.org/10.1016/j.cor.2006.05.010. part Special Issue: New Trends in Locational Analysis.

Degraeve, Z., Jans, R., 2007. A new dantzig-wolfe reformulation and branch-and-price algorithm for the capacitated lot-sizing problem with setup times. Operations Research 55, 909–920.

Delorme, M., Iori, M., 2019. Enhanced pseudo-polynomial formulations for bin packing and cutting stock problems. INFORMS Journal on Computing 109, 1–19.

Diegel, A., Miller, G., Montocchio, E., van Schalkwyk, S., Diegel, O., 2006. Enforcing minimum run length in the cutting stock problem. European Journal of Operational Research 171, 708–721. URL: `https://www.sciencedirect.com/science/article/pii/S037722170400760X`, doi:https://doi.org/10.1016/j.ejor.2004.09.039.

Doostmohammadi, M., Akartunalı, K., 2018. Valid inequalities for two-period relaxations of big-bucket lot-sizing problems: Zero setup case. European Journal of Operational Research 267, 86 – 95.

Eppen, G.D., Martin, R.K., 1987. Solving multi-item capacitated lot-sizing problems using variable redefinition. Operations Research 35, 832 – 848.

Fiorotto, D.J., Jans, R., de Araujo, S.A., 2017. An analysis of formulations for the capacitated lot sizing problem with setup crossover. Computers & Industrial Engineering 106, 338 – 350.

Gau, T., Wäscher, G., 1995. Cutgen1: A problem generator for the standard one-dimensional cutting stock problem. European Journal of Operational Research 84, 572–579. URL: `https://www.sciencedirect.com/science/article/pii/037722179500023J`, doi:https://doi.org/10.1016/0377-2217(95)00023-J. cutting and Packing.

Ghidini, C.T.L.S., Alem, D., Arenales, M.N., 2007. Solving a combined cutting stock and lot-sizing problem in small furniture industries, in: Proceedings of the 6th International Conference on Operational Research for Development (VI-ICORD).

Gilmore, P.C., Gomory, R.E., 1961. A linear programming approach to the cutting-stock problem. Operations Research 9, 849 – 859.

Gilmore, P.C., Gomory, R.E., 1963. A linear programming approach to the cutting stock problema - part ii. Operations Research 11, 863 – 888.

Golfeto, R.R., Moretti, A.C., de Salles Neto, L.L., 2009. A genetic symbiotic algorithm applied to the cutting stock problem with multiple objectives. Advanced Modeling and Optimization 11, 473 – 501.

Gramani, M.C.N., França, P.M., 2006. The combined cutting stock and lot-sizing problem in industrial processes. European Journal of Operational Research 174, 509 – 521.

Gramani, M.C.N., França, P.M., Arenales, M.N., 2009. A lagrangian relaxation approach to a coupled lot-sizing and cutting stock problem. International Journal of Production Economics 119, 219 – 227.

Gruson, M., Bazrafshan, M., Cordeau, J.F., Jans, R., 2019. A comparison of formulations for a three-level lot sizing and replenishment problem with a distribution structure. Computers & Operations Research 111, 297 – 310.

Göthe-Lundgren, M., T. Lundgren, J., A. Persson, J., 2002. An optimization model for refinery production scheduling. International Journal of Production Economics 78, 255–270. URL: `https://www.sciencedirect.com/science/article/pii/S0925527300001626`, doi:https://doi.org/10.1016/S0925-5273(00)00162-6.

Haessler, R.W., 1988. Selection and design of heuristic procedures for solving roll trim problems. Management Science 34, 1460–1471.

Hendry, L.C., Fok, K.K., Shek, K.W., 1996. A cutting stock and scheduling problem in the copper industry. The Journal of the Operational Research Society 47, 38 – 47.

Henn, S., Wäscher, G., 2013. Extensions of cutting problems: setups. Pesquisa Operacional 33, 133 – 162.

Johnston, R.E., Sadinlija, E., 2004. A new model for complete solutions to one-dimensional cutting stock problems. European Journal of Operational Research 153, 176– 183.

Kantorovich, L.V., 1960. Mathematical methods of organizing and planning production. Management Science 6, 366 – 422.

Kolen, A.W.J., Spieksma, F.C.R., 2000. Solving a bi-criterion cutting stock problem with open-ended demand: a case study. Journal of the Operational Research Society 51, 1238–1247. URL: `https://doi.org/10.1057/palgrave.jors.2601023`, doi:10.1057/palgrave.jors.2601023, `arXiv:https://doi.org/10.1057/palgrave.jors.2601023`.

Krarup, J., Bilde, O., 1977. Plant location, set covering and economic lot size: An 0 (mn) - algorithm for structured problems, in: Collatz, L., Meinardus, G., Wetterling, W. (Eds.), Numerische Methoden bei Optimierungsaufgaben Band 3. Birkhäuser Basel. volume 36 of *International Series of Numerical Mathematics*, pp. 155 – 180.

Luche, J., Morabito, R., Pureza, V., 2009. Combining process selection and lot sizing models for production scheduling of electrofused grains. Asia-Pacific Journal of Operational Research 26, 421–443. doi:10.1142/S0217595909002286.

Ma, N., Bai, X., Dong, X., 2021. Hybrid heuristic for the production replanning problem under varying demands in manufacturing industries. Engineering Optimization 0, 1–18. URL: https://doi.org/10.1080/0305215X.2021.2002315, doi:10.1080/0305215X.2021.2002315, arXiv:https://doi.org/10.1080/0305215X.2021.2002315.

Ma, N., Liu, Y., Zhou, Z., 2019. Two heuristics for the capacitated multi-period cutting stock problem with pattern setup cost. Computers & Operations Research 109, 218–229.

Ma, N., Liu, Y., Zhou, Z., Chu, C., 2018. Combined cutting stock and lot-sizing problem with pattern setup. Computers Operations Research 95, 44–55. URL: https://www.sciencedirect.com/science/article/pii/S0305054818300558, doi:https://doi.org/10.1016/j.cor.2018.02.016.

Martínez, K., Adulyasak, Y., Jans, R., Morabito, R., Toso, E., 2019. An exact optimization approach for an integrated process configuration, lot-sizing, and scheduling problem. Computers and Operations Research 103, 310–323.

Martínez, K.P., Morabito, R., Toso, E.A.V., 2018. A coupled process configuration, lot-sizing and scheduling model for production planning in the molded pulp industry. International Journal of Production Economics 204, 227–243.

McDiarmid, C., 1999. Pattern minimisation in cutting stock problems. Discrete Applied Mathematics 98, 121 – 130.

Melega, G.M., de Araujo, S.A., Jans, R., 2018. Classification and literature review of integrated lot-sizing and cutting stock problems. European Journal of Operational Research 271, 1 – 19.

Nonås, S.L., Thorstenson, A., 2000. A combined cutting-stock and lot-sizing problem. European Journal of Operational Research 120, 327 – 342.

Nonås, S.L., Thorstenson, A., 2008. Solving a combined cutting-stock and lot-sizing problem with a column generating procedure. Computers & Operations Research 35, 3371 – 3392.

Oliveira, W.A., Fiorotto, D.J., Song, X., Jones, D.F., 2021. An extended goal programming model for the multiobjective integrated lot-sizing and cutting stock problem. European Journal of Operational Research 295, 996–1007. URL: https://www.sciencedirect.com/science/article/pii/S0377221721002757, doi:https://doi.org/10.1016/j.ejor.2021.03.049.

Poldi, K.C., de Araujo, S.A., 2016. Mathematical models and a heuristic method for the multiperiod one-dimensional cutting stock problem. Annals of Operations Research 238, 497 – 520.

Quezada, F., Gicquel, C., Kedad-Sidhoum, S., Vu, D.Q., 2020. A multi-stage stochastic integer programming approach for a multi-echelon lot-sizing problem with returns and lost sales. Computers & Operations Research 116, 104865.

Reinertsen, H., Vossen, T.W., 2010. The one-dimensional cutting stock problem with due dates. European Journal of Operational Research 201, 701 – 711.

Santos, S.G., de Araujo, S.A., Rangel, M.S.N., 2011. Integrated cutting machine programming and lot sizing in furniture industry. Pesquisa Operacional para o Desenvolvimento 3, 1–17.

Shi, L., Jiang, Y., Wang, L., Huang, D., 2014. Refinery production scheduling involving operational transitions of mode switching under predictive control system. Ind. Eng. Chem. Res. 53, 8155—-8170.

Silva, E., Viães, C., Oliveira, J.F., Carravilla, M.A., 2015. Integrated cutting and production planning: A case study in a home textile manufacturing company, in: Póvoa, A.P.F.D.B., de Miranda, J.L. (Eds.), Operations Research and Big Data. Springer International Publishing. volume 15 of *Studies in Big Data*, pp. 213–220.

Song, X., Bennell, J., 2014. Column generation and sequential heuristic procedure for solving an irregular shape cutting stock problem. Journal of the Operational Research Society 65, 1037 – 1052.

Suliman, S.M.A., 2012. An algorithm for solving lot sizing and cutting stock problem within aluminum

fabrication industry, in: International Conference on Industrial Engineering and Operations Management, 2012.
Tomat, L., Gradisar, M., 2017. One-dimensional stock cutting: optimization of usable leftovers in consecutive orders. Cent Eur J Oper Res 25, 473–489.
Trkman, P., Gradisar, M., 2007. One-dimensional cutting stock optimization in consecutive time periods. European Journal of Operational Research 179, 291 – 301.
Umetani, S., Yagiura, M., Ibaraki, T., 2003. One-dimensional cutting stock problem to minimize the number of different patterns. European Journal of Operational Research 146, 388 – 402.
Valério de Carvalho, J.M., 1999. Exact solution of bin-packing problems using column generation and branch-and-bound. Annals of Operations Research 86, 629–659.
Valério de Carvalho, J.M., 2002. Lp models for bin packing and cutting stock problems. European Journal of Operational Research 141, 253 – 273.
Van Vyve, M., Wolsey, L., Yaman, H., 2014. Relaxations for two-level multi-item lot-sizing problems. Math. Program. 146, 495–523.
Vanderbeck, F., 2000. Exact algorithm for minimising the number of setups in the one-dimensional cutting stock problem. Operations Research 48, 915–926.
Vanderbeck, F., Wolsey, L.A., 2010. Reformulation and Decomposition of Integer Programs. Springer Berlin Heidelberg, Berlin, Heidelberg. pp. 431–502.
Vanzela, M., Melega, G.M., Rangel, S., de Araujo, S.A., 2017. The integrated lot sizing and cutting stock problem with saw cycle constraints applied to furniture production. Computers & Operations Research 79, 148 – 160.
Villas Boas, B.E., Camargo, V.C.B., Morabito, R., 2021. Modeling and mip-heuristics for the general lotsizing and scheduling problem with process configuration selection. Pesquisa Operacional 41, 1–29.
Wäscher, G., Haußner, H., Schumann, H., 2007. An improved typology of cutting and packing problems. European Journal of Operational Research 183, 1109 – 1130.
Wolsey, L., 1998. Integer Programming. Wiley Series in Discrete Mathematics and Optimization, Wiley. URL: `https://books.google.com.br/books?id=x7RvQgAACAAJ`.
Wolsey, L.A., 1977. Valid inequalities, covering problems and discrete dynamic programs. Annals of Discrete Mathematics 1, 527–538.
Wu, T., Liang, Z., Zhang, C., 2018. Analytics branching and selection for the capacitated multi-item lot sizing problem with nonidentical machines. INFORMS Journal on Computing 30, 236–258.
Wuttke, D.A., Heese, H.S., 2018. Two-dimensional cutting stock problem with sequence dependent setup times. European Journal of Operational Research 265, 303 – 315.
Yanasse, H.H., Limeira, M.S., 2006. A hybrid heuristic to reduce the number of different patterns in cutting stock problems. Computers & Operations Research 33, 2744 – 2756.
Zhao, M., Zhang, M., 2020. Multiechelon lot sizing: New complexities and inequalities. Operations Research 68, 534—-551.