Les Cahiers du GERAD ISSN: 0711–2440

Formulations and exact solution approaches for a coupled bin-packing and lot-sizing problem with sequence-dependent setups

G. M. Melega, S. A. de Araujo, R. Jans, R. Morabito

G–2021–49

Septembre 2021

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

Dépôt légal – Bibliothèque et Archives nationales du Québec, 2021
– Bibliothèque et Archives Canada, 2021

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

Legal deposit – Bibliothèque et Archives nationales du Québec, 2021
– Library and Archives Canada, 2021

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

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

Formulations and exact solution approaches for a coupled bin-packing and lot-sizing problem with sequence-dependent setups

Gislaine Mara Melega [a, b]

Silvio Alexandre de Araujo [b, c, d]

Raf Jans [b, c]

Reinaldo Morabito [a]

[a] *Departamento de Engenharia de Produção, UFSCar – Univ. Federal de São Carlos, SP, 13565-905, Brasil*

[b] *HEC Montréal, Montréal (Qc), Canada, H3T 2A7*

[c] *GERAD, Montréal (Qc), Canada, H3T 1J4*

[d] *Departamento de Matemática, IBILCE, UNESP – Univ. Estadual Paulista, São José do Rio Preto, SP, 15054–000, Brasil*

gislainemelega@gmail.com
silvio.araujo@unesp.br
raf.jans@hec.ca
morabito@ufscar.br

Septembre 2021
Les Cahiers du GERAD
G–2021–49

Abstract : We study bin-packing and lot-sizing decisions in an integrated way. Such a problem appears in several manufacturing settings where items first need to be cut and next assembled into final products. One of the main novelties of this research is the modeling of the complex setup operations for the cutting process. More specifically, we consider the operation regarding the insertion or removal of the knives in the cutting process. Since this operation depends on the number of items cut in the current cutting process and in the previous one, the number of insertions and removals is sequence-dependent. The setups in the lot-sizing problem related to the production of the final products are also sequence-dependent. To deal with such a problem, two compact formulations are proposed, which are based on the assignment variables to model the bin-packing decisions. The sequence-dependent setups in the bin-packing problem are modeled in two different ways. The first one is based on known constraints from the literature and the second one is based on the idea of micro-periods and a phantom cutting process. Due to the dependency of the setup decisions in the bin-packing problem with sequence-dependent setups, the resulting formulations are mixed-integer nonlinear mathematical models. Exact mixed-integer linear programming formulations are presented by applying linearization techniques. An exact branch-and-cut algorithm, which applies violated subtour elimination cuts to deal with the sequence-dependent production and cutting setups, is developed to solve the non-polynomial formulations. In addition, a Benders-based branch-and-cut algorithm using Benders cuts and violated cuts is also presented to solve the integrated problem. A computational study is conducted in order to analyze the impact of the proposed approaches to model sequence-dependent setups and the exact solution methods used to solve the coupled bin-packing and lot-sizing problem.

Keywords: Coupled bin-packing and lot-sizing problem, sequence-dependent setups, branch-and-cut algorithm, Benders-based branch-and-cut algorithm, cutting stock problems

Acknowledgements: This research was funded by the Conselho Nacional de Desenvolvimento Científico e Tecnológico (*CNPq*), Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (*CAPES*) and the Fundação de Amparo a Pesquisa do Estado de São Paulo (FAPESP process n 2016/01860–1, 2018/19893–9, 2019/03302–4).

1 Introduction

The bin-packing (*BP*) and lot-sizing (*LS*) problems have been widely studied in the literature on production planning and considerable progress has been made with respect to formulations and solution methods for these two problems separately. In the *BP* problem, items have to be assigned to a set of objects, which has fixed dimensions, and with the objective of minimizing the total number of objects used (Dyckhoff, 1990; Wäscher et al., 2007; Delorme et al., 2016). In a general definition, the *LS* problem considers the tradeoff between setup and inventory holding costs to determine a production plan that meets the demand of each product at a minimal cost, respecting the available resources (Karimi et al., 2003; Jans and Degraeve, 2008; Glock et al., 2014). In this paper, we are interested in studying the interdependency between these two types of decisions in order to have an extended and detailed view of the production process and contribute to better decision-making. The tendency of integrating decisions has been pointed out as an important aspect of future research in the production planning literature (Drexl and Kimms, 1997; Thomas and Griffin, 1996; Jans and Degraeve, 2008; Melega et al., 2018) and it is motivated by practical applications, for example in the paper, furniture, glass, aircraft, and tiling industries.

In the coupled bin-packing and lot-sizing problem, the decisions taken into account do not focus on the medium-term decisions, i.e., we are not interested in the production planning for the next weeks or months of the production process, where inventory, production quantity, backlog, should be considered. In this paper, we examine issues in a production scheduling perspective, i.e., we are interested in treating the short-term decisions of the integrated problem by addressing production scheduling decisions, for which the scheduling horizon consists of a few days or a week and the production schedule comprises day-to-day, shift-to-shift decisions. In order to model this, we take into account the operations required to perform the processes involved in the *BP* and the *LS* problems, which might result in setups that vary depending on the previous operations, i.e., sequence-dependent setups.

This study addresses a two-phase coupled bin-packing and lot-sizing problem. In a first phase, a strong heterogeneous assortment of items has to be assigned to a minimal number of identical size objects, in the so-defined Single Bin Size Bin Packing Problem (*SBSBPP*) (Wäscher et al., 2007). We will refer to these as (stock) objects rather than bins. This assignment of items to objects, relates to performing cutting operations on the objects in order to obtain the items. Next, in a second phase, these items pass through a process, modeled in the lot-sizing problem, before being ready as products and meet the external demand. Examples of such a process can be drilling, folding, painting, or assembling. The independent demand for products, already defined in the master production plan at the medium-term production planning level, should be met at the end of the short-term scheduling horizon. A lead-time has to be taken into account and imposes that an item cut in a specific time period is only ready to be used in the production process of the next time periods. In each time period of the scheduling horizon, which can be a day or shift, several objects can be cut, as well as several types of products can be produced.

The sequence-dependent setups in the bin-packing problem deal with the sequence of the cutting operations needed to cut each object and, in this specific application, consist of the removal/insertion process of knives in different positions of the cutting machine before the cutting process (referred to as knife movements). The setup time needed for a changeover from one cutting process to another is determined by the total number of knives inserted or removed compared to the previous setup configuration, and depends on the change in the number of items cut from these two subsequent cutting operations. These setups are hence sequence-dependent. After this cutting process, the cut items need to undergo a subsequent process such as drilling, folding or assembling. This second process is modeled by a lot-sizing problem with multiple products. In order to process a specific type of product, a setup is needed and this setup is also sequence-dependent. More specifically, both the setup times and costs are sequence-dependent and this requires decisions on the sequencing of the products. Considering this, the coupled bin-packing and lot-sizing problem deals with the simultaneous scheduling of the bin-packing and lot-sizing operations, respecting the available resources in each process, to meet, at the end of the scheduling horizon, the clients' demand, while minimizing the operational costs.

Figure 1 shows an example of the setup counting when changing over between cutting operations. There are three available objects, which according to the bin-packing decisions, are cut into items using knife configuration 1, 2 and 3 (cutting patterns). Note that an initial and a final knife are always required in this cutting process and this influences the setup count. The total number of knives in knife configuration 1 is three (to cut two items) and in knife configuration 2 is five (to cut four items). It is hence necessary to add two knives (knife 4 and knife 5) in order to cut an object using knife configuration 2 after knife configuration 1, i.e., two setups are necessary (two knives insertions) to perform this changeover. When cutting an object using knife configuration 3, with three items, after knife configuration 2, with 4 items, the difference in the total number of items is 1 and as it is shown, knife number 5 (for example) needs to be removed from the cutting machine. The number of knife movements is hence one. In this schedule of the cutting operations (knife configuration 1, followed by knife configuration 2, and then knife configuration 3), the total number of setups is three (two knives insertions and one knife removal). However, considering another schedule of the cutting operations, for example, knife configuration 1, followed by knife configuration 3 and then by knife configuration 2, the total number of setups needed is two, which consists of adding a knife in a changeover from knife configuration 1 to knife configuration 3 and another knife in a changeover from knife configuration 3 to knife configuration 2. Therefore, the decisions related to the scheduling of the cutting operations can have a large impact on the number of knife insertions and removals, i.e., on the number of setups. Consequently, it influences the available capacity as well as the total operating costs, since these setups can be time-consuming and costly. Note that the type of items cut from the objects is not relevant for this setup counting. Other possible variations to count the number of setups, depending on the number of knives and the type of operations needed to set up the cutting machine, are presented in Appendix B.

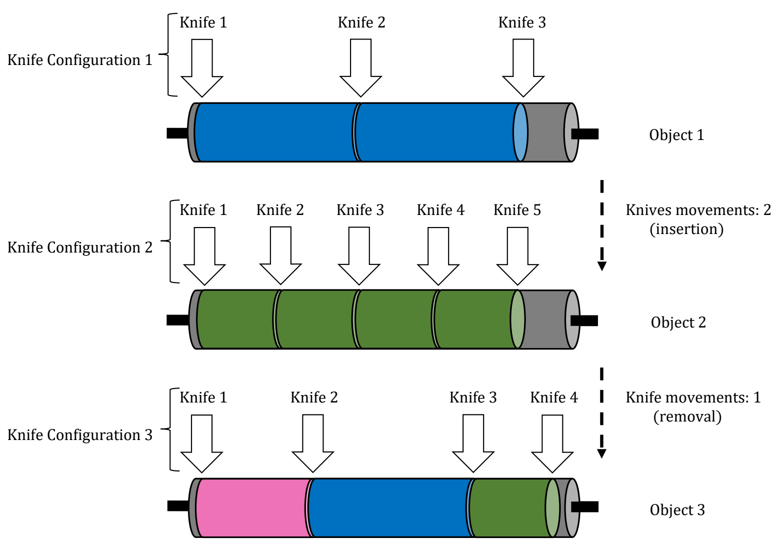

Figure 1: Setup counting in the scheduling of cutting operations

An example of the problem addressed in this paper can be found in the manufacturing process of light aircraft. In this problem, structural metallic tubes are cut into parts of wings, fuselage, instruments and aircraft systems and then used in the assembly of these structures (Abuabara and Morabito, 2008). In this one-dimensional cutting process, the metallic tubes comprise extremely expensive raw materials, which are cut into (mostly) unique parts of structures due to the requirements in the assembly process of aircraft parts, i.e., each cutting pattern is used only a small number of times, and often only once, because of the specific demand for structure parts. Another example arises in the two-dimensional cutting process of customized furniture companies. The orders consist of unique requirements of clients so that wooden plates are cut to produce items that go through some processes (drilling, painting, and assembling) to compose the customized products.

In such companies, due to the unique requirements of products, the manufacturing process of high-volume, low-variety products, that can be seen in several industries (Melega et al., 2020), is substituted by a manufacturing process with a high variety of products which are produced in a small volume, according to the customized requirements of clients. In these customized products, the cost of raw materials may represent a considerable amount of the total cost of the products and the waste of raw materials consists of good-quality scraps which are useless for practical purposes due to their small sizes. During the production of these products, the processes (cutting, drilling, folding, painting, or assembling) might trigger setup operations when switching between them, such as altering knife positions, changing tools, machine adjustments, and cleaning procedures. These setup operations can consume a substantial amount of the available capacity. A more effective scheduling of the operations, by explicitly considering their integration, might free up some capacity that can be used for additional orders, hence increasing productivity. This action might also reduce the amount of raw material used, consequently decreasing the total costs of production. Therefore, there are situations where the scheduling decisions, with sequence-dependent setups, should be simultaneously considered and well managed in order to avoid unnecessary setups, reduction or idleness in capacity, non-compliance of the client's deadline, accumulation of stock, losses in raw material, which can influence the total cost and even the feasibility of the solution when considering practical cases.

Our contributions in the field of integrated production planning problems are as follows: (i) we propose two compact formulations for the coupled bin-packing and lot-sizing problem, with sequence-dependent setups; (ii) we model the sequence-dependent setup counting in the cutting process using two different approaches; (iii) we extend the approaches to consider other possible ways to count the number of setups, depending on the number of knives and the type of operations to set up the cutting machine; (iv) we present an exact branch-and-cut algorithm, applying violated subtour elimination cuts, to solve the non-polynomial formulations; (v) we developed a Benders-based branch-and-cut algorithm using Benders cuts and violated cuts to solve the integrated problem.

This paper is organized as follows. Section 2 presents a discussion of the literature related to the coupled bin-packing and lot-sizing problem. In Section 3, the compact mathematical models proposed for the integrated problem with sequence-dependent setups is shown. Section 4 contains the exact solution approaches based on branch-and-cut and Benders decomposition for the integrated models, followed by a computational study performed in Section 5. Finally, we present our concluding remarks and discuss perspectives for future research in Section 6.

2 Literature review and further discussion

Before presenting the literature review on integrated problems, we firstly discuss the choice regarding the nomenclature used to address the problem in our study. According to the typology presented in Wäscher et al. (2007), the bin-packing (BP) and cutting stock (CS) problems have an identical structure, in which the choice between them may differ due to the assortment of the items. Bin-packing problems are defined as problems having a strong heterogeneous assortment of items, where the demand of items, also denoted as multiplicity factor, is close or even equal to one. On the contrary, in cutting stock problems a weakly heterogeneous assortment of items arises, where the average demand of items results in a large multiplicity factor. In both problems, the available objects are defined in one, two, three, or an even larger number of fixed dimensions. The assortment of objects can be identical, weakly, or strongly heterogeneous. The objective in these problems consists of minimizing the value, number, or total size of the objects necessary to accommodate all items. Another feature used to propose a different classification for bin-packing and cutting stock problems is related to the solution methods that have been developed to solve each one of these problems. According to Valério de

Carvalho (2002), in cutting stock problems, where the demand for items is large, the solution from the linear relaxation, traditionally used to solve cutting stock problems, can usually provide good quality integer solutions by combining this with heuristic procedures. Most of these strategies consider extended formulations based on cutting patterns (Gilmore and Gomory, 1961, 1963). On the other hand, for bin-packing problems, where the demand for items is low, it may not be so easy for heuristic approaches to find a good integer solution from a linear relaxation solution. Thus, to obtain integer solutions to bin-packing problems, other techniques have been used, such as approximation algorithms and exact solution methods. In terms of mathematical modeling, compact formulations have been proposed (Kantorovich, 1960; Valério de Carvalho, 1999, 2002).

Considering these definitions, we can state that the nomenclature regarding cutting and packing problems is directly related to the features of the problem (data) and the used solution approaches rather than the underlying process itself. So in our case, the underlying process is a cutting process, but we refer to it as a bin-packing problem due to the weakly heterogeneous assortment, according to the definitions in the literature (Wäscher et al., 2007). Consequently, our study comprehends a coupled bin-packing and lot-sizing problem. More specifically, we model the bin-packing decisions by a compact formulation using assignment variables (Kantorovich, 1960) and consider a production scheduling perspective by addressing production scheduling decisions, i.e., short-term decisions of the integrated problem.

The only other paper that we found in the literature which considers scheduling decisions within an integrated lot-sizing and cutting stock/bin-packing problem is Melega et al. (2020). The authors of this latter paper consider the Single Stock Size Cutting Stock problem (with a weakly heterogenous assortment of items and identical large objects) in the first production stage, which is integrated with a lot-sizing problem in the second production stage. At both levels, scheduling decisions are taken into account. However, there are two key differences with the paper presented here. The first main difference is that in Melega et al. (2020), the sequence dependent setups at the cutting level depend on the total number of different items between two cutting patterns, whereas in this paper we consider the number of knife insertions and removals, which requires a novel formulation and solution approach. The second main difference is then related to the proposed solution approaches. Melega et al. (2020) propose a price-and-branch approach, in which a column generation is used to generate the matrix of cutting patterns (Gilmore and Gomory, 1961, 1963) and next decomposition approaches, based on relax-and-fix procedures, are used to solve the resulting problem. Due to the formulation and solution methods, the scheduling decisions are only heuristically taken into account, in the sense that a cutting pattern is generated in the column generation approach only considering the dual variables from the master problem, i.e., there is no further concern regarding the scheduling while generating a cutting pattern. In fact, the total number of setups between two cutting patterns is only calculated after the generation of a cutting pattern. In the study presented here, the scheduling decisions are simultaneously taken into account with the bin-packing decisions in an exact and compact formulation, and the proposed solution methods are based on exact algorithm approaches using Benders decomposition in order to take advantage of the structure of the problem.

In a recent classification and literature review on integrated lot-sizing and cutting stock problems, Melega et al. (2018) also indicated that there is a lack of studies incorporating scheduling decisions in this integrated problem. In Melega et al. (2018), the papers are classified taking into account two types of integration: the integration across time periods and the integration between production levels. According to this classification, the problem in our study comprises 2 levels. The first one corresponds to level 2 in the proposed classification, consisting of the cutting process which in our case also includes the sequence-dependent scheduling decisions for the cutting operations. The other level corresponds to level 3 in the proposed classification and comprehends a lot-sizing problem with sequence-dependent setups. At each of the two levels, multiple time periods are taken into account. Even though a limit on the number of objects is needed in order to model the cutting decisions, there are no variables related to it, hence, level 1 of the classification related to the objects is not considered in our problem. The problem presented here can hence be classified as -/L2/L3/M according to the proposed classification notation. In contrast with the generalized mathematical model presented in Melega et al. (2018), the scheduling decisions are taken into account at both the cutting and final production level, with the consideration of sequence-dependent setup times and costs. Another difference between these two models lies in the variables used to model the cutting process. In the general model of Melega et al. (2018), these variables represent cutting patterns, whereas in the current paper we use assignment variables, which are needed to model the sequence-dependency in the cutting operations.

Much more attention has been paid to modeling scheduling decisions in the separate LS and BP problems. To deal with the scheduling decisions in the lot-sizing problem, several mathematical models and solution approaches have been proposed in the literature for a vast number of applications, such as pharmaceuticals (Vickery and Markland, 1986), tires (Jans and Degraeve, 2004), foundry (de Araujo and Clark, 2013; Hans and van de Velde, 2011), animal food (Toso et al., 2009), dairy and yogurt (Marinelli et al., 2007), beverage (Baldo et al., 2014; Toscano et al., 2020), and pulp packaging (Martínez et al., 2016). Some review papers that address scheduling decisions in lot-sizing problems are the studies of Drexl and Kimms (1997), Guimarães et al. (2014) and Copil et al. (2017).

Scheduling decisions in the bin-packing and cutting stock problem constitute an essential aspect in many practical applications when other factors, rather than the number of objects or waste, are important. For example, the number of open stacks (Yuen, 1991, 1995; Armbruster, 2002; Yanasse and Lamosa, 2007; Rinaldi and Franz, 2007; Aloisio et al., 2011; Matsumoto et al., 2011; Arbib et al., 2012, 2016), the spread of an order (Madsen, 1988; Foerster and Wäscher, 1998) and the order due dates (Li, 1996; Giannelos and Georgiadis, 2001; Johnston and Sadinlija, 2004; Reinertsen and Vossen, 2010; Bennell et al., 2013; Arbib and Marinelli, 2014; Braga et al., 2015, 2016). Diverse solution approaches have been used to solve these problems, such as greedy/rounding/LP-based heuristics, branch-and-bound, tabu search, generalized local search, implicit exhaustive search, implicit enumeration procedure, simulated annealing, genetic algorithm, Lagrangian relaxation, and general-purpose optimization packages.

Dealing with the change of knives, Wuttke and Heese (2018) present a study motivated by a textile industry, which consists of a two-dimensional cutting stock problem with sequence-dependent setup times. The problem requires finding a sequence to a selected set of cutting patterns, as well as the positions of knives, so as to minimize the required number of knife movements. A main methodological difference is that Wuttke and Heese (2018) use a limited set of precalculated cutting patterns, whereas we consider all possible cutting decisions since we use assignment variables. The two problems also differ with respect to the way the knife changes are counted.

3 Mixed integer mathematical models

In this study, the coupled bin-packing and lot-sizing problem is treated in a scheduling perspective, in which the scheduling horizon consists of a week and the time periods can be seen as days or work shifts within the week. Items are cut from objects, and these items need to undergo some additional processing before they are ready as products to meet the customers' demand. The demand for products should be met at the end of the scheduling horizon by production in any time period. Inventory is carried over to the end of the time horizon, when the demand needs to be satisfied. The synchronization between the bin-packing and lot-sizing decisions is done considering positive lead-times, i.e., an item that is cut in a specific time period only becomes available for further processing in the next time period. For simplicity and without loss of generality, we consider a lead-time of one time period. In addition, there is an initial inventory of items available in the first time period. At the end of the scheduling horizon, a prespecified amount of items needs to be available for the next scheduling horizon. Since the integrated problem treats short-term decisions, no backlogging is allowed and no holding costs are incurred for the inventory of items or products held between time periods.

The scheduling decisions with sequence-dependent setups in the bin-packing problem deal with the sequence of the cutting operations, whereas the lot-sizing problem manages the sequence of a process (drilling, folding, or assembling) needed to transform items into products. The cutting process considers the number of setups related to the change in the number of items cut from two subsequent objects. In this application, this is directly related to the operation of removing or inserting knives in different positions of the cutting machine. The production time and cost of cutting an object are independent of the number of items cut. This is because of the type of application where all the items of one object are cut simultaneously (as in the paper industry). The bin-packing decisions are modeled using a compact formulation, in which the cutting decisions describe how to cut each object in stock via assignment variables (Kantorovich, 1960). From a mathematical point of view, it does not matter whether the assignment for a given set of items and objects is interpreted as a cutting or a packing assignment. The two mathematical models proposed in this study differ from each other with respect to the strategy used to model the scheduling decisions in the bin-packing problem. In the first model we use known constraints from the literature with innovative aspects and in the second model, we use the idea of micro-periods and a phantom cutting process. Due to the dependency of the decisions in the bin-packing problem with sequence-dependent setups, the formulations are mixed-integer nonlinear mathematical models. For simplicity, we consider a one-dimensional bin-packing problem.

In order to define the integrated problem, we present the following sets, parameters, and variables that are common for both models:

Sets	
$T = \{1, \ldots, \overline{T}\}$	set of time periods (index t);
Bin-packing and scheduling sets	
$I = \{1, \ldots, \overline{I}\}$	set of different types (sizes) of items to be cut from objects (index i);
Lot-sizing and scheduling sets	
$F = \{1, \ldots, \overline{F}\}$	set of different demanded products (indexes f, q, g);
Bin-packing and scheduling parameters	
L	length of object;
l_i	length of item i;
sc	changeover cost for each insertion/removal of one knife in the cutting machine;
vc	cost of an object;
st	changeover time for each insertion/removal of one knife in the cutting machine;
vt	production time of cutting an object (independent of the number of items cut);
SI_i	initial inventory of items i;
r_{if}	number of items i needed to make product f;
$CapC_t$	cutting capacity (in time units) available in time period t;
Lot-sizing and scheduling parameters	
sc_{fq}	changeover cost from product f to product q;
st_{fq}	changeover time from product f to product q;
vt_f	unit production time of product f;
D_f	demand of product f (defined by the master production plan);
$CapF_t$	production capacity (in time units) available in time period t;
Lot-sizing and scheduling decision variables	
S_{ft}	inventory of product f at the end of time period t;
X_{ft}	production quantity of product f in time period t;
Y_{fqt}	variable indicating if there is a changeover from product f to product q in time period t;
$\overline{Y}_{ft}$	binary variable indicating if there is a setup state for product f at the beginning of time period t;

3.1 Model 1

In the first mathematical model proposed for the coupled bin-packing and lot-sizing problem with sequence-dependent setups, the scheduling decisions for both cutting and production process are modeled by the *ATSP* (*Asymmetric Traveling Salesman Problem*) constraints. In each time period several objects can be cut, as well as several types of products can be produced. Model 1 aims to find a single sequence for each level, i.e. a cutting sequence for the objects and a production sequence for the products, for the whole scheduling horizon. For this, the setup state in the cutting process and in the production of products is carried over the time periods, i.e., at the end of each time period the setup is saved and can be used at the beginning of the next time period. Hence, setup carryover is addressed in Model 1. Note that in this model when we refer to the scheduling of objects, this is equivalent to the scheduling of the cutting operations used in that object, since the objects are of the same type.

The definition of the specific sets and decision variables for Model 1 are as follows:

Bin-packing and scheduling sets	
$O = \{1, \ldots, \overline{O}\}$	set of available objects (of equal sizes) to be cut (indexes o, k, v);
Bin-packing scheduling decision variables	
a_{iot}	number of items i cut from object o in time period t;
C_{ot}	binary variable indicating if the object o is cut in time period t;
S_{it}	inventory of items i at the end of time period t;
na_{ot}	total number of items cut from object o in time period t;
TW_{okt}	total number of setups (knife insertion/removal) necessary for a changeover from object o to object k in time period t;
W_{okt}	binary variable indicating if there is a changeover from object o to object k in time period t;
$\overline{W}_{ot}$	binary variable indicating the setup state for object o at the beginning of time period t;

3.1.1 Objective function

The objective function (1) minimizes the costs of the coupled bin-packing and lot-sizing problem. The first term is related to the bin-packing decisions, which consists of the cost for the material of the objects used. The last two terms correspond to the sequence-dependent setup costs of a changeover from one cut object to another and from one produced product to another, respectively. The production costs of products are not considered in the objective function, as they consist of a fixed value related to the demand of the products, which is known at the beginning of the scheduling horizon. The inventory costs associated with items and products are not taken into account since the integrated problem treats short-term decisions.

$$min \sum_{t \in T} \left(\sum_{o \in O} vc\ C_{ot} + \sum_{o \in O} \sum_{k \in O} sc\ TW_{okt} + \sum_{f \in F} \sum_{q \in F} sc_{fq} Y_{fqt} \right) \tag{1}$$

3.1.2 Bin-packing and scheduling decisions

Constraints (2)–(19) are related to the bin-packing and scheduling decisions with sequence-dependent setups in the integrated problem. Constraints (2) are the capacity constraints of the cutting process (one machine) and take into account the time used for cutting objects, as well as the time consumed for setting up the machine for a different knife configuration. The setup requires a fixed time (st) for each insertion or removal of knives required to cut object k, considering the knife configuration when cutting object o (TW_{okt}), i.e., sequence-dependent setup times.

$$\sum_{o \in O} \left(vt\ C_{ot} + \sum_{k \in O} st\ TW_{okt} \right) \leq CapC_t \qquad \forall t \tag{2}$$

Constraints (3) couple the bin-packing and scheduling decisions by imposing that an object o can be cut in a time period t only if the cutting machine has been set up accordingly at the beginning of time period t ($\overline{W}_{ot} = 1$) or there is a changeover to cut this object in time period t. Constraints (4) ensure that the cutting machine is set up to cut an object at the beginning of each time period, and constraints (5) prohibit a changeover to cut the same object in a time period.

$$C_{ot} \leq \left(\overline{W}_{ot} + \sum_{k \in O} W_{kot} \right) \qquad \forall o, \forall t \tag{3}$$

$$\sum_{o \in O} \overline{W}_{ot} = 1 \qquad \forall t \tag{4}$$

$$W_{oot} = 0 \qquad \forall o, \forall t \tag{5}$$

Constraints (6)–(11) are the *ATSP* constraints handling the scheduling of objects in the cutting process for the whole scheduling horizon and consider setup carryover. Constraints (6) allow that an object enters the sequence only if there is a changeover to cut another object afterward or this object is the first one to be cut in the next time period. In constraints (7), the idea is similar to the previous constraints, in which an object enters the sequence only if there is a changeover to cut this object or it is the first one to be cut in the current time period. Constraints (8) prohibits a changeover to cut an object for which the cutting machine has already been set up at the beginning of the current time period. Constraints (9) avoid a changeover from an object that will be setup at the beginning of the next time period. Constraints (10) ensure that if an object has been set up at the beginning of a time period, then there is a changeover from that object to another one in the same time period or this setup is carried over to the beginning of the next period. In fact, constraints (9) and (10) guarantee that in time periods with no cutting process, the setup state can be preserved over idle time periods to the next time period with cutting. As known in the literature, mathematical models based on the *ATSP* constraints require subtour elimination constraints (11) in order to avoid subtours in the scheduling of the operations. An example for scheduling of objects using the *ATSP* constraints can be seen in Figure 2. Note that, in the scheduling of objects, object k has been scheduled as last object in time period t and due to setup carryover, object k is scheduled as the first object in period $t+1$. Therefore, object k can be cut either in period t or period $t+1$, depending on the available capacity and the costs involved during this process. The strategies employed to manage constraints (11) during the branch-and-bound algorithm are described in more detail in Section 4.

$$\sum_{k \in O} W_{kot} \leq \sum_{v \in O} W_{ovt} + \overline{W}_{o,t+1} \qquad \forall o, t = 1, \ldots, \overline{T}-1 \tag{6}$$

$$\sum_{k \in O} W_{okt} \leq \sum_{v \in O} W_{vot} + \overline{W}_{ot} \qquad \forall o, \forall t \tag{7}$$

$$\sum_{k \in O} W_{kot} \leq 1 - \overline{W}_{ot} \qquad \forall o, \forall t \tag{8}$$

$$\sum_{k \in O} W_{okt} \leq 1 - \overline{W}_{o,t+1} \qquad \forall o, t = 1, \ldots, \overline{T}-1 \tag{9}$$

$$\overline{W}_{ot} \leq \sum_{k \in O} W_{okt} + \overline{W}_{o,t+1} \qquad \forall o, t = 1, \ldots, \overline{T}-1 \tag{10}$$

$$\text{Subtour Elimination Constraints} \tag{11}$$

Constraints (12) guarantee the physical limitations of the cutting process in terms of object length, i.e., if an object o is cut in a period t ($C_{ot} = 1$), the combination of the lengths of items i that will be cut from it, should not exceed its length. Constraints (13) ensure that each object is available to be cut at most once during the whole scheduling horizon.

$$\sum_{i \in I} l_i a_{iot} \leq L \, C_{ot} \qquad \forall o, \forall t \tag{12}$$

$$\sum_{t \in T} C_{ot} \leq 1 \qquad \forall o \tag{13}$$

The symmetry-breaking constraints (14) and (15) are related to the assignment of objects in the cutting process and eliminate alternative solutions that can be obtained by renumbering the identical objects. Constraints (14) impose that if object $o+1$ is used in period t, then object o must have been used in period t or a previous period. Constraint (15) imposes that the total number of objects cut in the whole scheduling horizon is not smaller than an integer lower bound which can be calculated as the minimum number of objects needed to satisfy the demand for the final products assuming no waste during the cutting process. It is important to mention that these symmetry-breaking constraints are valid in this problem due to the assumption that all the objects considered in the cutting process are equal.

$$\sum_{\tau=1}^{t} C_{o\tau} \geq C_{o+1,t} \qquad o = 1, \ldots, \overline{O} - 1, \forall t \tag{14}$$

$$\sum_{t \in T} \sum_{o \in O} C_{ot} \geq O_{min} \tag{15}$$

Constraints (16)–(18) model the counting of setups (i.e., knife movements) needed in the cutting process for a changeover from object o to object k. These constraints will lead to the non-linearity in the formulation. The non-linearities arise as absolute values (constraints (16) and (17)) and the product of variables (constraints (18)). The details about the linearization of these constraints are presented in A. The setup counting is split into two mutually exclusive sets of constraints (16) and (17). Constraints (16) is activated when there is a changeover among two cut objects and it occurs inside of a time period, according to the setup variables W_{okt}, whereas constraints (17) consider the case when the object is already setup at the beginning of a time period by the setup carryover variables $\overline{W}_{ot}$. The number of setups is counted as the number of knife movements when changing from object o to object k. This includes both knife insertions and knife removals and it is counted as the absolute difference in the number of items cut from object o and k. As the setup variables (TW_{okt}) are being optimized in the objective function, they assume the smallest values in constraints (16) and (17).

We will explain these constraints in more detail (see Figure 2). Considering a changeover from object o to object k in time period t, i.e., $W_{okt} = 1$ and necessarily $\overline{W}_{kt} = 0$. In this case, constraints (17) are redundant and constraints (16) are active so that the setup (TW_{okt}) counts the difference (in absolute value) between the total number of items cut from object k in time period t (na_{kt}) and the total number of items cut from object o in time period t (na_{ot}). Next, we consider the case where the object k is setup at the beginning of time period $t+1$, $\overline{W}_{k,t+1} = 1$ and hence, $W_{ok,t+1} = 0$, $\forall o$. In this case, constraints (16) are redundant, whereas (17) are active. The setup ($TW_{kk,t+1}$) counts for the difference (in absolute value) between the total number of items cut from object k in time period $t+1$ ($na_{k,t+1}$) and the total number of items cut from object k in time period t (na_{kt}). As there is setup carry over and each object is cut only once during the scheduling horizon, constraints (17) counts for the setup between the first cut object in time period $t+1$ and the last cut object in time period t. This will become clear when we next explain constraint (18).

$$TW_{okt} \geq |na_{kt} - na_{ot}| - M(1 - W_{okt}) \qquad \forall o, k, k \neq o, \forall t \tag{16}$$

$$TW_{kk,t+1} \geq |na_{k,t+1} - na_{kt}| - M(1 - \overline{W}_{k,t+1}) \qquad \forall k, t = 1, \ldots, \overline{T} - 1 \tag{17}$$

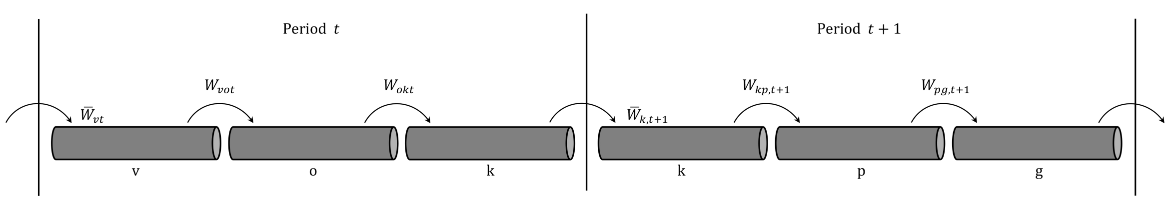

Figure 2: A scheduling for the objects during the cutting process

It is worth mentioning that depending on the practical application, the setup counting of the cutting machine in the cutting process may vary from the one addressed in Model 1. For this, we also present two different ways of setup counting, which depend on the number of knives needed to perform a cutting operation and the type of operation needed to setup the machine (see Appendix B).

Constraints (18) are the expressions used to count the total number of items cut from an object k in time period t (na_{kt}) and can be discretized into three cases:

(i) when the object k is cut in period t ($C_{kt} = 1$), then the total number of cut items consists of the total number of items cut from object k, i.e., $na_{kt} = \sum_{i \in I} a_{ikt}$ and all other terms in constraints (18) are zero;

(ii) when the object k is not cut in period t ($C_{kt} = 0$) and there is a changeover from object o to object k ($W_{okt} = 1$) (see Figure 3), then, the first and last terms of Constraints (18) are zero. This occurs because no items are cut from object k ($\sum_{i \in I} a_{ikt} = 0$) and $\overline{W}_{kt} = 0$, since $W_{okt} = 1$. In this case, constraints (18) are able to load the information of the number of cut items from previous cut object o to the current object k, that is, $na_{kt} = \sum_{i \in I} a_{iot} = na_{ot}$, hence, there is no setup from objects o to object k ($TW_{okt} \geq 0$), and the counting of setups is kept correctly to the whole scheduling horizon. The counting of setups between two cut objects will occur in the next period $t+1$, when object k that has been setup in period t will be cut;

(iii) when the setup carryover occurs for the object k ($\overline{W}_{kt} = 1$), from time period $t-1$ to time period t, and this object is cut in period t ($C_{kt} = 1$), the counting occurs as case (ii). However, when the setup carryover occurs and the object is not cut in the current time period t, this means that the object has been cut in the previous time period $t-1$ (see Figure 4), and the first and second terms of constraints (18) are zero, since no items are cut from object k in the current time period ($\sum_{i \in I} a_{ikt} = 0$) and $W_{okt} = 0, \forall o$ as $\overline{W}_{kt} = 1$. In this case, constraints (18) are able to load the information from the previous time period $t-1$, where the object has been cut, to the current time period t, i.e., $na_{kt} = na_{k,t-1}$, hence, there is no setup ($TW_{kkt} \geq 0$), and this keeps the counting of setups correctly to the whole scheduling horizon.

$$na_{kt} = \sum_{i \in I} a_{ikt} + \sum_{o \in O} \sum_{i \in I} a_{iot} W_{okt}(1 - C_{kt}) + na_{k,t-1} \overline{W}_{kt}(1 - C_{kt}) \qquad \forall k,\ \forall t \tag{18}$$

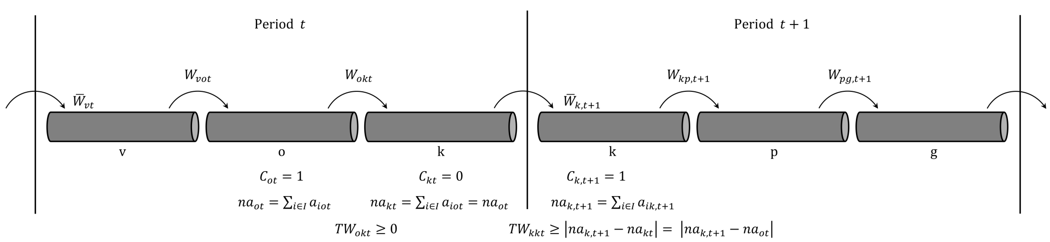

Figure 3: Exemplifying constraints (18) case (ii)

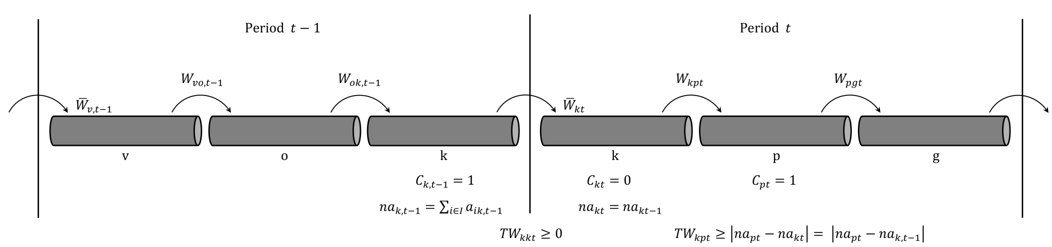

Figure 4: Exemplifying constraints (18) case (iii)

Finally, constraints (19) are the non-negativity and integrality constraints of the variables in the bin-packing and scheduling problem.

$$a_{iot} \in \mathbb{Z}_+,\ C_{ot},\ \overline{W}_{ot},\ W_{okt} \in \{0, 1\},\ na_{kt} \geq 0 \qquad \forall i,\ \forall o,\ k,\ \forall t \tag{19}$$

3.1.3 Linking decisions

Constraints (20) link the decisions between the bin-packing and the lot-sizing problems. These constraints model the interdependency of the problems and ensure that the dependent demand balance constraints of items are satisfied, considering a lead-time of one time period for the cut items to be available, i.e., items cut in a time period t, or in previous time periods, are available to be processed into products which are required in the time period $t+1$. Due to the positive lead-times, constraints (21) allow that products are processed in period 1 using the initial inventory of cut items. We assume that this initial inventory cannot be carried over to the next period, however, it would be straightforward to incorporate this. Constraints (22) impose a predefined extra amount of cut items in the last time period, that will be available in the next scheduling horizon.

$$S_{i,t-1} + \sum_{o \in O} a_{iot} - S_{it} = \sum_{f \in F} r_{if} X_{f,t+1} \qquad \forall i, \forall t \in T \backslash \overline{T} \tag{20}$$

$$\sum_{f \in F} r_{if} X_{f1} \leq SI_i \qquad \forall i \tag{21}$$

$$S_{i,\overline{T}-1} + \sum_{o \in O} a_{io\overline{T}} \geq SI_i \qquad \forall i \tag{22}$$

3.1.4 Lot-sizing and scheduling decisions

Constraints (23)–(35) are related to the lot-sizing and scheduling decisions for the products, with sequence-dependent setups, in the integrated problem. Constraints (23) and (24) define the demand balance constraints of products and consider that the demand for products should be met at the end of the scheduling horizon by production in any time period, which is carried over the time periods using inventory. The capacity constraints (25) take into account the production and the sequence-dependent setup times associated with the production of the products in each time period.

$$S_{f,t-1} + X_{ft} = S_{ft} \qquad \forall f,\ \forall t \in T \backslash \{\overline{T}\} \tag{23}$$

$$S_{f,\overline{T}-1} + X_{ft} - S_{f\overline{T}} = D_f \qquad \forall f \tag{24}$$

$$\sum_{f \in F} \left(vt_f X_{ft} + \sum_{q \in F} st_{fq} Y_{fqt} \right) \leq CapF_t \qquad \forall t \tag{25}$$

Constraints (26) couple the lot-sizing and the scheduling decisions considering setup carryover. Constraints (27)–(34) guarantee the scheduling of the operations needed to obtain the products and have the same meaning as those for the scheduling of objects in the cutting process (constraints (4)–(11)).

$$vt_f X_{ft} \leq CapF_t \left(\overline{Y}_{ft} + \sum_{q \in F} Y_{qft} \right) \qquad \forall f, \forall t \tag{26}$$

$$\sum_{f \in F} \overline{Y}_{ft} = 1 \qquad \forall t \tag{27}$$

$$Y_{fft} = 0 \qquad \forall f, \forall t \tag{28}$$

$$\sum_{q \in F} Y_{qft} \leq \sum_{g \in F} Y_{fgt} + \overline{Y}_{f,t+1} \qquad \forall f, t = 1, \ldots, \overline{T}-1 \tag{29}$$

$$\sum_{q \in F} Y_{fqt} \leq \sum_{g \in F} Y_{gft} + \overline{Y}_{ft} \qquad \forall f, \forall t \tag{30}$$

$$\sum_{q \in F} Y_{qft} \leq 1 - \overline{Y}_{ft} \qquad \forall f, \forall t \tag{31}$$

$$\sum_{q \in F} Y_{fqt} \leq 1 - \overline{Y}_{f,t+1} \qquad \forall f, t = 1, \ldots, \overline{T}-1 \tag{32}$$

$$\overline{Y}_{ft} \leq \sum_{q \in F} Y_{fqt} + \overline{Y}_{f,t+1} \qquad \forall f, t = 1, \ldots, \overline{T}-1 \tag{33}$$

$$\text{Subtour Elimination Constraints} \tag{34}$$

Finally, constraints (35) are the non-negativity and integrality constraints of the variables in the lot-sizing and scheduling problem.

$$X_{ft},\ S_{ft} \geq 0,\ \overline{Y}_{ft},\ Y_{fqt} \in \{0,1\} \qquad \forall f, q, \forall t \tag{35}$$

3.2 Model 2

The second mathematical model presented in this paper for the coupled bin-packing and lot-sizing problem is also based on the *ATSP* constraints to model the scheduling decisions in the lot-sizing problem. However, for the bin-packing problem, the scheduling decisions are based on the idea of micro-periods (Meyr, 2000) and a phantom cutting process. In this approach, each time period (macro-period) is split into a fixed number of non-overlapping subperiods (micro-periods) and these micro-periods are numbered in a consecutive way over the whole scheduling horizon. The length of each micro-period is flexible, in the sense that there is no capacity limitation regarding a micro-period, only in the macro-period, and it takes into account the setup time and production time of the cutting operations completed in that micro-period. More specifically, only one setup is allowed in each micro-period, i.e., at most one object can be cut in each micro-period. The counting of setups is performed for each micro-period considering the setup state of the previous micro-period, even if no cutting occurs. When no cutting process takes place, the idea of the phantom cutting process is used, which consists of performing cuts in such a way that there is no influence in the counting of setups, i.e., the phantom cutting process has the same number of items as the last cutting process. Its only responsibility is to carry the information regarding the number of cut items between micro-periods, until the next micro-period with a real cutting operation, in order to correctly count the setups over the whole scheduling horizon. As such, it does not influence the demand for items, setup costs, capacity consumption, or the final solution. Therefore, the micro-period decisions can be seen as devices to model the changeovers from one micro-period to another in the cutting process of objects, i.e., they determine the sequence in which the cutting operations are processed in each time period of the scheduling horizon.

In the definition of the decision variables to Model 2, the lot-sizing and scheduling decisions are the same as in Model 1, whereas the bin-packing and scheduling decisions are presented using the sets and variables, as follows:

Sets	
$U_t = \{bu_t, \ldots, eu_t\}$	set of micro-periods in time period t for the bin-packing problem ($bu_t = 1 + eu_{t-1}$);
Bin-packing scheduling decision variables	
S_{it}	inventory of items i at the end of time period t;
C_u	binary variable indicating if there is a cutting process in micro-period u;
a_{iu}	number of items i cut in micro-period u during a cutting process;
N_u	binary variable indicating if there is a phantom cutting process in micro-period u;
b_{iu}	number of items i cut in micro-period u during a phantom cutting process;
TW_u	total number of setups (knife insertion/removal) necessary for a changeover to micro-period u;

3.2.1 Objective function

The objective function (36), as in the previous model, minimizes the cost for the material of the objects used and the sequence-dependent setup costs of a changeover from one cutting operation to another and from one product to another, respectively. Note that for the cutting process, the setup decisions are considered for all micro-period of each time period.

$$min \sum_{t\in T} \left(\sum_{u\in U_t} (vc\ C_u + sc\ TW_u) + \sum_{f,q\in F} sc_{fq} Y_{fqt} \right) \tag{36}$$

3.2.2 Bin-packing and scheduling decisions

Constraints (37)–(46) are related to the bin-packing and scheduling decisions in the integrated problem. Constraints (37) are the capacity constraints of the cutting process (one machine) and consider, for all micro-periods of each time period, the time used for cutting objects (vt), as well as the time consumed for setting up the machine with a different knife configuration (st). The variable TW_u counts the total number of knife insertions or removals necessary to change the knife configuration between micro-period $u-1$ and u. As such, we account for the sequence-dependent setup time.

$$\sum_{u \in U_t} (vt\ C_u + st\ TW_u) \leq CapC_t \qquad \forall t \tag{37}$$

Constraints (38)–(40) link the cutting process and the phantom cutting process in the bin-packing problem, and model the physical limitations of these processes in each micro-period. Constraints (38) force that in each micro-period either the cutting process or the phantom cutting process takes place. If there is a cutting process in micro-period u, ($C_u = 1$ and $N_u = 0$), then due to constraints (39), the combination of the lengths of items that will be cut in that micro-period should not exceed the object length and by constraints (40), there are no items cut from the phantom cutting process. However, if no cutting occurs ($C_u = 0$), there is the execution of a phantom cutting process ($N_u = 1$) and constraints (40) guarantee the physical limitations of the phantom cutting process in the corresponding micro-period u.

$$C_u + N_u = 1 \qquad \forall t, u \in U_t \tag{38}$$

$$\sum_{i \in I} l_i a_{iu} \leq L\ C_u \qquad \forall t, u \in U_t \tag{39}$$

$$\sum_{i \in I} l_i b_{iu} \leq L\ N_u \qquad \forall t, u \in U_t \tag{40}$$

The symmetry-breaking constraints (41) and (42) are related to the assignment of the cutting process and the phantom cutting process to micro-periods. Constraints (41) force that the micro-periods with the cutting process occur at the beginning of each macro-period, whereas constraints (42) allocate the phantom cutting process, i.e., micro-periods without production, at the end of macro-periods. Constraints (43) restrict that, once the phantom cutting process is assigned to a micro-period, the number of times each item appear in it should be at least the same as in the previous phantom cutting process. These constraints will limit the generation of alternative symmetrical solutions. Constraints (44) impose that the total number of micro periods with cutting operations for the whole scheduling horizon is not smaller than an integer lower bound for the related problem, which is taken as the minimum number of objects needed to cut all the items, assuming no waste during the cutting process.

$$C_{u+1} \leq C_u \qquad \forall t, u = bu_t, \ldots, eu_t - 1 \tag{41}$$

$$N_u \leq N_{u+1} \qquad \forall t, u = bu_t, \ldots, eu_t - 1 \tag{42}$$

$$b_{iu} \leq b_{i,u+1} \qquad \forall i, \forall t, u = bu_t, \ldots, eu_t - 1 \tag{43}$$

$$\sum_{t \in T} \sum_{u \in U_t} C_u \geq O_{min} \tag{44}$$

Constraints (45) correspond to the non-linear set of constraints in Model 2 and are responsible for counting the setups needed in the cutting process for a changeover from micro-period $u-1$ to micro-period u (TW_u). The nonlinearity arises as an absolute value of the difference in the total number of items cut either in a cutting process or in a phantom cutting process, from micro-period $u-1$ to micro-period u. The details about the linearization of these constraints are presented in Appendix A. Considering a cutting process in micro-period $u-1$, then according to previous constraints, $\sum_{i \in I} b_{i,u-1} = 0$ and the total number of items cut in micro-period $u-1$ is $\sum_{i \in I} a_{i,u-1}$. In the next micro-period u, two cases can occur:

(i) there is a cutting process in micro-period u. According to previous constraints, no phantom cutting process occurs ($\sum_{i \in I} b_{iu} = 0$), and the total number of items cut in micro period u is

$\sum_{i \in I} a_{iu}$. Then, the total number of setups is calculated as the difference in the total number of items cut in a changeover from micro-period $u-1$ to micro-period u, i.e., the setup decision variables are given by $\left|\sum_{i \in I} a_{iu} - \sum_{i \in I} a_{i,u-1}\right|$;

(ii) there is no cutting process in micro-period u, then $\sum_{i \in I} a_{iu} = 0$, and the phantom cutting process takes place, with $\sum_{i \in I} b_{iu}$ being the total number of items cut in micro-period u. Then the total number of setups is given by the difference in the total number of items cut from a cutting process and a phantom cutting process, i.e., the setup decision variables assume the value of $\left|\sum_{i \in I} b_{iu} - \sum_{i \in I} a_{i,u-1}\right|$. As the setup decision variables (TW_u) are minimized in objective function (36), the phantom decision variables assume values in order to not influence in the counting of setups, i.e., $\sum_{i \in I} b_{iu} = \sum_{i \in I} a_{i,u-1}$ and $TW_u \geq 0$.

In case a phantom cutting process occurs in micro-period $u-1$, a similar logic can be followed. Considering the setup decision variables (TW_u) are minimized in the objective function, they assume the smallest value in constraints (45). Therefore, the phantom decision variables can be seen as devices that load the information of the total number of items cut and carry this information among micro-periods until the next micro-period with a real cutting process.

$$TW_u \geq \left| \sum_{i \in I} (a_{iu} + b_{iu}) - \sum_{i \in I} (a_{i,u-1} + b_{iu-1}) \right| \qquad \forall t, u \in U_t,\ u > 1 \tag{45}$$

Finally, constraints (46) define the domain of the bin-packing and scheduling variables to Model 2.

$$a_{iu},\ b_{iu} \in \mathbb{Z}_+,\ C_u,\ N_u \in \{0,1\}, TW_u \geq 0 \qquad \forall i, \forall t, u \in U_t \tag{46}$$

3.2.3 Linking decisions

Constraints (47)–(49) are the linking constraints between the bin-packing and lot-sizing problems, i.e., these constraints model the interdependency among the decisions of the problems and have the same meaning as constraints (20)–(22) in Model 1. The only difference is that in Model 2, the cutting of items is considered over all micro-periods belonging to the corresponding time period t.

$$S_{i,t-1} + \sum_{u \in U_t} a_{iu} - S_{it} = \sum_{f \in F} r_{if} X_{f,t+1} \qquad \forall i, \forall t \in T \backslash \overline{T} \tag{47}$$

$$\sum_{f \in F} r_{if} X_{f1} \geq SI_i \qquad \forall i \tag{48}$$

$$S_{i,\overline{T}-1} + \sum_{u \in U_{\overline{T}}} a_{iu} \geq SI_i \qquad \forall i \tag{49}$$

3.2.4 Lot-sizing and scheduling decisions

The lot-sizing and scheduling decisions in Model 2, can be described by the same constraints (23)–(19) from Model 1.

4 Solution methods

In this section, we present the solution approaches to solve the coupled bin-packing and lot-sizing problem with sequence-dependent setups. Firstly, we apply linearization techniques to the models, in order to obtain exact mixed-integer linear formulations. For more details about the linearization, we refer to Appendix A. After this, strategies are employed to deal with possible subtours present in a solution of the integrated problem. Subtours might occur due to the *ATSP* constraints used to model the scheduling problem, in which a solution using just such constraints might contain a subtour and hence it is not feasible to the integrated problem. Thus, to prevent subtours we employ two types of subtour elimination constraints, one consisting of a polynomial-sized set of constraints and the other of a non-polynomial-sized set. The subtour elimination constraints are inserted in Model 1 for objects and products, whereas in Model 2 they are inserted only for products since this model uses a different approach to model the scheduling decisions in the bin-packing problem, which do not need such constraints (micro-periods and the phantom cutting process).

Considering the mixed-integer linear mathematical models, with the subtour elimination constraints, a general-purpose optimization package is used to solve the models with the polynomial-sized set of constraints. The models with the non-polynomial-sized set of subtour elimination constraints are solved by an exact solution approach based on the branch-and-cut algorithm. We also apply a Benders-based branch-and-cut algorithm in an attempt to improve the solution and obtain better results to the the coupled bin-packing and lot-sizing problem. The details regarding each one of the methodologies are presented in the following sections.

4.1 Subtour elimination constraints

The first type of subtour elimination constraints consists of a polynomial-sized set of constraints that is included a priori in the mathematical models for each pair of objects and products, in each time period. These constraints consist of several linear systems, in which the solution assigns a position in the sequence to each object and product for the whole scheduling horizon. We analyze three variations of these subtour elimination constraints, as follows:

MTZ1: these subtour elimination constraints are based on the constraints presented in Miller et al. (1960) and are added to objects and products, respectively:

$$V_{kt} \geq V_{ot} + 1 - \overline{O}(1 - W_{okt}) \qquad \forall o, k, o \neq k, \forall t \tag{50}$$
$$0 \leq V_{ot} \leq \overline{O} \qquad \forall o, \forall t \tag{51}$$
$$V_{qt} \geq V_{ft} + 1 - \overline{F}(1 - Y_{fqt}) \qquad \forall f, q, f \neq q, \forall t \tag{52}$$
$$0 \leq V_{ft} \leq \overline{F} \qquad \forall f, \forall t \tag{53}$$

where V_{ot} indicates the order in which object o is cut in period t (it is responsible for scheduling the cutting of objects) and V_{ft} indicates the order in which product f is produced in period t (it is responsible for scheduling the final products).

MTZ2: these subtour elimination constraints are an improved version of the previous constraints (MTZ) and are based on the studies presented in Desrochers and Laporte (1991). They are added to objects and products, respectively:

$$V_{ot} - V_{kt} + \overline{O}W_{okt} + (\overline{O} - 2)W_{kot} \leq \overline{O} - 1 \qquad \forall o, k, o \neq k, \forall t \tag{54}$$
$$0 \leq V_{ot} \leq \overline{O} \qquad \forall o, \forall t \tag{55}$$
$$V_{ft} - V_{qt} + \overline{F}Y_{fqt} + (\overline{F} - 2)Y_{qft} \leq \overline{F} - 1 \qquad \forall f, q, f \neq q, \forall t \tag{56}$$
$$0 \leq V_{ft} \leq \overline{F} \qquad \forall f, \forall t \tag{57}$$

MTZ3: these subtour elimination constraints are based on the improved *MTZ1* constraints, with the addition of Clique constraints to 3 nodes (Bektaş and Gouveia, 2014). They are added to objects and products, respectively:

$$\begin{aligned} V_{ot} - V_{vt} + \overline{O}W_{okt} + \overline{O}W_{kvt} + (\overline{O} - 2)W_{vkt} \\ +(\overline{O} - 2)W_{kot} + (\overline{O} + 1)W_{ovt} \\ +(\overline{O} - 3)W_{vot} \leq 2\overline{O} - 2 \end{aligned} \qquad \forall o, k, v, o \neq k \neq v, \forall t \tag{58}$$
$$0 \leq V_{ot} \leq \overline{O} \qquad \forall o, \forall t \tag{59}$$

$$
\begin{aligned}
V_{ft} - V_{gt} + \overline{F}Y_{fqt} + \overline{F}Y_{qgt} + (\overline{F}-2)Y_{gqt} \\
+(\overline{F}-2)Y_{qft} + (\overline{F}+1)Y_{fgt} \\
+(\overline{F}-3)Y_{gft} \leq 2\overline{F} - 2 \qquad & \forall f, q, g, f \neq q \neq g, \forall t \qquad (60) \\
0 \leq V_{ft} \leq \overline{F} \qquad & \forall f, \forall t \qquad (61)
\end{aligned}
$$

The other type of subtour elimination constraints consists of a non-polynomial-sized set, which is based on the DFJ constraints proposed in Dantzig et al. (1954) and adapted in Carpaneto et al. (1995). The DFJ constraints, constraints (62) and (63), are added to the models for objects and products, respectively, in each time period of the scheduling horizon:

$$\sum_{o,k \in STO} W_{okt} \leq |STO| - 1 \qquad \forall STO \subset O, 2 \leq |STO| \leq \overline{O} - 1,\ \forall t \qquad (62)$$

$$\sum_{f,q \in STF} Y_{fqt} \leq |STF| - 1 \qquad \forall STF \subset F, 2 \leq |STF| \leq \overline{F} - 1,\ \forall t \qquad (63)$$

where, STO and STF are subsets of O and F, respectively.

Although there is a large number of these DFJ subtour elimination constraints, in an optimal solution, the vast majority of them will not be binding. Therefore, imposing the subtour elimination constraints selectively, for just those subtours that occur in the scheduling of objects and products, is the approach addressed in this paper (Absi and Kedad-Sidhoum, 2008). For this, an exact branch-and-cut algorithm (B&C) is presented. The B&C applies violated subtour elimination cuts for every subtour found at the nodes of the branch-and-bound tree while solving a polynomial version of the problem. These subtours are separated by the B&C algorithm for each integer solution found at a node of the branch-and-bound tree and can be added in many different ways, among which we have tested the following:

- B&C1: for every time period it is checked if there is a subtour for objects/products and the respective subtour elimination constraint is inserted for the corresponding time period.
- B&C2: for each time period it is checked if there is a subtour for objects/products and the respective subtour elimination constraint is inserted for all time periods.
- B&C3: these subtour elimination constraint are slightly different from constraints (62) and (63) because they consider also the reverse changeover variable of the corresponding subtour (Clark et al., 2010). Then, for each time period it is checked if there is a subtour for objects/products and the respective subtour elimination constraint is inserted for the corresponding time period, using the following constraints:

$$\sum_{o,k \in STO} (W_{okt} + W_{kot}) \leq |STO| - 1 \qquad \forall STO \subset \overline{O},\ 3 \leq |STO| \leq \overline{O} - 1, \forall t \qquad (64)$$

$$\sum_{f,q \in STF} (Y_{fqt} + Y_{qft}) \leq |STF| - 1 \qquad \forall STF \subset \overline{F},\ 3 \leq |STF| \leq \overline{F} - 1, \forall t \qquad (65)$$

4.2 Benders reformulation

In this section, we describe a Benders decomposition approach applied to the coupled bin-packing and lot-sizing problem with sequence-dependent setups. In Benders decomposition (Benders, 1962), instead of solving the original complex mixed-integer linear problem, the problem is partitioned into a pure integer master problem and a linear subproblem. The master problem consists of a simplified version of the original problem, where only binary and integer variables are maintained, along with the constraints that are restricted to them. The master problem also contains an artificial variable representing a lower bound on the cost of the linear subproblem. The subproblem comprises the original problem, without those constraints kept in the master problem, and the variables from the master problem which are fixed to given values. These two problems are then solved iteratively, and the solution of one problem is given as input to the other until the optimal solution is achieved. Benders decomposition has been applied to different types of problems such as facility location (Fischetti et al., 2016) and stochastic optimization (Bodur et al., 2017). For a recent review on Benders decomposition, the reader is referred to Rahmaniani et al. (2017).

In this study, the Benders decomposition is presented only for Model 2. However, its application to Model 1 is very similar. In the Benders master problem, we have the scheduling and cutting decisions, which are the binary and integer variables of the integrated problem. When those variables are fixed, the remaining continuous linear problem, with production and inventory decision variables, comprises the Benders subproblem. Note that, even though the number of setups (TW_u) is integer, it can be defined as continuous and will automatically assume integer values in this formulation, hence, it is allocated in the Benders subproblem. Let $\widetilde{\overline{Y}}_{ft}$, $\widetilde{Y}_{fqt}$ be the fixed values for the scheduling decisions and $\widetilde{a}_{iu}$, $\widetilde{b}_{iu}$ $\widetilde{C}_u, \widetilde{N}_u$ be the fixed values for the cutting decisions. Therefore, the Benders subproblem consists of minimizing the total number of changeovers in the cutting process and determining the production and inventory of products, while respecting the available resources, which is defined by:

Benders subproblem to Model 2

$$\min \sum_{t \in T} \sum_{u \in U_t} sc\ TW_u \tag{66}$$

Subject to:

$$S_{f,t-1} + X_{ft} - S_{ft} = 0 \qquad \forall f, \forall t \in T \backslash \{\overline{T}\} \tag{67}$$

$$S_{f,\overline{T}-1} + X_{ft} - S_{f\overline{T}} = D_f \qquad \forall f \tag{68}$$

$$\sum_{f \in F} vt_f X_{ft} \leq CapF_t - \sum_{f,q \in F} st_{fq} \widetilde{Y}_{fqt} \qquad \forall t \tag{69}$$

$$vt_f X_{ft} \leq CapF_t \left(\widetilde{\overline{Y}}_{ft} + \sum_{q \in F} \widetilde{Y}_{qft} \right) \qquad \forall f, \forall t \tag{70}$$

$$S_{i,t-1} - S_{it} - \sum_{f \in F} r_{if} X_{f,t+1} = -\sum_{u \in U_t} \widetilde{a}_{iu} \qquad \forall i, \forall t \in T \backslash \overline{T} \tag{71}$$

$$-S_{i,\overline{T}-1} \leq \sum_{u \in U_{\overline{T}}} \widetilde{a}_{iu} - SI_i \qquad \forall i \tag{72}$$

$$\sum_{f \in F} r_{if} X_{ft} \leq SI_i \qquad \forall i \tag{73}$$

$$\sum_{u \in U_t} st\ TW_u \leq CapC_t - vt\ \widetilde{C}_u \qquad \forall t \tag{74}$$

$$-TW_u \leq -\sum_{i \in I} (\widetilde{a}_{iu} + \widetilde{b}_{iu}) + \sum_{i \in I} (\widetilde{a}_{i,u-1} + \widetilde{b}_{i,u-1}) \qquad \forall t, u \in U_t, u > 1 \tag{75}$$

$$-TW_u \leq \sum_{i \in I} (\widetilde{a}_{iu} + \widetilde{b}_{iu}) - \sum_{i \in I} (\widetilde{a}_{i,u-1} + \widetilde{b}_{i,u-1}) \qquad \forall t, u \in U_t, u > 1 \tag{76}$$

$$X_{ft},\ S_{ft},\ TW_u \geq 0, \qquad \forall f, q, \forall t, u \in U_t \tag{77}$$

Let $\alpha = \{\alpha_{ft}; f \in F, t \in T\}$, $\beta = \{\beta_t \leq 0; t \in T\}$, $\gamma = \{\gamma_{ft} \leq 0; f \in F, t \in T\}$, $\delta = \{\delta_{it}; i \in I, t \in T\}$, $\lambda = \{\lambda_i \leq 0; i \in I\}$, $\mu = \{\mu_t \leq 0; t \in T\}, \pi 1 = \{\pi_{1tu} \leq 0; t \in T, u \in U_t, u > 1\}$ and $\pi 2 = \{\pi_{2tu} \leq 0; t \in T, u \in U_t, u > 1\}$, be the dual variables associated with constraints (67)–(68), (69), (70), (71)–(72), (73), (74), (75) and (76), respectively. Using the right-hand side of constraints (67)–(76) and the corresponding dual variables, two types of Benders cuts can be produced to add to the Benders master problem at each iteration during the Benders-based branch-and-cut algorithm.

Feasibility cut to Model 2: this type of cut is generated when the Benders subproblem is infeasible, and hence, the associated Benders dual subproblem is unbounded:

$$\sum_{f\in F} D_f \widetilde{\alpha}_{f\overline{T}} + \sum_{t\in T}\left(CapF_t - \sum_{f,q\in F} st_{fq}Y_{fqt}\right)\widetilde{\beta}_t + \sum_{t\in T}\sum_{f\in F}\left(CapF_t\overline{Y}_{ft} + CapF_t\sum_{q\in F}Y_{qft}\right)\widetilde{\gamma}_{ft}$$
$$-\sum_{t\in T\setminus\overline{T}}\sum_{u\in U_t}\sum_{i\in I} a_{iu}\widetilde{\delta}_{it} + \left(\sum_{u\in U_{\overline{T}}}\sum_{i\in I} a_{iu} + \sum_{i\in I} SI_i\right)\widetilde{\delta}_{i\overline{T}} + \sum_{i\in I} SI_i\widetilde{\lambda}_i + \sum_{t\in T}(CapC_t - vt\ C_u)\widetilde{\mu}_t$$
$$+\sum_{t\in T}\sum_{u\in U_t,u>1}\sum_{i\in I}(-a_{iu} - b_{iu} + a_{i,u-1} + b_{i,u-1})\widetilde{\pi}_{1tu}$$
$$+\sum_{t\in T}\sum_{u\in U_t,u>1}\sum_{i\in I}(a_{iu} + b_{iu} - a_{i,u-1} - b_{i,u-1})\widetilde{\pi}_{2tu} \leq 0 \tag{78}$$

where, $\widetilde{\alpha}$, $\widetilde{\beta}$, $\widetilde{\gamma}$, $\widetilde{\delta}$, $\widetilde{\lambda}$, $\widetilde{\mu}$, $\widetilde{\pi}$ are the values of the extreme rays of the unbounded dual space Λ of the Benders subproblem.

Optimality cut to Model 2: is generated when the Benders subproblem is feasible, hence, the Benders dual subproblem is bounded.

$$\sum_{f\in F} D_f \widetilde{\alpha}_{f\overline{T}} + \sum_{t\in T}\left(CapF_t - \sum_{f,q\in F} st_{fq}Y_{fqt}\right)\widetilde{\beta}_t + \sum_{t\in T}\sum_{f\in F}\left(CapF_t\overline{Y}_{ft} + CapF_t\sum_{q\in F}Y_{qft}\right)\widetilde{\gamma}_{ft}$$
$$-\sum_{t\in T\setminus\overline{T}}\sum_{u\in U_t}\sum_{i\in I} a_{iu}\widetilde{\delta}_{it} + \left(\sum_{u\in U_{\overline{T}}}\sum_{i\in I} a_{iu} + \sum_{i\in I} SI_i\right)\widetilde{\delta}_{i\overline{T}} + \sum_{i\in I} SI_i\widetilde{\lambda}_i + \sum_{t\in T}(CapC_t - vt\ C_u)\widetilde{\mu}_t$$
$$+\sum_{t\in T}\sum_{u\in U_t,u>1}\sum_{i\in I}(-a_{iu} - b_{iu} + a_{i,u-1} + b_{i,u-1})\widetilde{\pi}_{1tu}$$
$$+\sum_{t\in T}\sum_{u\in U_t,u>1}\sum_{i\in I}(a_{iu} + b_{iu} - a_{i,u-1} - b_{i,u-1})\widetilde{\pi}_{2tu} \leq \eta \tag{79}$$

where, $\widetilde{\alpha}$, $\widetilde{\beta}$, $\widetilde{\gamma}$, $\widetilde{\delta}$, $\widetilde{\lambda}$, $\widetilde{\mu}$, $\widetilde{\pi}$ are the optimal values associated with the dual variables in the dual space Ω of the Benders subproblem. The variable η is a continuous variable that is used to provide a lower bound on the costs of the Benders subproblem in the Benders master problem.

Therefore, Model 2 can be reformulated as the following Benders master problem:

Benders master problem to Model 2

$$\min\sum_{t\in T}\left(\sum_{u\in U_t} vc\ C_u + \sum_{f,q\in F} sc_{fq}Y_{fqt}\right) + \eta \tag{80}$$

Subject to:

(38)–(44), (27)–(34)

$$\sum_{f\in F} D_f\alpha_{f\overline{T}} + \sum_{t\in T}\left(CapF_t - \sum_{f,q\in F} st_{fq}Y_{fqt}\right)\beta_t$$
$$+\sum_{t\in T}\sum_{f\in F}\left(CapF_t\overline{Y}_{ft} + CapF_t\sum_{q\in F}Y_{qft}\right)\gamma_{ft}$$
$$-\sum_{t\in T\setminus\overline{T}}\sum_{u\in U_t}\sum_{i\in I} a_{iu}\delta_{it} + \left(\sum_{u\in U_{\overline{T}}}\sum_{i\in I} a_{iu} + \sum_{i\in I} SI_i\right)\delta_{i\overline{T}} + \sum_{i\in I} SI_i\lambda_i$$

$$+\sum_{t\in T}(CapC_t - vt\ C_u)\mu_t + \sum_{t\in T}\sum_{u\in U_t,u>1}\sum_{i\in I}(-a_{iu} - b_{iu} + a_{i,u-1} + b_{i,u-1})\pi_{1tu}$$

$$+\sum_{t\in T}\sum_{u\in U_t,u>1}\sum_{i\in I}(a_{iu} + b_{iu} - a_{i,u-1} - b_{i,u-1})\pi_{2tu} \leq 0 \quad \forall(\alpha,\beta,\gamma,\delta,\lambda,\mu,\pi)\in\Lambda \tag{81}$$

$$\sum_{f\in F} D_f\alpha_{f\overline{T}} + \sum_{t\in T}\left(CapF_t - \sum_{f,q\in F} st_{fq}Y_{fqt}\right)\beta_t$$

$$+\sum_{t\in T}\sum_{f\in F}\left(CapF_t\overline{Y}_{ft} + CapF_t\sum_{q\in F}Y_{qft}\right)\gamma_{ft}$$

$$-\sum_{t\in T\setminus\overline{T}}\sum_{u\in U_t}\sum_{i\in I}a_{iu}\delta_{it} + \left(\sum_{u\in U_{\overline{T}}}\sum_{i\in I}a_{iu} + \sum_{i\in I}SI_i\right)\delta_{i\overline{T}} + \sum_{i\in I}SI_i\lambda_i$$

$$+\sum_{t\in T}(CapC_t - vt\ C_u)\mu_t + \sum_{t\in T}\sum_{u\in U_t,u>1}\sum_{i\in I}(-a_{iu} - b_{iu} + a_{i,u-1} + b_{i,u-1})\pi_{1tu}$$

$$+\sum_{t\in T}\sum_{u\in U_t,u>1}\sum_{i\in I}(a_{iu} + b_{iu} - a_{i,u-1} - b_{i,u-1})\pi_{2tu} \leq \eta \quad \forall(\alpha,\beta,\gamma,\delta,\lambda,\mu,\pi)\in\Omega \tag{82}$$

$$\overline{Y}_{ft},\ Y_{fqt}\in\{0,1\} \qquad \forall f,q,\forall t \tag{83}$$

$$a_{iu},\ b_{iu}\in\mathbb{Z}_+,\ C_u,\ N_u\in\{0,1\} \qquad \forall i,\forall t,u\in U_t \tag{84}$$

The objective function (80) minimizes the cost for the material of the objects used, the sequence-dependent setup costs of products, and the lower bound on the cost of the Benders subproblem. Constraints (81) and (82) are the feasibility and optimality cuts from the Benders reformulation, respectively.

In the literature of Benders decomposition, it is well known that the Benders optimality cuts (82) and the Benders feasibility cuts (81) can be generated from any solution of the master problem, i.e., any fractional and/or integer feasible solution. In this way, the coupled bin-packing and lot-sizing problem can be solved in a standard B&C framework with the use of callbacks, available in general-purpose solvers via a Benders-based branch-and-cut algorithm (Adulyasak et al., 2015). For any feasible solution of the Benders master problem found at each node of the B&B tree, the Benders subproblem is solved, generating feasibility or optimality Benders cuts. In this study, the fractional solutions are only considered at the root node of the B&B tree. Note that in the Benders-based branch-and-cut algorithm, when the non-polynomial-sized set of subtour elimination constraints are used, in addition to the Benders feasibility and optimality cuts, the violated subtour elimination cuts are also added to the Benders master problem according to the selected strategy (B&C1, B&C2 or B&C3).

One of the drawbacks of the Benders decomposition is the weak estimation of the cost of the Benders subproblem in the Benders master problem during the search process, mainly at the earlier iterations of the algorithm. To deal with this issue, we added some lower bound inequalities to the master problem in order to better approximate the cost of the subproblem. The first constraints (85) guarantee that there is at least one setup for each product in the scheduling horizon, where this setup consists of either a setup at the beginning of a time period or a changeover to the product. Constraints (86) impose a minimum number of items to be cut in the scheduling horizon, based on the number of items needed to meet the demand of products and the initial inventory of items available in the first time period. As there is a lead-time of one time period, the items should be cut until the penultimate time period. The last valid constraints (87) are related to the capacity in the cutting process of objects, which impose that the total capacity consumed to cut the objects should not exceed the available capacity

$$\sum_{t\in T}\left(\overline{Y}_{ft} + \sum_{q\in F}Y_{qft}\right) \geq 1 \qquad \forall f \tag{85}$$

$$\sum_{t\in T\setminus\{\overline{T}\}}\sum_{u\in U_t} a_{iu} \geq \sum_{f\in F} r_{if}D_f - SI_i \qquad \forall i \tag{86}$$

$$\sum_{u\in U_t} vt\ C_u \leq CapC_t \qquad \forall t \tag{87}$$

5 Numerical experiments

This section presents the data generation and computational results obtained for the coupled bin-packing and lot-sizing problem with sequence-dependent setups. The proposed models and solution approaches were implemented in C++ using the C callable library of IBM ILOG Cplex 12.10.0. All the computational tests were conducted using a computer with 2 processors Intel(R) 3.07GHz (96 of RAM). We turned off CPLEX's parallel mode, set the CPLEX MIP tolerance to 10^{-3}, impose the use of a single processor (threads = 1), and a time limit of 3600 seconds, when solving the integrated problem. All other CPLEX parameters were set to their default values. The tested instances and computational experiments are discussed in the following sections.

5.1 Data set

The data set used to generate instances for the integrated problem is based on data from the literature (Trigeiro et al., 1989; Gau and Wäscher, 1995) with several adaptations. When a range $[a, b]$ is given, this indicates that the data is drawn from a uniform distribution in this range.

Number of time periods (days)	$\overline{T} = 5$
Number of products	$\overline{F} = \{2, 3, 5, 7, 10\}$
Number of items	$\overline{I} = \overline{F}$ (each item corresponds directly to a product)
Number of items i needed to make product f	$r_{if} = \begin{cases} 1, & \text{if } i = f; \\ 0, & \text{otherwise.} \end{cases}$
Demand of products	$D_f \in [1, Dmax]$, where $Dmax = \{3, 5, 7\}$
Object length	$L = 10,000$
Item length	$l_i \in [0.25, 0.75]\ L$
Number of objects available in the cutting process	$\overline{O} = \left\lceil \left(\frac{\sum_{f\in F}\sum_{i\in I} l_i r_{if} D_f}{L} \right) 1.5 \right\rceil$ Note that, as this study addresses mathematical models that use the assignment variables to model the bin-packing decisions (Kantorovich, 1960), it is necessary to impose an upper bound on the number of objects available to the bin-packing decisions. The basis for calculating the upper bound, after some preliminary computational tests, is given as the integer number when multiplying 1.5 times the total number of objects needed, without waste, to meet the independent demand of products in the scheduling horizon. The number of objects available is also used as the number of micro-periods in each time period, i.e., $eu_t = O,\ \forall t$.
Mean production time of cutting an object	$vt = 1$
Changeover time between objects	$st \in [1, 10]$
Cost of an object	$vc = L$

Changeover cost between objects	$sc = \frac{st}{\overline{M}}$; where $\overline{M}$ is an upper bound on the total number of items cut from an object, i.e., $$\overline{M} = \left\lfloor \frac{L}{\min_{i \in I} l_i} \right\rfloor .$$ Note that the value for the setup costs are taken as not relevant for the problem, i.e., the influence of the sequence-dependent setups is mainly in the resources (capacity). The setup costs are treated as small penalties in the objective function in order to prevent superfluous setups when there is an excess of capacity, i.e., they are used to prevent that the model performs changeovers of operations without having any cutting/production. However, according to the practical application other values to the setup cost can be used.
Initial inventory of items	$SI_i = 0.2 \sum_{f \in F} r_{if} D_f$; The initial inventory of items is used as a feasibility strategy due to the positive lead-times imposed during the production process. In this way, the items that are not for the production of products in the first time period, are not carried over in inventory in order to be used in latter time periods to produce products.
Capacity of the cutting machine	$CapC_t = \left\lceil \frac{vt\ \overline{O} + \overline{Tst}\ \overline{O}}{\overline{T}} \right\rceil$; The average changeover time $\overline{Tst}$ is calculated as the average of the total number of changeovers needed between objects considering a cutting process using only a homogenous assignment of items.
Production time of products	$vt_f = 1$
Changeover time between products	$st_{fq} \in [21, 65]$
Changeover cost between products	$sc_{fq} = \frac{st_{fq}}{65}$
Capacity in the production of products	$CapF_t = cap_t$; The capacity cap_t is calculated considering the total demand of products in the scheduling horizon and the average values of the changeovers time among the products ($\overline{st_{fq}}$), i.e., $$cap_t = \left\lceil \frac{cap}{\overline{T}} \right\rceil, \text{with } cap = \sum_{f \in F} \left(vt_f D_f + \frac{\sum_{q \in F} \overline{st_{fq}}}{\overline{F}} \right).$$

The data set described in this section defines 15 classes which consist of variations in terms of the number of products and the intervals of the demand distribution of products (low/medium/high). For each class, 5 instances were generated, totaling 75 instances. Table 1 shows the 15 classes obtained with the data variation.

Table 1: Classes of instances

Classes	Description	Classes	Description	Classes	Description
Class 1	$\overline{F} = 2/Dmax = 3$	Class 6	$\overline{F} = 2/Dmax = 5$	Class 11	$\overline{F} = 2/Dmax = 7$
Class 2	$\overline{F} = 3/Dmax = 3$	Class 7	$\overline{F} = 3/Dmax = 5$	Class 12	$\overline{F} = 3/Dmax = 7$
Class 3	$\overline{F} = 5/Dmax = 3$	Class 8	$\overline{F} = 5/Dmax = 5$	Class 13	$\overline{F} = 5/Dmax = 7$
Class 4	$\overline{F} = 7/Dmax = 3$	Class 9	$\overline{F} = 7/Dmax = 5$	Class 14	$\overline{F} = 7/Dmax = 7$
Class 5	$\overline{F} = 10/Dmax = 3$	Class 10	$\overline{F} = 10/Dmax = 5$	Class 15	$\overline{F} = 10/Dmax = 7$

5.2 Results: Subtour elimination constraints

In this section, we present a comparison between the different approaches used to deal with the subtour elimination constraints in the coupled bin-packing and lot-sizing problem. The strategies consist of three variations of the polynomial constraints, *MTZ1*, *MTZ2*, and *MTZ3*, which are solved by an optimization package, and three variations of the branch-and-cut algorithm for solving the non-polynomial-sized set of constraints, B&C1, B&C2, and B&C3. In this computational experiment, two instances are infeasible, one in Class 1 and another in Class 6. According to the data set, an explanation for the infeasibility is due to the number of objects available in the cutting process which is not sufficient to accommodate all the demanded items. This could be changed by altering the factor (1.5) used to generate the number of objects (see the calculation of the number of objects available in the cutting process in the previous subsection 5.1). The performance of the approaches are assessed with respect to the number of optimal/feasible solutions, found by each solution method (column Optimal/Feasible/Total) and the gap reported at the end of the time limit (column Gap), which is calculated considering the best values for the lower and upper bound found by each model and solution approach. The number in brackets corresponds to the number of instances for which all algorithms found a feasible solution and are used to calculate the average values for the gap in each class.

Table 2 shows the results of the subtour elimination constraints added to Model 1 for each class of instances. Comparing the number of feasible solutions found to the integrated problem, *MTZ1* is able to find the highest number (71 out of 73, from which 37 are optimal), whereas *MTZ3* presented the worst result (62 out of 73, with 35 optimal). The gaps present a similar behavior in most of the classes and are quite high for classes with more than five products. On average, the best results for the gap in each variation of subtour elimination constraints are 7.26% found by *MTZ1* and 8.02% found by B&C2. The computational time is not reported in this table, however, it is very similar among all the approaches, with an average of 1793 seconds.

Considering the approaches for Model 2, due to the different strategy addressed to model the sequence-dependent setups in the cutting process, Model 2 is able to find a feasible solution for all the instances present in this computational study, except the two infeasible instances. The computational time spent to solve the integrated problem is, on average, 1800 seconds. Table 3 presents, for each class and strategy to deal with the non-polynomial-sized set of subtour elimination constraints, the values for the gap found after either reaching the *MIP* tolerance or the time limit. The best overall value for the gap is 7.71% found by the B&C1, followed by *MTZ3* with 7.80%. However, the difference with the worst results is less than 1%.

The averages reported in Table 2 for Model 1 and in Table 3 for Model 2 cannot be directly compared to each other, since they are calculated over different sets of instances. In Table 4, we present the results for both models, calculated over the instances for which all approaches provide a feasible solution. In an overall comparison between Model 1 and Model 2, we can state that Model 2 is able to better manage the sequence-dependent setups in the cutting process, since it finds a feasible solution to all the instances in this computational study and the highest number of optimal solutions. In all the analyses of this section, the solution approaches are able to find good results for most of the classes with a small number of products ($F = 2, 3$), however, when the number of products increases the same behavior is not observed. Due to this fact, we present a Benders-based branch-and-cut solution approach for Model 2, in which the Benders cuts along with the B&C1 strategy for the violated subtour elimination cuts, are added at the nodes of the branch-and-bound tree.

5.3 Benders-based branch-and-cut algorithm - BbB&C

In this section, we present the computational results obtained by applying the Benders-based branch-and-cut algorithm to Model 2, called here BbB&C1. The Benders cuts and the violated subtour elimination cuts (B&C1 strategy) are applied to integer feasible solutions found during the search of the branch-and-bound tree. In addition, the Benders cuts are also added to fractional solutions at the root node of the B&B tree. The main advantage of adding cuts based on fractional solutions is that, when added at the root node, the generated cuts are valid for the whole search tree. We limit the number of cuts added at the root node to 10, 50, and 100 fractional cuts, in order to not add too many

Table 2: Computational results to Model 1: Subtour elimination constraints

	Optimal/Feasible/Total				Gap		
Classes	MTZ1	MTZ2	MTZ3		MTZ1	MTZ2	MTZ3
Class 1*	4/ 4/ 4	4/ 4/ 4	4/ 4/ 4	(4)	0.03	0.03	0.04
Class 2	5/ 5/ 5	5/ 5/ 5	5/ 5/ 5	(5)	0.02	0.02	0.02
Class 3	3/ 5/ 5	4/ 5/ 5	4/ 5/ 5	(5)	5.58	2.89	2.91
Class 4	1/ 5/ 5	1/ 5/ 5	1/ 5/ 5	(5)	13.82	15.64	16.19
Class 5	1/ 5/ 5	0/ 5/ 5	0/ 4/ 5	(4)	8.63	13.51	17.32
Sum/Average	14/24/24	14/24/24	14/23/24	(23)	5.73	6.39	7.18
Class 6*	4/ 4/ 4	4/ 4/ 4	4/ 4/ 4	(4)	0.02	0.02	0.05
Class 7	5/ 5/ 5	5/ 5/ 5	5/ 5/ 5	(5)	0.03	0.03	0.04
Class 8	2/ 5/ 5	1/ 5/ 5	2/ 5/ 5	(5)	7.31	11.91	10.00
Class 9	0/ 5/ 5	0/ 5/ 5	0/ 5/ 5	(5)	21.16	24.47	24.47
Class 10	0/ 4/ 5	0/ 4/ 5	0/ 1/ 5	(1)	13.35	16.42	16.44
Sum/Average	11/23/24	10/23/24	11/20/24	(20)	7.80	9.93	9.46
Class 11	5/ 5/ 5	5/ 5/ 5	5/ 5/ 5	(5)	0.03	0.03	0.03
Class 12	5/ 5/ 5	5/ 5/ 5	5/ 5/ 5	(5)	0.05	0.02	0.02
Class 13	1/ 5/ 5	1/ 5/ 5	0/ 5/ 5	(5)	19.82	21.45	28.85
Class 14	0/ 5/ 5	0/ 5/ 5	0/ 4/ 5	(4)	15.74	17.12	17.12
Class 15	1/ 4/ 5	0/ 3/ 5	0/ 0/ 5	(0)	—-	—-	—-
Sum/Average	12/24/25	11/23/25	10/19/25	(19)	8.55	9.26	11.21
Overall Sum/Average	37/71/73	35/70/73	35/62/73	(62)	7.26	8.41	9.15

	Optimal/Feasible/Total				Gap		
Classes	B&C1	B&C2	B&C3		B&C1	B&C2	B&C3
Class 1*	4/ 4/ 4	4/ 4/ 4	4/ 4/ 4	(4)	0.03	0.04	0.03
Class 2	5/ 5/ 5	5/ 5/ 5	5/ 5/ 5	(5)	0.03	0.02	0.03
Class 3	2/ 5/ 5	3/ 5/ 5	3/ 5/ 5	(5)	10.96	5.13	6.82
Class 4	1/ 5/ 5	1/ 5/ 5	1/ 5/ 5	(5)	13.22	13.83	13.82
Class 5	1/ 5/ 5	1/ 5/ 5	1/ 4/ 5	(4)	8.64	8.64	8.64
Sum/Average	13/24/24	14/24/24	14/23/24	(23)	6.77	5.63	6.00
Class 6*	4/ 4/ 4	4/ 4/ 4	4/ 4/ 4	(4)	0.03	0.03	0.02
Class 7	5/ 5/ 5	5/ 5/ 5	5/ 5/ 5	(5)	0.03	0.03	0.03
Class 8	2/ 5/ 5	1/ 5/ 5	1/ 5/ 5	(5)	7.30	10.06	8.72
Class 9	0/ 5/ 5	0/ 5/ 5	0/ 5/ 5	(5)	21.16	21.16	21.16
Class 10	0/ 3/ 5	0/ 2/ 5	0/ 1/ 5	(0)	—-	—-	—-
Sum/Average	11/22/24	10/21/24	10/20/24	(19)	7.50	8.23	7.88
Class 11	5/ 5/ 5	5/ 5/ 5	5/ 5/ 5	(5)	0.02	0.02	0.02
Class 12	5/ 5/ 5	5/ 5/ 5	5/ 5/ 5	(5)	0.04	0.05	0.04
Class 13	0/ 5/ 5	0/ 5/ 5	0/ 5/ 5	(5)	25.31	24.93	25.34
Class 14	0/ 5/ 5	0/ 5/ 5	0/ 5/ 5	(5)	15.27	15.27	17.26
Class 15	0/ 3/ 5	0/ 2/ 5	0/ 2/ 5	(2)	10.89	13.03	13.03
Sum/Average	10/23/25	10/22/25	10/22/25	(22)	10.22	10.34	10.88
Overall Sum/Average	34/69/73	34/67/73	34/65/73	(64)	8.17	8.02	8.24

(*) Class with one infeasible instance

cuts which would remove only a relatively small portion of the search space and would make the size of the Benders master problem increase considerably.

Table 3: Computational results to Model 2: Subtour elimination constraints

	Gap			Gap		
Classes	MTZ1	MTZ2	MTZ3	B&C1	B&C2	B&C3
Class 1	0.02	0.01	0.02	0.00	0.00	0.00
Class 2	0.05	0.06	0.03	0.05	0.05	0.04
Class 3	0.08	1.75	0.09	0.10	0.08	0.10
Class 4	15.75	14.56	15.27	13.66	14.68	13.70
Class 5	15.30	15.31	15.27	15.30	15.28	15.30
Average	6.24	6.34	6.14	5.82	6.02	5.83
Class 6	0.01	0.01	0.01	0.03	0.01	0.03
Class 7	0.08	0.07	0.04	0.05	0.04	0.05
Class 8	6.36	8.60	6.68	7.49	9.29	9.37
Class 9	23.40	24.36	22.50	23.33	21.61	23.21
Class 10	14.70	14.70	14.70	14.70	14.62	14.70
Average	8.91	9.55	8.79	9.12	9.11	9.47
Class 11	0.01	0.02	0.01	0.01	0.01	0.01
Class 12	0.04	0.02	0.05	0.03	0.02	0.03
Class 13	11.23	16.52	9.97	13.40	15.20	13.61
Class 14	17.01	17.09	17.09	16.65	16.85	16.72
Class 15	10.78	14.96	15.08	10.85	14.95	10.87
Average	7.81	9.72	8.44	8.19	9.41	8.25
Overall Average	7.85	8.54	7.80	7.71	8.18	7.85
Number Optimal Instances	36	36	35	38	36	37

All the instances found a feasible solution, except one infeasible instance in Class 1 and Class 6

Table 4: Computational results to Models 1 and 2: Gap

		Model 1						Model 2					
Classes		MTZ1	MTZ2	MTZ3	B&C1	B&C2	B&C3	MTZ1	MTZ2	MTZ3	B&C1	B&C2	B&C3
Class 1*	(4)	0.03	0.03	0.04	0.03	0.04	0.03	0.02	0.01	0.02	0.00	0.00	0.00
Class 2	(5)	0.02	0.02	0.02	0.03	0.02	0.03	0.05	0.06	0.03	0.05	0.05	0.04
Class 3	(5)	5.58	2.89	2.91	10.96	5.13	6.82	0.08	1.75	0.09	0.10	0.08	0.10
Class 4	(5)	13.82	15.64	16.19	13.22	13.83	13.82	15.75	14.56	15.27	13.66	14.68	13.70
Class 5	(4)	8.63	13.51	17.32	8.64	8.64	8.64	13.50	13.50	13.50	13.50	13.47	13.50
Average	(23)	5.73	6.39	7.18	6.77	5.63	6.00	5.80	5.91	5.70	5.35	5.56	5.36
Class 6*	(4)	0.02	0.02	0.05	0.03	0.03	0.02	0.01	0.01	0.01	0.03	0.01	0.03
Class 7	(5)	0.03	0.03	0.04	0.03	0.03	0.03	0.08	0.07	0.04	0.05	0.04	0.05
Class 8	(5)	7.31	11.91	10.00	7.30	10.06	8.72	6.36	8.60	6.68	7.49	9.29	9.37
Class 9	(5)	21.16	24.47	24.47	21.16	21.16	21.16	23.40	24.36	22.50	23.33	21.61	23.21
Class 10	(0)	—-	—-	—-	—-	—-	—-	—-	—-	—-	—-	—-	—-
Average	(19)	7.51	9.59	9.09	7.50	8.23	7.88	7.85	8.69	7.69	8.13	8.14	8.59
Class 11	(5)	0.03	0.03	0.03	0.02	0.02	0.02	0.01	0.02	0.01	0.01	0.01	0.01
Class 12	(5)	0.05	0.02	0.02	0.04	0.05	0.04	0.04	0.02	0.05	0.03	0.02	0.03
Class 13	(5)	19.82	21.45	28.85	25.31	24.93	25.34	11.23	16.52	9.97	13.40	15.20	13.61
Class 14	(4)	15.74	17.12	17.12	15.75	15.74	16.89	17.02	17.11	17.11	17.00	17.09	17.10
Class 15	(0)	—-	—-	—-	—-	—-	—-	—-	—-	—-	—-	—-	—-
Average	(19)	8.55	9.26	11.21	9.99	9.89	10.24	6.55	7.96	6.24	7.12	7.61	7.19
Overall Average	(61)	7.16	8.28	9.03	8.00	7.77	7.91	6.68	7.42	6.49	6.77	7.00	6.94

(*) Class with one infeasible instance

Table 5 shows the linear relaxation at the root node and the best lower bound found by the approaches considering no addition of fractional cuts and the three variations with respect to the number of fractional cuts added. We can see that, as the number of added fractional cuts increases, the quality of the linear relaxation at the root node of the branch-and-bound tree also increases for all classes. On average, the improvement in the linear relaxation is almost 7%, when comparing the algorithm without fractional cuts and with the addition of 100 fractional cuts. Considering the best lower bound found by the approaches (after exploring the branch-and-bound tree), the best values are those found by adding 100 fractional cuts. The lower bound based on this latter strategy are around 1.7% better compared to adding no fractional cuts at all. An interesting observation in this analysis is that the total number of cuts, over all the instances, added by the different approaches during the branch-and-bound process is 335, 343, 306, and 305 respectively. This fact shows that adding fractional cuts at the root node improves the lower bound and also avoids the generation of many cuts during the search of the branch-and-bound tree. However, as these fractional cuts are easy to generate, there is no significant impact on the computational time, which is on average 1830 seconds to find an optimal or feasible solution to the integrated problem.

In Table 6, the values for the gaps obtained by Benders-based branch-and cut are compared with those obtained, in the previous experiments, by the branch-and-cut algorithm, B&C1. Only those instances that are solved by all the solution approaches are considered. For one instance in Class 15, BbB&C1_10frac was not able to find a feasible solution and this instance has been removed from this analysis. The Benders-based algorithm is able to reduce the gap for almost all the classes (13 out of 15 classes) and for those classes with high values, i.e., classes with more than 5 items (Classes 3–5, 8-10 and 13–15), the improvements in the gap can reach up to 67% (44% on average, from 7.56% to 4.20%), consuming slightly less computational time.

5.4 Automatic Benders decomposition

In this section, we perform a comparison among the best results obtained in our computational study against the brand new Automatic-Benders decomposition available in the commercial solver IBM CPLEX (IBM, 2019). In the automatic Benders, which acts as a black-box, the decomposition occurs as in the classical Benders decomposition, where the integer variables are maintained in the Benders master problem, whereas the continuous variables are assigned to the Benders subproblem, together with the linking constraints that contains integers and continuous variables. Several different improvements are used in the implementation of the Benders decomposition, such as a generic stabilization to solve the linear relaxation of the Benders master problem, stabilization techniques to deal with the numerical instabilities of the artificial variable in the Benders master problem, feasibility and optimality Benders cuts on numerical tolerances, among others. Related to the Benders subproblem, the automatic Benders decomposition manages to simplify the subproblem by handling linking constraints with a special structure and applying two normalization strategies to deal with the feasibility cuts which are added to the Benders master problem (Fischetti, 2010; Conforti and Wolsey, 2019). More details about the automatic Benders decomposition can be found in Bonami et al. (2019).

The comparison among the approaches is performed considering the best Benders-based branch-and-cut solution method based on the previous computational results (BbB&C1_100frac), the automatic Benders decomposition, called here *AutoBenders*, and the automatic Benders decomposition without reductions, called here *AutoBenders**. In the *AutoBenders* and *AutoBenders**, the Benders decomposition is selected in CPLEX. However, in *AutoBenders** some of its features are disabled. The idea here is to disable in *AutoBenders** the same features (the presolve and the dual reductions), which are also disabled when using callbacks in the CPLEX in order to provide a fair comparison between the automatic Benders and our customized algorithm. The callbacks are the tools responsible for adding the cuts during the search in the branch-and-bound tree. The aim here is to provide insights about the advantages and disadvantages of each solution approach and the impact of the features in the optimization software when solving optimization problems.

In order to improve the computational results obtained with the BbB&C1_100frac solution approach, we have tested normalization strategies for the selection of feasibility Benders cuts based on the concept of the *Minimal Infeasible Subsystem* (Fischetti, 2010) and *Maximum Feasible Subsystem* (Saharidis and Ierapetritou, 2010). The computational results showed none or only slight improve-

Table 5: Computational results to BbB&C1: Lower bound results

	Linear relaxation at the root node			
Classes	BbB&C1	BbB&C1_10frac	BbB&C1_50frac	BbB&C1_100frac
Class 1*	30000.69	34000.69	36000.69	36000.69
Class 2	38000.48	38000.48	39000.48	39000.48
Class 3	52000.97	52000.97	55720.77	60200.17
Class 4	76001.10	76001.08	77341.08	77427.48
Class 5	100001.92	100001.93	101135.33	101547.84
Average	59201.03	60001.03	61839.67	62835.33
Class 6*	32000.67	36732.14	37732.14	37732.14
Class 7	48000.54	48000.54	49850.14	51283.74
Class 8	88000.89	88000.89	88592.89	88592.89
Class 9	116001.34	116001.34	124061.54	125793.14
Class 10	160001.96	160001.99	162179.39	165297.99
Average	88801.08	89747.38	92483.22	93739.98
Class 11	46330.97	45330.97	50942.57	50942.57
Class 12	56000.46	62000.45	63798.05	63798.05
Class 13	110000.89	114692.89	123122.69	123195.29
Class 14	128001.38	128001.34	134256.14	134256.14
Class 15	202002.11	205449.68	220117.88	220242.08
Average	108467.16	111095.07	118447.47	118486.83
Overall Average	85489.76	86947.82	90923.45	91687.38

	Best lower bound			
Classes	BbB&C1	BbB&C1_10frac	BbB&C1_50frac	BbB&C1_100frac
Class 1*	43994.70	44000.58	44000.70	44000.70
Class 2	39991.86	39991.86	39992.20	39992.54
Class 3	63226.80	60643.58	62897.40	63894.62
Class 4	80486.92	82277.24	82385.08	83296.18
Class 5	100459.24	100151.46	103228.70	103762.22
Average	65631.90	65412.94	66500.82	66989.25
Class 6*	45993.70	45993.34	45993.96	45991.56
Class 7	53972.58	53977.88	53978.60	53974.36
Class 8	93947.22	94256.54	93965.14	94014.44
Class 9	131235.60	131147.60	134712.20	135922.40
Class 10	162545.80	164798.60	164795.80	165298.20
Average	97538.98	98034.79	98689.14	99040.19
Class 11	55993.84	55997.56	55999.98	55999.92
Class 12	64909.00	63985.06	64612.96	65969.36
Class 13	138005.40	137276.00	138729.60	137679.00
Class 14	134257.60	134484.60	139007.00	136580.80
Class 15	199312.25	195818.25	205713.75	205869.00
Average	118495.62	117512.29	120812.66	120419.62
Overall Average	93888.83	93653.34	95334.20	95483.02

(*) Class with one infeasible instance

ments and due to this fact, they are not presented in this paper. One strategy that has shown better improvements is the addition of Pareto-optimal cuts to the Benders master problem, called here BbB&C1_100frac_PC. This approach consists of selecting tighter cuts through the use of different core points (Magnanti and Wong, 1981). In our solution approach, the initial core point is computed by solving the linear relaxation of the original problem, disregarding the objective function, with the barrier algorithm without crossover (Bonami et al., 2019). The core point is then updated at each

Table 6: Comparisons: Gap

		Benders-based branch-and-cut			
Classes	B&C1	BbB&C1	BbB&C1_10frac	BbB&C1_50frac	BbB&C1_100frac
Class 1*	0.00	0.02	0.00	0.00	0.00
Class 2	0.05	0.03	0.03	0.02	0.02
Class 3	0.10	3.74	8.30	4.45	2.79
Class 4	13.66	9.73	8.11	8.01	7.18
Class 5	15.30	10.77	9.85	7.59	7.16
Average	5.82	4.86	5.26	4.01	3.43
Class 6*	0.03	0.02	0.02	0.02	0.03
Class 7	0.05	0.05	0.04	0.04	0.05
Class 8	7.49	3.70	3.42	3.68	3.64
Class 9	23.33	10.75	10.72	8.02	7.57
Class 10	14.70	10.73	9.63	9.13	8.85
Average	9.12	5.05	4.77	4.18	4.03
Class 11	0.01	0.01	0.01	0.01	0.00
Class 12	0.03	1.37	2.52	1.74	0.04
Class 13	13.40	8.40	8.90	8.00	8.61
Class 14	16.65	11.32	11.14	8.42	9.77
Class 15$^{(1)}$	8.65	9.32	9.10	7.32	7.24
Average	7.75	6.08	6.34	5.10	5.13
Overall Average	7.56	5.33	5.45	4.43	4.20

(*) Class with one infeasible instance (1) Class with one unknown solution

iteration of the algorithm considering a convex combination of the previous core point, multiplied by 0.5 and the solution from the Benders master problem multiplied by 0.5.

Table 7 shows a comparison of these approaches considering the values obtained for the upper bound and gap. As we can see, the methods result in upper bound values that are close to each other. However, the values for the gap show the big difference in the quality of the lower bounds provided by each solution approach. The Pareto-optimal cuts obtained in the BbB&C1_100frac_PC improve the quality of the gap when compared to the BbB&C1_100frac, for almost all classes (except the Classes 1, 6, 11 and 12, which are the classes with the smallest number of products), with an overall improvement of 40% (from 4.13% to 2.49%). In terms of computational time, the Pareto-optimal cuts also accelerate the solution procedure by around 23%. The *AutoBenders* from CPLEX is able to obtain an optimal solution in 65 out of 75 instances, mainly in classes with low and medium demand variation and it is two and three times faster than BbB&C1_100frac_PC and BbB&C1_100frac, respectively. An interesting analysis occurs when disabling the reductions in the automatic Benders (*AutoBenders**), in a comparison with the standard automatic benders (*AutoBenders*). In the *AutoBenders** the values for the gap considerably increase, as well as the computational time. Compared to *AutoBenders** approach, the BbB&C1_100frac_PC is able to reduce the values for the gap by 72% on average (from 8.87% to 2.49%) and it is 27% faster than the *AutoBenders**. Therefore, the standard automatic Benders in CPLEX (*AutoBenders*) is a numerical robust implementation of the Benders decomposition to solve these problems which benefits from some specific features that are disallowed, when implementing a custom Benders approach with callbacks. However, in a fair comparison, our Benders-based method with Pareto-optimal cuts can improve the computational results obtained by the *AutoBenders**, as well as the classical branch-and-cut algorithm.

Table 7: Comparisons: Upper bound and gap

Classes	AutoBenders	Upper bound AutoBenders*	BbB&C1_100frac	BbB&C1_100frac_PC
Class 1*	44000.70	44000.70	44001.10	44008.70
Class 2	40002.88	40002.72	40003.66	40004.02
Class 3	66007.08	66006.64	66007.38	66008.14
Class 4	90006.16	90005.56	90008.26	90008.30
Class 5	112010.14	112007.76	112012.40	112012.56
Average	70405.39	70404.68	70406.56	70408.34
Class 6*	46001.06	46001.66	46001.06	46001.46
Class 7	54002.68	54002.68	54005.16	54003.70
Class 8	98004.12	98003.70	98007.14	98005.20
Class 9	148010.60	148005.40	148006.20	148006.40
Class 10$^{(1)}$	182008.60	182010.80	182016.40	182014.20
Average	105605.41	105604.85	105607.19	105606.19
Class 11	56003.20	56001.56	56002.34	56004.26
Class 12	66003.38	66002.18	66003.48	66007.78
Class 13	150005.00	150009.20	150003.80	150006.00
Class 14	152009.20	152007.60	152008.20	152009.00
Class 15	217512.50	217510.75	220009.75	220009.75
Average	128306.66	128306.26	128805.51	128807.36
Overall Average	101439.15	101438.59	101606.42	101607.30

Classes	AutoBenders	Gap AutoBenders*	BbB&C1_100frac	BbB&C1_100frac_PC
Class 1*	0.00	0.06	0.00	0.02
Class 2	0.00	0.07	0.02	0.02
Class 3	0.03	0.08	2.79	0.90
Class 4	0.05	16.52	7.18	0.75
Class 5	0.17	15.31	7.16	7.16
Average	0.05	6.41	3.43	1.77
Class 6*	0.03	0.04	0.03	0.04
Class 7	0.01	0.07	0.05	0.02
Class 8	0.02	11.34	3.64	1.78
Class 9	0.00	24.34	7.57	3.00
Class 10$^{(1)}$	2.98	14.67	7.80	7.80
Average	0.61	10.09	3.82	2.53
Class 11	0.00	0.04	0.00	0.01
Class 12	0.00	0.05	0.04	0.05
Class 13	0.01	19.19	8.61	1.70
Class 14	1.85	17.43	9.77	6.92
Class 15	2.32	13.78	7.24	7.24
Average	0.84	10.10	5.13	3.18
Overall Average	0.50	8.87	4.13	2.49

(*) Class with one infeasible instance (1) Class with one unknown solution

6 Conclusions and research directions

In this paper, we study the coupled bin-packing and lot-sizing problem with sequence-dependent setups. In the upstream bin-packing decisions, a strong heterogeneous assortment of items has to be assigned to a minimal number of identical size objects by performing cutting operations. These items are provided for the downstream lot-sizing decisions and have to go through some additional process before being ready as products and meet the client's requirements. In this paper, we are interested in the production scheduling decisions, i.e., in the scheduling of the cutting operations needed to cut objects into items and the processes performed in the production of products. In the bin-packing problem, the scheduling consists of the removal/insertion process of knives in different positions, whereas in the lot-sizing problem the scheduling comprises the processes (drilling, folding, or assembly) necessary to transform items into products. Each one of the processes considers capacity limitations.

The two formulations proposed for the integrated problem consist of mixed-integer nonlinear mathematical models that address the bin-packing decision using assignment variables (Kantorovich, 1960) and simultaneously incorporate the sequence-dependent setups according to knife movements. The scheduling decisions are modeled by the *ATSP* constraints in Model 1 and on the idea of micro-periods and a phantom cutting process in Model 2. The non-linear models are linearized and two types of subtour elimination constraints (polynomial and a non-polynomial-sized set of constraints) are employed in order to find a feasible solution to the integrated problem. The mixed-integer linear model with a polynomial set of subtour elimination constraints is solved by an optimization package, whereas the model with the non-polynomial-sized set of constraints is solved using an exact branch-and-cut algorithm. Finally, a Benders-based branch-and-cut algorithm that adds violated subtour elimination cuts, in addition to the Benders feasibility and optimality cuts, is presented to find better solutions to the integrated problem.

In the computational results, Model 1 has shown to have more difficulty finding feasible solutions, when compared to Model 2 which is able to find a feasible solution to all the instances. In terms of the quality of the solutions, Model 1 with *MTZ1* presented the best overall average for the gap, 7.26%, followed by Model 2 with B&C1 approach, 7.71% on average. The reason for Model 2 finding more feasible solutions, when compared to Model 1 is mainly due to the features used to model the sequence-dependent setups, which influences the number of constraints and variables of the mathematical model. In addition, due to the modeling of the problem, Model 1 can represent more general cases of practical applications, when compared to Model 2. More specifically, as Model 1 has indices which represent each object type specifically (indexes o, k, v), it can be used to model a cutting process where the type of objects is also relevant to the problem. In this case, these object types might have different costs, cutting times and sizes, hence, the definition of many parameters should be considered for each type of object o, i.e., vc_o, vt_o and L_o instead of vc, vt and L parameters. This fact can also be considered in addition to the sequence-dependent setups, in this way, the parameters related to the sequence-dependent setups should be changed from sc and st to sc_{ok} and st_{ok}. Such features can be modeled in Model 1 with minor changes, however, these changes cannot be directly incorporated in Model 2. Therefore, we can state that Model 2 might represent a more specific case of the problem, when compared to Model 1.

As some of the gaps are still quite high for large instances, a Benders-based branch-and-cut algorithm is developed to solve the integrated problem. The Benders-based algorithm with Pareto-cuts is able to reduce the gap for almost all the classes with an overall average improvement of around 40% and consuming slightly less computational time, when compared to branch-and-cut based approaches. Finally, we compare our Benders-based algorithm with the one presented in the commercial solver IBM CPLEX. Although, the automatic Benders is able to find optimal solutions for 85% of the instances by considerably improving the lower bound of the instances, in a fair comparison, by disabling the same features which are also disabled in our Benders-based algorithm (the presolve and the dual reductions), our approach considerably outperforms the results of the commercial solver.

Opportunities for future studies can be found by extending the assumptions of the models in order to embrace some other practical conditions. In the cutting process, a two/three-dimensional bin-packing can be considered, while in the lot-sizing problem, the backlog for the final products, when the demand of clients are not satisfied in the demanded time period (Pochet and Wolsey, 1988), and setup crossover (Fiorotto et al., 2017), in order to better describe some specific production processes, may be taken into account in the mathematical models. In an attempt to solve large problems, heuristics and matheursitics can be considered as alternative solution methods for these type of problems.

Appendix

A Linearization of the mathematical models

The mathematical Model 1 (see Section 3) consists of a mixed-integer nonlinear formulation. The first set of non-linear constraints comprises the absolute value, present in constraints (16) and (17). The absolute value linearization consists of replacing each one of the constraints (16) and (17) by two linear constraints (A1)–(A2) and (A3)–(A4), respectively, in order to always guarantee a non-negative value to the terms in the absolute value, hence, to the TW_{okt} variable.

$$TW_{okt} \geq na_{kt} - na_{ot} - M(1 - W_{okt}) \qquad \forall o, k, o \neq k, \forall t \tag{A1}$$

$$TW_{okt} \geq -na_{kt} + na_{ot} - M(1 - W_{okt}) \qquad \forall o, k, o \neq k, \forall t \tag{A2}$$

$$TW_{oot} \geq na_{ot} - na_{o,t-1} - M(1 - \overline{W}_{ot}) \qquad \forall o, \forall t > 1 \tag{A3}$$

$$TW_{oot} \geq -na_{ot} + na_{o,t-1} - M(1 - \overline{W}_{ot}) \qquad \forall o, \forall t > 1 \tag{A4}$$

The second type of non-linear constraints consist of the product of variables present in constraints (18). The linearization strategy comprises additional variables for each multiplication of variables and an additional set of constraints guaranteeing the equivalence among the formulations. The first multiplication of variables in (18) occurs between an integer variable ($\sum_{i \in I} a_{iot}$) and a binary variable (W_{okt}), creating the new variables according to constraints (A5), and the set of additional constraints (A6)–(A8).

$$LI1_{okt} = \sum_{i \in I} a_{iot} W_{okt} \qquad \forall o, \forall k, \forall t \tag{A5}$$

$$LI1_{okt} \leq \overline{M} W_{okt} \qquad \forall o, \forall k, \forall t \tag{A6}$$

$$LI1_{okt} \leq \sum_{i \in I} a_{iot} \qquad \forall o, \forall k, \forall t \tag{A7}$$

$$LI1_{okt} \geq \sum_{i \in I} a_{iot} - \overline{M}\left(1 - W_{okt}\right) \qquad \forall o, \forall k, \forall t \tag{A8}$$

The second multiplication of variables is between three variables. Considering the previous linearization, we are able to linearize by creating new variables, defined by (A9), and the set of additional constraints (A10)–(A12).

$$LI2_{okt} = LI1_{okt} C_{kt} \qquad \forall o, \forall k, \forall t \tag{A9}$$

$$LI2_{okt} \leq \overline{M} C_{kt} \qquad \forall o, \forall k, \forall t \tag{A10}$$

$$LI2_{okt} \leq LI1_{okt} \qquad \forall o, \forall k, \forall t \tag{A11}$$

$$LI2_{okt} \geq LI1_{okt} - \overline{M}\left(1 - C_{kt}\right) \qquad \forall o, \forall k, \forall t \tag{A12}$$

The idea is similar to the other multiplication of variables in other terms of the constraint (18), which can be defined by (A13), and the additional set of constraints (A14)–(A16).

$$LI3_{kt} = na_{k,t-1} \overline{W}_{kt} \qquad \forall k, \forall t > 1 \tag{A13}$$

$$LI3_{kt} \leq \overline{M}\ \overline{W}_{kt} \qquad \forall k, \forall t > 1 \tag{A14}$$

$$LI3_{kt} \leq na_{k,t-1} \qquad \forall k, \forall t > 1 \tag{A15}$$

$$LI3_{kt} \geq na_{k,t-1} - \overline{M}\left(1 - \overline{W}_{kt}\right) \qquad \forall k, \forall t > 1 \tag{A16}$$

The last multiplication of variables considers the previous linearization and are defined by (A17), and the set additional of constraints (A18)–(A20).

$$LI4_{kt} = LI3_{kt} C_{kt} \qquad \forall k, \forall t > 1 \tag{A17}$$

$$LI4_{kt} \leq \overline{M} C_{kt} \qquad \forall k, \forall t > 1 \tag{A18}$$

$$LI4_{kt} \leq LI3_{kt} \qquad \forall k, \forall t > 1 \tag{A19}$$

$$LI4_{kt} \geq LI3_{kt} - \overline{M}\left(1 - C_{kt}\right) \qquad \forall k, \forall t > 1 \tag{A20}$$

In all the linearization constraints presented before, $\overline{M}$ is an upper bound for $\sum_{i \in I} a_{iot}$ and na_{kt}.

Considering these linearization to the multiplication of variables, the non-linear constraints (18) can be replaced by the linear constraints (A21), presented as follows:

$$na_{kt} = \sum_{i \in I} a_{ikt} + \sum_{o \in O} \left(LI1_{okt} - LI2_{okt}\right) + LI3_{kt} - LI4_{kt} \qquad \forall k, \forall t \tag{A21}$$

After applying linearization techniques to Model 1, the resulting mixed-integer linear formulation obtained can be represented by the constraints: (2)–(15), (A1)–(A4), (A6)–(A8), (A10)–(A12), (A14)–(A16), (A18)–(A20), (A21) and (20)–(35).

The mathematical Model 2 (see Section 3) also consists of a mixed-integer nonlinear formulation, in which the non-linearity arises from the presence of absolute value in constraints (45). As in the linearization of Model 1, constraints (45) are replaced by the two linear constraints (A22) and (A23) in Model 2.

$$TW_u \geq \sum_{i \in I}(a_{iu} + b_{iu}) - \sum_{i \in I}(a_{iu-1} + b_{iu-1}) \qquad \forall t, u \in U_t, u > 1 \tag{A22}$$

$$TW_u \geq -\sum_{i \in I}(a_{iu} + b_{iu}) + \sum_{i \in I}(a_{iu-1} + b_{iu-1}) \qquad \forall t, u \in U_t, u > 1 \tag{A23}$$

B Variation in the count of setups

In this Appendix, a different variation in the counting of setups is discussed for the coupled bin-packing and lot-sizing problem with sequence-dependent setups. As mentioned before, variations in the counting of setups can be considered, however, it might cause further complications in the mathematical models, showing the highly dependency of the models on the setup counting.

In some industries, such as aluminium, tubular furniture, a slight difference in the cutting process occurs compared to the one previous presented in this paper, which is typically seen in paper industries, and this difference has an impact on the setup counting. In such industries, the cutting process does not need the initial and the final knives in order to trim the edge of each object. Furthermore, the final knife is only added in case the object is not fully used (see Figure 5). In terms of mathematical modeling, Model 1 and Model 2 consider the number of knives needed to cut an object as the number of items cut from it, plus one, due to the consideration of the initial and final knives (see Figure 1). In this new setting, the number of needed knives should be equal to the number of items when the objects is not fully used, i.e., when there is a waste of material and, when the object is fully used (zero waste), the number of needed knives consists of the total number of items minus one (see Figure 5).

In order to model such an approach, an additional variable and constraints are required and modeled, in Model 2 as follows:

Z_u: binary variable indicating if the object is fully used in the cutting process in micro-period u;

$$L - \sum_{i \in I} l_i a_{iu} \leq L(1 - Z_u) \qquad \forall t, u \in U_t \tag{A24}$$

$$L(1 - Z_u) \leq L\left(L - \sum_{i \in I} l_i a_{iu}\right) \qquad \forall t, u \in U_t \tag{A25}$$

The additional variable is linked to the other variables by adding it into the counting of setups in constraints (45), in the following way:

$$TW_u \geq \left| \sum_{i \in I}(a_{iu} + b_{iu}) - Z_u - \left(\sum_{i \in I}(a_{iu-1} + b_{iu-1}) - Z_{u-1}\right) \right| \qquad \forall t, u \in U_t, u > 1 \tag{A26}$$

Constraints (A24) guarantee that when the cutting process does not use the object fully, then the additional variable must be zero and in constraints (A26), the number of setups is equal to the number of cut items. However, when the object is fully cut, constraints (A24) are redundant and constraints (A25) ensure that the additional variable must be equal to 1, which means that no extra knife is needed at the end of the object in the cutting process and one unit must be removed from the number of required setups in constraint (A26).

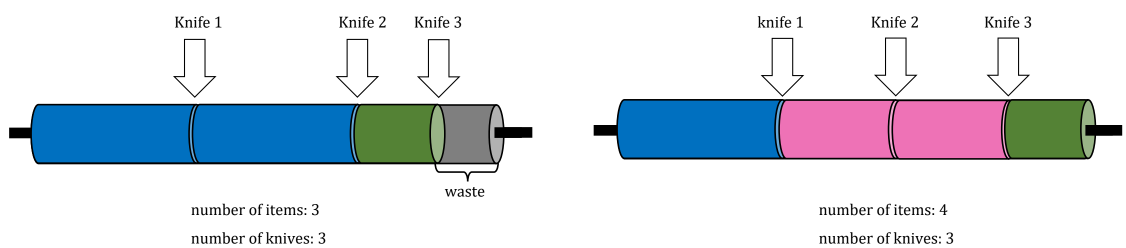

Figure 5: Variations in the counting of knife movements in the scheduling of the cutting operations

References

N. Absi and S. Kedad-Sidhoum. The multi-item capacitated lot-sizing problem with setup times and shortage costs. European Journal of Operational Research, 185(3):1351–1374, 2008.

A. Abuabara and R. Morabito. Cutting optimization of structural tubes to build agricultural light aircrafts. Annals of Operations Research, 169(1):149–165, 2008.

Yossiri Adulyasak, Jean-François Cordeau, and Raf Jans. Benders decomposition for production routing under demand uncertainty. Operations Research, 63(4):851–867, 2015.

A. Aloisio, C. Arbib, and F. Marinelli. Cutting stock with no three parts per pattern: Work-in-process and pattern minimization. Discrete Optimization, 8(2):315–332, 2011.

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

C. Arbib, F. Marinelli, and F. Pezzella. An lp-based tabu search for batch scheduling in a cutting process with finite buffers. International Journal of Production Economics, 136(2):287–296, 2012.

C. Arbib, F. Marinelli, and P. Ventura. One-dimensional cutting stock with a limited number of open stacks: bounds and solutions from a new integer linear programming model. International Transactions in Operational Research, 23(1-2):47–63, 2016.

M. Armbruster. A solution procedure for a pattern sequencing problem as part of a one-dimensional cutting stock problem in the steel industry. European Journal of Operational Research, 141(2):328–340, 2002.

T. A. Baldo, M. O. Santos, B. Almada-Lobo, and R. Morabito. An optimization approach for the lot sizing and scheduling problem in the brewery industry. Computers & Industrial Engineering, 72:58–71, 2014.

T. Bektaş and L. Gouveia. Requiem for the miller-tucker-zemlin subtour elimination constraints? European Journal of Operational Research, 236(3):820–832, 2014.

J. F. Benders. Partitioning procedures for solving mixed-variables programming problems. Numerische Mathematik, 4:238–252, 1962.

J. A. Bennell, L. S. Lee, and C. N. Potts. A genetic algorithm for two-dimensional bin packing with due dates. International Journal of Production Economics, 145(2):547–560, 2013.

M. Bodur, S. Dash, O. Günlük, and J. Luedtke. Strengthened benders cuts for stochastic integer programs with continuous recourse. INFORMS Journal on Computing, 29(1):77–91, 2017.

P. Bonami, D. Salvagnin, and A. Tramontani. Implementing Automatic Benders Decomposition in a Modern MIP Solver. http://www.optimization-online.org/DB_FILE/2019/12/7506.pdf, 2019.

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

N. Braga, C. Alves, and J. Valério de Carvalho. Exact solution of combined cutting stock and scheduling problems. In Computational Management Science, pages 131–139. Springer International Publishing, 2016.

G. Carpaneto, M. Dell'Amico, and P. Toth. Exact solution of large-scale, asymmetric traveling salesman problems. ACM Transactions on Mathematical Software (TOMS), 21(4):394–409, 1995.

A. R. Clark, R. Morabito, and E. A. V. Toso. Production setup-sequencing and lot-sizing at an animal nutrition plant through atsp subtour elimination and patching. Journal of Scheduling, 13(2):111–121, 2010.

M. Conforti and L. A. Wolsey. Facet separation with one linear program. Mathematical Programming, 178 (1-2):361–380, 2019.

K. Copil, M. Wörbelauer, H. Meyr, and H. Tempelmeier. Simultaneous lotsizing and scheduling problems: a classification and review of models. OR Spectrum, 39(1):1–64, 2017.

G. Dantzig, R. Fulkerson, and S. Johnson. Solution of a large-scale traveling-salesman problem. Journal of the Operations Research Society of America, 2(4):393–410, 1954.

S. A. de Araujo and A. Clark. A priori reformulations for joint rolling-horizon scheduling of materials processing and lot-sizing problem. Computers & Industrial Engineering, 65:557–585, 2013.

M. Delorme, M. Iori, and S. Martello. Bin packing and cutting stock problems: Mathematical models and exact algorithms. European Journal of Operational Research, 255(1):1–20, 2016.

M. Desrochers and G. Laporte. Improvements and extensions to the miller-tucker-zemlin subtour elimination constraints. Operations Research Letters, 10(1):27–36, 1991.

A. Drexl and A. Kimms. Lot sizing and scheduling - survey and extensions. European Journal of Operational Research, 99(2):221–235, 1997.

H. Dyckhoff. Cutting and packing a typology of cutting and packing problems. European Journal of Operational Research, 44(2):145–159, 1990.

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

M. Fischetti. A note on the selection of benders' cuts. Mathematical Programming, 124(3):175–182, 2010.

M. Fischetti, I. Ljubić, and M. Sinnl. Benders decomposition without separability: A computational study for capacitated facility location problems. European Journal of Operational Research, 253(3):557–569, 2016.

H. Foerster and G. Wäscher. Simulated annealing for order spread minimization in sequencing cutting patterns. European Journal of Operational Research, 110(2):272–281, 1998.

T. Gau and G. Wäscher. Cutgen1: A problem generator for the standard one-dimensional cutting stock problem. European Journal of Operational Research, 84(3):572–579, 1995.

N. F. Giannelos and M. C. Georgiadis. Scheduling of cutting-stock processes on multiple parallel machines. Chemical Engineering Research and Design, 79(7):747–753, 2001.

P. C. Gilmore and R. E. Gomory. A linear programming approach to the cutting-stock problem. Operations Research, 9(6):849–859, 1961.

P. C. Gilmore and R. E. Gomory. A linear programming approach to the cutting stock problema - part ii. Operations Research, 11(6):863–888, 1963.

C. H. Glock, E. H. Grosse, and Jörg M. Ries. The lot sizing problem: A tertiary study. International Journal of Production Economics, 155:39–51, 2014.

L. Guimarães, D. Klabjan, and B. Almada-Lobo. Modeling lotsizing and scheduling problems with sequence dependent setups. European Journal of Operational Research, 239(3):644–662, 2014.

E. Hans and S. van de Velde. The lot sizing and scheduling of sand casting operations. International Journal of Production Research, 49(9):2481–2499, 2011.

IBM. IBM CPLEX Optimizer: CPLEX User's Manual (2019). https://www.ibm.com/analytics/cplex-optimizer, 2019.

R. Jans and Z. Degraeve. Improved lower bounds for the capacitated lot sizing problem with setup times. Operations Research Letters, 32(2):185–195, 2004.

R. Jans and Z. Degraeve. Modeling industrial lot sizing problems: a review. International Journal of Production Research, 46(6):1619–1643, 2008.

R. E. Johnston and E. Sadinlija. A new model for complete solutions to one-dimensional cutting stock problems. European Journal of Operational Research, 153(1):176–183, 2004.

L. V. Kantorovich. Mathematical methods of organizing and planning production. Management Science, 6(4): 366–422, 1960.

B. Karimi, S. M. T. Fatemi Ghomi, and J. M. Wilson. The capacitated lot sizing problem: a review of models and algorithms. Omega, 31(5):365–378, 2003.

S. Li. Multi-job cutting stock problem with due dates and release dates. Journal of the Operational Research Society, 47(4):490–510, 1996.

O. B. G. Madsen. An application of travelling-salesman routines to solve pattern-allocation problems in the glass industry. The Journal of the Operational Research Society, 39(3):249–256, 1988.

T. L. Magnanti and R. T. Wong. Accelerating benders decomposition: Algorithmic enhancement and model selection criteria. Operations Research, 29(3):464–484, 1981.

F. Marinelli, M. E. Nenni, and A. Sforza. Capacitated lot sizing and scheduling with parallel machines and shared buffers: A case study in a packaging company. Annals of Operations Research, 150(1):177–192, 2007.

K. P. Martínez, E. A. V. Toso, and R. Morabito. Production planning in the molded pulp packaging industry. Computers & Industrial Engineering, 98:554–566, 2016.

K. Matsumoto, S. Umetani, and H. Nagamochi. On the one-dimensional stock cutting problem in the paper tube industry. Journal of Scheduling, 14(3):281–290, 2011.

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

G. M. Melega, S. A. de Araujo, and R. Morabito. Mathematical model and solution approaches for integrated lot-sizing, scheduling and cutting stock problems. Annals of Operations Research, 2020.

H. Meyr. Simultaneous lotsizing and scheduling by combining local search with dual reoptimization. European Journal of Operational Research, 120(2):311–326, 2000.

C. E. Miller, A. W. Tucker, and R. A. Zemlin. Integer programming formulation of traveling salesman problems. Journal of the ACM, 7(4):326–329, 1960.

Y. Pochet and L. A. Wolsey. Lot-size models with backlogging: Strong reformulations and cutting planes. Mathematical Programming, 40(1-3):317–335, 1988.

R. Rahmaniani, T. G. Crainic, M. Gendreau, and W. Rei. The benders decomposition algorithm: A literature review. European Journal of Operational Research, 259(3):801–817, 2017.

H. Reinertsen and T. W.M. Vossen. The one-dimensional cutting stock problem with due dates. European Journal of Operational Research, 201(3):701–711, 2010.

F. Rinaldi and A. Franz. A two-dimensional strip cutting problem with sequencing constraint. European Journal of Operational Research, 183(3):1371–1384, 2007.

G. K. D. Saharidis and M. G. Ierapetritou. Improving benders decomposition using maximum feasible subsystem (mfs) cut generation strategy. Computers & Chemical Engineering, 34(8):1237–1245, 2010.

D. J. Thomas and P. M. Griffin. Coordinated supply chain management. European Journal of Operational Research, 94(1):1–15, 1996.

A. Toscano, D. Ferreira, and R. Morabito. Formulation and mip-heuristics for the lot sizing and scheduling problem with temporal cleanings. Computers & Chemical Engineering, 2020.

E. A. V. Toso, R. Morabito, and A. R. Clark. Lot sizing and sequencing optimisation at an animal-feed plant. Computers & Industrial Engineering, 57(3):813–821, 2009.

W. W. Trigeiro, L. J. Thomas, and J. O. McClain. Capacitated lot sizing with setup times. Management Science, 35(3):353–366, 1989.

J. M. Valério de Carvalho. Exact solution of bin-packing problems using column generation and branch-and-bound. Annals of Operations Research, 86(0):629–659, 1999.

J. M. Valério de Carvalho. Lp models for bin packing and cutting stock problems. European Journal of Operational Research, 141(2):253–273, 2002.

S. K. Vickery and R. E. Markland. Multi-stage lot-sizing in a serial production system. International Journal of Production Research, 24(3):517–534, 1986.

G. Wäscher, H. Haußner, and H. Schumann. An improved typology of cutting and packing problems. European Journal of Operational Research, 183(3):1109–1130, 2007.

David A. Wuttke and H. Sebastian Heese. Two-dimensional cutting stock problem with sequence dependent setup times. European Journal of Operational Research, 265(1):303–315, 2018.

H. H. Yanasse and M. J. P. Lamosa. An integrated cutting stock and sequencing problem. European Journal of Operational Research, 183(3):1353–1370, 2007.

B. J. Yuen. Heuristics for sequencing cutting patterns. European Journal of Operational Research, 55(2): 183–190, 1991.

B. J. Yuen. Improved heuristics for sequencing cutting patterns. European Journal of Operational Research, 87(1):57–64, 1995.