Ann Oper Res (2016) 238:497–520
DOI 10.1007/s10479-015-2103-2

Mathematical models and a heuristic method for the multiperiod one-dimensional cutting stock problem

Kelly Cristina Poldi[1] · **Silvio Alexandre de Araujo**[2]

Published online: 18 January 2016

Abstract The multiperiod cutting stock problem arises in the production planning and programming of many industries that have the cutting process as an important stage. Ordered items are required in different periods of a finite planning horizon. It is possible to bring forward or not the production of items. Unused inventory in a certain period becomes available for the next period, all together with new inventory which may come to be acquired in the market. Based on mixed integer optimization models from the literature, extensions are proposed to deal with the multiperiod case and a residual heuristic is used. Computational experiments showed that effective gains can be obtained when comparing multiperiod models with the lot for lot solution, which is typically used in practice. Most of the instances are solved satisfactorily with a high performance optimization package and the heuristic method is used for solving the hard instances.

Keywords Cutting stock problem · Multiperiod · Mathematical models · Residual heuristic

1 Introduction

The cutting stock problem consists of finding an optimized way of cutting objects in stock of known dimension into smaller items in order to meet a given demand. In general, the objective to be optimized is related to the minimization of the waste of material. The multiperiod cutting stock problem consists basically of solving, in each period of a finite planning horizon, a cutting stock problem, to meet demand of items in the several periods of the planning

✉ Kelly Cristina Poldi
kellypoldi@ime.unicamp.br

Silvio Alexandre de Araujo
saraujo@ibilce.unesp.br

1 Instituto de Matemática, Estatística e Computação Científica-IMECC, Universidade Estadual de Campinas-UNICAMP, Rua Sergio Buarque de Holanda, 651, Campinas, SP 13083-859, Brazil

2 Departamento de Matemática Aplicada-DMAp, Universidade Estadual Paulista-UNESP, Rua Cristóvão Colombo, 2265, São José do Rio Preto, SP 15054-000, Brazil

horizon. However, production of some items might be brought forward or not. This allows new combinations to be considered, i.e., an item that has no demand in a given period might be brought forward from the next period, for example, if its combination with other demanded items decreases the waste. Stock objects not used in a period become available for the next period, together with the new stock for that period (such objects may be acquired in the market or produced in the same factory, as in the case study presented by Poltroniere et al. (2008), in a paper factory). The amount of objects in stock (acquired or produced) is considered, in this work, as input data. The objective function to be minimized combines the material waste, holding costs of items brought forward and stock object holding costs.

Due to current economic circumstances, industries try to make their production process more efficient and this stimulates academic research on optimization models to control industry's production planning as a whole. Therefore, coupled problems have been studied. In the coupled context, the cutting stock problem arises as a subproblem which has to be solved integrated with others optimization problems in industry. Common examples in different industries are cutting stock problems coupled with lot sizing problems (Trigeiro et al. 1989). Typically, lots are defined and a cutting stock problem is solved independently for each lot, so the waste in the cutting process does not interfere in the lot size determination. Taking the coupled lot sizing and cutting stock decisions, it is possible to bring forward the production of some items and to have better cutting patterns (as a larger variety of pieces possibly allows better combinations).

Gramani and França (2006) proposed a mathematical model for coupling lot sizing and two-dimensional cutting stock problems based on a case study of a furniture factory. The aim is to determine which and how many final products (desks, shelves, wardrobes, etc.) should be produced in a given period of a finite planning horizon. It consists of deciding the amount of final products such that it minimizes production, setup and holding costs (lot sizing problem) and also the number of stock objects cut into items to build the final products to meet the demand (cutting stock problem). Note that an optimal solution for the coupled problem may contain non-optimal solutions to the cutting stock problem and the lot sizing problem when considered separately. Later on, in 2009, Gramani et al. (2009) presented an integrated lot sizing and cutting stock model that incorporates the conjecture that it is more advantageous to bring forward the production of certain lots of final products. In this paper the importance of multiperiod cutting stock problem can be seen. This importance is confirmed in Gramani et al. (2011) in which the authors address the integrated problem by solving its linear relaxation using the column generation technique. They were able to improve the solution by up to 12.7 % when compared to the solution from the decomposed model. The coupled lot sizing and cutting stock problem in the furniture industry was also addressed by Alem and Morabito (2012, 2013) and Vanzela et al. (2013).

Other coupled cutting stock and lot sizing problems were studied by Arbib and Marinelli (2005), Poltroniere et al. (2008), Nonas and Thorstenson (2008), Malik et al. (2009), Suliman (2012) and Silva et al. (2014). Cutting stock problems with due dates have been considered by Li (1996), Johnston and Sadinlija (2004), Reinertsen and Vossen (2010) and Arbib and Marinelli (2014).

In particular, this study was motivated by a production planning problem that arises in a paper factory, which is well defined in Poltroniere et al. (2008). First of all, a lot sizing problem is considered in order to decide which should be the weight of jumbos (large reels) to be produced in each period of a planning horizon. After being made, the jumbos are cut into smaller reels of given widths (which may be cut in sequence into rectangles) in order to meet a given demand in such a way as to minimize the waste. In other words, it is an usual cutting stock problem. Typically, the problem of jumbo production is empirically solved by expert production managers, who focus mainly on setup minimization. Therefore, lots of jumbos are produced without paying attention to the next production stage of cutting the jumbos which waste depends on the jumbos (widths and quantities) previously made. Of course, the best widths and quantities of jumbos in terms of the cutting problem can introduce high setups when producing them. So, the planning decisions consist of choosing which jumbo reels (defined by their length and thickness of the paper) and how many (lot size) jumbo reels should be produced in each period of time, in order to meet demand, avoiding holding costs and minimizing the waste of material. Therefore, these two problems, i. e., the lot sizing problem and the cutting stock problem, are interdependent and should be solved in an integrated manner. The authors formulated two mathematical models and developed a method to solve the problem.

In most papers that consider coupled models for lot sizing and cutting stock problems including the paper from Poltroniere et al. (2008), the multiperiod cutting stock problem can be identified as a subproblem and this shows the importance of solving the multiperiod cutting stock problem efficiently.

In this context, this paper makes several contributions. First, it generalizes, for the multiperiod cutting stock problem, two mathematical models that were originally proposed for the single period classical cutting stock problem. To the best of our knowledge, no paper in the literature on multiperiod cutting stock problem has either so far adopted the arc flow model proposed in this paper. Second, the generalized models include different types of objects in stock, and the stock availability of objects is considered as a parameter, and also as a decision variable. Third, a residual heuristic adapted from Poldi and Arenales (2010) is presented for solving difficult instances. Forth, extensive computational results are presented showing the quality of the computational package in solving the proposed mathematical models and the quality of the proposed heuristic. It is worth noticing that the proposed mathematical models have direct practical application in different industries in which the cut products are the same as the ordered products, such as: pre-cast roof-slabs and industries that cut items for other industries, such as, in the furniture sector. Moreover, the addressed problem arises as a subproblem in industrial processes where the lot sizing problem is integrated with the cutting stock problem. The amount of research in recent years indicates the relevance of these integrated lot sizing and cutting stock problems in various industrial settings, such as, furniture and paper industries, among others.

In Sect. 2, the mathematical models that consist of generalizations of the model proposed by Gilmore and Gomory (1963) and the arc flow model (Valério de Carvalho 1999, 2002) are presented. In Sect. 3, a small example that shows the importance of considering the multiperiod problem is given. The residual heuristic is described in Sect. 4 and Sect. 5 presents, presents the computational results. Finally, in Sect. 6, the conclusions are drawn and suggestions for future research are made.

2 Problem definition and mathematical models: multiperiod cutting stock problem

Here is a short description of the multiperiod one-dimensional cutting stock problem with several types of stock objects.

Assume there is a finite planning horizon divided into T periods, $t = 1, \ldots, T$. One period can be a working-hour, a working-week or a working-month. Assume, also, there are available K types of objects (bars, reels, rolls, etc.) of given length $L_k, k = 1, \ldots, K$, each type available in quantity e_{kt}, $k = 1, \ldots, K$ in each period t of the planning horizon, $t = 1, \ldots, T$. In each period t, a set of items of given length l_i, $i = 1, \ldots, m$, has to be cut to meet demand d_{it}, $i = 1, \ldots, m$, $t = 1, \ldots, T$. The multiperiod cutting stock problem consists of producing the demanded items by cutting the stock objects available in each period of the planning horizon, in such a way that clients' demand is met and a certain objective function is minimized, e.g., minimize material waste and holding costs.

2.1 Generalized Gilmore and Gomory's model (GGG)

We present a generalization of the model proposed by Gilmore and Gomory (1961, 1963, 1965) for the cutting stock problem, to deal with the multiperiod cutting stock problem (Poldi and Arenales 2010). Consider the following:

Indices:

- $t = 1, \ldots, T$: number of periods in the planning horizon;
- $k = 1, \ldots, K$: number of types of objects available in stock;
- $j = 1, \ldots, N_k$: N_k is the number of cutting patterns for stock object type k, $k = 1, \ldots, K$;
- $i = 1, \ldots, m$: number of types of ordered items.

Data:

- L_k: length of stock object type k, $k = 1, \ldots, K$;
- e_{kt}: stock availability of object type k in period t, $k = 1, \ldots, K$, $t = 1, \ldots, T$;
- l_i: length of item type i, $i = 1, \ldots, m$;
- d_{it}: demand of item type i in period t, $i = 1, \ldots, m$, $t = 1, \ldots, T$ ($\boldsymbol{d}_t$: vector with components d_{it}).

Parameters:

- c_{jkt}: cost of cutting a stock object type k according to the jth cutting pattern in period t, $j = 1, \ldots, N_k$, $k = 1, \ldots, K$, $t = 1, \ldots, T$;
- c^r_{it}: holding cost for item type i in period t, $i = 1, \ldots, m$, $t = 1, \ldots, T$;
- c^s_{kt}: holding cost for stock object type k in period t, $k = 1, \ldots, K$, $t = 1, \ldots, T$.

Decision variables:

- x_{jkt}: number of stock objects cut according to cutting pattern j of object type k in period t;
- r_{it}: number of items type i which are brought forward to period t ($\boldsymbol{r}_t$: vector with components r_{it});
- s_{kt}: number of objects type k not used in period t, and available in period $t+1$.

Definition 1 We call a **cutting pattern** $\mathbf{a}_{kt}$ the way a stock object is cut to produce the ordered items. To each cutting pattern there is an m-dimensional vector which represents the produced items,

$$\mathbf{a}_{kt} = (a_{1kt}, a_{2kt}, \ldots, a_{mkt})^{\mathrm{T}}$$

where a_{ikt} is the amount of items type i, in a cutting pattern for object type k, in period t.

In one-dimensional cutting problems, a cutting pattern $\mathbf{a}_{kt}$ has to satisfy the capacity constraint of a knapsack problem:

$$l_1\, a_{1kt} + l_2\, a_{2kt} + \cdots + l_m \quad a_{mkt} \leq L_k \tag{1}$$

$$0 \leq \; a_{ikt} \leq d_{it}, \quad \text{and integers}, i = 1, \ldots, m, k = 1, \ldots, K, t = 1, \ldots, T. \tag{2}$$

Therefore, let us consider a_{ijkt} the amount of items type i, in the cutting pattern j for object type k, in period t. Consequently, let $\mathbf{a}_{jkt}$ be an m-dimensional vector with elements a_{ijkt}.

Mathematical model (GGG):

$$\text{Min} \quad \sum_{t=1}^{T}\left(\sum_{j=1}^{N_1} c_{j1t}x_{j1t}+\sum_{j=1}^{N_2} c_{j2t}x_{j2t}+\cdots+\sum_{j=1}^{N_K} c_{jKt}x_{jKt}+\sum_{i=1}^{m} c^r_{it}r_{it}+\sum_{k=1}^{K} c^s_{kt}s_{kt}\right) \tag{3}$$

$$\text{Subject to:} \quad \sum_{j=1}^{N_1}\mathbf{a}_{j1t}x_{j1t}+\sum_{j=1}^{N_2}\mathbf{a}_{j2t}x_{j2t}+\cdots+\sum_{j=1}^{N_K}\mathbf{a}_{jKt}x_{jKt}+\mathbf{r}_{t-1}-\mathbf{r}_t=\mathbf{d}_t,$$
$$t=1,\ldots,T, \tag{4}$$

$$\sum_{j=1}^{N_k} x_{jkt}-s_{k,t-1}+s_{kt}=e_{kt}, \quad k=1,\ldots,K, t=1,\ldots,T, \tag{5}$$

$$x_{jkt}\in Z_+,\ r_{it}\in R_+,\ s_{kt}\in R_+, \quad j=1,\ldots,N_k, k=1,\ldots,K, t=1,\ldots,T. \tag{6}$$

The objective function (3) minimizes the total waste of material, in all periods and the holding costs of items and stock objects. The cost of a column that represents a cutting pattern is considered equal to the waste in the corresponding cutting pattern: $c_{jkt} = L_k - \sum_{i=1}^{m} l_i \ \alpha_{ijkt}$, i. e. the waste in the jth cutting pattern for object type k. The cost of bringing forward the production of an item from one period to the previous period is given by: $c^r_{it} = \alpha l_i$. The holding cost for stock objects that are not used in a certain period and will become available in the next period is given by: $c^s_{kt} = \beta L_k$. Parameters α and β were defined to study the influence of holding costs in the proposed model. One can fix any value to c^r_{it} and c^s_{kt}, i. e. these parameters can be tuned by the user, according to the real needs of the industry.

Bringing forward the cut of some items may increase the items' holding cost (c^r_{it}), on the other hand, it may allow a better match of items, which minimizes total waste. The constraints (4) make sure that original demand is met and (5) that stock availability of each type of object will not be exceeded. Stock objects not used in a period t became available in period $t+1$, with a "penalty", i.e., the holding cost c^s_{kt}. When considering null holding costs, there is a tendency to bring forward the production of items, which is limited by the stock objects availability.

The GGG model (3)–(6) can be used to solve either one-dimensional or two-dimensional cutting stock problems, the only difference is how the cutting pattern is built. As in the multiperiod cutting stock problem, the cutting stock problem with several types of stock objects (Gilmore and Gomory 1963) also allows better combinations of items that can lead to smaller waste.

2.2 Generalized arc flow model (GAF)

Valério de Carvalho (1999, 2002) considered the bin-packing problem where variables that correspond to items of a given type are indexed by the physical position they occupy inside the large objects, i. e., a variable represents the placement of an item at a given distance from the border of the roll. There are other papers in the literature that consider mathematical models based on position-indexed (e.g. Beasley 1985) for the two-dimensional non-guillotine cutting stock problem; Pinho de Sousa and Wolsey (1992) for the scheduling problem (time-indexed formulation)).

Based on the position-indexed principle, Valério de Carvalho (1999, 2002) presents an arc flow formulation. Given the data described in the previous section, finding a valid cutting pattern in the proposed model is equivalent to finding a path in the acyclic oriented graph $G = (V, A)$, with a set of vertices $V = \{0, 1, \ldots, L_{max}\}$, where $L_{max} = max\{L_k\}$ is the length of the largest object; and the set of arcs A is defined as: there exists a directed arc between two vertices if there is an item of the corresponding size ($A = \{(j, h) : 0 \leq j < h \leq L_{max} \text{ and } h - j = l_i \text{ for every } 1 \leq i \leq m\}$). Furthermore, there are additional losses arcs $(j, j+1)$, $j = 0, 1, \ldots, L_{max} - 1$. There a cutting pattern in a single object of length L_k if there is a path between vertices 0 and L_k. The lengths of the arcs that constitute the path define the item sizes to be cut. In the same set of vertices, consider directed arcs from vertex L_k to vertex 0, if there is an object of length L_k, $k = 1, \ldots, K$.

Hereafter we present the generalized arc flow (GAF) formulation for the multiperiod cutting stock problem. Consider the following additional variables:

- f_{kt}: number of object of length L_k cut in period t (can be seen as a feedback arc, from vertex L_k to vertex 0);
- z_{jht}: number of items of size $(h - j)$ placed in any object at a distance j from the beginning of the object, considering all the cutting patterns cut in period t.

Mathematical model (GAF):

$$\text{Min} \left(\sum_{k=1}^{K} \sum_{t=1}^{T} L_k f_{kt} - \sum_{i=1}^{m} \sum_{t=1}^{T} \sum_{(j,j+l_i)\in A} l_i z_{j,j+l_i,t} \right) + \sum_{i=1}^{m} \sum_{t=1}^{T} c_{it}^{r} r_{it} + \sum_{k=1}^{K} \sum_{t=1}^{T} c_{kt}^{s} s_{kt} \tag{7}$$

Subject to:

$$\sum_{(j,0)\in A} z_{j0t} - \sum_{(0,g)\in A} z_{0gt} = -\sum_{k=1}^{K} f_{kt}, \quad t = 1, \ldots, T, \tag{8}$$

$$\sum_{(j,h)\in A} z_{jht} - \sum_{(h,g)\in A} z_{hgt} = 0, h = 1, \ldots, L_{max} - 1 (h \neq L_k, \forall k) \quad t = 1, \ldots, T, \tag{9}$$

$$\sum_{(j,L_k)\in A} z_{jL_kt} + \sum_{(L_k,h)\in A} z_{L_kht} = -f_{kt}, \quad k = 1, \ldots, K, \quad t = 1, \ldots, T, \tag{10}$$

$$\sum_{(j,j+l_i)\in A} z_{j,j+l_i,t} + r_{i,t-1} - r_{it} = d_{it}, \quad k = 1, \ldots, K, \quad t = 1, \ldots, T, \tag{11}$$

$$f_{kt} - s_{k,t-1} + s_{kt} = e_{kt}, \quad k = 1, \ldots, K, \quad t = 1, \ldots, T, \tag{12}$$

$$\begin{aligned} & f_{kt} \in Z_+, \; s_{kt} \in R_+, r_{it} \in R_+, z_{jht} \in Z_+, \\ & k = 1, \ldots, K, t = 1, \ldots, T, \quad i = 1, \ldots, m, (j, h) \quad \in A. \end{aligned} \tag{13}$$

The objective function (7) minimizes the total waste of material (first parcel) and the holding costs of items and stock objects. Constraints (8)–(10) are the flow conservation constraints. Constraints (11) and (12) are equivalent to (4) and (5), respectively; (11) guarantee that the demand is met and (12) that the stock availability is respected. As shown in the computational results, the GAF model enables the solving of medium-sized instances of the integer problem, using a standard MIP-solver.

It is important to note that, considering the single period problem, Valério de Carvalho (1999) proved that the linear programming arc flow model is equivalent to the classical Gilmore and Gomory's model, and hence the linear programming bounds are the same. Next we extend this proposition and proof for the multiperiod case.

Proposition 1 *The linear programming (LP) relaxations of the models* (3)–(6) *and* (7)–(13) *are equivalent and hence the linear programming bounds are equal.*

Proof Considering the LP relaxation of formulation (7)–(13), it is possible to obtain an equivalent formulation by applying a Dantzig–Wolfe decomposition, keeping constraints (11) and (12) in the master problem, and constraints (8)–(10) in the subproblem.

The set of constraints (8)–(10) and the non-negativity constraints without the integrality requirements define a homogeneous system that corresponds to a set X. According to Minkowski's theorem (see Nemhauser and Wolsey 1988), any point x of a nonempty polyhedron X can be expressed as a convex combination of the extreme points of X plus a nonnegative linear combination of the extreme rays of X.

The set X has only one extreme point, the solution with null flow in every period, and all other valid flows can be expressed, in each specific period, as non-negative linear combinations of circulation flows along cycles. Each cycle will correspond to a valid cutting pattern and is defined, in a given period, by a unique stock object and a set of items. Thus, in a given period, the cycles start at node 0, include a set of item arcs, and eventually loss arcs, and return to node 0, through a feedback arc, which corresponds to a stock object.

The circulation flows along each cycle cannot be expressed as non-negative linear combinations of other circulation flows, and are, therefore, extremal. The extremal flows are not bounded and each set of cycles on the planning horizon will correspond to an extreme ray. Therefore, the corresponding polyhedron has a single extreme point, the null solution, and a finite set of extreme rays, which are the directed paths, each corresponding to a valid cutting pattern for each period. As a consequence, the reformulated problem will not have a convexity constraint.

The subproblem will only generate extreme rays to the master problem. Let Ω be the set of feasible cycles. For each different capacity L_k, there will be a set of valid cutting patterns that can be used in a given period t. Let Ω^k be the set of feasible cycles for object type $k, k = 1, \ldots, K$. The sets Ω^k are mutually disjoint and $\Omega = \bigcup_k \Omega^k$.

Each cycle $r \in \Omega^k$ in a given period t can be described using the binary variables z^r_{jht} and f^r_{kt} that take the value 1, if the corresponding arc is included in the cycle of that period. For a period t a column in the master problem can be defined by $(\tilde{a}^r_{kt}, \tilde{b}^r_{kt})$, where $\tilde{a}^r_{kt} = (a^r_{1kt}, \ldots, a^r_{ikt}, \ldots, a^r_{mkt})$ is the vector that defines the number of items for each order and $\tilde{b}^r_{kt} = \tilde{e}_{kt} = (0, \ldots, 1, \ldots, 0) \in N^k$ is the kth unit vector, with a 1 in position k, that identifies the stock object where the items are cut in the fixed period t. The coefficients of these columns, a^r_{ikt}, are expressed in terms of the decision variables of the subproblem, z^r_{jht}, which correspond to the arcs (j, h) that take the value 1 in the shortest path subproblem between nodes 0 and $L_{k,}$ in period t:

$$a^r_{ikt} = \sum_{(j,h):h-j=l_i} z^r_{jht} \quad i = 1, \ldots, m, \tag{14}$$

while the element of the vector $\tilde{b}^r_{kt}$ that is equal to 1 is the one that matches f^r_{kt}.

Let μ^r_{kt} be the variables of the master problem, which mean the number of times the cutting pattern r is cut in object type k in period t. The replacement of these cutting patterns in (7), (11) and (12) provides a model which is equivalent to model (3)–(6).

Let *LGGG* and *LGAF* be the lower bounds provided by the classical model and by the arc flow model, respectively. From the equivalence, it follows that $LGGG = LGAF$. □

2.2.1 Reduction criteria

The arc flow model presents many symmetric solutions, i. e., several alternative solutions that correspond to the same set of cutting patterns. For example, if we consider an object with length 7, to cut two items of size $l_1 = 2$ and one item with size $l_2 = 3$ has an equivalent solution given by cutting one item with size $l_1 = 2$, one with size $l_2 = 3$ e one with size $l_1 = 2$. So, Valério de Carvalho (1999, 2002) presented some reduction criteria that allow for eliminating some arcs and therefore reducing the number of symmetric solutions, without eliminating any valid cutting patterns. This procedure reduces the number of variables (columns) in the problem.

We use two of these criteria in our generalized model. The first one consists in ordering the items according to their size and including them in a cutting pattern in decreasing order, i. e., an item of size i_1 can be placed only after another item of size i_2 with $l_{i1} \leq l_{i2}$, or at the beginning of the object. So, in our previous example, the items with sizes $l_2 = 3$ and $l_1 = 2$ would be placed in the object in the following order: $l_2 = 3, l_1 = 2$ and $l_1 = 2$.

The second reduction criterion used in our generalized model is that the first loss arc is inserted in our graph at a distance from the beginning of the object that is equal to the size of the smallest item. The intention is to put the losses arcs at the end of the object.

These two criteria were chosen because they can easily be adapted for the multiperiod case. Another criterion proposed by Valério de Carvalho (1999, 2002) is related to the demand of the items and does not allow a cutting pattern with excess of production. However, this criterion cannot be used in the multiperiod problem because the excess of production in a period is allowed as an inventory to meet future demand.

Some computational tests were carried out in order to check the effect of these reduction criteria for solving the problems and the conclusion is that, on average, they reduced the solution time in 90 % for solving the mixed-integer problems.

3 Example

In this section an example of a multiperiod cutting stock problem is presented. Two different solutions for the example are shown: a lot for lot solution that considers solving one single period cutting stock problem for each period and a multiperiod cutting stock problem solution. These solution exemplify the importance of considering multiperiod decisions.

Assume there are $T = 3$ periods and, for each of them, there are $K = 2$ types of stock objects, the specifications of which (length and availability) are given in Table 1 and, $m = 3$ types of ordered items, the length and demand of which are given in Table 2. Item holding costs (c^r_{it}) and stock objects holding costs (c^s_{kt}) were considered null for this example.

Multiperiod linear solution:
Table 3 presents the solution for the linear relaxation of the problem (3)–(6).

In the first period, 9.1818 stock objects type 1 are used and its stock availability is 10, so, the remaining amount 0.8182 is added to the stock object type 1 availability for the next period, that is, the second period. Stock object type 2 availability (that is 7) is completely used in the first period, so, there is no remaining object type 2 for the next period. Production

Table 1 Stock objects data

Id	Length	Stock availability		
		1st period	2nd period	3rd period
1	$L_1 = 161$	$e_{11} = 10$	$e_{12} = 5$	$e_{13} = 4$
2	$L_2 = 234$	$e_{21} = 7$	$e_{22} = 3$	$e_{23} = 3$

Table 2 Demanded items data

Item	Length	Demand		
		1st period	2nd period	3rd period
1	$l_1 = 31$	$d_{11} = 35$	$d_{12} = 20$	$d_{13} = 1$
2	$l_2 = 42$	$d_{21} = 20$	$d_{22} = 10$	$d_{23} = 10$
3	$l_3 = 57$	$d_{31} = 20$	$d_{32} = 5$	$d_{33} = 10$

Table 3 Multiperiod solution

Object id	x	Cutting pattern	Waste
1st period			
1	9.1818	(2 1 1)	0
2	5.7272	(3 2 1)	0
2	1.2727	(0 0 4)	6
Total waste of material in the first period = 7.6363			
2nd period			
1	5.7272	(2 1 1)	0
2	3.0000	(3 2 1)	0
Total waste of material in the second period = 0			
3rd period			
1	1.9090	(0 4 1)	9
2	1.0909	(0 0 4)	6
Total waste of material in the third period = 23.7272			

of items type 1 and 2 was brought forward (0.5454 units of type 1 and 0.6363 units of type 2).

In the second period, the stock object type 1 availability is $e_{12} = 5$ plus the remaining amount from the previous period: 0.8182, i.e., $e_{12} = 5 + 0.8182 = 5.8182$. Stock object type 2 availability remains $e_{22} = 3$ because no object type 2 remained unused from the first period. Production of three items was brought forward from period 3 to period 2: item type 1 (1 unit), item type 2 (2.3636 units) and item type 3 (3.7272 units).

In the third period, stock object type 1 availability is updated (the remaining amount from the previous period is added) to $e_{13} = 4 + 0.091 = 4.091$. Stock object type 2 availability is not changed because no object of this type remained from the previous period. Production of all items is finalized. Stock objects not used are: 4.091 objects type 1 and no stock object type 2 was left unused.

The total waste of material given by the linear relaxation of the multiperiod model is, 7.6363 + 0 + 23.7272 = **31.3636**.

Table 4 Lot for lot solution

Object id	x	Cutting pattern	Waste
1st period			
1	10.0000	(2 1 1)	0
2	5.0000	(3 2 1)	0
2	1.2500	(0 0 4)	6
Total waste of material in the first period = 7.5000			
2nd period			
1	0.4166	(1 3 0)	4
1	1.2500	(2 2 1)	0
1	1.1666	(5 0 0)	6
2	3.7500	(3 2 1)	0
Total waste of material in the second period = 8.6666			
3rd period			
1	1.0000	(1 3 0)	4
1	1.1818	(0 1 2)	5
2	1.4545	(0 4 1)	9
2	1.5454	(0 0 4)	6
Total waste of material in the third period = 32.2727			

Lot for lot linear solution:

Still regarding the data in Tables 1 and 2, Table 4 presents the solution for the lot for lot problem, i. e., demand is met in each period and no production of any item is brought forward.

The total waste of material given by the lot for lot solution, for the three periods, is 7.500 + 8.6666 + 32.2727 = **48.4393**, which is much worse than the solution for the multiperiod problem.

4 Heuristic method

The mathematical model (3)–(6) was solved using a home-made simplex method with column generation, with the relaxed integrality constraints on the variables x_{jkt} (Eq. (6)). Poldi (2007) proposed two approaches to obtain the integer solution for the multiperiod cutting stock problem, based on rolling horizon strategies (de Araujo et al. 2008; de Araujo and Clark 2013). Now, the best of the two strategies proposed in Poldi (2007) for rounding the fractional solution of the multiperiod cutting stock problem is given.

Since we are dealing with a problem regarding several periods, the rolling horizon strategy is used, meaning that an integer solution for the first period is found while allowing the solution for the other periods to be fractional. In a second stage, after the first period has started being implemented in practice, new ordered items may arrive and some may be canceled. Therefore, the ordered items is updated and a new and slightly different multiperiod cutting stock problem is formed, which will be solved and, again, only its first period solution should be rounded to an integer solution. Although the solution for the next periods stays fractional, it gives an estimate of what will happen in the next periods.

Algorithm
Step 1:
Solve the multiperiod cutting stock problem (3)–(6) disregarding the integrality constraints and consider:

$\mathbf{A}^1$: matrix with the cutting patterns in the first period;
$\mathbf{x}^1$: vector with the frequencies of the cutting patterns in the first period;
$\mathbf{d}^1$: vector with the original demand in the first period;
$\mathbf{e}^1$: vector with the stock availability in the first period;
$\mathbf{r}^1$: vector with the amount of anticipated items in the first period;
n: the number of cutting patterns in the first period considering all stock objects types.

Round down all the components in vector $\mathbf{r}^1$;
Do: $\mathbf{dr} = \mathbf{d}^1 + \mathbf{r}^1$, it means, vector $\mathbf{dr}$ holds the original demand in the first period plus the amount of items brought forward.
Step 2:
Round the frequencies $\mathbf{x}^1$ as follows:
Let n be the number of cutting patterns in the first period.
For $j = 1, \ldots, n$, do: $x_j^1 = \lfloor x_j^1 \rfloor$
Step 3:
Update demand/production of items: $\mathbf{dr}$
Update stock availability: $\mathbf{e}^1$
If $\mathbf{A}^1 \geq \mathbf{d}^1$ (it means that original demand is met)
then STOP.
Step 4:
Solve a constrained knapsack problem to $\mathbf{dr}$ for each object type k available in stock.
Choose the cutting pattern with the smallest waste, such waste is given by ϕ.
Let $l_{min} = \min\{l_1, l_2, \ldots, l_m\}$ be the smallest ordered item.
If $\phi \geq l_{min}$ (i. e, there is still space for allocating an item in the cutting pattern)
then solve an unconstrained knapsack problem of size ϕ and fulfill the cutting pattern.
Apply this cutting pattern as many times as possible.
Go back to step 3.
End-of-the-algorithm

5 Computational experiments

Computational experiments were carried out in order to analyze the multiperiod cutting stock models, proposed in Sect. 2, and the heuristic method described in Sect. 4. We compare its solution with lot for lot solution that is currently used in practical problems. The heuristic method was implemented in Delphi 7. The mathematical models were implemented using AMPL/CPLEX 12.5. Tests were carried out on an i7 with 6Gb RAM. Gau and Wäscher (1995) proposed a generator for one-dimensional cutting stock problems, called CUTGEN1. However, CUTGEN1 cannot be used here because it generates instances with only one standard stock object length and we are dealing with the problem with several stock object lengths. So, a random generator based on CUTGEN1 had to be developed. Instances are divided into 8 classes of problems. Each class has 20 instances. Here are the details on how the classes were built and the results obtained.

Before presenting the computational results it is important to discuss the utilization of the models GGG and GAF. As the linear relaxations of the models GGG and GAF are equivalent

Table 5 Parameters which define the classes of instances

Class	Number of periods	Number of objects	Number of items
1	3	3	10
2	3	3	20
3	3	5	10
4	3	5	20
5	6	3	10
6	6	3	20
7	6	5	10
8	6	5	20

we can choose one of them to solve the linear relaxation and we choose the GGG model. When solving the mixed integer programming problem the model GAF has advantages because it is more compact and the complete set of arcs can be easy included in the model while it is not easy to include the complete set of columns for the model GGG. Moreover we observed that the computational package can deal very well with the arc flow model, probably because it has special routines for this kind of problems. The heuristic method was implemented using as base the model GGG, because it is a residual heuristic and is based on the linear relaxation of the problem and, as said before, we implemented a home-made simplex method with column generation for solving model GGG.

5.1 The random generator

In order to make computational experiments, some parameters were fixed, such as:

- number of periods: $T = 3$ and 6;
- number of stock object types: $K = 3$ and 5;
- number of ordered item types: $m = 10$ and 20;
- stock objects holding cost: $c^s_{kt} = \beta\ L_k$, with $\beta = 0$; 0.01 and 0.1;
- items holding cost: $c^r_{it} = \alpha\ l_i$, with $\alpha = 0$; 0.01; 0.1; 0.5 and 1.

Other parameters to define the instances were randomly generated in the following intervals,

- stock object length: $L_k \in [300\ 1000]$;
- item length: $l_i \in [0.1\ 0.4]\frac{\sum_{k=1}^{K} L_k}{K}$;
- stock availability of object type k, in period t: $e_{kt} \in [\lceil \text{av}_t \rceil \lceil 2\text{av}_t \rceil]$, where $\text{av}_t = \frac{\sum_{i=1}^{m} l_i d_{it}}{\sum_{k=1}^{K} L_k}$;
- demand of items: $d_{it} \in [1\ 50]$.

Eight classes of problems were generated, each one containing 20 instances. These classes were defined as shown in Table 5.

5.2 Computational results for $\alpha = 0$ and $\beta = 0$

Results in Tables 6 and 7 were obtained considering null holding costs, i.e., item holding cost $\alpha = 0$ and stock object holding cost $\beta = 0$. So, the objective function shown in Tables 6 and 7 is the total waste, since all holding costs (for objects and items) are null. Obviously, bringing forward parts that have zero holding cost will reduce waste, this section is therefore intended to show the improvement of the multiperiod solution in relation to the lot for lot solution.

Table 6 Objective function for the linear relaxation (average of the 20 instances in each class) $\alpha = \beta = 0$

Class	Lot for lot (linear)	Multiperiod (linear)	Difference	Gain %
1	124.39	112.91	11.48	9.23
2	33.44	30.61	2.83	8.46
3	83.18	76.01	7.17	8.62
4	11.34	9.59	1.75	15.43
5	323.86	255.70	68.16	21.05
6	79.61	54.39	25.22	31.68
7	258.43	198.94	59.49	23.01
8	99.46	94.83	4.63	4.66
Average	126.71	104.12	22.58	17.82

Table 7 Objective function for integer problem (average of the 20 instances in each class) $\alpha = \beta = 0$

Class	Lot for lot (integer)	Multiperiod (integer)	Difference	Gain %	Gap % (linear)	Gap % (CPLEX)
1	150.55	124.4	26.15	17.37	15.07	12.70
2	92.85	40.25	52.6	56.65	43.12	42.57
3	94.3	79.3	15	15.91	7.08	3.52
4	55.95	62.2	−6.25	−11.17	63.36	61.78
5	500.45	265.8	234.65	46.89	10.06	9.05
6	791	2172.41 (3)[a]	−1381.41	−174.64	60.75	60.53
7	323.3	207.8	115.5	35.73	8.81	7.92
8	15008.15	11235 (7)	3773.15	25.14	92.67	92.66
Average	2127.069	1773.95	353.6738	16.63	37.62	36.34

[a] The numbers in brackets are the number of instances that CPLEX could not find a feasible solution in the time limit

In Table 6 we consider the linear relaxation of two different solution approaches. The first one is the lot for lot approach which considers each period individually, i.e. in each period a classical cutting stock problem is solved without leaving stock for the next period. The second approach considers the multiperiod GGG model (it could be GAF since the results for the linear relaxation are equivalent). The second and third columns represent the average of the linear relaxation for 20 instances for each class, considering the lot for lot and the multiperiod approaches, respectively. The fourth and fifth columns are the absolute difference and percentage difference, respectively. The multiperiod cutting stock model, without holding costs, i. e., the method is free to bring forward items if they match better, in fact obtained solutions with less waste. On average the gain is 17.82 %.

Although it is not the aim of the multiperiod model, the computational tests show that the multiperiod approach to cutting stock problem provides solutions with fewer different cutting patterns than the lot for lot solution. Considering the solutions presented in Table 6, the average number (for the 8 classes) of different cutting patterns in the lot for lot solution is 70.2. The average for the multiperiod solution is 38.23 different cutting patterns. So, we can notice that the multiperiod approach could reduce the number of different cutting patterns in 45 %. The minimization of the number of different cutting patterns is being well explored in the literature (Diegel et al. 1996; Foerster and Wäscher 2000; Vanderbeck 2000; Aloisio et al.

Table 8 Analysis of the CPLEX package applied to the model GAF

Class	Number optimal solutions	Nodes	Cuts	Computational time (s)
1	6	1185794	179	453.16
2	9	52716.95	105	376.39
3	10	1714000.5	171	325.93
4	6	38673.2	84	485.55
5	2	1636869.8	332	556.32
6	3	33790.8	158	550.65
7	7	1504889.3	309	439.31
8	0	9030.7	78	600.00
Average	5.37	771970.63	177	473.4

2009; Yanasse and Limeira 2006; Henn and Wäscher 2013; de Araujo et al. 2014; among others).

In Table 7, the same comparison is made but considering the integer problem. In order to solve this problem CPLEX 12.5 was used with a time limit of 10 min applied to the GAF formulation with the reduction criteria. In Table 7 we present two additional columns: Gap % (Linear) that represents the gap between the integer solution and the linear relaxation solution and is calculated by:

$$\text{Gap \% (Linear)} = \frac{100 \times (\text{Multiperiod (Integer)} - \text{Multiperiod (Linear)})}{\text{Multiperiod (Integer)}}.$$

The other additional column is Gap % (CPLEX) and presents the Gap provided by the solver CPLEX when solving the integer GAF formulation.

Comparing the multiperiod approach with the lot for lot approach, for the even classes (classes with 10 items) the multiperiod approach is much better than the lot for lot approach. However, for the odd classes (classes with 20 items) the CPLEX package cannot obtain good results by solving the multiperiod GAF formulation, and the results are not so good, except for class 2, which has 3 periods and 3 types of objects.

A similar conclusion can be reached when analyzing the columns Gap% (linear) and the Gap% (CPLEX). Analyzing the gap between the upper bound obtained by the Multiperiod (integer) solution and the lower bound given by the Multiperiod (linear) solution, we can see that for the classes with 10 items (even classes) the performance of the solver is relatively good. On average the Gap% (linear) for these classes is 10.25 %. However, when the number of items is increased to 20 items (odd classes), these gaps increase a lot, and for some instances of classes 6 and 8, the CPLEX package cannot find a feasible solution.

Table 8 shows the number of optimal solutions obtained by CPLEX limited by 10 min; the number of nodes of the branch-and-cut tree necessary to obtain such solution; the number of cuts applied at the root node of the branch-and-cut tree; and the computational time. As we can see, considering the even classes and class 2, CPLEX proved optimality of an expressive number of the 20 instances. It can explore a high number of nodes for the even classes.

5.3 Computational results for other values of holding costs

In order to better analyze the performance of the GAF formulation in solving the multiperiod problem, some additional results are shown where we vary the parameters: item holding cost ($c^r_{it} = \alpha\, l_i$) and object holding cost ($c^s_{it} = \beta\, L_k$). Table 9 shows the linear relaxation results,

Table 9 Linear relaxation objective function (average of the 120 instances in 6 classes)

	$\beta = 0.00$				$\beta = 0.01$				$\beta = 0.10$			
	Waste	Item holding cost	Object holding cost	Total costs	Waste	Item holding cost	Object holding cost	Total costs	Waste	Item holding cost	Object holding cost	Total costs
$\alpha = 0.00$	113.97	0.00	0.00	113.97	235.44	0.00	267.01	502.44	277.94	0.00	2451.22	2729.16
$\alpha = 0.01$	126.08	4.22	0.00	130.30	113.99	442.20	1350.22	1906.41	277.94	1541.52	2451.22	4270.69
$\alpha = 0.10$	135.26	1.16	0.00	136.42	135.26	1.16	1792.98	1929.41	116.12	4908.10	13013.07	18037.28
$\alpha = 0.50$	137.38	1.21	0.00	138.60	137.39	1.21	1793.01	1931.61	137.98	1.21	17929.14	18068.34
$\alpha = 1.00$	139.11	0.00	0.00	139.11	139.12	0.00	1792.98	1932.10	139.71	0.00	17928.87	18068.57

while Table 10 shows the results for the mixed integer GAF model, solved by CPLEX 12.5 limited to 10 min, considering variations in the parameters β and α. Holding costs considered here are fictitious; in practical situations they must be very carefully tuned so that the multiperiod model can give suitable solutions for this particular factory. Given the bad quality of the results for classes 6 and 8 presented in the previous section, in this section the results for such classes were not considered to compute the averages.

It can be seen in both tables that, keeping the object holding cost $\beta = 0$, when the parameter α is increased as well as the expected increasing of the item holding cost, the waste cost also increases. This happens because the reduction on the stocked items reduces the possible combinations which, in turn, increases the amount of waste. On the other hand, keeping the item holding cost $\alpha = 0$, the increasing of the parameter β also increases the waste cost, because it tries to use as many objects as possible in order to reduce the holding costs and this increases the waste cost. Considering the parameter $\alpha \neq 0$, when we vary the parameter β, the waste cost does not vary so much and the model focuses on minimizing the holding costs.

Comparing the linear relaxation (lower bound) solutions given in Table 9 and the integer (upper bound) solutions in Table 10 we can see that, for the considered instances, the CPLEX package obtained relatively good integer solutions independently of the variations in parameters α and β.

We have also compared the multiperiod strategy with the lot for lot strategy. However, as there is no inventory of items in the lot for lot strategy, there is no sense in varying the parameter α. So, Table 11 presents the results for the lot for lot strategy considering the variation of the parameter β. The first line of the results is the linear relaxation and can be compared to the first line of Table 9, and the second line is the integer results and can be compared to the first line of Table 10. These comparisons show the improvements made by the multiperiod strategy. When the parameter β is different from zero, the multiperiod strategy increases the waste of material; but this increasing is compensated by a decreasing of the object holding costs and consequently the multiperiod strategy obtains a smaller total cost.

5.4 Heuristic method results

Based on the results presented in the previous sections, when solving the GAF model with the CPLEX 12.5, good solutions can be obtained for some instances. However, two limitations of this procedure need to be emphasized. The first one is the computational time that needed to be limited to 10 min to obtain such solutions and this amount of time can be prohibitive in practice. The second one is the limitation on the size of instances that can be solved, as we can see in Table 7, classes 6 and 8. In order to overcome this limitation, the heuristic procedure presented in Sect. 4 was used and the results are presented in this section.

Since only the first period is indeed implemented, this is the only period whose solution will be rounded. However, it is not wise to have a very good solution for the first period and in the following periods, the quality of the solutions is not kept up. So, two estimated solutions are calculated for the following periods up to the planning horizon.

Such estimations were calculated as follows: all demanded items and available objects in stock, in all periods of the planning horizon, were summed. From these values, the production of the first period was subtracted, including items that were brought forward to the first period. Thus, we have a "super period" which represents the periods 2 up to T (where T is the last period). Therefore, a solution to the "super period" provides an estimate for the losses in the final period.

Table 10 Integer solution objective function (average of the 120 instances in 6 classes) after 10 min of CPLEX

	$\beta = 0.00$				$\beta = 0.01$				$\beta = 0.10$			
	Waste	Item holding cost	Object holding cost	Total costs	Waste	Item holding cost	Object holding cost	Total costs	Waste	Item holding cost	Object holding cost	Total costs
$\alpha = 0.00$	129.95	0.00	0.00	129.95	239.99	0.00	271.12	511.12	290.53	0.00	2451.21	2741.74
$\alpha = 0.01$	158.58	13.98	0.00	172.56	127.38	602.38	1189.67	1919.43	290.53	1541.19	2451.20	4282.93
$\alpha = 0.10$	183.36	33.26	0.00	216.62	174.70	21.06	1790.18	1985.93	138.70	6308.69	11607.94	18055.33
$\alpha = 0.50$	189.52	31.26	0.00	220.78	183.21	12.89	1791.73	1987.83	185.24	11.30	17915.34	18111.88
$\alpha = 1.00$	194.87	11.85	0.00	206.72	187.58	5.73	1791.83	1985.13	179.43	1.14	17919.46	18100.03

Table 11 Linear and Integer solution for the lot for lot strategy considering $\alpha = 0.00$ (average of the 120 instances in 6 classes)

	$\beta = 0.00$				$\beta = 0.01$				$\beta = 0.10$			
	Waste	Item holding cost	Object holding cost	Total costs	Waste	Item holding cost	Object holding cost	Total costs	Waste	Item holding cost	Object holding cost	Total costs
Linear	139.11	0.00	0.00	139.11	139.12	0.00	1792.98	1932.10	129.71	0.00	17928.87	18068.57
Integer	203.01	0.00	0.00	203.01	206.08	0.00	1792.02	1998.10	209.08	0.00	17910.86	18119.87

Table 12 Value of the objective function (total waste): rounded first period + linear relaxation solution for the other periods

Class	Lot for lot heuristic solution	Multiperiod heuristic solution
1	364.33	155.99
2	207.33	120.71
3	242.14	181.28
4	187.62	176.22
5	886.72	294.42
6	328.45	178.50
7	766.73	381.01
8	447.52	235.24
Average	428.83	215.42

Table 13 Value of the objective function (total waste): rounded first period + rounded other periods

Class	Lot for lot heuristic solution	Multiperiod heuristic solution
1	832.20	411.35
2	721.20	289.25
3	613.65	335.50
4	522.10	345.10
5	2003.65	482.90
6	1524.55	375.30
7	1690.30	573.60
8	1098.70	425.80
Average	1125.79	404.85

Two different approaches are used to find a solution for the "super period". In the first one, as there is no need to explicitly determine the integer solution for the future periods, we solve the "super period" by linear programming. Then, the waste of the rounded first period plus the waste obtained by linear relaxation provides a limiting factor for the overall waste. These values are given in Table 12.

In Tables 12 and 13, the results of the lot for lot solution obtained when the same heuristic is applied to round the linear solution are presented, i.e. the same algorithm described in Sect. 4 is applied but in step 1 the single period cutting stock problem is solved.

In the second approach, an integer solution to the "super period" is determined. The procedure used was the three versions of rounding heuristics proposed by Poldi and Arenales (2009). We compute the best solution among the three obtained. Thus, the waste of the rounded first period plus the waste obtained for the integer solution of the "super period" provides a further estimate of the total waste. These values are given in Table 13.

As expected, in both tables, the heuristic solution is much better than the lot for lot solution. Solving the linear relaxation of the "super period" (Table 12) gives better results than solving its integer version (Table 13). However, the integer version gives a more realistic estimate for periods 2 to T. It is important to observe that both approaches found feasible solution for all instances.

Table 14 Computational time for the heuristic solution (in seconds). Total time for the 20 instances in each class

Class	Total time
1	1.832
2	13.801
3	3.439
4	21.145
5	4.477
6	38.671
7	11.462
8	93.551

Table 15 Objective function for integer problem considering e_{kt} as variables (average of the 20 instances in each class) $\alpha = \beta = 0.00$

Class	Lot for lot (integer)	Multiperiod (integer)	Difference	Gain %
1	137.3	101.7	35.6	25.99
2	191.7	74.4	117.3	61.19
3	62.6	47.4	15.2	24.28
4	105.5	69.65	35.85	33.98
5	420.5	236.2	184.3	43.89
6	992.1	220.5	771.6	77.78
7	191.5	90.9	100.6	52.53
8	648.9	414.0	234.9	36.20
Average	343.7	156.80	186.9	54.38

Finally, in Table 14, the computational time for the heuristic solution (considering the rounded solution for the "super period") is presented and it shows that, indeed, it is much faster than the 10 min given by the solution of the mathematical model.

5.5 Computational results considering the stock availability as decision variables

In this section we present some additional computational results regarding "e_{kt} : stock availability of object type k in period $t, k = 1, \ldots, K, t = 1, \ldots, T$", as decision variables, different from the original definition as parameter in the previous model. Observe that in this case the objects replenishments policy is integrated with the decisions related to the multiperiod cutting stock problem. We will present some computational results considering the integer solution of the model GAF solved by CPLEX 12.5 with time limit of 10 min.

Table 15 is similar to Table 7 but considering e_{kt} as variables. The average gain of the multiperiod approach compared to the lot for lot approach is 54.38 % which is even bigger than the gain presented in Table 7. Observe that on this case the CPLEX package did not have problems for solving the odd classes for the multiperiod approach.

Aiming to better analyze the performance of the GAF formulation in solving the multiperiod problem, some additional results are shown in Table 16 where we vary the parameter: item holding cost ($c^r_{it} = \alpha\, l_i$). Observe that there is no sense in varying the object holding cost ($c^s_{it} = \beta\, L_k$) because, once it is a variable in this case, it will be always zero if we have costs different from zero. This happens because we are considering the object is available in the same period it is ordered, i.e., there is no lead time for the objects.

Table 16 Objective function for integer problem considering e_{kt} as variables (average of the 20 instances in each class) $\beta = 0.00$

Class	$\alpha = 0.00$		$\alpha = 0.01$		$\alpha = 0.10$		$\alpha = 0.50$		$\alpha = 1.00$	
	Waste	Item holding cost	Waste	Item holding cost	Waste	Item holding cost	Waste	Item holding cost	Waste	Item holding cost
1	101.7	0.00	109.75	6.14	144.90	4.78	151.30	0.00	149.50	0.00
2	74.4	0.00	140.35	10.20	72.30	31.05	56.20	8.55	65.30	15.5
3	47.4	0.00	55.0	2.88	93.45	0.49	93.85	0.99	93.95	0.00
4	69.65	0.00	143.0	15.73	59.35	13.34	42.95	50.6	58.15	27.75
5	236.2	0.00	343.75	36.78	477.40	73.98	456.75	132.67	462.20	57.8
6	220.5	0.00	704.45	53.41	531.50	179.06	2397.00	654.75	637.75	464.45
7	90.9	0.00	704.45	53.41	393.20	66.76	391.05	85.625	336.10	12.35
8	414	0.00	763.70	41.13	24613.05	1553.18	37502.15	8571.40	18692.00	13857.25
Average	156.80	0.00	306.79	25.32	3298.14	240.33	5136.41	1187.95	2561.87	1804.39
Total	156.80		332.1079		3538.476		6324.36		4366.26	

Analyzing the results of Table 16 it can be seen that keeping the object holding cost $\beta = 0$, when the parameter α is increased, as expected, the item holding cost also increase. However, the waste cost does not necessary increase as in Table 10. It is worth noticing the difficulties on solving some instances of classes 6 and 8 which make the average to increase a lot in comparison to the results of the other classes.

6 Conclusions and future research

In this work a cutting stock problem in which demand occurs along a planning horizon was defined. A mathematical model was presented that generalized the classical model proposed by Gilmore and Gomory (1963) and the column generation technique was adapted for solving the linear relaxation of the multiperiod problem. A generalization of an arc flow model (Valério de Carvalho 1999) was also proposed that can solve small instances of the mixed integer multiperiod problem. Finally, a heuristic procedure was presented in order to improve the computational time and to solve difficult instances of the problem.

Computational experiments have shown that the multiperiod cutting stock model seems to have a great potential to obtain better solutions than the lot for lot solution, which is generally used in practical situations. The generalized arc flow model can solve small instances of the mixed integer multiperiod problem and the heuristic procedure improves the computational time taken to solve the model and can be used to deal with difficult instances of the problem.

It has also been shown by the computational results that the multiperiod approach to cutting stock problem provides solutions with fewer different cutting patterns than the lot for lot solution. Reducing the number of different cutting patterns is an important and useful byproduct in some practical cases where the setup cost of a machine to perform each cutting pattern is relevant. An idea for future research is to explore multiperiod approach also with the aim of reducing the number of different cutting patterns.

Other ideas for future research are: to extend the solution method to two-dimensional cutting stock problems; the inclusion of capacity constraints; the extensions of the integration of objects replenishments policy with decisions related to the multiperiod cutting stock problem; the development of a branch-and-price method for solving the multiperiod problem; and finally, the authors intend to use the ideas presented in this paper on the integrated lot sizing and cutting stock problem.

Acknowledgments We would like to thank the anonymous referees for their constructive comments, which led to a clearer presentation of the material. This research was partially supported by CAPES, CNPq and FAPESP (2010/10133-0; 2014/01203-5 ans 2015/05193-7).

References

Alem, D. J., & Morabito, R. (2012). Production planning in furniture settings via robust optimization. *Computers & Operations Research*, *39*, 139–150.

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

Aloisio, A., Arbib, C., & Marinelli, F. (2009). On LP relaxations for the pattern minimization problem. *Networks*, *57*, 247–253.

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

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

Beasley, J. E. (1985). An exact two-dimensional non-guillotine cutting tree search procedure. *Operations Research*, *33*, 49–64.

de Araujo, S. A., Arenales, M. N., & Clark, A. R. (2008). Lot-sizing and furnace scheduling in small foundries. *Computers & Operations Research*, *35*, 916–932.

de Araujo, S. A., & Clark, R. (2013). A priori reformulations for joint rolling-horizon scheduling of materials processing and lot-sizing problem. *Computers & Industrial Engineering*, *65*, 577–585.

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

Diegel, A., Chetty, M., Van Schalkwyk, S. & Naidoo, S. (1996). *Setup combining in the trim-loss problem 3 to 2 and 2 to 1*. Durban: Business Administration, University of Natal. Working Paper.

Foerster, H., & Wäscher, G. (2000). Pattern reduction in one-dimensional cutting stock problems. *International Journal of Production Research*, *38*(7), 1657–1676.

Gau, T., & Wäscher, G. (1995). CUTGEN1: A problem generator for the standard one-dimensional cutting stock problem. *European Journal of Operational Research*, *84*(3), 572–579.

Gilmore, P. C., & Gomory, R. E. (1961). A linear programming approach to the cutting stock problem. *Operations Research*, *9*, 848–859.

Gilmore, P. C., & Gomory, R. E. (1963). A linear programming approach to the cutting stock problem-Part II. *Operations Research*, *11*, 863–888.

Gilmore, P. C., & Gomory, R. E. (1965). Multi-stage cutting stock problems of two and more dimensions. *Operations Research*, *13*, 94–120.

Gramani, M. C. N., & França, P. M. (2006). The combined cutting stock and lot sizing problem in industrial process. *European Journal of Operational Research*, *74*, 509–521.

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

Gramani, M. C. N., França, P. M., & Arenales, M. N. (2011). A linear optimization approach to the combined production planning model. *Journal of the Franklin Institute*, *348*, 1523–1536.

Henn, S., & Wäscher, G. (2013). Extensions of cutting problems: Setups. *Pesquisa Operacional*, *33*(2), 133–162.

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

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

Malik, M. M., Qiu, M. & Taplin J. (2009). An integrated approach to the lot sizing and cutting stock problems. In *Proceedings of the 2009 International Conference on Industrial Engineering and Engineering Management, IEEM 2009*, Hong Kong, China, pp. 1111–1115.

Nemhauser, G., & Wolsey, L. A. (1988). *Integer and combinatorial optimization*. New York: Wiley.

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

Pinho de Sousa, J., & Wolsey, L. A. (1992). A time-indexed formulation of non-preemptive single machine scheduling problems. *Mathematical Programming*, *54*, 353–367.

Poldi, K. C. (2007). O problema de corte de estoque multiperíodo. PhD Thesis, Instituto de Ciências Matemáticas e de Computação ICMC/USP – Universidade de São Paulo.

Poldi, K. C., & Arenales, M. N. (2009). Heuristics for the one-dimensional cutting stock problem with limited multiple stock lengths. *Computers & Operations Research*, *36*, 2074–2081.

Poldi, K. C., & Arenales, M. N. (2010). O problema de corte de estoque unidimensional multiperíodo. *Pesquisa Operacional*, *30*, 153–174.

Poltroniere, S. C., Poldi, K. C., Toledo, F. M. B., & Arenales, M. N. (2008). A coupling cutting stock-lot sizing problem in the paper industry. *Annals of Operations Research*, *157*, 91–104.

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

Silva, E., Alvelos, F., & Valério de Carvalho, J. M. (2014). Integrating two-dimensional cutting stock and lot sizing problems. *Journal of the Operational Research Society*, *65*, 108–123.

Suliman, S. M. A. (2012). An algorithm for solving lot sizing and cutting stock problem within aluminum fabrication industry. In *Proceedings of the 2012 International Conference on Industrial Engineering and Operations Management*, Bali, Indonesia, pp. 783–793.

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

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

Valério de Carvalho, J. M. (2002). LP models for bin packing and cutting stock problems. *European Journal of Operational Research*, *144*, 253–273.

Vanderbeck, F. (2000). Exact algorithm for minimising the number of setups in the one-dimensional cutting stock problem. *Operations Research*, *48*, 915–926.

Vanzela, M., Rangel, M. S., & Araujo, S. A. (2013). The integrated lot sizing and cutting stock problem in a furniture factory. In *Annals of the 11th IFAC Workshop on Intelligent Manufacturing Systems*, São Paulo, Brazil.

Yanasse, H. H., & Limeira, M. S. (2006). A hybrid heuristic to reduce the number of different patterns in cutting stock problems. *Computer & Operations Research*, *33*, 2744–2756.