# An extended goal programming model for the multiobjective integrated lot-sizing and cutting stock problem

Washington A. Oliveira*, Diego J. Fiorotto

*School of Applied Sciences, University of Campinas, R. Pedro Zaccaria, 1300, Limeira, 13484-350, SP, Brazil*

Xiang Song, Dylan F. Jones

*Centre for Operational Research and Logistics, School of Mathematics and Physics, University of Portsmouth, Lion Gate Building, Lion Terrace, Portsmouth PO1 3HF, UK*

**Abstract**

Lot sizing and cutting stock problems generally arise in manufacturing as a two stage connected process. Although these two problems have been addressed separately by many authors, recently, inspired by practical applications, some studies have emerged analyzing their integration. The mono-objective version of the integrated lot-sizing and cutting stock problem can be considered as an enhancement that minimizes the global production cost. However, it does not include the multiple criteria that can arise from the inclusion of multiple stakeholders in a modern, distributed manufacturing process. This paper aims to introduce a new extended goal programming model for the integrated lot-sizing and cutting stock problem which models the arising multiple criteria by a set of goals representing the interests of different stakeholders in the manufacturing process. This formulation allows for the consideration of the balance between the conflicting goals of multiple stakeholders and the cost efficiency of the overall process. In order to efficiently solve the proposed model, a column generation based heuristic procedure is applied. The trade-offs among the various criteria related to the problem are assessed, and a series of computational experiments are performed. The computational results compare individual criteria weighting schemes and evaluate the goals' sensitivity using performance profiles and time-to-target plots methodologies.

*Keywords:* Goal programming, lot-sizing problems, cutting-stock problems, integrated problems, extended goal programming

*Corresponding author

*Email addresses:* `waoliv@unicamp.br` (Washington A. Oliveira*), `fiorotto@unicamp.br` (Diego J. Fiorotto), `xiang.song@port.ac.uk` (Xiang Song), `dylan.jones@port.ac.uk` (Dylan F. Jones)

*March 20, 2021*

## 1. Introduction

The lot-sizing problem (LSP) and the cutting stock problem (CSP) have been widely considered since each problem in itself presents significant scientific challenges and appears in several industrial practical applications. Literature has mostly dealt with these two problems separately. However, recent papers are giving more emphasis to their integration. The interest in this combination comes from the integrated environments in various manufacturing industries since the lot-sizing and cutting stock problems arise as two-stage successive processes.

One such pair of processes consists of producing (or purchasing) objects of the selected sizes and material types from which LSP finds a trade-off among production, inventory and setup costs, respecting the capacity of the machine and meeting the demand for objects. Afterwards, the objects are cut into smaller pieces in a second process in which CSP selects the best cutting patterns and decides the number of objects to be used. In this process, the customer demand for pieces has to be met by minimizing, for example, the waste of material and inventory costs. Multiple authors address the lot-sizing problem in several environments, for instance it appears in the review papers of Drexl & Kimms (1997), Jans & Degraeve (2008), Glock et al. (2014), and Copil et al. (2017) classified according to different characteristics: single level or multi-level production structure; fixed or dynamic demand; finite or infinite time horizon, discrete or continuous time periods; and the consideration of capacity and setup time constraints. A CSP overview can be seen in the review papers of Sweeney & Paternoster (1992), Cheng et al. (1994), Valério de Carvalho (2002), and Delorme et al. (2016).

The integrated problem can be modeled as a deterministic mathematical model that covers several aspects found in LSP and CSP models. The integration can be classified with respect to two aspects: time horizon and production stages (Melega et al., 2018). Since the long-term planning horizon can be divided into multiple periods, a link between two consecutive periods is naturally formed – the inventory at the end of the one period becomes the inventory at the beginning of the following period. This link counts for the first type of integration. The two successive production stages, where objects are produced or purchased (lot-sizing stage) and then are cut into small pieces (cutting-stock stage), counts for the second type of integration.

The mathematical model proposed in this paper aims to plan the complete production process of LSP integrated with CSP in a way that a set of decision makers in different manufacturing sectors could achieve a series of targets associated with their respective goals. These goals are, more often than not, conflicting. Specifically, there are three sectors – production, machine setup and inventory in both lot sizing and cutting stock stages respectively in this integrated problem. The difference between LSP and CSP is that LSP consists of determining the production level, inventory level and the set of machines to setup in order to produce the required objects. Whereas, CSP consists of determining the number of the pieces to produce, to be stored in inventory and the cutting pattern (setup) to use. The six sectors (production, inventory and setup for both LSP and CSP respectively) involved in the complete manufacturing process each aims at minimizing their own cost incurred. This is in line with the ethos of distributed manufacturing where each of the functions may belong to a different company or a different division within a company. The six goals, and hence the interests of the entities to minimize their costs, are conflicting. For example, the production cost will be reduced by production leveling if a high inventory is held to satisfy surges in demand. In the mathematical model proposed in this paper, the range of solutions between optimization of the whole system and providing a balance between the six goals is investigated, which provides greater insight into the dynamics of the integrated model than the mono-objective models of the integrated LSP and CSP in the literature. In the following paragraphs, we present an overall literature review of the applications of the integrated LSP and CSP in different industries along with mono and multi-objective approaches to the integrated problem.

The practical applications for the integrated LSP/CSP naturally appear in various industries. For example, these two problems exist in successive stages in industries of paper, clothing, aluminum, copper, and furniture. In the paper industry (Campello et al., 2019), lot-sizing is a primary manufacturing process where a number of large reels are produced with a specified length and paper type. Next, the large reels are cut into smaller reels to meet the customer demands for which a one-dimensional cutting stock problem needs to be solved. Furthermore, the larger and smaller reels can be stored at the end of each period, and can be used for satisfying the demand in the following period.

In the clothing industry (Farley, 1988), different aspects overlap the classical formulation for the cutting-stock problem. The requests for clothes can be specified by style, material, color, and size. Waste minimization is only a subsidiary objective whereas the overall objective is to maximize long-term profitability. The integration among the production stages occurs in the laying, cutting and sewing operations. In the aluminum industry (Suliman, 2012), an aluminum extrusion section of standard length is cut into pieces, where the lot-sizing is determined for each period and then the best cutting patterns are generated. The demand for the pieces depends on the lot size decision of how many final products are to be manufactured in each period. The products to be manufactured are established based on the demand in each period and the available capacity. If the available capacity is less than that required to satisfy the period, then the capacity shortage of this period will be compensated from previous inventories. In the copper industry (Hendry et al., 1996), the major production operation is carried out in the foundry. The manufacture of copper consists of integrating three processes, the operation of the furnace to melt one lot of scrap copper, the process of converting the molten metal into one lot of "logs", and the cutting process of the logs into smaller "billets". Moreover, as the cost of storage is insignificant in comparison with the cost of a charge of the furnace, then it is sometimes desirable to hold the surplus billets among the periods if overproduction leads to more efficient use of the furnace. In the furniture industry (Toscano et al., 2017), a lot-sizing problem is solved for determining the number of wooden plates to be produced or purchased, and the integration of LSP and CSP is needed since these wooden plates need to be cut further into smaller wooden parts to satisfy the customers' demand. The possibility of holding wood plates and parts of plates across multiple periods is the key for satisfying all the customer requirements over a long-term planning horizon.

Note that in the aforementioned applications, a separate approach to LSP and CSP will produce reasonable local solutions, but the joint solution may lead to a high joint cost (Malik et al., 2009), since there is a dependency between these two problems. For example, the wastage incurred in CSP is dependent on the sizes and quantities of the objects cut out from the lot sizing stage. Thus quite a few literature works deal with the integrated LSP and CSP in a mono-objective approach. That is, they find the joint solution for a single objective – to minimize the joint total cost in the two stages over the whole planning horizon in this way, they achieve the overall cost reductions as reported in the literature.

Arbib & Marinelli (2005) incorporated two hierarchical decision levels (short-term operations vs. mid-term planning) and two functional areas (production vs. purchase of materials) for a two-stage flow line where the first stage produces components for the downstream assembly stage. Their approach ensured on average a further reduction of about 43% of the overall cost. The integrated LSP and CSP studied by Gramani & França (2006) achieved a profit margin of up to 13% in comparison to the solutions to the problems solved separately. The overall cost was reduced up to 25% from a similar modeling for the paper industry of Malik et al. (2009). The integrated model for the furniture production of Vanzela et al. (2017) provided better results reaching a gain of about 29% compared to the problems solved separately. Similar integrated mono-objective approaches and tailored solution algorithms to achieve overall cost reduction are discussed by Hendry et al. (1996), Nonas & Thorstenson (2000), Ghidini et al. (2007), Poltroniere et al. (2008), Gramani et al. (2009, 2011), Alem & Morabito (2012), Suliman (2012), Silva et al. (2014), Leao et al. (2017), and Melega et al. (2020).

The overall cost reduction achieved in the aforementioned examples and the references justify that further relevant studies are essential for integrated problems characterized by multiple goals to be achieved. Multiobjective approaches have been studied for LSP and CSP separately. Wäscher (1990) solved a CSP with multiple objectives for which the trade-off among the inventory costs, extra cost of inputs, and waste of material during the cutting process were evaluated. Golfeto et al. (2009) used a symbiotic genetic algorithm to solve a bi-objective model for CSP. Aliano Filho et al. (2017, 2019) presented a bi-objective optimization problem to minimize the total number of different cutting patterns of CSP. Ustun & Demirtas (2008) connected the selection of raw material suppliers with LSP costs, and Rezaei & Davoodi (2011) introduced two multi-objective mixed integer non-linear models for multi-period LSP involving multiple products and suppliers. To our best knowledge, only one paper from the literature provides a bi-objective optimization model for studying the trade-off between LSP and CSP (Campello et al., 2019). A set of Pareto optimal solutions were obtained through classical weighted sum and $\varepsilon$-constraint methods for verifying the conflict between two objective functions that encompass the main production stages of the paper manufacturing.

### *1.1. Outline of this contribution*

Although the integrated lot-sizing and cutting stock problem arises in a unique production line, in general, the production of final products in the manufacturing process is complex and therefore is separated into several sectors. Each one of them has a specific target to be achieved that competes for limited resources. The different goals of the sectors of the problem, which as previously mentioned increasingly belong to different companies or divisions within a company, result in a conflicting production environment. In particular, each LSP and CSP can be divided into production, setup and inventory sectors. It makes the methods that consider multiple objectives suitable approaches to solve the studied problem. Note that the trade-offs among the goals can appear in different aspects. A potential conflict among the objectives happens, for example, for the setup, production and inventory levels. To meet large demands a very high amount of products needs to be produced in each period. Thus, it is interesting to make few setups of the machine (decreasing the setup costs), which can increase the amount of products in storage significantly. On the other hand, in order to minimize the inventory cost the space limitations and operating costs to keep products in storage have to be considered.

To our best knowledge, there are currently no studies that apply a goal programming model to analyze the integrated CSP and LSP from a multiobjective viewpoint. The goal programming methodology is an excellent tool to deal with conflicting objectives and the balance between them (Jones & Tamiz, 2010). The essence of goal programming is the minimization of unwanted deviation variables, which are combined into an achievement function whose purpose is to minimize them and thus ensure that a solution is found that is “as close as possible” to the set of desired goals.

### *1.2. Main contributions*

This paper proposes an extended goal programming model (Romero, 2004) for the integrated LSP and CSP. The model determines how many pieces and objects of each type must be produced over each period of the planning horizon while analyzing the trade-off among the various costs associated with two production stages. Additionally, it selects a production plan in line with the decision makers’ preferences for the achievement of the multiple goals. The main contributions include:

-to present a new multi-objective mathematical formulation for the integrated lot-sizing and cutting stock problem with parallel machines and multiple periods;

-to apply a column generation based heuristic procedure to solve an extended goal programming model;

-to investigate, through performing extensive computational experiments, the trade-off between the several different well-known lot-sizing and cutting stock objective functions utilizing a multiple objective methodology;

-to use a performance metric given by several plots to conduct a sensitivity analysis of model outputs in the sense that the difference between the values of the goals and the targets are examined by varying some parameters. Here, the performance profiles and time-to-target plots methodologies are adapted to compare individual weights of the achievement function and to evaluate the goals’ sensitivity. These approaches provide further interpretations of the computational results.

The remainder of this paper is organized as follows. Section 2 provides the proposed mathematical model. Section 3 describes the utilized solution method. Section 4 presents a series computational experiments to validate the proposed approach. The last section provides a summary of our findings and some avenues for future research.

## 2. Mathematical modeling

Extended goal programming (EGP) can be used to model the developed integrated problem due to its ability to combine the multiple underlying philosophies of satisficing, optimizing, and balancing in a multiobjective environment (Jones & Tamiz, 2010).

The non-lexicographic version of the classic extended goal programming formulation of Romero (2004) is chosen because the decision maker wishes to consider the number of goals achieved and does not have a natural order in which they wish to satisfy their goals. From the different solutions (scenarios) obtained from EGP, the decision maker can select the solution, from amongst a manageably small set, that gives their preferred mix of efficiency versus balance between the set of goals.

This section presents the extended goal programming model for the integrated lot-sizing and cutting stock problem (EGP-ILCSP). This model allows an evaluation of the trade-offs between the achievement of the different goals involved in the manufacturing process with multiple dimensions of integration. EGP-ILCSP is a deterministic model for a production environment composed of two stages and multiple periods of a production process, where objects are produced and then converted into pieces from a cutting stage. To compose the formulation, LSP and CSP models are adapted from the models proposed by Trigeiro et al. (1989) and Gilmore & Gomory (1963, 1965), respectively. The latter model can be easily adapted to consider multi-dimensional problems and other CSP extensions while respecting the physical limitations of the object to define the way that the pieces are cut (cutting patterns) from an object.

In relation to these two production stages, our EGP-ILCSP model is more general than the model proposed by Melega et al. (2018, 2020), since it considers multiple machines and the objects and pieces can be produced from different types of raw materials. Note that some industries use different raw materials for assembling a final product. For example, bars of iron, plastic and aluminum need to be produced (or purchased) and cut for forming components of a finished product, also the paper rolls from the paper industry have different grammages and lengths.

The model proposed in this article can be seen as a two-stage integration of production, for which Stage 1 plans the acquisition of the objects of different sizes to be cut into pieces. The acquisition can be done either by internal production or by ordering from an outside supplier. Stage 2 corresponds to the cutting process, in which the objects are cut into pieces from the cutting patterns. The connection between Stages 1 and 2 corresponds to the first integration. Demand for pieces are independent but this causes a dependent demand for objects. However, objects may also have independent demand. The connection among the different periods provides the second integration since both objects and pieces can be manufactured in previous periods and stored from one period to the subsequent period. Demand must be satisfied either from production in the current period or from inventory carried over from the previous period.

The following notations and input parameters are defined for the EGP-ILCSP model, where $|X|$ denotes the cardinality of set $X$.

*Sets and indices*:

$I$: set of objective functions (index $i$);

$T$: set of time periods (index $t$);

$M$: set of machines (index $m$);

$K$: set of raw materials (index $k$);

$O_m$: set of different types of objects (index $o$) from machine $m$;

$P_k$: set of different types of pieces (index $p$) of raw material $k$;

$J_o$: set of cutting patterns for object type $o$ (index $j$);

$O := \{O_1, O_2, \ldots, O_{|M|}\}$: set of all types of objects;

$P := \{P_1, P_2, \ldots, P_{|K|}\}$: set of all types of pieces.

*Parameters*:

$sc^o_{kmt}$: setup cost for producing object type $o$ of raw material $k$ on machine $m$ in period $t$;

$vc^o_{kmt}$: unit cost for producing an object type $o$ of raw material $k$ on machine $m$ in period $t$;

$hc^o_{kt}$: unit holding cost for object type $o$ of raw material $k$ in period $t$;

$D^o_{kt}$: demand for object type $o$ of raw material $k$ in period $t$;

$Se^o_{kmt}$: setup time for producing object type $o$ of raw material $k$ on machine $m$ in period $t$;

$Ve^o_{kmt}$: production time per object type $o$ of raw material $k$ on machine $m$ in period $t$;

$CapO_{mt}$: production capacity available to produce the objects on machine $m$ in period $t$;

$u^o_{jk}$: setup cost for producing cutting pattern $j$ of object type $o$ of raw material $k$;

$c^o_{jk}$: unit cost of producing cutting pattern $j$ of object type $o$ of raw material $k$;

$a^p_{oj}$: number of piece type $p$ cut from object type $o$ using the cutting pattern $j$;

$hc^p_{kt}$: unit holding cost for piece type $p$ of raw material $k$ in period $t$;

$d^p_{kt}$: demand for piece type $p$ of raw material $k$ in period $t$;

$se^o_{jkt}$: setup time for cutting pattern $j$ of object type $o$ of raw material $k$ in period $t$;

$ve^o_{jkt}$: production time per cutting pattern $j$ of object type $o$ of raw material $k$ in period $t$;

$CapP_t$: cutting capacity available in period $t$;

$Q_w$: arbitrarily large constant;

$Q_y$: arbitrarily large constant;

$b^i_t$: target value $i$ in period $t$;

$v^i_t$: preferential weight for positive deviation from the target value $i$ in period $t$;

$\alpha$: parameter that controls the mixture of balance between, and efficient achievement of, goals in the achievement function, $\alpha \in (0, 1)$.

*Decision variables*:
$X^o_{kmt}$: production (or purchase) quantity of object type $o$ of raw material $k$ on machine $m$ in period $t$;
$S^o_{kt}$: inventory of object $o$ of raw material $k$ at the end of period $t$;
$Y^o_{kmt}$: binary variable for the production setup; $Y^o_{kmt} = 1$ if object type $o$ of raw material $k$ is produced on machine $m$ in period $t$, otherwise $Y^o_{kmt} = 0$;
$S^p_{kt}$: inventory of piece $p$ of raw material $k$ at the end of period $t$;
$Z^o_{jkt}$: number of objects of type $o$ of raw material $k$ cut according to cutting pattern $j$ in period $t$;
$W^o_{jkt}$: binary variable for the cutting setup; $W^o_{jkt} = 1$ if cutting pattern $j$ is used to cut object type $o$ of raw material $k$ in period $t$, otherwise $W^o_{jkt} = 0$;
$\lambda$: the maximal deviation from amongst the set of normalized unwanted deviations;
$n^i_t$: negative deviation from the target value $i$ in period $t$;
$p^i_t$: positive deviation from the target value $i$ in period $t$.

Let $\underline{x} := (X,Y,Z,S,W)^T$ be a vector containing all variables $X^o_{kmt}, Y^o_{kmt}, Z^o_{jkt}, S^o_{kt}, S^p_{kt}$, and $W^o_{jkt}$, $I := \{1,\ldots,6\}$ be the set of function index, and let

$f^1_t(\underline{x}) := \sum_{k\in K}\sum_{m\in M}\sum_{o\in O_m} vc^o_{kmt} X^o_{kmt}$ be the *LSP production cost* in period $t$ at the solution $\underline{x}$;

$f^2_t(\underline{x}) := \sum_{k\in K}\sum_{o\in O} hc^o_{kt} S^o_{kt}$ be the *LSP holding cost* in period $t$ at the solution $\underline{x}$;

$f^3_t(\underline{x}) := \sum_{k\in K}\sum_{m\in M}\sum_{o\in O_m} sc^o_{kmt} Y^o_{kmt}$ be the *LSP setup cost* in period $t$ at the solution $\underline{x}$;

$f^4_t(\underline{x}) := \sum_{k\in K}\sum_{o\in O}\sum_{j\in J_o} c^o_{jk} Z^o_{jkt}$ be the *CSP cutting cost* in period $t$ at the solution $\underline{x}$;

$f^5_t(\underline{x}) := \sum_{k\in K}\sum_{p\in P_k} hc^p_{kt} S^p_{kt}$ be the *CSP holding cost* in period $t$ at the solution $\underline{x}$; and

$f^6_t(\underline{x}) := \sum_{k\in K}\sum_{o\in O}\sum_{j\in J_o} u^o_{jk} W^o_{jkt}$ be the *CSP setup cost* in period $t$ at the solution $\underline{x}$.

Note that each $f^i_t(\underline{x})$, for $i \in I$, is a cost function that measures a standard cost of the manufacturing process in period $t \in T$. The mathematical formulation for the EGP-ILCSP model is as follows.

$$\text{Min} \quad a = \alpha\lambda + (1-\alpha)\left(\sum_{t\in T}\sum_{i\in I}\left(\frac{v_t^i p_t^i}{b_t^i}\right)\right) \tag{1}$$

$$\text{s.t.} \quad \frac{v_t^i p_t^i}{b_t^i} \leq \lambda, \quad i \in I, \quad t \in T; \tag{2}$$

$$f_t^i(\underline{x}) + n_t^i - \underline{p_t^i} = b_t^i, \quad i \in I, \quad t \in T; \tag{3}$$

$$\sum_{o\in O}\sum_{j\in J_o} a_{oj}^p Z_{jkt}^o + S_{k(t-1)}^p = d_{kt}^p + S_{kt}^p, \quad k \in K, \quad p \in P_k, \quad t \in T; \tag{4}$$

$$Z_{jkt}^o \leq Q_w W_{jkt}^o, \quad o \in O, \quad j \in J_o, \quad k \in K, \quad t \in T; \tag{5}$$

$$\sum_{o\in O}\sum_{j\in J_o}\sum_{k\in K}\left(ve_{jkt}^o Z_{jkt}^o + se_{jkt}^o W_{jkt}^o\right) \leq CapP_t, \quad t \in T; \tag{6}$$

$$\sum_{m\in M} X_{kmt}^o + S_{k(t-1)}^o = \sum_{j\in J_o} Z_{jkt}^o + D_{kt}^o + S_{kt}^o, \quad o \in O, \quad k \in K, \quad t \in T; \tag{7}$$

$$X_{kmt}^o \leq Q_y Y_{kmt}^o, \quad m \in M, \quad o \in O_m, \quad k \in K, \quad t \in T; \tag{8}$$

$$\sum_{o\in O_m}\sum_{k\in K}(Ve_{kmt}^o X_{kmt}^o + Se_{kmt}^o Y_{kmt}^o) \leq CapO_{mt}, \quad m \in M, \quad t \in T; \tag{9}$$

$$S_{k0}^o = s_0^o,\ S_{k0}^p = s_0^p, \quad k \in K, \quad p \in P_k, \quad o \in O; \tag{10}$$

$$S_{kt}^o,\ S_{kt}^p \in \mathbb{Z}_+, \quad o \in O, \quad k \in K, \quad p \in P_k, \quad t \in T; \tag{11}$$

$$X_{kmt}^o \in \mathbb{Z}_+,\ Y_{kmt}^o \in \{0,1\}, \quad m \in M, \quad o \in O_m, \quad k \in K, \quad t \in T; \tag{12}$$

$$Z_{jkt}^o \in \mathbb{Z}_+,\ W_{jkt}^o \in \{0,1\},\ k \in K,\ p \in P_k,\ o \in O,\ j \in J_o,\ t \in T; \tag{13}$$

$$\underline{x} = (X,Y,Z,S,W)^T; \tag{14}$$

$$n_t^i, p_t^i \geq 0, \quad i \in I, \quad t \in T. \tag{15}$$

The achievement function (1) of the EGP-ILCSP model provides the match of efficiency versus equity by minimizing the average deviation and the worst deviation from amongst the set of targets, through the extended goal programming technique which utilizes what is sometimes referred to as the efficiency-equity trade-off (Romero, 2004). The balance between optimization and balance of the achievement function (1) can be controlled through the parameter $\alpha$, which can be varied between a complete emphasis on optimization ($\alpha = 0$) and a complete emphasis on balance ($\alpha = 1$). Thus, $\alpha$ is interpreted as a parameter that allows investigation of the trade-off between obtaining the "best average solution" and the "most balanced solution". Moreover, the balancing philosophy is enforced through the maximal deviation terms $\lambda$ in (2).

The model allows the decision maker to achieve the goals as closely as possible through the $6|T|$ one-sided goals in (3), where $f_t^i(\underline{x})$ is the achieved value of the $i$th objective at the solution $\underline{x}$ which has an associated target value of $b_t^i > 0$ in period

$t$. Each objective $f_t^i(\underline{x})$ contains deviation variables $n_t^i$ (negative) and $p_t^i$ (unwanted positive) from a goal to be achieved. The approach used in (3) is described by Jones & Tamiz (2010) as one-sided goals that involve cost, where any positive deviation above the goal level is penalized. The six objective functions are defined in a sense "less is better", so the only unwanted deviation variables to be minimized and included in the achievement function are the positive ones. These variables are also underlined in their respective goal equation (3) for emphasis. Penalizations or weights $v_t^i$ are assigned to the unwanted deviation variables. The target value $b_t^i$ is set for each objective, which also acts as a normalizing factor for each deviation variable.

Constraints (4)–(6) and (10)–(13) can be seen as a multi-period cutting stock problem with capacity constraints at Stage 2. Constraints (4) are balance constraints for demand for pieces, where $\sum_{o \in O} \sum_{j \in J_o} a_{oj}^p Z_{jkt}^o$ is the production quantity of piece $p$ of raw material $k$ in period $t$ cut out from the objects. These constraints provide an integration between CSP and LSP and give the number of pieces as a function of the selected cutting patterns. If a new cutting pattern is used in the cutting machine, (5) expresses the fact that its CSP setup cost must be computed. Constraints (6) are the capacity constraints of cutting machine in the cutting process. Constraints (7)–(9), and (10)–(13) represent a multi-period lot sizing problem with capacity constraints at Stage 1. Constraints (7) are balance constraints for demand for objects. These constraints also correspond to another integration between LSP and CSP, where the production (or purchase) of objects is guaranteed for the cutting process. If an object is produced then (8) computes its LSP setup cost. The capacity limit for the production of objects is modeled by constraint (9). If some objects are purchased from a supplier instead of internally produced, they may not be considered in the capacity constraints.

Constraints (10) define the opening stock levels, and the sets of constraints (11)–(13) specify the non-negativity and integrality of variables for the model. The feasible solution $\underline{x}$ must satisfy (14). The last set of constraints (15) specifies the non-negativity of the deviation variables.

## 3. Solution method: a column generation based heuristic procedure

A column generation based heuristic procedure is applied to the model (1)–(15), which deals with the very high number of possible cutting patterns (if set $J_o$ is too large, there will be too many decision variables $Z_{jkt}^o$) that usually appear in cutting stock problems. Furthermore, the motivation for studying this technique comes from the results obtained by several authors that have successfully used this technique to solve different problems (Fiorotto et al., 2015). Although there are distinct investigations applying column generation, to the best of our knowledge, there are currently no studies that employ this technique to solve a goal programming model.

### *3.1. General overview*

In general, the column generation based heuristic procedure used can be described as follows.

- Step 1: *Initialization*. To deal with the enormous number of cutting patterns of the problem (a large number of variables $Z^o_{jkt}$) the first restricted master problem created has the integrality of the variables relaxed and uses only the homogeneous cutting patterns;
- Step 2: *Main iteration*. Solve the linear relaxation of the restricted master problem. The relaxed solution of the current restricted master problem gives the value of the dual variables related to the constraints that contain the cutting pattern variables. Then the resolution of a particular subproblem for each machine and period possibly provides a new column to be added on the next restricted master problem;
- Step 3: *Stopping criteria*. The main iteration stops if there are no more attractive columns, or the improvement of the achievement function of the restricted master problem is smaller than a specific fixed value;
- Step 4: *Getting a feasible solution*. Solve the latest restricted master problem (with all generated columns in Step 2) considering the integrality of the relaxed variables in an attempt to find a feasible integer solution.

### *3.2. Initialization*

The column generation based heuristic procedure starts by building the first restricted master problem. Although there are some different ways to build it, this method takes into account only the homogeneous cutting patterns, which are the patterns/columns that produce only one type of item. In the case of one-dimensional cutting stock problem, for each object $o$ and piece $p$ calculate $a^p_{oj} = \lfloor \frac{L^o}{\ell^p} \rfloor$, where $L^o$ is the length of object $o$, and $\ell^p$ is the length of piece $p$. Thus each homogenous cutting pattern $j$ of this first restricted master problem is $(0,0,\ldots,a^p_{oj},0,\ldots,0)^T$. Note that when building the model (1)–(15), these cutting patterns are associated only with the variables $Z^o_{jkt}$.

### 3.3. Main iteration

The integrality of the variables is relaxed and the optimization package CPLEX 12.10 solves the current relaxed restricted master problem (the first of them has only the homogeneous columns), and the values of cutting pattern variables $Z^o_{jkt}$ at the solution yield the dual variables $\theta_t$, $\pi_{pkt}$, $\gamma_t$ and $\tau_{okt}$ related to the constraints (3), (4), (6) and (7), respectively. Observe also that each period $t$ defines the variable $\theta_t$ for the function $f^4_t(x)$ that is the only function containing variables $Z^o_{jkt}$. Next, each period $t$, raw material $k$, and object $o$ produce the following subproblem (knapsack problem) of a particular cutting pattern $j$.

$$SP^o_{kt} = \text{Min} \quad \sum_{p \in P_k} (-\pi_{pkt} a^p_{oj}) - \gamma_t ve^o_{jkt} + \tau_{okt} - \theta_t c^o_{jk} \tag{16}$$

$$\text{s.t.} \quad \sum_{p \in P_k} \ell^p a^p_{oj} \leq L^o; \tag{17}$$

$$a^p_{oj} \in \mathbb{Z}_+, \quad p \in P_k. \tag{18}$$

The resolution of (16)–(18) indicates for each period $t$, raw material $k$, and object $o$ if there is an attractive column. Thus, if $SP^o_{kt} < 0$ a new column is added on the next restricted master problem. The same optimization package solves the subproblems. Observe that the dual information of constraints (5) is not used in the knapsack subproblems. This is because the setup constraints (5) do not exist before creating each cutting pattern $j$, unlike constraints (3), (4), (6) and (7). Appendix A.1 details this issue based on the proof presented by Melega et al. (2020). Moreover, the reason that the objective function of the knapsack problem expresses the reduced cost of a column is described in Appendix A.2.

It is known from the literature that the column generation procedure usually presents a tailing off effect (Gilmore & Gomory, 1963), for which the column generation is affected by the instability of the dual variables that are influenced, among other factors, by primal degeneracy and dual oscillations. This effect can be avoided by different stabilization methods (Vanderbeck, 2005; Clautiaux et al., 2011). This paper uses a strategy that consists of a slackening of demand balance constraints, imposing a sign constraint on the dual variables to reduce the dual space and, consequently, accelerate the procedure. Considering that the achievement function of the proposed model is to minimize a combination of positive deviations in the sense that "less is better" for the objective functions, and the CSP cutting cost parameter $c^o_{jk}$ (related to the variables $Z^o_{jkt}$) takes non-negative values, the sign "$=$" of equality constraints (4) and (7) is replaced by the sign "$\geq$".

### *3.4. Stopping criteria*

The column generation based heuristic procedure stops by optimality or by five successive solutions without improvements. The optimality arises from the absence of attractive columns for the current restricted master problem. This absence means that for all periods, all raw material and object types, $SP^o_{kt} \geq 0$ holds when the corresponding knapsack problem (16)–(18) is solved to optimality. In this case, it achieves the optimal solution to the relaxed master problem. On the other hand, the procedure computes the difference between the achievement function values of two consecutive iterations of the restricted master problem. If this difference is smaller than 0.01% for five successive verifications then this implies that the procedure does not present a significant improvement, then it stops by solutions without improvements.

### *3.5. Getting a feasible solution*

After finishing the column generation procedure (Steps 1–3), the latest master problem has all generated columns. If the set containing all generated columns results on a feasible integer master problem, this step consists of using the optimization package CPLEX 12.10 to solve the latest generated master problem with the addition of the integrality constraints of the variables to try to obtain a integer feasible solution of the EGP-ILCSP model.

Let $\bar{v}^r_{RMP}$ be the achievement function value of the restricted master problem at pricing step $r$ and let $(SP^o_{kt})_r$ be the reduced cost of the column generated at iteration $r$. It is important to note that at each iteration of the column generation procedure we have not only the upper bound $\bar{v}^r_{RMP}$ on the value of the optimal solution of the master problem ($\bar{v}^*_{MP}$) but also the following lower bound (Lasdon, 1970):

$$LB = \bar{v}^r_{RMP} + \sum_{t\in T}\sum_{k\in K}\sum_{o\in O}(SP^o_{kt})_r \leq \bar{v}^*_{MP} \leq \bar{v}^r_{RMP} \tag{19}$$

Consequently, if the current restricted master solution, $\bar{v}^r_{RMP}$, is equal to the $LB$ presented above, no column prices out favorably and the column generation procedure stops. Furthermore, theoretically, since $\bar{v}^*_{MP} \leq a$ (value of the optimal solution of the model (1)–(15)), $LB$ is also a lower bound on $a$. Therefore, $LB$ can be used to analyze the quality of the integer feasible solutions generated by the used heuristic.

## 4. Numerical experiments

This section measures and interprets the quality and performance of our integrated approach via the extended goal programming model. The column generation based heuristic procedure was coded in C++ using the concert technology and IBM ILOG CPLEX 12.10 as the solver. The tests were executed on a computer with an Intel(R) Core(TM) i7-8565U processor, 1.99GHz with 16GB of RAM, and the Windows operating system. Details regarding the complexity and parameters of model and the analysis methods used appear in Appendix B–Appendix E.

### *4.1. Problem instances and setting of parameters*

In the computational experiments, four classes of instances of different sizes are generated: very small, small, medium, and large size (see Table 1). Each class of instances is differentiated by a different combination of the number of piece types, the number of periods, number of the available machines, and the number of raw material types. However, the size of each class of instance is defined as the sum of the number of piece types and number of periods. In each class, five instances are generated randomly following the ways proposed in the literature. And to make the parametric analysis, the set $\{w^0, w^1, \ldots, w^6\}$ containing seven weight vectors (see Table 2) and the set $\{0.01, 0.2, 0.4, 0.6, 0.8, 0.99\}$ of six values of parameter $\alpha$ (see Table 5) were combined for each instance. Thus there are in total $4\times5\times7\times6 = 840$ data sets to run in the computational experiments.

Appendix C.1 details the weight sensitivity algorithm of Jones (2011) used to obtain the weights in Table 2 for the achievement function of model (1)–(15). The algorithm is designed to give a selection of sufficiently diverse solutions for consideration by the decision maker. Searching each $v_t^i$ with a greater degree of granularity improves the accuracy of the exploration but also involves increasing the number of subproblems (C.1)–(C.5) to be solved from Appendix C.1.1, then is considered equal weights for all cost functions $f_t^i(\underline{x})$, $t \in T$, for each $i \in I$. This consideration produced a sample with the seven weight vectors in Table 2 from the weight space $W \subset \mathbb{R}^{|I|\times|T|}$ in order to examine the sensitivity of the model from varying the parameter $\alpha$. The values for the parameter $\alpha$ were selected in a sense that $\alpha = 0.01$ gives the complete emphasis on optimization and $\alpha = 0.99$ gives complete emphasis on balance for the proposed extended goal programming. Appendix C.2 explains how to use the *Ideal* and *Anti-ideal* objective vectors to obtain each target value $b_t^i$ for the artificial instances.

Note that for the second stage of production, the proposed model is free of the

| class | very small size | | | | small size | | | | medium size | | | | large size | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $\lvert P_k\rvert$ | $\lvert T\rvert$ | $\lvert M\rvert$ | $\lvert K\rvert$ | $\lvert P_k\rvert$ | $\lvert T\rvert$ | $\lvert M\rvert$ | $\lvert K\rvert$ | $\lvert P_k\rvert$ | $\lvert T\rvert$ | $\lvert M\rvert$ | $\lvert K\rvert$ | $\lvert P_k\rvert$ | $\lvert T\rvert$ | $\lvert M\rvert$ | $\lvert K\rvert$ |
| | 2 | 5 | 2 | 2 | 3 | 10 | 2 | 2 | 7 | 7 | 3 | 3 | 5 | 20 | 2 | 2 |

Table 1: Data of instances

| $W$ | $v_t^1$ | $v_t^2$ | $v_t^3$ | $v_t^4$ | $v_t^5$ | $v_t^6$ |
|---|---|---|---|---|---|---|
| $w^0$ | 0.167 | 0.167 | 0.167 | 0.167 | 0.167 | 0.167 |
| $w^1$ | 0.3 | 0.1 | 0.3 | 0.1 | 0.1 | 0.1 |
| $w^2$ | 0.1 | 0.3 | 0.3 | 0.1 | 0.1 | 0.1 |
| $w^3$ | 0.1 | 0.1 | 0.5 | 0.1 | 0.1 | 0.1 |
| $w^4$ | 0.1 | 0.1 | 0.1 | 0.3 | 0.1 | 0.3 |
| $w^5$ | 0.1 | 0.1 | 0.1 | 0.1 | 0.3 | 0.3 |
| $w^6$ | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.5 |

Table 2: Weights obtained from the weight sensitivity algorithm

cutting dimension. Here, the one-dimensional cutting stock problem was the choice for computational experiments. Thus, let $L^o$ and $\ell^p$ be the lengths of object type $o \in O_m$ and piece type $p \in P_k$, respectively. In addition, let the cardinality of $O_m$ be one ($|O_1| = 1$ and $|O_2| = 1$), that means the machine $m$ produces an unique object $o$ of length $L^o$. Moreover, the parameters $Q_w$ and $Q_y$ are set to the maximum possible amount of pieces and objects that can be produced in each period with the given capacity, i.e., the difference between the capacity available and a setup time is divided by the production time to obtain $Q_w$ and $Q_y$ considering the pieces and objects, respectively.

The dataset used to generate the instances is based on some data from the literature of the lot-sizing and cutting stock problems. More specifically, it is based on some instances proposed by Trigeiro et al. (1989) (to lot-sizing problem) and the CUTGEN1 generator proposed by Gau & Wäscher (1995) (to cutting stock problem). The parameter values were generated randomly in intervals with a uniform distribution as follows.

- setup cost for object: $sc^o_{kmt}$: $\in [3,4]$; - production cost for object: $vc^o_{kmt} = 1$;
- holding cost for object: $hc^o_{kt} \in [0,2]$; - demand for object: $D^o_{kt} = 0$;
- setup time of object: $Se^o_{kmt} \in [20,90]$; - production time of object: $Ve^o_{kmt} = 1$;
- production capacity to produce the objects: $CapO_{mt} \in [100,2000]$;
- setup cost for cutting pattern: $u^o_{jk} \in [2000,7000]$; - cost of cutting an object: $c^o_{jk} = 1$;
- holding cost for piece: $hc^p_{kt} \in [0.4,2]$; - demand for piece: $d^p_{kt} \in [0,200]$;
- setup time of cutting pattern: $se^o_{jkt} \in [3,55]$;
- production time of cutting pattern: $ve^o_{jkt} = 1$; - cutting capacity: $CapP_t \in [200,2500]$;
- objects length: $L^o \in [8000,13000]$; - pieces length: $\ell^p \in [300,8000]$.

Each instance of the problem can be difficult to solve since the model becomes relatively large due to the number of parameters. The estimated values for the number of constraints and variables of the model described in Appendix B give some idea of the size and difficulty of solving each instance of the problem, which justifies the use of the approximated method developed in this paper to obtain solutions whose optimality is not guaranteed but with acceptable quality in a reasonable time.

### *4.2. An illustrative example*

To help the reader understand the integrated LSP and CSP well, we provide an illustrative example and its solution. In this example, there are three periods in the planning horizon, two different raw material types, and two different piece types. For each piece type, there are three different piece lengths: $\ell^A = 1985$, $\ell^B = 1575$, and $\ell^C = 2497$. These three pieces are cut from produced objects of lengths equal to 10000. Table 3 provides the demands, capacities and other parameter values for the illustrative example. Note that the parameters in Table 3 are time-independent. Moreover, we used the weight vector $w^0$ and $\alpha$ equal to 0.2.

Table 4 shows the relevant non-zero variable values obtained by the column generation based heuristic procedure used in this paper. We note from the LSP solution in the first stage that it produces the object of raw material 1 in periods 1 and 2 (6 and 105 units in periods 1 and 2, respectively), and produces the object of raw material 2 in periods 1 and 3 (92 and 27 units in period 1 and 3, respectively). And at the CSP solution in the second stage, all these produced objects are cut using 5 different cutting patterns (variables $Z^o_{jkt}$) that meet the demand for pieces.

| | | | $k=1$ | | | $k=2$ | | | | $k=1,2$ |
|---|---|---|---|---|---|---|---|---|---|---|
| | piece | parameter | $t=1$ | $t=2$ | $t=3$ | $t=1$ | $t=2$ | $t=3$ | parameter | $t=1,2,3$ |
| | | | | | | | | | $vc^o_{kmt}$ | 1 |
| | | $sc^o_{kmt}$ | | 4080 | | | 3300 | | $Ve^o_{kmt}$ | 1 |
| LSP | | $hc^o_{kt}$ | | 0.8 | | | 0.9 | | $CapO_{mt}$ | 200 |
| | | | | | | | | | $b^1_t$ | 109 |
| | | $Se^o_{kmt}$ | | 55 | | | 47 | | $b^2_t$ | 21 |
| | | | | | | | | | $b^3_t$ | 2473 |
| | $p=A$ | | 6 | 94 | 60 | 10 | 90 | 40 | | 0.8 |
| | $p=B$ | $d^p_{kt}$ | 26 | 100 | 50 | 70 | 83 | 35 | $hc^p_{kt}$ | 1.2 |
| | $p=C$ | | 0 | 80 | 120 | 70 | 60 | 110 | | 0.9 |
| | | | | | | | | | $c^o_{jk}$ | 1 |
| CSP | | $u^o_{jk}$ | | 2040 | | | 2900 | | $ve^o_{jkt}$ | 1 |
| | | | | | | | | | $CapP_t$ | 160 |
| | | | | | | | | | $b^4_t$ | 7400 |
| | | $se^o_{jkt}$ | | 13.4 | | | 15.2 | | $b^5_t$ | 44 |
| | | | | | | | | | $b^6_t$ | 792 |

Table 3: Demands, capacities and parameter values for the illustrative example

| | $Y^o_{kmt}$ | | $X^o_{kmt}$ | | $S^o_{kt}$ | |
|---|---|---|---|---|---|---|
| $k=$ | 1 | 2 | 1 | 2 | 1 | 2 |
| $t=1$ | 1 | 1 | 6 | 92 | | 26 |
| $t=2$ | 1 | | 105 | | 34 | |
| $t=3$ | | 1 | | 27 | | |

| | $W^o_{j1t}$ | | | | | $W^o_{j2t}$ | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $j=$ | 1 | 2 | 3 | 4 | 5 | 1 | 2 | 3 | 4 | 5 |
| $t=1$ | | | | 1 | | | | 1 | 1 | 1 |
| $t=2$ | 1 | 1 | 1 | | | 1 | | | | |
| $t=3$ | | | 1 | | 1 | | | 1 | | |

| | $Z^o_{j1t}$ | | | | | $Z^o_{j2t}$ | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $j=$ | 1 | 2 | 3 | 4 | 5 | 1 | 2 | 3 | 4 | 5 |
| $t=1$ | | | | 6 | | | | 33 | 10 | 23 |
| $t=2$ | 30 | 21 | 20 | | | 26 | | | | |
| $t=3$ | | | 30 | | 4 | | | 27 | | |

| | $X^p_{1t}$ | | | $X^p_{2t}$ | | | $S^p_{1t}$ | | | $S^p_{2t}$ | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $p=$ | 1 | 2 | 3 | 1 | 2 | 3 | 1 | 2 | 3 | 1 | 2 | 3 |
| $t=1$ | 6 | 30 | | 33 | 165 | 132 | | 4 | | | 118 | 62 |
| $t=2$ | 150 | 126 | 80 | 130 | | | 56 | 30 | | 40 | 35 | 2 |
| $t=3$ | 4 | 20 | 120 | | | 108 | | | | | | |

| | $f^1_t$ | | $f^2_t$ | | $f^3_t$ | | $f^4_t$ | | $f^5_t$ | | $f^6_t$ | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $n^1_t$ | $p^1_t$ | $n^2_t$ | $p^2_t$ | $n^3_t$ | $p^3_t$ | $n^4_t$ | $p^4_t$ | $n^5_t$ | $p^5_t$ | $n^6_t$ | $p^6_t$ |
| $t=1$ | 11 | | | 2.4 | | 4907 | | 7886 | | 109.8 | | 9948 |
| $t=2$ | 4 | | | 6.2 | | 1607 | | 8590 | | 105.6 | | 8228 |
| $t=3$ | 82 | | 21 | | | 827 | 6156 | | 44 | | | 6188 |

Table 4: Variables values of the example

We also recall that the six objective functions model the conflicting production sectors, with associated stakeholders, in this approach so that each one has a specific target value by period. Table 4 shows that while the target for the LSP production cost $f_t^1$ for objects is achieved in all the 3 periods in a sense "less is better" (all positive deviations take zero values), the targets of the LSP setup cost $f_t^3$ for objects and the CSP setup cost $f_t^6$ for cutting patterns are not achieved in all the 3 periods (all negative deviations take zero values). The previous comments and the obtained values for negative and positive deviations of other objective functions ($f_t^2$, $f_t^4$, and $f_t^5$) show an apparent conflict among goals in achieving the targets, i.e., the production sectors compete for resources.

*4.3. Performance profile analysis for achieving the targets*

Appendix D explains the details about the performance profiles methodology proposed by Dolan & Moré (2002) that are used in this paper to investigate the sensitivity of model from the performance of a portion of weight space $W$. This methodology is able to evaluate the results from a statistical viewpoint providing a complementary tool for the data analysis. The performance of a particular weight involves verifying if the goal is achieving the target at the solution. Thus, the choice performance metric for the goal $i$ in the period $t$ is the evaluation of relative target deviation (RTD) that, according to Appendix D.1, is the evaluation of $p_t^i/b_t^i$ from the target $b_t^i$. The series of graphs in Figures D.2–D.6 of Appendix D.2 show the performance profiles of each weight vector of Table 2. When the mathematical model is solved with a specific value of parameter $\alpha$, the weights are compared from its performance profile.

This section verifies which weight in $W$ performed best concerning the performance ratio based on RTD evaluations, which means that weight obtained solutions for a particular set of instances whose goals are closer to the targets while varying the parameter $\alpha$. Table 5 provides the best performing weight for each $\alpha$ and a class of instances by considering the following three measures: (i) the best performing weight for $\rho_w(1.5)$ is the weight that had the highest proportion of performance ratios within a factor $\tau = 1.5$. Thus it is the most robust with up to 1.5 times the best performance ratio; (ii) the best performing weight for $\rho_w(1)$ is the weight that had the highest proportion of RTD evaluations with the best performance ratio. Then the goals achieved the targets in this proportion; and (iii) the best performing weight for $\rho_w(\tau) = 0.9$ is the weight that attained 90% of the best RTDs within a smaller factor $\tau$.

Table 6 summarizes the best performing weights from Table 5. A straightforward observation of the experiments performed in this paper shows that an appropriate weight vector would be the weight $w^3$, followed by the weight $w^2$, because they appear with 46% and 24% in the table, respectively. The results of the performance profile analysis show that the weight vectors $w^2$ and $w^3$ achieve the best global performance most frequently.

| | very small size intances | | | | | | | small size intances | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $\alpha$ | 0.01 | 0.2 | 0.4 | 0.6 | 0.8 | 0.99 | 0.01 | 0.2 | 0.4 | 0.6 | 0.8 | 0.99 |
| $\rho_w(1.5)$ | | $w^1$ | $w^0$ | $w^3$ | $w^5$ | $w^3$ | $w^3$ | $w^2$ | $w^3$ | $w^2$ | $w^2$ | $w^2$ | $w^2$ |
| $\rho_w(1)$ | | $w^3$ | $w^2$ | $w^2$ | $w^6$ | $w^5$ | $w^5$ | $w^2$ | $w^3$ | $w^3$ | $w^3$ | $w^1$ | $w^2$ |
| $\rho_w(\tau) = 0.9$ | | $w^2$ | $w^6$ | $w^6$ | $w^0$ | $w^5$ | $w^0$ | $w^0$ | $w^1$ | $w^3$ | $w^3$ | $w^2$ | $w^3$ |

| | medium size intances | | | | | | | large size instances | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $\alpha$ | 0.01 | 0.2 | 0.4 | 0.6 | 0.8 | 0.99 | 0.01 | 0.2 | 0.4 | 0.6 | 0.8 | 0.99 |
| $\rho_w(1.5)$ | | $w^1$ | $w^1$ | $w^3$ | $w^3$ | $w^3$ | $w^2$ | $w^0$ | $w^3$ | $w^3$ | $w^3$ | $w^2$ | $w^3$ |
| $\rho_w(1)$ | | $w^3$ | $w^2$ | $w^1$ | $w^3$ | $w^3$ | $w^2$ | $w^2$ | $w^3$ | $w^2$ | $w^2$ | $w^2$ | $w^1$ |
| $\rho_w(\tau) = 0.9$ | | $w^1$ | $w^1$ | $w^3$ | $w^3$ | $w^3$ | $w^1$ | $w^0$ | $w^3$ | $w^3$ | $w^3$ | $w^3$ | $w^3$ |

| | all size of intances | | | | | | |
|---|---|---|---|---|---|---|---|
| | $\alpha$ | 0.01 | 0.2 | 0.4 | 0.6 | 0.8 | 0.99 |
| $\rho_w(1.5)$ | | $w^1$ | $w^3$ | $w^3$ | $w^3$ | $w^2$ | $w^3$ |
| $\rho_w(1)$ | | $w^3$ | $w^3$ | $w^1$ | $w^2$ | $w^2$ | $w^1$ |
| $\rho_w(\tau) = 0.9$ | | $w^1$ | $w^3$ | $w^3$ | $w^3$ | $w^3$ | $w^3$ |

Table 5: Comparing the weights from the variation range of $\alpha$

| | # the best performing weight | | | | | | |
|---|---|---|---|---|---|---|---|
| | $w^0$ | $w^1$ | $w^2$ | $w^3$ | $w^4$ | $w^5$ | $w^6$ |
| very small | 3 | 1 | 3 | 4 | 0 | 4 | 3 |
| small | 1 | 2 | 8 | 7 | 0 | 0 | 0 |
| medium | 0 | 6 | 3 | 9 | 0 | 0 | 0 |
| large | 2 | 1 | 5 | 10 | 0 | 0 | 0 |
| all classes | 0 | 4 | 3 | 11 | 0 | 0 | 0 |
| proportion | 0.07 | 0.16 | 0.24 | 0.46 | 0.00 | 0.04 | 0.03 |

Table 6: Global performance of a portion of weight space $W$

Considering the corresponding weight vectors in Table 2, a certain degree of similarity is observed since the weights differ only for goals $f_t^2$ and $f_t^3$. However, $w^2$ gives 0.3 weighting to goals $f_t^2$ and $f_t^3$ while $w^3$ gives 0.1 weighting to goal $f_t^2$ and 0.5 weighting to goal $f_t^3$. The global performance of the weight vectors $w^2$ and $w^3$ along with the previous comment indicate that the conflict among the objective functions is lower when the total weighting given to the set of goals $f_t^2$ and $f_t^3$ is 50% larger than the total weighting given to the set of goals $f_t^1$, $f_t^4$, $f_t^5$ and $f_t^6$.

*4.4. Parameter sensitivity analysis*

This section examines the balance between optimization and balance of the achievement function (1) from varying $\alpha$ at the extremes of its range. Here, the complete emphasis on optimization corresponds to $\alpha = 0.01$, and the complete emphasis on balance corresponds to $\alpha = 0.99$.

Let $CT^{(\alpha)} = \sum_{i \in I} \sum_{t \in T} f_t^i(\underline{x})$ be the estimation of production total cost at the solution $\underline{x}$ when the model is solved with a specific $\alpha$. In terms of production planning, the decision maker can evaluate each $CT^{(\alpha)}$ in its overall plan. $|CT^{(\alpha_2)} - CT^{(\alpha_1)}|/CT^{(\alpha_1)}$ gives the relative balance of production total costs provided by $\alpha_1$ and $\alpha_2$. Thus it is possible to evaluate the sensitivity of the model concerning production total costs by varying the parameter $\alpha$ in the achievement function of an extended goal programming approach.

Table 7 shows the average of the relative balance of the production total costs calculated for the set of instances and each weight vector of weight space $W$, where it compares the extremes $\alpha_1 = 0.01$ and $\alpha_2 = 0.99$. The basis for comparison is the complete emphasis on optimization $CT^{(0.01)}$ for which is verified the relative balance (amplitude value) from the complete emphasis on balance $CT^{(0.99)}$. Then the cell values of Table 7 register how much the variation at the extremes of $\alpha$ represents in the production total costs. Note that the production total costs vary from 4.4% to 13.6%,

| | $\vert CT^{(0.99)} - CT^{(0.01)}\vert / CT^{(0.01)}$ | | | | | | |
|---|---|---|---|---|---|---|---|
| | $w^0$ | $w^1$ | $w^2$ | $w^3$ | $w^4$ | $w^5$ | $w^6$ |
| very small | 0.096 | 0.097 | 0.086 | 0.136 | 0.088 | 0.044 | 0.055 |
| small | 0.077 | 0.099 | 0.188 | 0.080 | 0.187 | 0.195 | 0.102 |
| medium | 0.326 | 0.461 | 0.591 | 0.852 | 0.173 | 0.185 | 0.247 |
| large | 0.269 | 0.046 | 0.120 | 0.133 | 0.211 | 0.179 | 0.203 |
| average | 0.192 | 0.176 | 0.246 | 0.300 | 0.165 | 0.151 | 0.152 |

Table 7: Average of the relative balance of production total costs for the set of instances

7.7% to 19.5%, 17.3% to 85.2% and 4.6% to 26.9% for very small, small, medium, and large size instances, respectively. Then the amplitude value is more significant for medium size instances. About the different weight vectors, while the average amplitude values for all size instances indicates that $w^5$ presents a more moderate relative balance (15.1%), $w^3$ gives a more significant relative balance (30.0%).

### *4.5. Goal-to-target plots analysis for achieving the targets*

The previous section naturally compared and chose the most appropriate weights to obtain reasonable solutions for the proposed model. However, the same comparative is not suitable to analyze the set of goals because there is a compromise relationship connecting the goals. Appendix E provides the details respecting the goal-to-target plots (GTTplots) used in this section to check the goals' sensitivity from varying the parameter $\alpha$ and a portion of weight space $W$. The mentioned is an adaptation of time-to-target plots (TTTplots) methodology proposed by Aiex et al. (2007) and Reyes & Ribeiro (2018). This methodology can evaluate the results from a statistical perspective yielding a complementary tool for the sensitivity analysis of the model outputs.

The performance of a particular goal involves verifying if it is achieving the target at the solution. Here, it is taken into account the specific goal for achieving the given target rather than considering the time for achieving the given target. Thus, RTD evaluations are used, and the designed names for the plots are goal-to-target plots. When the mathematical model is solved with specific values of parameter $\alpha$ and weight $w \in W$, the characteristics of each goal $f_t^i(\underline{x})$, for all $t \in T$, is disclosed from its GTTplot in Figure 1.

The figure summarizes the following conclusions. Excepting $f_t^5$ and $f_t^6$, the goals can achieve the targets for more than half of RTD evaluations. Note that $f_t^2$ and $f_t^4$ are the most important in achieving the targets, which is above 90% of the RTD evaluations. The figure also suggests that $f_t^6$ has more difficulty in achieving the targets for

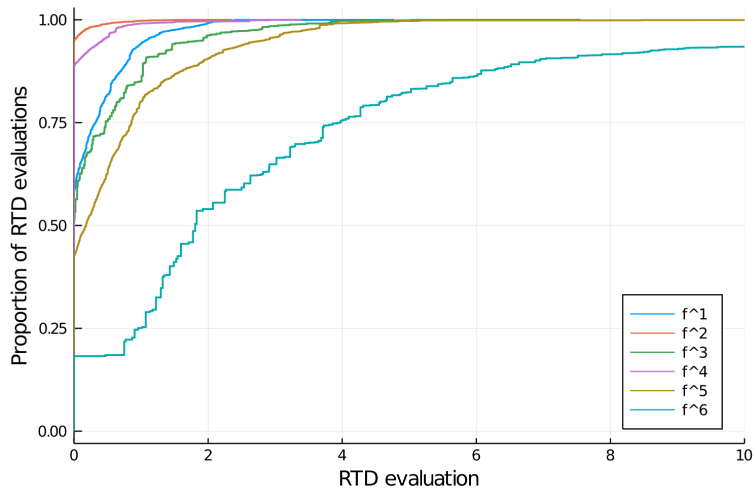


Figure 1: GTTplots for the goals

the selected collection of experiments. The appearance of curves for the goals $f_t^3$ and $f_t^6$ shows that they are most sensitive to the variation of the parameters on the part of the RTD evaluations. In fact, the sharp changes in $f_t^3$ and $f_t^6$ are presumably related to the binary variables of these functions. On the other hand, the curves for the goals $f_t^1$, $f_t^2$, $f_t^4$, and $f_t^5$ have smooth trendlines for the majority of the proportion of RTD evaluations.

The amplitude of the curves concerning to the abscissa axis exhibits the magnitude of deviations from the targets. Note that excepting $f_t^6$, the maximal deviation is two times of target values for over 90% of the proportion of RTD evaluations, and it is four times of target values for around 100% of the proportion of RTD evaluations.

The considered experiments suggest that the stakeholders could be advised to reconsider the target values for some of their goals. For example, there is an apparent conflict between the goals $f_t^2$ and $f_t^6$ regarding the holding and setup costs. The target value for the LSP holding cost could be underestimated, while that of the CSP setup cost could be overestimated. The interpretation of the GTTplots is useful for a posteriori estimation of the preference weights of the achievement function (1) and the target values since the stakeholders could then pay more attention to those more sensitive goals. Moreover, the behavior of the curves in the GTTplots reveals which goals have more difficulty in achieving their targets. Hence, a suitable parameter variation in the GTTplots helps to study the conflict among goals.

The next subsection closes the numerical experiments with statistics summarizing the results and remarks on the achievement of the column generation based heuristic method for the proposed model.

### 4.6. *Analysis of the column generation based heuristic procedure*

This section describes the overall characteristics of the problems and performance of the column generation procedure applied to the proposed extended goal programming model. Note that since all possible cutting patterns of the used instances are not known a priori, a solution obtained by the applied column generation based heuristic procedure cannot be compared with a solution of the model (1)–(15). It is important to note that this is the first study that employs this technique for a goal programming model.

Table 8 summarizes the average of relevant data collected in the numerical experiments with the column generation based heuristic procedure. The first and second columns indicate the class of instances and the weight for the achievement function. The next ten columns are as follows: #iter is the total number of iterations of the column generation procedure; $T_1(s)$ is the computational time used to solve all the linear restricted master problems; #col is the total number of columns generated in all iterations; #var and #con are the total number of variables and constraints of the integer master problem, respectively; #nod is the total number of nodes analyzed by CPLEX; $T_2(s)$ is the computational time used to solve the integer master problem; $OS(\%)$ is the percentage of instances solved to optimality found by CPLEX when solving the integer master problem with all generated columns and time-limit equal to 3600 seconds; $Gap(\%)$ is the gap found by CPLEX from the integer master problem with all generated columns; and AF is the value of the achievement function (1) of the EGP-ILCSP model.

The overall analysis of Table 8 shows that the total number of column generation iterations range on average from 7.7 to 18.5, and in general, this is mainly related to the number of pieces. It is important to note that the computational times used to solve all the linear master problems are very small for all sizes of instances (maximum value is equal to 16.5 seconds). Table 8 also shows that the complexity of the integer master problem built after all iterations of the column generation based heuristic procedure depends on the size of the instances, and the difference among them is significant. For example, the average number of columns generated, variables, and constraints is equal to 182, 524, and 317, respectively, for the very small instances. These values increase to 1324, 3528, and 1984, respectively, for the large size instances.

The collected results show that the computational time to solve the integer master problem with all generated columns is high for many instances, especially for those

| | weight | #iter | $T_1(s)$ | #col | #var | #con | #nod | $T_2(s)$ | $OS(\%)$ | $Gap(\%)$ | AF |
|---|---|---|---|---|---|---|---|---|---|---|---|
| very small size instances | $w^0$ | 7.6 | 0.8 | 180 | 520 | 315 | 1042 | 1.9 | 100 | 0 | 3.3 |
| | $w^1$ | 7.7 | 0.9 | 182 | 524 | 317 | 1847 | 2.8 | 100 | 0 | 2.2 |
| | $w^2$ | 7.6 | 0.8 | 180 | 520 | 315 | 1344 | 1.9 | 100 | 0 | 2.1 |
| | $w^3$ | 7.6 | 0.8 | 180 | 520 | 315 | 1010 | 1.6 | 100 | 0 | 2.2 |
| | $w^4$ | 7.7 | 0.8 | 183 | 527 | 318 | 1429 | 1.9 | 100 | 0 | 5.3 |
| | $w^5$ | 7.5 | 0.8 | 177 | 515 | 312 | 897 | 1.1 | 100 | 0 | 5.7 |
| | $w^6$ | 7.9 | 0.9 | 189 | 539 | 324 | 1904 | 2.1 | 100 | 0 | 8.5 |
| average | | 7.7 | 0.8 | 182 | 524 | 317 | 1353 | 1.9 | 100 | 0 | 4.2 |
| small size instances | $w^0$ | 10.5 | 3.2 | 539 | 1438 | 829 | 148569 | 1356 | 73 | 0.23 | 3.9 |
| | $w^1$ | 10.4 | 3.0 | 536 | 1433 | 826 | 200072 | 1682 | 63 | 0.91 | 2.6 |
| | $w^2$ | 10.3 | 2.8 | 532 | 1425 | 822 | 227145 | 1643 | 63 | 0.69 | 2.5 |
| | $w^3$ | 10.4 | 2.9 | 535 | 1430 | 825 | 207015 | 1830 | 56 | 1.33 | 2.6 |
| | $w^4$ | 10.3 | 3.1 | 531 | 1422 | 821 | 116530 | 1253 | 76 | 0.41 | 5.6 |
| | $w^5$ | 10.6 | 3.0 | 544 | 1449 | 834 | 70258 | 876 | 86 | 0.08 | 6.5 |
| | $w^6$ | 10.2 | 2.9 | 528 | 1417 | 818 | 252554 | 1019 | 80 | 0.28 | 8.6 |
| average | | 10.4 | 3.0 | 535 | 1431 | 825 | 174592 | 1380 | 71 | 0.60 | 4.6 |
| medium size instances | $w^0$ | 18.5 | 16.2 | 1500 | 3568 | 1885 | 127597 | 3284 | 13 | 2.93 | 7.4 |
| | $w^1$ | 18.6 | 15.9 | 1509 | 3586 | 1894 | 132204 | 3231 | 16 | 2.42 | 5.1 |
| | $w^2$ | 18.5 | 15.9 | 1508 | 3585 | 1893 | 150895 | 3020 | 26 | 2.24 | 5.0 |
| | $w^3$ | 18.5 | 15.5 | 1506 | 3580 | 1891 | 142111 | 2788 | 26 | 2.01 | 5.2 |
| | $w^4$ | 18.5 | 16.5 | 1498 | 3564 | 1883 | 189896 | 3334 | 13 | 2.81 | 11.0 |
| | $w^5$ | 18.2 | 15.1 | 1484 | 3536 | 1869 | 236649 | 3320 | 16 | 1.93 | 11.7 |
| | $w^6$ | 18.7 | 13.4 | 1512 | 3593 | 1897 | 179258 | 2934 | 23 | 2.99 | 17.0 |
| average | | 18.5 | 15.5 | 1502 | 3573 | 1887 | 165516 | 3130 | 19 | 2.48 | 8.9 |
| large size instances | $w^0$ | 13.0 | 13.3 | 1300 | 3479 | 1959 | 67711 | 3583 | 0 | 8.54 | 5.5 |
| | $w^1$ | 13.4 | 13.6 | 1336 | 3552 | 1996 | 62467 | 3593 | 0 | 8.23 | 4.4 |
| | $w^2$ | 13.4 | 13.5 | 1330 | 3530 | 1990 | 70747 | 3594 | 0 | 8.57 | 3.9 |
| | $w^3$ | 13.2 | 13.3 | 1314 | 3509 | 1974 | 78537 | 3562 | 16 | 7.68 | 4.4 |
| | $w^4$ | 13.5 | 14.5 | 1332 | 3544 | 1992 | 66086 | 3578 | 0 | 6.51 | 7.9 |
| | $w^5$ | 13.5 | 13.7 | 1334 | 3550 | 1994 | 73241 | 3421 | 10 | 6.03 | 8.4 |
| | $w^6$ | 13.3 | 12.7 | 1324 | 3529 | 1984 | 76517 | 3391 | 10 | 5.39 | 12.4 |
| average | | 13.3 | 13.5 | 1324 | 3528 | 1984 | 70758 | 3532 | 5.1 | 7.28 | 6.7 |

Table 8: Average results of the column generation based heuristic procedure

of the medium and large size. However, the heuristic used found an integer feasible solution for all proposed instances. Note also that although the gaps (*Gap*(%)) found by CPLEX are not high even for medium and large size instances, the percentage of instances (*OS*(%)) solved by CPLEX to optimality with the generated columns is small for these sizes of instances. It is important to note that the gaps found by CPLEX and the percentage of instances solved to optimality with all generated columns are reasonable measures to determine the difficulty of solving the integer master problem. However, these results cannot be used to evaluate the quality of the solutions obtained by the column generation based heuristic procedure used.

In order to evaluate the quality of the obtained integer feasible solutions, one could use the lower bound proposed by Lasdon (1970) to calculate an optimality gap (see (19), Section 3.5). We decided to not report it in our experiments because we empirically observed that this lower bound was close to zero in most of our instances, making the optimality gaps very high even for the cases in which good quality integer solutions were found. Therefore, in this paper, instead of the optimality gap, the achievement function value is used to determine the quality of the obtained feasible solutions. The results show that the achievement function values are typically small (maximum value on average is equal to 8.9), and depend on the sizes of instances and the considered preference weights. These values range from 2.1 to 8.5, 2.5 to 8.6, 5.0 to 17.0, and 3.9 to 12.4 for the very small, small, medium, and large instances, respectively. Since the achievement function provides the match of efficiency versus equity by minimizing the average deviation and the worst deviation from amongst the set of targets, it weighs two parcels of the positive deviations (the worst deviation plus the average deviation). Therefore, by considering only the positive deviations from amongst the set of targets, their small values show that the used column generation based heuristic procedure is efficient to solve the proposed formulation.

## 5. Conclusions

This paper investigates the integrated lot-sizing and cutting stock problem with parallel machines and multiple raw materials. A new general multi-objective mathematical formulation by using an extended goal programming model is presented. Distinct from the previous works that study the integrated approach, a column generation based heuristic procedure is used to solve this proposed model. Moreover, this paper adapts some well-known methodologies (performance profiles and time-to-target plots analysis) to the goal programming context. Then the trade-offs among the different well-known lot-sizing and cutting stock objective functions were analyzed using the performance profiles and goal-to-target plots analysis. Finally, the efficiency of the column generation based heuristic procedure applied to a goal programming model was also examined.

The performance profile analysis used in this paper can support the decision maker to find the most appropriate preference weights for each cost function in the achievement function. The parameter sensitivity analysis for examining the trade-off between the "best average solution" and the "most balanced solution" by computing the estimation of production total cost provided a set of relevant solutions in decision and objective space. In terms of production planning, the decision maker can evaluate each estimation of production total cost and the solution in decision and objective space to decide the overall plan. The proposed goal-to-target plots (GTTplots) analysis showed that the setup cost of objects and setup cost of pieces is most sensitive to the parameters' variation. Indeed, the plots' sharp changes are presumably related to the binary variables contained in these functions. The interpretation of the GTTplots is useful for a posteriori estimation of the preference weights of the achievement function and the target values since the stakeholders could pay more attention to those more sensitive goals. Moreover, the behavior of curves in the GTTplots reveals which goals have more difficulty or more facility in achieving their targets. Finally, the computational results also indicate that the column generation based heuristic procedure efficiently solves the extended goal programming model. However, it is challenging to solve the integer master problem for medium and large instances.

Some interesting issues can be explored as further research. For example, to advance the performance profiles and goal-to-target analysis considered in this paper to propose a complementary methodology to obtain and update appropriate weight vectors and targets. Additionally, computational tests could be carried out for other cutting dimensions.

## References


Aiex, R. M., Resende, M. G., & Ribeiro, C. C. (2007). TTT plots: a perl program to create time-to-target plots. *Optimization Letters*, *1*, 355–366.

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

Aliano Filho, A., Moretti, A. C., & Pato, M. V. (2017). A comparative study of exact methods for the bi-objective integer one-dimensional cutting stock problem. *Journal of the Operational Research Society*, (pp. 1–18).

Aliano Filho, A., Moretti, A. C., Pato, M. V., & Oliveira, W. A. (2019). An exact scalarization method with multiple reference points for bi-objective integer linear optimization problems. *Annals of Operations Research*, (pp. 1–35).

Arbel, A. (1993). Large-scale optimization methods applied to the cutting stock problem of irregular shapes. *The International Journal of Production Research*, *31*, 483–500.

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.

Bazaraa, M. S., Jarvis, J. J., & Sherali, H. D. (2008). *Linear programming and network flows*. John Wiley & Sons.

Campello, B. S., Ghidini, C. T. L. S., Ayres, A. O. C., & Oliveira, W. A. (2019). A multiobjective integrated model for lot sizing and cutting stock problems. *Journal of the Operational Research Society*, (pp. 1–13).

Cheng, C. H., Feiring, B. R., & Cheng, T. C. E. (1994). The cutting stock problem – a survey. *International Journal of Production Economics*, *36*, 291–305.

Clautiaux, F., Alves, C., Valério de Carvalho, J., & Rietz, J. (2011). New stabilization procedures for the cutting stock problem. *INFORMS Journal on Computing*, *23*, 530–545.

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

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

Dolan, E. D., & Moré, J. J. (2002). Benchmarking optimization software with performance profiles. *Mathematical programming*, *91*, 201–213.

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

Farley, A. A. (1988). Mathematical programming models for cutting-stock problems in the clothing industry. *Journal of the Operational Research Society*, *39*, 41–53.

Fiorotto, D. J., de Araujo, S. A., & Jans, R. (2015). Hybrid methods for lot sizing on parallel machines. *Computers & Operations Research*, *63*, 136–148.

Gau, T., & Wäscher, G. (1995). Cutgen1: A problem generator for the standard one-dimensional cutting stock problem. *European journal of operational research*, *84*, 572–579.

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

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

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

Glock, C. H., Grosse, E. H., & Ries, J. M. (2014). The lot sizing problem: A tertiary study. *International Journal of Production Economics*, *155*, 39–51.

Golfeto, R. R., Moretti, A. C., & Salles Neto, L. L. (2009). A genetic symbiotic algorithm applied to the one-dimensional cutting stock problem. *Pesquisa Operacional*, *29*, 365–382.

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

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

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.

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

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

Jones, D. (2011). A practical weight sensitivity algorithm for goal and multiple objective programming. *European Journal of Operational Research*, *213*, 238–245.

Jones, D., & Tamiz, M. (2010). *Practical goal programming* volume 141. Springer.

Lasdon, L. S. (1970). *Optimization Theory for Large Systems*. Macmillan Company, New York.

Leao, A. A. S., Furlan, M. M., & Toledo, F. M. B. (2017). Decomposition methods for the lot-sizing and cutting-stock problems in paper industries. *Applied Mathematical Modelling*, *48*, 250–268.

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

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

Melega, G. M., de Araujo, S. A., & Morabito, R. (2020). Mathematical model and solution approaches for integrated lot-sizing, scheduling and cutting stock problems. *Annals of Operations Research*, *295*, 695–736.

Miettinen, K. (1999). *Nonlinear multiobjective optimization* volume 12. Norwell: Kluwer Academic Publishers.

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

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.

Reyes, A., & Ribeiro, C. C. (2018). Extending time-to-target plots to multiple instances. *International Transactions in Operational Research*, *25*, 1515–1536.

Rezaei, J., & Davoodi, M. (2011). Multi-objective models for lot-sizing with supplier selection. *International Journal of Production Economics*, *130*, 77–86.

Romero, C. (2004). A general structure of achievement function for a goal programming model. *European Journal of Operational Research*, *153*, 675–686.

Saaty, T. L. (1990). How to make a decision: the analytic hierarchy process. *European journal of operational research*, *48*, 9–26.

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. (2001). Pattern generating procedure for the cutting stock problem. *International Journal of Production Economics*, *74*, 293–301.

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, Istanbul, Turkey, July 3–6* (pp. 783–793).

Sweeney, P. E., & Paternoster, E. R. (1992). Cutting and packing problems: a categorized, application-orientated research bibliography. *Journal of the Operational Research Society*, (pp. 691–706).

Toscano, A., Rangel, S., & Yanasse, H. H. (2017). A heuristic approach to minimize the number of saw cycles in small-scale furniture factories. *Annals of Operations Research*, *258*, 719–746.

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

Ustun, O., & Demirtas, E. A. (2008). An integrated multi-objective decision-making process for multi-period lot-sizing with supplier selection. *Omega*, *36*, 509–521.

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

Vanderbeck, F. (2005). Implementing mixed integer column generation. In *Column generation* (pp. 331–358). Springer.

Vanzela, M., Melega, G. M., Rangel, S., & de Araujo, S. A. (2017). The integrated lot sizing and cutting stock problem with saw cycle constraints applied to furniture production. *Computers & Operations Research*, *79*, 148–160.

Wäscher, G. (1990). An LP-based approach to cutting stock problems with multiple objectives. *European Journal of Operational Research*, *44*, 175–184.

## Appendix A. Computing bounds and attractive columns for the column generation procedure

### *Appendix A.1. Computing bounds*

This appendix explains why the dual variables related to the constraints (5) are not taken into account in the column generation procedure. This explanation is based on the proof presented by Melega et al. (2020) (which has similar situation for the integrated lot sizing and cutting stock problem considering a mono-objective problem). More specifically, note that the variables $Z^o_{jkt}$ are presented in the constraints (5), however the dual variables of these constraints are not used in the objective function of the subproblems (16)–(18). First of all, in contrast to the constraints (3), (4), (6) and (7), we observe that the constraints (5) do not exist before generating the cutting pattern $j$. In fact, for each cutting pattern $j$ the constraints (5) are created as the cutting patterns are generated. It is important to note that this lack of information in the column generation does not prevent the procedure of generating a lower bound to the problem. Indeed, considering $\delta_{jokt}$ the dual variables related to the constraints (5), the corresponding subproblems with all the constraints related to the $Z^o_{jkt}$ variables in the columns generation procedure would be given by:

$$\overline{SP^o_{kt}} = \text{Min} \quad \sum_{p\in P_k} (-\pi_{pkt} a^p_{oj}) - \gamma_t v e^o_{jkt} + \tau_{okt} - \theta_t c^o_{jk} - \delta_{jokt} \tag{A.1}$$

$$\text{s.t.} \quad \sum_{p\in P_k} \ell^p a^p_{oj} \leq L^o; \tag{A.2}$$

$$a^p_{oj} \in \mathbb{Z}_+, \quad p \in P_k. \tag{A.3}$$

By duality theory, the dual variables $\delta_{jokt}$ assume only negative values (because of the sign "$\leq$ " of the constraints (5)), thus their corresponding terms in (A.1) consist of only positive values. Therefore, there is the following relation between the objective functions of the subproblems (16) and (A.1): $SP^o_{kt} \leq \overline{SP^o_{kt}}$. This relation guarantees that, in the case of $\overline{SP^o_{kt}} < 0$, every column generated by the subproblem with all dual variables (A.1)–(A.3) can be generated by the subproblem (16)–(18). On the other hand, in the case of $\overline{SP^o_{kt}} \geq 0$, the subproblem (16)–(18) might have to generate additional columns to reach the optimality of the column generation procedure (Melega et al., 2020). An assumption in the column generation procedure regarding the use of the subproblems is that due to the similarities of the generated columns, the objective function of the restricted master problems, at optimality of the procedure, might have similar values, with both being a lower bound to the integrated problem.

*Appendix A.2. Computing attractive columns*

It is well known from the literature that linear programming problems denoted by the form $\min\{c^T x \mid Ax \geq 0,\ x \geq 0\}$ (primal problem) can be solved from the revised simplex method (Bazaraa et al., 2008). In such a method, each value $(c_j - c_B^T B^{-1} a^j)$ is referred to as *reduced cost coefficient* since it is the coefficient of the nonbasic variable $x_j$ in this reduced space. If the reduced cost coefficient is negative then the nonbasic variable $x_j$ of primal problem is a candidate to enter the master basis. In this notation, $c_B$ is the basic cost coefficient vector of the objective function, $B$ is the basis matrix, $a^j$ is a column of the corresponding nonbasic matrix to the variable $x_j$, and $c_j$ is the cost coefficient of variable $x_j$ in the objective function. Also, the vector $w^T = c_B^T B^{-1}$ contains the dual variables corresponding to the column $a^j$. Considering the relaxed problem (1)–(4) and (6)–(15) as a primal problem and the notation from Sections 2 and 3, we obtain $w := (\mathbf{w_1}, \underbrace{\mathbf{w_2}, \overline{\theta}, \mathbf{w_3}}_{(3)}, \underbrace{\overline{\pi}}_{(4)}, \underbrace{\overline{\gamma}}_{(6)}, \underbrace{\overline{\tau}}_{(7)}, \mathbf{w_4})^T$, $a^j := (\mathbf{0}, \underbrace{\mathbf{0}, \overline{c}, \mathbf{0}}_{(3)}, \underbrace{\overline{a}}_{(4)}, \underbrace{\overline{ve}}_{(6)}, \underbrace{-\mathbf{1}}_{(7)}, \mathbf{0})^T$, and $c_j := 0$, where the symbol "$\underbrace{\qquad}$" emphasizes the subset of constraints associated with the variables $Z^o_{jkt}$, the vector $(\overline{\theta}, \overline{\pi}, \overline{\gamma}, \overline{\tau})^T$ contains all the dual variables $\theta_t$, $\pi_{pkt}$, $\gamma_t$, and $\tau_{okt}$, the vector $(\mathbf{w_1}, \mathbf{w_2}, \mathbf{w_3}, \mathbf{w_4})^T$ contains all the remaining dual variables, the vector $(\overline{c}, \overline{a}, \overline{ve})^T$ contains all the parameters $c^o_{jk}$, $a^p_{oj}$, and $ve^o_{jkt}$, and $\mathbf{0}$ and $\mathbf{1}$ are vectors of the appropriate dimensions containing zeros and ones. Thus, $c_j - w^T a^j = \sum_{p \in P_k} (-\pi_{pkt} a^p_{oj}) - \gamma_t ve^o_{jkt} + \tau_{okt} - \theta_t c^o_{jk}$.

In the column generation based heuristic procedure, the knapsack subproblem (16)–(18) aims to minimize the reduced cost coefficient while respecting the physical limitations of the object to define the way that the pieces are cut (cutting patterns) from an object. And, if $\min\{c_j - w^T a^j\} < 0$, then the solution of the knapsack subproblem provides the required attractive columns. In particular, note that the coefficient of the variable $\theta_t$, the dual information of constraints (3) for the CSP cutting cost $f_t^4$, in the knapsack subproblem is equal to $c^o_{jk}$.

## Appendix B. Complexity analysis of model

Solving the problem can be a difficult task once the model becomes relatively large in relation to its parameters. This relative growth is illustrated from the computation of the total number of constraints (NC) and number of variables (NV) of the model according to the size of sets $I$, $T$, $M$, $K$, $O$, $P$, and $J_o$. Constraints (2)–(9) and (10)–(15) yield the following values for NC and NV, respectively.

$$\text{NC} = |T|(1 + |M| + 2|I| + |P|) + |K||T| \left( 2|O| + \sum_{o \in O} |J_o| \right),$$

$$\mathrm{NV} = 1 + |P| + |K||O| + |T|(2|I| + |P|) + |K||T|\left(3|O| + 2\sum_{o\in O}|J_o|\right).$$

Each $|J_o|$ is the number of possible combinations in which the piece lengths are organized into an object $o \in O$. Consider the simplest case: one-dimensional cutting stock problem, ignoring the issues of feasible combinations, and $|P|$ pieces for which there is a unit demand. To enumerate all possible combinations needs at least $2^{|P|} - 1$ different combinations, where $|P|$ is the number of required pieces. Therefore, NC and NV have exponential growth (Arbel, 1993; Suliman, 2001). The estimated values for NC and NV provide some idea of the size and difficulty of solving an instance of the problem.

## Appendix C. Setting of parameters for the computational experiments

### *Appendix C.1. Setting the weights for the achievement function*

In order to investigate the range of solutions available by varying the weights the weight sensitivity algorithm of Jones (2011) is applied to the weight $v_t^i$, $i \in I$ and $t \in T$, of the achievement function of model (1)–(15). This algorithm is designed to give a selection of sufficiently diverse solutions for consideration by the decision maker. This section provides a version of mathematical formulation for the hybrid subproblems solved by the weight sensitivity algorithm, and the computation of the targets.

#### *Appendix C.1.1. Weight sensitivity algorithm*

The weight sensitivity algorithm (WSA) for goal and multiple objective programming investigates a portion of weight space $W$ of interest to the decision maker for a weighted programming approach. In this paper one running of WSA provides a weight vector $w \in W \subseteq \mathbb{R}^\ell$ for the weighted goal programming approach that solving an instance for the EGP-ILCSP model. The exploration of weight space $W$ stars from an weight initial solution $w^0 \in W$ and solves one of the two sets of suitable hybrid subproblems presented by Jones (2011) in order to determine the limit of weight space exploration in line with the decision maker's preferences. Each subproblem is a lexicographic goal programming, for which this paper presents a rewritten version of the mathematical formulations of Jones (2011).

The decision maker gives their initial estimate of preference information $w^0$ or, for example, the pairwise comparision method of Saaty (1990) provides $w^0$. Let $L = \{1,\ldots,\ell\}$ be the index set of $w$, and $D = \{1,\ldots,i-1,i+1,\ldots,\ell\}$ be the index set of $w$ with the exception of the index $i$. For each $i \in L$, the lexicographic goal programming (LGP) is formed to find the maximum values of each weight direction for a single weight change. The LGP model, assuming pairwise comparisons are given, is rewritten as follows.

$$\text{Lex Min} \quad a = \left[ -w_i, \sum_{j=1}^{\ell-2} \sum_{k=j+1}^{\ell-1} \left( n_{jk} + p_{jk} \right) \right] \tag{C.1}$$

$$\text{s.t.} \quad \frac{w_{D(j)}}{w_{D(k)}} + n_{jk} - p_{jk} = \frac{w^0_{D(j)}}{w^0_{D(k)}}, \; j = 1, \ldots, \ell-2, \; k = j+1, \ldots, \ell-1; \tag{C.2}$$

$$\sum_{r=1}^{\ell} w_r = 1; \tag{C.3}$$

$$Aw \leq 0; \tag{C.4}$$

$$w_r, n_{jk}, p_{jk} \geq 0, \; r = 1, \ldots, \ell, \; j = 1, \ldots, \ell-2, \; k = j+1, \ldots, \ell-1. \tag{C.5}$$

The first component of the lexicographic achievement function (C.1) maximizes the value of the weight whose growth direction is restricted to the set of decision maker weight space constraints (C.2)–(C.5). Matrix $A$ in (C.4) provides the additional preference of decision maker among the weights. For example, a pairwise cardinal information regarding weights, for which the penalization of positive deviations from the $i$th goal at least half as important as the penalization of positive deviations from the $j$th goal, i.e., $w_i \geq 0.5w_j$. The second component of the lexicographic achievement function (C.1) then provides as close as possible to a correct balance (ensured by goal set (C.2)) between the ratios of the other weights in the weighting vector in the case that there exist alternative optimal solutions to the first component.

The multiplication of each constraint of (C.2) by the variable $w_{D(k)}$ provides an equivalent linear lexicographic goal programming whose the total of constraints (C.2) is $\binom{\ell-1}{2}$. The initial solution $w^0$, and the solutions of $\ell$ subproblems LGP return up to $\ell+1$ weighted vector $w$ to solve each instance of a weighted goal programming approach.

*Appendix C.1.2. Setting the weight $v_t^i$*

A version of WSA is applied to find a set of the weights $v_t^i$, $i \in I$ and $t \in T$. It consists of solving $\ell$ subproblems LGP that provides a systematic exploration of the portion of weight space $W$ of interest to the decision maker, which means in setting the parameter inputs to WSA of Jones (2011) at TMax=2 and Maxlevel=1. Searching each $v_t^i$ with a greater degree of granularity improves the accuracy of the exploration but also involves increasing the number of goal programs LGP to be solved, then is considered equal weights for all cost functions $f_t^i(\underline{x})$, $t \in T$, for each $i \in I$, thus $v_t^i = w_i$, for each $i \in I$, and for all $t \in T$. And, $\ell = 6$ in the formulation (C.1)–(C.5).

This paper considers that the penalization of positive deviations from the setup cost must be regarded as at least as important as penalization of positive deviations from the production and holding costs, which leads to constraints of the form $v_t^3 \geq v_t^1$, $v_t^3 \geq v_t^2$, $v_t^6 \geq v_t^4$, and $v_t^6 \geq v_t^5$, for all $t \in T$. In addition, each goal must have at least 10% of the absolute value of importance associated with it, and no goal must have more than 60%, i.e., $0.1 \leq v_t^i \leq 0.6$, for each $i \in I$, and for all $t \in T$.

Table 2 provides seven weight vectors containing the preference weights obtained from weight sensitivity algorithm for solving an instance of proposed EGP-ILCSP model.

*Appendix C.2. Setting the targets*

In practice the targets are supplied from decision maker, however each target value $b_t^i$ for the our artificial instances was obtained by using two typical vectors in multiobjective optimization, namely *Ideal* and *Nadir* objective vectors that getting information about the ranges of the objective function values in the Pareto optimal set.

A multiobjective optimization problem (MOP) with $n$ objective functions, $n > 1$, can be formulated as $\min\{z = F(x) : x \in X\}$, where $z_k = F_k(x)$, $k = 1, \ldots, n$, is the value of the $k$th objective function to be minimized, $z = (F_1(x), F_2(x), \ldots, F_n(x))^T$ is the vector of the objective functions, $x \in \mathbb{R}^n$ is the vector of the decision variables, and $X \subseteq \mathbb{R}^n$ is the feasible space. The Ideal objective vector $z^*$ is formed by the individual optima of each objective function of MOP subject to the constraints. The Nadir objective vector $z^{nad}$ gives the upper bounds of the objective vectors correspond to $X$, which is approximated by using the *payoff table* method described by Miettinen (1999). This approximation is called anti-ideal objective vector.

This paper uses the *relaxed* Ideal and Nadir objective vectors, where the relaxed $z^*$ is calculated by considering a MOP whose the vector of the objective functions are the functions $f_t^i(\underline{x})$, $i \in I$ and $t \in T$, the feasible space is formed by constraints (4)–(10), and the decision variables (11)–(13) have the integrality relaxed. Thus the relaxed $z^{nad}$ is approximated by the payoff table method, and for all the instances the chosen targets were $b_t^i = z_{it}^* + 0.3\left(z_{it}^{nad} - z_{it}^*\right)$, for $i = 1, 2, 4, 5$, and $t \in T$. However, due to the poor quality of relaxed setup variables (related to $f_t^3(\underline{x})$ and $f_t^6(\underline{x})$), for all the instances the chosen targets were $b_t^i = z_{it}^* + 0.6\left(z_{it}^{nad} - z_{it}^*\right)$, for $i = 3, 6$, and $t \in T$.

## Appendix D. Performance profile

*Appendix D.1. Performance profile for achieving the targets*

Dolan & Moré (2002) proposed performance profiles as a tool for comparing the performance of optimization softwares while solving a set of instances, in which the performance profile for a particular solver is the (cumulative) distribution function for a performance metric (for example, running time). This paper investigates the sensitivity of model from varying the weights. The comparison concerns the performance of a portion of weight space $W$, for which the performance of a particular weight involves to verify if the goal is achieving the target at the solution.

The series of graphs in Figures D.2–D.6 show the performance profiles of each weight vector of Table 2. When the mathematical model is solved with a specific value of parameter $\alpha$, the weights are compared from its performance profile. A required baseline for comparisons is to verify the performance of each weight so that the goals achieve the targets at the solution. The profiles represent the cumulative distribution functions for the performance ratio based on a particular metric. The goal achieves the target when the unwanted deviations are minimized at the solution. Therefore, the choice performance metric for the goal $i$ in the period $t$ is the evaluation of relative target deviation $p_t^i/b_t^i$ from the target $b_t^i$.

Let $P$ be the set of instances of the proposed EGP-ILCSP model, $Q = P \times I \times T$ be the set of indexes $\{p,i,t\}$, for $p \in P$, $i \in I$ and $t \in T$, and $W$ be the set of weights for a column generation algorithm experimentation. For each $\{p,i,t\} \in Q$ and weight $w \in W$, $\gamma_{pit}^w$ is the performance metric $p_t^i/b_t^i$ obtained by solving the instance $p$ with the weight $w$. Then for each set of instances $P$ and set of weights $W$, $\gamma_{pit}^w$ is the entry of the matrix $\Gamma$ that has $m_Q = |P||I||T|$ rows and $|W|$ columns.

The following performance ratio $r_{pit}^w = \gamma_{pit}^w/(\min\{\gamma_{pit}^w \mid w \in W\})$ compares the weights with each other with respect to set of instances $P$, where the performance metric of the instance $p$, goal $i$, and period $t$ by utilizing the weight $w$ is compared with the best performance metric among all the weights of $W$. The overall assessment of the performance of the weight $w$ is given by $\rho_w(\tau) = (1/m_Q)\text{size}\{\{p,i,t\} \in Q \mid r_{pit}^w \leq \tau\}$. Thus, $\rho_w(\tau)$ is the probability for the weight $w \in W$ for which a performance ratio $r_{pit}^w$ is within a factor of $\tau \in \mathbb{R}$ of the best possible ratio. Note that the function $\rho_w(\tau)$ is the cumulative distribution function for the performance ratio; the plot of the performance profile discloses the major performance characteristics; weights with large probability $\rho_w(\tau)$ are to be preferred; $\rho_w(1)$ is the proportion that the weight $w$ outperforms (the best performing weight) the other weights within $m_Q$ evaluations of the relative target deviation (RTD); and for a fixed value of $\tau$, $\rho_w(\tau)$ quantifies the most robust weight.

*Appendix D.2. Plots of performance profiles*

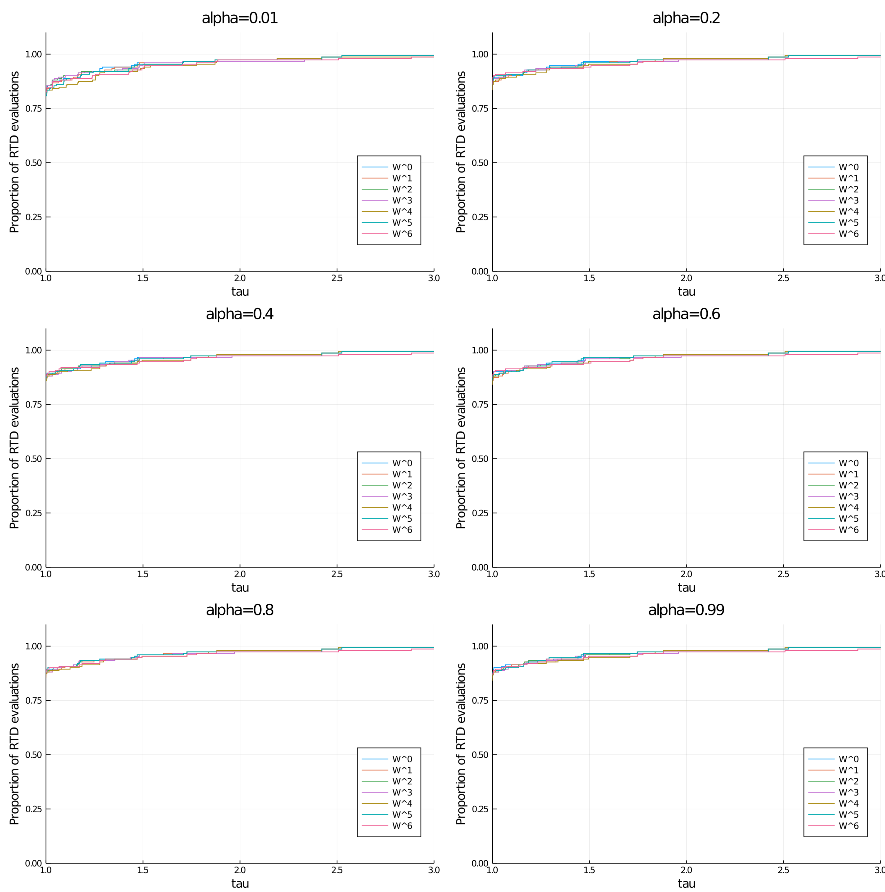


Figure D.2: Performance profiles for very small size instances

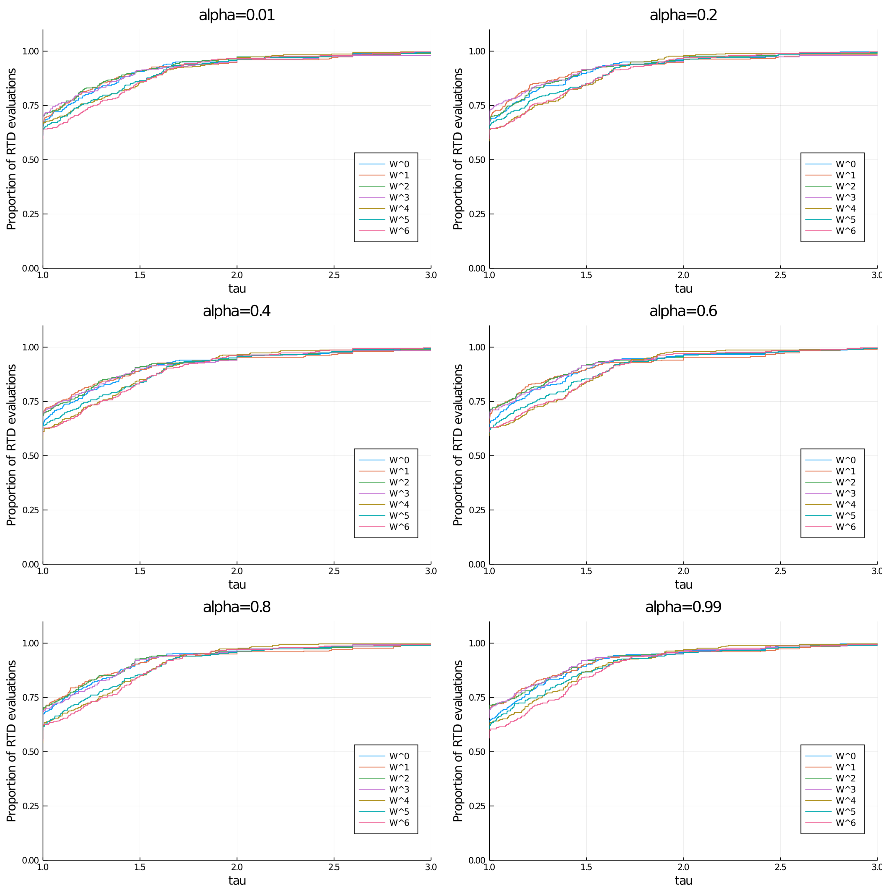


Figure D.3: Performance profiles for small size instances

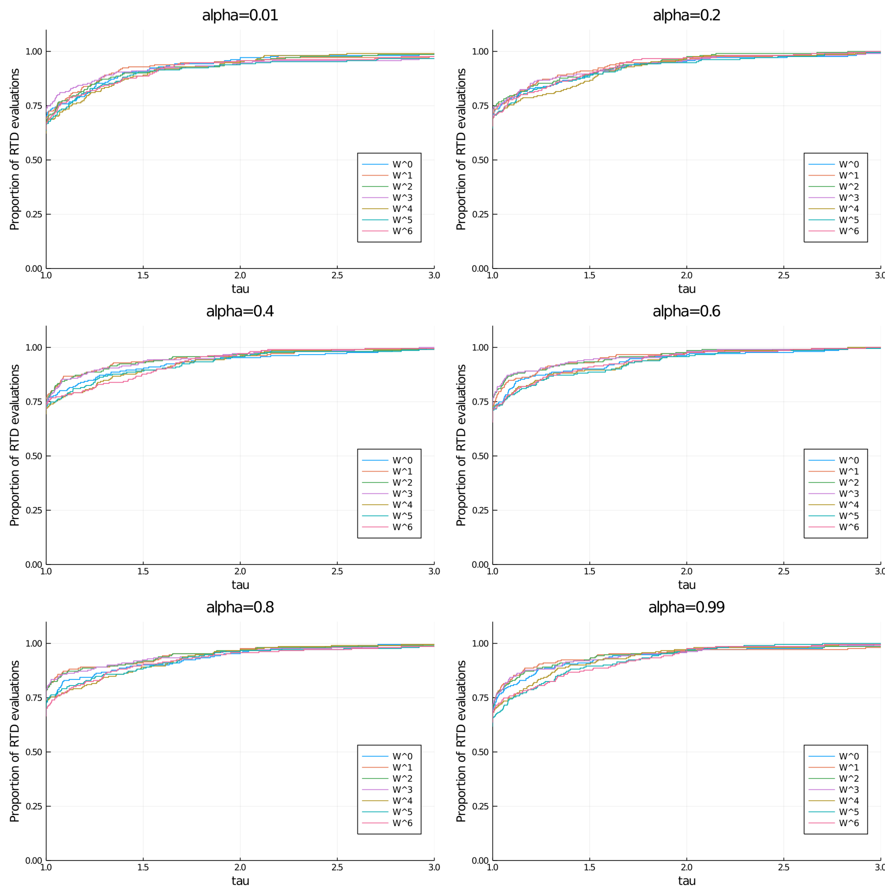


Figure D.4: Performance profiles for medium size instances

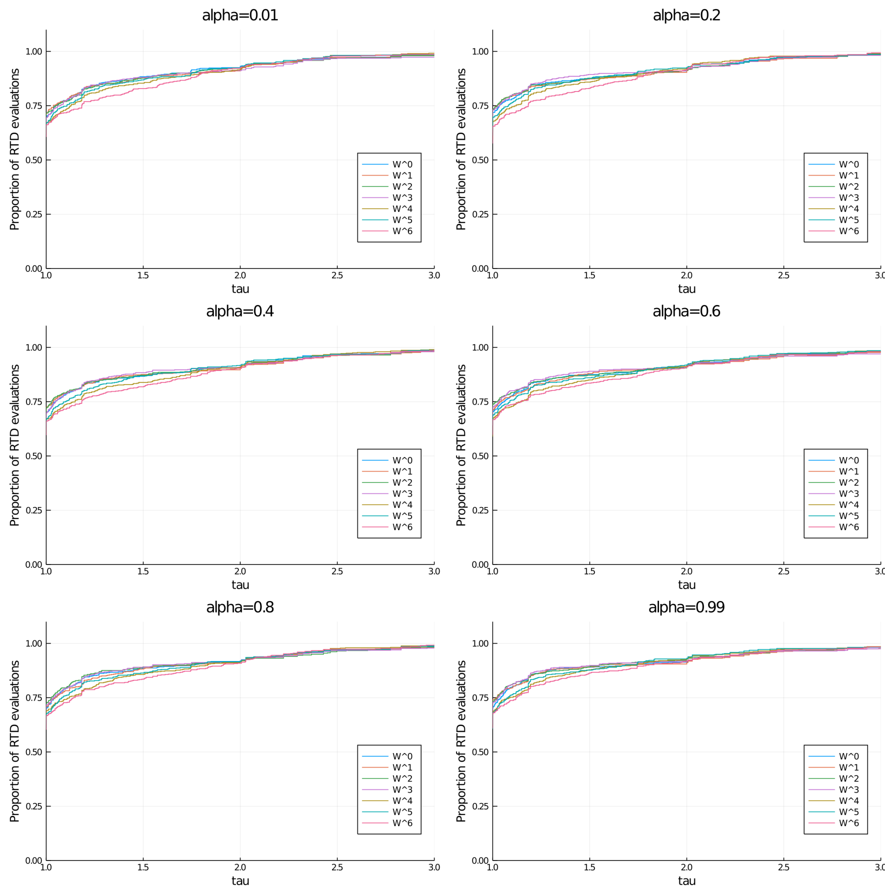


Figure D.5: Performance profiles for large size instances

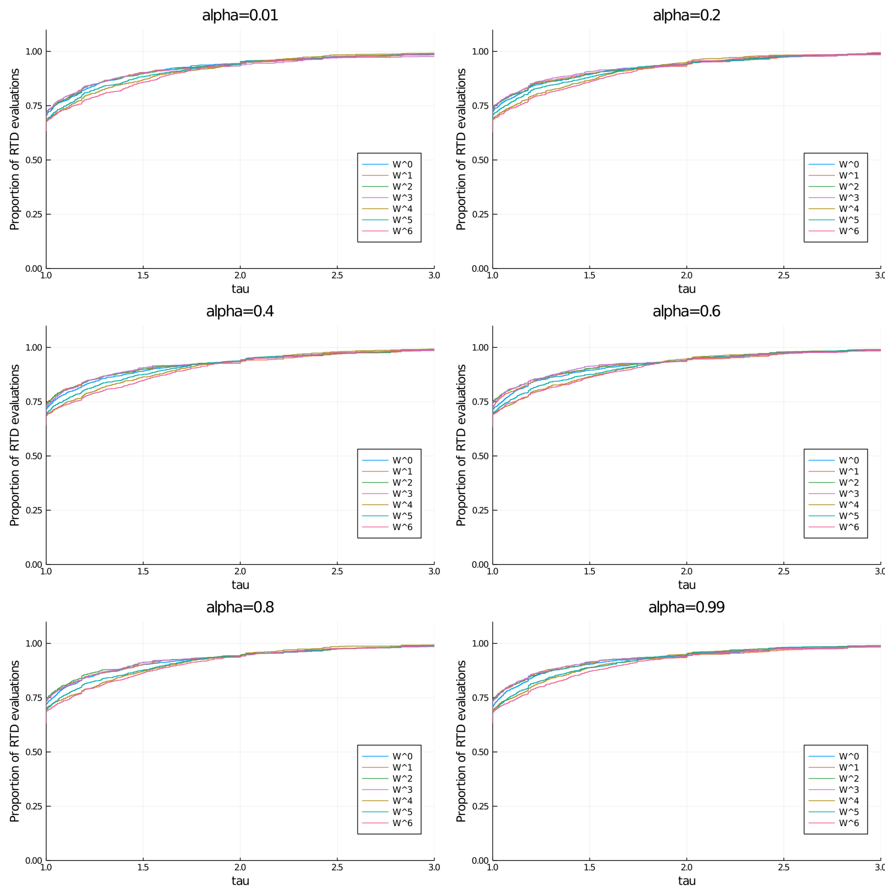


Figure D.6: Performance profiles for all size of instances

## Appendix E. Goal-to-target plots

Aiex et al. (2007) developed a code to create the known time-to-target solution plots as a tool for comparison of different algorithms or strategies for solving a given combinatorial problem in terms of CPU times. Reyes & Ribeiro (2018) extended these ideas to sets of multiple instances. Time-to-target plots (TTTplots) display on the ordinate axis the probability which an algorithm finds a solution that is at least as good as a given target value within a given running time. The latter is shown on the abscissa axis (time to sub-optimal solution).

This paper considers an adaption of the TTTplots to investigate the sensitivity of the model from varying the parameter $\alpha$ and a portion of weight space $W$. The performance of a particular goal involves verifying if it is achieving the target at the solution.

Here, it is taken into account the specific goal for achieving the given target rather than considering the time for achieving the given target. Thus, the designed names for the plots are goal-to-target plots (GTTplots). The plots in Figure 1 show the proposed GTTplots of goals $f_t^1(\underline{x}), \ldots, f_t^6(\underline{x})$ for all $t \in T$. When the mathematical model is solved with specific values of parameter $\alpha$ and weight $w \in W$, the characteristics of each goal is disclosed from its GTTplot.

A baseline used for comparison verifies how the output of each goal is close to the target. The goal achieves the target when the unwanted deviations are minimized at the solution. Therefore, the choice output metric for this approach is the same as in Appendix D.1 for the performance profiles. Thus the relative target deviation (RTD) for the goal $i$ in the period $t$ is the evaluation of $p_t^i/b_t^i$ in relation to the target $b_t^i$. Unlike comparison from the performance profiles, which provided an internal comparison among the different weights by using the performance ratio of RDTs, the GTTplot represents the cumulative distribution function for an individual goal from the absolute value of output metric RDT.

Let $P$ be the set of instances of the proposed EGP-ILCSP model, $R = W \times P \times T$ be the set of indexes $\{j, p, t\}$, for $j = 1, \ldots, |W|$, $p \in P$, $t \in T$, and $W$ be the set of weights ($w^j \in W$) for a column generation algorithm experimentation. For each $\{j, p, t\} \in R$ and $i \in I$, $\theta^i_{jpt}$ is the output metric $p_t^i/b_t^i$ obtained by solving the instance $p$ with the weight $w^j$. Then for each set of instances $P$ and set of weights $W$, $\theta^i_{jpt}$ is the entry of the matrix $\Theta$ that has $m_R = |W||P||T|$ rows and $|I|$ columns, for which the RTD evaluations in each column are sorted in ascending order.

The overall assessment of the output of each goal $i$ is given by $\sigma_i(\theta) = (1/m_R)\text{size}\{\{j, p, t\} \in R \mid \theta^i_{jpt} \leq \theta\}$. Thus, $\sigma_i(\theta)$ is the proportion for which $\theta^i_{jpt}$ is within a factor of $\theta \in \mathbb{R}$ of the best possible RTD evaluation for the goal $i \in I$, i.e., $\min\{\theta^i_{jpt} \mid \{j, p, t\} \in R\}$. Note that the function $\sigma_i(\theta)$ is the cumulative distribution function for the RTD evaluation; the GTTplot discloses the major output characteristics; $\sigma_i(0)$ is the proportion that the goal $i$ achieved the target within $m_R$ evaluations of RTD; and for a fixed value of $\theta$, $\sigma_i(\theta)$ quantifies the robustness of the goal $i$ in achieving the target within a factor of $\theta \in \mathbb{R}$.

The GTTplots display on the ordinate axis the cumulative distribution function that the column generation experimentation finds a solution taking each goal value less than or equal to a given target within a fixed value of $\theta$. The latter is shown on the abscissa axis, which, in this case, is the RTD evaluations sorted in ascending order.