Journal of Information & Optimization Sciences
ISSN 0252-2667 (Print), ISSN 2169-0103 (Online)
Vol. 47 (2026), No. 6, pp. 2491–2510
DOI : 10.47974/JIOS-1970

A three-objective mathematical model for a one-dimensional cutting stock problem with trim loss recovery

Boualem Slimi *
Department of Mathematics
Faculty of Sciences
University Saad Dahlab
Blida, Soumma Road
BP 270 Blida (09000)
Algeria

Redha Sakri †
Department of Urban Hydraulics
National High School for Hydraulic (ENSH)
National Road No. 29, Soumaa
BP 31
Blida (09000)
Algeria

Abstract

In this study, we present a mathematical model with three objectives, by which we maximize the number of standard items in stock, by recovering the materials lost in the cutting process by welding, and minimize the total waste of raw materials and the number of modes to be carried out. We also propose a solution technique consisting of two steps, the first of which is to create the cutting matrix and the second of which is to find all possible solutions that satisfy the orders and from them all the efficient solutions, thanks to a subset of these cutting modes. In both steps, we recover lost materials through welding and form new standard items. Each solution features the number of standard items recovered by total waste weld, number of cutting modes, and total waste.

Subject Classification: *90C29.*

Keywords: *Standard items recovered, Total waste of raw material, Possible cutting modes, Three-objectives cutting stock problem.*

* *ORCID-ID:* `0000-0001-6056-0286`
† *ORCID-ID:* `0000-0002-9750-4357`

* *E-mail:* **Boualemslimi@yahoo.com (Corresponding Author)**
† *E-mail:* **r.sakri@ensh.dz**

1. Introduction

The cutting stock problems remain of great interest to researchers in the field of operations research due to several factors, including their wide range of applications and extensions, such as reproducing standard items by welding missing materials from previous cutting operations, the number of installations, or reducing the number of objects in the inventory, or inventory costs, etc. Among these extensions, Cui and Yan studied a single-cutting stock problem using multi-objective optimization, which considered: the final costs of the final product, the benefit gained from the excess length of unused sheet metal when it exceeds a maximum, and the benefit gained from the benefits resulting from previous cutting operations. To solve this problem, they proposed a method consisting of an analytical technique that cuts a portion of the demand and a heuristic approach that cuts the remaining demand [1]. Campello *et al*., also proposed a multi-objective model of the one-dimensional cutting problem with the following objectives in mind: reduce total costs, inventory costs, costs incurred, and reduce total material loss and object inventory costs [2].

While we wanted in this work to present a study of the one-dimensional cutting stock problems with three objectives and propose a solution approach. The importance of this study in particular lies in maximizing the standard items by recovery the waste, and in minimizing waste to the maximum extent, firstly as an objective function to minimize the waste of raw materials and secondly by waste recovery. Where the objectives are, maximizing the standard items recovered from the remains of previous cutting operations, minimize the waste and minimize the number of installations. To achieve this, we assume that we have a stock of identical standard items to fulfill the order, we cut them or part of them into small parts according to the order, and this cutting operation is carried out according to the cutting modes that are generated according to a certain cutting plan. Using set of these modes, we form cutting strategies. While generating cutting modes and building cutting strategies, we recover the wastes and weld them to form new standard items that add to the stock of standard items. We also present an approach for the solution, which consists of a heuristic method for generating cutting modes as an alternative to the column generation technique of Gilmore and Gomory [3] and modeling the cutting modes in the form of a graph to create cutting strategies. We also demonstrate the effectiveness of this proposed heuristic in providing solutions to the problem, through the empirical study described below. The remainder of the article is organized as follows: In the next section, we provide a detailed explanation of the proposed model, while we devote the third paragraph to presenting the solution procedures represented by several algorithms through which the proposed model is solved. As for the fourth section, we devoted it to the results and discussion, and then we conclude this work with the most important results obtained.

2. Literature Review

Previously, classical studies of cutting stock problems were limited to trim loss minimization. Now, thanks to the use of multi-objective optimization, it has been expanded to include studying a variety of objectives at the same time. Such as minimizing waste and maximizing profit, or minimizing waste and minimizing the number of standard items. In the early sixties of the last century, more precisely between 1961 and 1965, Gilmore and Gomory developed a method for solving cutting problems used in industry, the column generation technique, based on linear programming methods for solving this type of problem [4, 5]. This method was later modified, with Haessler using a more restrictive formula to control the generation of cutting modes [6]. Farley also made some improvements to it, which involved generating an initial solution, with the aim of making it more practical [7]. Umitani *et al*., proposed a technique that uses two types of search operations, namely single-neighbor addition and neighbor displacement, respectively, to solve a single-dimensional cutting problem with a fixed number of poses, in order to reduce the number of parts in the inventory while limiting the number of different part models within user-defined limits [8]. While Poltronier *et al*., studied the integrative problem of determining parts inventory and order sizes in the sheet metal industry, with the aim of receding the costs of determining parts inventory, installation, waste, and finished product inventory. They used integer linear programming and two heuristics to solve this proposed model [9]. De Armas *et al*., proposed a model for the cutting problem represented by a model with two objectives: total benefit and number of settings. To solve this model, they proposed three algorithms: genetic algorithm, Pareto algorithm, and evolutionary algorithm [10]. Henn and Wäscher demonstrated in a classification study of cutting problems showed that cutting stock problem with setups can be replaced by a corresponding bi-objective optimization model, these results in a vector optimization model based on single stock size cutting stock problem inputs with setups [11]. Araujo *et al*., used a heuristic technique utilizing concepts of genetic algorithms, and proposed a solution to the one-dimensional cutting stock problem with installations, taking into account two objective functions: minimizing the number of standard items and the number of different cutting modes used [12]. To solve a one-dimensional cutting stock problem with installations, Cui *et al*., used a mathematical model with two objectives: reduce primary elements used in cutting and installation costs for a given mode set. They proposed an algorithm for generating cutting mode sets to solve this problem. They then described a sequential assembly procedure for generating modes in the set [13]. Liao *et al*., have proposed three mathematical models: a cutting pattern model, a time-period model, and a cutting machine model to study the integrated cutting problem, which is a one-dimensional cutting problem and to determine the sizes of sheets used in various industries. To solve these proposed models, they developed two algorithms: the first is a heuristic technique using the column generation method, and the second is a column generation method. In a comparative study of these models, the cutting machine model is the best in terms of real data [14]. While Filho *et al*., proposed a mathematical model for the one-dimensional cutting problem, which is a two-objective cutting model: the first is the installation costs and the second is the number of different cutting positions. They also proposed several solution algorithms, such as the weighted sum technique and some improved versions of the Chebyshev metric, with the goal of generating all efficient solutions. [15]. Meloli *et al*., also modeled a one-dimensional cutting problem as a one-dimensional cutting problem with the two goals of reducing both the waste and the number of settings. They also proposed a solution method in the form of a genetic algorithm that gives a vector of efficient solutions [16]. Campello *et al*., also proposed a multi-objective model of the one-dimensional cutting problem with the following objectives in mind: reduce total costs, inventory costs, costs incurred, and reduce total material loss and object inventory costs [2].

3. Model building and solving

Definition 3.1: A recovered standard object is any object formed from cutting scraps and whose size is equal to that of the original standard object.

Definition 3.2: A possible cut mode is a set of cut parts of a standard item and the location of each within the standard item.

Definition 3.3: A possible cut strategy is a set of possible cut positions to meets different types of orders.

3.1 *Modeling the problem*

In order to construct a three-dimensional mathematical model for the one-dimensional cutting stock problem with cutting waste recovery, we assume that we have a set of standard items of the same sizes, where we cut a standard items of size S into a set of parts of sizes $s_h < S$, to meet g different order sizes noted by o_h. On the other hand, we consider the number of installations by entering the variable $\psi(w_t)$ which equal to 1 when $w_t > 0$, meaning that it has moved from one installation to another new installation, is equal to 0 when $w_t = 0$, meaning that no transition has taken place from one installation to a new installation. Where t = 1 … r. Therefore, the cost to be minimized in the case of installations is: $\sum_{t=1}^{r} \psi(w_t)$. At the same time, we recover the waste resulting from generating the cutting mode, which is the size of the standard item from which we subtract the size used to generate the cutting mode and denote it as R_h. As for the parts h in excess of demand resulting from the use of cutting strategy t, they are the same as the parts required, but they are in excess of order, and are considered a waste, we denote it as q_{ht}. Welding these waste gives standard items y_t, where we denote by $\gamma(y_t)$ the boolean variable equal to 1 when $y_t > 0$, i.e. that is, the retrieved standard object has been formed is equal to 0 when $y_t = 0$, i.e. That is, no standard item was retrieved, thus the objective function at each stage becomes maximized $\sum_{t=1}^{r} \gamma(y_t)$.

Thus, the cutting model can be constructed as follows:

Model parameters and decision variables

- S be the size of standard item (body),
- s_h be the size of order part h, ($h = 1, \dots, g$, where g number of part types),

- o_h be the order quantity of the part type h, $(h = 1 \dots, g)$,
- R_h be the waste when using the cutting mode h,
- C_h be the quantity of waste when using cutting mode h,
- q_{ht} be the number of parts of type h in excess of the order in the cutting modes used in cutting strategy t,
- $s_h^{'}$ be the size of the parts exceeding the order of type h,
- r be the number of cutting modes,
- $L_k = \{R_h,\ s_h^{'}\}$ set of waste sizes, k = 1, 2…
- $Q_l = \{C_h,\ q_{ht}\}$ set of waste quantity,
- a_{ht} be the frequency of the h^{th} part in the t^{th} mode,
- P_t be the cutting strategies.
- w_t be the frequency of the t^{th} cutting mode is used, t = 1,…, r,
- y_t be the number of standard items recovered at each step, t = 1,…, r,

Let us have three objective functions:

- Total waste

$$\text{Minimize}\big(f_1(w)\big) = \text{Minimize}\ \big(S \times \textstyle\sum_{t=1}^{r} w_t - \sum_{h=1}^{g} s_h o_h\big) \tag{1}$$

- number of cutting modes

$$\text{Minimize}\big(f_2(w)\big) = \text{Minimize}(\textstyle\sum_{t=1}^{r} \psi(w_t)) \tag{2}$$

- Number of standard items retrieved

$$\text{Maximize}\ \big(f_3(y)\big) = \text{Maximize}\big(\textstyle\sum_{t=1}^{r} \gamma(y_t)\ \big) \tag{3}$$

Model constraints:

- $\sum_{t=1}^{r} a_{ht} w_t \geq o_h\ h = 1, \dots, g$ (4)
- $w_t \in \mathbb{N}\ t = 1, \dots, r$ (5)
- $y_t \in \mathbb{N}\ t = 1, \dots, r$ (6)
- $\psi(w_t) = \begin{cases} 1 \text{ if } w_t > 0 \\ 0 \text{ if } w_t = 0 \end{cases}$ (7)
- $\sum_{h=1}^{r}(R_h + s_h^{'}) = S$ (8)
- $\gamma(y_t) = \begin{cases} 1 \text{ if } y_t > 0 \\ 0 \text{ if } y_t = 0 \end{cases}$ (9)

Based on the previous notices, we formulate the proposed model, which is to maximize the number of recovered items, minimize the number of installations, and also minimize waste, as follows:

$$\text{Minimize}\big(f_1(w)\big) = \text{Minimize}\ \big(S \times \textstyle\sum_{t=1}^{r} w_t - \sum_{h=1}^{g} w_h o_h\big) \tag{10}$$

$$\text{Minimize}\big(f_2(w)\big) = \text{Minimize}(\textstyle\sum_{t=1}^{r} \psi(w_t)) \tag{11}$$

$$\text{Maximize}\left(f_3(y)\right) = \text{Maximize}\left(\sum_{t=1}^{r} \gamma(y_t)\right) \tag{12}$$

$$\sum_{t=1}^{r} a_{ht} w_t \geq o_h \; h = 1, \dots, g \tag{13}$$

$$w_t \in \mathbb{N} \; t = 1, \dots, r \tag{14}$$

$$y_t \in \mathbb{N} \; t = 1, \dots, r \tag{15}$$

$$\sum_{t=1}^{r} (R_t + s'_t) = S \tag{16}$$

$$\psi(w_t) = \begin{cases} 1 \text{ if } w_t > 0 \\ 0 \text{ if } w_t = 0 \end{cases} \tag{17}$$

$$\gamma(y_t) = \begin{cases} 1 \text{ if } y_t > 0 \\ 0 \text{ if } y_t = 0 \end{cases} \tag{18}$$

The fourth constraint ensures that the number of times of the h^{th} part in the t^{th} mode, multiplied by the frequency of the t^{th} mode of cut is used, will always meet the order, and the eighth constraint ensures that the size of the standard item formed from the waste is equal in size to the original standard item.

3.2 *Solution procedures*

The solution process consists of three steps:

Step 1: Forming a cutting methodology and calculating cutting waste in cutting modes, to construct a cutting methodology, first, we write the sizes of the part types as follows: $s_1 > s_2 > \ldots > s_g$. Then we determine the frequency of the size of the first part type is cut from the size of body, denoted by a_{11}, and determine the frequency of the size of the types of residual parts that can be cut from the residual size of the body. Then we lower a_{11} by 1, (i.e. we put $a_{21} = a_{11} - 1$). As in the first case, we cut the sizes of the residual part types from the remainder of the body size. We continue to reduce $a_{11}$by 1 each time, until no more a_{t1} remains where $t \geq 1$. Then, we set a_{t1} at each of the resulting values, and for each value of a_{t1}, we lower the frequency of the size of the second type of part is cut from the residual size of the body (i.e. we set a_{t1} where $t \geq 1$ to its initial values, and start to lower the value of a_{12} every time by 1), and we determine the frequency of the residual part types are cut from the residual body size as we did in the previous case. These operations are repeated for every size of the part type until $h = g - 1$. Lastly, thanks to this cutting methodology, a cutting matrix is formed, each line of which represents a mode of cut, which is referred to as a_h. Then, we calculate the waste in each $a_h$by $R_h = S - \sum_{t=1}^{g} a_{ht}$, h = 1, 2,..

In practice, the first element of the first row of the cutting matrix is determined by the following formula: $a_{11} = \left\lfloor \frac{S}{s_1} \right\rfloor$, where $\lfloor\rfloor$ the lower integer part. Then the rest of the line elements are completed as follows: for the sake of $t = 1$ and for h varies from t to g then $a_{1h} = \left\lfloor \frac{S - \sum_{z=1}^{h-1} a_{1z} \times s_z}{s_h} \right\rfloor$, [17], Then we lower a_{11} by the formula $a_{21} = a_{11} - 1$, then we set the a_{th} elements where $t \geq 1$ in the values found previously and the same operations are repeated for the size of the second type of parts, and in this way the algorithm is updated. The rest of the sizes of the part types are treated in the same way as mentioned above until $h = g - 1$, to form a matrix with f rows and g columns.

Algorithm 1. Cutting methodology algorithm and calculation of cutting waste in cutting modes

Input: The size of the main object and the sizes of the part types,
Output: possible cutting modes and the waste of cutting mode,
Let $s_1 > s_2 > \ldots > s_g$, g is the number of parts ordered

1. Determine $a_{11} = \left\lfloor \frac{S}{s_1} \right\rfloor$, put t = 1,
   for h = 2 to g do $a_{1h} = \left\lfloor \frac{S - \sum_{z=1}^{h-1} a_{1z} \times s_z}{s_h} \right\rfloor$, put h = 1, v = 1, e = 1,
   a) put t = 1, e = 1,
   b) if $a_{th} > 0$ then t = t + 1, l = 1,
   c) v = v + 1,
   d) if h = 1 then $a_{vh} = a_{t-1,h} - 1$
   For h := e + 1 to g do $a_{vh} = \left\lfloor \frac{S - \sum_{z=1}^{h-1} a_{vz} \times s_z}{s_h} \right\rfloor$,
   e) else for z := 1 to h-1 do $a_{vz} = a_{t-1,z}$,
   For h := e do $a_{vh} = a_{t-1,h} - 1$,
   For h: = e + 1 to g do $a_{vh} = \left\lfloor \frac{S - \sum_{z=1}^{h-1} a_{vz} \times s_z}{s_h} \right\rfloor$,
   f) if $a_{vh} > 0$ then put l = l + 1, b = b + 1 and go to (c) else go to (g),
   g) b = b + 1,
   h) if b < f, then go to (b),
   i) else if h < g -1 then set h = h + 1, e = e + 1 and go to (a), else stop,
2. Else set t = t + 1, b = b + 1 and go to (b).
3. For h = 1 to f do for t = 1 to g do $R_h = S - \sum_{t=1}^{g} a_{ht}$.

Step 2: Cutting strategy construction and cutting waste calculation in cutting strategies. We construct the cutting strategies by modeling the cutting modes using a graph $G = (X, U)$, where X: the set of summits, U: the set of edges. In which, each cutting mode generated in the step of construction of the cutting matrix represents a summit x of G and every pair of cutting modes, which do not cut the same type of parts, are connected by an edge u of G, that is, each pair of these modes forms cutting strategy P_t. Such that in the first phase, we merge each pair of summits u associated with an edge into a single summit E_t. We then connect these summits to the summits that are not edge-embedded u, which means that we add a new cutting mode to each previous cutting strategy to form a new cutting strategy with a larger number of cutting mode. Then we connect the summits resulting from linking summits E to summits x, to summits x that are not included in E. The number of iterations continues until the solution is unable to evolve and the test is performed by the formula: $\sum_{t=1}^{r} w_t = \left\lceil \frac{\sum_{h=1}^{g} s_h o_h}{S} \right\rceil$, where $\lceil\ \rceil$ the upper integer part. At the same time, we calculate the cutting waste in each cutting strategy constructed. We first calculate the value w_t, which represents the frequency of the j^{th} cutting mode is used, which is equal to the maximum value between ordered quantities divided by the number of times of part h in the cutting mode t use, for each type of part $\left(w_h = \mathrm{Max}\left(\left\lceil\frac{o_h}{a_{ht}}\right\rceil\right)\right)$, $t = 1, 2\ldots$ then the quantity of each type of part remaining in waste of trim in each cutting strategy noted q_{kt} where $q_{kt} = w_t - \left\lceil\frac{o_h}{a_{ht}}\right\rceil$.

Algorithm 2: Algorithm for generating cutting strategies and calculating wastes resulting from creating cutting strategies

Input: summits and edges,
Output: cutting strategies and waste resulting from creating cutting strategies
1. Let the following sets: $a = \{a_1, a_2, \ldots a_f\}$ set of cutting models, set of summits of type $X = \{x_1, x_2, \ldots, x_f\}$, set of summits of type $E = \{e_1, e_2 \ldots\}$, $U = \{u_1, u_2 \ldots\}$ set of the edges and n the size of the cutting model.
2. for h := 1 to g do
for t := t to g do
begin
Ex := false (boolean type);
for k:=1 to g do
if (a_{hk}= 0) and (a_{tk}= 0) then Ex := true;
if not Ex then relate the summit x_h by the summit x_t by an edge u.
and put $e_t = \{x_h, x_t\}$,
else there is no edge between x_h and x_t,
3. For h:= 1 to cardinal (E) do
For t := 1 to f do
begin
If $x_h \in e_t$ then there is no edge between x_h and e_t,
Else, relate the summit x_h by the summit e_t by an edge u.
relate the summit e_h by the summit e_t by an edge u, end,
4. Calculate $w_t = \mathrm{Max}\left(\left\lceil\frac{o_h}{a_{ht}}\right\rceil\right)$, $t = 1, 2\ldots$the frequency of the t^{th} cutting mode is used,
5. Calculate the quantity of each type of part remaining in waste of trim in each cutting strategy $q_{kt} = w_t - \left\lceil\frac{o_h}{a_{ht}}\right\rceil$,
6. Repeat steps 2 to 5 until the solution is unable to evolve $\sum_{t=1}^{r} w_t = \left\lceil\frac{\sum_{h=1}^{g} s_h o_h}{S}\right\rceil$.

Step 3: The waste recovery and creation of standard items, to recover the waste resulting from the cutting process, whether when generating cutting modes or

building cutting strategies, we weld the cutting wastes resulting from the cutting process to form new standard items that are used again to meet order. Firstly, we arrange the lengths of the wastes L_t, in descending order from longest size to smallest size. As long as the integer t is less than or equal to the quantity of waste Q_j and $(t \times L_h \leq (S - \sum_{z=1}^{h-1} l_z \times L_z, t = 1, 2 \ldots)$. Calculate the number of times t where waste of type h can be welded, (i.e. $l_h = t$) and from this calculate whether the waste has been finite or not (i.e. $Q_h' = Q_h - l_h$), in the case where it is finite (i.e. $Q_h' = 0$), this type of waste will be removed from the waste list, and the waste quantity set L will be organized according to the new waste quantity list. In the other case, where Q_h remains greater than zero (i.e. $Q_h' > 0$), the set of quantities of waste Q becomes of the form $Q = \{Q_1, Q_2, \ldots Q_h' \ldots, \}$. Each time we keep adding a new waste that realizes the inequality $S - \sum_{z=1}^{h} l_z \times L_z) \geq L_{h+1}$ until we reach realization $S - \sum_{z=1}^{h} l_z \times L_z \leq 0$. If this inequality is equal to zero, this means that the new standard object has been formed. However, if the inequality is completely less than zero, this means that the sizes of the wastes that were welded are larger than the size of the standard item, so we cut off that excess size and what's left is the new standard object. This recent surplus remains an irrecoverable waste.

Algorithm 3: The waste recovery algorithm

Input: waste,
Output: standard items,

1. Arrange the sizes of waste, L_h, h = 1, 2…, in descending order, i.e. $(L_1 > L_2 > \ldots)$,
2. let be h =1, t = 1, B = Ø, $Q = \{Q_1, Q_2, \ldots\}$,
3. While Q <> Ø do
   a) While $(t \leq Q_h)$ and $(t \times L_h + \sum_{z=1}^{h-1} l_z \times L_z \leq B)$ do $l_h = t$, $Q_h' = Q_h - l_h$, $B = B \cup \{Q_h'\}$,
   b) if $h \leq$ card (Q) then go to (4), (card (Q) is the number of elements of Q),
   c) else if $S - \sum_{z=1}^{h} l_z \times L_z = 0$ then $S = \sum_{z=1}^{h} l_z \times L_z$, else $S = \sum_{z=1}^{h} l_z \times L_z - C$, (where $C = \sum_{z=1}^{h} l_z \times L_z - S$),
   d) for z: = 1 to card (B) do if $Q_h' = 0$ then $Q_h = Q_h'$ and $Q = \overline{Q \cap \{Q_h\}}$, and go to (2) (where $\overline{Q \cap \{Q_h\}}$ complement of $Q \cap \{Q_h\}$), else for z: = 1 to card (B) do $l = \mathrm{Min}\left(\left\lfloor \frac{Q_h}{l_h} \right\rfloor\right)$, $Q_h' = Q_h - l \times l_h$, if $Q_h' = 0$ then $Q_h = Q_h'$,
   $Q = \overline{Q \cap \{Q_h\}}$ and go to (2) else let be $Q_h = Q_h'$ and go to (3),
   e) else for z: = 1 to card (B) do if $Q_h' = 0$ then let be $Q_h = Q_h'$, $Q = \overline{Q \cap \{Q_h\}}$, and go to (3a) else let be $Q_h = Q_h'$ and go to (2),
4. If $\sum_{z=1}^{h} l_z \times L_z) + L_{h+1} \geq S$ then
   a) let be h: = h +1, and go to (3a),
   b) else if $h \leq$ (card (Q) -1) then h: = h +1, $l_h = 0$, $Q_h' = Q_h$ and go to (4) else go to (3c),

3.3 *Limitations of the Algorithm*

In this paragraph, we verify the limitations of the number of stages and the non-repeatability of the algorithm:

Algorithm 1: In this methodology, the first element of the first row of the cutting matrix is determined by the following formula: $a_{11} = \left\lfloor \frac{S}{s_1} \right\rfloor$. Then the rest of the line elements are completed as follows: for the sake of $t = 1$ and for h varies from t to g then $a_{1h} = \left\lfloor \frac{S - \sum_{z=1}^{h-1} a_{1z} \times s_z}{s_h} \right\rfloor$, the rest of the sizes of the part types are treated in the same way as mentioned above until $h = g - 1$, Therefore, each part is used once and does not return to the processing process again. Therefore, the algorithm is not repeated. Since the number of parts is limited, the algorithm is limited.

Algorithm 2: The idea of algorithm No. 2 in providing possible solutions is that every pair of cutting modes, which do not cut the same type of cutting, is connected by an edge u of G, that is, every pair of these modes constitutes a possible solution (cutting strategy). Then we add a new cutting mode to each previous cutting strategy to form a new cutting strategy with a larger number of cutting mode. Then we connect the vertices resulting from connecting vertices E to vertices x, with vertices x that do not enter E. The calculation continues in the same way until we reach the theoretical value, reaching the theoretical value means that the number of steps is limited, and at each step we add a new cutting mode to form a new possible solution with a larger number of numbers of cutting modes, which means that the algorithm does not loop.

Algorithm 3: Algorithm 3 trains the standard items based on the waste vector. Each time a part of the waste vector is used, it tests whether the type of part of waste is zero or not. If it is equal to zero, this type of part of waste will be deleted from the list of vector of wastes, and all the quantities of wastes L will be organized according to the new list of quantities of wastes without the quantity is equal to zero, the algorithm. Continues to take waste until we cannot form a new standard item or finish the waste vector, for this, the algorithm does not loop and as the waste vector is finite dimension then the algorithm is finished.

3.4 *Illustrative example*

In order to clarify the operation of the proposed algorithm for solving the three-objective one-dimensional cutting problem with waste recovery, we implement it on the following example, where we initially have the following data: Standard item size, required piece sizes, and orders as shown in table 1.

Table 1
Standard item size, required cutting sizes and orders

$g = 3$ S = 90		
h	s_h	o_h
1	60	5
2	40	6
3	30	12

1. By applying algorithm number one, we obtain the following cutting matrix and the waste resulting from applying each cutting mode:

$$A = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 2 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 3 \end{pmatrix}, R = \begin{pmatrix} 0 \\ 10 \\ 20 \\ 0 \end{pmatrix}$$

1st iteration

1. The theoretical value $\sum_{t=1}^{r} w_t = 10$,
   a) Let set of cutting modes $a = \{a_1, a_2, a_3, a_4\}$, each cutting mode represents a vertex x, then the vertices are $X = \{x_1, x_2, x_3, x_4\}$, set of vertices of type $E = \emptyset$, and the set of edge $U = \emptyset$,
   b) According to step 2 of algorithm 2, the pairs of cutting modes, which do not cut the same type of parts are the pairs a_1, a_2 and a_1,a_3. Therefore, we connect the vertices x_1, x_2 by an edge u_1 and the vertices x_1, x_3 by an edge u_2, Hence the set of edges $U = \{u_1, u_2\}$. As shown in figure 1.

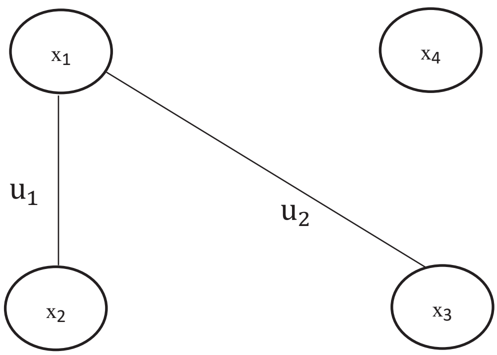

Figure 1
Construction of cutting strategies

The number of parts in excess of the order and their quantities resulting from the application of various cutting plans at this step are respectively $L_t = (60, 20, 10)$, $Q_t = (8, 1, 1)$,

c) By applying algorithm No. 2 to build the cutting planes, and algorithm No. 3 to recover the cutting waste, for the first step of the solution, the results obtained and shown in table No. 2, which are represented by the feasible solutions, which are characterized by the number of optimal standard items recovered, the number of modes and the percentage of waste.

Table 2
Results of applying the first step of the solution

No of Solutions	Cutting strategies	Percentage of waste	number of modes	Number of bodies retrieved	execution time
1	$a_1 = (1, 0, 1)$ $a_2 = (0, 2, 0)$	4.5	2	4	1.00
2	$a_1 = (1, 0, 1)$ $a_3 = (0,1, 1)$	1.8	2		

2nd iteration

a) Set of vertices of type $X = \{x_4\}$, and set of vertices of type $E = \{\{x_1, x_2\}, \{x_1, x_3\}\} = \{e_1, e_2\}$,

b) To build the cutting strategies of three modes, we connect the vertices e_1, e_2 by an edge u_3, the vertices e_1, x_4 by an edge u_4, and the vertices e_2, x_4 by an edge u_5, and hence the set of edges $U = \{u_3, u_4, u_5\}$. As shown in figure 2.

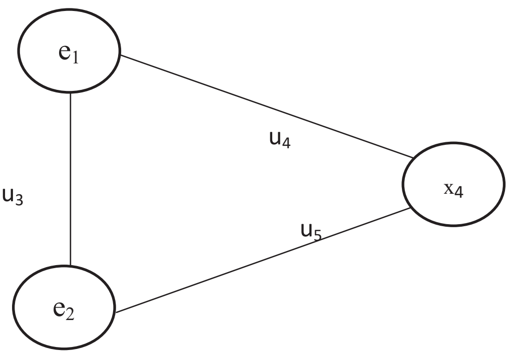

Figure 2
Construction of cutting strategies

c) By applying algorithm No. 2 to build the cutting strategies, the number of parts in excess of the order and their quantities resulting from the application of various cutting strategies at this step are respectively $L_t = (60, 0, 0)$, $Q_t = (1, 0, 0)$,

d) By applying algorithm No. 2 to build the cutting strategies, and algorithm No. 3 to recover the cutting mode, for the second step of the solution, the results obtained and shown in table No. 3, which are represented by the feasible solutions, which are characterized by the number of optimal standard items recovered, the number of modes and the percentage of waste.

Table 3
Results of applying the second step of the solution

No of Solutions	Cutting strategies	Percentage of waste	number of modes	Number of items retrieved	execution time
1	$a_1 = (1,0,1)$ $a_2 = (0,2,0)$ $a_3 = (0,1,1)$	3.6	3	0	1.20
2	$a_1 = (1,0,1)$ $a_2 = (0,2,0)$ $a_4 = (0,0,3)$	0.9	3		
3	$a_1 = (1,0,1)$ $a_3 = (0,1,1)$ $a_4 = (0,0,3)$	3.6	3		

e) Stop because there is no progress in the solution. The set of effective solutions is shown in the table below as follows:

Table 4
Effective solutions set

No of solutions	Effective solutions	Percentage of waste	Number of modes	Number of items recovered	execution time (S)
1	$a_1 = (1,0,1)$ $a_3 = (0,1,1)$	1.8 %	2	4	1.00
2	$a_1 = (1,0,1)$ $a_2 = (0,2,0)$ $a_4 = (0,0,3)$	0.9 %	3	0	1.20

The results obtained through the illustrative example and presented in Table No. 4 show that two cutting strategies, the first with two cutting modes and the second with three cutting modes represented the efficient solutions. The solution was distinguished by two cutting modes with an estimated the waste by 1.8 % and two numbers of modes, and four standard items were recovered and with an execution time of one second. As for the solution with three modes, the waste estimated at 0.9 % and for three numbers of modes, no standard item was recovered with an execution time of one second and twenty-hundredths.

4. Results and discussion

In order to test the effectiveness of the proposed method to solve the model studied in this work, which is called the method of constructing cutting strategies with waste recovery (CPTM). We first generate 12 random instances to carry out a study empiric on these. Then we used real data taken from the Japanese Chemical Fiber Manufacturing Company to compare our proposed method with both the symbiotic algorithm proposed by Golfeto *et al*. called GSAM [18] and the genetic algorithm proposed by Araujo *et al*. called GOAM [12]. They are two techniques that study the minimization of both waste and the number of modes. Our proposed algorithm was coded in Delphi language and tested on a HP Z8 2x Xeon® Silver 4110 machine (2.1 GHz/8 cores/11.00 MB/85 W) Ram 16 GB (2 x 8 GB) DDR4 2400T HDD 1 TB Sata DVD-RW Windows 10 Pro 64 Bit.

4.1 *Randomly Generated Data Set*

We initially generated 12 random cases with the following specifications.

1. Number of types of ordered parts: g varies from 3 to 50,
2. Item size: S = 4060 for the first 3 cases, S = 7050 for the remaining 3 cases, S = 22780 for the remaining 3 cases, S = 25621 for the remaining 2 cases.
3. Item size: s_h were randomly generated in the interval [40, 3500],
4. Order: o_h were randomly generated in the interval [3, 658].

The results obtained by our proposed method CPTM as appear below in Table 5.

Table 5
The set of solutions provided by CPTM

Mode	Waste (%)	Recovered items	Execution time (S)	Mode	Waste (%)	Recovered bodies	Execution time (S)	Mode	Waste (%)	Recovered bodies	Execution time (S)
Instance 1 S = 4060 g = 3				Instance 2 S = 4060 g = 20				Instance 3, S = 4060 g = 50			
2	0.07	3	1.56	7	0.49	3	6.00	11	0.34	6	10.0
1	1.04	2	1.00	6	0.89	2	5.63	10	0.55	3	9.07
Instance 4, S = 7050 g = 5				5	1.76	4	4.54	9	0.93	0	8.00
4	1.06	2	2.74	4	2.65	5	4.13	8	1.07	5	7.60
3	2.65	4	2.00	3	4.87	3	2.04	7	1.76	4	6.00
2	5.45	1	1.54	2	6.45	1	1.50	6	2.53	3	5.66
Instance 5 S = 7050 g = 25				Instance 6 S = 7050 g = 45				5	3.64	2	4.76
8	0.04	3	6.00	11	0.05	5	9.00	4	5.32	1	4.10
7	0.76	2	5.05	10	0.12	7	8.08	3	7.45	6	2.04
6	1.25	5	4.00	9	0.76	2	7.00	Instance 7 S = 7050 g = 50			
5	2.66	1	3.65	8	1.53	0	6.46	13	0.08	6	11.0
4	4.63	7	2.00	7	2.65	3	6.00	12	0.76	4	10.3
3	6.59	6	1.45	6	3.08	4	5.58	11	0.92	1	10.00
Instance 8 S = 22780 g = 35				5	3.84	5	5.02	10	1.00	5	9.07
10	0.06	7	12.00	4	4.66	4	4.32	9	1.43	3	8.21

9	0.21	4	11.22	3	6.34	6	2.00	8	2.02	6	7.55
8	0.47	2	10.34	Instance 9 S = 22780 g = 45				7	3.47	2	7.01
7	1.01	3	7.21	13	0.02	4	15.0	6	4.05	0	5.44
6	1.56	0	5.00	12	0.09	2	14.0	5	5.24	3	5.00
5	2.74	1	4.06	11	0.54	6	13.5	4	7.72	2	4.36
4	4.08	2	3.05	10	1.04	1	12.0	Instance 11 S = 22780 g = 50			
Instance 10 S = 25621 g = 35				9	2.12	0	11.6	14	0.02	5	16
13	0.06	6	15.00	8	3.45	5	11.2	13	0.08	3	15.0
12	0.71	7	14.07	7	4.09	3	10.08	12	0.65	1	14.0
11	1.08	3	13.00	6	4.76	1	10.0	11	1.04	0	13.5
10	2.23	5	12.22	5	6.02	2	7.43	10	1.9	2	12.0
9	3.44	6	11.04	4	6.75	4	4.00	9	2.84	4	11.0
8	3.76	4	10.21	3	7.77	5	3.44	8	3.52	5	10.3
7	5.06	1	9.45	Instance 12 S = 25621 g = 5				7	5.13	3	9.08
6	5.34	1	8.34	4	0.21	5	5.07	6	6.34	2	8.80
5	7.25	2	7.00	3	0.76	1	4.00	5	6.98	1	5.00
4	8.90	2	5.06	2	2.00	2	3.05	4	8.07	7	4.13

Table 5 represents vectors of efficient solutions for 12 instances of different types of parts, as well as different request sizes and different standard item sizes, which were randomly generated data. The accuracy of the results obtained varies from one instance to another. For example, instance N^{o}. 1, where the standard item size was 4060, where the efficient solutions were a solution with a two numbers of modes and a percentage of waste estimated at 0.07%, and 3 standard items were retrieved with an execution time of one second and fifty-six hundredths and a second solution with one number of mode and a percentage of waste estimated at 1.04%, and 2 standard items were retrieved with an execution time of one second. For example, instance 11, where the size of the standard item was S = 22780 and the number of parts g = 50. The precision of the solutions provided varied 8.07% of the total missing size and 4 numbers of modes to 0.028% of the total missing mode and 16 numbers of modes, with the execution time ranged from 16 seconds to 4 seconds and thirteen hundredths, while the number of standard items retrieved ranged from 7 to 0 standard items. On a careful reading of the results obtained in this experimental study, it can be seen that our proposed method provides a set of efficient solutions that form the Pareto front in a reasonable execution time.

4.2 *Practical Data Set*

In this paragraph, we conduct a comparative study between the CPTM algorithm and the algorithm of Golfito et al. [18] and the algorithm of Araujo et al. [12]. We take into account the waste and the number of modes for a sample of 20 cases taken from a Japanese company specialized in chemical fibers as stated by to the data below:

1. Number of types of ordered parts: g varies from 4 to 20,

2. item size: S = 5180 for the first 5 cases and S = 9080 for the remaining 6 cases,
3. part size: s_h were randomly generated in the interval [500, 2000];
4. Order: o_h were randomly generated in the interval [2, 264].

The solutions provided by the symbiotic algorithm, the genetic algorithm and our method CPTM as visible in Table 6.

Table 6

The solutions provided by the algorithms GOAM, GSAM and CPTM

mode	waste (%)			modes	waste (%)			modes	waste (%)		
	GOAM	GSAM	CPTM		GOAM	GSAM	CPTM		GOAM	GSAM	CPTM
Instance Fiber06 5180				Instance Fiber28b 5180				Instance Fiber28b 9080			
7			1.413	22		2.86	1.751	19	2.42		2.502
6	2.09		1.829	21		4.49	3.094	18	5.23		3.743
5	8.27	5.19	2.557	20		7.76	3.762	17	23.50		5.372
4			3.365	19		28.90	4.105	16	32.54		6.865
3		8.28	5.632	18			4.845	15			7.427
Instance Fiber07 5180				17	8.27		5.312	14			8.014
10			1.263	16			5.982	13			8.455
9			1.748	15			6.836	12			10.463
8			2.152	14	7.40		7.128	11			11.054
7			2.836	13	9.12		7.471	10			11.658
6			3.062	12			8.326	9			13.124
5			3.475	11	11.98		8.852	8		9.95	13.461
4	4.98		4.26	10	10.8		9.12	7			13.982
3	8.16	4.98	6.113	9	13.42		11.04	6		12.96	15.547
2		11.34	7.541	8	17.72			5		22.00	18.076
1		68.61	11.76	7	22.87			4		41.52	20.654
Instance Fiber11 5180				6	41.78			3			23.065
12			1.575	5	97.77		11.45	2			31.521
11			2.123	4			13.68	Instance Fiber26 9080			
10			3.857	3			14.54	15	1.00		1.074
9			4.379	2			17.04	14			1.364
8			4.937	Instance Fiber11 9080				13	1.94		2.875
7	2.98		5.075	6			1.054	12	3.764		3.764
6	9.13		6.654	5	2.38		2.648	11			4.547
5	12.20		7.475	4	5.07		2.975	10		5.72	5.056
4	14.67	6.65	8.536	3	11.90	5.08	4.073	9		6.66	6.423
3		9.13	10.05	2		13.16	7.751	8		7.61	7.522
2		52.16	20.23	1		228.7	22.54	7		9.49	8.043
InstanceFiber16 5180				Instance Fiber18 9080				6		10.44	14.346
16		4.01	1.052	9	2.34		1.058	5		11.38	16.871
15			1.317	8	4.20		2.673	4		23.65	18.346
14		5.23	1.836	7	6.06		4.012	3		46.30	22.065

13		6.45	3.101	6	6.86		4.831	2			34.432
12		10.13	4.532	5		2.35	5.463	Instance Fiber29 9080			
11			4.621	4		4.21	6.561	13	3.03		2.054
10			5.034	3		7.93	7.842	12			3.021
9		5.77	5.636	Instance Fiber13a 9080				11	5.89		3.213
8		8.92	7.551	6	2.25		1.843	10		1.843	2.25
7	4.01	04.01	8.642	5	44.25		3.612	9		3.612	44.2
6	5.23	5.23	8.851	4			4.221	8		4.221	
5	7.68	7.68	9.023	3		3.08	5.547	7	3.08	5.547	
4	16.25	16.25	9.702	2		9.53	7.452	6	9.53	7.452	
3	66.41	66.41	20.47	1			10.761	5		10.76	

In reading the results shown in Table No. 6, which is a sample of 11 instances, taken from a chemical company in Japan, it was solved using three methods: the genetic symbiotic algorithm, the genetic algorithm, and our proposed method. We analyze in detail three different cases, in one of which the standard object size is S = 5180 and in two of which the standard object size is S = 9080. We use the concept of dominance to do the charts and provide only the best solutions to the method.

Instance Fiber16, 5180. In this instance, CPTM provided better results than the other two methods, as it provided solutions that dominated all other methods in most cases. CPTM dominated the other two methods from the number of modes 16 to the number of modes 8. In all of these solutions, the waste percentage was lower, while GOAM and GSAM dominates in solutions with numbers of 7, 6 and 5. As shown in figure 3 below.

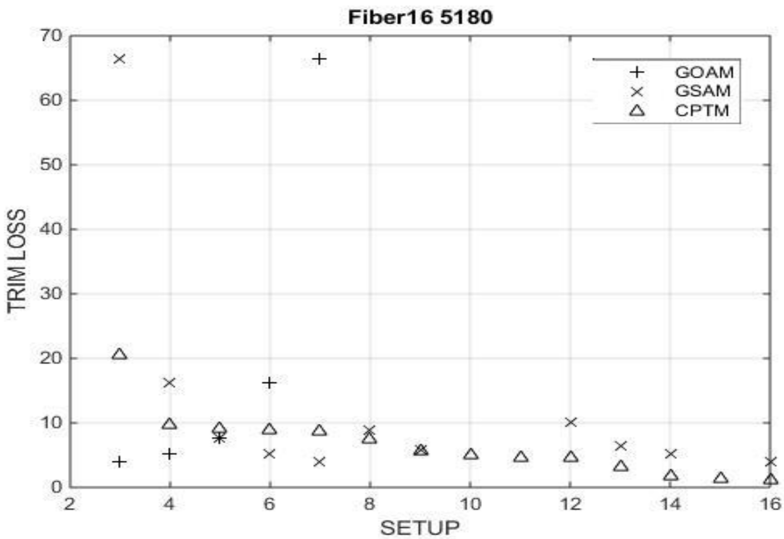

Figure 3
Fiber 16

Instance Fiber13a, 9080. From this instance, we see that GSAM solutions are dominant over GOAM solutions, while CPTM solutions are completely dominant. In this instance shown in figure 4.

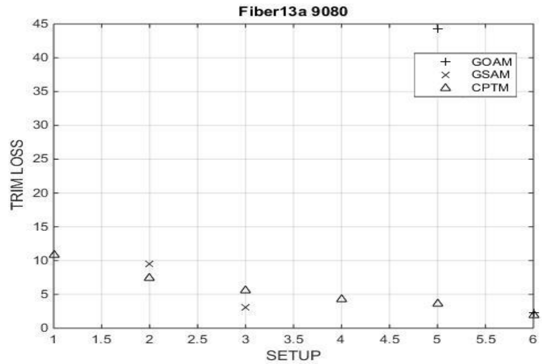

Figure 4
Fiber13a

.

Instance Fiber26, 9080. In this instance, the results of the three algorithms were close, as none of them dominated the others. For example, the solutions with settings 6, 5, and 4 were CPTM dominant, but at number 3, the preference in the result went to GOAM. As shown in figure 5.

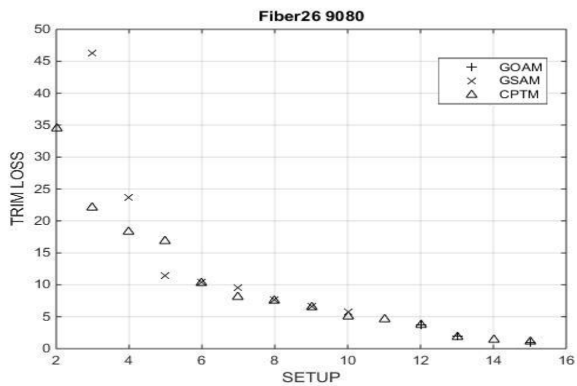

Figure 5
Fiber26

5. Conclusions

In this study, we addressed the one dimensional cutting stock problem, and the overall objective was to minimize the cost of production by controlling the waste, whether by maximizing the recovered standard items by recovering waste or retrieving parts in excess of satisfying the order, because the buyer cannot be forced to buy the excess parts that were cut during building cutting plans, these excess parts are considered a waste. Minimizing the number of modes, in addition to saving waste, also preserves the cutting machine. To achieve this, we built a cutting model with three objectives: to maximize the standard objects recovered by welding and to minimize the waste and the number of modes using a multi-objective optimization method, As evaluating each objective separately and at the same time allows providing a set of efficient solutions, it also allows clarifying the relationship between these objectives. The empirical study of the proposed solution method has proven that it gives a set of efficient solutions in a reasonable time with different sizes of order quantity or different sizes of standard objects. Comparing it to other methods from the literature using real data has proven to be a real competitor to these techniques, such as symbiotic genetic algorithm or genetic algorithm.

References

[1] Y. Cui and Y. Yang, "A heuristic for the one-dimensional cutting stock problem with usable leftover," *European Journal of Operational Research,* vol. 204, no. 2, pp. 245–250 (2010).

[2] S. C. Campello, C. T. Ghidini, A. O. C. Ayres, and W. A. Oliveira, "A multi-objective integrated model for lot sizing and cutting stock problems," *Journal of the Operational Research Society*, vol. 71, no. 9, pp. 1466–1478 (2020).

[3] P. C. Gilmore and R. E. Gomory, "A linear programming approach to the cutting stock problem," *Operations Research*, vol. 9, pp. 849–859 (1961).

[4] P. C. Gilmore and R. E. Gomory, "A linear programming approach to the cutting stock problem—Part II," Operations Research, vol. 11, no. 6, pp. 863–888, 1963.

[5] P. C. Gilmore and R. E. Gomory, "Multistage cutting stock problems of two and more dimensions," *Operations Research*, vol. 13, no. 1, pp. 94–120 (1965).

[6] R. W. Haessler, "A note on computational modifications to the Gilmore–Gomory cutting stock problem," *Operations Research*, vol. 28, pp. 1001–1005 (1980).

[7] A. A. Farley, "Mathematical programming models for cutting-stock problems in the clothing industry," *Journal of the Operational Research Society*, vol. 39, no. 1, pp. 41–53 (1988).

[8] S. Umetani, M. Yagiura, and T. Ibaraki, "One-dimensional cutting stock problem with a given number of setups: A hybrid approach of meta-heuristics and linear programming," *Journal of Mathematical Modelling and Algorithms*, vol. 5, pp. 43–64 (2006).

[9] P. S. C. Oltroniere, K. C. Poldi, F. M. B. Toledo, and M. N. A. Arenales, "Coupling cutting stock lot sizing problem in the paper industry," *Annals of Operations Research*, vol. 157, no. 1, pp. 91–104 (2008).

[10] J. De Armas, G. C. Miranda, and C. León, "A multi-objective approach for the 2D guillotine cutting stock problem," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 6692, pt. 2, pp. 292–299 (2011).

[11] G. Wäscher and S. Henn, "Extensions of cutting problems, setups," *Pesquisa Operacional*, vol. 33, no. 2, pp. 133–162 (2013).

[12] S. Araujo, K. C. Poldi, and J. Smith, "A genetic algorithm for the one-dimensional cutting stock problem with setups," *Pesquisa Operacional*, vol. 34, no. 2, pp. 165–187 (2014), doi: 10.1590/0101-7438.2014034.020165.

[13] Y. Cui, C. Zhong, and Y. Yao, "Pattern-set generation algorithm for the one-dimensional cutting stock problem with setup cost," *European Journal of Operational Research*, vol. 243, no. 2 (2015), doi: 10.1016/j.ejor.2014.12.01.

[14] A. A. S. Leão, M. M. Furlan, and F. M. B. Toledo, "Decomposition methods for the lot sizing and cutting stock problems in paper industries," *Applied Mathematical Modelling*, vol. 48, pp. 250–268 (2017).

[15] A. A. Filho, A. C. Moretti, and M. V. Pato, "A comparative study of exact methods for the bi-objective integer one-dimensional cutting stock problem," *Journal of the Operational Research Society*, vol. 69, no. 1, pp. 91–107 (2018).

[16] A. Mellouli, R. Mellouli, and F. Masmoudi, "An innovative genetic algorithm for a multi-objective optimization of two-dimensional cutting-stock problem," *Applied Artificial Intelligence*, vol. 33, no. 6, pp. 531–547 (2019).

[17] S. M. Suliman, "Pattern generating procedure for the cutting stock problem," *International Journal of Production Economics*, vol. 74, no. 1–3, pp. 293–301 (2001), doi: 10.1016/S0925-5273(01)00134-7.

[18] R. Golfeto, A. C. Moretti, and L. L. Salles-Neto, "A genetic symbiotic algorithm applied to the cutting-stock problem with multiple objectives," *Advanced Modeling and Optimization*, vol. 11, pp. 473–501 (2009).

Received December, 2024