Mathematical Modelling and Numerical Simulation with Applications, 2024, 4(3), 335–350

https://dergipark.org.tr/en/pub/mmnsa
ISSN Online: 2791-8564 / Open Access
https://doi.org/10.53391/mmnsa.1492749

RESEARCH PAPER

Column generation approach for 1.5-dimensional cutting stock problem with technical constraints

Müjgan Sağır [1,*,‡] and **Tuğba Saraç** [1,‡]

[1]Eskişehir Osmangazi University, Faculty of Engineering and Architecture, Department of Industrial Engineering, 26040 Odunpazarı, Eskişehir, Türkiye

[*] Corresponding Author
[‡] mujgan.sagir@gmail.com (Müjgan Sağır); tsarac@ogu.edu.tr (Tuğba Saraç)

Abstract

In this study, the 1.5-dimensional cutting stock problem with technical constraints is considered. In the literature, this problem is also defined as a strip packing or open dimension problem. When given a strip of infinite length and bounded width, the problem is to define a packing of rectangular objects into a strip that minimizes its final length. Technical constraints, such as the order type and the number of strips, are indispensable in real life; however, they are often neglected in the literature because they make the problem difficult to solve. Only one study was reached in the literature that took into account technical constraints, but in that mentioned study, only a mathematical model was proposed for the problem. In this context, our aim is to solve the problem with a more effective approach. The research question in this study is the usability of the column generation technique to solve the 1.5-dimensional cutting stock problem. In this study, the column generation approach was proposed for the first time for the considered problem. To demonstrate the performance of the proposed solution method, randomly generated test problems were solved with GAMS/Cplex. As we report the results, proposed column generation approach (CG) reaches very close (such as 1% and 2% error) solutions to integrated mathematical model (IM) for small sized problems in a second. On the other hand, while CG solved all the problems in a reasonable time, IM could not produce a feasible solution to some problems. Numerical experiments showed that the column generation algorithm outperforms the integrated mathematical model for the problem.

Keywords: 1.5-dimensional cutting problem; column generation; mixed-integer linear programming

AMS 2020 Classification: 90C06; 90C10; 90C11; 90C90

1 Introduction and literature survey

The essential characteristics to be considered to generate the cutting plans are the number of dimensions of the stock material and the order of pieces. Cutting items such as paper rolls and metal

➤ **Received:** 30.05.2024 ➤ **Revised:** 27.08.2024 ➤ **Accepted:** 16.09.2024 ➤ **Published:** 30.09.2024

rods are one-dimensional. The problems, such as pallet placement and cutting small rectangular order pieces from large rectangular stock materials, are two-dimensional, and problems such as container insertion and packing into packing boxes, are three-dimensional. Four-dimensional problems can arise when the time dimension is added to a three-dimensional problem.

One-and-a-half-dimensional problems are a particular case of two-dimensional problems. Such problems arise when rectangular order pieces are to be placed on very long rolls. Although rectangular order pieces are being cut, the problem is not two-dimensional because the side waste along the stock material can be defined by one dimension.

Dyckhoff et al. (1985) made significant contributions to the literature on cutting problems and named the one-dimensional cutting stock problem in continuous form as one plus half-dimensional, 1.5-dimensional [1]. This definition in Dyckhoff's classification has not been included much in the literature. The related problems are defined as 'one-dimensional and various-sized stock materials cut-off' instead of the 1.5-dimensional problem [2]. Moreover, in the following years, Dyckhoff grouped the problems according to their dimensions as one, two, three, and $N-$dimensional ($N > 3$) problems. This classification does not include the 1.5- dimensional problems [3].

Song et al. (2006) defined the 1.5-dimensional problem differently. Accordingly, the 1.5- dimensional cutting stock problem is cutting smaller rectangular order pieces from large rectangular stock materials. In this problem, the order piece requested by the customers can sometimes be longer than the primary material. In this case, several stock materials are brought together so that their total length matches the size of the order pieces. This problem is also defined as a 1.5-dimensional problem because of the assumption that the stock materials can be combined [4].

In the definitions above, the 1.5-dimensional problem is the placement of rectangular order pieces to the stock material, which can be accepted as a fixed width and continuous form. Although this problem is not mentioned as much as other cutting problems in the literature, it has a significant area, especially in production environments with inputs such as paper, metal, sheet metal.

Saraç and Özdemir (2003) discussed the 1.5-dimensional cutting stock problem with the limited number of strips, piece types, and stock material selection. They proposed a two-step approach to solve the problem. The cutting plans are derived for the first stage, and complete enumeration is considered under the constraints of the number of strips and the order of piece type. In the second stage, a bi-objective, nonlinear, mixed-integer mathematical model is proposed to determine which cutting plans and stock materials will be selected. A genetic algorithm has been developed because a mathematical model cannot solve real-life problems [5]. Gasimov et al. (2007) developed a new multi-objective mixed-integer linear mathematical model for the 1.5-dimensional cutting stock and stock material selection problem. This model requires the cutting plans to be derived in advance [6]. Kokten and Sel (2022) developed a nonlinear mixed-integer mathematical model for the 1.5-dimensional cutting stock and stock material selection problem. They used a decomposition method in which the sub-models were solved sequentially to solve the problem [7]. Saraç and Sağır (2021) developed a mixed-integer linear mathematical model for the 1.5-dimensional cutting stock problem with limited part type and strip number that does not need to be derived in advance of cutting plans [8]. Duysak et al. (2022) proposed a metaheuristic algorithm for the 1.5-dimensional cutting and assortment problem. They considered the due dates of the cutting parts [9]. Vasilyev et al. (2023) proposed a few mathematical models and two solution algorithms for the problem [10]. Liu et al. (2023) dealt with two aspects of the problem: the uncertain demand for items and the need for diverse types of strips to cater to varying customer needs. They proposed a robust optimization model to cope with these difficulties [11].

The variations of the considered problem have also been referred to in the literature as strip packing or open dimensional problems. The strip packing problem is defined as follows. Consider a set of n rectangular items with dimensions (wj, h_j) where wj and h_j represent the width and the height of item j, respectively, and wj is an integer value. Let R be a rectangular object (strip) with fixed width W and height H large enough (infinite height) to pack all items. The objective of the strip packing problem is to pack all items without overlapping while minimizing the height of the strip [12]. On the other hand, according to Wäscher et al. (2007), 2D strip packing corresponds to the Two-Dimensional Rectangular Open Dimension Problem [13].

A few studies solve the strip packing problem with column generation. In their study, Sugi et al. (2020) considered the rectangular strip packing problem with a three-stage guillotine cutting constraint and the limitations of slitter blades. They propose a new algorithm based on the column-generation technique for this problem [14]. While Cintra et al. (2008) and Bettinelli et al. (2008) developed solution approaches based on the column generation technique for the classical two-dimensional level strip packing problem [15, 16], respectively, Cui et al. (2017) suggested it for the rectangular level strip packing problem [17]. When it comes to level, the structure of the problem differs significantly from that of the classical problem since the cutting process is done on a level basis. The problems addressed in these studies are different from our problems. Order type and strip number constraints were not considered together in any of these studies. In summary, we have reached only one study [8] in the literature that considers technical constraints. Furthermore, when considering the 1.5-dimensional cutting problem literature, it becomes evident that the column generation solution approach has not been applied to this problem previously. In other words, in this study, the column generation approach was proposed for the first time for the considered problem.

The following Section 2 presents the problem in detail and gives the mathematical model of the problem. Section 3 proposes a column generation approach to solve the problem. Section 4 provides experimental results, Section 5 gives the discussion, and Section 6 presents the conclusion.

2 Problem definition and mathematical model

n rectangular order pieces of different dimensions are cut from G-width stock material. The length (L) of the stock material is long enough to neglect the length restriction when creating cutting plans. For this reason, cutting plans are created by considering only the 'width' constraints. Then, the total lengths are calculated separately for each order piece included in a cutting plan. The largest of these determines the size of the cutting plan. The number of knives $(t-1)$ that can cut the stock material into strips by cutting parallel to the length is limited. Therefore, the stock material can be cut into a maximum of t strips. In other words, a maximum t order pieces can be placed on the stock material. While cutting the order pieces, they cannot be rotated; that is, cutting should be made so that the width of the order piece is parallel to the width and length of the stock material. Also, the maximum number of order piece types (c) that can be included in a cutting plan is limited.

An integrated mathematical model (IM) proposed by Saraç and Sağır (2021) generates and selects cutting plans for 1.5-dimensional cutting stock problems with technical constraints [8]. Table 1 gives the indices of the mathematical model, Table 2 shows the parameters, and Table 3 shows the decision variables.

Table 1. Indices

r	Quantity index with the order piece at the width of the cutting plan $r \in \{1, \dots, enb_j \{q_j\}\}$
j	Order piece index $j \in \{1, \dots, n\}$
k	Cutting plan index $k \in \{1, \dots, m\}$

Table 2. Parameters

n	number of order pieces
m	maximum number of cutting plans that can be derived
e_j	width of the order piece j (cm)
b_j	length of the order piece j (cm)
d	the demand of the order piece
q_j	the quantity that the order piece j can fit in the width of the stock material $q_j = \left\lfloor \frac{G}{e_j} \right\rfloor$
G	width of the stock material (cm)
L	length of the stock material (cm)
t	maximum number of strips that can be included in a cutting plan
c	maximum variety of order parts that can be included in a cutting plan

Table 3. Decision variables

μ_{jk}	the total amount of order piece j included in the cutting plan k
y_{jk}	the quantity that the order piece j can fit in the width of the k^{th} cutting pattern
z_k	1, if k^{th} cutting pattern is used, 0 otherwise
w_{jk}	1, if j^{th} order is included in the k^{th} cutting pattern, 0 otherwise
x_k	Net amount to be used from the k^{th} cutting pattern (cm)
s_{jkr}	1, if there is an r row of the order j at the width of the cutting pattern k, 0 otherwise
$\propto_{jk}$	the quantity of the order piece j can fit in the length of the cutting pattern k
σ_k	amount to be used from the k^{th} cutting pattern (Each cutting plan is assumed as 100 cm. In the mathematical model, and σ_k decision variable shows how many times 100 cm cutting patterns are used. Therefore, the number of uses of the cutting patterns also means how many meters are used
M'	a big positive number $M' = \left\lfloor \frac{G}{e_j} \right\rfloor \left\lfloor \frac{L}{b_j} \right\rfloor$
M''	a big positive number $M'' = \max q_j$

The IM model is given below:

$$\sum_k \mu_{jk} \geq d_j, \qquad \forall j, \tag{1}$$

$$\sum_j e_j y_{jk} \leq G z_k, \qquad \forall k, \tag{2}$$

$$\sum_k x_k \leq L, \tag{3}$$

$$\mu_{jk} \leq y_{jk} M', \qquad \forall j,k, \tag{4}$$

$$\mu_{jk} \geq y_{jk}, \qquad \forall j,k, \tag{5}$$

$$\propto_{jk} \leq \frac{100\sigma_k}{b_j}, \qquad \forall j,k, \tag{6}$$

$$\mu_{jk} \leq r\propto_{jk} + (1 - s_{jkr})M', \qquad \forall j,k,r | r \leq q_j, \tag{7}$$

$$y_{jk} = \sum_{r|r\leq q_j} r s_{jkr}, \qquad \forall j,k, \tag{8}$$

$$\sum_{r|r\leq q_j} s_{jkr} \leq 1, \qquad \forall j,k, \tag{9}$$

$$\sum_{j} \sum_{r|r \le q_j} s_{jkr} \le c, \qquad \forall k, \tag{10}$$

$$\sum_{j} y_{jk} \le t, \qquad \forall k, \tag{11}$$

$$\sum_{j} y_{jk} \le \sigma_k M'', \qquad \forall k, \tag{12}$$

$$\sigma_k \ge z_k, \qquad \forall k, \tag{13}$$

$$\sigma_k \le L\, z_k, \qquad \forall k, \tag{14}$$

$$x_k \ge b_j \propto_{jk}, \qquad \forall j,k, \tag{15}$$

$$x_k \ge 0, \qquad \forall k, \tag{16}$$

$$y_{jk} \ge 0 \text{ and integer}, \qquad \forall j,k, \tag{17}$$

$$\mu_{jk} \ge 0 \text{ and integer}, \qquad \forall j,k, \tag{18}$$

$$z_k \in \{0,1\}, \qquad \forall k, \tag{19}$$

$$w_{jk} \in \{0,1\}, \qquad \forall j,k, \tag{20}$$

$$s_{jkr} \in \{0,1\}, \qquad \forall j,k,r, \tag{21}$$

$$\sigma_k \ge 0 \text{ and integer}, \qquad \forall k, \tag{22}$$

$$\propto_{jk} \ge 0 \text{ and integer}, \qquad \forall j,k. \tag{23}$$

Objective function

$$f = enk \sum_{k} \frac{x_k}{L} + \sum_{k} \frac{Z_k}{m}. \tag{24}$$

Constraint (1) is the demand constraint. It is ensured by constraint (2) that the total width of all order pieces placed in a cutting plan does not exceed the width of the stock material. The constraint (3) is for the total amount of cutting plan used not to exceed the length of the stock material. The constraints (4) and (5) are the relationship constraints between variables μ_{jk} and y_{jk}. If y_{jk} is zero, they ensure that the variable μ_{jk} is also zero. Constraint (6) calculates how many times the order pieces j are included in the cutting plan k considering only its length. Decision variable σ_k indicates how many pieces of the cutting plan are used, and it is multiplied by 100 to convert to cm. Constraint (7) calculates exactly how many pieces of order j are included in the total amount of cutting plan k. Constraint (8) calculates how many pieces of order j are included in a cutting plan considering only its width. Constraint (9) indicates that if an order piece is used in a cutting plan, the amount that can fit in the end can be a single value. Constraint (10) indicates that maximum c different order pieces can be included in the cutting plan. Constraint (11) indicates that there may be no more than t order pieces that can be cut across a cutting plan. The constraint number (12) is the relational constraint between y_{jk} and σ_k. Constraints (13) and (14) are the relational constraints between σ_k and z_k. The constraint number (15) calculates the net used amount of the cutting plan. (16) - (23) constraints are sign constraints. The objective function (24) is to minimize the total length and type of cutting plan used. These terms are combined using the weighted sum scalarization method.

3 Problem-solving with column generation

Although mathematical models are developed to give the best solution for t his problem, they cannot be solved when the problem size increases. In addition, the problem has some special technical limitations that have not been included in the literature by now. Our motivation comes from this need to solve the model more effectively and quickly by using the column generation method, as well as to take into account more realistic constraints are aimed, as we explained before. This section presents the column generation method.

Column generation method

Column Generation is a technique for solving linear programs where the numbers of variables are hard to enumerate. According to this approach, only a few variables are needed to determine the optimal solution, as most will assume a zero value [18]. In each step, the master model looks for the best solution, considering only a certain number of variables (equivalent columns). The sub-problem (knapsack problem) investigates whether new columns are added to the master problem in the next increment, reducing the objective function. The objective function of the knapsack problem is the reduced cost of the columns concerning the optimal dual variables corresponding to the optimal solution of the current master problem. If there is a column with a negative reduced cost, that column is added to the master problem and proceeded to the next iteration; otherwise, optimality is achieved [18].

Figure 1 gives the flow chart of the column generation algorithm.

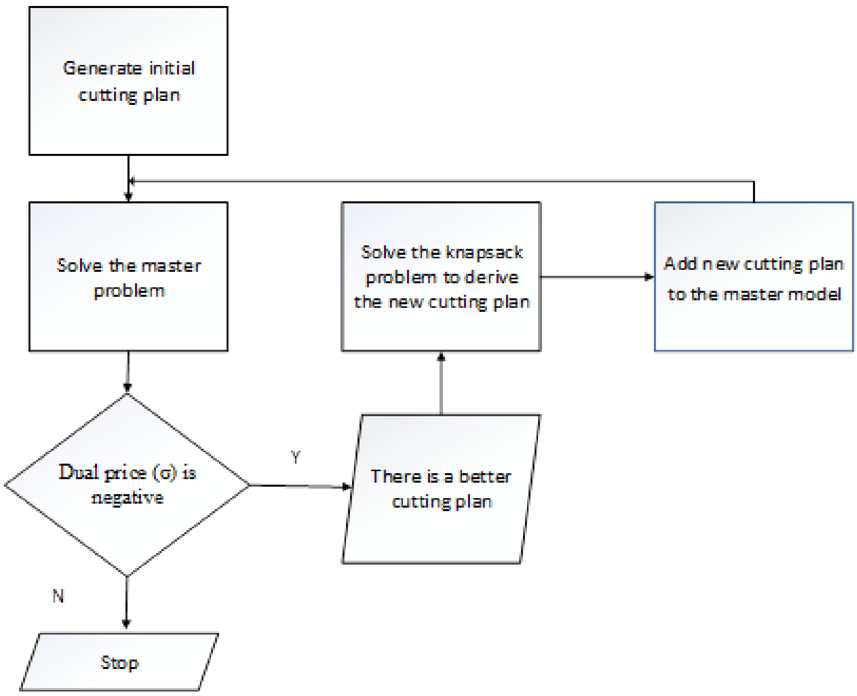

Figure 1. Main steps of the CG algorithm

According to the algorithm, initial cutting plans are derived. The master model is solved using these cutting plans. If the dual price (σ) is negative, there is a better cutting pattern. From the solution of the master model, the dual variable of the related constraint is sent to the knapsack model. The knapsack model is solved, and the new cutting plan will be added to the master model. Then, the loop continues until no better cutting plan is derived.

The complexity of column generation depends on the structure of the main problem and the pricing sub-problems. Because the process involves solving a constrained main problem and a set of sub-problems iteratively to create new columns, the computational burden is usually incurred

by solving the sub-problems and can be complex and time-consuming, especially when these problems involve combinational structures. For the problem considered in this study, a significant reduction is observed in the complexity of the sub-problem compared to the complexity of the integrated problem. The solution times given in Section 5 are an indicator of this reduction.
The column generation technique is used to solve many problems in the literature. Kamran et al. (2020) used a column-generation-based heuristic algorithm and Benders' decomposition technique to schedule patients in operating rooms. With the growth in the number of patients, the feasible sequencing plans grow exponentially. On the other hand, the CG method does not need to price out all the columns, and this makes it beneficial [19].
Faiz et al. (2019) used a column generation framework to solve large-scale instances of vehicle scheduling and routing problems. They first suggest various linear programming models for the problem. According to the experimental results, the column-generation-based approach provides better solutions in terms of solution time [20].
Changchun et al. (2018) developed a column generation-based distributed scheduling algorithm for a constrained project scheduling problem. They decomposed the problem into two parts as: production planning and vehicle scheduling [21].
Section 3 presents the proposed column generation algorithm's main and sub (knapsack) problems and the mathematical models developed to solve them.

Master and knapsack models

Master and knapsack models are developed as below, respectively. Since the indices, parameters, and decision variables are explained in Section 2, only the newly defined ones are included here.
Master model

Parameters

y'_{jk} : the quantity that the order piece j can fit in the width of the k^{th} cutting pattern.

Constraints are given by the following equations including Eq. (3),

$$\propto_{jk} \leq \frac{100x_k}{b_j}, \quad \forall j,k, \tag{25}$$

$$\sum_k \propto_{jk} y'_{jk} \geq d_j, \quad \forall j, \tag{26}$$

$$x_k \geq z_k, \quad \forall k, \tag{27}$$

$$x_k \leq L\, z_k, \quad \forall k, \tag{28}$$

$$x_k \geq b_j \propto_{jk}, \quad \forall j,k. \tag{29}$$

Objective function

$$f = min \sum_k x_k. \tag{30}$$

Constraint (25) calculates how many times order piece j is included in the cutting plan k considering only its length. Decision variable x_k indicates how long cutting plans are used in meters, and it is multiplied by 100 to convert to centimeters. Constraint (26) is demand constraint. Constraints (27) and (28) are the relational constraints between x_k and z_k. Constraint (29) calculates the net amount to be used from the k^{th} cutting pattern (cm).
The objective function (30) minimizes the total net amount used from the cutting patterns.

Knapsack Model

Parameters

y_j : the quantity that the order piece j can fit in the width of the cutting pattern,
w_j : 1, if j^{th} order is included in the cutting pattern, *0* otherwise,
φ_j : dual variables of constraint (26) of main model.

$$y_j \leq t w_j, \qquad \forall j, \tag{31}$$

$$y_j \geq w_j, \qquad \forall j, \tag{32}$$

$$\sum_j e_j y_j \leq G, \tag{33}$$

$$\sum_j w_j \leq c, \tag{34}$$

$$\sum_j y_j \leq t, \tag{35}$$

$$y_j \geq 0, \qquad \forall j, \tag{36}$$

$$w_j \in \{0,1\}, \qquad \forall j, \tag{37}$$

$$f = min\left(1 - \sum_j \varphi_j \, y_j \, (\frac{1}{b_j})\right). \tag{38}$$

Constraints (31) and (32) indicate that if an order piece is included in the cutting pattern, at least one and no more than t order pieces can be cut across the cutting plan. It is ensured by constraint (33) that the total width of all order pieces placed in a cutting plan does not exceed the width of the stock material. Constraint (34) indicates that maximum c different order pieces can be included in the cutting plan. Constraint (35) indicates that there may be no more than t order pieces that can be cut across a cutting plan. (36) - (37) are sign constraints. The objective function is given in Eq. (38).

4 Experimental results

An instance taken from the literature with different numbers of orders is solved both with MI and CG. CPLEX solver of GAMS (version 24.0.2) is used on a PC with 3.60 GHz Intel Core i7 and 16 GB RAM. The time limit of 86,400 CPU seconds is applied for CPLEX runs.
The following section presents the test problems, followed by the toy problem and the test results, which are introduced in the first and second subsection of Section 4, respectively.

Test problems

The first four samples (problem instances) are taken from [8]. The remaining are obtained by adapting the examples generated for one-dimensional cutting problems by Kasımbeyli et al. (2011) [22]. The widths of the order pieces in the one-dimensional problem are multiplied by a parameter, and the length values are obtained. The parameter values of all samples are given below.

Sample 1.

$$n = 5, \quad G = 110, \quad e = (10,20,30,40,60), \quad b = (13,26,39,52,78), \quad d = (6,11,4,20,15).$$

Sample 2.

$$\begin{aligned} n &= 10, \quad G = 120, \\ e &= (10, 20, 30, 40, 60, 15, 25, 35, 45, 65), \\ b &= (13, 26, 39, 52, 78, 19, 32, 45, 58, 84), \\ d &= (7, 11, 3, 20, 15, 5, 10, 13, 20, 15). \end{aligned}$$

Sample 3.

$$\begin{aligned} n &= 20, \quad G = 130, \\ e &= (10, 20, 30, 40, 60, 15, 25, 35, 45, 65, 11, 12, 13, 14, 21, 22, 23, 24, 31, 32), \\ b &= (13, 26, 39, 52, 78, 19, 32, 45, 58, 84, 14, 15, 16, 18, 27, 28, 29, 31, 40, 41), \\ d &= (16, 11, 13, 20, 15, 15, 10, 13, 20, 15, 15, 11, 13, 20, 15, 15, 10, 13, 2, 15). \end{aligned}$$

Sample 4.

$$\begin{aligned} n &= 30, \quad G = 280, \\ e &= (10, 20, 30, 40, 60, 15, 25, 35, 45, 65, 11, 12, 13, 14, 21, 22, 23, 24, 31, 32, 33, 34, 41, 42, 43, 44, 51, \\ &\quad 52, 53, 54), \\ b &= (13, 26, 39, 52, 78, 19, 32, 45, 58, 84, 14, 15, 16, 18, 27, 28, 29, 31, 40, 41, 42, 44, 53, 54, 55, 57, 66, \\ &\quad 67, 68, 70), \\ d &= (5, 11, 3, 20, 15, 5, 10, 13, 20, 15, 5, 11, 3, 20, 15, 5, 10, 13, 20, 15, 5, 11, 3, 20, 15, 5, 10, 13, 20, 15). \end{aligned}$$

Sample 5.

$$\begin{aligned} n &= 40, \quad G = 130, \\ e &= (10, 20, 30, 40, 60, 15, 25, 35, 45, 65, 11, 12, 13, 14, 21, 22, 23, 24, 31, 32, 33, 34, 41, 42, 43, 44, 51, \\ &\quad 52, 53, 54, 61, 62, 63, 64, 66, 67, 68, 69, 70, 71), \\ b &= (13, 26, 39, 52, 78, 19, 32, 45, 58, 84, 14, 15, 16, 18, 27, 28, 29, 31, 40, 41, 42, 44, 53, 54, 55, 57, 66, \\ &\quad 67, 68, 70, 79, 80, 81, 83, 85, 87, 88, 89, 91, 92), \\ d &= (5, 11, 3, 20, 15, 5, 10, 13, 20, 15, 5, 11, 3, 20, 15, 5, 10, 13, 20, 15, \\ &\quad 5, 11, 3, 20, 15, 5, 10, 13, 20, 15, 5, 11, 3, 20, 15, 5, 10, 13, 20, 15). \end{aligned}$$

Sample 6.

$$\begin{aligned} n &= 40, \quad G = 10000, \\ e &= (732, 1746, 1210, 290, 1212, 715, 1471, 1405, 1974, 344, 1699, 172, 351, 1227, 1739, 272, 1903, \\ &\quad 1121, 1326, 107, 726, 1917, 1116, 501, 1599, 439, 821, 485, 361, 860, 1252, 562, 1131, 271, 1075, \\ &\quad 987, 1171, 1979, 228, 1370), \\ b &= (951, 2269, 1573, 377, 1575, 929, 1912, 1826, 2566, 447, 2208, 223, 456, 1595, 2260, 353, 2473, \\ &\quad 1457, 1723, 139, 943, 2492, 1450, 651, 2078, 570, 1067, 630, 469, 1118, 1627, 730, 1470, 352, \\ &\quad 1397, 1283, 1522, 2572, 296, 1781), \end{aligned}$$

$$
\begin{aligned}
d &= (217, 232, 265, 249, 266, 269, 215, 215, 213, 267, 299, 259, 287, 284, 277, 223, 200, 255, 269, 226,\\
&\quad 240, 209, 266, 254, 241, 264, 229, 257, 285, 204, 255, 257, 283, 222, 218, 289, 244, 214, 223, 290).
\end{aligned}
$$

Sample 7.

$$
\begin{aligned}
n &= 100, \quad G = 800,\\
e &= (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30,\\
&\quad 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57,\\
&\quad 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84,\\
&\quad 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100),\\
b &= (1, 2, 3, 5, 6, 7, 9, 10, 11, 13, 14, 15, 16, 18, 19, 20, 22, 23, 24, 26, 27, 28, 29, 31, 32, 33, 35, 36, 37,\\
&\quad 39, 40, 41, 42, 44, 45, 46, 48, 49, 50, 52, 53, 54, 55, 57, 58, 59, 61, 62, 63, 65, 66, 67, 68, 70, 71, 72,\\
&\quad 74, 75, 76, 78, 79, 80, 81, 83, 84, 85, 87, 88, 89, 91, 92, 93, 94, 96, 97, 98, 100, 101, 102, 104, 105,\\
&\quad 106, 107, 109, 110, 111, 113, 114, 115, 117, 118, 119, 120, 122, 123, 124, 126, 127, 128, 130),\\
d &= (5, 11, 3, 20, 15, 5, 10, 13, 20, 15, 5, 11, 3, 20, 15, 5, 10, 13, 20, 15, 5, 11, 3, 20, 15, 5, 10, 13, 20, 15,\\
&\quad 5, 11, 3, 20, 15, 5, 10, 13, 20, 15, 13, 13, 13, 13, 13, 13, 13, 13, 13, 14, 14, 14, 14, 14, 14, 14, 14, 14,\\
&\quad 14, 14, 14, 14, 14, 14, 14, 14, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 16, 16, 16,\\
&\quad 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 17, 17).
\end{aligned}
$$

Sample 8.

$$
\begin{aligned}
n &= 20, \quad G = 110,\\
e &= (50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 32, 33, 42, 44, 27, 19, 10, 40, 20, 30),\\
b &= (65, 66, 67, 68, 70, 71, 72, 74, 75, 76, 41, 42, 54, 57, 35, 24, 13, 52, 26, 39),\\
d &= (273, 20, 27, 19, 32, 28, 100, 82, 55, 42, 48, 35, 29, 50, 35, 40, 23, 42, 51, 32).
\end{aligned}
$$

Sample 9.

$$
\begin{aligned}
n &= 200, \quad G = 1400,\\
e &= (1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30,\\
&\quad 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57,\\
&\quad 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84,\\
&\quad 85, 86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108,\\
&\quad 109, 110, 111, 112, 113, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128,\\
&\quad 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148,\\
&\quad 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168,\\
&\quad 169, 170, 171, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188,\\
&\quad 189, 190, 191, 192, 193, 194, 195, 196, 197, 198, 199, 200),\\
b &= (1, 2, 3, 5, 6, 7, 9, 10, 11, 13, 14, 15, 16, 18, 19, 20, 22, 23, 24, 26, 27, 28, 29, 31, 32, 33, 35, 36, 37,\\
&\quad 39, 40, 41, 42, 44, 45, 46, 48, 49, 50, 52, 53, 54, 55, 57, 58, 59, 61, 62, 63, 65, 66, 67, 68, 70, 71, 72,\\
&\quad 74, 75, 76, 78, 79, 80, 81, 83, 84, 85, 87, 88, 89, 91, 92, 93, 94, 96, 97, 98, 100, 101, 102, 104, 105,
\end{aligned}
$$

$$106, 107, 109, 110, 111, 113, 114, 115, 117, 118, 119, 120, 122, 123, 124, 126, 127, 128, 130, 131, 132, 133, 135, 136, 137, 139, 140, 141, 143, 144, 145, 146, 148, 149, 150, 152, 153, 154, 156, 157, 158, 159, 161, 162, 163, 165, 166, 167, 169, 170, 171, 172, 174, 175, 176, 178, 179, 180, 182, 183, 184, 185, 187, 188, 189, 191, 192, 193, 195, 196, 197, 198, 200, 201, 202, 204, 205, 206, 208, 209, 210, 211, 213, 214, 215, 217, 218, 219, 221, 222, 223, 224, 226, 227, 228, 230, 231, 232, 234, 235, 236, 237, 239, 240, 241, 243, 244, 245, 247, 248, 249, 250, 252, 253, 254, 256, 257, 258, 260),$$

$$d = (5, 11, 3, 20, 15, 5, 10, 13, 20, 15, 5, 11, 3, 20, 15, 5, 10, 13, 20, 15, 5, 11, 3, 20, 15, 5, 10, 13, 20, 15, 5, 11, 3, 20, 15, 5, 10, 13, 20, 15, 13, 13, 13, 13, 13, 13, 13, 13, 13, 14, 14, 14, 14, 14, 14, 14, 14, 14, 14, 14, 14, 14, 14, 14, 14, 14, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 15, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 16, 17, 17, 16, 16, 17, 17, 17, 17, 17, 17, 17, 17, 17, 17, 17, 17, 17, 17, 17, 17, 17, 17, 17, 18, 18, 18, 18, 18, 18, 18, 18, 18, 18, 18, 18, 18, 18, 18, 18, 18, 18, 18, 19, 19, 19, 19, 19, 19, 19, 19, 19, 19, 19, 19, 19, 19, 19, 19, 19, 19, 20, 20, 20, 20, 20, 20, 20, 20, 20, 20, 20, 20, 20, 20, 20, 20, 20, 20, 20, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 21, 22, 22, 22).$$

Two cases are considered for all samples: Cutting plans can be included (1) at most two kinds of order pieces and eight strips ($c = 2$ and $t = 8$) and (2) at most three types of parts and sixteen strips ($c = 4$ and $t = 16$).

Toy problem

Toy problem includes four order pieces, P1, P2, P3, P4, with sizes as 10×13, 20×26, 30×39, 40×52, respectively. The stock material's width is 50. Demands are 4, 5, 3, and 2, respectively.
The toy problem was solved with both CG and IM, and the same solution was obtained. However, while CG achieved this solution in 0.48 seconds, MI runs to the time limit of 86400 seconds. Obtained cutting plans are presented in the following Figure 2, Figure 3, and Figure 4.

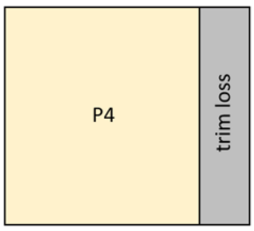

Figure 2. Cutting Plane 1

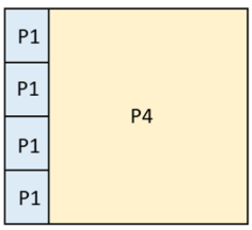

Figure 3. Cutting Plane 2

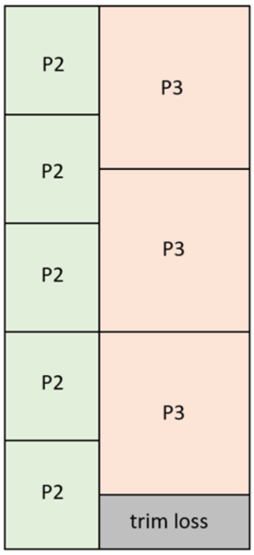

Figure 4. Cutting Plane 3

5 Test results and discussion

Samples have been solved both with IM and CG for two parameter sets: set 1 ($c = 2$ and $t = 8$) and set 2 ($c = 4$ and $t = 16$). The results obtained with set 1 are given in Table 4 and set 2 in Table 5. According to Table 4, IM obtains only the optimum solution for Samples 1 and 2. CG reaches very close solutions (1% and 2% error) to IM in a second. IM could not achieve any feasible solution for other problems in reasonable times. On the other hand, feasible solutions are achieved in reasonable times with CG. For none of the instances in Table 5, an optimum solution is achieved for IM, but feasible solutions are obtained for Samples 1, 2, and 8. For sample 1, both IM and CG obtained the same solution. The solution time of CG is only one second, while the solution time of IM is 86400 seconds (IM stopped by the time limit). For Sample 2, in a second, CG approaches 1% close to the solution obtained by MI in 31288 seconds. For sample 8, on the other hand, CG achieved a more successful objective function value in a much shorter time than IM. CG obtained feasible solutions in reasonable times for the rest of the instances. On the other hand, IM has not obtained a solution within the time limit for these instances. Due to the difference in d values (much bigger than the other instances' d values), the z_{CG} value in Sample 6 for CG is obtained bigger than the other z_{CG} values.

Table 4. Test results for $c = 2$ and $t = 8$

	IM z_{IM}	IM t_{IM}	CG z_{CG}	CG t_{CG}
*sample*1	1248	6244	1261	1
*sample*2	2531	1181	2587	1
*sample*3	-	86400	3002	141
*sample*4	-	86400	2714	822
*sample*5	-	86400	10980	86400
*sample*6	-	86400	1672106	28561
*sample*7	-	86400	3240	1
*sample*8	-	86400	28239	3
*sample*9	-	86400	69514	86400

Table 5. Test results for $c = 4$ and $t = 16$

	IM	IM	CG	CG
	z_{IM}	t_{IM}	z_{CG}	t_{CG}
*sample*1	1222	86400	1222	1
*sample*2	2510	31288*	2533	1
*sample*3	-	86400	2976	4322
*sample*4	-	86400	2754	288
*sample*5	-	86400	10948	86400
*sample*6	-	86400	1661374	86400
*sample*7	-	86400	9604	86400
*sample*8	28621	86400	28144	11777
*sample*9	-	86400	52265	86400

*out of memory.

When Table 4 and Table 5 are examined regarding the effects of different parameter sets on the objective function and solution times, it is observed that when the c and t parameters are increased, generally better objective function values can be obtained, but the solution time is prolonged.

When the experimental results are evaluated in general, it is observed that the mathematical modeling method has some critical limitations for the 1.5-dimensional cutting stock problem with technical constraints. In particular, it has been determined that as the problem size increases, the solution times with traditional mathematical models increase dramatically, and even in some cases, a feasible solution cannot be found. This situation shows that complex constraints and high-dimensional problem structures are difficult to cope with only using classical mathematical modeling techniques.

In this context, the proposed Column Generation (CG) solution method has emerged as an effective alternative, especially for cutting stock problems with large-scale and complex constraints. The CG method offers an approach that can divide the problem into smaller sub-problems and solve each of them in reasonable times. In the numerical experiments, it has been observed that this method produces faster and more successful solutions even for large-sized problems. In particular, it has been concluded that it is successful in coping with technical constraints and therefore can be used as a practical solution method in real-world applications.

6 Conclusion

In this study, the 1.5-dimensional cutting stock problem is considered. This problem covers situations where materials of certain widths and lengths commonly encountered in industrial production processes need to be cut with minimum waste. Unlike the studies in the literature, technical constraints such as order type and number of strips are taken into account together in this study. This approach allows obtaining results closer to real-world applications. In order to solve the problem, a column generation technique, which has been proven to be effective in large-scale and complex cutting stock problems, is proposed. This technique was developed to increase the solution time and accuracy even in high-dimensional problems.

In order to test the accuracy and effectiveness of the study, the test problems and a linear integrated mathematical model from the literature were used. The numerical experiments revealed that as the problem size increases, the solution time of the mathematical model increases dramatically, and even in some cases, a feasible solution cannot be found. However, the proposed column generation approach produces faster and more successful solutions even for large-sized problems. The column generation algorithm performed better than the mathematical model. The column generation approach is not limited to the problem considered in this study but can also be applied to other cutting stock problems in the future. This approach provides a valuable solution, especially

for researchers dealing with large data sets and complex production processes.
The CG solution approach may be insufficient when the problem dimensions are very large, such as in big data. Applying heuristic methods to solve the sub-problem may be a solution in this case.

Declarations

Use of AI tools

The authors declare that they have not used Artificial Intelligence (AI) tools in the creation of this article.

Data availability statement

There are no external data associated with the manuscript.

Ethical approval (optional)

The authors state that this research complies with ethical standards. This research does not involve either human participants or animals.

Consent for publication

Not applicable

Conflicts of interest

The authors declare that they have no conflict of interest.

Funding

No funding was received for this research.

Author's contributions

M.S.: Conceptualization, Methodology, Validation, Writing-Original draft preparation, Software. T.S.: Conceptualization, Methodology, Validation, Writing-Original draft preparation, Data Curation, Software. All authors have read and agreed to the published version of the manuscript.

Acknowledgements

Not applicable

References

[1] Dyckhoff, H., Kruse, H.J., Abel, D. and Gal, T. Trim loss and related problems. *Omega*, 13(1), 59-72, (1985). [CrossRef]

[2] Adakçı, S. *Stok Kesme Problemi: Alüminyum Sektöründe Uygulaması*. Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, (2001).

[3] Dyckhoff, H. A typology of cutting and packing problems. *European Journal of Operational Research*, 44(2), 145-159, (1990). [CrossRef]

[4] Song, X., Chu, C.B., Nie, Y.Y. and Bennell, J.A. An iterative sequential heuristic procedure to a real-life 1.5-dimensional cutting stock problem. *European Journal of Operational Research*, 175(3), 1870-1889, (2006). [CrossRef]

[5] Saraç, T. and Özdemir, M.S. A genetic algorithm for 1.5 dimensional assortment problems with multiple objectives. In Proceedings, *International Conference on Industrial, Engineering and*

Other Applications of Applied Intelligent Systems (IEA/AIE), pp. 41-51, Heidelberg, Berlin, (2003, June). [CrossRef]

[6] Gasimov, R.N., Sipahioglu, A. and Saraç, T. A multi-objective programming approach to 1.5-dimensional assortment problem. *European Journal of Operational Research*, 179(1), 64-79, (2007). [CrossRef]

[7] Kokten, E.S. and Sel, Ç. A cutting stock problem in the wood products industry: a two-stage solution approach. *International Transactions in Operational Research*, 29(2), 879-907, (2022). [CrossRef]

[8] Saraç, T. and Sağır, M. Mixed-Integer programming models for 1.5-dimensional cutting problem with technical constraints. *Journal of the Faculty of Engineering and Architecture of Gazi University*, 36(1), 291-302, (2021). [CrossRef]

[9] Duysak, E., Dülger, İ., Yıldız, N.S., Gümüş, S. and Saraç, T. Teslim zamanlarinin dikkate alındığı 1,5 boyutlu kesme Ve ana malzeme seçimi problemi için bir matsezgisel algoritma. *Endüstri Mühendisliği*, 33(2), 402-412, (2022). [CrossRef]

[10] Vasilyev, I., Ushakov, A.V., Zhang, D. and Ren, J. Generalized multiple strip packing problem: Formulations, applications, and solution algorithms. *Computers & Industrial Engineering*, 178, 109096, (2023). [CrossRef]

[11] Liu, K., Zhang, H., Wang, C., Li, H., Chen, Y. and Chen, Q. Robust optimization for the two-dimensional strip-packing problem with variable-sized bins. *Mathematics*, 11(23), 4781, (2023). [CrossRef]

[12] Bezerra, V.M.R., Leao, A.A.S., Oliveira, J.F. and Santos, M.O. Models for the two-dimensional level strip packing problem-a review and a computational evaluation. *Journal of the Operational Research Society*, 71(4), 606–627, (2020). [CrossRef]

[13] Wäscher, G., Haußner, H. and Schumann, H. An improved typology of cutting and packing problems. *European Journal of Operational Research*, 183(3), 1109-1130, (2007). [CrossRef]

[14] Sugi, M., Shiomi, Y., Okubo, T., Nagai, H., Inoue, K. and Ota, J. Solution of the rectangular strip packing problem considering a 3-stage guillotine cutting constraint with finite slitter blades. *International Journal of Automation Technology*, 14(3), 447-458, (2020). [CrossRef]

[15] Cintra, G.F., Miyazawa, F.K., Wakabayashi, Y. and Xavier, E.C. Algorithms for two-dimensional cutting stock and strip packing problems using dynamic programming and column generation. *European Journal of Operational Research*, 191(1), 61-85, (2008). [CrossRef]

[16] Bettinelli, A., Ceselli, A. and Righini, G. A branch-and-price algorithm for the two-dimensional level strip packing problem. *4OR*, 6, 361-374, (2008). [CrossRef]

[17] Cui, Y.P., Zhou, Y. and Cui, Y. Triple-solution approach for the strip packing problem with two-staged patterns. *Journal of Combinatorial Optimization*, 34, 588-604, (2017). [CrossRef]

[18] Bertoli, F., Kilby, P. and Urli, T. A column-generation-based approach to fleet design problems mixing owned and hired vehicles. *International Transactions in Operational Research*, 27(2), 899-923, (2020). [CrossRef]

[19] Kamran, M.A., Karimi, B. and Dellaert, N. A column-generation-heuristic-based benders' decomposition for solving adaptive allocation scheduling of patients in operating rooms. *Computers & Industrial Engineering*, 148, 106698, (2020). [CrossRef]

[20] Faiz, T.I., Vogiatzis, C. and Noor-E-Alam, M. A column generation algorithm for vehicle scheduling and routing problems. *Computers & Industrial Engineering*, 130, 222-236, (2019). [CrossRef]

[21] Changchun, L., Xi, X., Canrong, Z., Qiang, W. and Li, Z. A column generation based distributed scheduling algorithm for multi-mode resource constrained project scheduling problem. *Computers & Industrial Engineering*, 125, 258-278, (2018). [CrossRef]

[22] Kasimbeyli, N., Saraç, T. and Kasimbeyli, R. A two-objective mathematical model without cutting patterns for one-dimensional assortment problems. *Journal of Computational and Applied Mathematics*, 235(16), 4663-4674, (2011). [CrossRef]

Mathematical Modelling and Numerical Simulation with Applications (MMNSA)
(https://dergipark.org.tr/en/pub/mmnsa)

How to cite this article: Sağır, M. & Saraç, T. (2024). Column generation approach for 1.5-dimensional cutting stock problem with technical constraints. *Mathematical Modelling and Numerical Simulation with Applications*, 4(3), 335-350. https://doi.org/10.53391/mmnsa.1492749