Asia-Pacific Journal of Science and Technology: Volume: 29. Issue: 02. Article ID.: APST-29-02-04. Research Article

APST

Asia-Pacific Journal of Science and Technology

https://www.tci-thaijo.org/index.php/APST/index

Published by the Research and Graduate Studies Division,
Khon Kaen University, Thailand

A separation technique for solving the two-dimensional skiving and cutting stock problem

Kasitinart Sangngern and Aua-aree Boonperm*

Department of Mathematics and Statistics, Faculty of Science and Technology, Thammasat University, Thailand
*Corresponding author: aua-aree@mathstat.sci.tu.ac.th

Received 12 October 2022
Revised 6 February 2023
Accepted 8 February 2023

Abstract

A two-dimensional skiving and cutting stock problem (2D-SCSP) is a version of the cutting stock problem (CSP) that allows for skiving. In this problem, output sheets are assumed to be longer but thinner than input coils. Consequently, subsets of two or more coils must be considered for joining and cutting to meet the demand of output sheets, leading to the joining cost. Solving this problem requires all subsets of coil groups and all patterns of each subset. Afterward, an integer linear programming problem is formulated to find optimal subsets of coil groups and patterns in order to minimize material and setup costs. However, it is nearly impossible to find all subsets of coil groups and their feasible patterns. Therefore, in this paper, we introduce a heuristic method that reduces the dimensions of a 2D-SCSP by separating it into two steps: generating the feasible subset of coil groups and finding optimal patterns by solving a one-dimensional CSP. The algorithm repeats until the demand is met. The results are then compared with the results obtained by using the column-and-row generation (C&R) approach. Based on the computational results, the proposed method can reduce the computational time by at least 70% compared with the C&R method, and the differences in the objective values of the two methods were found to be less than 0.0001%.

Keywords: Heuristic, Integer programming problem, Separation technique, Skiving and cutting stock problem

1. Introduction

A cutting stock problem (CSP), first mentioned by Gilmore and Gomory, [1, 2] is an obstacle that challenges many manufacturing industries. It is characterized by a set of large objects (input items) that are to be cut into a set of smaller objects (output sheets). It seeks the optimal solution for cutting a large object into a smaller object to meet the demand of output sheets while minimizing the cost (or minimizing the cutting patterns used). A method to solve CSP is to find all feasible patterns and formulate an integer programming model to find the optimal number of patterns to use. Therefore, it is a combinatorial optimization problem. However, it is nearly impossible to find all feasible cutting patterns. This method for solving CSP calls for an enormous model, leading to significant computational time. As a result, many heuristic methods have been introduced to reduce computational time [3, 4]. CSP can be classified based on four criteria: dimensionality, the kind of assignment, the assortment of large objects, and the assortment of small objects [5, 6]. There are two main types of CSP widely employed: one- and two-dimensional CSP (1D-CSP and 2D-CSP) [7-9].

The skiving stock problem (SSP) is a counterpart of the CSP [10, 11]. It seeks to maximize single product output by skiving (joining) multiple tiny stock pieces. Zak [11] referred to the dual bin packing problem as an SSP. There is a wealth of research on solving SSPs [12, 13].

The combined skiving and cutting stock problem (SCSP) refers to a 1D-CSP with the skiving option. Johnson et al. [14] proposed the earliest study in 1997 to solve the 1D-CSP with the skiving option. Their algorithm can be divided into two parts. In the first stage, the completed and auxiliary rolls are cut from the master rolls. In the second stage, some auxiliary rolls are merged with the completed rolls to fulfill the demand. The authors used the commercial software package MAJIQTRIM to solve the problem. Another interesting research on one-dimensional SCSP (1D-SCSP) was presented by Tanir et al. [15]. They defined the 1D-CSP with divisible items. Then they addressed a problem in the steel sector to minimize trim loss and the number of welds. Moreover, a nonlinear programming model for this NP-hard problem was proposed.

Song et al. [16] introduced a 1.5-dimensional CSP in 2006, in which output sheets are longer and composed of several short sheets. Since the component lengths of the sheets to be cut out are not specified, this problem is not a classical 2D-CSP. The problem has similarities to the 1D-CSP with more complex combinatorial conditions. Therefore, it is called 1.5D-CSP. With an incomplete enumerative method, the authors solved the problem of optimizing material consumption and manufacturing time while considering several operational constraints.

Next, Chen et al. [17] studied the SCSPs encountered in the paper and plastic film industries in 2017. Several nonstandard rolls, derived from previous cutting processes, were used to produce finished rolls through the skiving and cutting process. The authors also devised and solved an integer programming model for two types of SCSP using the sequential value correction approach.

In 2020, Wang et al. [18] addressed the problem definition and basic formulation of a 2D-SCSP, in which input coils are shorter but wider than output sheets, and skiving is allowed for input coils. In a 2D-SCSP, the total cost comprises both material and setup costs. The material cost covers the area of input coils; the setup cost is related to the new cutting pattern used. Thus, the setup cost for each pattern should be different. The horizontal guillotine is used for cutting since the demand requires the total length. Based on the cutting stage, it is a 1D-CSP. The authors proposed a heuristic method, called the column-and-row generation (C&R) method. Their algorithm consists of two subproblems: a y-generating subproblem and a row-generating subproblem that is a nonlinear program. Then, a decomposition-based exact solution method for dealing with the nonlinear subproblem and a heuristic method to find an integer solution are proposed. Both generating subproblems are based on a subproblem in the column generation method. A y-generating subproblem is a subproblem for finding a new pattern for the current subset of the coil group. If a new pattern does not exist, a new subset of the coil group and its pattern are needed to solve a row-generating subproblem. Although it avoids the complexity of solving the nonlinear subproblem by linearization and relaxing constraints, the obtained solution may not be feasible. Consequently, it takes more time to fix an infeasible solution.

Generally, in a 2D-SCSP, it is difficult to construct all suitable subsets of input coil groups and their feasible patterns. Although the algorithm of Wang et al. [18] reduced the step of finding all subset input coil groups and their feasible patterns, their method leads to a significant amount of computation time since a row-generating subproblem is used to find a suitable subset of input coil groups. Consequently, if the suitable set of coil groups and the corresponding good pattern can be generated without solving a nonlinear program, it may lead to a shorter computational time. The present paper proposes a heuristic method to solve the 2D-SCSP. The algorithm starts by reducing the dimensions of the 2D-SCSP. The method is separated into two steps: generating a possible subset of coil groups without solving a nonlinear program and finding optimal patterns by solving a 1D-CSP.

This paper is organized as follows. Section 2 presents the details of the problem, the basic model, and the proposed method. In section 3, the computational results are discussed. Finally, findings are concluded in the last section.

2. Materials and methods

2.1 Problem definition and basic formulation

A 2D-SCSP aims to cut rectangular steel coils into smaller rectangular sheets to minimize the overall production cost. A crucial assumption for this problem is that output sheets are usually longer but thinner than input coils. Each output sheet is covered by the total length which must be greater than or equal to the required demand. Skiving is necessary to join several coils, followed by the cutting process.

Figure 1 illustrates this problem. It shows five input coils and four output sheets of varying sizes. The allowable length of the production sheet should be more than the length of the output sheet. For example, on output sheet 1, it is permissible to make a group of short items with a width of 5 ft and a total length of at least 200 ft. By skiving and cutting processes, the input coil 1 and 5 are joined and cut to satisfy the demand of output sheet 1, as shown in Figure 1. To solve this problem, Wang et al. [18] proposed a mathematical model for the 2D-SCSP by using the notations provided in Table 1.

Table 1 List of notations

Sets	Definition
J	Input coil set; each coil $j \in J$ be characterized by its length L_j, width W_j, and area $A_j = W_j L_j$
I	Output sheet set; each sheet $i \in I$ be characterized by its length (or demand) d_i and width w_i
S	Coil group set; each group $s \in S$ be characterized by its length $l_s = \sum_{j\in s} L_j$, area $a_s = \sum_{j\in s} A_j$, narrowest width $W_{s,min}$, and widest width $W_{s,max}$

Table 1 (continued) List of notations

Sets	Definition
P	Patterns set; each pattern $p \in P$ be characterized by a vector $\left(u_1^p, \dots, u_{\lvert I \rvert}^p\right) \in Z_+^{\lvert I \rvert}$
P_s	Feasible pattern set for coil group s; each pattern $p \in P_s$, and its width $w_p = \sum_{i \in I} w_i \, u_i^p$ must not exceed the narrowest width $W_{s,min}$
Parameters	
c_{ps}	The total production cost of using pattern p for coil group s, where $c_{ps} = c_{ps}^w + c_{ps}^u$
c_{ps}^w	The material cost of using pattern p for coil group s, where $c_{ps}^w = \sum_{j \in s} A_j = a_s$
c_{ps}^u	The unit setup cost
Variables	
x_s	$x_s = 1$, if coil group s is used; otherwise, $x_s = 0$
y_{ps}	$y_{ps} = 1$, if coil group s uses a cutting pattern p; otherwise $y_{ps} = 0$

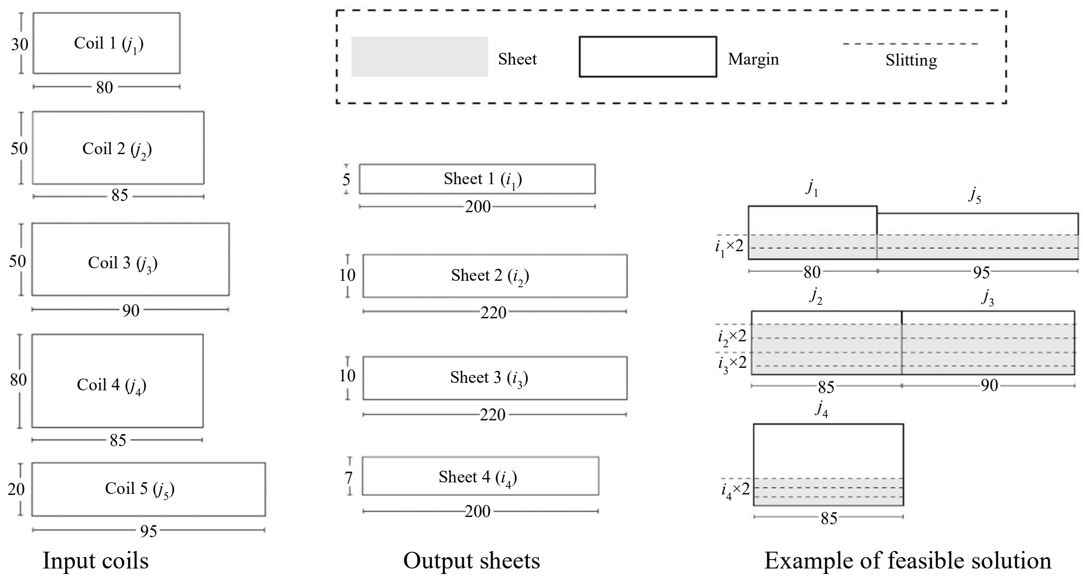

Figure 1 Schematic representation of 2D-SCSP

From the given notations, the mathematical model proposed by Wang et al. [18] for solving a 2D-SCSP can be written as follows:

$$\text{Min} \quad \sum_{s \in S} \sum_{p \in P_s} c_{ps} \, y_{ps} \tag{1}$$

$$\text{s.t.} \quad \sum_{s \in S : j \in s} x_s \leq 1 \qquad \forall j \in J \tag{2}$$

$$\sum_{p \in P_s} y_{ps} \leq x_s \qquad \forall s \in S \tag{3}$$

$$\sum_{s \in S} \sum_{p \in P_s} l_s \, u_i^p \, y_{ps} \geq d_i \qquad \forall i \in I \tag{4}$$

$$x_s \in \{0,1\} \qquad \forall s \in S \tag{5}$$

$$y_{ps} \in \{0,1\} \qquad \forall s \in S \quad \forall p \in P_s. \tag{6}$$

The objective function (1) aims to minimize the total cost of all skiving and cutting patterns**.** Constraint (2) guarantees that each coil has only one piece. As per constraint (3), if a coil group s is used, then one pattern at most is selected for this group. The demand constraints are shown in constraint (4). Constraints (5) and (6) are binary conditions for each decision variable.

All subsets of input coil groups and all patterns must be found to solve this model, which is a difficult task. Thus, Wang et al. [18] proposed the C&R method for a 2D-SCSP and a diving heuristic method for finding an integer solution.

2.2 C&R method for 2D-SCSP

To solve a 2D-SCSP, the C&R approach begins by initializing a short-restricted master problem **(**SRMP**)** using a limited subset of coil groups and several possible patterns. Each variable y_{ps} updates the SRMP. This step is known as addressing y-generating pricing subproblems (y- PSPs). When the pattern for group s cannot be determined, the row generation pricing subproblem (row-PSP) is solved to provide a pair for a new subset of the coil group and its feasible pattern. If they exist, the SRMP is updated and the y-PSP and row-PSP are repeated. Otherwise, the optimum linear programming solution is reached.

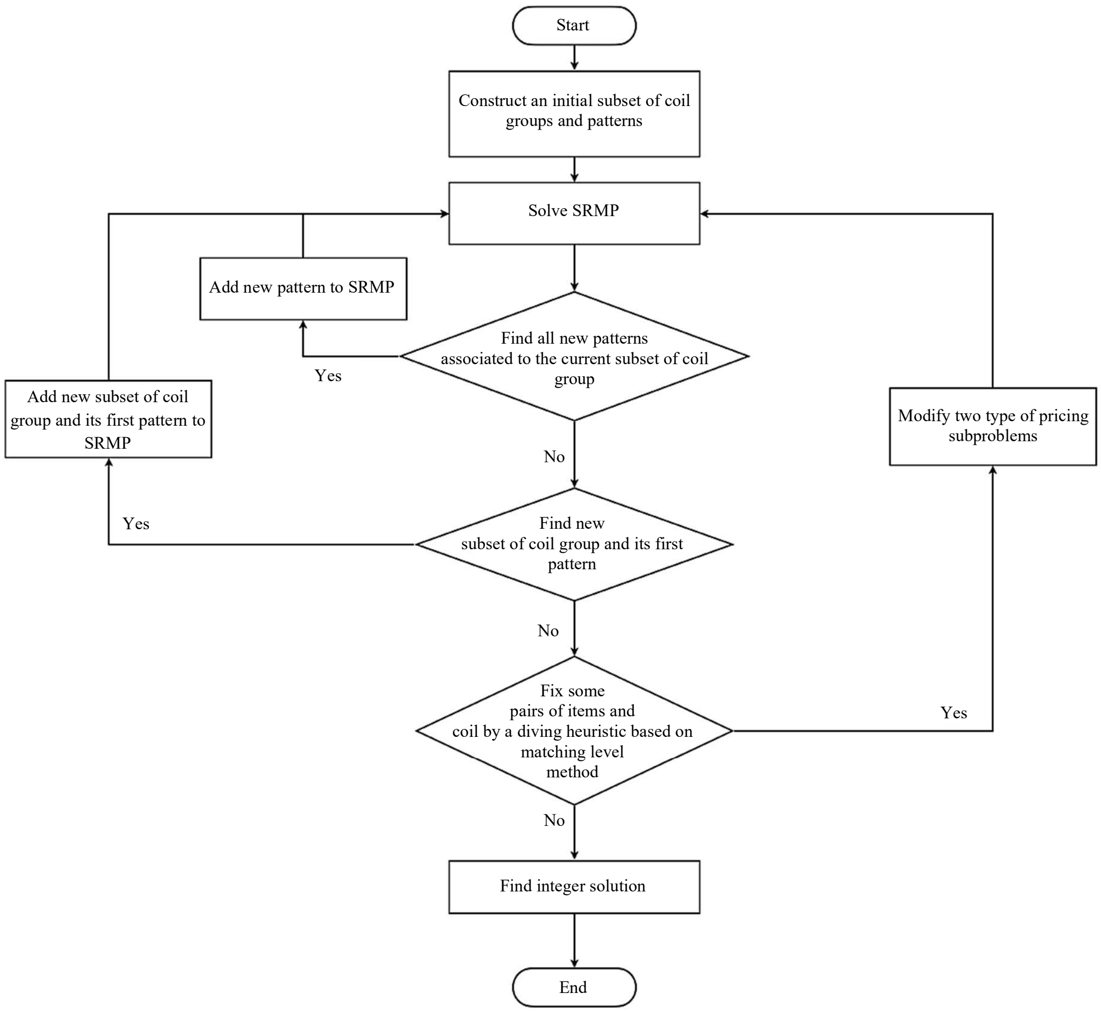

Figure 2 Flow chart of Wang et al.'s [18] framework

Since the row-PSP is formed as a nonlinear integer program, Wang et al. [18] linearized and solved it. If the solution is not inconsistent, a heuristic method is used for fixing some pairs of x and y. The C&R approach by Wang et al. can be summarized using the flow chart in Figure 2.

This algorithm wastes time solving a nonlinear program to find a suitable subset of coil groups. Thus, we can decrease the computational time if a suitable subset of coil groups can be generated easily before cutting.

2.3 Proposed method

A heuristic method for solving a 2D-SCSP, which avoids solving a nonlinear subproblem, is introduced in this paper. It is separated into three steps: 1) generating a feasible subset of coil groups, 2) assigning output sheets and 3) finding optimal patterns by solving a 1D-CSP. These processes can avoid solving the nonlinear subproblem presented by Wang et al. [18].

2.3.1 Generating a feasible subset of coil groups

In the C&R method, the nonlinear subproblem is required for finding a new subset of the coil group and its pattern. To avoid solving the nonlinear subproblem, the generation of suitable coil groups before cutting is significant. Hence, we first consider all input coils by width to generate the coil groups. If the widths of all coils in the same group are similar, then the gap between $W_{s,min}$ and $W_{s,max}$ will be small, and trim loss after the cut can be reduced (Figure 3). Thus, the first step of the proposed algorithm is considering input coils by their width. This can be achieved by using the frequency distribution table.

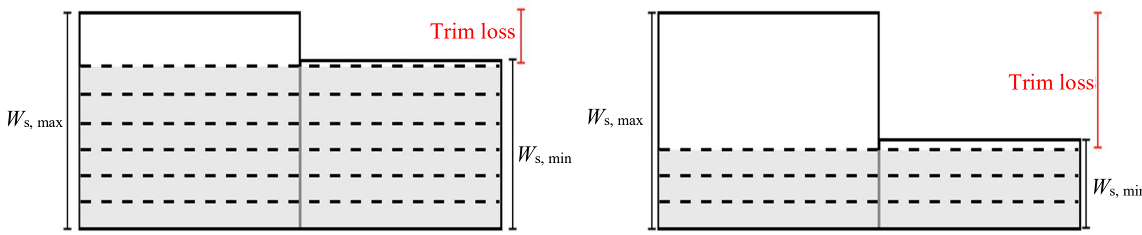

Figure 3 Trim loss in different cases

2.3.2 Assigning output sheets

After all subsets of coil groups are grouped, we consider the width of all output sheets. We have the condition for each input coil group that limits the pattern by the narrowest width. Hence, we randomly assign an output sheet to each input coil. Subsequently, each group' s available width, area of each group, residual demand, and area before reinserting other sheets are computed until all sheets are assigned. For each loop of the random insertion, if the available area of the coil group is not enough for the residual area sheet, we randomly reinsert until sufficient available area is available.

2.3.3 Forming the subproblem

Following the assignment of all output sheets, the subproblem is formed by a pair of an input coil group and the corresponding group of the output sheet. Let I_s be a set of output sheets assigned to a coil group s. The integer program of the subproblem according to a coil group s, which is 1D-CSP, can be shown as follows:

$$\text{Min} \quad \sum_p x_p \tag{7}$$

$$\text{s.t.} \quad \sum_p u_i^p x_p \geq d_i \qquad \forall i \in I_s \tag{8}$$

$$x_p \geq 0 \text{ and integer} \qquad \forall p \in P_s, \tag{9}$$

where u_i^p is the number of output sheets i cut from pattern p, and x_p is the number of times pattern p used. Therefore, the proposed method can be summarized in the following algorithm.

Algorithm 1. A separation technique for 2D-SCSP
Input: (W_j, L_j) for all input coils $j \in J$, and (w_i, d_i) for all output sheets $i \in I$.
1. Use the frequency distribution table in statistics for grouping input coils.
2. Assign output sheets by considering their width.
3. Solve each subproblem generated by a pair of a coil group and a group of output sheets and return the used pattern for computing the total cost.

Output: the patterns used and the total cost of production.

2.3.4 Illustration of the proposed method

The following example illustrates the steps of the proposed heuristic. The problem consists of five input coils and four output sheets, as shown in Figure 1. In the first step, a frequency distribution table is used to summarize the data for all input coils, as shown in Table 2.

Table 2 Frequency distribution table

Width	Coil
19.5–39.5	1, 5
39.5–59.5	2, 3
59.5–79.5	-
79.5–99.5	4

Then, all input coils are grouped, as shown in Figure 4.

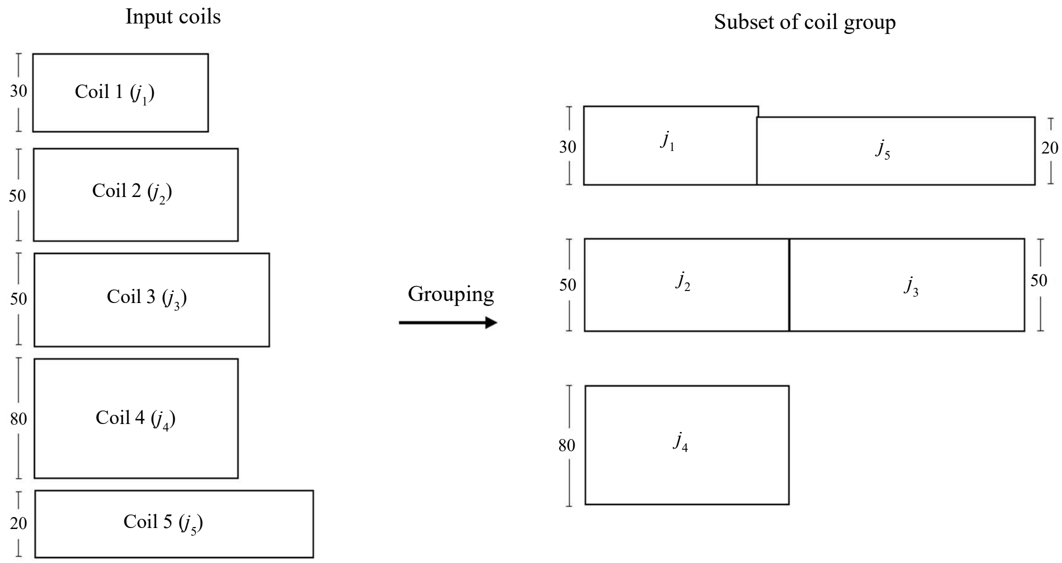

Figure 4 Grouping of input coils by using the frequency distribution table

Next, all output sheets are randomly assigned to the available subset of the coil group (Figure 5).

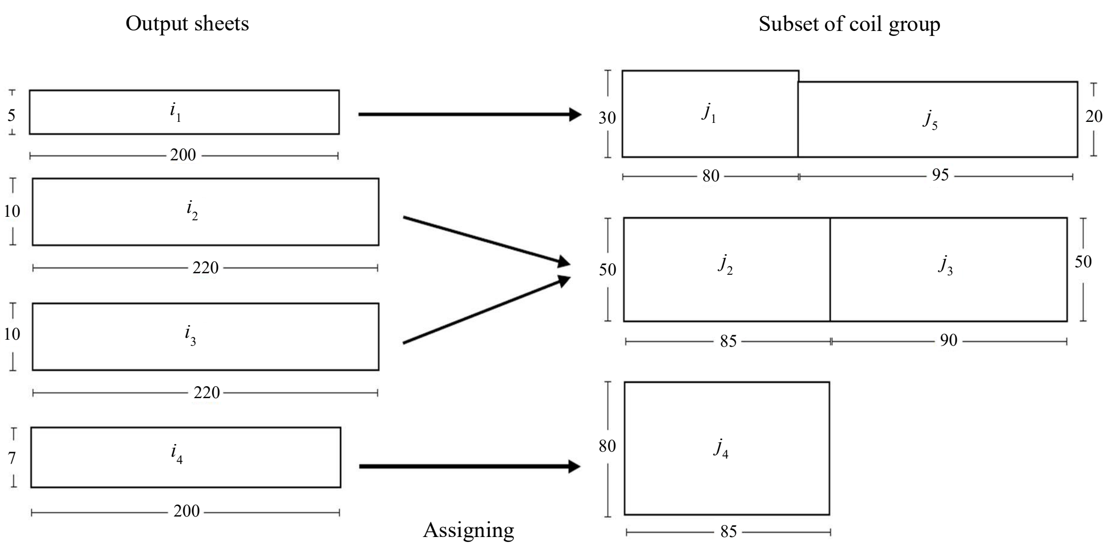

Figure 5 Assigning of output sheets to the available subset of the coil group

Then, a pair of a subset of the coil group and an assigning output sheet form a 1D-CSP by ignoring the width of the subset. Finally, all 1D-CSPs are solved, and subsets of coil groups and patterns used are obtained (Figure 6).

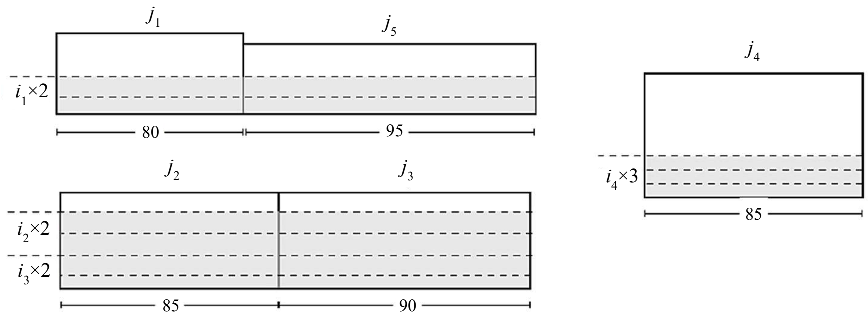

Figure 6 An example of the solution

3. Results and discussion

This section delineates the characteristics of the random test cases, after which the results obtained by using the proposed algorithm are presented. The last subsection covers the discussion.

3.1 Information on test cases

The experiments to solve a 2D- SCSP using the proposed heuristic method were conducted using the characteristic ranges of input coils and output sheets used by Wang et al. [18]. We divided the test cases into four types: small scale ($s_1 - s_5$), medium scale ($m_1 - m_5$), large scale ($d_1 - d_5$), and extremely large scale ($e_1 - e_5$). The intervals of the characteristics of each scale used are shown in Table 3. We implemented the proposed algorithm by using Python coding on Google Colaboratory, and these tests were run on an 8GB MacBook Air (M1, 2020).

Table 3 The interval of characteristics of input coils and output sheets for each scale

	Input coils	Interval	Output sheet	Interval
small scale ($s_1 - s_5$),	Number of input coils	[15–25]	Number of output sheets	[10–20]
	Range of width	[375–1600]	Range of width	[75–650]
	Range of length	[100–5000]	Range of length	[1000–12000]
medium scale ($m_1 - m_5$),	Number of input coils	[30–55]	Number of output sheets	[15–25]
	Range of width	[375–1600]	Range of width	[75–1000]
	Range of length	[100–5000]	Range of length	[1000–20000]
large scale ($d_1 - d_5$),	Number of input coils	[90–110]	Number of output sheets	[50–60]
	Range of width	[375–1600]	Range of width	[75–1000]
	Range of length	[100–5000]	Range of length	[1000–30000]
extremely large scale ($e_1 - e_5$),	Number of input coils	[115–135]	Number of output sheets	[55–65]
	Range of width	[375–1600]	Range of width	[75–1000]
	Range of length	[100–5000]	Range of length	[1000–50000]

3.2 Production cost

Production costs comprise material cost and unit setup cost. The material cost of coil group s is $c_{ps}^{w} = \sum_{j \in s} A_j$, and the unit setup cost is set as the number of coils in the group multiplied by a constant value c_0; $c_{ps}^{u} = c_0 \times \sum_{i \in I} u_i^p$.

3.3 Computational results

For generating a subset of coil groups, the maximum number of coils in each coil group is determined first. In this experiment, we used the roundup of the square root of the number of input coils as the maximum number of coils in each coil group (t_0), since the roundup of the square root value leads to each coil group having a similar number of coils in each coil group. Moreover, to test whether the maximum number of coils in each coil group affects the computational time, we tested the performance for each test case by comparing the different maximum numbers of coils in each coil group— t_0, t_1, t_2, t_3, where t_1, t_2, and t_3 are $t_0 + 1, t_0 + 2$, and $t_0 + 3$, respectively. The increment of t_0 may lead to varying the number of coils in each coil group. However, each coil group's maximum number of coils should not be vastly different.

As shown in Tables 4 and 5, the computational results consist of the computational time in seconds (time), the objective value, and the maximum number of coils in each coil group. The bold text in the time column means that the time is minimum compared with others.

Table 4 Computational results ($c_0 = 10$)

Case	t_0		t_1		t_2		t_3	
	Time	Obj.	Time	Obj.	Time	Obj.	Time	Obj.
s_1	3.2641	48,007,085	2.3668	48,007,085	1.172	48,007,085	1.3317	48,007,085
s_2	3.4369	62,517,179	2.4284	62,517,189	0.9343	62,517,179	1.1118	62,517,179
s_3	3.0814	24,272,585	1.8692	24,272,605	1.0634	24,272,575	0.9522	24,272,575
s_4	0.9935	23,173,748	0.9141	23,173,748	0.9379	23,173,748	0.8056	23,173,748
s_5	2.9227	39,071,639	2.4324	39,071,659	1.0058	39,071,749	1.0309	39,071,679
m_1	14.4238	84,982,809	13.2197	84,982,749	11.5027	84,982,749	11.6098	84,982,749
m_2	11.5616	114,682,739	11.1898	114,682,739	11.2609	114,682,769	11.2563	114,682,719
m_3	11.6124	71,466,702	11.0232	71,466,702	11.1424	71,466,712	10.9775	71,466,722
m_4	12.312	130,907,804	11.3806	130,907,844	11.2519	130,907,704	11.1735	130,907,704
m_5	11.959	94,302,569	11.6223	94,302,449	11.4456	94,302,459	11.5475	94,302,589
d_1	55.1257	218,354,761	53.2274	218,354,931	52.5888	218,354,871	53.2289	218,354,611
d_2	45.0349	228,808,683	42.4122	228,808,763	42.5871	228,808,813	41.9928	228,808,543
d_3	68.7935	330,436,273	62.7017	330,436,403	66.6453	330,436,433	63.0712	330,436,783
d_4	84.8826	206,299,167	72.9681	206,298,067	73.5997	206,298,077	83.2457	206,298,097
d_5	62.7448	230,716,333	62.2407	230,716,393	63.1544	230,716,323	63.0226	230,717,173
e_1	136.0156	444,690,823	133.2944	444,691,293	133.8246	444,691,353	133.1077	444,692,193
e_2	143.0847	340,353,909	142.5423	340,353,869	143.0133	340,353,999	143.278	340,354,289
e_3	136.0704	345,495,513	133.1373	345,494,973	133.0126	345,494,883	133.1262	345,494,793
e_4	234.2235	316,948,480	132.5521	316,948,420	133.6144	316,948,720	133.0483	316,947,720
e_5	145.1543	311,901,076	142.9264	311,901,296	143.5267	311,901,146	143.4422	311,902,316

Table 5 Computational results (average values)

Case	t_0		t_1		t_2		t_3	
	Time	Obj.	Time	Obj.	Time	Obj.	Time	Obj.
Small	2.7397	39,408,447.2	2.0022	39,408,457.2	1.0227	39,408,467.2	1.0464	39,408,453.2
Medium	12.3738	99,268,524.6	11.6871	99,268,496.6	11.3207	99,268,478.6	11.3129	99,268,496.6
Large	63.3163	242,923,043.4	58.7100	242,922,911.4	59.7151	242,922,903.4	60.9122	242,923,041.4
Ext. L.	158.9097	351,877,960.2	136.8905	351,877,970.2	137.3983	351,878,020.2	137.2005	351,878,262.2
Avg.	59.3349	183,369,493.9	52.3225	183,369,458.9	52.3642	183,369,467.4	52.6180	183,369,563.4

From Tables 4 and 5, it can be concluded that the maximum number of coils in each group which is greater than the roundup of the square root of the number of input coils can decrease the computational time. However, the difference in the maximum size of the coil group does not significantly affect the objective value. To show the performance of the proposed method, we compared the results with those obtained by using the C&R method.

3.4 Comparison of the two methods

The C&R method used by Wang et al. [18] gives the optimal linear programming solution. However, it involves linearization and decomposition to avoid the nonlinear program, after which the diving heuristic is performed to find an integer solution. On the other hand, our method starts with a group of input coils and then assigns each output sheet to each group of coils by considering the width. Thus, the proposed method can avoid the nonlinear program and moderate the process of finding the new subset. To compare Wang et al.'s method with our method, we implemented the proposed algorithm and C&R method by using Python coding on Google Colaboratory, and these tests were run on an 8GB MacBook Air (M1, 2020). The intervals of the characteristics of the small scale ($sp_1 - sp_5$) used are shown in Table 6. The computational times and the objective values of the integer solutions obtained by the proposed method and C&R approach are presented in Table 7, and the comparison of the computational time between the proposed method and C&R method is shown in Figure 7.

Table 6 The interval of characteristics of input coils and output sheets for small-scale cases

Input coils	Interval	Output sheet	Interval
Number of input coils	[5–15]	Number of output sheets	[5–10]
Range of width	[400–1600]	Range of width	[75–650]
Range of length	[100–4500]	Range of length	[1000–12000]

Table 7 Comparison of the computational results of the proposed method and the C&R approach

Case	t_0		t_1		t_2		t_3		C&R	
	Time	Obj.	Time	Obj.	Time	Obj.	Time	Obj.	Time	Obj.
sp_1	0.6182	25,996.511	0.6804	25,996.509	0.5960	25,996.509	0.5844	25,996.509	2.5337	25,996.504
sp_2	0.7001	32,813.413	0.5532	32,813.413	0.4350	32,813.413	0.6036	32,813.413	4.7829	32,813.408
sp_3	0.6264	17,370.320	0.5170	17,370.317	0.5600	17,370.317	0.6408	17,370.317	2.4687	17,370.312
sp_4	0.3883	13,785.505	0.4047	13,785.505	0.4009	13,785.505	0.4908	13,785.505	3.2488	13,785.500
sp_5	1.8860	37,140.853	1.1775	37,140.853	1.1476	37,140.853	1.1737	37,140.853	6.5228	37,140.848
Average	0.8438	25,421.320	0.6665	25,421.319	0.6279	25,421.319	0.6987	25,421.319	3.9114	25,421.314

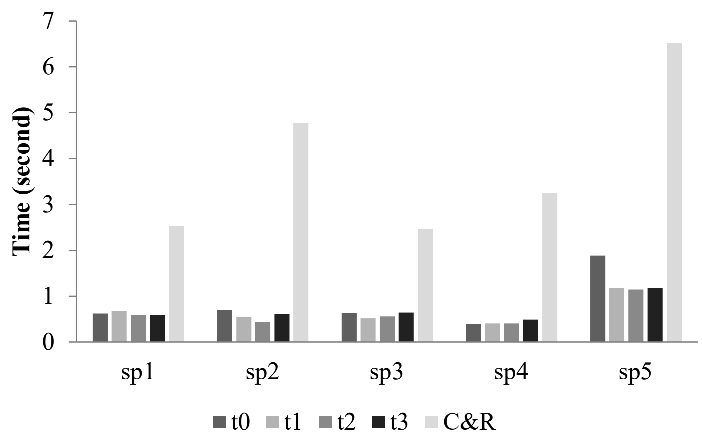

Figure 7 Comparison of the computational times of the proposed method and the C&R approach

Table 8 shows percentage differences between our method with a fixed maximum size of the coil group and the C&R method used by Wang et al., [18] where the columns designated "t_0, C&R," "t_1, C&R," "t_2, C&R," and "t_3, C&R" represent the percentage differences. The percentage difference of the integer solution is computed by $\frac{SEP-CR}{CR} \times 100$, where SEP and CR are the objective values from our method and the C&R method, respectively. The percentage difference in the computational time is computed by $\frac{TSEP-TCR}{TCR} \times 100$, where $TSEP$ and TCR are the computational time of our method and the C&R method, respectively.

Table 8 Percentage differences of objective values and computational times

Case	t_0, C&R		t_1, C&R		t_2, C&R		t_3, C&R	
	Time (%)	Obj. (%)	Time (%)	Obj. (%)	Time (%)	Obj. (%)	Time (%)	Obj. (%)
sp_1	-75.6002	0.000027	-73.1451	0.000019	-76.4748	0.000019	-76.9323	0.000019
sp_2	-85.3620	0.000015	-88.4333	0.000015	-90.9032	0.000015	-87.3781	0.000015
sp_3	-74.6260	0.000046	-79.0565	0.000028	-77.3142	0.000028	-74.0410	0.000028
sp_4	-88.0468	0.000036	-87.5422	0.000036	-87.6601	0.000036	-84.8901	0.000036
sp_5	-71.0847	0.000013	-81.9477	0.000013	-82.4055	0.000013	-82.4055	0.000013

Table 7 shows that the percentage differences are less than 0.0001%, meaning that our method results in almost the same integer solution as obtained by Wang et al. [18] Moreover, our method can reduce the computational time by at least 70% compared to the C&R method.

4. Conclusion

The 2D-SCSP is an interesting challenge faced by many industries. It seeks the optimal solution for cutting a large object into a smaller object to meet the demand of output sheets while minimizing the cost (or while minimizing the cutting patterns used). In this paper, we proposed a heuristic method that reduces the dimensions of the 2D-SCSP by separating it into two steps: 1) generating a feasible subset of coil groups and 2) finding optimal patterns by solving a 1D-CSP. The first step avoids solving the nonlinear subproblem, and the second step involves the cutting phase to cover demand.

The performance of the proposed technique was assessed by solving the random test cases. From the results, we found that our method involves less computational time since the proposed algorithm can easily pair a subset of coil groups and output sheets. Our method divides the master problem into many subproblems for a pair of a subset of coil groups and corresponding output sheets, leading to less computational time than that achieved with the C&R method. However, our method does not provide the optimal integer solution. Moreover, since the assigning phase affects the solution and each sheet can be cut from more than one subset of the coil group, it needs to be more flexible. If the assigning phase can be improved, the solution will be close to the optimal solution. In the future, we would attempt to extend the problem (add some property) and improve the method to decrease the disadvantage of randomization.

5. Acknowledgments

The first author gratefully acknowledges the financial support provided by a Ph. D. scholarship from Thammasat University, 1/2020. This work was supported by Thammasat University Research Unit in Fixed Points and Optimization.

6. References

[1] Gilmore PC, Gomory RE. A linear programming approach to the cutting stock problem. Oper Res. 1961;9(6):849-859.

[2] Gilmore PC, Gomory RE. A linear programming approach to the cutting stock problem—Part II. Oper Res. 1963;11(6):863-888.

[3] Wäscher G, Gau T. Heuristics for the integer one-dimensional cutting stock problem: A computational study. OR Spektrum. 1996;18(3):131-144.

[4] Ma N, Liu Y, Zhou Z. Two heuristics for the capacitated multi-period cutting stock problem with pattern setup cost. Comput Oper Res. 2019;109:218-229.

[5] Dyckhoff H. A typology of cutting and packing problems. Eur J Oper Res. 1990;44(2):145-159.

[6] Wäscher G, Haußner H, Schumann H. An improved typology of cutting and packing problems. Eur J Oper Res. 2007;183(3):1109-1130.

[7] Cherri AC, Arenales MN, Yanasse HH, Poldi KC, Gonçalves Vianna AC. The one-dimensional cutting stock problem with usable leftovers—A survey. Eur J Oper Res. 2014;236(2):395-402.

[8] Dolatabadi M, Lodi A, Monaci M. Exact algorithms for the two-dimensional guillotine knapsack. Comput Oper Res. 2012;39(1):48-53.

[9] Hadjiconstantinou E, Christofides N. An exact algorithm for general, orthogonal, two-dimensional knapsack problems. Eur J Oper Res. 1995;83(1):39-56.

[10] Peeters M, Degraeve Z. Branch-and-price algorithms for the dual bin packing and maximum cardinality bin packing problem. Eur J Oper Res. 2006;170(2):416-439.

[11] Zak EJ. The skiving stock problem as a counterpart of the cutting stock problem. Int Trans Oper Res. 2003;10(6):637-650.

[12] Martinovic J, Scheithauer G. Integer linear programming models for the skiving stock problem. Eur J Oper Res. 2016;251(2):356-368.

[13] Martinovic J, Scheithauer G. The skiving stock problem and its relation to hypergraph matchings. Discrete Optim. 2018;29:77-102.

[14] Johnson MP, Rennick C, Zak E. Skiving addition to the cutting stock problem in the paper industry. SIAM Rev. 1997;39(3):472-483.

[15] Tanir D, Ugurlu O, Guler A, Nuriyev U. One-dimensional cutting stock problem with divisible items: A case study in steel industry. TWMS J App Eng Math. 2019;9(3):473-484.

[16] Song X, Chu CB, Nie YY, Bennell JA. An iterative sequential heuristic procedure to a real-life 1.5-dimensional cutting stock problem. Eur J Oper Res. 2006;175(3):1870-1889.

[17] Chen Y, Song X, Ouelhadj D, Cui Y. A heuristic for the skiving and cutting stock problem in paper and plastic film industries. Int Trans Oper Res. 2017;26(1):157-179.

[18] Wang D, Xiao F, Zhou L, Liang Z. Two-dimensional skiving and cutting stock problem with setup cost based on column-and-row generation. Eur J Oper Res. 2020;286(2):547-563.