

Research Article

A Heuristic Approach for Two-Dimensional Rectangular Cutting Stock Problem considering Balance for Material Utilization and Cutting Complexity

Dianjian Wu^{1,2} and Guangyou Yang^{1,2}

¹Agricultural Machinery Engineering Research and Design College, Hubei University of Technology, Wuhan 430068, China

²Hubei Province Agricultural Equipment Intelligent Engineering Technology Center, Wuhan 430068, China

Correspondence should be addressed to Guangyou Yang; pekka@126.com

Received 12 May 2021; Accepted 10 October 2021; Published 8 November 2021

Academic Editor: Francisco Javier Fernández Fernández

Copyright © 2021 Dianjian Wu and Guangyou Yang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The common staged patterns are always required during the cutting process for separating a set of rectangular items from rectangular plates in manufacturing industries. Two-staged patterns can reduce cutting complexity at the expense of material utilization; three-staged patterns do the opposite. Combining these two types of staged patterns may be a good balance for two contradictory objectives of material utilization and cutting complexity. A heuristic approach is proposed to solve the two-dimensional rectangular cutting stock problem with a combination of two-staged general patterns (2SGP) and three-staged homogenous patterns (3SHP). Firstly, the 2SGP and 3SHP are constructed by using recursive techniques. The pattern with the larger value is selected as the candidate pattern. Then, the value of each item is corrected according to the current candidate pattern. A cutting plan accurately satisfying all items demand is obtained by using the sequential heuristic algorithm. Finally, the cutting plan with a minimum number of used plates is achieved by applying the iterative algorithm. The computational results indicate that the proposed heuristic approach is more effective in material utilization and cutting complexity than the two published algorithms with staged patterns.

1. Introduction

The two-dimensional rectangular cutting stock problem (2DRCSP) is always encountered in many manufacturing industries, such as steel products, paper, wood, and glass. A set of rectangular items with specific demand need to be produced from cutting the same rectangular plates with the optimization objective of minimizing the number of used plates. This feature of the 2DRCSP is referred to as a two-dimensional rectangular single stock size cutting stock problem (2DRSSCSP) in Wäscher et al. [1]. The solution of the 2DRSSCSP is a cutting plan including a set of different patterns with their frequencies. Therefore, how to design patterns with optimization strategy becomes the most noteworthy problem.

Some practical restrictions need to be observed while solving the 2DRSSCSP. Lodi et al. [2] introduced two

common application restrictions of “Guillotine cuts” and “Orientation.” Based on these two restrictions, the 2DRSSCSP can be categorized as four variants, including 2DRSSCSP-GO, 2DRSSCSP-GR, 2DRSSCSP-FO, and 2DRSSCSP-FR, which are shown in Figure 1. *G* represents that the cutting style must be guillotine cuts, where each cut must be performed from edge-to-edge parallel to each side of the used rectangular plate. *F* represents that the cutting style can be free, namely, not considered guillotine cuts. *O* denotes that the items are oriented, that is, not rotated. *R* denotes that the items can be rotated by 90°.

This paper mainly focuses on the research of the 2DRSSCSP-GR, where the cutting style must be guillotine cuts, and each item is allowed to be rotated by 90°. The staged patterns are always constructed to satisfy guillotine cuts. The common staged patterns are two-staged patterns and three-

staged patterns. In general, the two-staged patterns have simpler cutting complexity but less material utilization than the three-staged patterns. Two-staged patterns are welcomed to reduce cutting complexity when the material cost is relatively cheaper than the guillotine cuts cost. Three-staged patterns are selected to improve material utilization otherwise. It is difficult to consider two contradictory objectives of material utilization and cutting complexity during solving the 2DRSSCSP-GR with only two-staged patterns or three-staged patterns. Therefore, combining two types of staged patterns for the 2DRSSCSP-GR may be an effective approach to balance material utilization and cutting complexity.

We firstly introduce two-staged patterns and three-staged patterns based on two types of strips in Figure 2. Each homogenous strip in Figure 2(a) is only composed of multiple same items. Each general strip in Figure 2(b) is composed of multiple items of different sizes. Therefore, there are four common types of staged patterns, including two-staged homogenous patterns (2SHP), two-staged general patterns (2SGP), three-staged homogenous patterns (3SHP), and three-staged general patterns (3SGP) in Figure 3. The arrows denote the position of a guillotine cut. The numbers nearby the arrows denote the sequence of guillotine cut in Figure 3. For example, as shown in Figure 3(a), the first guillotine cut (GC_1) is horizontal to cut off the topside strip. The second guillotine cut (GC_2) is vertical to separate three items. Generally, the 2SHP have the lowest material utilization and simplest cutting complexity, and 3SGP have the opposite. The 2SGP and 3SHP can compromise material utilization and cutting complexity. Meanwhile, all items designed on the plate can be divided exactly with three guillotine cuts according to the 2SGP and 3SHP, which may be more suitable for a cutting machine during the cutting stock process. Therefore, we choose the 2SGP and 3SHP as candidate staged patterns for constructing a cutting plan in this paper.

Staged patterns are necessary to satisfy guillotine cutting, which is restricted by cutting machines during the manufacturing process of cutting flat glass in the glass industry, rocks in the granite and marble industries, steel blocks in the metallurgical industry, and so on. For instance, the cutting process for steel plates is shown in Figure 4, where a rectangular steel plate is cut into demanded items by three guillotine cuts. In the first guillotine cutting with single side crop shear, a rectangular steel plate is divided into segment 1 and segment 2. In the second guillotine cutting with double side crop shear, segment 1 is separated into strips 1, 2, and 3. Meanwhile, segment 2 is separated into strips 4, 5, and 6. In the final guillotine cutting with multiple side crop shear, strips 1, 2, and 3 are all cut into two pieces of item 1. Strip 4 is cut into three pieces of item 1 and one waste. Strip 5 is cut into two pieces of item 3 and another waste. Strip 6 is exactly cut into four pieces of item 2. Similar to the paper cutting problem and the glass cutting problem, the steel plate cutting problem has also been treated as the application of the 2DRCSPP with staged patterns.

Based on the 2SGP and 3SHP, a heuristic approach is proposed to solve the 2DRSSCSP-GR. Firstly, the 2SGP and

3SHP are built by using recursive techniques. The candidate pattern with the objective of a large pattern value can be chosen from the generated 2SGP and 3SHP. Then, the value of each item is corrected according to the current candidate pattern. A cutting plan including 2SGP and 3SHP is achieved with a sequential heuristic algorithm until all items demands are fulfilled. Finally, the cutting plan with the minimum number of used plates is achieved by applying the iterative algorithm.

The remainder of the paper is organized as follows: the related literature is reviewed in Section 2. The generation procedures for the 2SGP and 3SHP are introduced in Section 3. An iterative sequential heuristic procedure (ISHP) considering both 2SGP and 3SHP is proposed for solving the 2DRSSCSP-GR in Section 4. The comparison results are reported in Section 5, followed by the conclusions in Section 6.

2. Literature Review

To review the gradual advancement of the literature for solving the 2DRCSPP clearly, the related literature in the last two decades is reviewed according to chronological order. The critical analysis of the proposed methodology in each literature is also presented simultaneously.

Hifi and Roucairol [3] presented an approximate algorithm and an exact algorithm for solving the 2DRCSPP with restriction of two-staged guillotine cutting. Their approximate algorithm was constructed by using a dynamic programming procedure to solve multiple bounded knapsack problems. Their exact algorithm was built by a tailored branch-and-bound procedure, which determined lower and upper bounds by dynamic programming properties and branched based on the solution space. As their proposed algorithms mainly apply dynamic programming, which is a typical greedy algorithm. The solution by their proposed algorithms is easily trapped into local optimization. Therefore, the proposed algorithms can be improved to get a better solution.

Vanderbeck [4] focused on the 2DRCSPP with restriction of three-staged guillotine cutting. The optimization objective focused on minimizing waste and other production goals, including aging stock, optional orders, and fixed setup cost. Their solution approach is a repeated use of the column generation method to get an approximate solution for the 2DRCSPP. This approach can get an optimal solution in acceptable computation time when the type number of order pieces is less. However, the solution space becomes huge when the type number of order pieces is large. It is hard to achieve an optimal solution.

Alvarez-Valdes et al. [5] developed several heuristic approaches for solving the 2DRCSPP based on the framework of the Gilmore–Gomory column generation scheme [6]. They use different heuristic algorithms, including the GRASP and tabu search, to generate each new cutting pattern by solving the subproblem. It can increase solution space at the expense of computation time. Their proposed method actually belongs to column generation techniques, which is suitable for 2DRCSPP with less number of order piece type.

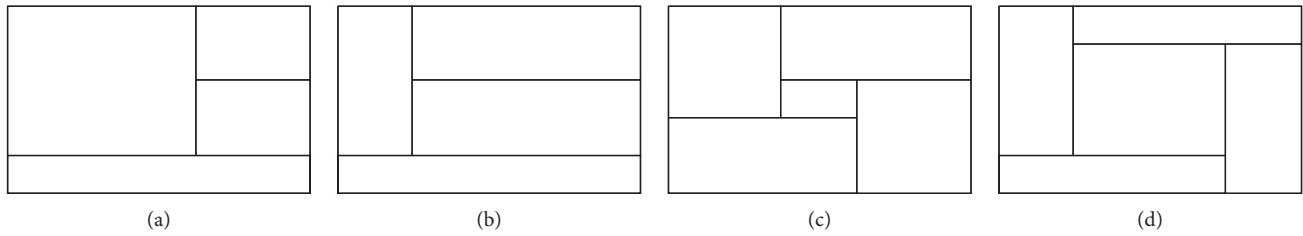


FIGURE 1: Four variants of the 2DRSSCSP: (a) 2DRSSCSP-GO. (b) 2DRSSCSP-GR. (c) 2DRSSCSP-FO. (d) 2DRSSCSP-FR.

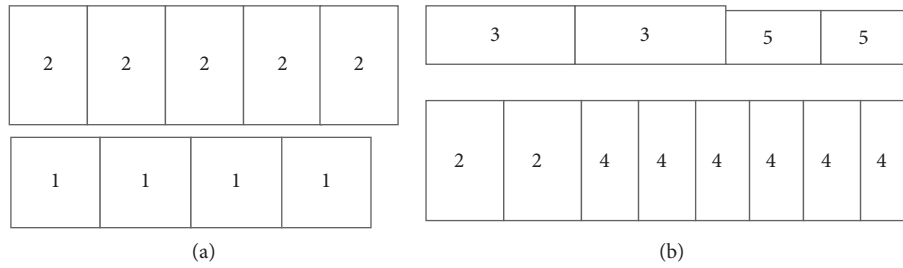


FIGURE 2: Two types of strips. (a) Homogenous strips. (b) General strips.

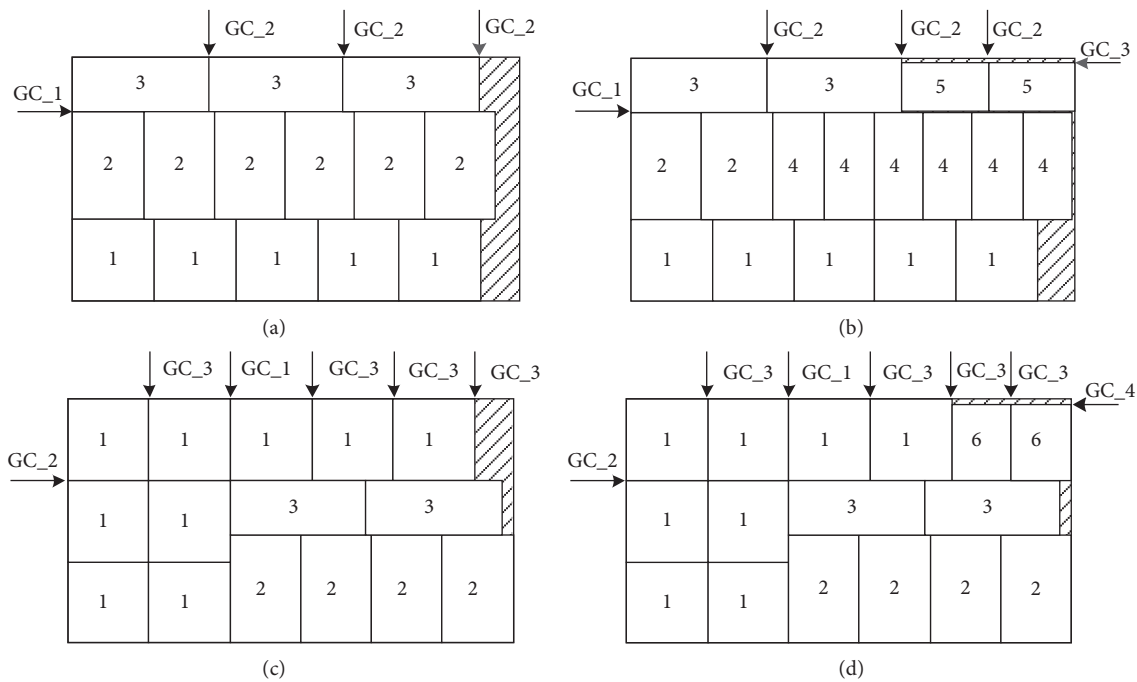


FIGURE 3: Four common types of staged patterns. (a) 2SHP. (b) 2SGP. (c) 3SHP. (d) 3SGP.

Suliman [7] proposed a sequential heuristic procedure for the 2DRCSPP to minimize the trim loss. The sequential heuristic procedure consisted of three steps. A width-cutting pattern with the minimum width trim loss was obtained in the first step. A cutting pattern was achieved when the length was determined with the minimum length trim loss. The frequency of the generated cutting pattern was confirmed based on the remaining number of each item. The proposed procedure is the simple sequential heuristic algorithm, which can get a solution in a short computation time. But it

can trap into locally optimal solution and hard to get globally optimal solution.

Belov and Scheithauer [8] solved the 2DRCSPP with restriction of two-staged guillotine cutting by applying a branch-and-cut-and-price algorithm, which is a combination of the LP-based approach in [6] and branch-and-bound with column generation in the Chvatal-Gomory cutting planes. Based on their combination approach, all optimal solutions can be achieved for medium instances in acceptable computation time. However, there were still

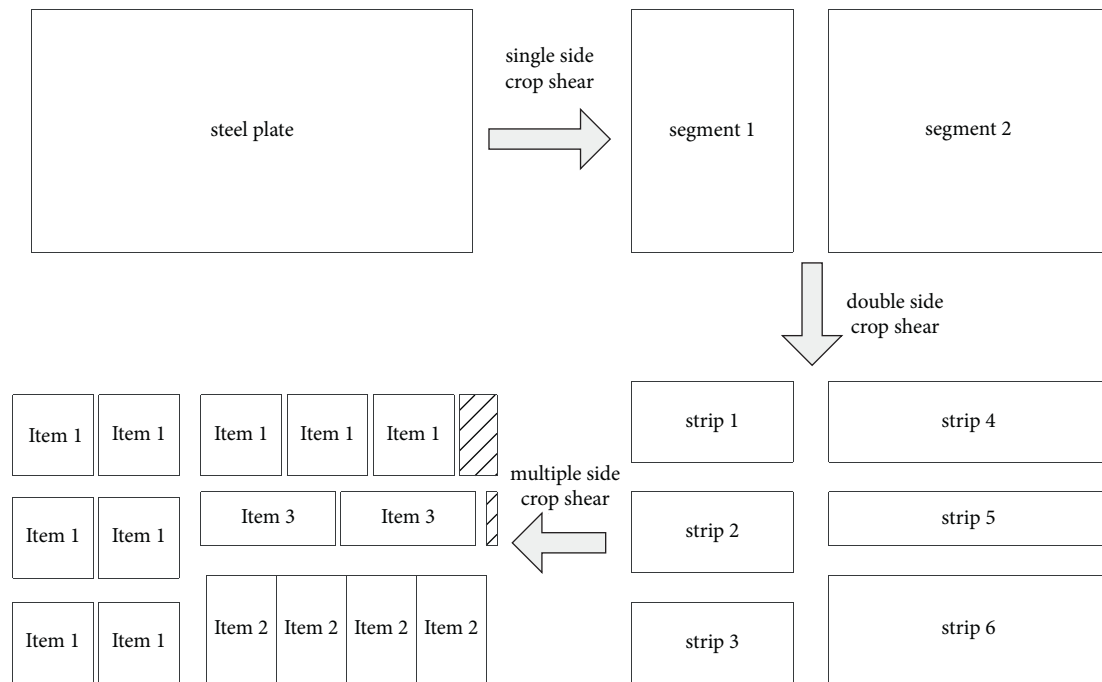


FIGURE 4: Cutting process for steel plates.

unsolved instances for larger instances because of the huge complexity of column generation with the large item type.

Cintra et al. [9] focused on solving the 2DRCS with the constraint of guillotine cutting. Their approach framework was also column generation. For generating each new cutting pattern, they used two dynamic programming-based algorithms for the Rectangular Knapsack problem. Their method also was not suitable for the instances with large item type.

Cui [10] proposed exact and heuristic algorithms to solve the 2DRCS with three-staged guillotine cutting. The exact algorithm was actually a branch-and-bound approach, where the lower and upper bounds were obtained by applying dynamic programming techniques to solve the knapsack problems. Meanwhile, one heuristic algorithm was used to get the initial lower bound for the main framework of the exact algorithm. The other heuristic algorithm underestimated the upper bounds of the blocks generated for the main framework of the exact algorithm. The proposed approach just only considered homogenous strips, which was the basic element of the cutting pattern. It can be capable of solving problems with large scale at the expense of reducing the solution space.

Hifi et al. [11] solved the 2DRCS with restriction of two-staged guillotine cutting and fixed-orientation item. The two-staged cutting patterns were generated by a strip-generation procedure. Their proposed beam search was a truncated branch-and-bound algorithm, which selected the elite nodes based on an evaluation operator. Their method can solve the problems with large scale. But the two-staged cutting pattern was a simple and special cutting pattern, whose material utilization was hard to be improved with iterative value correction. The approach was also to get approximate solution by reducing solution space.

Silva et al. [12] introduced an integer programming model for the 2DRCS with four restrictions: two-stage, three-stage, exact, and nonexact. They also addressed other constraints, such as the length of the cuts and the value of the remaining plate. Their proposed model was solved by a commercial solver (CPLEX). Their conclusion claimed that most instances with many item types could be solved in two hours. Several instances were dealt with over two hours. Obviously, it is a long computational time to get an optimal solution for the 2DRCS occurring in the real enterprise application.

Macedo et al. [13] presented an integer linear programming arc-flow model for the 2DRCS with restriction of two-staged guillotine cutting. They solved the model with the commercial solver (CPLEX). The computational results showed that the proposed arc-flow model also could not solve all the instances. Meanwhile, some instances needed a long computational time to get their solutions.

Cui and Huang [14] focused on the 2DRCS with restriction of three-staged guillotine cutting. The optimization objective was minimizing the sum of used plate cost and cutting cost. The problem was solved by the classical column generation framework, where the new cutting patterns were generated by a pattern-generation procedure. They researched the 2DRCS with multiple objectives in a different perspective.

Cui [15] also addressed the 2DRCS with three-staged patterns, which can be constructed in three steps. A set of rectangular items can be selected to generate strips in the first step. A set of generated strips can be chosen to build segments in the second step. A set of generated segments can be selected to achieve cutting patterns in the third step. Three large knapsack problems are constructed according to each

step. The proposed dynamic programming algorithm was used to solve each knapsack problem to get an optimal pattern. The approach is simple and fast to get the solution. However, the greedy nature of dynamic programming may be influenced to get the optimal solution.

Cui and Zhao [16] presented a heuristic algorithm to solve the 2DRCSP with two-staged patterns. The proposed algorithm was a special column generation method, which solved the residual problems repeatedly until all demands were satisfied. Each pattern was generated by using a procedure for the constrained single large object placement problem. They combined the column generation and heuristic algorithm. It is a new idea to solve the 2DRCSP.

Cui et al. [17] focused on the 2DRCSP with the objective of plate minimization and pattern reduction. They proposed a sequential grouping heuristic based on the framework of the sequential heuristic procedure. It also applied a grouping technique to select the items, which generated the next pattern by using the dynamic programming algorithm. Meanwhile, the item value can be adjusted with the sequential value correction method. Their approach was actually an improved sequential heuristic procedure. It can be suitable for solving the 2DRCSP with multiple optimization objectives.

Cui et al. [18] also addressed the 2DRCSP with three-staged patterns. They proposed a heuristic algorithm according to the construction process of the three-staged pattern. They used the dynamic programming algorithm three times to generate strips, segments, and patterns, respectively. Their approach can get a solution in a short computation time. But it was hard to get an optimal solution because of the greedy nature of the used algorithm.

Chen et al. [19] proposed a sequential value correction heuristic for solving the 2DRCSP with constrained three-staged patterns. Their main method was similar to the sequential grouping heuristic proposed in Cui and Zhao [16]. The difference was that Chen et al. [19] used three-staged homogenous patterns as the element for the cutting plan. Cui and Zhao [16] applied simple block patterns, otherwise.

Andrade et al. [20] solved the nonexact two-stage 2DRCSP with usable leftovers. They built two bilevel mathematical programming models, which can be reformulated as one-level MIP models. They solved the models by applying the branch-and-cut method of CPLEX.

Cui et al. [21] proposed a heuristic algorithm for the 2DRCSP with guillotine cutting. Their algorithm included three procedures. A pattern-generation procedure was used to generate a set of triple-block patterns in the first procedure. A plan-generation procedure was applied to generate a solution in the second procedure. A plan-improvement procedure is also used to generate an improved solution. Their heuristic algorithm was also the same as the sequential value correction heuristic presented in Chen et al. [19].

Nascimento et al. [22] presented the 2DRCSP with guillotine cutting. They considered the possibility of 2 and 3 cutting stages in a predefined sequence. A two-level iterative approach was used to solve integer linear programming and constraint programming models for the special 2DRCSP.

The models that are in the approach were solved with the libraries of the CPLEX. They also cannot solve all instances in acceptable computation time.

Velasco and Uchoa [23] considered the constrained two-dimensional guillotine cutting problem. They presented a weight adjusting algorithm based on integer programming, which was solved using a dynamic programming state-space relaxation. The solution can be obtained by using several iterations of a GRASP-based primal heuristic. The computation time of their approach is not short to get an optimal solution.

Long et al. [24] solved the 2DRCSP with nonexact three-stage patterns. Two additional practical constraints had appeared in the cutting production line of steel mills. They were the length between any two adjacent guillotine cuts and the number of guillotine cuts in the second stage by one. Four heuristic algorithms based on column generation were introduced to solve the practical 2DRCSP. Their approach mainly included the column generation technique. Obviously, the computation time may be too long for some instances with larger item type.

Martin et al. [25] proposed a top-down cutting approach to solve the mixed-integer linear programming models for the 2DRCSP with Guillotine cutting. Their approach was similar to the branch-and-bound algorithm. It used a binary tree to represent cutting patterns. The root node was the initial plate, and branches correspond to guillotine cuts. They also used a commercial solver (CPLEX) to solve the established model.

We can give the summary based on all reviewed literature. There are mainly three optimization methods for solving the 2DRCSP. The first optimization method is based on the integer programming model, which is established according to the 2DRCSP with special constraints and solved with branch-and-bound or existing commercial solver such as CPLEX. Based on [3, 10, 13, 20, 22, 25], the computational complexity is too high with large item type. The runtime of solving several instances is even over two hours. The second optimization method is based on a linear programming model, which is built with special constraints, such as the staged pattern. It can often be solved with the framework of the column generation approach. Based on [4, 6, 8, 9, 14, 24], the computational complexity becomes high with item type increase. But the runtime of solving all instances is in two hours (acceptable computation time) at the expense of solution space, such as using staged patterns. The third optimization method is the heuristic procedure, whose main thought is to generate a cutting pattern using the remaining items with some pattern-generation algorithm repeatedly until all items demand is fulfilled. Based on [7, 15, 19, 21, 23], the computational complexity is moderate with large item type. The runtime of solving all instances is in a short time.

According to the contrastive analysis, we select the third method to solve the 2DRCSP. Meanwhile, our method is different from other reviewed literature. We consider the combination of multiple staged patterns to solve the 2DRCSP. It is the first time to integrate two types of staged patterns to solve the 2DRSSCSP-GR. Therefore, this paper aims to build two types of staged patterns (2SGP and 3SHP)

and present the ISHP approach combined with the 2SGP and 3SHP in the remaining content.

3. Generation Procedures for 2SGP and 3SHP

In the 2DRSSSCSP-GR, m types of rectangular items need to be cut off exactly from plates of the same size $L \otimes W$ (length \otimes width). The type- i item can be characterized by the size $l_i \otimes w_i \otimes v_i \otimes d_i$ (length \otimes width \otimes value \otimes demand). To satisfy two restrictions (guillotine cuts and rotation) in the 2DRSSSCSP-GR, the 2SGP and 3SHP are used to meet the restriction of guillotine cuts (G). $2m$ types of strips are considered to satisfy the restriction of the item allowed to be rotated by 90° (R). We firstly introduce generation procedures for 2SGP and 3SHP. To simplify the presentation, only X-patterns are presented in this section. The Y-patterns are easy to be achieved by swapping the length and width of the used plates.

The notations are used to denote the following related parameters:

K : number of types of item allowed to be rotated by 90° , that is, $K = 2m$

l_k : length of type- k item, $l_k = l_i$ when $k = 1, 2, \dots, m$; $l_k = w_i$ when $k = m + 1, m + 2, \dots, 2m$, $i = 1, 2, \dots, m$

w_k : width of type- k item, $w_k = w_i$ when $k = 1, 2, \dots, m$; $w_k = l_i$ when $k = m + 1, m + 2, \dots, 2m$, $i = 1, 2, \dots, m$

r_i : remaining demand of type- i item

$f_{2SGP}(k, x)$: value of type- k strip $x \otimes w_k$ for 2SGP, $k = 1, 2, \dots, m$

$n_{2SGP}(k, x, i)$: number of type- i item in type- k strip $x \otimes w_k$ for 2SGP, $i = 1, 2, \dots, m$

$F_{2SGP}(y)$: value of segment $L \otimes y$ for 2SGP, $y = 1, 2, \dots, W$

$V_{2SGP}(k)$: total value of possible items in type- k strip $L \otimes w_k$ for 2SGP

$N_{2SGP}(y, i)$: number of type- i item in segment $L \otimes y$ for 2SGP, $y = 1, 2, \dots, W$

$F_{3SHP}(x, y)$: value of segment $x \otimes y$ for 3SHP

$V_{3SHP}(k, x)$: total value of possible existing items in type- k strip $x \otimes w_k$ for 3SHP

$n_{3SHP}(k, x, i)$: number of only including type- i item in type- k strip $x \otimes w_k$ for 3SHP, $i = 1, 2, \dots, m$

$N_{3SHP}(x, y, i)$: number of type- i item in segment $x \otimes y$ for 3SHP

$T_{3SHP}(x)$: value of 3SHP including segment $x \otimes W$ and segment $(L - x) \otimes W$ at the dividing position x

3.1. Generation Procedure for 2SGP. The generation procedure for 2SGP always includes generation of general strips and generation of 2SGP. The generation of general strips provides the candidate strips for 2SGP. Assume that all items are ordered by their width, that is, $w_1 \leq w_2 \leq \dots \leq w_{2m}$. $2m$ types of general strips can be generated for 2SGP in the plate $L \otimes W$, where the type- k strip has the size $L \otimes w_k$,

$k = 1, 2, \dots, 2m$. The number of each item in type- k strip $L \otimes w_k$ is determined by solving the following recursion:

$$f_{2SGP}(k, x) = \max\{f_{2SGP}(k-1, x), v_i + f_{2SGP}(k, x - l_k) | l_k \leq x\}, \quad (1)$$

$$f_{2SGP}(k, x) = \sum_{i=1}^m v_i n_{2SGP}(k, x, i), n_{2SGP}(k, x, i) \leq r_i.$$

The recursion is initialized with $f_{2SGP}(0, x) = 0$ and $n_{2SGP}(0, x, i) = 0$. The value of type- k strip $x \otimes w_k$ ($x = 1, 2, \dots, L$) can be selected from the maximum value between the value of strip $x \otimes w_{k-1}$ and the other one, which is the total value of one type- i item and strip $(x - l_k) \otimes w_k$. Meanwhile, the number of each item in type- k strip $L \otimes w_k$ can be determined by $n_{2SGP}(k, x, i)$.

All general strips can be determined by the decision variable $n_{2SGP}(k, x, i)$, which is considered by $k = 1, 2, \dots, 2m$, $x = 1, 2, \dots, L$, $i = 1, 2, \dots, m$. Therefore, the complexity of recursion (1) is $O(2m^2L)$.

The general strips have been generated through a horizontal combination of demanded items with the above recursion. The 2DGP for the plate $L \otimes W$ can be achieved through a vertical combination of generated strips with the following recursion:

$$F_{2SGP}(y) = \max\{F_{2SGP}(y-1), V_{2SGP}(k) + F_{2SGP}(y - w_k) | w_k \leq y\},$$

$$V_{2SGP}(k) = \sum_{i=1}^m \{v_i \min[n_{2SGP}(k, L, i), r_i - N_{2SGP}(y - w_k, i)]\}. \quad (2)$$

This recursion is initialized with $F_{2SGP}(0) = 0$ and $N_{2SGP}(0, i) = 0$. The value of segment $L \otimes y$ ($y = 1, 2, \dots, W$) can be chosen from the maximum value between the value of segment $L \otimes (y - 1)$ and the other value, which is the total value of strip $L \otimes w_k$ and segment $L \otimes (y - w_k)$. Considering the remaining demand of each item, the value of type- k strip $L \otimes w_k$ can be computed by the sum of the price of each item multiplying the possible existing number of each item, which is determined with the minimum between the number of strip $L \otimes w_k$ and the remaining number of each item excluding used for segment $L \otimes (y - w_k)$.

The 2DGP can be achieved with the maximum value $F_{2SGP}(y)$, which is considered by $y = 1, 2, \dots, W$. Meanwhile, the total value of type- k strip $V_{2SGP}(k)$ is determined by $n_{2SGP}(k, L, i)$, which is considered by $k = 1, 2, \dots, 2m$, $i = 1, 2, \dots, m$. Therefore, the complexity of recursion (2) is $O(2m^2W)$. Subsequently, the complexity of the generation procedure for 2SGP is $O[2m^2(L + W)]$.

3.2. Generation Procedure for 3SHP. The generation procedure for 3SHP always includes the generation of homogenous strips, homogenous segments, and 3SHP. The generation of homogenous strips provides the candidate strips for homogenous segments. The homogenous segments are the basic element for 3SHP. Similar to the generation procedure for 2SGP, the homogenous strips are constructed

along the length direction of the used plate $L \otimes W$. Considering item rotation, $2m$ type of homogenous strips $x \otimes w_k$, including single item, are determined as follows:

$$\begin{aligned} n_{3\text{SHP}}(k, x, i) &= \min \left\{ \left\lfloor \frac{x}{l_i} \right\rfloor, r_i \right\}, \quad k = 1, 2, \dots, m, \\ n_{3\text{SHP}}(k, x, i) &= \min \left\{ \left\lfloor \frac{x}{w_i} \right\rfloor, r_i \right\}, \quad k = m+1, m+2, \dots, 2m. \end{aligned} \quad (3)$$

The homogenous segments can be obtained with a vertical combination of generated homogenous strips. The combination of homogenous strips for the segment $x \otimes W$ can be determined by solving the following recursion:

$$\begin{aligned} F_{3\text{SHP}}(x, y) &= \max \{ F_{3\text{SHP}}(x, y-1), V_{3\text{SHP}}(k, x) \\ &\quad + F_{3\text{SHP}}(x, y-w_k) | w_k \leq y \}, \\ V_{3\text{SHP}}(k, x) &= v_i \min [n_{3\text{SHP}}(k, x, i), r_i - N_{3\text{SHP}}(x, y-w_k, i)]. \end{aligned} \quad (4)$$

Recursion (4) is initialized with $F_{3\text{SHP}}(x, 0) = 0$ and $N_{3\text{SHP}}(x, 0, i) = 0$. The value of segment $x \otimes y$ can be computed with the maximum value between the value of segment $x \otimes (y-1)$ and the other value, which is the total value of homogenous strip $x \otimes w_k$ and segment $x \otimes (y-w_k)$. The value of homogenous strip $x \otimes w_k$ is the total value of including type- i items, which are selected from minimum between the number of homogenous strip $x \otimes w_k$, and the remaining number of type- i item for the used segment $x \otimes (y-w_k)$.

The generation of homogenous strips can be built with formula (3). Based on all generated homogenous strips, the homogenous segments can be generated by solving recursion (4). The decision variable is $N_{3\text{SHP}}(x, y, i)$ for $x = 1, 2, \dots, L$, $y = 1, 2, \dots, W$, $i = 1, 2, \dots, m$. The complexity of recursion (2) is $O(mLW)$.

The 3SHP can be obtained with a horizontal combination of at least two segments. Although 3SHP including more segments can improve material utilization, the cutting complexity is also increased. We present the simplest 3SHP, including two segments, which can be extended to 3SHP, including more segments. The value of 3SHP is the total value of segment $x \otimes W$ and segment $(L-x) \otimes W$, shown in formula (5). The dividing position of plate $L \otimes W$ can be determined by selecting the maximum value of $T_{3\text{SHP}}(x)$:

$$T_{3\text{SHP}}(x) = \max \{ F_{3\text{SHP}}(x, W) + F_{3\text{SHP}}(L-x, W) \}, L \otimes w_k. \quad (5)$$

The formula is used to find the maximum value of $T_{3\text{SHP}}(x)$, $L \otimes w_k$. The complexity of formula (5) is $O(L)$. Considering that $L \ll mLW$, subsequently, the complexity of the generation procedure for 3SHP is $O(mLW)$.

3.3. Normal Sizes for 2SGP and 3SHP. The staged patterns with recursive techniques commonly consider the normal sizes, which are the possible combination sizes of all

demanded items. The variable is increased by one unit using general recursive techniques at each time, whereas only the positions of normal sizes are considered to generate staged patterns. Hence, considering normal sizes can reduce the computation time and cannot affect the solution quality. The normal sizes of strips, segments, and patterns for T-shape homogenous patterns are defined in Cui [26].

Four sets of normal sizes are shown in formula (6). For the generation procedure for 2SGP, the normal sizes from set X_1 can be considered to generate strips $x \otimes w_k$ when recursion (1) is solved. The normal sizes from set Y_1 can be considered to generate patterns $L \otimes y$ when recursion (2) is solved. For the generation procedure for 3SHP, the normal sizes from set X_2 can be considered to generate strips $x \otimes w_k$ when recursion (3) is solved. The normal patterns from set Y_2 can be considered to generate segments $x \otimes y$ when recursion (4) is solved. The normal patterns from set X_2 can be considered to generate segments $x \otimes W$ when formula (5) is solved:

$$\begin{aligned} X_1 &= \left\{ x | x = \sum_{k=1}^{2m} e_k l_k \leq L, e_k \in \left\{ 0, 1, \dots, \left\lfloor \frac{L}{l_k} \right\rfloor \right\} \right\}, \\ Y_1 &= \left\{ y | y = \sum_{k=1}^{2m} e_k w_k \leq W, e_k \in \left\{ 0, \dots, \left\lfloor \frac{W}{w_k} \right\rfloor \right\} \right\}, \\ X_2 &= \left\{ x | x = e_k l_k, e_k \in \left\{ 0, 1, \dots, \left\lfloor \frac{L}{l_k} \right\rfloor \right\} \right\}, \\ Y_2 &= \left\{ y | y = e_k w_k, e_k \in \left\{ 0, 1, \dots, \left\lfloor \frac{W}{w_k} \right\rfloor \right\} \right\}. \end{aligned} \quad (6)$$

3.4. Correction Value for Each Item. The recursive techniques are used repeatedly during the generation procedure for 2SGP and 3SHP. The solution may include good patterns in the early recursive stage and bad patterns in the later recursive stage because of the greedy nature. To avoid trapping into local optimization, the correction value is formula (7), where the new value of each item can be changed according to those items in the current pattern. Let n_i be the number of type- i item in the current pattern and u the material utilization of the current pattern:

$$\begin{aligned} v_i &= g_1 v_i + g_2 \frac{(l_i w_i)}{u}, \quad i = 1, 2, \dots, m, \\ g_1 &= 1 - g_2, g_2 = \frac{\alpha n_i}{d_i + r_i}, \\ u &= \frac{\sum_{i=1}^m (n_i l_i w_i)}{LW}. \end{aligned} \quad (7)$$

The new value of each item is determined with a weighted combination of the existing value v_i and the other part $(l_i w_i)^\beta / u$, which is called overproportional material consumption norm in [27]. Two weighted parameters g_1 and g_2 are computed by each computational formula. Parameter α is selected from interval $[0.6, 0.9]$. Parameter β is

slightly bigger than 1 in [16]. The value of the items non-existent in the current pattern will not be changed because $g_2 = 0$ and $g_1 = 1$ when $n_i = 0$.

For overproportional material consumption norm $(l_i w_i)^\beta / u$, the numerator $(l_i w_i)^\beta$ denotes that the value of the item is overproportional to its area. These items with a large area have a big new value. The denominator u shows that the value of these items constructing the current pattern with bad material utilization has a big change. These items with hard combination possess a big new value. Therefore, the correction value for the item in the current pattern is aimed to select those items with large area and hard combination. They are applied to generate patterns in the early recursive stage. This approach can reduce the greedy nature of recursive techniques and improve solution quality.

4. An ISHP Approach

The proposed ISHP approach tries to achieve a cutting plan with the minimum number of plates for solving the 2DRSSSCSP-GR. The flowchart of the ISHP approach is shown in Figure 5. Based on data information of the 2DRSSSCSP-GR, A 2SGP and a 3SHP are constructed with the generation procedure for 2SGP (Section 3.1) and generation procedure for 3SHP (Section 3.2) based on normal sizes for 2SGP and 3SHP (Section 3.3), respectively. The pattern with a larger value is selected as the candidate pattern. The frequency of the candidate pattern is determined according to the residual demand of each item. The information of the candidate pattern and its frequency is recorded. The residual demand of each item is updated based on the information of the candidate pattern. A new cutting plan is generated when the demand for each item is zero. Otherwise, the value of each item is corrected based on the current candidate pattern with a correction value for each item (Section 3.4). The newly generated cutting plan is used to replace the previous cutting plan when the newly generated cutting plan with the number of used plates is less than those in the previous cutting plan. Otherwise, the newly generated cutting plan is discarded. Another newly generated cutting plan is constructed repeatedly when the number of iterations is not reached. Otherwise, a cutting plan with a minimum number of used plates is achieved for solving the current 2DRSSSCSP-GR.

The following notations are used to conveniently describe the procedure of the ISHP approach.

G : number of iterations, $G = 1, 2, \dots, G_{\max}$

G_{\max} : maximum number of iterations

$N_{\text{pl_best}}$: number of used plates for the best cutting plan

$N_{\text{pl_cur}}$: number of used plates for the current cutting plan

P : set of patterns

p_j : type- j pattern

F : set of frequency of patterns

f_j : frequency of type- j pattern

The procedure of the ISHP approach mainly includes the following steps:

Step 1: initialize the relevant parameters: $G = 1$, $N_{\text{pl_best}} = +\infty$, and $v_i = l_i w_i$, $i = 1, 2, \dots, m$.

Step 2: generate a cutting plan with a combination of 2SGP and 3SHP.

Step 2.1: let $P =$, $F =$, and $r_i = d_i$, $i = 1, 2, \dots, m$

Step 2.2: for the remaining demand r_i of type- i item, use generation procedure for 2SHP and 3SHP to generate 2SHP and 3SHP based on normal sizes

Step 2.3: select one with a large value from 2SHP and 3SHP in Step 2.2, and take it as the candidate pattern p_j

Step 2.4: compute the frequency of the current candidate pattern p_j by the formula $f_j = \min\{[r_i/n_i], i = 1, 2, \dots, m\}$. Put p_j and f_j into P and F , respectively

Step 2.5: update the remaining demand by the formula $r_i = r_i - f_j n_i$, $i = 1, 2, \dots, m$

Step 2.6: if $\sum_{i=1}^m r_i > 0$, correct value of each item v_i , $i = 1, 2, \dots, m$. $j = j + 1$ and go to Step 2.2; otherwise, record the number of used plate $N_{\text{pl_cur}}$ for the current cutting plan, and go to Step 3

Step 3: if $N_{\text{pl_cur}} < N_{\text{pl_best}}$, record the current cutting plan as the best cutting plan. Otherwise, ignore this current cutting plan. The number of iterations increased by one; that is, $G = G + 1$.

Step 4: if $G > G_{\max}$, achieve the best cutting plan for the 2DRSSSCSP-GR. Otherwise, go to Step 2.

The whole steps run through G main iterations, each of which is the SHP in Step 2. Step 1 is used to initialize iterations, the number of used plates, and pattern types for the best cutting plan. The 2SGP and 3SHP are generated for constructing a cutting plan until all items demand is fulfilled in Step 2. Step 2 includes generating a 2SGP and a 3SHP, selecting the current candidate pattern, updating the items demand, and correcting the value for each item. Step 3 compares the number of used plates of the current cutting plan and the previous cutting plan. The iterations are run G_{\max} times and achieve the best cutting plan with the minimum number of used plates for solving the 2DRSSSCSP-GR.

As a new cutting plan can be generated by each iteration, and the ISHP approach is used to generate G_{\max} cutting plans. Meanwhile, the number of patterns in a cutting plan can be estimated by $H = (\sum_{i=1}^m l_i w_i d_i) / (LW)$. The complexity of generation for 2SHP and 3SHP is $O[2m^2(L + W)] + O(mLW) = O[2m^2(L + W) + mLW]$ according to Sections 3.1 and 3.2. Therefore, the complexity of the proposed ISHP approach is $O[(2m^2(L + W) + mLW)HG_{\max}]$.

5. Computational Results

This section is presented in two parts. The relevant parameters are set in the first part. The proposed ISHP approach is compared with one published algorithm with 2SGP in [16] and one published algorithm with 3SHP in [19]

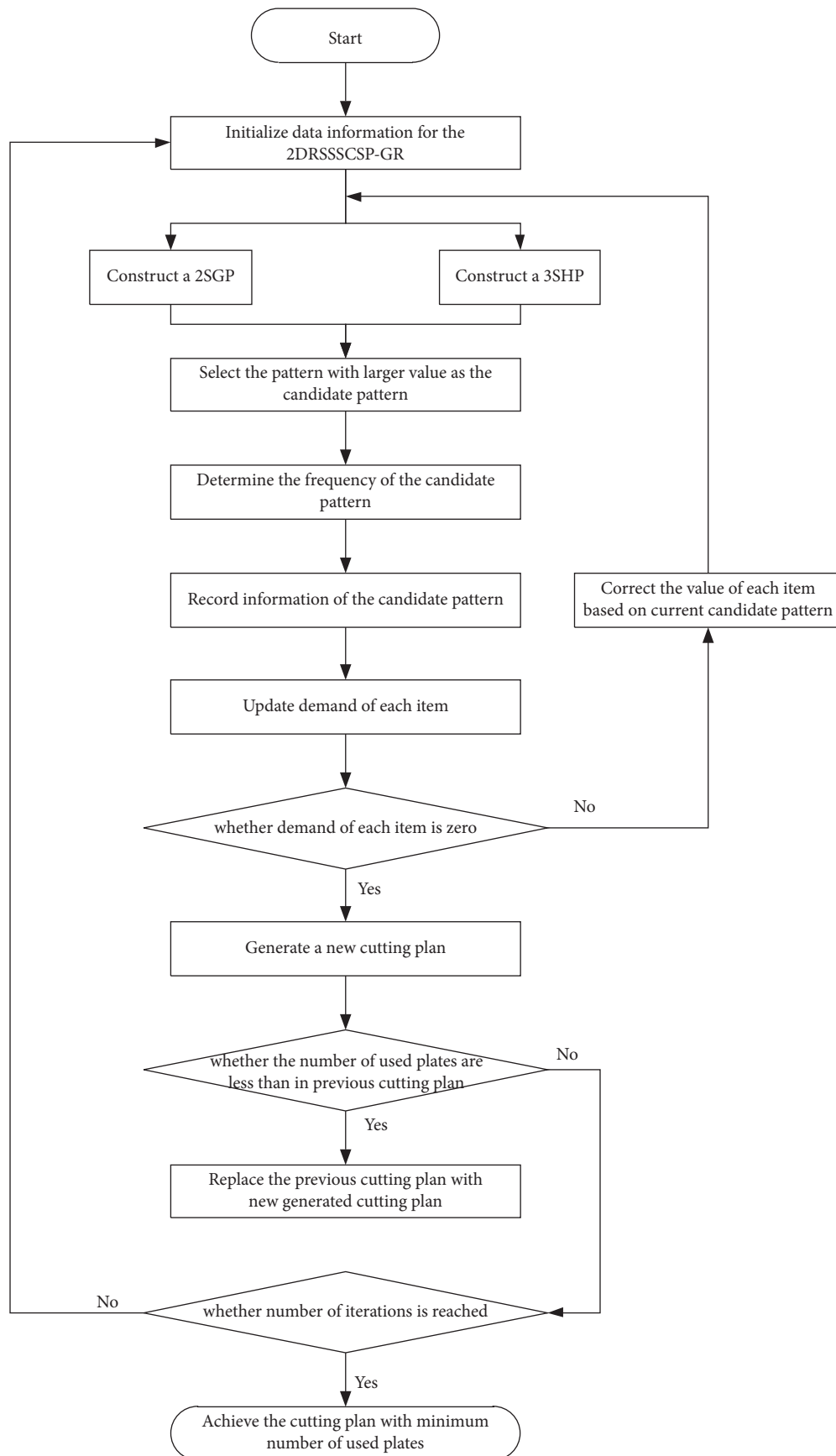


FIGURE 5: Flowchart of the ISHP approach.

in the second part. Three types of heuristic algorithms are used to deal with two sets A and B, which are obtained from Silva et al. [12]. The ISHP runs on an Acer laptop with 2.30Gz Inter Core i5 and 8 GB RAM.

5.1. Parameter Setting. The solution quality and computational time are determined by main parameters α , β , and G_{\max} . Three parameters are considered with $\alpha \in [0.6, 0.9]$, $\beta \in [1.00, 1.08]$, and $G_{\max} \in [1, 100]$. The ISHP is applied to solve instances from set A. The computational results are shown in Table 1, where N_{tot} denotes the total used plates for all instances and t represents the average time in seconds for each instance.

It can be seen from the computational results of Table 1 that (1) the two parameters α and β with the given intervals have little effect on the solution quality and computation time; (2) the solution quality tends to be stable when the maximum iterations G_{\max} is larger than 50; and (3) the computation time is very short for all instances from set B. Hence, we set the value of three parameters: $\alpha = 0.7$, $\beta = 1.02$, and $G_{\max} = 100$.

5.2. Comparing the ISHP with Two Published Algorithms. The published algorithm in [16] applied the repeated constrained CG (RCCG) to solve the 2DRSSCSP-GR with 2SGP. The other published algorithm in [19] used sequential value correction heuristic (SVCH) to solve the 2DRSSCSP-GR with 3SHP. The lower bound N_{LB} of solution for each instance is copied from [19]. Notations N_{ISHP} , N_{RCCG} , and N_{SVCH} are the number of used plates using the ISHP, RCCG, and SVCH for solving two sets A and B, respectively. Notation Δ represents that the solution is equal to the lower bound for each instance.

Three algorithms are applied to solve set A of 30 small-scale instances. The computational results for set A are displayed in Table 2. RCCG and SVCH solve 19 and 26 instances to their lower bounds, respectively. But the ISHP solves 28 instances to their lower bounds. Only the instances (CU2 and CW1) need one more plate. As one more plate is used for a staged pattern, three guillotine cuts are added for cutting complexity. Therefore, the comparison results indicate that the proposed ISHP can achieve a better solution (higher material utilization and less cutting complexity) than the other two published algorithms for solving small-scale instances.

Meanwhile, three algorithms are also used to solve the set B of 20 medium-scale instances. The computational results for set B are shown in Table 3. In the same comparison way, RCCG and SVCH solve 11 and 16 instances to their lower bounds, respectively. However, ISHP solves 18 instances to their lower bounds. Only the instances (APT39, APT47) need one more plate. In case of instance APT30, the ISHP can reach the lower bound. But the other two published algorithms cannot reach the lower bound. The data information of instance APT30 is generated in Table 4. The cutting plan figure for instance APT30 is shown in Figure 6. The comparison results also indicate that the proposed ISHP can achieve a better solution (higher material utilization and

TABLE 1: Computational results of different parameter setting.

Constants	Variables	N_{tot}	t
$\beta = 1.02, G_{\max} = 100$	$\alpha = 0.6$	252	0.7
	$\alpha = 0.7$	251	0.7
	$\alpha = 0.8$	251	0.8
	$\alpha = 0.9$	252	0.7
$\alpha = 0.7, G_{\max} = 100$	$\beta = 1.00$	252	0.6
	$\beta = 1.02$	251	0.7
	$\beta = 1.04$	252	0.7
	$\beta = 1.06$	252	0.8
$\alpha = 0.7, \beta = 1.02$	$G_{\max} = 1$	264	0.1
	$G_{\max} = 10$	258	0.1
	$G_{\max} = 30$	254	0.2
	$G_{\max} = 50$	251	0.3
	$G_{\max} = 80$	251	0.5
	$G_{\max} = 100$	251	0.7

TABLE 2: Computational results for set A.

Instances	N_{LB}	N_{ISHP}	N_{RCCG}	N_{SVCH}
A1	23	Δ	Δ	Δ
A2	12	Δ	13	Δ
A3	8	Δ	Δ	Δ
A4	5	Δ	Δ	Δ
A5	4	Δ	5	Δ
CHL1	6	Δ	7	Δ
CHL2	3	Δ	Δ	Δ
CHL5	3	Δ	4	4
CHL6	5	Δ	6	6
CHL7	6	Δ	Δ	Δ
CU1	12	Δ	Δ	Δ
CU2	14	15	15	15
CW1	9	10	10	10
CW2	12	Δ	13	Δ
CW3	16	Δ	Δ	Δ
Hchl2	6	Δ	Δ	Δ
Hchl3s	3	Δ	Δ	Δ
Hchl4s	2	Δ	3	Δ
Hchl6s	5	Δ	Δ	Δ
Hchl7s	7	Δ	Δ	Δ
Hchl8s	2	Δ	Δ	Δ
Hchl9s	10	Δ	Δ	Δ
HH	2	Δ	Δ	Δ
OF1	4	Δ	Δ	Δ
OF2	4	Δ	5	Δ
STS2	12	Δ	Δ	Δ
STS4	5	Δ	Δ	Δ
W	24	Δ	Δ	Δ
2	2	Δ	Δ	Δ
3	23	Δ	24	Δ

less cutting complexity) than the other two published algorithms for solving medium-scale instances.

The data information of instance APT30 is obtained as follows.

The used plates have the same size $927 \otimes 152$. The number of items types is 38, and the total number of all items is 192. The size and demand of each item are randomly selected from $l_i \in [57, 371]$, $w_i \in [7, 47]$, and $d_i \in [4, 6]$. The detailed data information of all demand items is shown in Table 4.

TABLE 3: Computational results for set B .

Instances	N_{LB}	N_{ISHP}	N_{RCCG}	N_{SVCH}
APT30	8	Δ	9	9
APT31	14	Δ	Δ	Δ
APT32	12	Δ	13	Δ
APT33	12	Δ	13	Δ
APT34	6	Δ	Δ	Δ
APT35	8	Δ	Δ	Δ
APT36	8	Δ	Δ	Δ
APT37	11	Δ	12	Δ
APT38	10	Δ	11	Δ
APT39	11	12	12	12
APT40	15	Δ	16	Δ
APT41	12	Δ	Δ	Δ
APT42	15	Δ	Δ	Δ
APT43	12	Δ	13	13
APT44	9	Δ	Δ	Δ
APT45	8	Δ	Δ	Δ
APT46	11	Δ	Δ	Δ
APT47	12	13	13	13
APT48	8	Δ	Δ	Δ
APT49	5	Δ	Δ	Δ

TABLE 4: Detailed data information of all demand items.

ID	l_i	w_i	d_i
1	273	44	6
2	302	7	5
3	77	23	5
4	304	21	5
5	216	37	5
6	67	38	6
7	124	13	5
8	189	26	4
9	318	24	6
10	320	36	4
11	86	35	6
12	322	36	5
13	318	17	5
14	189	33	4
15	269	32	4
16	81	12	5
17	169	10	6
18	305	26	6
19	266	45	6
20	319	18	5
21	223	29	5
22	48	15	6
23	284	36	5
24	311	16	4
25	230	26	4
26	255	34	4
27	251	42	5
28	160	45	4
29	223	28	5
30	90	11	5
31	239	12	5
32	47	16	6
33	124	40	6
34	51	16	6
35	67	39	4
36	276	15	4
37	235	44	6
38	136	20	5

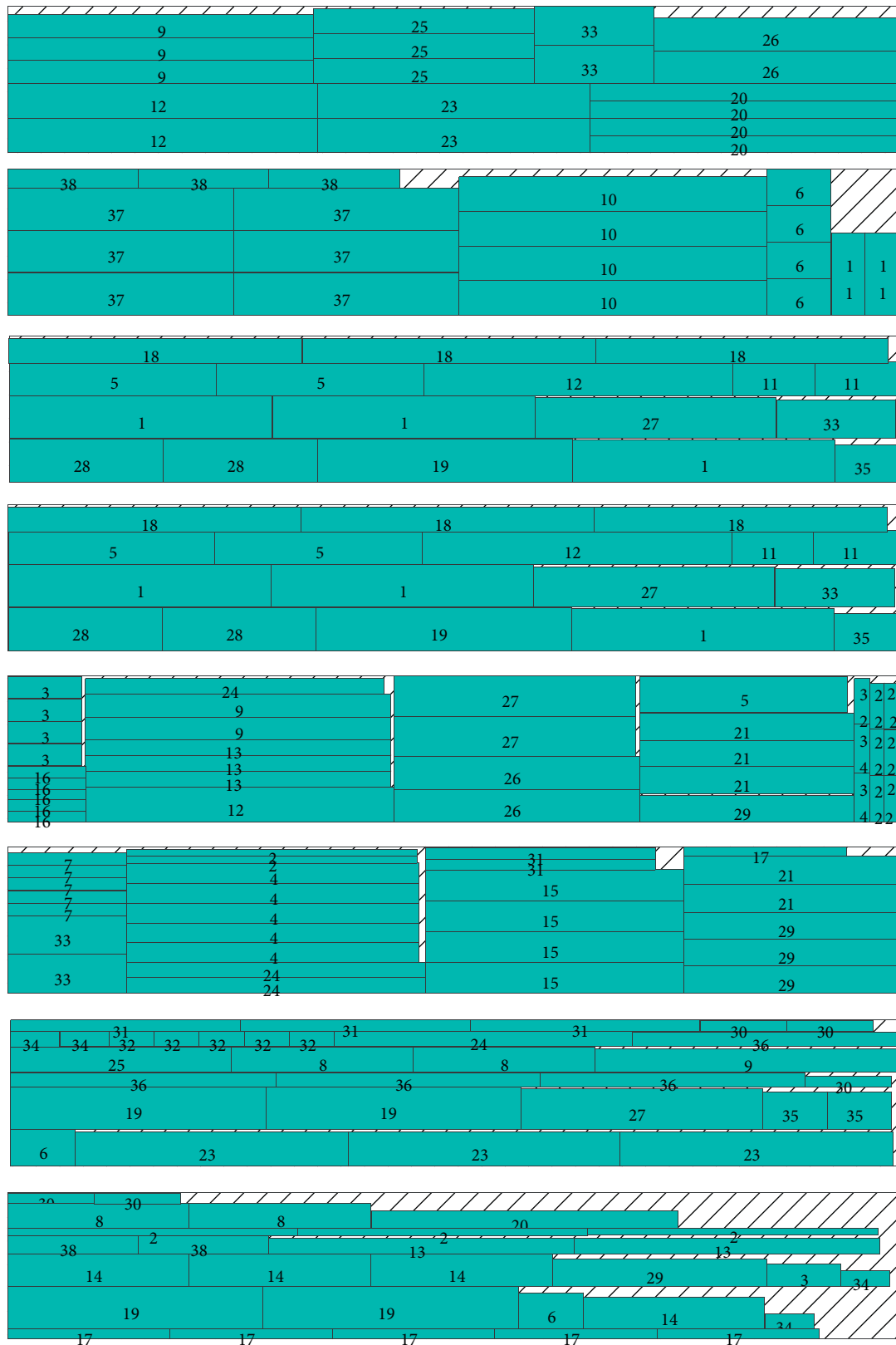


FIGURE 6: A cutting plan figure for instance APT30 solved by the ISHP approach.

TABLE 5: Computational results of three approaches.

ID	$U_{\text{ISHP}}(\%)$	$U_{\text{RCCG}}(\%)$	$U_{\text{SVCH}}(\%)$	N_{ISHP}	N_{RCCG}	N_{SVCH}
Set A	92.79	89.58	92.07	53	55	53
Set B	93.33	89.66	92.28	143	150	145

The cutting plan figure for instance APT30 solved by the ISHP approach is shown in Figure 6, where the number indexes from 1 to 38 in the filled blue box represents “ID” of each item in Table 4. The filled slash box represents useable leftovers.

5.3. Influence of Combining 2SGP and 3SHP. This section analyses the influence of combining 2SGP and 3SHP for the optimization objectives, including material utilization and cutting complexity. For generating a staged pattern, 2SGP and 3SHP are used in our proposed ISHP approach, simultaneously. The only 2SGP is applied by the RCCG presented in [16]. Only 3SHP is applied by the SVCH in [19]. Three methods are all used to solve all instances in set A and set B. The computational results of the three approaches are shown in Table 5.

In Table 5, let U and N denote the average material utilization and the average number of cuts, respectively. It is noted that the average number of cuts is upper rounding. For example, Notations $U_{\text{ISHP}}(\%)$ and N_{ISHP} represent the average material utilization and the average number of cuts for all instances in set A solved by the ISHP approach. Other parameters are the corresponding meaning.

From a material utilization perspective, our proposed ISHP approach can get higher average material utilization than the RCCG and SVCH approaches for solving instances in both set A and set B. This verifies that our proposed ISHP approach is more effective in improving material utilization.

From a cutting complexity perspective, our proposed ISHP approach can obtain the same average number of cuts as the SVCH approach, less than the RCCG approach for solving instances in set A. Or proposed ISHP approach can achieve less than the RCCG and SVCH approaches for solving instances in set B. This indicates that our proposed ISHP approach also can reduce cutting complexity without affecting material utilization.

Therefore, our proposed ISHP approach considering the combination of 2SGP and 3SHP can get better solution quality than two other published approaches considering only 2SGP or 3SHP.

6. Conclusions

The ISHP approach in this paper is proposed to solve the 2DRSSSCSP-GR with the combination of 2SGP and 3SHP. The 2SGP and 3SHP are as candidate staged patterns. The SHP algorithm is used to select a candidate staged pattern with a larger value. A new cutting plan is generated until all items demand is fulfilled accurately. The ISHP approach tries to generate many cutting plans by the iterative algorithm. The cutting plan with the minimum number of used plates is

selected as the best cutting plan for the current 2DRSSSCSP-GR. The computational results also indicate that the ISHP approach is more effective in material utilization and cutting complexity than the published algorithms with staged patterns.

The common staged patterns are welcomed in the cutting process of the manufacturing industry, such as the production of steel products, paper, wood, glass, and other industries. 2SGP and 3SHP can be easily obtained with recursive techniques. The ISHP with the combination of 2SGP and 3SHP can get the most instances to their lower bounds, which verifies that combining 2SGP and 3SHP can balance material utilization and cutting complexity while solving the 2DRSSSCSP-GR.

The ISHP in this paper considers the 2DRSSSCSP-GR with two contradictory objectives of material utilization and cutting complexity. Future research may consider other factors, such as setup cost or useable leftovers.

Abbreviations

2SGP:	Two-staged general patterns
3SHP:	Three-staged homogenous patterns
2DRCSPP:	Two-dimensional rectangular cutting stock problem
2DRSSSCSP-GR:	Two-dimensional rectangular single stock size cutting stock problem with guillotine cuts and item rotated by 90°
ISHP:	Iterative sequential heuristic procedure
$L \otimes W$:	Plates of length and width
$l_i \otimes w_i \otimes v_i \otimes d_i$:	Type- i item of length, width, value, and demand
K :	Number of types of item allowed to be rotated by 90° ; that is $K = 2m$
l_k :	Length of type- k item, $l_k = l_i$ when $k = 1, 2, \dots, m$; $l_k = w_i$ when $k = m + 1, m + 2, \dots, 2m$, $i = 1, 2, \dots, m$
w_k :	Width of type- k item, $w_k = w_i$ when $k = 1, 2, \dots, m$; $w_k = l_i$ when $k = m + 1, m + 2, \dots, 2m$, $i = 1, 2, \dots, m$
r_i :	Remaining demand of type- i item
$f_{2\text{SGP}}(k, x)$:	Value of type- k strip $x \otimes w_k$ for 2SGP, $k = 1, 2, \dots, 2m$
$n_{2\text{SGP}}(k, x, i)$:	Number of type- i item in type- k strip $x \otimes w_k$ for 2SGP $i = 1, 2, \dots, m$
$F_{2\text{SGP}}(y)$:	Value of segment $L \otimes y$ for 2SGP $y = 1, 2, \dots, W$
$V_{2\text{SGP}}(k)$:	Total value of possible items in type- k strip $L \otimes w_k$ for 2SGP
$N_{2\text{SGP}}(y, i)$:	number of type- i item in segment $L \otimes y$ for 2SGP $y = 1, 2, \dots, W$
$F_{3\text{SHP}}(x, y)$:	Value of segment $x \otimes y$ for 3SHP
$V_{3\text{SHP}}(k, x)$:	Total value of possible existing items in type- k strip $x \otimes w_k$ for 3SHP
$n_{3\text{SHP}}(k, x, i)$:	Number of only including type- i item in type- k strip $x \otimes w_k$ for 3SHP, $i = 1, 2, \dots, m$
$N_{3\text{SHP}}(x, y, i)$:	Number of type- i item in segment $x \otimes y$ for 3SHP

$T_{3\text{SHP}}(x)$:	Value of 3SHP including segment $x \otimes W$ and segment $(L - x) \otimes W$ at the dividing position x
α and β :	Two adjustable parameters for correction value for each item
G :	Number of iterations $G = 1, 2, \dots, G_{\max}$
G_{\max} :	Maximum number of iterations
$N_{\text{pl_best}}$:	Number of used plates for the best cutting plan
$N_{\text{pl_cur}}$:	Number of used plates for the current cutting plan
P :	Set of patterns
p_j :	Type- j pattern
F :	Set of frequency of patterns
f_j :	Frequency of type- j pattern.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Key Research and Development Program of China (grant nos. 2018YFB0105300 and 2017YFD0700600).

References

- [1] G. Wäscher, H. Haussner, and H. Schumann, "An improved typology of cutting and packing problems," *European Journal of Operational Research*, vol. 183, no. 3, pp. 1109–1130, 2007.
- [2] A. Lodi, S. Martello, and D. Vigo, "Heuristic and meta-heuristic approaches for a class of two-dimensional bin packing problems," *INFORMS Journal on Computing*, vol. 11, no. 4, pp. 345–357, 1999.
- [3] M. Hifi and C. Roucairol, "Approximate and exact algorithms for constrained (un) weighted two-dimensional two-staged cutting stock problems," *Journal of Combinatorial Optimization*, vol. 5, no. 4, pp. 465–494, 2001.
- [4] F. Vanderbeck, "A nested decomposition approach to a three-stage, two-dimensional cutting-stock problem," *Management Science*, vol. 47, no. 6, pp. 864–879, 2001.
- [5] R. Alvarez-Valdes, A. Parajon, and J. M. Tamarit, "A computational study of LP-based heuristic algorithms for two-dimensional guillotine cutting stock problems," *Spectrum*, vol. 24, no. 2, pp. 179–192, 2002.
- [6] P. C. Gilmore and R. E. Gomory, "Multistage cutting stock problems of two and more dimensions," *Operations Research*, vol. 13, no. 13, pp. 94–120, 1965.
- [7] S. M. A. Suliman, "A sequential heuristic procedure for the two-dimensional cutting-stock problem," *International Journal of Production Economics*, vol. 99, no. 1, pp. 177–185, 2006.
- [8] G. Belov and G. Scheithauer, "A branch-and-cut-and-price algorithm for one-dimensional stock cutting and two-dimensional two-stage cutting," *European Journal of Operational Research*, vol. 171, no. 1, pp. 85–106, 2006.
- [9] G. F. Cintra, F. K. Miyazawa, Y. Wakabayashi, and E. C. Xavier, "Algorithms for two-dimensional cutting stock and strip packing problems using dynamic programming and column generation," *European Journal of Operational Research*, vol. 191, no. 1, pp. 61–85, 2008.
- [10] Y. Cui, "Heuristic and exact algorithms for generating homogenous constrained three-staged cutting patterns," *Computers & Operations Research*, vol. 35, no. 1, pp. 212–225, 2008.
- [11] M. Hifi, R. M'Hallah, and T. Saadi, "Algorithms for the constrained two-staged two-dimensional cutting problem," *INFORMS Journal on Computing*, vol. 20, no. 2, pp. 212–221, 2008.
- [12] E. Silva, F. Alvelos, and J. M. Valério de Carvalho, "An integer programming model for two- and three-stage two-dimensional cutting stock problems," *European Journal of Operational Research*, vol. 205, no. 3, pp. 699–708, 2010.
- [13] R. Macedo, C. Alves, and J. M. Valério de Carvalho, "Arc-flow model for the two-dimensional guillotine cutting stock problem," *Computers & Operations Research*, vol. 37, no. 6, pp. 991–1001, 2010.
- [14] Y. Cui and B. Huang, "Reducing the number of cuts in generating three-staged cutting patterns," *European Journal of Operational Research*, vol. 218, no. 2, pp. 358–365, 2012.
- [15] Y. Cui, "A new dynamic programming procedure for three-staged cutting patterns," *Journal of Global Optimization*, vol. 55, no. 2, pp. 349–357, 2013.
- [16] Y. Cui and Z. Zhao, "Heuristic for the rectangular two-dimensional single stock size cutting stock problem with two-staged patterns," *European Journal of Operational Research*, vol. 231, no. 2, pp. 288–298, 2013.
- [17] Y. Cui, L. Yang, Z. Zhao, T. Tang, and M. Yin, "Sequential grouping heuristic for the two-dimensional cutting stock problem with pattern reduction," *International Journal of Production Economics*, vol. 144, no. 2, pp. 432–439, 2013.
- [18] Y.-P. Cui, Y. Cui, T. Tang, and W. Hu, "Heuristic for constrained two-dimensional three-staged patterns," *Journal of the Operational Research Society*, vol. 66, no. 4, pp. 647–656, 2015.
- [19] Q. Chen, Y. Cui, and Y. Chen, "Sequential value correction heuristic for the two-dimensional cutting stock problem with three-staged homogenous patterns," *Optimization Methods and Software*, vol. 31, no. 1, pp. 68–87, 2016.
- [20] R. Andrade, E. G. Birgin, and R. Morabito, "Two-stage two-dimensional guillotine cutting stock problems with usable leftover," *International Transactions in Operational Research*, vol. 23, no. 1/2, pp. 121–145, 2016.
- [21] Y. P. Cui, Y. Yao, and D. Zhang, "Applying triple-block patterns in solving the two-dimensional bin packing problem," *Journal of the Operational Research Society*, vol. 69, 2017.
- [22] O. Nascimento, T. Queiroz, and L. Junqueira, "A MIP-CP based approach for two- and three-dimensional cutting problems with staged guillotine cuts," *Annals of Operations Research*, Article ID 214313095, 2019.
- [23] A. S. Velasco and E. Uchoa, "Improved state space relaxation for constrained two-dimensional guillotine cutting problems," *European Journal of Operational Research*, vol. 272, no. 1, pp. 106–120, 2019.
- [24] J. Long, Z. Zheng, X. Gao, P. M. Pardalos, and W. Hu, "An effective heuristic based on column generation for the two-dimensional three-stage steel plate cutting problem," *Annals of Operations Research*, vol. 289, 2020.

- [25] M. Martin, R. Morabito, and P. Munari, "A top-down cutting approach for modeling the constrained two- and three-dimensional guillotine cutting problems," *Journal of the Operational Research Society*, no. 11, pp. 1–15, 2020.
- [26] Y. Cui, "An exact algorithm for generating homogenous T-shape cutting patterns," *Computers & Operations Research*, vol. 34, no. 4, pp. 1107–1120, 2007.
- [27] G. Belov and G. Scheithauer, "Setup and open-stacks minimization in one-dimensional stock cutting," *INFORMS Journal on Computing*, vol. 19, no. 1, pp. 27–35, 2007.