

# Heuristic Algorithm for 2D Cutting Stock Problem

Georgi Evtimov and Stefka Fidanova<sup>(✉)</sup>

Institute of Information and Communication Technologies,  
Bulgarian Academy of Sciences, Acad. G. Bonchev Str., bl. 25A,  
1113 Sofia, Bulgaria  
gevtimov@abv.bg, stefka@parallel.bas.bg

**Abstract.** Every day optimization problems arise in our life and industry. Many of them require huge amount of calculations and need special type of algorithms to be solved. An important industrial problem is cutting stock problem (CSP). Cutting with less possible waste is significant in some industries. The aim of this work is to cut 2D items from rectangular stock, minimizing the waste. Even the simplified version of the problem, when the items are rectangular is NP hard. When the number of items increases, the computational time increases exponentially. It is impossible to find the optimal solution for a reasonable time. Only for very small problems the exact algorithms and traditional numerical methods can be applied. We propose a stochastic algorithm which solves the problem, when the items are irregular polygons.

## 1 Introduction

The 2D CSP is a significant problem coming from the industry. Most of the authors solve the simplified model when the items are rectangular. CSP with rectangular items appear in paper and glass industries [6], container loading, Very-large-scale integration (VLSI) design, and various scheduling tasks [8]. When the items are not rectangular the problem becomes much more complicate. This problem arises in building constructions in fasteners production, clothes production, shoes production and so on. In some applications the rotation is not possible, for example in some types of clothes, while in other it is possible and can be used for minimizing the waste.

In [6] the main topic is a two-dimensional orthogonal packing problem, where a fixed group of small rectangles must be fitted into a large rectangle so that, most of the material is used, and the unused area of the large rectangle is minimized. The algorithm combines a replacement method with a genetic algorithm. In [1] a number of heuristic algorithms for two-dimensional cutting problems (on large scales) are developed. In this study, there is a large primary stock that has to be cut into smaller pieces, so as to maximize the value of the pieces. They developed a greedy randomized adaptive search procedure. Cintra et al. [3] propose an exact algorithm based on dynamic programming. This kind of algorithms are appropriate for small problems, because the problem is NP-hard. For these problems

some method based on stochastic search is more appropriate to apply. Stochastic search do not guarantee finding optimal solution, but for the practitioners, solution which is about 5% from the optimal solution is acceptable. Dusberger and Raidl [4,5] propose two metaheuristic algorithms based on variable neighborhood search.

Above mentioned works solve the simplified problem with rectangular items. In this paper investigation on more general problem is presented, when the items are arbitrary polygons including convex and concave. Proposed algorithm is tested on real data. The performance is compared with commercial software.

The rest of the paper is organized as follows. In Sect. 2 the CSP problem is described. In Sect. 3 we propose a stochastic algorithm for solving CSP. In Sect. 4 experimental results and comparison with other algorithms are shown. In Sect. 5 some concluding remarks and directions for future work are done.

## 2 Problem Formulation

The CSP where the items are arbitrary polygons is very difficult. Therefore most of the authors prefer to concentrate their effort on simplified variant of the problem, where the items are rectangles. The rectangular variant of the problem arises in paper and glass industries. Some authors try to solve the problem with arbitrary polygons by completing the items to rectangle, but in most of the cases it is not effective [2]. Completing some polygons to rectangle, the received rectangles can have surface more than two times larger, than the original polygon. In some of the existing softwares, the ordered polygons are sorted by their surface. After, only the first half of the polygons with larger surface are completed to rectangles. It is verified if some of the polygons with smaller surface, can be positioned in some of the rectangles, without overlapping with the other polygon. This variant of the algorithm improve the achieved solutions, but is not effective too.

In our variant of CSP rectangular sheet with fixed width and unlimited length is done. The set  $E = \{i_1, i_2, \dots, i_n\}$  of ordered items are polygons, which can be convex and concave. On Fig. 1 examples of items, which are plates from metal building construction, are shown.

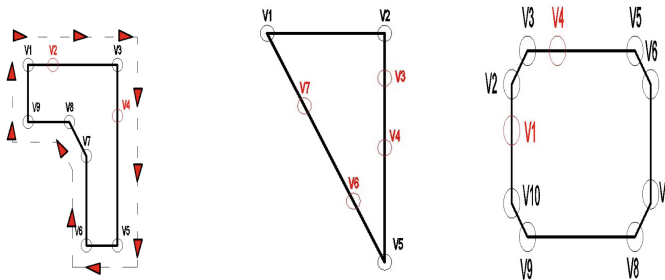


Fig. 1. The polygons which will be cut.

Every item is specified with the coordinates of his nodes and number of orders  $d_i$ , for  $i = 1, \dots, n$ . When positioning, the items can be rotated.

The objective is to find a cutting pattern  $P$  with a minimal waste. The solution is the arrangement of the items from  $E$  on the stock sheet, without overlapping. Let  $x$  is the width of the sheet and  $y$  to be cutting height of  $P$ . The cutting pattern with a minimal waste is equivalent to the cutting pattern with a minimal cutting height, because the area of all ordered items is fixed and the width sheet is fixed. So the objective function  $C(P)$  is:

$$C(P) = \min(y). \quad (1)$$

The solution can be represented by cutting sequence and coordinates of the nodes of the cutting items.

### 3 Algorithm Description

In many industries a task of cutting stock arises. Cutting with minimal waste is important for practitioners. The problem needs exponential number of calculations, but even the simplified problem, where the cutting items are rectangular continue to be NP-hard [2, 7]. Before starting the main optimization algorithm we verify if the input data are correct: if there is self-crossing; if all the points, describing the polygon, are the line; are there more than two points on the line, redundant points.

The cutting items are not necessary to be convex, they need to be correct. Each edge is linear and the nodes of the polygon describe it. The sheet, from which we will cut the item is rectangular with fixed width and infinity length. A random item is chosen from the set  $E$  of items and one of its nodes is fixed on the point with minimal height in the sheet. Let  $m$  is the number of the nodes of the current item. We will translate the item  $m$  times and will rotate it as it is shown on Fig. 2. Feasible positioning is the one for which all points inside the polygon are in the sheet of cutting. The best positioning between all feasible is one with a minimal cutting height. In our example from Fig. 2 only two of possible positioning are feasible and the cutting with minimal cutting height is shown on Fig. 3.

When the next item is positioned, again it is chosen in a random way from the polygons in the set  $E$  of items and the procedure of fixing it with in a position with a minimal height is repeated. If there are more than one possible positioning with a same minimal height, we select the next positioning between them in a random way. If it is impossible to position chosen item on this point we select the next one with minimal height. We do this till there are no more items in the set  $E$ . Figure 4 illustrates this kind of positioning. The stochastic element of the algorithm is a way for diversification of the search process.

When all items are positioned we run a procedure for decreasing the cutting height. In above described algorithm a possible nodes of positioning are the nodes of positioned items. The next procedure starts from the item with minimal positioning height, which is more than 0. We try to decrease this height with a

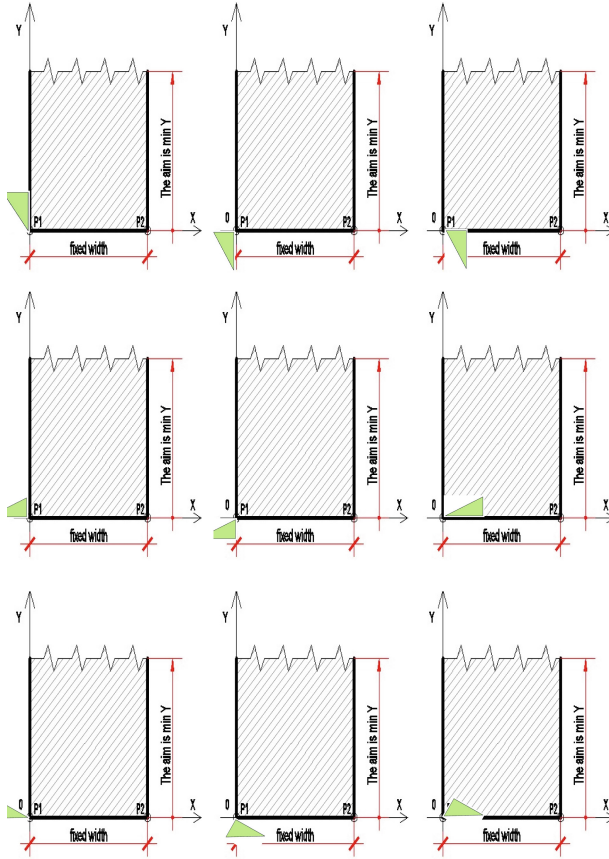


Fig. 2. Translations and rotations of a polygon

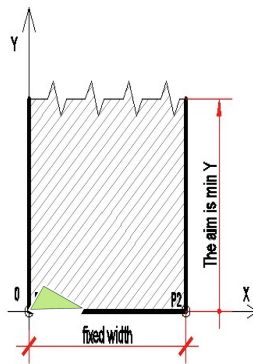


Fig. 3. Best positioning of the cutting item

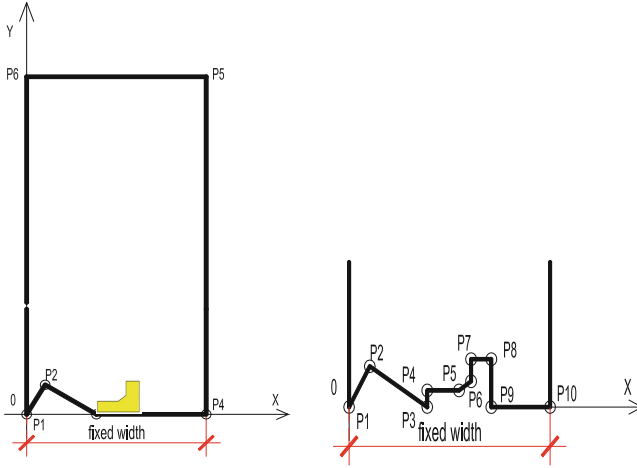


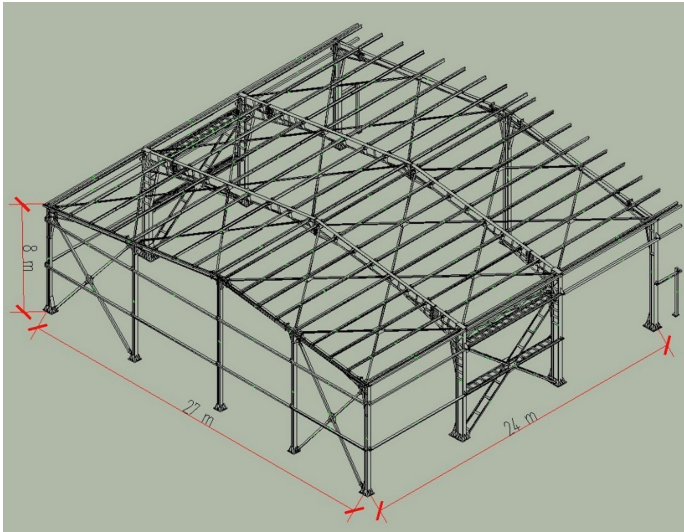
Fig. 4. Algorithm performance

fixed step. Thus an item can be positioned in a point which is on some of the edges of other items. This procedure is like adjusting, “shaking” the solution with aim to improve it. With this procedure we decrease the cutting height of all items, respectively the waste. Proposed algorithm can be applied when the bottom part of the stock sheet is not a straight line. This situation arise when cutting of some order is finished and later the producer is prepared to cut a new order. With very small changes our algorithms can solve the variant of the problem, where there are several stock sheets with fixed length.

#### 4 Test Results

The CSP is a real industrial problem and is very important if we can test it on real data. Assessment of the results require comparison with other algorithms. Although many scientists are working hard in this direction, published results are not sufficient and require further efforts. Therefore our results are compared with the results, achieved by commercial software.

Our test example consists of steel plates from real steel structure, Fig. 5. Some of them are convex and others are concave polygons. The plates are 242 different shapes and 1958 targets. The overall area of the plates is 129,053,789 mm<sup>2</sup>. The width of the cutting sheet is fixed and is equal to 1500 mm. The minimum waste is equal to find a solution with minimal cutting height and respectively minimal filling factor. The filling factor is the ratio between the sum of the area of the all plates and the cutting area ( $x \times y$ ). Processing the input data, it is taken in to account that the cut width is 5 mm. We process the input data, before running the optimization algorithm. The input data are polygons, described by points and edges between them. We verify the input data and we remove redundant



**Fig. 5.** Steel structure. Composite from steel profiles and steel plates.

points. After the processing the plates are described only with their nodes. The algorithm is run on desktop computer with 2.8 GHz CPU.

We run the proposed algorithm 30 times and we compare achieved best result with result achieved by one commercial software used by professionals. We chose a professional software, which do not complete the items to rectangular. In Table 1 we report the value of cutting height and filling factor of the commercial algorithm, algorithm without improvement procedure and the algorithm with improvement procedure. Regarding the table we observe that proposed algorithm achieves better result than the commercial one and improvement procedure improves the achieved results. Our algorithm achieves solution with smaller height and respectively less filling factor, which is equal to solution with less waste. We can conclude that proposed algorithm performs better and receives very encouraging results.

**Table 1.** Results comparison

Properties	Commercial algorithm	Proposed algorithm	Algorithm with “shaking”
Cuttet height	200,014 mm	170,273 mm	136,133
Filling factor	0.43	0.505	0.632

The Fig. 6 shows part of the solution, found by the algorithm with improvement procedure.

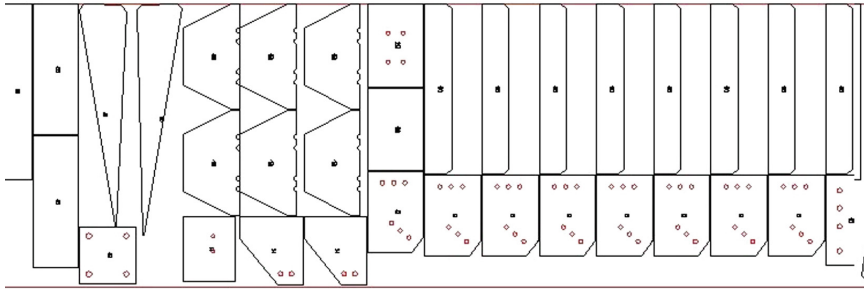


Fig. 6. Arrangement of the plates.

## 5 Conclusion

In this paper we propose a stochastic algorithm to solve cutting stock problem where the cutting items are convex and concave polygons. The algorithm can be divided in to two main parts, feasible solution construction and solution improvement. The algorithm is applied on real data from steal structure. We compare the results achieved by our algorithm with the results achieved by one commercial software. We show that our algorithm performs better than other. For a future work we plan to propose an heuristic algorithm for 2D cutting stock problem.

**Acknowledgment.** Work presented here is partially supported by the Bulgarian National Scientific Fund under Grants DFNI I02/20 “Efficient Parallel Algorithms for Large Scale Computational Problems” and DFNI DN 02/10 “New Instruments for Data Mining and their Modeling”.

## References

1. Alvarez-Valdes, R., Parajon, A., Tamarit, J.M.: A computational study of heuristic algorithms for two-dimensional cutting stock problems. In: 4th Metaheuristics International Conference (MIC 2001), pp. 16–20 (2001)
2. Alvarez-Valdes, R., Parreno, F., Tamarit, J.M.: A Tabu Search algorithm for two dimensional non-guillotine cutting problems. *Eur. J. Oper. Res.* **183**(3), 1167–1182 (2007)
3. Cintra, G., Miyazawa, F., Wakabayashi, Y., Xavier, E.: Algorithms for two-dimensional cutting stock and strip packing problems using dynamic programming and column generation. *Eur. J. Oper. Res.* **191**, 61–85 (2008)
4. Dusberger, F., Raidl, G.R.: A variable neighborhood search using very large neighborhood structures for the 3-staged 2-dimensional cutting stock problem. In: Blesa, M.J., Blum, C., Voß, S. (eds.) HM 2014. LNCS, vol. 8457, pp. 85–99. Springer, Cham (2014). [https://doi.org/10.1007/978-3-319-07644-7\\_7](https://doi.org/10.1007/978-3-319-07644-7_7)
5. Dusberger, F., Raidl, G.R.: Solving the 3-staged 2-dimensional cutting stock problem by dynamic programming and variable neighborhood search. *Electron. Notes Discret. Math.* **47**, 133–140 (2015)

6. Gonçalves, J.F.: A hybrid genetic algorithm-heuristic for a two-dimensional orthogonal packing problem. *Eur. J. Oper. Res.* **183**(3), 1212–1229 (2007)
7. Parmar, K., Prajapati, H., Dabhi, V.: Cutting stock problem: a survey of evolutionary computing based solution. In: *Proceedings of Green Computing Communication and Electrical Engineering* (2014). <https://doi.org/10.1109/ICGCCEE.2014.6921411>
8. Lodi, A., Martello, S., Vigo, D.: Recent advances on two-dimensional bin packing problems. *Discret. Appl. Math.* **123**, 379–396 (2002)