TWMS J. App. Eng. Math. V.9, N.3, 2019, pp. 473-484

ONE-DIMENSIONAL CUTTING STOCK PROBLEM WITH DIVISIBLE ITEMS: A CASE STUDY IN STEEL INDUSTRY

D. TANIR[1], O. UGURLU[2], A. GULER[3], U. NURIYEV[2], §

Abstract. This paper considers the one-dimensional cutting stock problem (1D-CSP) with divisible items, which arises in the steel industries. While planning the steel cutting operations, each item can be divided into smaller pieces, then they can be recombined by welding. The objective is to minimize both the trim loss and the number of the welds. The problem can be seen as a natural generalization of the cutting stock problem (CSP) with skiving option [1] where recombining operation has a cost. In this paper, a mathematical model for the problem is given and a dynamic programming based heuristic algorithm is proposed in accordance with the company needs. Furthermore, a software, which is based on the proposed heuristic algorithm, is developed to use in MKA Company, and its performance is analyzed by solving real-life problems in the steel industry. The computational experiments show the efficiency of the proposed algorithm.

Keywords: Production; Cutting Stock Problems; Skiving option; Heuristics; Steel Industry.

AMS Subject Classification: 90C10, 90C59, 90C27

1. Introduction

The cutting stock problem (CSP) has many applications in the production planning of many industries such as the metallurgy, plastics, paper, glass, furniture and textiles. In general, cutting stock problems are based on cutting unlimited large pieces of length c into a set of smaller items with demand v_i and length w_i while optimizing a certain objective function. The objective function can minimize waste, maximize profit, minimize cost, minimize the number of items used, etc [2].

[1] Department of Management and Information Systems, Faculty of Economics and Administrative Sciences, Kafkas University, Kars, Turkey.
e-mail: tanirdeniz35@gmail.com; ORCID: https://orcid.org/0000-0001-6593-6625.
[2] Department of Mathematics, Faculty of Science, Ege University, Izmir, Turkey
e-mail: onurugurlu@mail.ege.edu.tr; ORCID: https://orcid.org/0000-0003-2743-5939.
e-mail: urfat.nuriyev@ege.edu.tr; ORCID: https://orcid.org/0000-0002-3337-5859.
[3] Department of Computer Technologies, Vocational School, Yasar University, Izmir, Turkey
e-mail: asli.guler@yasar.edu.tr; ORCID: https://orcid.org/0000-0003-2633-5792.
§ Manuscript received: July 29, 2017; accepted: January 15, 2018.
TWMS Journal of Applied and Engineering Mathematics, Vol.9, No.3
The first and second authors gratefully acknowledge the support of TUBITAK (The Scientific and Technological Research Council of Turkey) 2211 program.

The CSP has many practical applications in real-life problems and is easy to formulate. However, these problems are difficult to solve, since they are in NP-hard [3]. Therefore, it is important to solve these problems efficiently so that the production expenses are as low as possible. As a natural counterpart of the CSP, Skiving Stock Problem (SSP) was proposed by Zak [4]. SSP aims to find the maximum number of products with minimum length c that can be constructed by connecting given smaller item lengthw_i with availability v_i. Zak [4] showed that the SSP and the CSP are closely related not only mathematically but often in real-life applications.

The first known formulation of CSP was presented in 1939 by Kantorovich [6]. The most important advance in solving the 1D-CSP was the inventive work of Gilmore and Gomory [7], where they proposed delayed pattern generation method to solve the problem by using linear programming. Dyckhoff [8] classifies the solutions of CSP into two groups: pattern-oriented and item-oriented. Moreover, he classifies cutting problems by using four characteristics: dimensionality, kind of assignment, assortment of large objects, assortment of small objects.

Waescher and Gau [9] concluded that optimal integer solutions could be obtained in most cases. Gradisar et. al. [10] presented sequential heuristic procedure to optimize the 1D-CSP when all stock lengths are different. Furthermore, they proposed a hybrid approach which combines the pattern-oriented LP method and item-oriented sequential heuristic procedure [11]. Vance et.al. [12], Vance [13], Valerio de Carvalho [14] and Vanderbeck [15,16] presented some attempts at combining column generation and branch-and-bound. They were able to obtain exact solutions for quite large instances of CSPs.

Scheithauer et.al. [17] presented a cutting plane algorithm to solve the 1D-CSP exactly. Mukhacheva et.al. [18] proposed a modified branch-and bound method for 1D-CSP. Belov and Scheithauer [19] proposed an approach combining a cutting plane algorithm with column generation for the 1D-CSP with multiple stock lengths. Umetani et.al. [20] considered that the number of different cutting patterns in the 1D-CSP is limited within a given bound. Then, they proposed an approach based on metaheuristics and incorporates an adaptive pattern generation method. Johnston and Sadinlija [21] generated a new model that resolves the non-linearity in the 1D-CSP, between pattern variables and pattern run-lengths by a novel use of 0-1 variables. Belov and Scheithauer [22] developed a branch-and-cut-and-price algorithm for one-dimensional stock cutting and twodimensional two-stage cutting problems. They investigated a combination of the LP relaxation at each branch-and-price node which is strengthened by Chvatal-Gomory and Gomory mixed-integer cuts.

In recent years, there have also been several efforts to solve this problem by using different approaches. Dikili et. al. [23] presented a successive elimination method to solve 1D-CSP, in ship production, directly by using the cutting patterns obtained by the analytical methods at the mathematical modeling stage. Reinertsen and Vossen [24] considered the CSP when orders have due dates and proposed new optimization models and solution procedures to solve this problem. Cherri et. al. [25] studied the cutting stock problem with usable leftovers and modified well-known heuristics in the literature to solve this problem. Furthermore, Cherri et. al. [26] assumed that the available retails in stock have priority in-use during the cutting process and developed their heuristics considering these priorities. Berberler and Nuriyev [27] considered the subset-sum problem as a sub-problem to solve the 1D-CSP and proposed a dynamic programming-based heuristic. Mobasher and Ekici [28] studied a more general case of the classical CSP, called cutting stock problem with setup cost in which the objective is to minimize the total production cost including material and setup costs. They developed a mixed integer linear model and proposed an algorithm for a special case of the problem.

The CSP with skiving option, where cutting of large pieces into small ones is complemented by joining small parts to form large parts, was studied first by Johnson et. al. [29]. They proposed solution approaches for the problem arising in the paper industry. Arbib et. al. [30] combined cutting and skiving problem for the production of gear belts. Arbib and Marinelli [31] proposed an integer linear programming model for the CSP with skiving option .

In this paper, we consider the one-dimensional cutting stock problem with divisible items, which is a natural generalization of the cutting stock problem with skiving option. In the problem, each item (demand) can be divided into smaller pieces, then they can be recombined by means of welding. The objective is to minimize both the trim loss and the number of the welds.

The remainder of the paper is organized as follows. In Section 2, we provide a formal definition of the classical one-dimensional cutting stock problem. We discuss the related studies in the literature in Section 3. We give the definition of the problem in Section 4 and present the new mathematical model in Section 5. In Section 6, a dynamic programming-based heuristic algorithm for the problem is presented. In Section 7, the performance of the proposed algorithm is investigated on real life instances. The conclusion is provided in Section 8.

2. The Classical 1-D Cutting Stock Problem

The one-dimensional cutting stock problem (1D-CSP) is a well-known NP-hard problem [3] that occurs during manufacturing processes in many industries such as steel industry, clothing industry and aluminum industry, and has gained much attention from all over the world in recent years.

The 1D-CSP is a linear integer programming problem with one decision variable for each possible pattern. If the number of order widths is small, then the number of patterns may be small enough so that the problem can be solved using a standard algorithm. However, there may be an exponential number of patterns if the number of order widths is large. In these cases, an alternative solution approach is needed.

The classical 1D-CSP is the problem of cutting standard-sized pieces of stock material into pieces of specified sizes while minimizing the trim loss. In the standard definition of the problem the following data is used:

- c stock lengths.
- w_i order length of item i, $i = 1, ..., n$.
- v_i demand number of item i.
- x_{ij} number of item i having been cut from stock j.
- y_j indicates whether the stock j is used in the cutting plan ($y_j = 1$ if stock j is used in the cutting plan), $j = 1, ..., m$.
- t_j indicates the length of the leftover of stock j.

The objective is to minimize trim loss (wastage). Using the notation above, the classical 1D-CSP can be formulated as follows [5]:

$$min \sum_{i=1}^{m} t_j \tag{1}$$

Subject to

$$\sum_{i=1}^{n} x_{ij} w_i + t_j = c.y_j,\ j = 1, ..., m. \tag{2}$$

$$\sum_{i=1}^{n} x_{ij} = v_i,\ j = 1, ..., m. \tag{3}$$

$$x_{ij} \geq 0 \text{ integer } \ i = 1, ..., n;\ j = 1, ..., m. \tag{4}$$

$$t_j \geq 0 \text{ integer } \ j = 1, ..., m. \tag{5}$$

$$y_j \in \{0, 1\}. \tag{6}$$

In the model, constraint (2) calculates the waste of each stock in the cutting process, constraint (3) guarantees that all demands for each item are met. Finally, the objective function minimizes the total trim loss. An example for the 1D-CSP is illustrated in Figure 1.

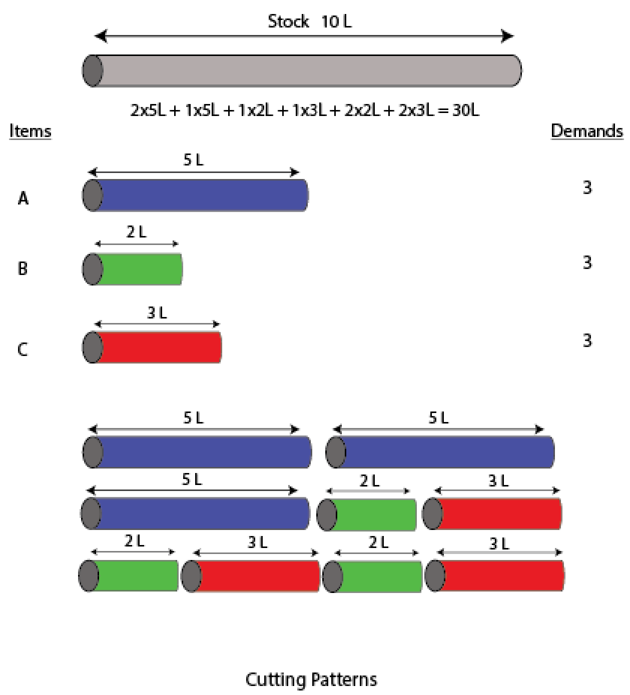

FIGURE 1. Cutting patterns for classical 1D-CSP with one stock type.

3. Problem Statement

In the steel industry, smaller-sized items are usually supplied from a standard length of 6000 mm or 12000 mm. The main goal of the companies is to determine the optimal cutting plan for minimizing the trim loss.

Since minimizing trim-loss is a vital matter in the steel industry, some companies have tried to divide items into two smaller pieces and recombined them to minimize the trim loss. When they encounter a large leftover from a cutting pattern, they try to use the leftover as a small piece of an item (demand). Then, they try to cut the residual part of the item in another cutting pattern, and they recombine the pieces by the help of welding. To be able to perform welding, these pieces must be greater than a pre-set upper bound (the bound for welding is 1000 mm in this study). Moreover, only one piece can be used for a part of an undivided item in a single cutting pattern, but the residual parts of the divided items can be used if required. (Note that this constraint is mainly valid for application purposes.) However, the welding operation is an extra cost for the companies. Thus, unlike to classical CSP, the problem has two objectives: The primary objective is to minimize the trim loss, and the secondary objective is to minimize the number of welds.

Furthermore, the leftovers obtained after cutting steels in stock can be stored for future use, if they are longer than a pre-set upper bound (the bound is 1000 mm in this study). Thus, the companies have tried to seek a solution to the cutting stock problem that will not only decrease trim loss in a period, but also improve the total results over the whole time-span. However, dealing with usable leftovers can cause excessive work-load for companies. A good solution to this problem must consider the amount of the trim loss, the number of the recombinations and the quantity of the usable leftovers.

MKA is a software company which was founded in May 2005 and the aim of the company is to be specialist in engineering software [32]. This study was motivated by MKA to develop an efficient algorithm for the problem. In section 6, a heuristic algorithm which takes all constraints of the company into consideration is described.

The problem of optimal steel manufacturing in the company can be modeled as a 1D-CSP using following nonlinear integer programming. An example for the the problem is given in Figure 2.

4. The Formulation of the Proposed Problem

In this section, a nonlinear integer programming formulation is presented for the described generalization of CSP with skiving option. In the proposed problem, we are given an unlimited number of stocks (let m be the number of stocks) of length c and n different types of items. To quantify the upper bounds for usable leftovers and welding, we use the parameters β and θ, defined by the user (decision maker).

- β : upper bound for trim loss.
- θ : upper bound for welding.

Since we have two main factors for total-cost minimization, the following cost coefficients for trim loss and the number of the welds are used.

- γ : the cost of 1 mm trim loss.
- δ : the cost of one welding operation.

β, θ γ and δ can be set according to the companies requirement. The problem can be formulated as follows:

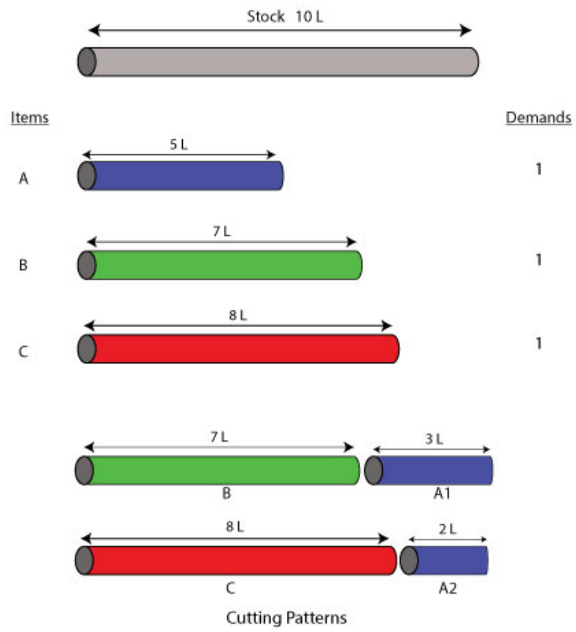

FIGURE 2. Cutting patterns 1D-CSP with divisible items.

- z_{ij}^k indicates whether k^{th} piece of item i is divided in stock j ($z_{ij}^k = 1$, if the k^{th} piece of item i is divided in stock j).
- α_{ij}^k indicates the length of the residual part of the k^{th} divided piece of item i in stock j.
- b_{ij}^k indicates whether the residual part of the k^{th} divided piece of item i is used in stock length j ($b_{ij}^k = 1$, if the residual part of the k^{th} divided piece of item i is used in stock length j).
- u_j indicates whether the leftover of stock j is counted as trim loss ($u_j = 0$, if the lenght of the leftover of stock j is greater than β and it is not counted as trim loss).

$$min(\gamma \sum_{i=1}^{m} t_j u_j + \delta \sum_{i=1}^{n} \sum_{k=1}^{v_i} \sum_{j=1}^{m} z_{ij}^k) \tag{7}$$

s.t.

$$\sum_{i=1}^{n} x_{ij} w_i + \sum_{i=1}^{n} \sum_{k=1}^{v_i} z_{ij}^k (w_i - \alpha_{ij}^k) + \sum_{i=1}^{n} \sum_{k=1}^{v_i} \sum_{l=1, l \neq j}^{m} b_{ij}^k \alpha_{il}^k + t_j = c.y_j, \ j = 1, ..., m. \tag{8}$$

$$\sum_{j=1}^{m} x_i + \sum_{k=1}^{v_i} \sum_{j=1}^{m} z_{ij}^k = v_i, \ i = 1, ..., n. \tag{9}$$

$$\sum_{k=1}^{v_i}\sum_{j=1}^{m} z_{ij}^k = \sum_{k=1}^{v_i}\sum_{j=1}^{m} b_{ij}^k,\ i = 1, ..., n. \tag{10}$$

$$\sum_{i=1}^{n}\sum_{k=1}^{v_i} z_{ij}^k \leq 1,\ j = 1, ..., m. \tag{11}$$

$$\alpha_{ij}^k \geq \theta,\ i = 1, ..., n;\ j = 1, ..., m;\ k = 1, ..., v_i. \tag{12}$$

$$w_i - \alpha_{ij}^k \geq \theta,\ i = 1, ..., n;\ j = 1, ..., m;\ k = 1, ..., v_i. \tag{13}$$

$$v_j = \begin{cases} 0 & \text{if } t_j \geq \beta; \\ 1 & otherwise. \end{cases} \tag{14}$$

$$\alpha_{ij}^k \geq 0 \text{ integer},\ i = 1, ..., n;\ \text{j=1}, ..., m;\ \text{k=1}, ..., v_i. \tag{15}$$

$$x_{ij} \geq 0 \text{ integer},\ i = 1, ..., n;\ \text{j=1}, ..., m. \tag{16}$$

$$t_j \geq 0 \text{ integer},\ \text{j=1}, ..., m. \tag{17}$$

$$y_j \in \{0, 1\},\ z_{ij}^k \in \{0, 1\},\ b_{ij}^k \in \{0, 1\}. \tag{18}$$

Objective (7) minimizes the total cost according to γ and δ. The optimal cutting pattern for each of the stock is defined by decision variables b_{ij}^k, z_{ij}^k , α_{ij}^k, x_{ij} in constraint (8). By constraint (9), all demands for each item are met. Constraint (10) guarantees that residual parts of divided items are used. Constraint (11) controls that only one undivided item can be divided and used in a single cutting pattern. Constraint (12) and (13) control that the lengths of the pieces of a divided item are greater than the pre-set upper bound θ. Finally, constraint (14) determines whether the leftover of stocks can be used in the future.

5. The Proposed Algorithm

It is quite obvious that the proposed problem is NP-Hard in the strong sense like the classical 1D-CSP [3]. Since we are dealing with real life instances, solving those large instances with the proposed nonlinear integer programming formulation may take long running time. Therefore, we proposed a heuristic algorithm to obtain high-quality solutions in reasonable time periods. The heuristic procedure was obtained by modifying the heuristic (BBP) proposed by Berberler and Nuriyev [27]. The proposed algorithm was designed to minimize primarily the trim loss, and secondly, usable leftovers and the number of welding.

The BBP algorithm considers the subset problem as a sub-problem of 1D-CSP. At each step, the algorithm tries to find a cutting pattern via dynamic programming. The algorithm uses only one one-dimensional array A instead of two arrays with dimension nxc. To construct A, the following function F_k(s) is used;

$$F_k(s) = \max\{\sum_{i=1}^{k} w_i x_i \mid \sum_{i=1}^{k} w_i x_i \leq s, 0 \leq x_i \leq v_i, x_i \in N, i = \overline{1, k}\} \\ where(s = \overline{1, c}, k = \overline{1, n}) \tag{19}$$

Initial values of this function ($s = \overline{1, c}$, $k = 1$) are determined as follows;

$$F_1(s) = \begin{cases} (1, s/w_1) & if\ s \leq v_1 w_1\ and\ s/w_1 = \lfloor s/w_1 \rfloor \\ (0,0) & otherwise \end{cases} \tag{20}$$

The recursive formula for $k > 1$ occurs as below, (k= $\overline{2, n}$, s= $\overline{1, c}$)

$$F_k(s) = \begin{cases} F_{k-1}(s), & \text{if } F_{k-1}(s) \neq 0\ \ s < w_k\ \ or\ s > \sum_{j=1}^{k} w_j v j \\ (k, p+1) & \text{if } F_k(s - w_k) = (k, p)\ and\ p < v_k \\ (k, 1) & \text{if } s = w_k\ or\ F_k(s - w_k) \neq 0 \end{cases} \tag{21}$$

After dynamic programming is performed, if the last cell of the array is filled, this means that the stock is filled. Otherwise, the full cell with maximum number yields an optimal solution. This procedure continues until all demands are completed. For detailed information about the algorithm, the reader is referred to Berberler and Nuriyev [27]. To modify BBP according to the proposed problem, we use another Array B along with Array$A$, which both have the same size, and both of these arrays are constructed simultaneously. While array A is being constructed according to above function F_k(s), array B tries to improve solutions, which are found in array A, by finding a proper divided piece of an item at each step. If a divided piece of an item fulfills the stock when it is added to the solution, then it is marked in array B.

After A and B are constructed, the cutting pattern which has minimum cost for the company in accordance with γ and δ, is selected from the arrays. The residual parts of the divided items are considered as a new single item in next steps and they are used in the construction of A . This procedure continues until all demands are met. It is easy to see that for small δ values, the trim loss converges to zero. However, for bigger δ values, the algorithm will find a solution which has less welding operations, but a lot more trim loss. Parameters can be set in many different ways in the direction of the goals of the company.

The advantage of the proposed approach is to determine an optimal cutting pattern at each step. Furthermore, the best cutting pattern which includes a divided item is found at each step. By minimizing the trim loss, the number of welds and the quantity of usable leftovers, the proposed method can capture the ideal solution of the analytical methods. The algorithm has been developed at the request of MKA Company. Currently, MKA uses this algorithm in their software package to create cutting plans for steel manufactories. The computational experiments on real life instances demonstrate the efficiency of the algorithm.

6. Computational Experiments

To the best of our knowledge, this generalization of the 1D-CSP with skiving option has been studied for the first time. Thus, we have compared our algorithm with two classical 1D-CSP algorithms and two commercial optimization software packages to show the efficiency of both the skiving option and the proposed algorithm. Also, we have reported the results of the algorithm for difficult real-life optimization problems that were taken from MKA. Computational experiments have been carried out on a computer with Intel Core i7 4700 HQ 2.4 GHz CPU, 32GB of RAM running Windows 8.1 64-bit Edition. The parameters γ and δ have been set to 1 and 500, respectively.

In Table1, the proposed algorithm has been compared with EP and mofTS which were proposed by Liang et. al. [33] and Yang et. al. [34], respectively. Also, since the proposed algorithm has been designed for commercial use, we have also compared the algorithm with Real Cut1D [35] and Plus 1D [36] in Table 2. For comparison, we have used the dataset which was used in Liang et. al. [33].

Table 1. The comparison of the proposed algorithm with EP and mofTS.

Problems	Number of demands	The Proposed Algorithm		EP Algorithm		mofTS	
		number of the used stocks	percentage of trim loss	number of the used stocks	percentage of trim loss	number of the used stocks	percentage of trim loss
1a	20	9	2,44	9	2,44	9	2,44
2a	50	23	3,92	23	3,92	23	3,92
3a	60	15	0	15	0	15	0
4a	60	19	2,37	19	2,37	19	2,37
5a	126	51	1,29	54	7,28	53	5,29
6a	200	78	0,25	82	5,40	80	2,82
7a	200	68	1,02	69	2,53	69	2,53
8a	400	144	1,24	149	4,76	145	1,95
9a	400	150	0,80	155	4,16	151	1,47
10a	600	216	0,50	224	4,23	218	1,43

Table 2. The comparison of the proposed algorithm with Plus 1D and Real Cut 1D.

Problems	The Proposed Algorithm			Plus 1D			Real Cut 1D		
	number of the used stocks	percentage of trim loss	time	number of the used stocks	percentage of trim loss	time	number of the used stocks	percentage of trim loss	time
1a	9	2,44	0,010	9	2,44	0,013	9	2,44	0,016
2a	23	3,92	0,011	23	3,92	0,11	23	3,92	0,14
3a	15	0	0,011	16	6,66	0,256	15	0,00	0,344
4a	19	2,37	0,016	20	7,75	0,238	19	2,37	0,297
5a	51	1,29	0,252	56	10,11	0,384	53	5,29	0,485
6a	78	0,25	0,046	85	9,25	0,687	80	2,82	0,813
7a	68	1,02	0,035	71	5,49	0,874	68	1,04	1,029
8a	144	1,24	0,144	149	4,75	0,902	143	0,54	1,047
9a	150	0,80	0,160	153	2,81	0,092	150	0,80	0,141
10a	216	0,50	0,331	221	2,83	0,119	216	0,50	0,172

The Table 1 shows that when the number of demands increases, the proposed algorithm can find better solutions than EP and mofTS. Also, it can be seen in Table 2 that the proposed algorithm can find similar solutions with Real Cut 1D and it can find better solutions than Plus 1D. Since these are well-known commercial optimization software packages, the results suggest that the proposed algorithm can be used in real-life applications of 1D-CSP.

To test the performance of the proposed algorithm on real-life optimization problems, we have considered the cutting plans of a whole apartment building. The whole building consists of 1,465 items and it weighs 110,530 kg. We compared results of cutting plan with skiving option and without skiving option in Table 3 for the building. The columns with asterisk (*) demonstrate the results of cutting plans with skiving option. The algorithm with skiving option has used 247 stocks which totally weigh 114,171 kg. It has created cutting pattern with 0,98 percent-trim loss, 2,14 percent-usable leftover, and only 34 welding operations. However, the algorithm without skiving option has used 269 stocks which weigh 119,429 kg and has created cutting pattern with 1,54 percent-trim loss and 6,14 percent-usable leftover. Both of the algorithms have found the whole cutting pattern within 2 seconds. The experiment results show that the proposed algorithm with skiving option reduces both the percentage of the trim loss and the percentage of usable leftovers significantly with an acceptable number of welds for the considered building.

Table 3. The results of the cutting plans for the considered building

	number of total items	total weight of all demands	number of the used stocks	number of the used stocks*	weight of trim loss	weight of trim loss*	weight of usable leftovers	weight of usable leftovers*	number of welds
pset1	195	33.435	36	34	168	131	2.273	310	6
pset2	143	24.451	54	53	371	311	595	182	8
pset3	24	19.526	18	17	845	376	708	0	11
pset4	216	6.889	25	25	1	1	153	153	0
pset5	76	6.664	19	16	95	71	1.324	69	5
pset6	88	4.645	10	10	69	69	337	337	0
pset7	92	4.424	26	26	54	21	99	132	2
pset8	37	2.242	8	8	7	7	118	118	1
pset9	130	2.217	11	10	4	1	249	36	0
pset10	7	1.808	4	4	59	59	416	416	0
pset11	68	1.700	5	5	7	7	126	126	0
pset12	103	1.472	13	12	3	9	143	12	1
pset13	72	556	18	18	16	16	11	11	0
pset14	114	192	3	3	1	1	74	74	0
pset15	1	171	1	1	0	0	131	131	0
pset16	67	86	2	2	0	0	38	38	0
pset17	17	33	1	1	0	0	47	47	0
pset18	10	14	1	1	0	0	123	123	0
pset19	5	5	1	1	0	0	47	47	0

7. Conclusions

In this paper, we have considered a natural generalization of cutting stock problem with skiving option, since the recombination has a cost in the proposed problem. The problem seems to be very promising from the practical point of view in many different areas. The main aim of the considered problem is to minimize both total trim loss and the number of welds in the steel industry. We have first developed a nonlinear integer formulation for this NP-hard problem. A dynamic programming based heuristic has also been designed to fulfil the companies needs. Moreover, a computer program which is based on the proposed algorithm has been developed to use in MKA Company. The computational experiments have shown that the algorithm performs well on real life instances.

References

[1] Johnson, M. P., Rennick, C. and Zak, E., (1997), A new model for complete solutions to one-dimensional stock problems, Siam Review, 39(3), pp. 472-483.

[2] Kellerer, H., Pferschy, U. and Pisinger, D., (2004), Knapsack Problems, Springer-Verlag, Berlin Heidelberg.

[3] Garey, M. R. and Johnson, D. S., (1979), Computers and Intractability: A Guide to the Theory of NP-Completeness, Freemann, New York.

[4] Zak E. J., (2003), The Skiving Stock Problem as a Counterpart of the Cutting Stock Problem, Int. Trans. in Op. Rs., 10: 637-650.

[5] Jahromi, M. H., Tavakkoli-Moghaddam R., Makui A. and Shamsi A., (2012), Solving a one-dimensional cutting stock problem by simulated annealing and tabu search, Journal of Industrial Engineering International, 8(1): 1-8.

[6] Kantorovich L. V., (1960), Mathematical methods of organizing and planning production, Management Sci., 6(4): 366-422.

[7] Gilmore, P. C., and Gomory R.E., (1961), A linear programming approach to the cutting-stock problem, European Journal of Operational Research, 9(6), pp. 849-859.

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

[9] Waescher, G. and Gau T., (1996), Heuristics for the Integer one-dimensional Cutting Stock Problem - A Computational Study, OR Spektrum, 18(3), pp. 131-144.

[10] Gradisar, M., Kljaji M., Resinovi G., and Jesenko J., (1999a), A sequential heuristic procedure for one-dimensional cutting, European Journal of Operational Research, 114(3), pp. 557-568.

[11] Gradisar M., Resinovic G. and Kljajic M., (1999b), A hybrid approach for optimization of one-dimensional cutting, European Journal of Operational Research, 119(3), pp. 719-728.

[12] Vance, P., Barnhart C., Johnson E.L., and Nemhauser G.L., (1994), Solving binary cutting stock problems by column generation and branch-and-bound, Computational Optimization and Applications, 3(2), pp. 111-130.

[13] Vance P., (1998), Branch-and-price algorithms for the one-dimensional cutting stock problem, Computational Optimization and Applications, 9(3), pp. 211-228.

[14] Valerio de Carvalho J.M., (1999), Exact solution of bin-packing problems using column generation and branch-and-bound, Annals of Operation Research, 86, pp. 629-659.

[15] Vanderbeck F., (1999), Computational study of a column generation algorithm for bin-packing and cutting stock problems, Mathematical Programming, 86(3), pp. 565-594.

[16] Vanderbeck F., (2000), On DantzigWolfe decomposition in integer programming and ways to perform branching in a branch-and-price algorithm, Operations Research, 48(1), pp. 111-128.

[17] Scheithauer, G., Terno J., Muller A., and Belov G., (2001), Solving one-dimensional cutting stock problems exactly with a cutting plane algorithm, Journal of the Operational Research Society, 52(12), pp. 1390-1401.

[18] Mukhacheva, E. A., Belov G., Kartak V., and Mukhacheva A. S., (2001), One-dimensional cutting stock problem: Numerical experiments with the sequential value correction method and a modified branch-and-bound method,Pesquisa Operacional 2000(2), pp. 153-168.

[19] Belov, G., and Scheithauer G., (2002), A cutting plane algorithm for the one-dimensional cutting stock problem with multiple stock lengths, European Journal of Operational Research, 141(2), pp. 274-294.

[20] Umetani, S., Yagiura M., and Ibaraki T., (2003), One-dimensional cutting stock problem to minimize the number of different patterns, European Journal of Operational Research, 146(2), 388-402.

[21] Johnston, R.E., and Sadinlija E., (2004), A new model for complete solutions to one-dimensional stock problems, European Journal of Operational Research, 153(1), pp. 176-183.

[22] Belov, G., and Scheithauer G., (2006), A branch-and-cut-and-price algorithm for one-dimensional stock cutting and two-dimensional two-stage cutting, European Journal of Operational Research, 171(1), pp. 85-106.

[23] Dikili A. C., Sarz E. and Pek, N. A., (2007), A successive elimination method for one-dimensional stock cutting problems in ship production, Ocean engineering, 34(13), pp. 1841-1849.

[24] Reinertsen, H., and Vossen, T. W., (2010), The one-dimensional cutting stock problem with due dates, European Journal of Operational Research, 201(3), pp. 701-711.

[25] Cherri, A. C., M. N. Arenales, and Yanasse H.H., (2009), The one-dimensional cutting stock problems with usable leftover: a heuristic approach, European Journal of Operational Research, 196 (3), pp. 85-106.

[26] Cherri, A. C., Arenales M. N., and Yanasse H. H., (2013), The usable leftover one-dimensional cutting stock problem-a priority-in-use heuristic, Intl. Trans. in Op. Res., 20(2), pp. 198-199.

[27] Berberler, M. E., and Nuriyev U., (2010), A New Heuristic Algorithm for the One-Dimensional Cutting Stock Problem, Appl. Comp. Math, 9 (1), pp. 19-30.

[28] Mobasher, A., and Ekici A., (2013), Solution approaches for the cutting stock problem with setup cost, Computers and Operations Research, 40(1), 225-235.

[29] Johnson, M. P., Rennick C. and Zak E., (1997), A new model for complete solutions to one-dimensional stock problems, Siam Review, 39(3), pp. 472-483.

[30] Arbib, C., Marinelli F., Rossi F., and Di Iorio F., (2002), Cutting And Reuse: An Application from Automobile Component Manufacturing, Operations Research, 50 (6), 923-934.

[31] Arbib, C., and Marinelli F., (2005), Integrating Process Optimization and Inventory Planning in Cutting-Stock with Skiving Option: An Optimization Model and its Application, European Journal of Operational Research, 163 (3), pp. 617-630.

[32] MKA Company, Accessed January 6, 2017. https://www.mkasteel.com

[33] Liang, K. H., Yao, X., Newton, Y., and Hoffman, D., (2002), A new evolutionary approach to cutting stock problems with and without contiguity, Comput. Oper. Res., 29(12), pp. 164159.

[34] Yang, C. T., Sung, T. C. and Weng, W. C., (2006), An improved tabu search approach with mixed objective function for one-dimensional cutting stock problems, Advances in Engineering Software, 37(8), pp. 502-513.

[35] http://www.optimalprograms.com/realcut1d.htm (Real Cut optimization 1D software)

[36] http://www.nirvanatec.com/bar_nesting_software.html (Plus 1D optimization software)

Deniz TANIR received his master and PhD degree from Computer Science of Department of Mathematics of Ege University. He has been working in Management and Information Systems at Kafkas University as Assistant Professor since 2018. His research interests include bioinformatics, combinatorial optimization and theory of algorithms.

Onur Ugurlu received his PhD degree from Computer Science of Department of Mathematics of Ege University. He completed his undergraduate studies at Ege University with a major in computer science and a minor in mathematics. He is mainly interested in algorithms, computational theory and graph theory. In particular, his researches focus on finding efficent algorithms for discrete optimization problems on graphs.

Asli Guler graduated from Ege University and works as a Lecturer at Yasar University, Vocational School, Department of Computer Technologies. Her research interests include discrete mathematics, combinatorial optimization, graph theory, and theory of algorithms.

Urfat Nuriyev received his master degree from Azerbaijan State University in 1976, his doctoral degree from Cybernetic Institue of Academy of Sciences of Ukraine in 1983. He gave Associate Professor thesis in 1989. Nuriyev, who won title of professor in 2005, has been interested to operation researches, combinatorial optimization and computer mathematics. He has been working in the Department of Mathematics at Ege University since 1997.