Three Dimensional Cutting Stock Problem in Mattress Production: A Case Study

Selin Altın, Tezcan Aydilek, Umut Şirvan, Damla Kesikburun, Adalet Öner(✉), and Nejat Kutup

Department of Industrial Engineering, Yaşar University, İzmir, Turkey
selin_altn@hotmail.com, tezcanaydilek@gmail.com, umutsirvan@windowslive.com, {damla.kesikburun, adalet.oner,nejat.kutup}@yasar.edu.tr

Abstract. This study involves in a real-world application of 3-D cutting stock problem in a local bedding company. The company produces various types of mattresses (beds). The bill of materials of a mattress usually includes foam rubber (sponge) components. Each mattress has different foam rubber components in the form of orthogonal rectangular prisms with different dimensions. Those components should be cut from big foam rubber blocks. The properties of current cutting machinery used in the company imposes "guillotine" cuts. The aim is to minimize total waste in cutting operations. There are mathematical models to solve cutting stock problems, however they require "cutting patterns" to be generated in advance. Cutting patterns represent potential combinations of how rectangular prisms are fitted into foam rubber blocks. Column Generation method and a Branch and Bound algorithm are used to generate cutting patterns. Mathematical model then solves the problem provided that the cutting patterns are supplied as input. A user friendly Decision Support System (DSS) has been developed in order to incorporate proposed procedures to solve the problem. It enables the user in the company to prepare efficient cutting plans easily and quickly.

Keywords: Optimization · Cutting stock problem · Guillotine cut Cutting patterns · Column generation · Branch and bound Decision Support System

1 Introduction

Foam rubber is used as the cushioning material in mattresses. In the production of mattresses, big foam rubber blocks should be cut into different types of orthogonal rectangular prisms that are the components of a mattress such as *head bars, side bars,* and *comfort layers* as shown in Fig. 1. In this form, the problem is a standard three-dimensional cutting stock problem (3DCSP).

A local bedding company in Izmir produces mattress and suffers a high level of waste in foam rubber cutting operations. Currently, there isn't any systematic or scientific approach for planning of cutting operations. A planning engineer develops cutting plans manually, depending on his/her own judgment and experience. Company

N. M. Durakbasa and M. G. Gencyilmaz (Eds.): ISPR 2018, *Proceedings of the International Symposium for Production Research 2018*, pp. 949–960, 2019.
https://doi.org/10.1007/978-3-319-92267-6_76

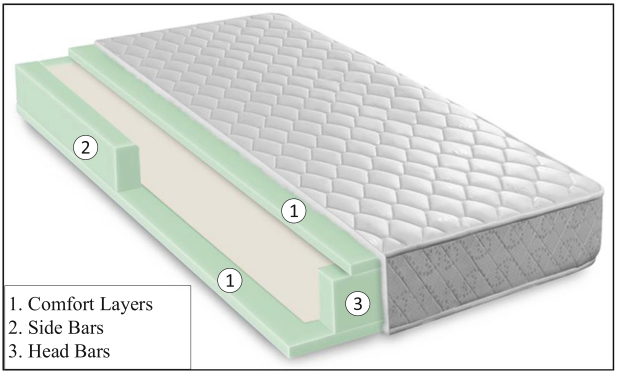


Fig. 1. Foam rubber components of a mattress

management is not happy with the current level of waste. The purpose of this study is to investigate and compile techniques and methods of three-dimensional cutting stock problem and then develop a planning tool that will be used to generate efficient cutting plans. Therefore, the focus of the study is an industrial application of three-dimensional cutting stock problem.

2 Problem Definition

The standard 3-D cutting stock problem (3DCSP) can be defined simply as follows: There is an unlimited quantity of identical big foam rubber block $B = (L, W, H)$ as raw material in producing mattress, where L, W, and H define the length, width and height of the blocks respectively. On the other hand, there is a set of n types of components or items (l, w, h, d) to be cut from big blocks B. The problem is to determine how to cut the smallest possible number of blocks B so as to produce d_i units of each items type i, $i = 1,\ldots, n$. A basic mathematical model can be defined for the standard 3DCSP as follows.

Decision Variables

x_j: Number of Stock material to cut according to pattern j

Cutting patterns represent potential combinations of how components (items) are fitted into foam rubber blocks B.

Parameters

n = Number of items
m = Number of patterns
i = index for *item*, $i = 1, \ldots.\ n$
j = index for *pattern*, $j = 1, \ldots.\ m$
p_{ij} = Number of occurances of the i^{th} item in the j^{th} pattern
d_i = Demand for item i

Mathematical Model

$$\min Z = \sum_{j}^{m} x_j \tag{1}$$

Subject to

$$\sum_{j}^{m} p_{ij} x_j \geq d_i \quad \forall i \in \{1, ..n\} \tag{2}$$

$$x_j \in Z^+ \quad \forall j \in \{1, ..m\} \tag{3}$$

The objective function minimizes total number of cutting patterns to be used. Constraint (2) ensures that the items are produced in desired amounts. Constraint (3) indicates that an integer solution is requested. It is obvious that cutting patterns should be determined in advance as the input of the problem. The hard part of the problem is to generate valid patterns.

A cutting pattern shows how many items of each type are cut from stock material. An example of generating cutting patterns are explained below in Figs. 2 and 3. For simplicity, the idea is shown on two dimensions. However, it can be extended easily for three dimensions. Assume that there are four types of items and they should be cut from stock material as shown in Fig. 2.

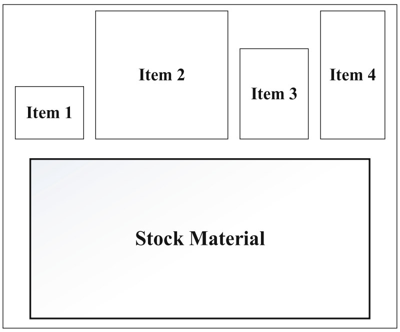


Fig. 2. Example case for generating cutting patterns

The items can be placed on the stock material in different combinations, and each combination represents a pattern. Figure 3 shows that one of the alternative patterns such that five instances of item 1, one instance of item 2 and two instances of item 3 could be placed on the stock material. This placement is just one of many possible combinations. In order to solve the problem optimally, all valid pattern should be determined and listed.

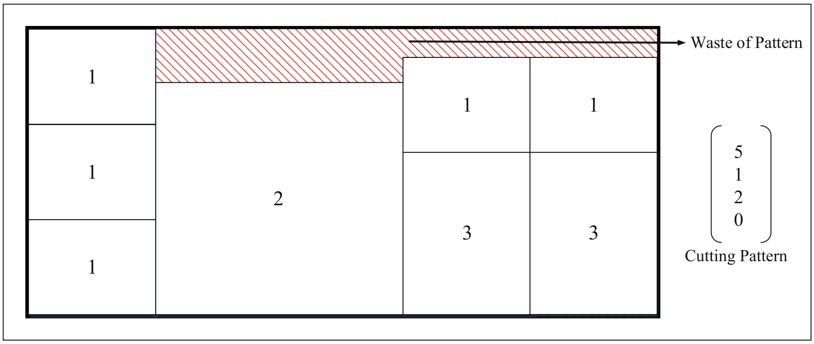


Fig. 3. Generating a cutting pattern

There are many variants and extensions of standard 3DCSP depending on the environment and technological requirements. In this study, we consider the real-world requirements in the bedding company, which can be described as follows:

- Technical specifications and settings of blades of every cutting machine used in the company impose *guillotine* cuts. A guillotine cut is a cut that is parallel to one of the sides of the object and goes from one side to the opposite one.
- There are two types of cutting machines, and each type is set to cut along different Cartesian planes. On one of the machine type, the block B is loaded on the machine and the machine cuts "layers" of foam rubbers parallel to x-y plane that is called "horizontal" cut. The orientation of the block *B* on the machine is fixed and can't be changed. The other type of machine cuts parallel to y-z plane, and it is called "vertical" cut. In this type of machine, the orientation of the objects may be changed as desired. The cutting process is sequential such that any block B is first cut horizontally and generated layers are then cut vertically as many times as required to get components to be used in mattress production. In such a case, cutting process is called *k-staged cutting* in literature [5]. Guillotine cuts are applied in both stages.

The requirements described above leads to a special variant of 3DCSP. The problem turns out to be a 2-staged cutting problem constrained with guillotine cuts. This study focuses on this variant of the problem, and the aim is to develop a solution procedure and then embed it in a decision support tool.

3 Literature Review

In literature, the first known definition and formulation of the cutting stock problem (CSP) were made by Russian economist Kantorovich [1]. Gilmore and Gomory [2] proposed *column generation* method in order to solve single dimension CSP. Generating a column means generating a cutting pattern. Then, they have extended their study to describe how to use column generation method in order to solve multistage CSP of two and more dimensions [3, 4]. However, the proposed method doesn't provide integer solution. Queiroz et al. [5] provide an extensive survey on the 3DCSP.

There are various studies in literature that can be used in order to formulate and solve 2-staged 3DCSP. Among others, some mathematical models are the one-cut concept by Dyckoff [6], the arc-flow concept by Valerio de Carvalho [7], the DP-flow concept by Cambazard and O'Sullivan [8], and the general arc-flow with graph compression by Brandao and Pedroso [9].

There are also exact methods proposed for CSP depending on the *branch & bound* and *branch & cut* algorithms such as Hadjiconstantinou and Christofides [10] and Martello and Toth [11]. In meta-heuristics realm, Kampke [12] and Alvim et al. [13] implemented simulated annealing a tabu search methods respectively.

4 Modelling and Solution Methodology

This study focuses on 2-staged 3DCSP with guillotine cuts as described in "Problem Definition" section. Technological requirements lead us to divide the 3D problem into two consecutive stages. The first stage contains a collection of two-dimensional cutting stock problems (2DCSP), and the second stage contains a one-dimensional cutting stock problem (1DCSP). However, these two stages are not independent of each other, since the output of the first stage (2DCSP) is used as the input in the second stage (1DCSP). Figure 4 shows the relations between the stages.

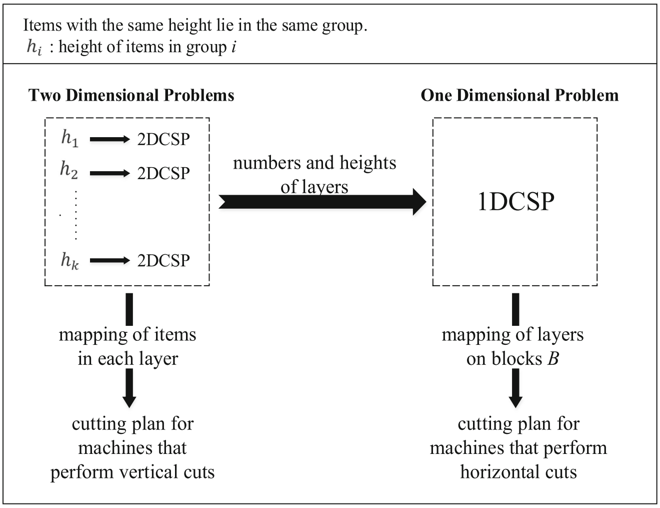


Fig. 4. Problem setting of 2-Staged 3DCSP

In modeling and solution process, the sequence of stages is reversed when compared to the sequence of cutting process in practice. The first stage of the solution procedure corresponds to the second stage of cutting process in practice, which is the vertical cutting stage.

In this problem setting, first, the set of items (components) to be cut are grouped by their heights. The items with the same heights will be in the same group. For each group of items, a 2-D CSP is solved. It leads more than one 2-D problems such that each one corresponds to a particular height. The 2-D problems are independent of each other since they correspond to distinct heights. The solution of each 2-D problem gives the minimum number of "layers" to be cut while ensuring required numbers of items would be produced in a given height. The solutions of those 2-D problems also provide the information of how the items are located on the layers. That information is actually the cutting plan for the machines that performs vertical cuts. (See Fig. 4.)

The output of independent 2-D problems are collected to make a list of layers along with their heights. The layers in that list are to be cut from blocks B. The list of layers is used as the input of the second stage which is a 1DCSP. The solution of the 1-D problem gives the minimum number of blocks B to be used. The solution also provides the information of how the layers are located in block B. That information corresponds to the cutting plan for the machines that performs horizontal cuts.

For the solution procedure, it has been decided to use the model and formulation given in (1)–(3) and it is coded in Lingo 17.0. However, it was required to find a systematic way to determine valid cutting patterns. As mentioned earlier, Gilmore and Gomory [2–4] proposed a column (pattern) generation method.

In column generation method, the problem is divided into two problems: master problem and the auxiliary problem. First, a set of feasible cutting patterns are found and placed in the simplex method as the starting feasible solution. Then a special procedure is applied to find a promising pattern if any. That procedure creates the auxiliary problem which is a knapsack problem whose input is the shadow prices from current basis in the simplex method. The solution of the auxiliary problem is then fed to master problem for the next iteration of simplex method.

Column generation method is easy to implement for 1DCSP but not for multi-dimensional problems. Therefore, other relevant pattern generation methods have been reviewed in the literature. Malaguti et al. [14] present an extensive survey for such methods. Pattern generation methods are usually based on the branch and bound algorithm. Among many others, the methods proposed by Suliman [15] and Rodrigo et al. [16] have been scrutinized, and the latter has been chosen to implement in solution procedure because it is easy to implement and embed into decision support tool. The selected pattern generation methods which are column generation method and the algorithm of Rodrigo et al. [16] are coded in Lingo 17.0 and Excel VBA, respectively. The proposed solution procedure is described in Fig. 5.

A sample problem is prepared below to show that how the proposed solution procedure works. It is assumed that the dimensions of the foam rubber blocks are $200 \times 180 \times 100$ cm that corresponds to length, width and height (L, W, H) respectively. Assume also that production plan is received and it is ordered to produce seven different mattress types. The types and order amounts of mattresses are shown in Table 1.

The BOM data indicates the number and dimensions of the components for each mattress type. Usually, each mattress must have two comfort layers, two head bars, and two side bars (see Fig. 1). Using BOM data, the dimensions and numbers of all the components (items) may be listed easily. For simplicity, comfort layers component were excluded from the list, therefore only head and side bars were considered. The list

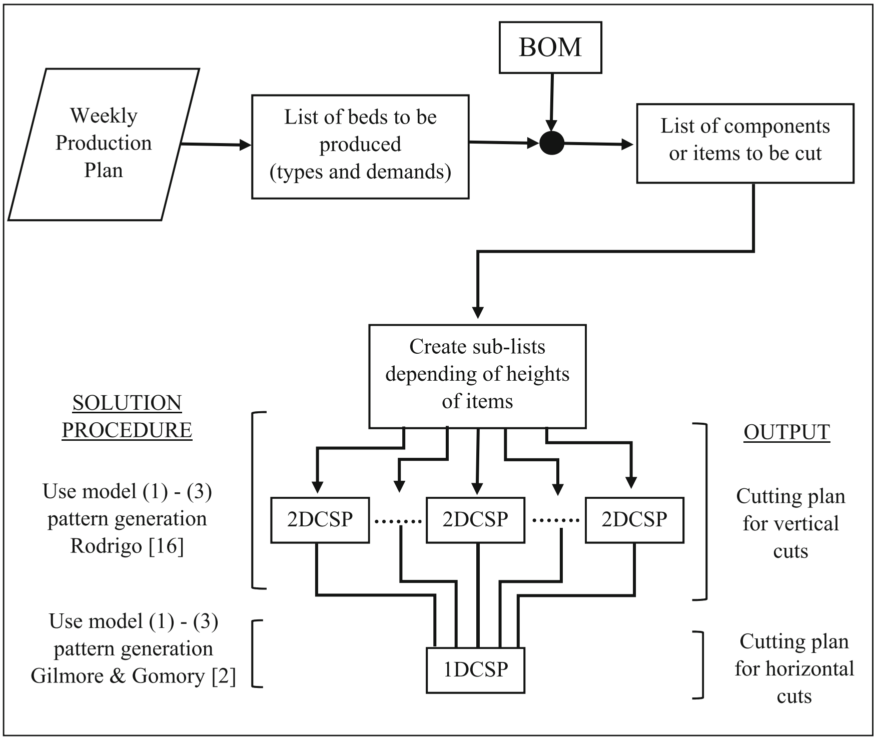


Fig. 5. Proposed solution procedure

Table 1. Sample problem – production orders

Mattress type	Demand
A	15
B	18
C	21
D	14
E	13
F	12
G	20

of the items are then divided into sub-lists depending on their heights. Each sub-list contains the items with the same heights. The production plan and BOM data lead us to have three item sub-lists shown in Tables 2, 3 and 4.

Each sub-list is the input for separate and independent 2DCSPs. The lengths and widths of items are considered in those two-dimensional problems whereas the heights are ignored. The branch and bound method in Rodrigo [16] has been used to generate the cutting patterns and then cutting patterns have been used to find the optimal solution in model defined in (1)–(3). Tables 5, 6 and 7 presents the output of the two-dimensional problems.

Table 2. Item sub-list 1 (All heights = 8 cm)

Mattress type	Component	Length (cm)	Width (cm)	Height (cm)	Demand
A	Head bar	130	16	8	30
	Side bar	170	16	8	30
B	Head bar	150	14	8	36
	Side bar	190	14	8	36
C	Head bar	190	20	8	42
	Side bar	200	20	8	42

Table 3. Item sub-list 2 (All heights = 13 cm)

Mattress type	Component	Length (cm)	Width (cm)	Height (cm)	Demand
D	Head bar	130	16	13	28
	Side bar	170	16	13	28
E	Head bar	190	20	13	26
	Side bar	200	20	13	26

Table 4. Item sub-list 3 (All heights = 17 cm)

Mattress type	Component	Length (cm)	Width (cm)	Height (cm)	Demand
F	Head bar	130	16	17	24
	Side bar	170	16	17	24
G	Head bar	150	14	17	40
	Side bar	190	14	17	40

Table 5. Output of two dimensional problem for sub-list 1 (All heights = 8 cm)

Mattress Type	Component	Cutting patterns selected to be used								
		1	2	3	4	5	6	Demand	Generated	
A	Head bar	0	0	0	0	0	6	30	30	0
A	Side bar	0	0	1	1	7	3	30	30	0
B	Head bar	0	0	0	0	6	6	36	36	0
B	Side bar	0	4	6	3	0	0	36	37	1
C	Head bar	0	0	4	6	0	0	42	42	0
C	Side bar	9	6	0	0	0	0	42	42	0
		Number of								
		4	1	3	5	1	5	= 19	Total number of patterns to be used	

Table 6. Output of two dimensional problem for sub-list 2 (All heights = 13 cm)

Mattress Type	Component	Cutting patterns selected to be used								
		1	2	3	4	5	6	Demand	Generated	Surplus
D	Head Bar	0	0	0	2	4	11	28	28	0
D	Side Bar	0	0	2	10	8	4	28	28	0
E	Head Bar	1	8	7	0	0	0	26	26	0
E	Side Bar	8	1	0	0	0	0	26	26	0
		Number of patterns selected to be used								
		3	2	1	1	1	2	= 10	Total number of patterns to be used	

Table 7. Output of two dimensional problem for sub-list 3 (All heights = 17 cm)

Mattress Type	Component	Cutting patterns selected to be used									
		1	2	3	4	5	6	7	Demand	Generated	Surplus
F	Head Bar	0	0	0	0	1	2	11	24	25	1
F	Side Bar	0	2	3	3	3	3	4	24	25	1
G	Head Bar	0	0	0	12	11	10	0	40	45	5
G	Side Bar	12	10	9	0	0	0	0	40	43	3
		Number of patterns selected to be used									
		2	1	1	2	1	1	2	= 10	Total number of patterns to be used	

Once two-dimensional problems are solved, the output specifies selected cutting patterns. Those patterns have merged into a cutting plan for the machines that perform vertical cuts. The output also provides an information on the total numbers of patterns to be used. Then, the total numbers of patterns along with their respective heights have fed into the problem in the second stage, which is a 1DCSP. Tables 8 and 9 show the input and output of the 1DCSP, respectively.

For the sample problem, the waste level is calculated as 6.25%. The solution procedure has been tested for 5 different problems in local bedding company. As a result of solving these problems, reduction in waste levels are shown in the Table 10. It has been observed that the solution time of test problems vary between 1 and 2 min.

Table 8. Input of one dimensional problem

Height	Demand
8	19
13	10
17	10

Table 9. Output of one dimensional problem

Height	Selected Cutting Patterns				
	1	**2**	**Demand**	**Generated**	**Surplus**
8	6	1	19	20	1
13	4	0	10	12	2
17	0	5	10	10	0
	Number of blocks required		**Total blocks required**		
	3	2	5		

Table 10. Reduction in waste level for test problems

Test problem	Reduction in waste level
1	2.79%
2	2.61%
3	4.16%
4	3.28%
5	1.42%

5 Decision Support System

The solution procedure has been implemented in a user friendly decision support tool. The tool enables the user to develop the cutting plans easily and efficiently. The interface of the tool is shown in Fig. 6. The interface incorporates three main sections. The first section is related to managing the production plan which contains the production orders of the mattresses. In this section, it is possible to list, delete, add or edit the entries in the production plan. The order management screen is shown in Fig. 7. It helps the user to have the updated version of the production plan. The second section takes the entries in production plan and process them with BOM data in order to generate item list to be cut and then solves the problem. At this point, the user is

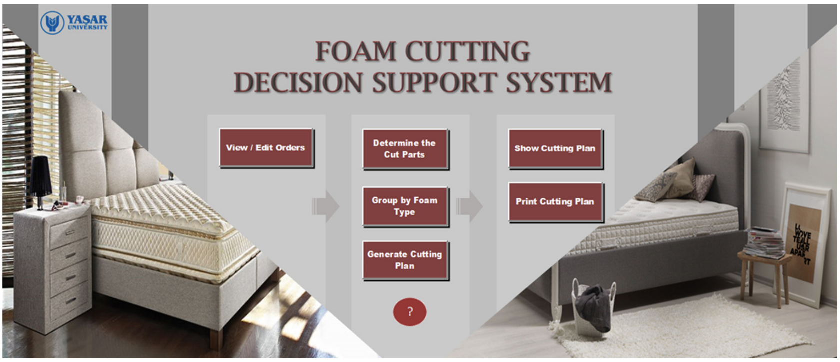
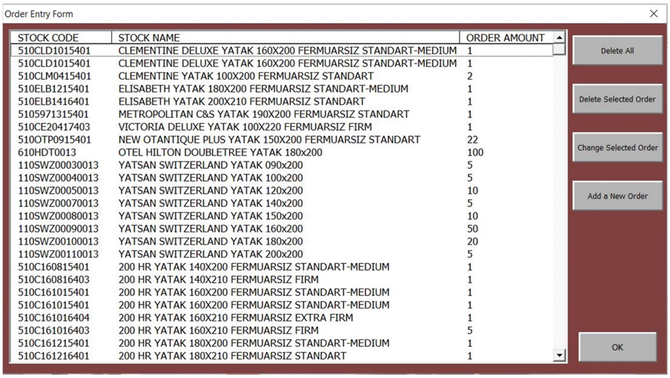


Fig. 6. Main interface of planning tool

informed about the stages of the solution. Finally, the third section allows the user to see and review the cutting plans. The outcome can be displayed in different format and in different detail levels. It is also possible to print the outcome in selected format.



STOCK CODE	STOCK NAME	ORDER AMOUNT
510CLD1015401	CLEMENTINE DELUXE YATAK 160X200 FERMUARSIZ STANDART-MEDIUM	1
510CLD1015401	CLEMENTINE DELUXE YATAK 160X200 FERMUARSIZ STANDART-MEDIUM	1
510CLM0415401	CLEMENTINE YATAK 100X200 FERMUARSIZ STANDART	2
510ELB1215401	ELISABETH YATAK 180X200 FERMUARSIZ STANDART-MEDIUM	1
510ELB1416401	ELISABETH YATAK 200X210 FERMUARSIZ STANDART	1
5105971315401	METROPOLITAN C&S YATAK 190X200 FERMUARSIZ STANDART	1
510CE20417403	VICTORIA DELUXE YATAK 100X220 FERMUARSIZ FIRM	1
510OTP0915401	NEW OTANTIQUE PLUS YATAK 150X200 FERMUARSIZ STANDART	22
610HDT0013	OTEL HILTON DOUBLETREE YATAK 180x200	100
110SWZ00030013	YATSAN SWITZERLAND YATAK 090x200	5
110SWZ00040013	YATSAN SWITZERLAND YATAK 100x200	5
110SWZ00050013	YATSAN SWITZERLAND YATAK 120x200	10
110SWZ00070013	YATSAN SWITZERLAND YATAK 140x200	5
110SWZ00080013	YATSAN SWITZERLAND YATAK 150x200	10
110SWZ00090013	YATSAN SWITZERLAND YATAK 160x200	50
110SWZ00100013	YATSAN SWITZERLAND YATAK 180x200	20
110SWZ00110013	YATSAN SWITZERLAND YATAK 200x200	5
510C160815401	200 HR YATAK 140X200 FERMUARSIZ STANDART-MEDIUM	1
510C160816403	200 HR YATAK 140X210 FERMUARSIZ FIRM	1
510C161015401	200 HR YATAK 160X200 FERMUARSIZ STANDART-MEDIUM	1
510C161015401	200 HR YATAK 160X200 FERMUARSIZ STANDART-MEDIUM	1
510C161016404	200 HR YATAK 160X210 FERMUARSIZ EXTRA FIRM	1
510C161016403	200 HR YATAK 160X210 FERMUARSIZ FIRM	5
510C161215401	200 HR YATAK 180X200 FERMUARSIZ STANDART-MEDIUM	1
510C161216401	200 HR YATAK 180X210 FERMUARSIZ STANDART	1

Fig. 7. Management of the production plan

6 Conclusion and Future Work

In this study, 3DCSP has been solved in two consecutive stages, since the block orientation is not allowed in the company. The first stage contains a collection of two-dimensional cutting stock problems (2DCSP), and the second stage contains a one-dimensional cutting stock problem (1DCSP). The important point in this study is that the output of the first stage (2DCSP) is used as the input in the second stage (1DCSP). According to the results of 5 sample data, the decrease in waste level is between 1.4% and 4.2%.

A flexible and user friendly tool has been developed for the company to enable the planning engineer to prepare quick and efficient cutting plans. It has been a useful tool for the company since they don’t have such a tool before this study. It manages the input of the problem and present the solutions to the user through a user-friendly interface. The solution process and outcomes has been verified and validated. An observable decrease (in between 1.4%–4.2%) in waste level has been reported. On the other hand, the company now have a means of data collection and history of cutting process.

An important point can be studied in the future, if the orientation of the blocks *B* can be changed. Because, more decrease in waste levels may be attained by using some other solution methods, once the orientation of the blocks is allowed within the company.

Acknowledgment. This study is supported by TUBITAK (The Scientific and Technological Research Council of Turkey) in the program of “2209-B Undergraduate Thesis Support Program for Industrial Applications”. We would like to thank our colleagues Buğra Akboy, Tevfik Ercan, Simay Karaşahin and Yusuf Rıdvanoğulları for their help and contribution to the study.

References

1. Kantorovich LV (1960) Mathematical methods of organizing and planning production. Manag Sci 6:366–422
2. Gilmore PC, Gomory RE (1961) A linear programming approach to the cutting stock problem. Oper Res 9:849–859
3. Gilmore PC, Gomory RE (1963) A linear Programming approach to the cutting stock problem: Part II. Oper Res 11:863–888
4. Gilmore PC, Gomory RE (1965) Multistage cutting stock problems of two and more dimensions. Oper Res 13:94–120
5. Queiroz TA, Miyazawa FK, Wakabayashi Y, Xavier EC (2012) Algorithms for 3D guillotine cutting problems: unbounded knapsack, cutting stock and strip packing. Comput Oper Res 39:200–212
6. Dyckoff H (1981) A new linear programming approach to the cutting stock problem. Oper Res 29:1092–1104
7. Val´erio de Carvalho JMV (1999) Exact solution of bin packing problems using column generation and branch and bound. Annal Oper Res 86:629–659
8. Cambazard H, O'Sullivan B (2010) Propagating the bin packing constraint using linear programming. In: Principles and practice of constraint programming – CP 2010. LNCS, vol 6308, pp 129–136
9. Brandao F, Pedroso JP (2016) Bin packing and related problems: general arc-flow formulation with graph compression. Comput Oper Res 69:56–67
10. Hadjiconstantinou E, Christofides N (1995) An exact algorithm for general, orthogonal, two-dimensional knapsack problems. Eur J Oper Res 83:39–56
11. Martello S, Toth P (1990) Knapsack problems: algorithms and computer implementations. Wiley, Chichester
12. Kämpke T (1988) Simulated annealing: use of a new tool in bin packing. Annal Oper Res 16:327–332
13. Alvim ACF, Ribeiro CC, Glover F, Aloise DJ (2004) A hybrid improvement heuristic for the one-dimensional bin packing problem. J Heuristics 10:205–229
14. Malaguti E, Durán RM, Toth P (2014) Approaches to real world two-dimensional cutting problems. Int J Manag Sci 47:99–115
15. Suliman MA (2001) Pattern generating procedure for the cutting stock problem. Int J Prod Econ 74:293–301
16. Rodrigo WNP, Daundasekera WB, Perera AAI (2012) Pattern generation for two-dimensional cutting stock problem. Int J Math Trends Technol 3:54–62