DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

RESOLVING MARBLE CUTTING PROBLEMS WITH MATHEMATICAL PROGRAMMING AND HEURISTIC APPROACHES

by

Burcu KUBUR

June, 2016

İZMİR

RESOLVING MARBLE CUTTING PROBLEMS WITH MATHEMATICAL PROGRAMMING AND HEURISTIC APPROACHES

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Master of Science
in Industrial Engineering, Industrial Engineering Program

by

Burcu KUBUR

June, 2016

İZMİR

M.Sc THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"RESOLVING MARBLE CUTTING PROBLEMS WITH MATHEMATICAL PROGRAMMING AND HEURISTIC APPROACHES"** completed by **BURCU KUBUR** under supervision of **PROF. DR. ADİL BAYKASOĞLU** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Prof. Dr. Adil BAYKASOĞLU

Supervisor

Asst. Prof. Dr. Ömer ÖZTÜRKOĞLU

Jury Member

Assoc. Prof. Dr. A. Serdar TAŞAN

Jury Member

Prof. Dr. Ayşe OKUR

Director

Graduate School of Natural and Applied Sciences

ACKNOWLEDGMENT

First and foremost, I would like to express my sincere gratitude to my supervisor, Prof. Dr. Adil BAYKASOĞLU for his being accessible no matter how busy schedule he had and for his guidance, continuing support and encouragement that helped me overcome challenges throughout the present work. It is my great pleasure and honor to be associated with him.

I would like to thank, everyone who have helped me throughout the progress of dissertation, my colleagues and all staff member in the Industrial Engineering Department of Dokuz Eylül University. Especially I am grateful to my friends Res. Assist. İlker GÖLCÜK and Res. Assist. M. Emre ŞENOL for their invaluable support.

I would like to introduce my great thanks to my friend Raziye OKYAY for her support, whenever I need, and listening to my complaints during this period.

I would like to express my deep thankfulness to my parents, who have loved and supported me whole my life, Sevim KUBUR, Ali KUBUR and my sister Bahar KAYÇEN.

Last but not the least; I want to present my special thanks to my fiancé, Yaşar Can ÖZBEL for his endless patience, encouragement, continuous support.

Burcu KUBUR

RESOLVING MARBLE CUTTING PROBLEMS WITH MATHEMATICAL PROGRAMMING AND HEURISTIC APPROACHES

ABSTRACT

In this study, we deal with the one-dimensional cutting stock problem of a marble company. The marble plane cutting problem can be defined as follows: how to cut a set of stock lengths of marble blocks with known sizes, qualities and available quantities, in order to produce exactly a set of marble planes with specified sizes, qualities and demands, so that the total cost of transportation and overgrading is minimized. One of the salient features of this problem is that it does not require a priori enumeration of all possible cutting patterns. In this study, generation of retail to reuse is permitted.

We developed a mixed integer linear programming (MILP) model for formally describing the extended problem which is solved to global optimality by using standard techniques. However, the above constitutes a complex combinatorial problem therefore its solution is carried out by utilizing a specifically developed SDS algorithm.

Furthermore, we considered abovementioned problem of one-dimensional marble plane cutting problem with knife dependent setup cost with usable leftover. In industrial cutting processes setup cost occur whenever changing over one cutting pattern to another since the cutting equipment has to be prepared in order to meet technical requirements of the new pattern. Therefore, cutting stock problem and sequencing problem have to be simultaneously considered in a cutting process. The goal here is to minimize transportation, overgrading and knife dependent setup costs. The main endeavor of this study is to specifically developed SDS algorithm to solve the one-dimensional marble plane cutting problem by taking into sequence dependent setups cost between cutting marble blocks. A number of real problem instances are presented in order to illustrate the efficiency and applicability of the

proposed models. Computational experiments show the usefulness of the proposed approaches.

Keywords: Cutting stock problem, sequence-dependent setup time, stochastic diffusion search, marble cutting industry.

MERMER KESME PROBLEMLERİNİN MATEMATİKSEL PROGRAMLAMA VE SEZGİSEL YÖNTEMLERLE ÇÖZÜMLENMESİ

ÖZ

Bu çalışmada, mermer endüstrisinde karşılaşılan tek boyutlu kesme problemi ele alınmıştır. Mermer plaka kesme problemi şu şekilde tanımlanabilir: Miktar, ebat ve kaliteleri bilinen mermer plaka taleplerinin ebatları, kalitesi ve miktarları bilinen mermer bloklarından karşılanırken toplam ulaşım ve üst düzey kaliteye sahip mermer blok ile talep karşılamadan kaynaklı maliyetlerin minimize edilmesi problemidir. Bu problemin dikkat çeken noktalarından biri ise önceden bütün kesme plan alternatiflerinin oluşturulmasına gerek duyulmamasıdır. Bu çalışmada, kesimden sonra geriye kalan artık malzemenin yeniden kullanılmasına izin verilmektedir.

Ele alınan problemi formal olarak tanımlamak ve standart teknikleri kullanarak çözmek için karışık tamsayılı doğrusal programlama modeli geliştirilmiştir. Fakat ele alınan problem kompleks kombinatoriyal yapıda olduğu için stokastik yayılım arama algoritması geliştirilmiştir.

Ayrıca, ele alınan bir boyutlu mermer plaka kesme problemi artık malzemenin yeniden kullanılmasının yanında bıçak bağımlı hazırlık maliyeti ile birlikte düşünülmüştür. Endüstriyel kesim işlemlerinde bir kesme planından diğerine geçerken hazırlık maliyeti oluşur. Çünkü kesme ekipmanlarının yeni kesim planı için hazırlanması gerekmektedir. Bu nedenle, endüstriyel kesim işlemlerinde kesme problemi ve sıralama problemi birlikte düşünülmelidir. Burada amaç, ulaşım maliyetleri, üst düzey kaliteye sahip mermer blok ile talep karşılamadan kaynaklı maliyetler ve bıçak bağımlı hazırlık maliyetleri toplamının minimize edilmesidir. Bu çalışmadaki asıl çaba problemin çözümü için stokastik yayılım arama algoritması temelli yeni bir sezgisel algoritma önerilmesidir. Önerilen modellerin etkililik ve uygulanabilirliğini göstermek için bir dizi gerçek problem örnekleri sunulmuştur. Sonuçlar önerilen modellerin kullanışlılığını göstermiştir.

Anahtar kelimeler: Stok kesme problemi, sıra-bağımlı hazırlık zamanı, stokastik yayılım arama algoritması, mermer kesme endüstrisi.

CONTENTS

Page

M.Sc THESIS EXAMINATION RESULT FORM ii
ACKNOWLEDGMENT iii
ABSTRACT iv
ÖZ vi
LIST OF FIGURES xii
LIST OF TABLES xiii

CHAPTER ONE-INTRODUCTION 1

1.1 Importance of the Problem 1
1.2 Framework of the Dissertation 5
1.3 Research Contributions 6
1.4 Outline of the Thesis 8

CHAPTER TWO-CUTTING AND PACKING PROBLEMS 9

2.1 Definition of Cutting and Packing Problems 9
2.2 Characteristic of Cutting and Packing Problems 11
2.2.1 Characteristic Based on Logical Structure 11
2.2.1.1 Dimensionality 11
2.2.1.2 Type of Assignment 12
2.2.1.3 Characteristic of Large Objects and Small Items 12
2.2.1.3.1 Type of Quantity Measurement 12
2.2.1.3.2 Figure (shape) 13
2.2.1.3.3 Assortment 13
2.2.1.3.4 Availability 13
2.2.1.4 Pattern Restrictions 13
2.2.1.4.1 Geometric Characteristics 14

2.2.1.4.2 Operational Characteristics 15
2.2.1.4.3 Allocation Dynamics 16
2.2.1.5 Objectives 16
2.2.1.6 Status of Information and Variability of Data 17
2.2.1.7 Solution Methods 17
2.2.2 Reality-Based Characteristics 17
2.2.2.1 Kind of Objects and Items, and Branch of Industry 18
2.2.2.2 Planning Context 18
2.2.2.3 Software 19
2.3 Cutting and Packing Problem Classifications 21
2.4 Classic Cutting and Packing Problems with Mathematical Notations 25
2.4.1 General 1DCSP 25
2.4.2 Standard 1DCSP 27
2.5 Cutting and Packing Problem Complexity and Solution Approaches 28
2.6 Literature Review of Existing Algorithms 32

CHAPTER THREE-A REAL-WORLD MARBLE CUTTING PROBLEM AND DEVELOPMENT OF MIXED INTEGER LINEAR PROGRAMMING MODEL 41

3.1 Chapter Introduction 41
3.2 Existing Mathematical Models 46
3.3 Process Description 50
3.4 Problem Statement 53
3.5 The Mixed Integer Linear Programming Model 54
3.5.1 Model Notations 55
3.6 An Illustrative Examples 61
3.7 Computational Experiments 67
3.8 Chapter Conclusions 69

CHAPTER FOUR-STOCHASTIC DIFFUSION SEARCH ALGORITHM FOR SOLVING A REAL-WORLD MARBLE CUTTING PROBLEM 70

4.1 Introduction 70
4.2 SDS Algorithm 71
4.2.1 Origins 71
4.2.1.1 Hinton Mapping 71
4.2.1.2 Template Matching 71
4.2.2 SDS Algorithm 71
4.2.3 Positive and Negative Aspects of SDS Algorithm 74
4.2.4 Selection Mechanisms of SDS Algorithm 76
4.3 The Proposed Constructive SDS Algorithm 76
4.3.1 Solution Representation 77
4.3.2 Generating Initial Solution 81
4.3.3 Calculate Fitness Procedure 82
4.3.4 Neighborhood Structures 84
4.3.4.1 Single Swap 84
4.3.4.2 Two Block Swap 84
4.3.4.3 Three Block Swap 85
4.3.4.4 Inversion 85
4.3.4.5 Lox Crossover 86
4.4 Application of the Proposed Solution Method to a Real-World Marble Cutting Problem Examples 87
4.5 Chapter Summary 88

CHAPTER FIVE-A STOCHASTIC DIFFUSION SEARCH ALGORITHM FOR THE ONE-DIMENSIONAL CUTTING STOCK PROBLEM WITH SETUPS……..... 89

5.1 Introduction 89
5.2 Solution Approaches 90
5.3 Proposed Constructive SDS Algorithm 91

5.4 Application of the Proposed SDS Algorithm to a Real-World Marble Cuttting Problem Examples 93
5.5 Chapter Summary 94

CHAPTER SIX-CONCLUSION 95

6.1 Summary 95
6.2 Contribution of the Study 96
6.3 Future Works 97

REFERENCES 98

LIST OF FIGURES

Page

Figure 2.1 Phenomena of cutting and packing……..... 10
Figure 2.2 Example of orthogonal and non-orthogonal pattern..... 14
Figure 2.3 Examples of guillotine patterns in C&P problems..... 15
Figure 2.4 Example of nested pattern 15
Figure 3.1 Some cutting plan examples 51
Figure 3.2 Three quality level of marble blocks..... 52
Figure 3.3 Marble plane production process..... 53
Figure 3.4 Model input and output..... 55
Figure 3.5 Solution representations of problem instance1..... 68
Figure 4.1 Solution representation of X..... 79
Figure 4.2 Solution representation of Y..... 81
Figure 4.3 Single swap move..... 84
Figure 4.4 Two block swap move..... 85
Figure 4.5 Three block swap move..... 85
Figure 4.6 Inversion move 86
Figure 4.7 Lox crossover move..... 86

LIST OF TABLES

Page

Table 2.1 Reality-based characteristic 19
Table 2.2 Characteristic based on logical structure 20
Table 2.3 Classification of Cutting and Packing Problems 21
Table 2.4 C&P problem classes according to assortment and assignment 22
Table 2.5 C&P problem classes additional dimensionality characteristic to assortment and assignment 22
Table 2.6 Classification of solution approaches 29
Table 3.1 Model Notations..... 56
Table 3.2 Number of customer demand types of marble planes for problem instance1 62
Table 3.3 Available marble blocks data for cutting operation for problem instance1 63
Table 3.4 Number of customer demand types of marble planes for problem instance2 63
Table 3.5 Available marble blocks data for cutting operation for problem instance2 64
Table 3.6 Number of customer demand types of marble planes for problem instance3 64
Table 3.7 Available marble blocks data for cutting operation for problem instance3 65
Table 3.8 Number of customer demand types of marble planes for problem instance4 65
Table 3.9 Available marble blocks data for cutting operation for problem instance4 66
Table 3.10 Number of customer demand types of marble planes for problem instance5 66
Table 3.11 Available marble blocks data for cutting operation for problem instance5 67

Table 3.12 Computational statistics for problem instances 67
Table 4.1 Standard SDS operation 72
Table 4.2 Test phase with comparison of targets 72
Table 4.3 Standard SDS diffusion phase 73
Table 4.4 Proposed constructive SDS algorithm 77
Table 4.5 Marble block data of example problem 78
Table 4.6 Customer order data of example problem 79
Table 4.7 Initial solution construction procedure 81
Table 4.8 Calculate fitness procedure 82
Table 4.9 Computational results for problem instances 88
Table 5.1 Calculate fitness procedure for knife dependent setup 91
Table 5.2 Computational results for problem instances 93

CHAPTER ONE
INTRODUCTION

1.1 Importance of the Problem

The problems called as material cutting and packing (C&P) in manufacturing process is one of the classical and useful problem. In industrial production process raw materials need to be transformed into end products or intermediate products that are required by the consumer. These transformations include many processes that vary according to different types of industries. Cutting process is one of the most commonly applied transformation techniques in wide range of industries. Cutting stock is especially important problem when the customers require smaller size products. In these situations, larger stock materials are required to be cut efficiently in order to satisfy the customer order. Generally, cutting a large material produces waste as the pieces of the material cannot be joined together. Therefore, sophisticated techniques are entailed in CSPs in order to minimize trim-loss material.

Gilmore and Gomory (1961) defined classical form of CSP as finding a convenient cutting pattern by satisfying customer demand with minimum waste. Based on the pioneering study of Gilmore and Gomory (1961), efficient solution methods for the classical formulation of the CSP have been developed. These methods are aimed at identifying the patterns and the run-lengths which are needed to meet a predefined customer demand with a minimum waste. In addition, the term cutting plan generally is used for two different meanings in the scope of CSP. The first one refers to determination of cutting patterns, and the other means frequency of cutting patterns (i.e., their run-lengths). To generate appropriate cutting patterns is the most critical point of CSP. Because the suitable cutting plan minimizes used stock lengths or the amount of trim-loss, it results in increased overall system efficiency.

In the literature, there exist some cutting problems which are called under different names. For example, if time is considered as the limited stock lengths, then

the objective of the CSP will be distribution of these times, instead of parts. For instance, in assembly line balancing problem tasks are completed before predetermined times in a certain sequence. The aim of the assembly line balancing problem is to assign jobs to minimum number of workstations. In this problem, if tasks are considered as small items, stations as the large objects, and the assigned operation as packed operation, then it will basically become CSP problem. Another example will be, if the limited stock lengths are taken as a money, then CSP will correspond to capital budgeting problem which distributes money into the different cost items.

One-dimensional cutting stock problem (1DCSP) was introduced by Kantorovich (1939). In 1DCSP, various numbers of large items are cut into smaller pieces along a single dimension. Afterwards, CSP has attracted many researchers and industries. The reason is that there are many real life applications of the CSP in wide range of industries. However, there is no method which can solve all of these problems optimally in a reasonable time limitation due to their combinatorial nature.

Since C&P problems cannot be solved by using polynomial time algorithms, they are known to be non-deterministic polynomial-time hard (NP-hard) problem (Scheithauer, 1991). In the literature, there exist small number of exact solution methods since these problems are combinatorial. Small-sized problems can be solved by using mathematical modelling approaches optimally. However, when the size of the problem gets larger, then problem complexity increases exponentially. For this reason, in recent years, large-sized problems have been solved by using metaheuristic approaches. Metaheuristic algorithms cannot guarantee the optimal solutions but they provide acceptable solutions in a reasonable time duration. Hence, heuristic and metaheuristic methods become the only viable algorithms to find a solution for large-sized problems.

In the literature, there is no solution methodology which can solve all types of CSP. Developed algorithms can find optimum solution in reified case; i.e., objective of the problem is minimizing trim-loss and there are only geometric and demand

constraints. Unlike reified cases, real life CSP problems include specific constraints and objectives related with process labor, capacity, inventory, transportation cost etc. In many industrial settings, these constraints and objectives make it difficult to utilize existing models in real world application.

Due to its industrial importance, CSP has been studied extensively. Some of the applications of 1DCSPs in different industries can be given as follows: paper industry (Chauhan et al., 2008), metal industry (Sinuany-Stern, & Weiner, 1994) steel industry (Armbruster, 2002), wood industry (Koch et al., 2009), aluminum industry (Stadtler, 1990), construction industry (Shahin, & Salem, 2004). On the other hand, two or more dimensional CSPs can take place in real life situations. For example, plates and sheet materials may be cut into rectangles or irregular shapes. Wäscher et al. (2007) provided an overview of the different variations of CSP, i.e. multi-dimensional cases, including the close relationship with bin packing problems.

Companies generally try to optimize cutting processes due to the economic reasons, such as minimizing costs while considering the filling of the customer orders that depend on the use of stock material. The aim of minimizing used material is equivalent to minimization of trim-loss. However, in some cases trim-loss material indeed cannot be taken as waste. If trim-loss material is long enough, it can be returned to stock area and can be used again in the subsequent planning periods. This problem is known as cutting stock problem with usable leftover (CSPUL). If only one dimension is considered in the problem, it is called one-dimensional cutting stock problem with usable leftover (1DCSPUL). In cutting material process, when the leftover is large enough and it is erroneously considered as a waste, then this incurs extra costs for the company.

In solution methods of 1DCSPUL, decision maker sets the threshold value for leftover material. In other words, if the leftover amount is smaller than the pre-determined threshold value, it is considered as waste. On the other hand, if the leftover amount is equal and/or greater than the pre-determined threshold value, it is considered as leftover. Cherri et al. (2009) emphasized some of the critical

assumptions in the 1DCSPUL which need to be satisfied. For instance, customer order and stock material are known exactly.

In the 1DCSP literature, it is usually assumed that the setup times are negligible. One of the most important reasons is setup time takes lower time than cutting operation time for most of the industrial cutting processes. However, changing from one cutting pattern to another involves significant setup time (Vanderbeck, 2000). Therefore, in some studies, number of different patterns are minimized so that the total setup times are reduced as a result of fewer changeovers. This problem is called minimization of different pattern numbers. In C&P problems, main objectives are generally as follows: minimization of trim-loss, minimization of used stock material, or value maximization. However, in real life problems, there exist many other objectives and constraints. Some of the costs and constraints in 1DCSP problem which are defined by Belov and Scheithauer (2003) as follows:

Cost components in 1DCSP:

- Used material costs
- Setup costs between altering cutting patterns
- Costs of open stacks during the cutting sequence

Constraints in 1DCSP:

- Number of customer orders
- Number of open stacks
- The highest amount of knives which can be used in a cutting pattern
- The maximum distance between knives
- The frequency of a cutting pattern
- Layout constraints, i.e. some items may be placed in certain portions of a given pattern
- Machine eligibility constraints.

1.2 Framework of the Dissertation

Nature of the cutting process in the marble industry is encountered with high amounts of waste. In general, the efficiency of the marble sector is evaluated with two main factors related with processes. One is quarrying process of marble blocks and the other one is marble production process in factories. Both of these activities produce raw marble material residual or processed marble material residual. These residuals are mostly considered as a waste marble material. Data shows production efficiency in a marble process is very low. In addition to above mentioned factors, efficiency can be improved by using cutting planning optimization tools or by increasing cutting machine capacity.

The main aim of this work is to present a real life marble plane cutting problem and develop mathematical models and metaheuristic approaches in order to solve the considered problem. Firstly, a MILP model is proposed for describing the presented problem in which the notion of leftover is considered. To the best of our knowledge, cutting problems with usable leftover in marble plane manufacturing is not considered in the literature. Standard techniques are able to solve the small-scale problems global optimally. However, due to the above mentioned combinatorial characteristic of the problem, the developed MILP model is not capable of solving large-scale problems. As the C&P problems are NP-hard, the presented marble plane cutting problem is also NP-hard. Therefore, its solution is carried out by utilizing a specifically developed constructive SDS algorithm. Finally, presented marble plane cutting problem is considered by taking into consideration knife dependent setup costs along with SDS algorithm.

In this study, one-dimensional marble plane cutting problem is developed in order to obtain cutting plane with minimum overgrading and transportation costs. The presented model includes both marble process and reuse situation related constraints. Both of the mathematical programming and metaheuristic approaches are considered within the scope of the presented study.

1.3 Research Contributions

Contributions of this study can be outlined as follows:

- In the literature, in most of the studies in CSPs, all stock lengths are assumed to be the equal or very few stock lengths are considered as different. However, choosing appropriate stock lengths, which are not assumed to be equal, based on the customer demand will increase material usage efficiency and reduce amount of waste. Therefore, in the considered problem, selection of the convenient stock materials from the available set of different stock lengths is considered. Selection and cutting of these materials are incorporated into model simultaneously in order to obtain better results.
- Current CSP algorithms require a predetermined cutting pattern as an input in which there is a large number of cutting pattern combinations. Even though the algorithms result in an optimal cutting plan for the small-sized problems, they generate poor quality solutions in large-sized problems. One of the reasons is that very limited number of possible combinations is taken into account in the large-sized problems. Thus, some of the high quality solutions are ignored. The presented model does not necessitate pre-specification of cutting plans in advance as the cutting plans are being generated after the model is solved. This property gives flexibility to model in coping with the large-sized problems or changes in demand. The obtained cutting plans are also used to arrange the sequence of the cutting stock materials.
- Despite being an important concern in industrial applications, the incorporation of different quality levels in CSP has taken very little interest in the literature. One of the reason is that the traditional approaches are employed in real life CSP. In traditional approaches, set of orders are chosen manually and related cutting plans are created subsequently. The quality level of materials is determined by the operator (or sometimes it is ignored), and quality levels are not used in generating the cutting plans. Therefore, industries endure extra cost of overgrading. Overgrading arises when supplying customer order with materials which have higher quality level than specified by the customer. In the

proposed model, different quality levels of customer orders and cutting stock materials are considered.

- In the literature, most of the studies in CSP concentrate on minimizing the amount of required raw materials which is equivalent to minimizing trim-loss. However, in real life CSP, different cost components such as transportation cost, overgrading cost, and etc. are as important as raw material cost. For this reason, the proposed model minimizes transportation cost and overgrading cost simultaneously. The objective factors of this model are expressed in the subsequent sections.
- In the literature, most of the studies in CSP include limited and standard constraints such as guillotine constraints and 90-degree rotation constraints. However, the proposed model has been developed for solving real life problems with unique characteristics. Hence many practical constraints related to raw material and/or cutting processes are incorporated into the model. Some of the raw material constraints are width of material, quality of material, type of customer demand etc. Some of the cutting process constraints are capacity of cutting machine, special constraints for the cutting machine dimensions, and the constraints for knife portion in cutting process etc.
- In the literature, most of the studies related with the cutting and sequencing problems have been handled independently. Conducted literature review has revealed that there are a few studies in which both of the problems are combined. Typically, these problems are solved separately. In order to fulfill the final product demand, it is necessary to determine cutting plan of each stock materials. Therefore, firstly CSP is solved, and then sequences of the stock materials are determined. However, the proposed integrated solution approach by considering aforementioned costs and constraints find the trade-off between objectives. For example, if manufacturer expect high amount of demand for some final products, in this case, the storage costs may increase. However, the trim-loss amount and setup costs are reduced due to the fact that better cutting patterns are generated. Therefore, the CSP and sequencing problem should be considered jointly. The main endeavor of this study is to specifically develop SDS algorithm so as to solve the one-dimensional marble

plane cutting problem with usable leftovers by taking into account sequence dependent setup costs.

The proposed model has unique properties which obstacle to use traditional algorithms. Since a real life one-dimensional marble plane cutting problem is considered, a novel MILP model is developed in order to solve the problem.

1.4 Outline of the Thesis

There are six chapters in this thesis. In chapter two, literature review of the CSP problems, in terms of solution approaches, general concepts and typologies, are given. In chapter three, marble plane cutting problem is introduced and developed mathematical model is explained in detail. Developed mathematical model is applied to real problem instances. In chapter four, constructive heuristic method based on SDS algorithm is developed for large-sized problems. In this chapter, developed algorithm is explained explicitly and is applied on real problem instances in order to show the applicability of the developed approach. Results have shown that the proposed model is able to solve the realistic problems in a reasonable time which can be a useful tool in industrial settings. Moreover, the proposed model can enhance the overall system efficiency significantly compared to current industry practice. In chapter five, sequence dependent setup cost is considered within the developed model. Also, aforementioned real problem instances are solved by considering knife dependent setup costs. In chapter six, solutions of the proposed model are evaluated and some suggestions and future research directions are given.

CHAPTER TWO
CUTTING AND PACKING PROBLEMS

2.1 Definition of Cutting and Packing Problems

CSP depends on filling the customer demand with minimum cost by cutting small parts from specified lengths of large material or generating cutting plans by placing the customer order into the stock materials. It is not always possible to cut the stock material without leftovers. These leftovers parts are named ‘retail’ if their lengths are long enough. Otherwise, leftovers are called ‘trim-loss’, ‘wastage’ or ‘scrap’.

In CSPs, when small items are being cut out of large material objects, typically two problems appear. One problem is called assortment problem which deals with choosing suitable dimensions for the large material objects. The next problem is the trim-loss problem which tries to minimize the scrap material when cutting out of the small items from the large material objects (Hinxman, 1980; Dyckhoff, 1990).

The core term “cutting patterns” (cutting plans) in C&P problems can be defined as the layout plans of small items on the large material objects. Since cutting patterns are combinatorial problems, the C&P can be said to belong to the field of geometric combinatorics in which problem types are similar to each other. Accordingly, C&P problem has many applications in real life situations. For instance, in the literature, the bin packing problem (BPP) can be seen as a 1DCSP. In this problem, the aim is to minimize the number of used bins that the objects are being packed into these bins. The cause of distinction between these two problem types is that the items are assigned with different types of assortment. Particularly, in the CSP, small items have identical lengths, whereas in the BPP there exist usually many objects which have different lengths. Also, we can see C&P problem structure in the literature within some known problems. For example, in assembly line balancing problem there exists work pieces (tasks) which are consecutively assigned to (packed into) work stations in the assembly line. Work pieces are moved in a certain sequence between stations. At each station, definite operations are performed by operator by

taking into account cycle time restriction. The aim is optimally solving the problem with respect to some objective. Therefore, if the individual tasks considered as the small items and the stations taken as the large objects, then it demonstrates similar structure with C&P. Researchers have proposed many different solutions to these problems, since hundreds of problem types exist. All of these requires classification which obviously sets the differences between such problems. As a result, classification framework for the CSP was presented and structured overview with examples are given by Dyckhoff (1990) as in the Figure 2.1.

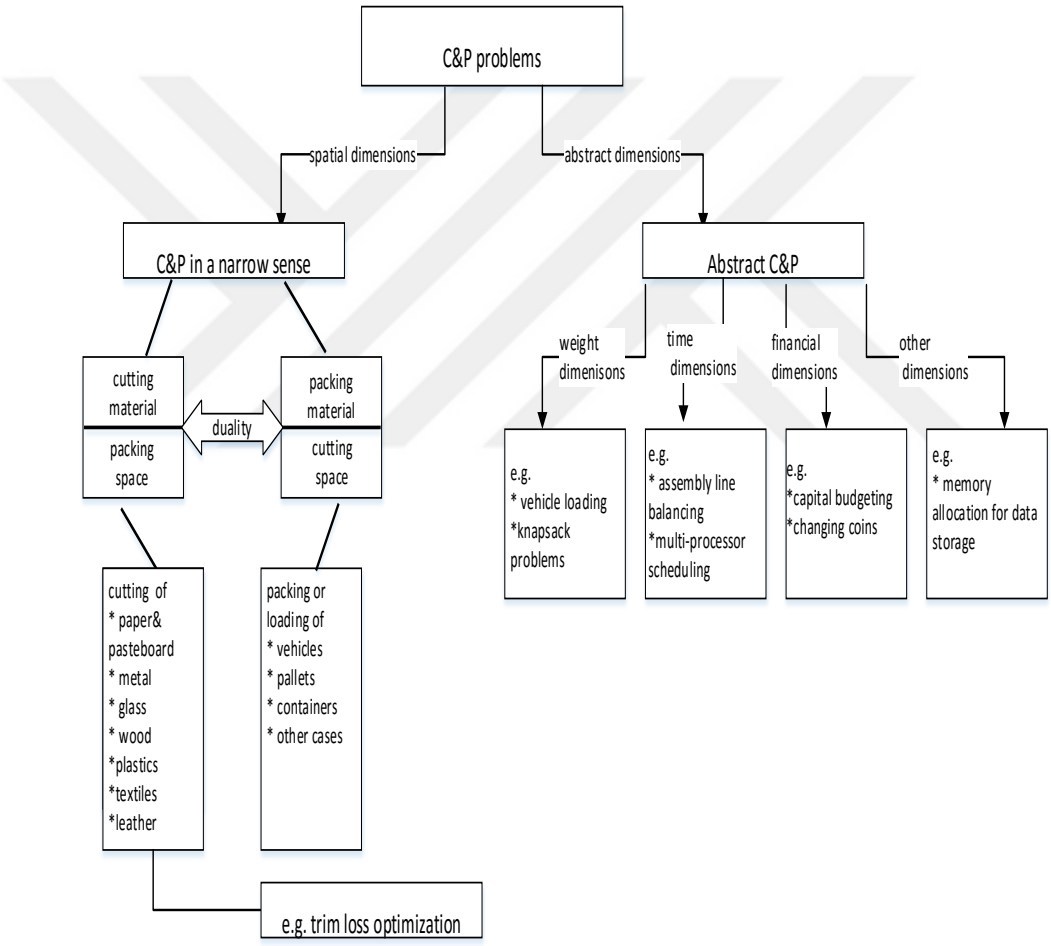


Figure 2.1 Phenomena of cutting and packing (Dyckhoff, 1990; Cheng et al.,1994)

C&P problems are splitted into two classes according to three dimensions of the Euclidian space, and other dimensions, namely spatial dimensions and abstract dimensions. Spatial dimensions include CSPs and packing problems. In the CSPs, stock objects must be divided into smaller customer order items in contrast with

packing problems in which small items must be compounded into large objects. Because of the essential similarities between these two problems, they can be concerned under duality framework. Abstract dimensions are divided into four different classes according to weight dimensions, time dimensions, financial dimensions and other dimensions.

2.2 Characteristic of Cutting and Packing Problems

Main characteristics of C&P problems are determined carefully when defining the types of problems or solution approaches. These attributes are the logical structure based and reality-based.

2.2.1 Characteristic Based on Logical Structure

Logical structure categories concern with the following attributes: Dimensionality, assignment type, characteristics of stock material objects, characteristics of small customer order items, pattern restrictions, objectives, knowledge of information and data variability and solution methods.

2.2.1.1 Dimensionality

The most significant characteristic is the dimensionality which can be defined as the minimum real numbers to express the necessary dimensions of the cutting plans geometry. In the C&P, problems can be one-dimensional, two dimensional, three dimensional and multi-dimensional. 1DCSP example can be cutting steel bars from the fixed lengths of the stock bars in which the only one dimension of the stock is taken into account in the solution. In two dimensional problems the customer order items can be placed on the stock material according to the two dimensions. Basically, this case occurs when both the stock material sheet and the customer order items are in the rectangular form. However, besides aforementioned dimensions, there exist other problem dimension which is called 1.5-dimensional problems (Hinxman, 1980; Haessler et al., 1991). In this problem, dimension complexity changes between one

and two dimension where the stock material has one fixed and one variable dimension. For instance, consider the cutting steel bars case which is given in one-dimensional CSP example. In this case, width of the stock reel may be variable while the length of the stock reel is fixed.

2.2.1.2 Type of Assignment

In the C&P process assignment of items to objects arrangement according to four different type, namely Type I, Type II, Type III, Type IV. In Type I (Pure layout problems) case all of the objects and items must be assigned. In Type II (Beladeprobleme) each large object is assigned to selection of items. In Type III (Verladeprobleme) each small items must be assigned to selection of objects. In Type IV (Ladeprobleme) each items and objects are selected and assignment is realized so that unused objects and items may remain.

2.2.1.3 Characteristics of Large Objects and Small Items

Characteristics of large material objects and small items can examine from four perspectives which are type of quantity measurement, figure (shape), assortment and availability.

2.2.1.3.1 Type of Quantity Measurement. The number of objects or items in C&P problems is measured by discrete (or integer) or continuous (or fractional) variable. Firstly, some values such as the run-lengths, objects and/or items number can be considered in discrete measurement. Secondly, in continuous quantity measurement customer orders of same types are collected into a single demand with regard to relevant dimensions. Also, the length of the objects or items are being summed up with respect to a further dimension that is not essential to the geometry of the patterns (Gilmore, 1979). Continuous quantity measurement is similar to two-dimensional case which is generally assumed that the one dimension is fixed and items are cut according to other dimension.

These one-dimensional problems with continuous measurements are sometimes called 1.5-dimensional problems.

2.2.1.3.2 Figure (shape). The figure (shape) of an object or item can be taken identical if its form, size and orientation are same within the relevant dimension. In the literature, numerous possible forms of objects and items such as regular (standard) shapes and irregular (non-standard) shapes can be found. Orientation affects the form of an objects or items. Orientation is taken as fixed or variable. If rotation is possible, C&P problems differ from each other. 90-degree rotation is the most commonly employed in the literature.

2.2.1.3.3 Assortment. Assortment is defined by using figure characteristic and number of permitted figures. Assortment of objects and items can be classified according to homogeneous and heterogeneous types of two different possibilities. In homogeneous assortments all objects or all items have the same form and size but orientation may be different. In heterogeneous assortments, the objects or items may have different size, form, and orientation.

2.2.1.3.4 Availability. Availability is associated with quantity, sequence and date attributes. Quantity of objects and items stored in stock can be limited or non-limited or some types may be limited only. These properties create different types of problems. In some C&P problems, objects are necessarily cut in predetermined sequence order. The reason may be machine eligibility or assembly line constraints. Date is also considered under availability criteria when items have different delivery times.

2.2.1.4 Pattern Restrictions

Pattern restrictions refer to technical and organizational restrictions based on geometric characteristics, operational characteristics and allocation dynamics.

2.2.1.4.1 Geometric Characteristics. Geometric characteristics such as distancing restrictions, frequency limitations, separation restrictions should be considered when designing the pattern. Distancing restrictions are the most important one. Because when cutting items from objects, there may be left some distance between items. For example, when cutting marble blocks, knife thickness is also considered. Frequency limitations mean that how many times each item is counted in pattern design. In some situation, the number of items in a pattern may be limited. For example, in C&P problems with pattern sequencing constraints, if there are much items in one pattern, and there are less items in another pattern, it may result in high pattern changeover cost. Separation restrictions are related with arrangement for small items in a pattern. It is possible to cut items orthogonal or non-orthogonal way in cutting pattern. In orthogonal way of cutting, items are placed parallel to the stock object's edges in cutting pattern. However, in non-orthogonal way of cutting, there is no restriction on the selection of cutting angle. Example of orthogonal pattern and non-orthogonal pattern are given in Figure 2.2.

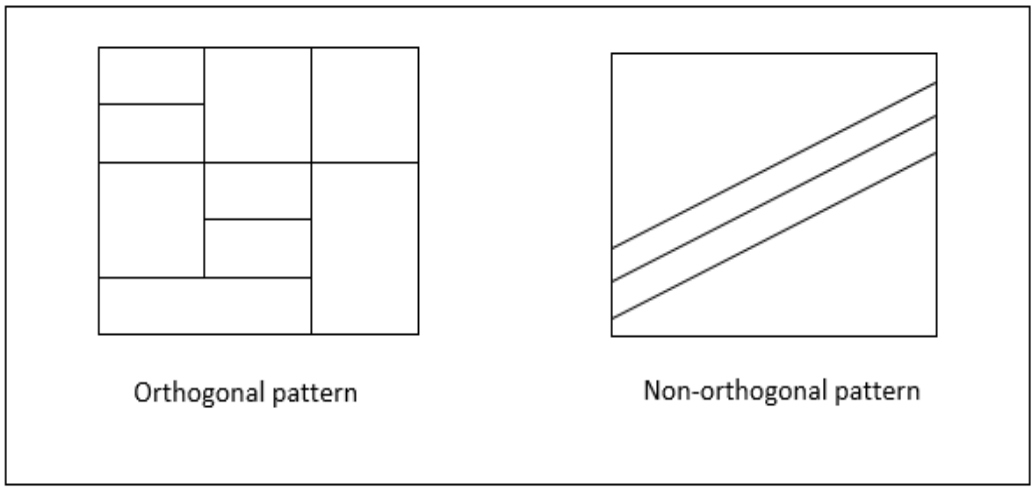


Figure 2.2 Example of orthogonal and non-orthogonal patterns

Another type of cutting pattern in cutting problems can be guillotine pattern or nested pattern. In guillotine pattern, cuttings are structured from one end point of the object to the other points uninterruptedly. Example of guillotine pattern is given in following Figure 2.3.

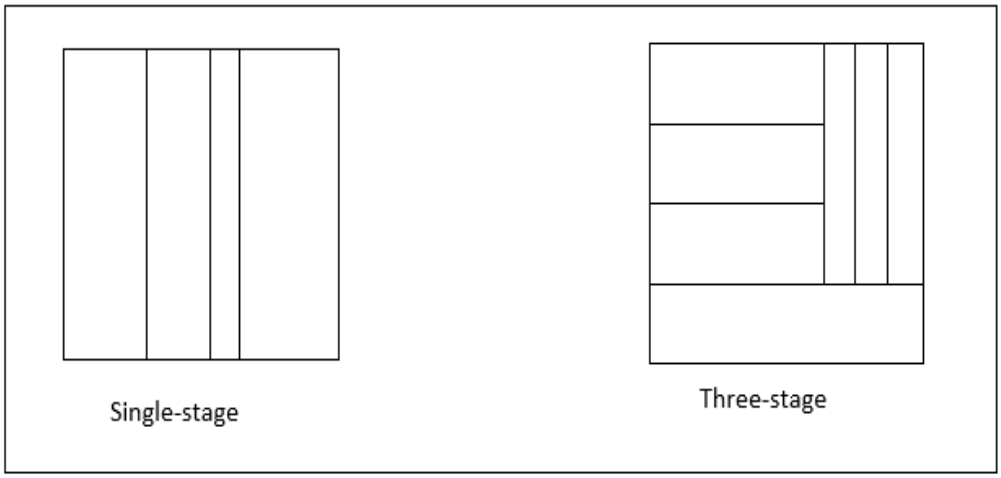


Figure 2.3 Examples of guillotine patterns in C&P problems

In the nested pattern, interrupted cuts can be achieved at any point. Example of nested pattern is given in Figure 2.4.

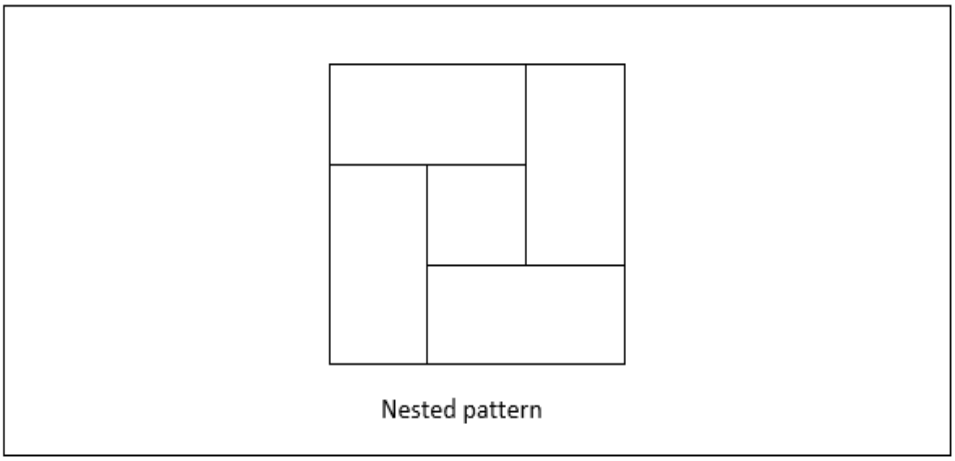


Figure 2.4 Example of nested pattern

2.2.1.4.2 Operational Characteristics. Number of stages, pattern sequence and pattern frequency are operational characteristics in C&P problems. Continuous operation in each stage of C&P problems, a cutting task is performed independently in a work cycle. We can assume that in single-stage process items cut in single, continuous operation whereas small interruptions are negligible. In other words, it is accepted that all small items are cutting at the same time. However, in multi-stage process of C&P problems, there are discontinuances between operations.

For example, two dimensional CSPs in marble industry are performed two separate operations. Firstly, marble blocks are cut in gangsaw machine to get marble planes, and secondly marble planes are cut in bridge cutter to obtain smaller marble slab products (strips, stairs, tiles, etc.) of different dimensions. One must also take into account the capacity for buffer between the stages.

In C&P process, the pattern sequence process is generally considered as separate planning process. However, for several factors such as finding optimal solution instead of finding sub-optimal one, it is necessary to develop cutting pattern and sequencing pattern simultaneously.

2.2.1.4.3 Allocation Dynamics. C&P problems are generally studied as static problems. In reality, there are some situations (such as deadlines or machine failures) which prevent from delivering customer order in the same period of time.

2.2.1.5 Objectives

There are many objectives related to problem in C&P planning. Since C&P occurs in real life in many forms so that many objectives can be listed in the literature. Some of the objectives which are taken from the literature can be listed as follows:

- Input minimization
- Trim-loss minimization
- Material cost minimization
- Change-over cost minimization
- Setup cost minimization
- Inventory cost minimization
- Overgrading minimization
- Transportation cost minimization
- Profit maximization

2.2.1.6 Status of Information and Variability of Data

Since the type and quantity of items are known before the cutting planning horizon, C&P problem data is generally assumed to be deterministic. However, in many practical settings, the cutting operations are part of a dynamic manufacturing environment where the customer orders arrive randomly over time and have to be completed by a specified due date. In this case, status of information of problem data can be stochastic or uncertain.

2.2.1.7 Solution Methods

C&P problem solution approaches can be divided into two categories. These are object-oriented methods and pattern-oriented methods.

In object-oriented methods, the small items are individually assigned to the large objects. These methods are most applicable when there are many small, variously-shaped items. Pattern-oriented methods involve constructing diverse feasible patterns, and then choosing the best ones for assignment. This category involves different kind of versions. For example, single-pattern methods produce one pattern after another and immediately select an appropriate assignment of objects and items. On the contrary, multiple-pattern methods produce several patterns and simultaneously assignments are performed with regard the possible interaction between the patterns.

2.2.2 Reality-Based Characteristics

Although the reality-based categories have many attributes which lead to hassle in presenting classification, however based on some assumption reality-based categories concern the following attributes: Kind of objects and items, branch of industry, planning context and software.

2.2.2.1 Kind of Objects and Items, and Branch of Industry

In C&P problems, objects and items depend on the industrial sector in which they are applied. For example, cutting marble blocks can serve the production of marble planes or marble slabs, whereas pallet loading can be taken belonging to service industry.

2.2.2.2 Planning Context

C&P planning problems can be seen as a part of production planning problems such as:

- order management
- timing
- capacity planning
- sequencing
- choice of manufacturing method
- numerical (machine) control
- inventory planning
- material requirements planning
- procurement planning

These planning areas are generally planned simultaneously. However, sometimes it cannot be possible for some situations. So following two different procedures are applied to cover these situations:

- All areas are planned successively, where solutions of previous problems provide restrictions for further planning. In this procedure the cutting process is planned separately.
- Two or more areas of planning are integrated into a simultaneous planning process, while the other areas are planned successively.

2.2.2.3 Software

Software can be developed for the identification of the C&P problems. The aim can be evaluating the applicability of the developed model or comparing the performance by testing algorithms.

Reality-based characteristics are shown in the Table 2.1. Also, an overview of characteristics based on logical structure and their properties are shown in the Table 2.2.

Table 2.1 Reality-based characteristic (Dyckhoff, 1990)

Attributes	Properties
kind of objects and items and branch of industry	-kind of large objects -kind of small items -branch of industry
planning context	-order management -timing -capacity planning -sequencing -choice of manufacturing method -numerical control -inventory planning -material requirement planning -procurement planning -others
software	-practical application -simulation

Table 2.2 Characteristic based on logical structure (Dyckhoff, 1990)

attributes	properties	
dimensionality	-one-dimensional -three-dimensional -1.5-dimensional -multi-dimensional -two-dimensional	
type of assignment	-Type I (all objects, all items) -Type II (all objects, selection of items) -Type III (selection of objects, all items) -Type IV (selection of objects, selection of items)	
characteristics of small items and large objects	**(1) type of quantity measurement** - discrete - continuous	**(3) assortment** -homogeneous (extreme: identical) . . fixed . . variable -heterogeneous . . fixed . . variable . . few objects per figure . . many objects per figure
	(2) figure (shape) -form . . rectangular . . non-rectangular -orientation . . fixed . . optional . . 90^0 -turns (parallel) . . 90^0 -turns (optional	**(4) availability** -number per figure . . limited (extreme:1) . . unlimited . . mixed -sequence . . no order . . partial order . . complete order -dates . . identical time references . . different time references
pattern restrictions	**(1) geometric characteristics** -distancing restrictions . . between items . . to the edges -frequency limitations . . limited combinations of figures . . limited number of figures . . limited number of items -separation restrictions . . non-orthogonal patterns . . nested patterns . . guillotine patterns with fixed number of stages . . guillotine patterns with unlimited number of stages	**(2) operational characteristics** -number of stages . . single-stage . . multi-stage -pattern sequence . . optional . . restricted -pattern frequency . . limited number of different patterns . . different time references
		(3) dynamics of allocation -dynamics -static . . off-line . . on-line
objectives	-input minimization -(material/space) cost minimization -trim-loss minimization (absolute) -change-over cost minimization -trim-loss minimization (relative) -inventory cost minimization -value maximization -others	
Status of information/variability	**(1) status of information** - deterministic -stochastic -uncertain	**(2) variability of data** -fixed data -variable data
Solution methods	-object-orientated methods -pattern-orientated methods . . single-patterned . . multi-patterned	

2.3 Cutting and Packing Problem Classifications

Previously mentioned characteristics can be aggregated into four fundamental attributes that allow the classification of C&P problems. This classification is given in the Table 2.3. The appropriate notations are given in brackets.

Table 2.3 Classification of Cutting and Packing Problems (Dyckhoff, 1990)

Characteristic	Symbol	Definition
(α) Dimensionality	1 2 3 N	One-dimensional Two-dimensional Three-dimensional N-dimensional (N>3)
(β) Kind of assignment	B V	All objects and a selection of items A selection of objects and all items
(γ) Assortment of large items	O I D	One object Identical figure Different figures
(δ) Assortment of small items	F M R C	Few items Many items of many different figures Many items of relatively few different figures Congruent figures

According to this classification 4x2x3x4=96 different C&P problem can be defined and these problems can be represented with the notation α/ β/ γ/ δ.

According to assignment type and customer order item characteristics, large stock objects assortment C&P problem is divided into four general classes. These are:

- Bin packing (BP)
- Cutting stock (CS)
- Knapsack (KS)
- Pallet loading (PL)

General properties of this classes according to type of assignment and assortment are shown in Table 2.4.

Table 2.4 C&P problem classes according to assortment and assignment

			Type of assignment	
			Type II	**Type III or Type IV**
Assortment	**Homogeneous**		PL	
	Heterogeneous	Few objects per figure	KS	BP
		Many objects per figure	KS	CS

Dimensionality constitute 12 different C&P problem classes which are shown in Table 2.5

Table 2.5 C&P problem classes additional dimensionality characteristic to assortment and assignment

	BP	CS	KS	PL
One-dimensional	BP1	CS1	KS1	PL1
Two-dimensional	BP2	CS2	KS2	PL2
Three-dimensional	BP3	CS3	KS3	PL3

The presented C&P problem classes can be shown with Dyckhoff classification in the following notations:

Classic BP Problem: 1/V/I/M

Classic CS Problem: 1/V/I/R

General CS Problem: 1-2-3/ / /

Classic KS Problem: 1/B/O/ /

Classic PL Problem: 2/B/O/C

Note that, problem classes can be different from above classes with their special characteristics.

Classification which was proposed by Dyckhoff has been criticized for some angles and some expansion is presented. Gradišar et al. (2002) proposed additional classes in the large items assortment group of Dyckhoff ’s classification and in the following related notations are given:

- G: a few groups of identical large objects.

Further improved typology is added by Trkman and Gradišar (2007) in order to fill the inadequacy of the Dyckhoff classification to distinguish between consecutive and instantaneous solutions of CSPs. They called this new class as the time-span of optimization and places a variety of three options:

- IN: (Instantaneous) shows the given data which is used in only one period of time for the optimization/solution of cutting/packing problem.
- CD: (Consecutive Deterministic) used optimization for problem types when the all demand and supply of material number is known at least for two or more consecutive time periods. Also unused material can be used later time period. A particular sub-type of CD type is an example where some predicted demand lengths from an early time period could also be held until they appear as demanded length in a later time period.
- CS: (Consecutive Stochastic) shows the data is known certainly only in the first period and in the following periods it is estimated by using the probability distributions and in this type of problem unused material can be used in the later time period as a reused material.

The abovementioned developments create a new important aspect for the CSPs. The one is involvement of time periods in the cutting CSPs increase the importance of using leftover materials from the previous periods into the next periods. The

second one is taken into account stochastic demands instead of exactly known values.

Recently, papers which are published in C&P problem research areas have raised year by year. Accordingly, some shortcomings emerge related with generality concepts of Dyckhoff's classification. Wäscher et al. (2007) defined the shortcomings in the following ways:

- It cannot be possible to classify all C&P problem types with Dyckhoff classification scheme,
- Classification scheme of Dyckhoff may give inconsistent results. The reason is that same problem simultaneously belongs to various classes,
- All classes have not included identical problems.

Therefore, Wäscher et al. (2007) presented a new typology for the C&P problems in which the following classification was used:

1. Dimensionality:
   - One-dimensional
   - Two-dimensional
   - N-dimensional with N>3

2. Problem type:
   - Output maximization
   - Input maximization

3. Customer order length distribution:
   - The same order lengths
   - Weak heterogeneity of customer order length distribution
   - Strong heterogeneity of customer order length distribution

4. Distribution of items lengths:

- One large object
- Many large objects

5. The shape of order objects:

- Usual shape
- Unusual shape

2.4 Classic Cutting and Packing Problems with Mathematical Notations

In this subchapter, standard and general 1DCSP definitions are given.

2.4.1 General 1DCSP

One of the key factors is trim-loss notion in the solution of general 1DCSP. Gradišar et all. (2002) considered following assumptions when solving the general 1DCSP:

- There is s adequate number of stock materials to meet every demand,
- Amount of customer order pieces being cut from stock materials with required order lengths is equal to exactly customer demand amount,
- The lack of stock material is not allowed (since it is assumed that the requested stock material is supplied in case of shortage),
- The lengths are assumed to be integer numbers or they are transformed into integer numbers by multiplying it a factor.

Used decision variables and notations of the parameters in the proposed mathematical formulation are given as follows:

l_i : Lengths of customer order pieces, i=1, … , n,
d_i : Number of customer order pieces that demanded with length l_i,
U_j : Lengths of stock material, j=1, … , m,

$p_{i,j}$: Amount of customer order pieces with length l_i that having cut from stock material with length j.

z_j : Binary decision variable which take value 1 if stock material with length j is being cut in cutting plane and otherwise it takes value 0.

$$\begin{cases} 1, if\ p_{i,j} = 0 \\ 0, otherwise \end{cases} \quad \forall i$$

w_j: Binary decision variable which take value 1 if remainder of stock length j is residual and otherwise it takes value 0.

$$\begin{cases} 1, if\ z_j = 1\ and\ \delta_j > \max l_i \\ 0, otherwise \end{cases}$$

t_j: Trim-loss amount which is related to stock length j

$$\begin{cases} \delta_j, if\ z_j = 1\ and\ \delta_j \leq \max l_i \\ 0, otherwise \end{cases}$$

Formulation of integer programming model:

$$\min \sum_{j=1}^{m} t_j \text{ (minimize trim-loss)} \tag{2.1}$$

Subject to:

$$U_j - \sum_{i=1}^{n} l_i\, p_{i,j} = \delta_j \quad \forall j \quad \text{(knapsack constraint)}, \tag{2.2}$$

$$\sum_{i=1}^{m} p_{i,j} = d_i \quad \forall i \quad \text{(demand constraint)}, \tag{2.3}$$

$$\sum_{i=1}^{m} w_j \leq 1 \quad \forall i \quad \text{(number of residual constraint)}, \tag{2.4}$$

$$p_{i,j} \geq 0, \quad \forall i, j, \text{ integer}, \tag{2.5}$$

$$t_j \geq 0, \ \forall j, \tag{2.6}$$

$$\delta_j \geq 0, \ \forall j, \ \text{integer}, \tag{2.7}$$

In this model formulation, remainder material after cutting process is considered residual if it is greater than $\max l_i$ otherwise it is considered as trim-loss. Also, Gradišar et all. (2002) indicated that the importance of number of residual constraint must be equal to one. The reason is that if more than one residual is left, then they pile up in the stock area. Furthermore, it is possible to assume that more than one residual materials can be combined into one piece (Gradišar et al., 1997).

2.4.2 Standard 1DCSP

Standard 1DCSP was described by Gau and Wäscher (1995). Used notations of the decision variables and parameters in the proposed mathematical formulation are given as follows:

l_i : Lengths of customer order, i=1, … , n,
d_i : Number of customer order items that demanded with length l_i,
L_k : Number of different stock material lengths, k=1, … , p,
c: Indicate the cutting pattern when having cut from stock material with length k,
t_k: Number of cutting pattern which having cut from stock material with length k,
a_{ick}: Number of times that order with length l_i appears in a cutting pattern c,
x_{ck}: Frequency that cutting pattern c cut from stock length k.

Proposed integer programming mathematical model formulation is:

$$\text{Min } \sum_{k=1}^{p} \sum_{c=1}^{t_k} x_{ck} L_k \text{ (Minimize total cutting stock lengths)} \quad (2.8)$$

Subject to:

$$\sum_{k=1}^{p} \sum_{c=1}^{t_k} a_{ick} x_{ck} \geq d_i \,, \forall i \quad (2.9)$$

$$\sum_{i=1}^{n} l_i a_{ick} \leq L_k, \quad \forall c, k \quad (2.10)$$

$$x_{ck} \geq 0, \quad \text{integer } \forall c, k \quad (2.11)$$

$$a_{ick} \geq 0, \quad \text{integer } \forall i, c, k. \quad (2.12)$$

There are two main differences between the standard 1DCSP and general 1DCSP. First difference is objective of problems expressly general 1DCSP aim is to minimize trim-loss but aim of standard 1DCSP is to minimize the total stock lengths to be cut. Second difference is that solution method of the problem in the general 1DCSP is item-oriented, and pattern-oriented approaches are more suitable for the standard 1DCSP.

In practice, some problems cannot be solved via using both pattern-oriented or item-oriented approaches. For these reason, Gradišar et al. (1999) combined these approaches into one approach which was named a hybrid 1DCSP for the following aims:

- To cut customer order with required amount of demand,
- To aggregate residual material into one item for reusing it in the following planning period,
- To use non-standard stock materials for aiming to improve system efficiency.

2.5 Cutting and Packing Problem Complexity and Solution Approaches

CSPs are the member of NP (Non-polynomial) problem class. NP-problems generate solutions within polynomial time by using nondeterministic algorithms. According to Bischoff and Wäscher (1995) majority of cutting problems belong to NP-complete problem class. In NP-complete problem, solution time increases with respect to size of the problem and hence it is not convenient large-sized problems. Therefore, Bischoff and Wäscher (1995) proposed different types of solution approaches which considering different characteristics such as practicality of research, the complexity and the diversity of the real-life problems. Based on the complexity of C&P problem solution approaches split up two groups which are exact approaches and heuristic approaches. Exact method approaches provide the optimum solution but are only useful for small-sized problems since computational complexity are exponential. On the other hand, heuristic methods do not give guarantee of the optimality of the solution result but they converge to the nearly optimal solution in a polynomial time.

In practice, efficiently managing companies solve their problems in a short time. Hereby, time is very fatal criteria to measure the efficiency of business. According to Erjavec et al. (2011), companies expect to obtain solution in a short time with lowest possible costs. However, if this case not possible they tend to prefer the solution that produce much scrap in a short time duration than solution that produce less scrap in a

long time duration. For that reason, heuristic approaches are mostly preferred in real world problems.

Dyckhoff (1990) divided solution procedure of the 1DCSP into two group namely pattern-oriented methods and item-oriented methods. In pattern-oriented methods, firstly patterns constructed then customer order assign some of these patterns. Plenty of pattern-oriented approaches are rooted from developed algorithm by Gilmore and Gomory (1963). If stock materials have identical lengths or limited number of them have different lengths than the others, in this case pattern-oriented approaches are useful. On the contrary, item-oriented approaches are applied when there exist stock materials that have both different lengths or identical lengths. In this method, there is no need to prepare the cutting pattern before, since methods promptly assign items to objects.

Trkman and Gradišar (2007) emphasized the importance of using convenient methods. According to them, standard stock lengths match with pattern-oriented methods, and non-standard stock lengths match with item-oriented methods.

Classification of solution approaches according to some properties is given in Table 2.6.

Table 2.6 Classification of solution approaches (Dyckhoff,1990)

Item-oriented		**Pattern-oriented**	
Exact methods	Approximation algorithms	One-pattern	Several patterns
E.g branch and bound, dynamic programming	Bin packing algorithms	Knapsack algorithms, various methods in more dimensions	Linear programming-based and general heuristics

Pattern-oriented approach has a negative side of finding the optimal solution, since large number of possible cutting patterns occur for having cut each small items. For instance, if there exist 40 different customer order and each stock material which

stands in stock area cut into 4 identical or different item lengths. Then the all possible pattern numbers are computed via using combination formula for repetition case as given in Equation 2.13:

$$C^{r}_{n+r-1} = \binom{n+r-1}{r} \tag{2.13}$$

In this formula, n represents the number of customer order which has different lengths and r indicates the number of stock material lengths which are having cut to obtain an individual item. Hereby using above formula, it is calculated 123.410 different patterns are needed for an individual item. All potential number of cutting pattern combinations are going to be raised when using more than one small item. As a result, except small case problem instances, all methods which testing the all enumerated alternatives are going to be failed to find optimal solution (Trkman & Gradisãr, 2007).

Due to the abovementioned bases, when using the pattern-oriented approach for large problem instances, some of the cutting patterns are selected. The vital points for method's performance is related to finding the appropriate patterns which will be selected. By using different approaches, convenient cutting patterns are obtained, then decision has made about the how many times each of selected patterns are used (Trkman & Gradisãr, 2007).

When item-oriented approach based methods are applied, all cutting processes are considered separately. Cutting plan is dynamically determined for each cutting item. Mostly it is preferable to use this approach in case of all stock material lengths are not identical, which is commonly encountered in real life applications. However, there can be some cases where all stock materials have same lengths, and also using item-oriented approach is appropriate (Trkman & Gradisãr, 2007).

Both heuristic and exact methods can handle solving item-oriented and pattern-oriented approaches. Exact methods have some advantages and disadvantages. The biggest advantage, it gives exactly optimum solution after executed limited number

of algorithm steps. On the contrary, the biggest disadvantage is that the most of the exact methods require exponential time to obtain optimum solution in large-scaled problems. Some of the most known exact method examples are dynamic programming, branch and bound approach, and linear programming (LP) etc. (Trkman & Gradisãr, 2007).

The heuristic approaches are solution methodologies which obtain solutions with small deviations from optimal solutions. Mostly, they are preferred when solving large-sized problems. Also, they are useful for some algorithms where optimal solution has not known yet or it takes much time to get optimal solution (Eiselt & Sandblom, 2000).

Even heuristic methods are mostly useful in the abovementioned problems, however they have disadvantages. One of the disadvantage is that they are generally problem dependent and are developed specifically for the problem in hand. The reason is that developed heuristic approach gives efficient solutions to considered problem under certain specific aims and constraints which is not going to be proper for other problem types, even they are obviously similar to each other (Trkman & Gradisãr, 2007).

Sometimes combined approaches of exact and heuristic methods provide better solutions. For instance, Gradisãr and Trkman (2005) integrated two approaches branch and bound algorithm and sequential heuristic procedure to produce a new approach. In this method, firstly, using the sequential heuristic method the large-sized pieces of the problem are solved. Second, smaller sub-problems are being solved using the exact method. This combined method gives almost optimal solutions in a lower time duration. The result shows that combined method gives better results than the single approach.

2.6 Literature Review of Existing Algorithms

In this subchapter some related literature is given to the best known methods for 1DCSP and 1DCSPUL.

Gilmore and Gomory (1961) was developed the first efficient heuristic algorithm instead of the LP method by aiming to accomplish the obstacle. This heuristic method gets solution using a sampling approach. The objective is same as the general case which is to maintain a suitable cutting pattern with least total cost value. Total cost has gained by adding related cutting cost value from given stock material. It is assumed that there exists infinite amount of material in stock that each of them has identical length. Since there are enough stock materials, all customer order is satisfied in each situations except the case of if any dimension of customer order is longer than any dimension for all stock material. As mentioned before, mostly encountered problem is the number of high cutting plans. In the proposed approach, it is prevented from solving knapsack problem. In this problem if a new pattern is needed then only relevant patterns are looked by solving an auxiliary problem, instead of all possible patterns within each phase of the simplex method.

Vahrenkamp (1996) developed a randomized approach to solve the 1DCSP that only contain efficient patterns with the smaller trim-loss value than some predetermined threshold. The quality of results is measured by the cross over method from genetic algorithms with discussed the multiple occurrence of equal solutions in different patterns.

Genetic algorithms (GA) can be also implemented to the 1DCSP. For instance, Ragsdale and Zobel (2004) introduced a combination metaheuristic approach for solving the problem. In this method GA combined with other heuristic methods for aiming fasten the cutting process by cutting each order sequentially.

Another example of genetic algorithm was applied can be seen in the construction industry (Shahin & Salem, 2004). Since traditional optimization methods were

inadequate. In this study, problem instances that taken from a local steel workshop are tested and their results show that, proposed method achieve considerable savings.

Sardiñas et al. (2006) proposed an optimization technique which shows the benefits of multi-objective optimization. The aim of the considered problem was to find optimal cutting parameters of turning processes under operation time and material tool life objectives. The proposed model was used a genetic based algorithm.

Genetic symbiotic algorithm was presented for 1DCSP which tried to minimize setups and amount of used stock materials. Proposed algorithm simultaneously produce cutting plans and determine each of cutting plans frequency (Golfeto et al., 2009).

Goulimis (1990) proposed a method that obtains an optimal solution for the 1DCSP. In this method, integer program which is integration of two approaches branch-and-bound algorithm and cutting plans was solved. The program depends on counting all possible cutting patterns. Efficiency of the method was tested on one-dimensional real life CSP with using large board mill data. The solutions indicated that average savings of production were up to 2.5% better than previously used computer programs. Besides, in the trim-loss minimization, blade setups between changing cutting pattern and number of cutting plans were significantly reduced.

The modified integer round-up property (MIRUP) is valid when optimum value of all instances of the linear integer minimization problem (IP) is smaller than corresponding optimum value of LP relaxation problem, and this property is illustrated for the 1DCSP. A new heuristic approach was tested on the randomly generated problem instances and good solutions were obtained (Scheithauer & Terno, 1995).

Arbib et al. (2016) considered 1DCSP with pattern sequence constraints. In this problem, the number of no more than different types can have produced at any time

(open stacks). The objective of the proposed problem was trim-loss minimization. In literature, considered problem known as one-dimensional cutting stock problem with a limited number of open stacks problem (1-CS-LOSP). Solutions were obtained by developing a novel integer programming formulation. Results were made by linear relaxation with standard column generation procedure.

Sinuany-Stern and Weiner (1994) considered 1DCSP with two objectives for metal workshop. In this problem, first objective was to minimize the trim-loss amount in a given piece of metal work. The second objective was to generate a cutting plan that collected the biggest leftovers in the last bar(s). An algorithm was developed which was provided the optimum solution for small-sized problems. Therefore, a heuristic approach was developed for large-sized problems.

For the purposes of minimizing the total production costs in the fashion industry, a mixed integer programming (MIP) model was presented. The algorithm searches for an optimal range of cutting patterns which gives a combination of articles to be cut in one operation, and corresponding stack heights. Then it allocates a combination of pieces that need to be cut to those patterns (Degraeve, & Vandebroek, 1998).

Poldi and Araujo (2016) considered 1DCSP with multiperiod. Since, in many industries multiperiod cutting process can be seen in the production planning stage. In this case, there was a possibility to produce ordered items in the forward period or did not produce the items. Reuse situation was permitted. In other words, unused material in a one period can be used for the next periods. Extension of mixed integer optimization models from the literature was developed for multiperiod case. Solutions were obtained by using the residual heuristic. Results showed that most of the problem instances obtained satisfactorily solution.

CSP complexity of integer programming problem increases with the large number of cutting patterns. Suliman (2001) was introduced a new approach. In this approach, integer variables were taken as free variable then number of cutting pattern was decreased. Search tree based on pattern generation method was used.

Kim et al. (2016) discussed real-life cutting problem in window frame manufacture. In this problem, there exist four different types of bars that must be cut from aluminium profiles for satisfying each window frame order. The aim was to minimize weighted sum of trim-loss, bar type imbalance and the degree of order spreading. It was possible to assign bars to same raw profile if productivity increased. Moreover, number of different bar types which was provided from raw material profile must be balanced. The reason was, this case produce machine load imbalance. The proposed problem was solved via MIP model and knapsack-based heuristic approach. The results showed that, proposed algorithm gave better solutions than the company's approach. Therefore, considered firm now being used proposed algorithm.

Holthaus (2002) proposed a method for solving an integer 1DCSP that objective is minimizing the cost with taken into account multiple types of different standard stock lengths. Solution procedure was depending on the column-generation technique and integrated some known heuristic and exact methods.

In 2003, another approach was introduced. In this case, aim was saving total material cost by using two or more different stock lengths (Holthaus, 2003).

A noval model for 1DCSP with some practical constraints are introduced. In this method, aim was to show a novel use of binary variables for preventing the non-linearity between cutting plan variables and cutting plan frequencies. In this model pre-determined cutting patterns were not necessary. The resultant MIP and its application to four problem instances was presented. (Johnston, & Sadinlija, 2004).

A combined method of the exact and heuristic methods can obtain to approximately optimal solutions to the 1DCSP. In 2005, method for the 1DCSP with different stock lengths was introduced. It was integrated sequential heuristic procedure and the branch-and-bound approach. Firstly, the problem was solved by using heuristic method and good solution was founded. Then the smaller sized part

of the problem was solved with using exact method. This combined method gave good quality solutions for all sizes of the problem in a less time (Gradisãr & Trkman, 2005).

Tabu search approach was developed for solving 1DCSP with mixed objective function that was treated as similar to sequence problem. In this problem, mixed objective function was varied that considers total incentive trim-loss. The proposed method was performed on real data from shipyards and literatures. Computational results showed that better solutions were obtained than previously developed methods (Yang et al., 2006).

Belov and Scheithauer (2006), presented an approach that was able to solve the 1DCSP and a two-stage, two-dimensional guillotine constrained cutting problem (2D-2CP). They considered the combination of a branch-and-price scheme with cutting plans. The solution was primarily obtained for LP relaxation and later was improved with mixed-integer cuts. Finally, result showed that, general-purpose cut useful in 1DCSP. Also, for the 2D-2CP, combined approach was more effective than single approach.

For solution of the 1DCSP, a combined method of LP model and metaheuristics approach were presented by Umetani et al. (2006). The aim of the model was minimization of amount of stock materials. To increase the performance two types of local search, 1-add neighborhood and the shift neighborhood, were combined with LP technique. A sensitivity analysis technique was introduced to solve large-scale problems.

The problem of 1DCSP was considered with the setups and the maximum number of open stacks during the cutting process. Solution of the problem was handled by a sequential heuristic method. To evaluate the solution with regard to several objectives, the pareto criterion was used (Belov & Scheithauer, 2007).

There exist many studies related with pattern reduction in the 1DCSP. Another heuristic approach was developed by Cerqueria and Yanasse (2009). In this method, products were divided into two different class according to frequencies of customer order. Patterns were created with products of these classes. As a result, generated patterns with small scraps were acceptable.

Given due date in industrial cutting operations of the 1DCSP was observed by Reinertsen and Vossen (2010) when it was required to meet the orders before a given due dates. They proposed new optimization models and algorithms which considered customer order due dates. The developed approach was tested using data from large manufacturer of reinforcement steel company. Result showed that considered model was capable of solving the large-sized problems. Also, common cutting matters such as cutting of different item types in the same machine, existence of non-standard stock materials and the notion of customer orders aggregation were studied.

Generally main objective of 1DCSP is to minimize amount of trim-loss, however in reality it is not always possible. In 1990, Wäscher proposed a LP model that includes several conflicted objectives, such as costs of overgrading, stock material, warehouse and wastage (Wäscher, 1990).

LP model was developed for steel company. For technological causes, one cutting operation was not adequate. Therefore, two-stage cutting stock problem was necessary. Intermediate items were processed in the first stage to get necessary qualifications before cutting into final products in the second stage. The main objective of the problem was decrease the number of setups and trim-loss amount. Also in the presented method, amount of intermediate item was minimized. This LP model was solved via column generation technique (Valério de Carvalho, & Guimarães Rodrigues, 1995).

In 1995, a Swedish company operation research project was presented. The project struggled with satisfying the wood material demand from wooden panels. This problem was different than other existing problems in the literature. Since both wood materials and boards have different quality grade. The principal difficulty of

the problem was time limitations. In other words, cross cutting of boards should be made in 2 seconds before the wood material was sent to the cutting machine. A mathematical model was developed based on the set packing problem. Then some problem instances were solved with combined lagrangean relaxation method and subgradient optimization method (Rönnqvist, 1995).

In 1995, two exact approaches were developed for optimizing the sequence of cutting patterns. The objective was to minimize the maximum queue of cut orders in the sequencing stage. The first method was simple and was depend on the lower bound approach. The other method was based on an implicit exhaustive search procedure. In this work, remarkable effort spent on reducing the required patterns via graph theory approach (Yuen, & Richardson, 1995).

In 1998, a new mathematical model was developed for the purposes of solving the paper-converting industry problem. The problem was how to produce demanded paper reels from large raw paper stock reels with taking into account of minimization of cost objective. Considered problem generally is non-convex structure since objective function is bilinear which contain the cost of scrap, the stock cost of paper reels and knife changing cost in the machine. Solution procedure was based on two-step optimization approach. Firstly, all possible cutting plans were enumerated and later developed MILP model were solved (Westerlund et al., 1998).

Schilling and Georgiadis (2002) presented a new mathematical programming formulation for the paper industry. In this study, efficient cutting plans were determined. The aim was to maximize the total profit value which taking into the revenues from sales, the costs of the material, the costs of changing the cutting pattern, and the costs of the scrap material. To obtain solution of the considered problem, MILP model was developed and case problem instances from industry was tested.

Based on metaheuristics, Umetani et al. (2003) proposed a new combined method. In this model, aim was to obtain minimum cost. This goal was achieved by reducing

the different cutting pattern numbers. In model formulation, it was allowed to knife movements when it was required in pattern changing. Trim-loss cost was involved in objective function with lower cost than the pattern change cost. There was restriction on number of different cutting pattern and maximum trim-loss amount in the proposed method. Results showed that, proposed method gives considerable solutions.

Gramani and França (2006) considered the combined cutting stock and lot-sizing problem where the aim was to minimize the trim-loss, the storage and setup costs for different planning periods. They proposed a mathematical model formulation with similar approach exists in shortest path problem. Experimental results showed that combined problem solutions give 28% high profit value than the actualized solutions separately in industry. Also in this study, quality analysis for small problem instances were included.

Minimization of the trim-loss objective would not be valid when stock material length is not sufficiently large. For instance, in Hajizadeh and Lee (2007) work, sum of cutting time and pattern changing time were tried to minimize. The goal was to realize the better efficiency usage of cutting machine capacities. Linear model formulations are developed. In this formulation, available time for a pattern change was formulated as a linear function. The experiments showed that, presented approach could give 400% better results than any existing methods.

In the paper industry, paper mills have different quality level of material. Orders were fulfilled by transforming paper mills into a high amount of sheeted products. Hence, huge reels of different paper grades were produced. Later, these reels were being cut into smaller rolls. Reuse of the material was not possible. Proposed binary linear model used two approaches. The first one was branch-and-bound based on column generation and the second was the heuristic method for marginal cost minimization. The two methods were tested on real data and also on randomly generated problem instances. The proposed approach was implemented in a large pulp and paper company (Chauhan et al., 2008).

Finally, similar with above described methods can be found for solving two-dimensional, three-dimensional or multi-dimensional cutting problems.

CHAPTER THREE
A REAL-WORLD MARBLE CUTTING PROBLEM AND DEVELOPMENT OF MIXED INTEGER LINEAR PROGRAMMING MODEL

3.1 Chapter Introduction

The importance of using more efficiently exhaustible natural resources parallel with the rapidly growing world population is increasing every day with the rising demand for limited raw materials. Marble processing industry is one of the industries of which inputs are non-renewable natural resources. Since having 40% of marble reserves of all the world, Turkey is on the way to become the largest marble exporter of the world. It can be seen that marble has a great importance for the Turkish economy, and the problems of marble factories face well-known CSPs. The lack of technology and unscientific methods applied in quarrying marble within cutting marble process generates a high volume of inevitable wastes in marble cutting industry. In general, since there exist limited available resources in nature, the waste of marble is aimed to be decreased, and also aggregation of the waste of marble is desired. Hebhoub et al. (2011) presented the study methodology about waste marble aggregates and various practical formulations of concrete.

Industrial production process involving CSPs are common in many industrial process such as steel industry (Ferreira et al., 1990), clothing industry (Gradišar et al., 1997), aluminium industry (Stadtler et al., 1990). In these industries, reducing production costs or improving efficiency are often associated with using appropriate cutting plans or cutting strategies. CSPs are mostly described as a problem, where a number of ordered small pieces must be cut from larger pieces. The goal is to satisfy customer orders, and to take into account market needs. In this work, the 1DCSP is studied that arises the manufacturing process of a marble products, such as marble plane. The problem is how to cut a set of objects (stock lengths of marble blocks) with known sizes, qualities and quantities to make exactly a set of items (order lengths of marble planes) with specified sizes, qualities and demands to be fulfilled. According to Dyckhoff (1990)’s C&P typology, discussed problem can be classified

as 1|V|D|M when the stock size is sufficient, where 1 refers to a one-dimensional problem, V means that all ordered items must be cut, D means that the available objects may be different sizes and M means that there are many items and they are all different sizes. If the stock size is insufficient, the problem can be described as 1|B|D|M, in which B stands for all large objects and a selection of small items.

Wäscher et al. (2007) extended Dyckhoff (1990)'s typology with introduce new criteria that define other new problems, beyond the ones defined previously. According to this extended typology, the discussed problem can be classified as a residual cutting stock problem (RCSP), multiple stock-size cutting stock problem (MSSCSP) categorized as a hybrid one-dimensional cutting stock problem (HODCSP).

The 1DCSPUL is frequently encountered in practical applications. In this problem, different types of objects and retails which are generated in previous cuts in stock are considered. After cutting marble blocks according to the cutting plan, if the leftover is greater than a given limit, it returns to stock as retail to be used in the future. Length of the smallest item in customer demand is taken as a limit. Abuabara and Morabito (2009) proposed two mathematical models to solve the 1DCSPUL problem in a Brazilian aeronautical company.

Customer orders can be cut from marble blocks that are equal to or better quality grades than as specified by the customer quality grades. Over grading is defined as using a higher quality metallurgical grade than the grade originally specified by a customer to fill an order. This situation can occur, for example, when orders for a certain grade still need to be filled, but only higher quality stock remains or in some cases cutting simultaneously two block at gangsaw machine instead of cutting individual is more advantageous even cutting marble blocks which have higher quality grade than the customer orders quality grade. Also in the literature, there exist studies dealing 1DCSP with multiple quality grades. For example, Sweeney and Haessler (1990) considered a 1DCSP in paper industry in which different quality levels occur within the same master roll. To solve the problem, they described two-

stage sequential heuristic. Vasko et al. (1999) considered a 1DCSP in the steel industry. In their model they developed a mathematical formulation that simultaneously balances the cost of yield loss against the cost of using higher quality materials.

This problem is particularly important in the marble industry when a set of marble plane need to be cut from marble blocks. Since the width of a marble plane is fully independent of the width of the marble blocks a highly combinatorial problem arises. Although there exist many studies in the literature dealing with CSPs only a few of them examine in marble processing factories. For instance, Ozfirat (2012) studied the problem of minimizing the marble waste in three-dimensional cutting planning problem with single stock length. For proposed problem two different integer programming models were developed in order to minimize total spoilage of marble.

However, to the best of our knowledge, there exists no studies which try to simultaneously determine the cutting plans and select blocks for using efficiently gangsaw machine's available space, considering multiple quality levels and leftovers. The presented model has desirable features which have been proposed for the first time in the literature. The presented study contributes to literature in the following ways:

- Identifies the cutting problem of marble processing industry as a one-dimensional cutting problem with multiple stock size and different quality levels.
- Develops mixed integer programming model for the considered problem.
- Determines cutting plans in order to minimize overgrading and transportation costs.
- The salient feature of this model is that it does not require a priori enumeration of all possible cutting patterns.
- A peculiar aspect of this cutting process is the possibility of generating object remainders (leftovers) with enough size to fulfill future item demands. These leftovers can be maintained in the stock of objects for future usage. Therefore,

the presented problem can be seen as a one-dimensional cutting stock problem with possible leftover generation and reuse. The company does not consider anticipating the cut of items and keeping them in stock because there are difficulties with uncertain demand and quality control.

- In our case, some constraints such as using gangsaw machine and different qualities of marble blocks have its own specificity that do not allow the application of known models and solution algorithms. In order to cope with real-world requirements of analyzed problem of marble factory operating in Turkey, mixed integer linear programming (MILP) model is developed. Using the real life daily customer requirements, the proposed model and the solution methodology are tested. This model is solved using an optimization software LINGO 14.0. Determination of the cutting plans and the selection of blocks for efficient usage of available machine space according to the marble blocks in stock area is an important consideration in the problem formulation. The objective of model is to create a plan of one-dimensional marble blocks cutting into marble planes by minimizing overgrading and transportation costs.

The optimal solution of the CSP is often associated with the minimization of the ‘’trim’’ waste that is generally unavoidable since rolls of standard widths are used. However, trim-loss minimization does not necessarily imply when several standard roll sizes are available or all roll sizes are different. A more direct economic criterion is the minimization of the operation cost incorporating following items:

- The cost of transportation between marble stock area and gangsaw machine area.
- The overgrading cost of cutting marble blocks which satisfy customer order specification.

As pointed out by different authors, there are no general and efficient solution methods for CSPs due to their computational complexity and the diversity of cases in which they can appear.

The relevant literature about the solution procedures of the CSPs can be divided into three groups: (i) linear programming with column-generation and rounding procedures, (ii) MILP and (iii) heuristics and approximate methods. In this study we use approach (ii) and approach (iii). Because we believed it was possible to find optimal integer solutions within a reasonable runtime for small-sized problem instances. However, in the real life as problems get larger and more complex, we need to use metaheuristic methods to solve the problems.

The above mentioned features cause a highly combinatorial problem and it is not surprising that its solution has often been carried out manually based on human expertise. Due to the diversity of situations where CSP arises, we are always faced with new constraints and/or objectives for which the available methods have limited value. Hence, heuristic and metaheuristic approaches were widely used in practice without any evaluation of their performance in order to cope with the problem. The field of hybrid approaches has become very popular among researchers because of insufficient performance of heuristics and pure metaheuristics while exploring the solution space effectively as problems get larger and more complex as in the real life. On the other hand, mathematical programming approaches are used to formally describe the problem.

In this study we proposed a new mathematical programming model for a real-world marble plane cutting problem with some particular features of the real-world problems such as non-used material in the cutting patterns may be used in the future (if large enough), marble blocks are made in different standard sizes, i.e. lengths and widths, and with different requirements of quality levels and guillotine constraints. We developed a MILP model for formally describing the extended problem which is solved to global optimality by using standard techniques. A number of real problem instances is presented to illustrate the efficiency and applicability of the proposed model.

The rest of this chapter is organized as follows. Existing mathematical models with usable leftover which is related to our work is given in section 3.2. Marble plane

production process and product characteristics are described in section 3.3. The formal statement of the problem under consideration is outlined in section 3.4. The proposed MILP model is given in section 3.5. Some real problem instances are solved in section 3.6. Computational experiments are given in section 3.7. Finally, the discussions and conclusions are presented in section 3.8.

3.2 Existing Mathematical Models

Gradišar et al. (1997) was the first one proposing a model to the CSP that considers the existence of usable leftover (retails). This model was applied in the clothing industry to solve the one-dimensional roll cutting for different roll lengths available in stock. The aim of the problem was reducing trim-loss. This model is given as follows:

- m: number of objects
- d_j: size of object j ($1 \leq j \leq m$)
- n: number of items
- s_i: length of item i ($1 \leq i \leq n$)
- b_i: demand of item i ($1 \leq i \leq n$)
- *UB*: minimum size of a leftover material to be considered as retail
- *Y*, *M*: constants to determine the maximum number of different items cut out of one object (the number of different order lengths which are cut out of one roll Y should be not higher than four. (Y=M=4))

Variables:

- x_{ij}: number of item type i is being cut from object j ($1 \leq i \leq n \; and \; 1 \leq j \leq m$)
- δ_j : leftover material of the object j ($1 \leq j \leq m$)
- Δ_i: uncut order of items type i ($1 \leq i \leq n$)
- t_j: loss of the object j ($1 \leq j \leq m$)
- y_{ij}:indicates if an item type i is being cut from object j ($1 \leq i \leq n \; and \; 1 \leq j \leq m$)
- z_j: indicates if object j is used in the cutting plan ($1 \leq j \leq m$)

Model:

$$\min \sum_{i=1}^{n} \Delta_i \quad (minimize\ number\ of\ uncut\ order\ lengths) \tag{3.1}$$

$$\min \sum_{i=1}^{n} t_j \quad (minimize\ trim\ loss\ which\ is\ smaller\ than\ UB) \tag{3.2}$$

Subject to:

$$\sum_{i=1}^{n} s_i\, x_{ij} + \delta_j = d_j \qquad \forall j \quad (knapsack\ constraints) \tag{3.3}$$

$$\sum_{i=1}^{n} x_{ij} + \Delta_i = b_i \qquad \forall i \quad (demand\ constraints) \tag{3.4}$$

$$\sum_{i=1}^{n} y_{ij} \leq Y \leq M \quad \forall j \quad (maximum\ number\ of\ items\ for\ an\ object) \tag{3.5}$$

$$x_{ij} \geq 0\ and\ integer\ , \delta_j \geq 0, t_j \geq 0, \Delta_i \geq 0, y_{ij}, z_j \in \{0,1\} \tag{3.6}$$

$$(1 \leq i \leq n\ and\ 1 \leq j \leq m)$$

For the above model the following functions are used:

$$t_j = \begin{cases} \delta_j\ , & if\ z_j = 1\ and\ \delta_j < UB \\ 0, & otherwise \end{cases} \qquad \forall j \tag{3.7}$$

$$y_{ij} = \begin{cases} 0\ , & if\ x_{ij} = 0 \\ 1, & otherwise \end{cases} \qquad \forall i, j \tag{3.8}$$

$$z_j = \begin{cases} 0\ , & if\ \sum_{i=1}^{n} x_{ij} = 0 \\ 1, & otherwise \end{cases} \qquad \forall j \tag{3.9}$$

In Gradišar et al. (1999) and Gradišar and Trkman (2005) the same model of Gradišar et al. (1997) is analyzed by cases and are added a new variable and a constraint associated to the maximum length of retails (there must not be more than one retail larger than the largest item):

$$\sum_{j=1}^{m} u_j \leq 1, \tag{3.10}$$

$$\text{where: } u_j = \begin{cases} 1\ , & if\ \delta_j < d_j\ and\ \delta_j > max_i s_i \\ 0, & otherwise \end{cases} \qquad \forall j \tag{3.11}$$

Ababuara and Morabito (2009) modified the model of Gradišar et al. (1997) and they implemented their model in manufacturing of agricultural light aircrafts. The proposed two models are given as follows:

Model 1:

Model 1 is an adaptation of Gradišar et al. (1997) model under the condition of not having limits on the number of different items allocated per object, and of having enough availability of objects in stock where they are defined:

UB : $min_i s_i$

M : $max_j d_j$-$min_i s_i$

RUB : Upper bound for the number of retails in the solution (In this model it is taken as *RUB*=1)

ε : A small positive value to change the signal of inequalities

Adds the auxiliary binary variable: w_j (1≤j≤ m)

$$w_j = \begin{cases} 1, & if\ \delta_j < UB \\ 0, & \delta_j \geq UB \end{cases} \qquad \forall j \tag{3.12}$$

$$u_j = \begin{cases} 1, & if\ z_j = 1\ and\ \delta_j \geq UB \\ 0, & otherwise \end{cases} \qquad \forall j \tag{3.13}$$

$$min \sum_{j=1}^{m} t_j \tag{3.14}$$

Subject to:

$$\sum_{i=1}^{n} s_i\, x_{ij} + \delta_j = d_j \qquad \forall j \quad (knapsack\ constraints) \tag{3.15}$$

$$\sum_{j=1}^{m} x_{ij} = b_i \qquad \forall i \quad (demand\ constraints) \tag{3.16}$$

$$z_j \leq \sum_{i=1}^{n} x_{ij} \qquad \forall j \quad (object\ constraints) \tag{3.17}$$

$$\sum_{i=1}^{n} x_{ij} \leq \mathrm{M} z_j \qquad \forall j \quad (object\ constraints) \tag{3.18}$$

$$\delta_j - UB \geq -\mathrm{M} w_j + \varepsilon \qquad \forall j \quad (lost\ constraints) \tag{3.19}$$

$$\delta_j - UB \leq \mathrm{M}(1 - w_j) \qquad \forall j \quad (lost\ constraints) \tag{3.20}$$

$$t_j - \mathrm{M} w_j \leq 0 \qquad \forall j \quad (lost\ constraints) \tag{3.21}$$

$$t_j - \mathrm{M} z_j \leq 0 \qquad \forall j \quad (lost\ constraints) \tag{3.22}$$

$$t_j - \delta_j \leq 0 \qquad \forall j \quad (lost\ constraints) \tag{3.23}$$

$$\delta_j - t_j + M w_j + M z_j \leq 2\mathrm{M} \qquad \forall j \quad (lost\ constraints) \tag{3.24}$$

$$-z_j + u_j \leq 0 \qquad \forall j \quad (retail\ constraints) \tag{3.25}$$

$$w_j + u_j \leq 1 \qquad \forall j \quad (retail\ constraints) \tag{3.26}$$

$$z_j - w_j - u_j \leq 0 \qquad \forall j \quad (retail\ constraints) \tag{3.27}$$

$$\sum_{j=1}^{m} u_j \leq RUB \qquad (maximum\ number\ of\ retails) \tag{3.28}$$

$$x_{ij}, \delta_j, t_j \geq 0, x_{ij} \in \mathbb{Z}\ ,\ z_j, u_j, w_j \in \{0,1\} \quad (1 \leq i \leq n\ and\ 1 \leq j \leq m) \tag{3.29}$$

In this model, the object constraints set z_j to 1 if object j is used in the solution and to 0 otherwise. The lost constraints guarantee that if the object is being used then the retail from that object is not a leftover which means the retail of that object is trim-loss. The retail constraints guarantee that if the object is being used and the retail is not trim-loss material, then it must be leftover.

Since, there may be no feasible solutions for some instances, even if the lengths of the items are shorter than those of the stock objects. The reason is that if there are more than one leftover in the cutting plan, due to the constraint of maximum number of retails, the model becomes infeasible. To prevent from this shortcoming, model 2 is proposed.

Model 2:

$$min \sum_{j=1}^{m} t_j \tag{3.30}$$

Subject to:

$$\sum_{i=1}^{n} s_i\, x_{ij} \leq\ d_j \qquad \forall j \quad (knapsack\ constraints) \tag{3.31}$$

$$\sum_{j=1}^{m} x_{ij} = b_i \qquad \forall i \quad (demand\ constraints) \tag{3.32}$$

$$\sum_{j=1}^{m} u_j \leq RUB \qquad (maximum\ number\ of\ retails) \tag{3.33}$$

$$UBu_j \leq d_j z_j - \sum_{i=1}^{n} s_i\, x_{ij} \qquad \forall j \quad (leftover\ constraints) \tag{3.34}$$

$$d_j z_j - \sum_{i=1}^{n} s_i\, x_{ij} \leq t_j + Mu_j \qquad \forall j \quad (leftover\ constraints) \tag{3.35}$$

$$x_{ij}, t_j \geq 0, x_{ij} \in \mathbb{Z}\ , u_j, z_j \in \{0,1\} \quad (1 \leq i \leq n\ and\ 1 \leq j \leq m) \tag{3.36}$$

The leftover constraints substitute all the object constraints, lost constraints, and retail constraints presented in the previous model.

Model 2 with multiple periods:

Model 2 can be extended to multi-period cutting planning by defining k as the time period index (k=1, 2, …, q). Model 2 with multiple periods can be stated as:
C_k: Cutting capacity of period k (in terms of total length cut)

$$min \sum_{k=1}^{q} \sum_{j=1}^{m} t_{jk} \quad (3.37)$$

Subject to:

$$UBu_{jk} \leq d_j z_{jk} - \sum_{i=1}^{n} s_i x_{ijk} \quad \forall jk \quad (leftover\ constraints) \quad (3.38)$$

$$d_j z_{jk} - \sum_{i=1}^{n} s_i x_{ijk} \leq t_{jk} + Mu_{jk} \quad \forall jk \quad (leftover\ constraints) \quad (3.39)$$

$$\sum_{p=1}^{k} \sum_{j=1}^{n} x_{ijp} \geq \sum_{p=1}^{k} d_{ip} \quad \forall i, \forall k, k \neq q \quad (knapsack\ constraints) \quad (3.40)$$

$$\sum_{k=1}^{q} \sum_{j=1}^{n} x_{ijk} = \sum_{k=1}^{q} d_{ik} \quad \forall i \quad (knapsack\ constraints) \quad (3.41)$$

$$\sum_{k=1}^{q} \sum_{j=1}^{n} u_{jk} \leq RUB \quad (maximum\ number\ of\ retails) \quad (3.42)$$

$$\sum_{j=1}^{n} b_j z_{jk} \leq C_k \quad \forall k \quad (cutting\ capacity\ constraints) \quad (3.43)$$

$$x_{ijk}, t_{jk} \geq 0, x_{ijk} \in \mathbb{Z}, u_{jk}, z_{jk} \in \{0,1\} \quad (1 \leq i \leq n \ and \ 1 \leq j \leq m) \quad (3.44)$$

The possible benefits of multi-period cutting planning can be advancing the cutting of items and efficient stock management.

The models which are given above do not solve marble plane cutting problem, so some modifications are performed to get a model that actually solves the marble plane cutting problem.

3.3 Process Description

Marble blocks which are supplied from marble quarries are cut, polished and turned into marble products in marble processing factories. Some of the marble processing factories only produce marble planes, whereas others produce final marble products. In our case we consider the cutting operation of the marble blocks into marble planes stage. Factories that producing marble planes have cutting

machines in order to cut marble block into marble planes. The marble block cutting machines are classified according to their relevant cutting principles as circular saw (stripper, trimmer), blade (gangsaw), and wire (multiwire) block cutting machine according to the cutting pattern of the marble. In this paper, operation of cutting the marble blocks into marble planes in the gangsaw machine which is the most preferred machine by the marble processing plants is considered. Gangsaw consists of a frame fitter on four columns, and diamond-socket blades stretched on the frame, and a flywheel. The frame on which the blades are fitted is eccentrically attached to the flywheel by an arm, and reciprocates with rotation of the flywheel. Cutting takes place by lifting the block upside in some models, or by lowering the blades in other models. The assignment of marble blocks to customer demand width generates cutting plans. The cutting pattern for each customer demand is determined by the position of the knives of gangsaw machine. Number of different type of demand width which is assigned to marble blocks generate different cutting plans. In order to illustrate the different types of cutting plans, some plan examples are shown in Figure 3.1. Explicitly, in Figure 3.1.a 7-units type 2 cm demand width with trim-loss is given. In Figure 3.1.b, 1-unit type 2 cm demand width and 4-units type 3 cm demand width and 1-unit type 4 cm demand width are depicted. In Figure 3.1.c, 5-units type 2 cm demand and 2-units type 3 cm demand width are shown. In Figure 3.1.d, 2-units type 2 cm demand width and 3-units type 2 cm demand width with trim-loss are given.

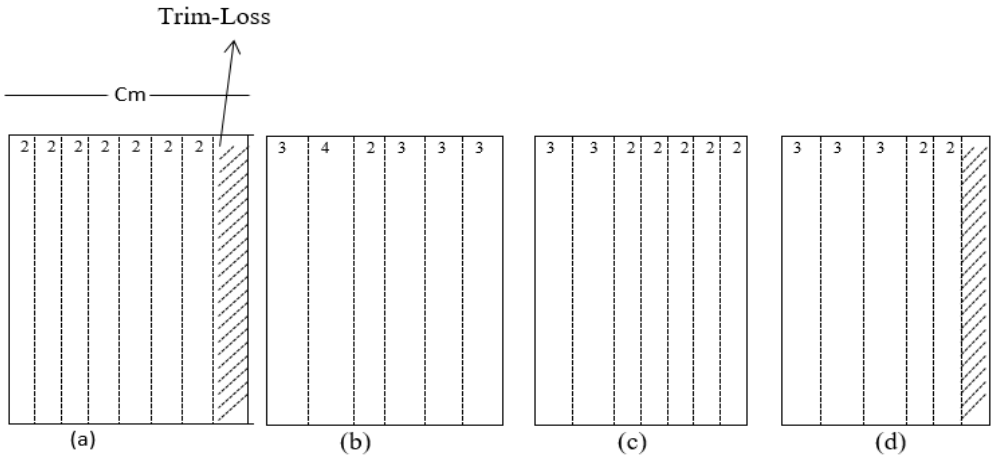


Figure 3.1 Some cutting plan examples.

Aesthetic homogeneity is an important issue in marble planes. The reason is that they are mostly used in application in covering surfaces for decoration. In earth usually identical marble in terms of color, texture and completely homogeneous structure cannot be available. Most of the time, in the same marble quarry or even in the same block, differences may be observed. It is necessary to classify marble planes into homogeneous classes or groups according to their quality and appearance. According to Selver et al. (2009) there are some criteria such as color homogeneity, texture, spots, distribution of limestone, cohesive material or thin joints filled by cohesive material to classify marble planes into homogeneous classes. In this study marble factory operating in Izmir region of Turkey is contacted, and experts have performed the classification. Under above-stated criteria, three quality groups (sorted from high to low quality) have been considered. These are homogenous limestone, limestone with thin joints, brecciated limestone which is shown in Figure 3.2.

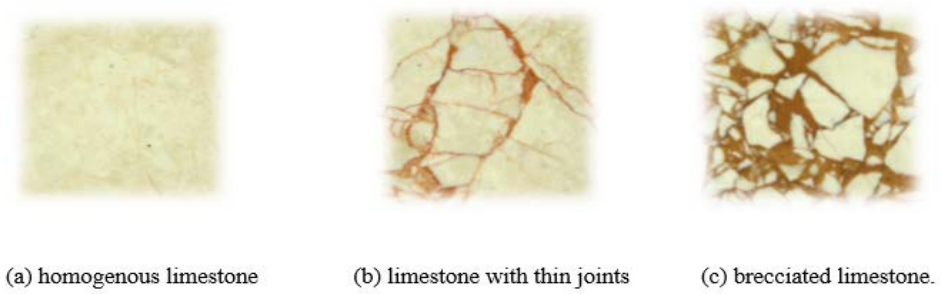


Figure 3.2 Three quality level of marble blocks (Selver et al., 2009)

In the marble industry, production systems start with quarrying raw natural stone blocks by separating the raw block from the excavation site. The following step is conveying irregular marble blocks from marble quarries by trucks. Before placed into marble stock area, these irregular marble blocks are cut with Diamond monoblade machine. Monoblade machine is used to give rectangular shape to irregular blocks. So, all dimensions become integer numbers. Next, these blocks are placed to the marble stock area. In marble stock area, marble blocks are classified according to their quality level by experts. The most important stage of the production system is the selection of quarried blocks. Blocks are selected according

to meet customer demand with requested quality and dimension of cutting machine. After the selection, blocks are transported to plants and they are sorted according to production schedule and transferred to the stone cutting lines. Then selected blocks are deployed into the gangsaw machine for the cutting process. In this cutting process stage, two different situations occur. These are cutting simultaneously two blocks or cutting one block in the gangsaw machine. After cutting planes that were cut in the gangsaw machine, they are either sent directly to customers, or they are sent to further cutting operation of cutting plane into slabs. Figure 3.3 shows main stages of the cutting process of marble plane.

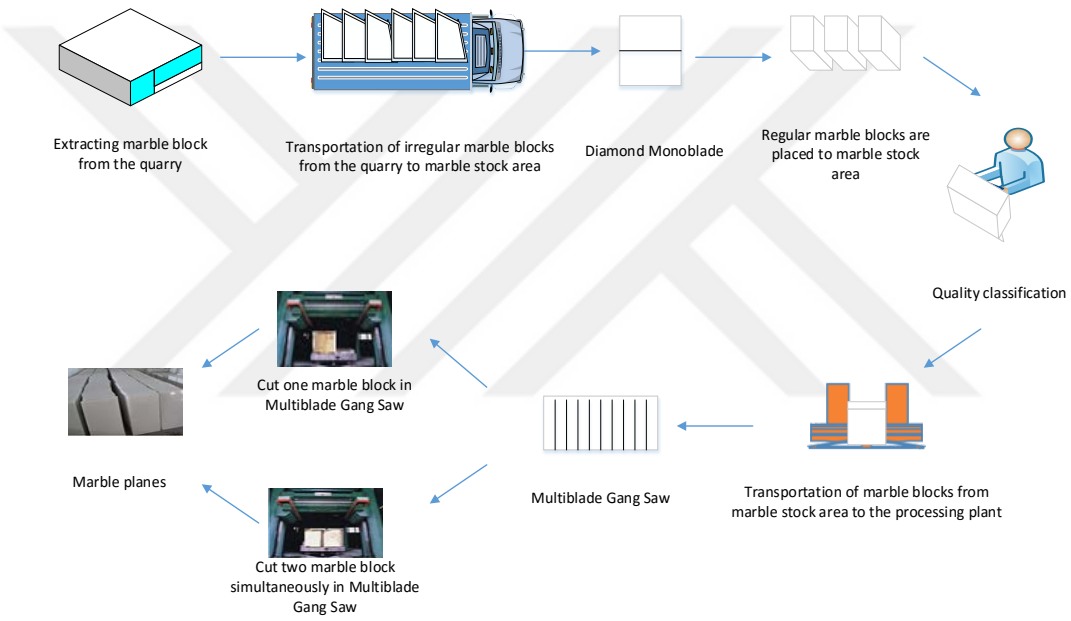


Figure 3.3 Marble plane production process

3.4 Problem Statement

The production planning of cutting marble process is mainly based on the product demand. A typical planning horizon is one or more weeks, and order lists are released daily or weekly. Each order list is planned individually, and for each list there are different available stock lengths. So planning horizon is determined by the customer order. We called each cutting process time as a period.

This paper presents a procedure for solving 1DCSPs when both stock lengths and customer orders have multiple quality gradations. In our study, three categories or qualities have been considered by evaluating their colorimetric and textural properties. Marble categories are hierarchically ordered, with quality-1 (homogenous limestone) marbles the most highly qualified, quality-2 (limestone with thin joints) the next, and so on. Proposed model allows with more qualified marble, being able to substitute for less qualified. The highest quality material which is quality-1 can be used to satisfy orders for any quality level, quality-2 material can be used for all orders except quality-1, and so forth. In this situation, there is clearly a value in assigning material to the highest quality orders possible. In general, the importance of doing so depends on the revenue gains between the various quality grades. For the problem considered here, feasibility of cutting plan depends not only on its total width, but also on the assigning of the ordered widths to corresponding quality requirements. The purpose of the work is to develop a procedure to prepare cutting plans of marble blocks by taking into account the customer demand.

3.5 The Mixed Integer Linear Programming Model

The aim of the mathematical formulation is to determine the followings:

- Determine the used marble blocks in each cutting order
- Number of cutting customer order type from marble blocks in each cutting order
- Determination of the marble blocks which have been cut simultaneously in each cutting order
- Cutting order of each marble blocks
- Determination of the amount of retail, and whether the retail is trim-loss or leftover in each cutting order.

Input and output of the model are represented in Figure 3.4

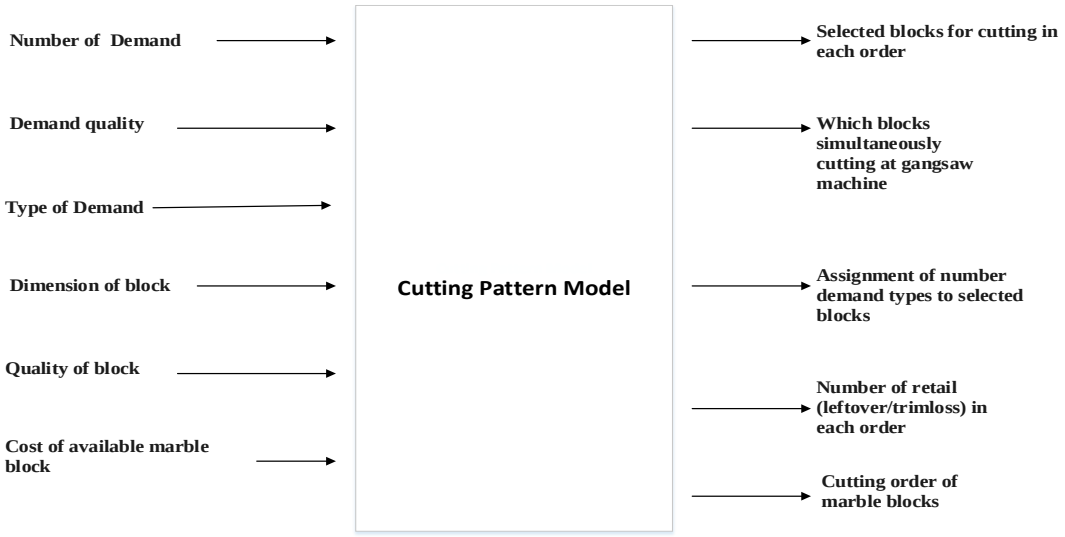


Figure 3.4 Model input and output.

3.5.1 Model Notations

In order to describe the proposed model more clearly, the stated assumptions are made as given below, and notations are given in the Table 3.1.

Assumptions:

- The dimension of marble blocks and customer orders are assumed to be known.
- Marble blocks are available in stock in different sizes, qualities, and enough quantities to cut all ordered marble planes are available as well. (Because they are extracted by the company's own marble quarry to prevent a lack of material in the cutting process).
- The setup times of the cutting equipment are not relevant and are disregarded.
- Without loss of generality, we consider demand as integer number.

Table 3.1 Model Notations

	Notation	Definition
Sets & Indices	B	Total number of marble blocks
	T	Total number of customer order types
	P	Total number of cutting order in the planning horizon
	Q	Total number of quality levels of marble blocks and customer order types
	j	Set of customer order types $j \in \{1,2,\ldots,T\}$
	i	Set of marble blocks in stock $i \in \{1,2,\ldots,B\}$
	k	Set of cutting order in the planning horizon $k \in \{1,2,\ldots,P\}$
	m	Set of quality levels $m \in \{1,2,\ldots,Q\}$
Parameters	WM	Width of cutting machine (cm)
	$bigM$	A very large number
	Wb_i	Width of marble block i (cm)
	Wt_j	Width of customer order type j (cm)
	H_i	Height of marble block i (cm)
	D_j	Demand of customer order type j in the planning horizon
	L_i	Threshold value for decision of trim or leftover for marble block i
	k_i	Quality level of marble block i
	q_j	Quality level of customer order type j
	$CutCost_{ij}$	Unit cost of cutting marble block i for customer order type j
	$TCost$	Fixed transportation cost between marble stock area and gangsaw machine area
	δ	Height difference between simultaneously cutting marble blocks
	ε	A small positive integer
Decision variables	z_{ijk}	Number of customer order type j to be cut out of marble block i in cutting order k
	s_{ik}	Non-negative slackness variable representing retail of marble block either a trim-loss or leftover of marble block i in cutting order k
	u_{ik}	Retail of marble block i in cutting order k equals to 1 if b_{ik}=1 and $s_{ik} \geq L_i \ \forall i \in B, k \in P$ and 0 if b_{ik}=0 or $s_{ik} < L_i \ \forall i \in B, k \in P$
	t_{ik}	Trim-loss of marble block i in cutting order k and equals to s_{ik} if b_{ik}=1 and $s_{ik} < L_i \ \forall i \in B, k \in P$ and 0 if b_{ik}=0 or $s_{ik} \geq L_i \ \forall i \in B, k \in P$

Table 3.1 Model Notations (continue)

Decision variables	$b_{ik} \in \{0,1\}$	Equals to 1 if marble block *i* is cutting in cutting order *k* and 0 otherwise
	$x_{irk} \in \{0,1\}$	Equals to 1 if marble block *i* and marble block *r* is cutting simultaneously in cutting order *k* and 0 otherwise
	$w_{ik} \in \{0,1\}$	Equals to 1 if $s_{ik} < L_i$ and 0 if $s_{ik} \geq L_i$

Determination of cutting patterns model:

The objective of the optimization is minimizing the total cost of the operation as given in Equation 3.45 - 3.46:

- minimize overgrading (using a more expensive quality level than the customer specified)

$$\sum_i^{\mathrm{B}} \sum_j^{\mathrm{T}} \sum_k^P CutCost_{ij} z_{ijk} \tag{3.45}$$

- minimize transportation cost

The minimum transportation cost solution becomes particularly important for the present company for which the cost of transportation of marble blocks from marble stock area to the processing plant represents at least one-fourth of the total costs of the final product. When two blocks simultaneously cutting, one of the transportation cost is subtracted since they transport together.

$$\sum_i^B \sum_k^P TCostb_{ik} - \sum_i^{\mathrm{B}} \sum_r^{\mathrm{B}} \sum_k^P TCostx_{irk} \tag{3.46}$$

The above terms can be collected in the following objective function as given in Equation 3.47.

$$\text{Min } = \sum_i^{\mathrm{B}} \sum_j^{\mathrm{T}} \sum_k^P CutCost_{ij} z_{ijk} + \sum_i^B \sum_k^P TCostb_{ik} - \sum_i^{\mathrm{B}} \sum_r^{\mathrm{B}} \sum_k^P TCostx_{irk} \tag{3.47}$$

Demand constraints:

$$\sum_{k}^{P} \sum_{i|k_i \leq q_j}^{B} z_{ijk} = D_j \qquad \forall j \in T \qquad (3.48)$$

Constraints shown in Equation 3.48 state that the total customer order type j must be cut exactly equal to the demand D_j by using the stock material in which the block quality is equal or higher than specified customer quality.

Machine eligibility constraints:

$$(H_i - H_r)\, x_{irk} \leq \delta \qquad \forall i \in B, r \in B, k \in P \quad i < r \qquad (3.49)$$

$$(H_i - H_r)\, x_{irk} + \delta \geq 0 \qquad \forall i \in B, r \in B, k \in P \quad i < r \qquad (3.50)$$

Constraints shown in Equation 3.49 - 3.50 satisfy that the height difference between two entering blocks which are being cut in the same order. The height difference between the two blocks where cutting in gangsaw machine at the same time should not exceed a certain value δ. In the entry part of the cutting process, gangsaw machine speed must be reduced when cutting the marble blocks. The reason of the lowering the speed of the gangsaw machine is to prevent from the high pressure effect on the block surface. Blade speed is accelerated afterwards. In the beginning of cutting blocks with using gangsaw machine, blades have 8-10 cm/h speed. After entering blades into the blocks, speed increases according to block's thickness. In medium hardness stone strength, it will be 30 cm/h, and in soft stone strength it will be 50 cm/h. If the height difference is too high, it will increase process time of cutting marble blocks. Because speed is adjusted according to height of the marble block.

$$Wb_i b_{ik} + Wb_r b_{rk} \leq WM \qquad \forall i \in B, r \in B, k \in P \quad i \neq r \qquad (3.51)$$

Constraints shown in Equation 3.51 state that if two blocks simultaneously enter into cutting process, sum of their width must be equal or smaller than machine width in each cutting order.

$$b_{ik} + b_{rk} \leq bigM(1 - x_{irk}) + 2 \qquad \forall i \in B, r \in B, k \in P \quad i < r \qquad (3.52)$$

$$b_{ik} + b_{rk} \leq bigMx_{irk} + 1 \qquad \forall i \in B, r \in B, k \in P \quad i < r \qquad (3.53)$$

$$b_{ik} + b_{rk} > x_{irk} \qquad \forall i \in B, r \in B, k \in P \quad i < r \qquad (3.54)$$

Constraints shown in Equation 3.52 - 3.54 satisfy that if the block *i* and block *r* are cutting at the same time, then variable x_{irk} takes the one value and satisfy that maximum of two blocks enter into cutting process simultaneously.

$$\sum_r^B \sum_i^B x_{irk} \leq 1 \qquad \forall k \in P \quad i < r \qquad (3.55)$$

Constraints shown in Equation 3.55 state that maximum of two blocks simultaneously enter into the cutting process in each cutting order.

$$x_{irk} = 0 \qquad \forall i \in B, r \in B, k \in P \quad r \leq i \qquad (3.56)$$

Without loss of generality, it is possible to reduce the symmetry of the model by adding the constraint $i < r$ in case of marble block *i* and marble block *r* are being cut simultaneously. Constraints shown in Equation 3.56 significantly reduce the solution space and contributes to the computational efficiency. This constraints state that the variable x_{irk} in case of $r \leq i$ is equal to zero.

Cutting constraints:

$$b_{ik} \leq \sum_j^T z_{ijk} \qquad \forall i \in B, k \in P \qquad (3.57)$$

$$\sum_j^T z_{ijk} \leq bigMb_{ik} \qquad \forall i \in B, k \in P \qquad (3.58)$$

Constraints shown in Equation 3.57 - 3.58 express that if marble block *i* is not actually cut in cutting order *k*, then no customer order of any type can be cut from it.

Knapsack constraints:

$$\sum_j^T Wt_j\, z_{ijk} + s_{ik} = Wb_i \qquad \forall i \in B, k \in P \qquad (3.59)$$

Constraints shown in Equation 3.59 express the total length of customer order of all type *j* which is cutting from block *i* must not exceed the block length Wb_i in each order. The knapsack constraints guarantee that the sum of lengths of the items cut from an object is not larger than the length of the object.

Leftover or trim constraints:

$$t_{ik} \leq s_{ik} \qquad \forall i \in B, k \in P \qquad (3.60)$$

$$t_{ik} \leq bigMb_{ik} \qquad \forall i \in B, k \in P \qquad (3.61)$$

$$t_{ik} \leq bigMw_{ik} \qquad \forall i \in B, k \in P \qquad (3.62)$$

$$w_{ik} + u_{ik} \leq 1 \qquad \forall i \in B, k \in P \qquad (3.63)$$

$$u_{ik} \leq b_{ik} \qquad \forall i \in \mathrm{B}, k \in \mathrm{P} \qquad (3.64)$$

$$b_{ik} \leq w_{ik} + u_{ik} \qquad \forall i \in B, k \in P \qquad (3.65)$$

$$s_{ik} + bigMw_{ik} \geq L_i + \varepsilon \qquad \forall i \in B, k \in P \qquad (3.66)$$

$$s_{ik} < bigM(1 - w_{ik}) + L_i \qquad \forall i \in B, k \in P \qquad (3.67)$$

$$s_{ik} + bigM\, w_{ik} + bigM\, b_{ik} \leq 2bigM + t_{ik} \qquad \forall i \in B, k \in P \qquad (3.68)$$

Constraints between Equation 3.60 - 3.68 determine the retail of block is either leftover or trim-loss.

Ordering constraints:

$$x_{irk} \leq b_{ik} \qquad \forall i \in B, r \in B, k \in P \quad i < r \qquad (3.69)$$

$$x_{irk} \leq b_{rk} \qquad \forall i \in B, r \in B, k \in P \quad i < r \qquad (3.70)$$

$$x_{irk} \geq b_{ik} + b_{rk} - 1 \qquad \forall i \in B, r \in B, k \in P \quad i < r \qquad (3.71)$$

Constraints shown in Equation 3.69 - 3.71 satisfy that if marble block *i* and marble block *r* are being cut in the same cutting order. It means that they are being

cut simultaneously. Hence variable x_{irk} take a value one, otherwise it takes a value zero.

$$\sum_k^P x_{irk} \leq 1 \qquad \forall i \in B, r \in B \quad i < r \qquad (3.72)$$

Constraints shown in Equation 3.72 state that two marble blocks can be cut simultaneously in one exact cutting order.

$$\sum_k^P b_{rk} \leq 1 \qquad \forall\, r \in B \qquad (3.73)$$

Constraints shown in Equation 3.73 state that marble block can be cut in one exact cutting order.

$$\sum_j^T \sum_i^B z_{ijk} \geq \sum_j^T \sum_i^B z_{ijk+1} \qquad \forall k \in P \quad k < B \qquad (3.74)$$

Constraints shown in Equation 3.74 ensure that the total number of customer order type *j* to be cut out of marble block *i* in a cutting order *k+1* if and only if its preceding cutting order is cut for satisfying customer demand.

Integer constraints:

$$z_{ijk} \geq 0 \text{ and integer} \qquad \forall i \in B, j \in T, k \in P \qquad (3.75)$$

Constraints shown in Equation 3.75 satisfy that the variables z_{ijk} are to be integers.

3.6 An Illustrative Examples

In order to demonstrate computational efficiency of the proposed model, the cutting problem of marble processing factory operating in Izmir, Turkey is considered. The optimization software LINGO 14.0. has been used for the solution in a Windows 8 laptop with Intel Core i5-4210U CPU 1.7 GHz processor and 4 GB

RAM. In this section, we consider five example problem instances of increasing complexity in order to investigate the computational behavior of our formulation. When satisfying the customer demand, we add 1 cm to width of each customer order because the gangsaw cutting blades lose 1 cm for each demand width. In other words, for the customer order of 2 cm width plane, width of 3 cm is considered in the model. In this study, gangsaw machine has a dimension of 1,385 (length) x 275 (width) x 600 (height) cm. Customer order marble planes have dimension lengths of 30,5 cm, thickness 4 - 7 cm, width 2 - 5 cm. Height difference between two entering blocks are taken as 10 cm and threshold value for decision of trim-loss or leftover for each marble blocks are taken as 3 cm. The reason is minimum customer type requires 2 cm width and if we add 1 cm for gangsaw cutting blades it will be equal to 3 cm. Unit cost (TL) of cutting marble blocks and fixed transportation cost (TL) are considered as $CutCost_{ij}$ = [30, 50, 60; 0, 20, 30; 0, 0, 10] and $TCost$ = 2000 respectively.

The data for the problem instance1, which shows the customer orders, is given in Table 3.2. Data, which shows the available marble block for cutting operation, is given in Table 3.3.

Table 3.2 Number of customer demand types of marble planes for problem instance1

Customer Orders			
Customer Order_No	**Width (cm)**	**Customer Demand (Unit)**	**Quality**
C1	3	145	2
C2	5	80	1
C3	4	60	3
C4	2	110	2

Table 3.3 Available marble blocks data for cutting operation for problem instance1

Available Marble Stocks			
Blok_No	**Width of block (cm)**	**Height of block (cm)**	**Quality of block**
B1	180	150	3
B2	165	160	3
B3	110	165	2
B4	200	155	3
B5	125	150	2
B6	125	160	3
B7	140	155	1
B8	150	165	2
B9	100	155	3
B10	170	150	1
B11	120	160	2
B12	130	165	3
B13	140	155	1
B14	160	150	2
B15	180	160	1
B16	175	155	2
B17	165	160	3
B18	180	155	1
B19	170	160	2
B20	160	150	3
B21	160	155	2
B22	180	165	3
B23	120	160	1
B24	130	150	2
B25	140	155	3

The data for the problem instance2, which shows the customer orders, is given in Table 3.4. Data, which shows the available marble block for cutting operation, is given in Table 3.5.

Table 3.4 Number of customer demand types of marble planes for problem instance2

Customer Orders			
Customer Order_No	**Width (cm)**	**Customer Demand (Unit)**	**Quality**
C1	3	85	2
C2	5	95	1
C3	4	50	3
C4	2	130	2

Table 3.5 Available marble blocks data for cutting operation for problem instance2

Available Marble Stocks			
Blok_No	**Width of block (cm)**	**Height of block (cm)**	**Quality of block**
B1	180	150	3
B2	165	160	3
B3	110	165	2
B4	130	155	3
B5	125	150	2
B6	125	160	3
B7	140	155	1
B8	180	155	2
B9	100	155	3
B10	170	150	1
B11	120	160	2
B12	130	165	3
B13	140	155	1
B14	160	150	2
B15	180	160	1
B16	175	155	2
B17	165	150	3
B18	180	155	1
B19	170	160	2
B20	160	150	3
B21	150	150	3
B22	160	155	3
B23	130	160	1

The data for the problem instance3, which shows the customer orders, is given in Table 3.6. Data, which shows the available marble block for cutting operation, is given in Table 3.7.

Table 3.6 Number of customer demand types of marble planes for problem instance3

Customer Orders			
Customer Order_No	**Width (cm)**	**Customer Demand (Unit)**	**Quality**
C1	3	85	2
C2	5	70	3
C3	4	50	1
C4	2	90	2

Table 3.7 Available marble blocks data for cutting operation for problem instance3

Available Marble Stocks			
Blok_No	**Width of block (cm)**	**Height of block (cm)**	**Quality of block**
B1	170	150	3
B2	165	160	3
B3	110	165	2
B4	200	155	3
B5	125	150	2
B6	125	160	3
B7	140	155	1
B8	150	165	2
B9	100	155	3
B10	170	150	1
B11	120	160	2
B12	130	165	3
B13	140	155	1
B14	160	150	2
B15	180	160	1
B16	175	155	2
B17	165	160	3
B18	180	155	1
B19	170	160	2
B20	160	150	3
B21	130	155	1
B22	145	170	2

The data for the problem instance4, which shows the customer orders, is given in Table 3.8. Data, which shows the available marble block for cutting operation, is given in Table 3.9.

Table 3.8 Number of customer demand types of marble planes for problem instance4

Customer Orders			
Customer Order_No	**Width (cm)**	**Customer Demand (Unit)**	**Quality**
C1	3	55	2
C2	5	75	1
C3	4	85	3
C4	2	100	2

Table 3.9 Available marble blocks data for cutting operation for problem instance4

Available Marble Stocks			
Blok_No	**Width of block (cm)**	**Height of block (cm)**	**Quality of block**
B1	180	150	3
B2	165	160	3
B3	110	165	2
B4	200	155	3
B5	125	150	2
B6	125	160	3
B7	140	155	1
B8	150	165	2
B9	100	155	3
B10	170	150	1
B11	120	160	2
B12	130	165	3
B13	140	155	1
B14	160	150	2
B15	180	160	1
B16	175	155	2
B17	165	160	3
B18	180	155	1
B19	170	160	2
B20	160	150	3

The data for the problem instance5, which shows the customer orders, is given in Table 3.10. Data, which shows the available marble block for cutting operation, is given in Table 3.11.

Table 3.10 Number of customer demand types of marble planes for problem instance5

Customer Orders			
Customer Order_No	**Width (cm)**	**Customer Demand (Unit)**	**Quality**
C1	3	25	2
C2	5	40	1
C3	4	30	3
C4	2	80	2

Table 3.11 Available marble blocks data for cutting operation for problem instance5

Available Marble Stocks			
Blok_No	**Width of block (cm)**	**Height of block (cm)**	**Quality of block**
B1	180	150	3
B2	165	160	3
B3	110	165	2
B4	200	155	3
B5	125	150	2
B6	125	160	3
B7	140	155	1
B8	150	165	2
B9	100	155	3
B10	170	150	1
B11	120	160	2
B12	130	165	3
B13	140	155	1
B14	160	150	2
B15	180	160	1
B16	175	155	2
B17	165	160	3

3.7 Computational Experiments

The datasets of the problem instances are not large. They are extracted from real-world problem instances. Table 3.12 presents the performance of the proposed model.

Table 3.12 Computational statistics for problem instances.

Problems	Optimal objective	Variables	Constraints	Computational time
Instance1	24100 TL	13125	82879	4746 sec
Instance2	21650 TL	10580	64864	4311 sec
Instance3	16500 TL	9438	56929	243 sec
Instance4	18200 TL	7400	43054	1218 sec
Instance5	9600 TL	4913	26779	109 sec

The MILP model can solve problems optimally. However, this model cannot obtain optimal solutions for large-scale problems. Therefore, this approach can produce high-quality solutions for only small-sized instances. Cutting plan for problem instance1 is given in Figure 3.5.

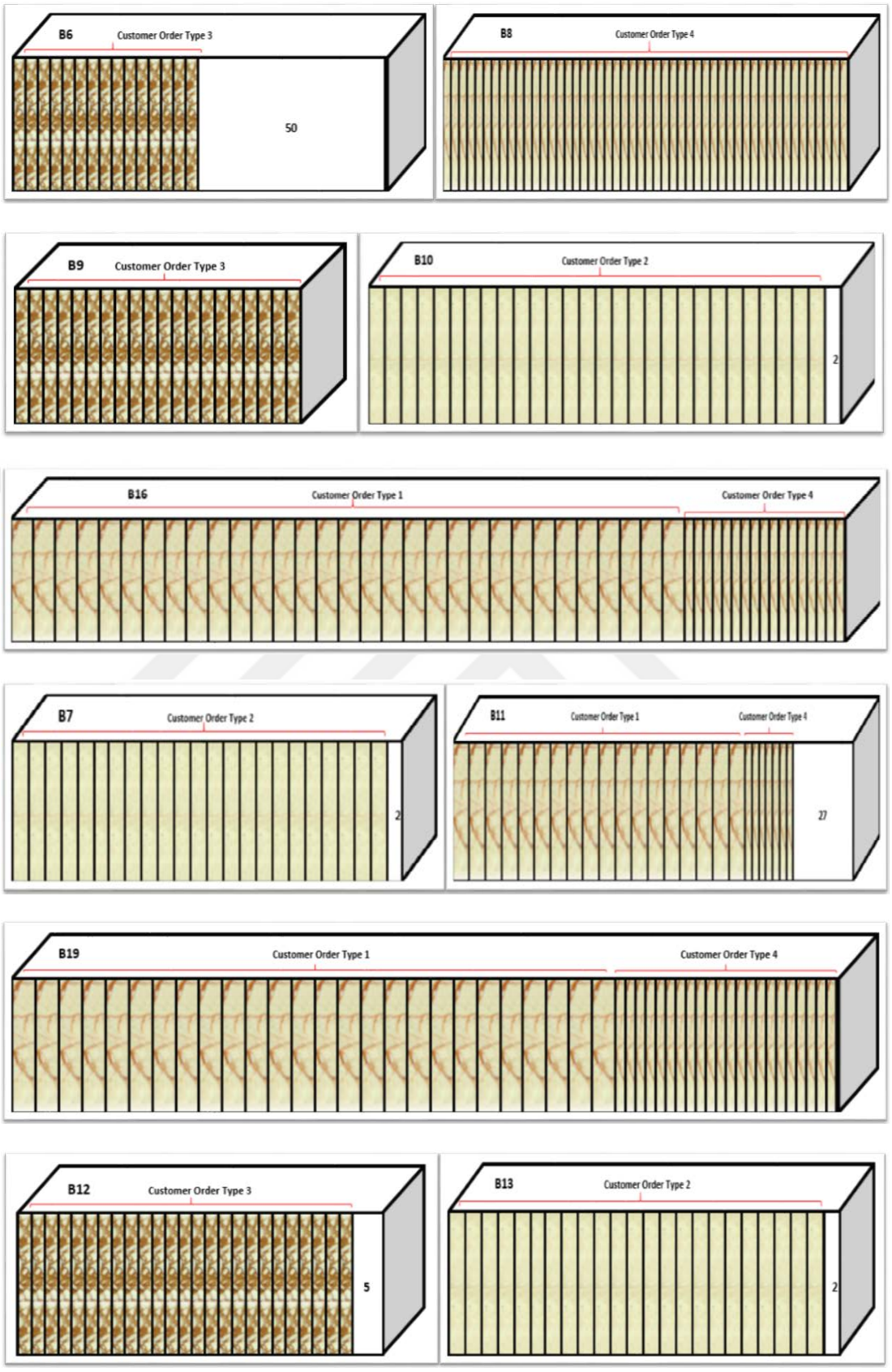


Figure 3.5 Solution representation of problem instance1

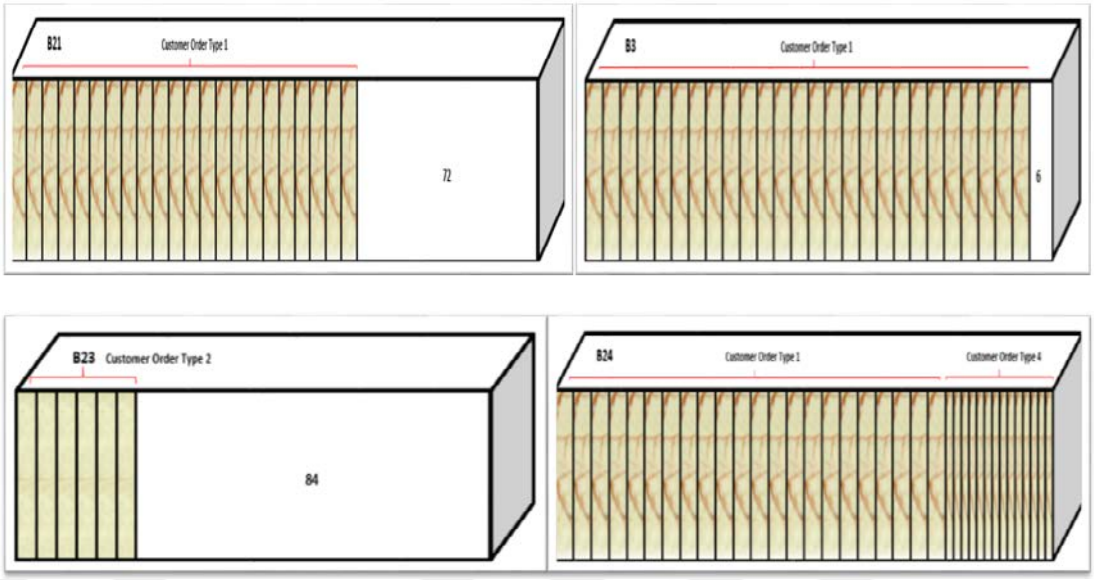


Figure 3.5 Solution representation of problem instance1 (Continue)

The solution representation shows that in each cutting order marble blocks which are cut together, and corresponding customer orders which are satisfied, and leftover amounts for each marble blocks.

3.8 Chapter Conclusions

In this work we study one-dimensional, multiple stock size cutting stock problem of marble blocks in the manufacturing of marble industry. MILP approach is developed for the solution of the problem. This is achieved without the need to resort to a priori generated cutting patterns. Developed approach is used to handle the cutting problem of a real life marble processing factory in Turkey. The discussed company does not use any cutting planning optimization tool and in practice the operators often cut the objects into the items without a detailed plan. Real problem instances are solved. The results have showed that an optimized plan can be useful for supporting inventory management decisions. Additionally, productivity increases as a result of the reduced overgrading and transportation costs. Hence marble factories are recommended to use such optimization algorithms in order to maintain their operations in an efficient and effective way.

CHAPTER FOUR
STOCHASTIC DIFFUSION SEARCH ALGORITHM FOR SOLVING A REAL-WORLD MARBLE CUTTING PROBLEM

4.1 Introduction

Several solution techniques including exact and heuristic methods are applied to solve 1DCSP. However, these methods are not able to solve all of the problem instances global optimally. For that reason, researchers have extensively employed metaheuristic based approaches. In recent years, variety of metaheuristic optimization algorithms have been proposed. Also, wide variety of hybridizations of different metaheuristic methods have been studied. As the metaheuristic algorithms are able to solve large-scale and NP-hard combinatorial optimization problems successfully, in this chapter, a metaheuristic based solution approach is utilized for the one-dimensional marble plane cutting problem.

The rest of this chapter is organized as follows. In section 4.2, SDS algorithm is discussed along with the four main subtitles: Origins, algorithms, positive and negative aspects, and selection mechanisms. In section 4.3, details of the developed solution methodology are described. More specifically, initial solution generation, objective function calculation, and local search mechanisms are clarified by using various instances for the sake of understandability. In section 4.4, a real-world marble plane cutting problem instances are given, and the proposed metaheuristic based solution methodology is implemented in the problem instances which are given in the previous chapter. Finally, in section 4.5, the chapter summary and concluding remarks are provided.

4.2 SDS Algorithm

4.2.1 Origins

SDS method is based on the concept of invariant pattern recognition. Invariant pattern recognition aims to classify a pattern or to identify pattern within a large data structure. SDS employs two methods in order to perform invariant pattern recognition. The first method is called Template Matching and second method is called Hinton Mapping. According to invariant pattern matching description, SDS utilizes number of agents in order to identify the best match to a pre-defined target pattern.

4.2.1.1 Hinton Mapping

A Hinton Mapping network contains two sets of detectors. These are canonical and retinocentric. In these methods, retinocentric feature patterns are mapped into canonical feature patterns.

4.2.1.2 Template Matching

The traditional template matching uses information from the best matching template to perform classification. Larsen and Bundesen (1996) discussed the common view about insufficiency of template matching based methods for recognition of real-life patterns. They recognized that the recognition rates achieved by a traditional template matching approach underperforms human performance.

4.2.2 SDS Algorithm

SDS method is a population-based algorithm and agents evaluate a hypothesis partially. The reason is that searching the entire search space is difficult and often leads to infeasible solution. In each iteration, agents communicate to share information about the quality of their own hypothesis and other agents' hypothesis.

Agents are classified based on the quality of their hypothesis. Agents which are located in the same cluster share information more frequently. Finally, SDS makes use of the cluster information instead of individual solutions of agents. SDS is not a deterministic algorithm, as it contains several randomized steps. In standard SDS algorithm, agents operate harmoniously and realize various of stages as shown in Table 4.1.

Table 4.1 Standard SDS operation.

```
Initialize (Agents);
 Repeat
    Test (Agents);
    Diffuse (Agents);
 Until (Halting Criterion)
```

The first stage begins with initialization of the agents' parameters. There are different initialization methods.

In test stage, all agents are evaluated in terms of their hypothesis with respect to targets. Evaluation results are compared, and agents are divided into two groups based on the evaluation results as: active agents or inactive agents. This means that active agents satisfy targets whereas inactive agents do not satisfy targets. The test phase is shown in Table 4.2.

Table 4.2 Test phase with comparison of targets.

```
for agent=1 to all agents
    t1=Pick (targets);
    t2=Find (targets, agent.hypothesis);
    if (t1= = t2)
       agent.activity = TRUE;
    else
      agent.activity = FALSE;
    end
end
```

In diffusion stage, each inactive agent randomly chooses another agent from the agents set. In this case, two situation occurs: First one is that selected agent is active, then selected agent transfers its hypothesis into inactive agent. Due to the exchange of information, this phase is named diffusion. The search mechanisms are performed in the test and diffusion phases. If agents attract more agents, the chance of finding high-quality solutions increases. The second situation is that if the selected agent is inactive, in this case, agent changes its hypothesis with a random hypothesis. Since inactive agents do not communicate, there is no information flow between agents. The diffusion phase is shown in Table 4.3.

Table 4.3 Standard SDS diffusion phase.

```
for agent=1 to all agents
   if (agent.activity = = FALSE )
       agent2 =Pick-Random-Agent (Agents);
       if (agent2.activity = = TRUE)
          agent.hypothesis =agent2.hypothesis;
       else
         agent.hypothesis =Pick-Random-Hypothesis();
        end
   end
end
```

The algorithm continues until termination criteria is satisfied. Generally, termination criteria is the maximum number of iteration number. Also, there are other termination criteria in SDS algorithm such as time-based criteria, activity-based criteria, and cluster-based criteria. In time-based criteria, the algorithm run for a fixed number of iterations, or until a fixed amount of computational time is elapsed. In activity-based criteria, the process is terminated if the overall activity exceeds a certain level; this level can be predefined by the user or calculated from the overall activity level during the first iterations of the algorithm. Alternatively, the process can be terminated if the solution converges to a fixed point. In cluster-based criteria, execution of the algorithm is terminated when the stable clusters are formed. In some

cases, especially in dynamic systems, there is no predefined terminating criteria. The algorithm continues until manually terminated by the user.

4.2.3 Positive and Negative Aspects of SDS Algorithm

Standard SDS allocates plenty of its resources to the best match which has been discovered so far. This property has both positive and negative sides. From the positive perspective, standard SDS only focuses on the best solution. From the negative perspective, the solution can be trapped in suboptimal solutions and time to converge to optimal solution increases. In general, standard SDS find an optimal solution when enough period of time is given. Since, standard SDS does not use hill-climbing mechanism, it is easy to find optimal solution without using complex information such as derivative of objective function. However, the SDS algorithm suffers from the suboptimal solutions.

Standard SDS is similar to template matching approach in terms of search mechanism. For instance, if number of iterations is set to 1000 and number of agents are determined as 1000, it will require 1000000 comparisons. Another parameter of standard SDS is the number of agents. Examples from literature shows that the use of 100 or 2000 agents will generally result in similar behavior.

In the literature, many real world applications of the SDS algorithm are reported. The first real world application was conducted in British Telecom CONNEX project (Bishop, & Torr, 1992). The problem here is that locating eye positions within grey-scale images from human faces. This difficult task was solved by using the Hybrid Stochastic Network method which is combination of a Stochastic Search Network method and an n-tuple network method. Results showed that locating accurately eye features which has been trained, has performance over 60% success. However, authors concluded that the performance of the system would be improved in case of making more research on this area. There is also another real-world problem on which SDS has been applied successfully, i.e., self-localization of an autonomous wheelchair (Beattie, & Bishop, 1998).

There is a balance between exploration and exploitation. There is no standardized procedure to design a good search method for particular problems. SDS method gives good solutions to specific problems when the level of exploitation and exploration level is well-adjusted. High level of exploration without exploitation result in low-quality solutions. On the other hand, high level of exploitation and without exploration results in trapping into suboptimal solutions.

Many metaheuristic algorithms mostly use complete function evaluation during the search in the solution space. Also they include local search methods which make of use partial evaluations to update the objective function value. However, partial evaluation is a critical step in the search process of SDS algorithm and agents perform only partial evaluations in each iteration. The solution quality depends on the problem structure.

Optimization problems which have decomposable structure can be evaluated partially. If the objective function of optimization problem can be described as summation of its terms, then the problem is defined as decomposable.

Properties of SDS Algorithm

- We know from the former aforementioned, SDS is population-based method. Its solution time depends on number of agents. Performance of the method relies on the interaction of agents because single agent has no ability to assess and compare the quality of solutions. Population size is usually between 100 to 1000 agents in application.
- In the beginning of the search, most of the agents (in some cases all agents) are inactive. Information exchange among agents occurs in diffusion phase. Direction of the search is determined based on the knowledge about the objective function value. So, interactions among agents lead to active agents which direct the moves. Note that, in test and diffusion phases it is assumed that agents move rationally.

4.2.4 Selection Mechanisms of SDS Algorithm

In SDS, agents choose to reject, retain or replicate a hypothesis based on information received other agents. There are different selection strategies which are based on the comparison of activity levels, hypothesis parameters, or in some cases both of them. Assume that there exist two agents namely agent A and agent B, then one of the following cases occurs:

1. The activity levels of agent A and B are both low. In this case, agent A ought to reject its own hypothesis and should not accept the hypothesis of agent B.
2. Agent A has a low activity level, but agent B has a high activity level. In this case, agent A will reject its own hypothesis and accept the hypothesis of agent B. Since agent B has a high level of activity.
3. Agent A has a high activity level, but agent B has a low activity level. In this case, due to the rationality assumption in test and diffusion phases, agent A retains its own hypothesis.
4. The activity levels of agent A and B are both high. In this case, two different situations occur. These are: agent A may retain its own hypothesis or agent A rejects its own hypothesis.

4.3 The Proposed Constructive SDS Algorithm

In this section, the proposed constructive SDS algorithm framework and the principal algorithms for the initial solution and neighborhood solution generation for constructive SDS algorithm are presented. The general steps of the proposed constructive SDS algorithm are presented in Table 4.4.

Table 4.4 Proposed constructive SDS algorithm

1. Initialize population with initial solution construction procedure

2. Apply calculate fitness procedure

While (stopping criterion not met)

3. Agents are evaluated through their fitness

3.1 Best half results are named active

3.2 Remain half results are named passive

4. For passive agents choose randomly number between (1, number of agents)

4.1 If passive agents are chosen take a randomly number between (1,4) and apply following operator:

If 1 is taken apply single swap operator

If 2 is taken apply two block swap operator

If 3 is taken apply three block swap operator

If 4 is taken apply inversion operator

4.2 If active agents are chosen apply lox crossover operator

End While

4.3.1 Solution Representation

Two different solution representations are used in this problem. The first one is used to represent order of marble blocks which are available in stock area, and second one is used to represent the selected marble blocks order with the information of simultaneously entering marble blocks for cutting process. The first representation X is represented as a vector that consists of a sequence of marble blocks, with dimension D=m, where m is the number of marble blocks available in stock area. This encoding is so-called permutation encoding. The second solution representation Y for the marble plane cutting problem is represented as a matrix, with dimension 2 x (n), where value 2 shows which marble blocks are being cut simultaneously together, otherwise it takes 0 value and n is the less than or equal to the selected

block number that consists of selected marble blocks which are enough for satisfying the customer orders value. Because some of the marble blocks are being cut together, value of n is smaller than selected marble block number. Solution representation of X for a following example problem with four customer orders and twenty marble blocks is given in Figure 4.1. Example problem data are given in Tables 4.5-4.6.

Table 4.5 Marble block data of example problem

Block_No	Quality of block	Height of block (cm)	Width of block (cm)
B1	3	150	150
B2	2	150	165
B3	3	165	110
B4	2	155	100
B5	1	150	125
B6	2	160	125
B7	3	155	140
B8	1	155	150
B9	2	155	100
B10	3	150	170
B11	2	150	150
B12	3	165	120
B13	1	150	100
B14	2	150	190
B15	3	160	200
B16	2	155	120
B17	1	155	130
B18	1	150	145
B19	1	160	120
B20	2	150	160

Table 4.6 Customer order data of example problem

Customer_No	C1	C2	C3	C4
Customer_Order Number	50	30	30	30
Quality	2	1	3	2
Width	4	6	5	3

In this example, there exist twenty marble blocks in stock area which are represented by X = [B1, B12, B2, B9, B19, B13, B3, B11, B15, B18, B10, B8, B17, B20, B16, B4, B5, B6, B14, B7]. Marble block B1 is in the first order and B12 is in the second order, B2 is the next etc. according to sequence.

B1	B12	B2	B9	B19	B13	B3	B11	B15	B18	B10	B8	B17	B20	B16	B4	B5	B6	B14	B7

Figure 4.1 Solution representation of X

When cutting marble blocks for satisfying customer order, marble blocks are selected according to solution representation of X. In this problem example, firstly marble block with block no 1 (B1) is being cut first. Since the quality level of this block is level 3, this block can be only used for satisfying customer order with quality level 3. Customer 3 (C3) has quality level 3, so this demand can be met by cutting B1. Marble block B1 has enough width to satisfy all the demand of C3. In the second order, there is marble block B12. However, this block cannot be cut. Since its quality level 3, and there exist no left customer order with quality level 3. We pass this marble block and search for the following marble which is block B2. Marble block B2 has quality level 2, and there exist customer orders which demand quality level 2. So marble block B2 is being cut and 41 pieces of Customer 1 (C1) demand are satisfied and left 1cm as a trim-loss. Now we check the machine eligibility constraints for the selected blocks whether they can be simultaneously cut or not. Since marble block B1 and B2 have total width 315 cm which is over the machine capacity constraint 275 cm, they cannot be cut simultaneously. In other words, they will be cut in the different order. Since there exists unsatisfied demand, we cut marble blocks according to chosen solution representation order. There exists marble

block B9 in order. Marble block B9 has quality level 2 and there exist customer orders which demand quality level 2. So marble block B9 is being cut and 9 pieces of unmet Customer 1 (C1) demand in the previous order and 21 pieces of Customer 4 (C4) demand are satisfied, and 1cm is left as a trim-loss. Now we check the machine eligibility constraints for the block B9 whether it can be simultaneously cut or not. Since marble block B1 and B9 have total width 250 cm which is convenient for the machine width capacity constraint 275 cm and also the height differences between marble block B1 and B9 is 5 cm which is smaller than 10 cm height difference constraint. So, they can be cut simultaneously. In other words, they are cut in the same order. Since there exists unsatisfied demand, we will to cut the marble blocks according to chosen solution representation order. There exists marble block B19 in order. Marble block B19 has quality level 1 which satisfy all customer order quality levels. So marble block B19 is being cut and 20 pieces of Customer 2 (C2) demand with quality level 1 is satisfied. Again we check the machine eligibility constraints for the block B19 whether it can be simultaneously cut or not. Since marble block B2 and B19 have total width 285 cm which is over the machine capacity constraint 275 cm, they cannot be cut simultaneously. In other words, they will be cut in the different order. Since there exists unsatisfied demand we will cut marble blocks according to chosen solution representation order. There exists marble block B13 in the order. Marble block B13 has quality level 1 which satisfy all customer order quality level. So marble block B13 is being cut and 10 pieces of unmet Customer 2 (C2) demand with quality level 1 in the previous order and 9 pieces of Customer 4 (C4) demand with quality level 2 is satisfied and 13 cm is left as retail. Here, satisfaction of 9 pieces Customer 4 (C4) demand with quality level 2 cause the overgrading cost. Again we check the machine eligibility constraints for the block B13 whether it can be simultaneously cut or not. Since marble block B2 and B13 have total width 265 cm which is convenient for the machine width capacity constraint 275 cm, they are cut simultaneously. In other words, they will be cut in the same order. Since there is no unsatisfied demand we will stop cutting marble blocks and the second solution representation Y is formed as given in Figure 4.2.

Block_1	B1	B2	B19
Block_2	B9	B13	0

Figure 4.2 Solution representation of Y

From solution representation of Y given in Figure 4.2, it is observed that B1 and B9 are being cut together. In the second order, marble blocks B2 and B13 are being cut together, and finally marble block B19 is being cut in the third order.

4.3.2 Generating Initial Solution

Generally, two different initial solution construction strategies are employed by the researchers for the CSPs. The first strategy is to use a heuristic method to generate an initial solution, and the second one is to construct it randomly. A heuristic algorithm initiates the search process from a better quality solution, and it can accelerate the convergence of the search method. But, it has a negative effect on the diversification of the search by preventing the algorithm from exploring different search regions. For this reason, we use a random solution construction procedure which is given in Table 4.7

Table 4.7 Initial solution construction procedure

```
Procedure initial_solution_generator
Input: n = Available number of marble block in stock
       m = Number of agents used in the algorithm
Output: Initial solutions X
Initialize the parameters marble blocks in stock (n) and initialize the agents list AL = {1, . . ., m};
While exit == 0
    Sequentially, select an agent from ordered list AL;
    Generate random permutation of the integers from 1 to n;
        While generated permutation list is different the previous ones
            Assign this created permutation list to the selected agent;
            Remove selected agent from list AL;
        end
   if AL == 0;
        exit ←1;
   end
end
```

The procedure starts with initialization of the parameters, such as number of available marble blocks in stock, and the agent list AL is constructed. Then, random permutation will generate of the integers from 1 to number of marble blocks. This random permutation will assign agents according to order in AL list. When the random permutation becomes same as the previous ones, in this case, new random permutation will be generated. This procedure continues until no agents remains in AL.

4.3.3 Calculate Fitness Procedure

In the application of this algorithm, we use fitness function which is calculated as explained in Table 4.8. Our objective function is to minimize the cost of overgrading and transportation.

Table 4.8 Calculate fitness procedure

Step 0. //Read data related with marble blocks available in stock, customer order and cost
$CutCost_{ij}$ = Block cost;
TCost = Transportation cost;
WM = Gangsaw machine width;
k_i = Block quality;
H_i = Block height;
Wb_i = Block width;
q_j = Customer order quality;
D_j = Total number of customer order amount;
Wt_j = Customer order width;
Step 1. //Set parameters of algorithm in machine eligibility constraint
δ = Height difference between simultaneously cutting marble blocks;
L_i= Threshold value for decision of trim or leftover for marble block *i*
Step 2. //Initialize variables
Leftover of marble block *i* = Wb_i;
Unmetdemand = D_j;
Used marble blocks = [ ];
Customer demand = D_j;

Table 4.8 Calculate fitness procedure (continue)

Step 3. //Determination of which marble block will cut
for j = 1: length (number of marble block)
while Unmet demand ~ = 0
if $k_i <= q_j$
//There exist marble blocks in stock area to meet the customer order and create candidate marble block set CMB which will be cut
else
// These customer order does not meet so algorithm will terminate
endif
while Leftover of (CMB(marble block(j))) ~ = 0
if length (customer order demand sets) > 0
if $Wt_j * D_j$ (customer order demand (j)) <= Leftover of (CMB(marble block(j)))
Amount of meeting customer order demand (j) = Total number of customer order demand (j);
Amount of unmet demand = 0;
Trim-loss (marble block(j)) = Leftover of (marble block(j))-amount of meeting customer order demand(j) *Wt_j (customer order demand (j));
Leftover of (marble block(j)) =Leftover of (marble block(j))-amount of meeting customer order demand *Wt_j (customer order demand (j));
Cost1 = $CutCost_{ij}$ * amount of meeting customer order demand(j);
else // $Wt_j * D_j$ (customer order demand (j)) > Leftover of (CMB(marble block(j)))
Amount of meeting customer order demand (j) =fix (Leftover of(marble block(j)) /Wt_j (CMB (marble block(j))) ;
Amount of unmet demand= D_j-Amount of meeting customer order demand(j);
New demand= Amount of unmet demand; //update demand
Trim-loss (marble block(j)) = Leftover of (marble block(j))-amount of meeting customer order demand(j) *Wt_j (customer order demand (j));
Leftover of (marble block(j)) = 0;
Cost2 = $CutCost_{ij}$ * amount of meeting customer order demand(j);
endif
if (customer order demand (j)) == 0
Remove this customer order demand from customer order demand set
endif
endif
j=j+1;
endwhile
endwhile
endfor
Cost = Cost1+Cost2;
Total cost = All cost until the customer order demand set empty
Step 4. //Determination of being cut marble blocks
if Leftover of (marble block(j)) ~= initial leftover of (marble block(j));
Used marble blocks = (marble block(j));
endif
Step 5. / /Determination of being cut simultaneously marble blocks in gangsaw machine
if Used marble blocks (i) < Used marble blocks (j) // Prevent symmetry in the solution representation
if Wb_i (used marble blocks (i)) +Wb_j (used marble blocks (j)) <= WM and
|H_i (used marble blocks (i))-H_j (used marble blocks (j)) |<= δ
Construct being cut simultaneously marble block sets CSMB
endif
endif
Step 6. // Construct solution representation of Y and calculate number of marble blocks which simultaneously being cut
Step 7. // Calculate fitness value
Total TCost = TCost * number of used marble blocks;
Fitness = Fitness-Total TCost *number of marble blocks which simultaneously being cut +Total TCost;

4.3.4 Neighborhood Structures

In this study, five neighborhood structures are applied in SDS. These are single swap, two swap block, three swap block, inversion, and lox crossover operators. Because of our new methodology, implementation of new operators has become necessary, such as lox crossover operators. Only an overview of the operators has been given in this section.

4.3.4.1 Single Swap

A swap-move SWP (X, i, j) swaps the positions of two configurations in the current arrangement X, which is given in Figure 4.3. Configuration x_i at position i is exchanged with configuration x_j at position j. The configurations at all other positions are unaffected by this move. Assume that the randomly chosen positions are i = 3 and j = 7.

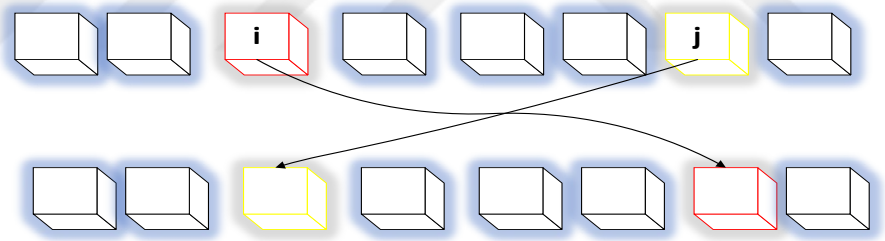


Figure 4.3 Single swap move

4.3.4.2 Two Block Swap

Two swap block is aimed at exchanging two different blocks with a length two in the solution. The positions of the two blocks are randomly selected and the lengths of these two blocks are equivalent. Figure 4.4 portrays an example of this neighborhood structure. Assume that the randomly chosen positions are 2 and 5. That means that the two framed blocks (7,5) and (3,6), which are depicted in Figure 4.4, are randomly selected. Then, these two blocks are swapped to produce a new solution.

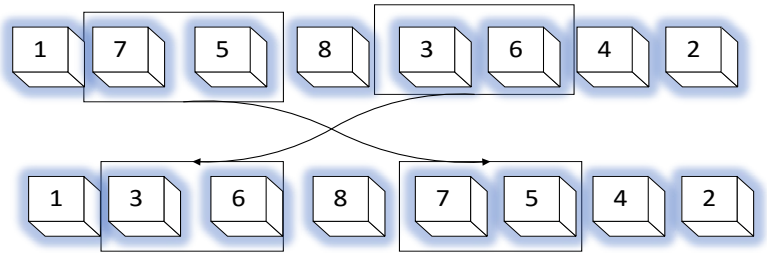


Figure 4.4 Two block swap move

4.3.4.3 Three Block Swap

Three swap block is aimed at exchanging two different blocks with a length three in the solution. The positions of the two blocks are randomly selected and the lengths of these two blocks are equivalent. Figure 4.5 portrays an example of this neighborhood structure. Assume that the randomly chosen positions are 3 and 9. That means the two highlighted blocks (9,4,7) and (2,6,10), which are depicted in Figure 4.5, are randomly selected. Then, these two blocks are swapped to produce a new solution.

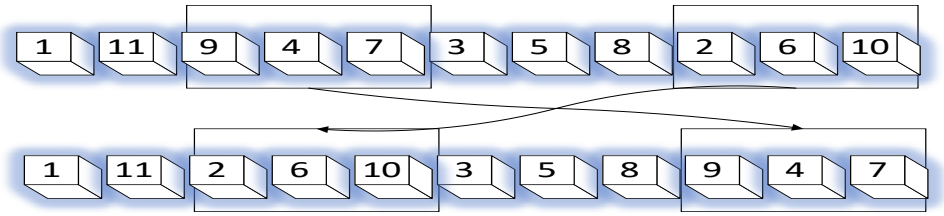


Figure 4.5 Three block swap move

4.3.4.4 Inversion

The inversion move is a unary operator where the elements between two random crosspoints are reversed. Figure 4.6 portrays an example of this neighborhood structure. Assume that the randomly chosen crosspoint positions are 4 and 7. An example of the inversion operator is presented where the job block (3, 2, 9, 6) is inverted.

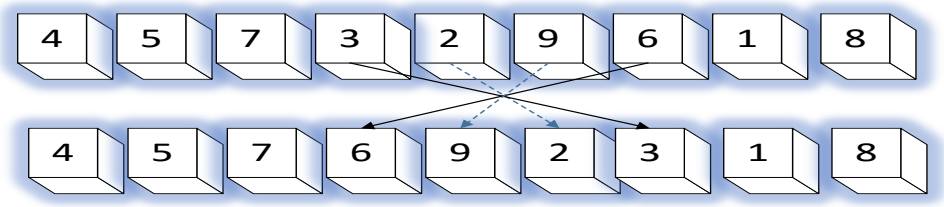


Figure 4.6 Inversion move

4.3.4.5 Lox Crossover

Crossover LOX, which was initially proposed by Falkenauer and Bouffouix (1991), works as follows:

Step 1. Select a subsequence of operations from one parent at random.

Step 2. Produce a proto-offspring by copying the subsequence into the corresponding positions of it.

Step 3. Delete the operations which are already in the subsequence from the second parent. The resulted sequence of operations contains the operations that the proto-offspring needs.

Step 4. Place the operations into the unfixed positions of the proto-offspring from left to right according to the order of the sequence to produce an offspring. This procedure is illustrated in Figure 4.7. It gives an example of producing one offspring.

Crossover LOX tries to preserve as much as possible both the relative positions between genes and the absolute positions relative to the extremities of parents. The extremities correspond to the high- and low-priority operations.

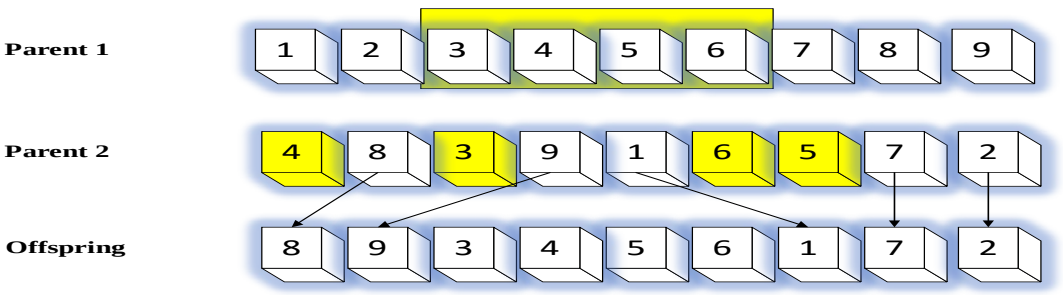


Figure 4.7 Lox crossover move

4.4 Application of the Proposed Solution Method to a Real-World Marble Cutting Problem Examples

In order to assess the empirical performance of the proposed SDS algorithm, the computational experiments have been carried out on a set of previously generated 5 problem instances. As previously mentioned, various parameters affect the performance of the SDS algorithm. Therefore, different number of agents is tested in all problem instances with fixed number of iteration number, which is equal to 400. Computational results for problem instances are given in Table 4.9.

The values assigned to the agents affect the computational time of the SDS algorithm which are reported in Table 4.9 for each group of problem instances.

The percentage reduction in objective function value for MILP model of marble cutting plane has been compared with proposed constructive SDS algorithm. Percentage difference values are calculated based on Equation 4.1. Calculation results are shown in Table 4.9. The proposed MILP model has been coded and run via LINGO 14.0.

$$\text{Percentage difference} = \frac{SDS\ Best\ sol - Optimal\ Value\ of\ MIP}{Optimal\ Value\ of\ MIP} x100 \quad (4.1)$$

Table 4.9 Computational results for problem instances

Problem	Number of Agent	Proposed SDS Best sol	Computational Time SDS	Optimal Value MILP	Computational Time MILP	Percentage Difference
Instance1	50	24100 TL	4.669915 sec	24100 TL	4746 sec	-
	100	24100 TL	8.616764 sec	24100 TL	4746 sec	-
	200	24100 TL	17.155058 sec	24100 TL	4746 sec	-
Instance2	50	21650 TL	3.998300 sec	21650 TL	4311 sec	-
	100	21650 TL	7.766756 sec	21650 TL	4311 sec	-
	200	21650 TL	15.536897 sec	21650 TL	4311 sec	-
Instance3	50	16510 TL	3.390022 sec	16500 TL	243 sec	0.0606
	100	16510 TL	6.667798 sec	16500 TL	243 sec	0.0606
	200	16510 TL	13.388071 sec	16500 TL	243 sec	0.0606
Instance4	50	18220 TL	3.592465 sec	18200 TL	1218 sec	0.1098
	100	18220 TL	6.506186 sec	18200 TL	1218 sec	0.1098
	200	18220 TL	12.944093 sec	18200 TL	1218 sec	0.1098
Instance5	50	9600 TL	2.520171 sec	9600 TL	109 sec	-
	100	9600 TL	4.983395 sec	9600 TL	109 sec	-
	200	9600 TL	9.426879 sec	9600 TL	109 sec	-

The proposed SDS has found the optimal solution for the three of the five test problem instances. Accordingly, the proposed SDS algorithm has produced better quality solutions in a relatively short computation times.

4.5 Chapter Summary

This research introduces the marble cutting plane problem. The problem incorporates additional real world constraints such as machine eligibility constraints. In previous chapter, a MILP model has been developed that addresses the problem of marble cutting plane. Since the problem of interest is NP-hard, a SDS algorithm is proposed to solve the developed model. Hence, in this study, for the first time in the literature, a real life 1DCSPUL has been solved. Therefore, five problem instances have been generated to investigate the performance of the proposed SDS. The results of the SDS algorithm have been compared with the optimum values obtained from the MILP model. The SDS algorithm has produced better quality solutions in rather short computing times for the all problem instances.

CHAPTER FIVE
A STOCHASTIC DIFFUSION SEARCH ALGORITHM FOR THE ONE-DIMENSIONAL MARBLE CUTTING STOCK PROBLEM WITH SETUPS

5.1 Introduction

C&P literature is still rapidly growing. Wäscher et al. (2007) introduced the criteria for a typology of C&P problems to provide a tool for the systematic organization and categorization of existing and new problem types. In the classical C&P problems, there exist cutting patterns and the frequency of each pattern occurrence. The objective function value is calculated by using cutting patterns and frequencies. From the practical point of view, however, C&P problems are integrated with some additional problems. For instance, sequencing of cutting patterns, lot-sizing or due date determination. These problems are not independent from the C&P problems and should be handled simultaneously.

In this chapter, we focus on the issue of setups which arise in industrial cutting processes whenever a new cutting pattern is started and the cutting equipment is prepared in order to meet the technological requirements of the new pattern. Setups of this kind involve the loss of production time capacity and the consumption of resources, which can be avoided or, at least, be reduced by taking into account these consequences.

Although, in real-world cutting problems, setups are an important issue, in the literature there exist few studies related to C&P problems. Some of the mathematical models take into account setups. Most of them define setup problems depending on extension of Single Size Cutting Stock Problem (SSSCSP). SSSCSP was defined by the typology of Wäscher et al. (2007). In the next section, literature review of the setups in the CSPs is given to emphasize theoretical deficiencies.

5.2 Solution Approaches

Thus it is not surprising that when one has to deal with setups, certain heuristic approaches are often used. These approaches have been popular in the early days of applying operation research methods for cutting problems. One of the most important practical requirements in CSP is setups.

Garraffa et al. (2016) presented a new generalization of the 1DCSP. In this problem, main consideration was that trim-loss depended on items' cutting order. Specifically, pattern-based heuristic (HSD) approach was developed for solving the sequence-dependent cut losses (SDCL) and two industrial cases were solved in joinery industry and in roofing industry. Results were showed that, HSD heuristic was useful.

Cui et al. (2015) developed integer programming model for 1DCSP that minimize the both material cost and setup cost. They introduced sequential grouping method to create the cutting patterns in the set. Proposed model was tested on two different sets of benchmark instances. Computational results were showed that the developed approach was efficient in terms of solution quality.

Burkard & Zelle (2003) considered a (one-dimensional) reel cutting problem in the paper industry which was characterized by several specific pattern constraints (maximum number of short items/reels in the pattern, maximum number of long items/reels in the pattern, minimum width of the pattern). They introduced a randomized local search (RLS) heuristic which was applied in two phases: In the first phase a near-optimal solution to the equality-constrained was determined by means of move, swap then in a second phase, the RLS heuristic – limited to a subset of the neighborhood moves – was applied in order to reduce the number of cutting patterns.

5.3 Proposed Constructive SDS Algorithm

The proposed constructive SDS algorithm framework for the one-dimensional marble cutting stock problem with setups for the initial solution and neighborhood solution generation are the same as given in the previous chapter. The only difference is in the calculation fitness procedure because in this case our objective function is to minimize the cost of overgrading, setups and transportation. Fitness function calculation procedure is given in Table 5.1

Table 5.1 Calculate fitness procedure for knife dependent setup

Step 0. //Read data related with marble blocks available in stock, customer order demand and costs
KNCost=Knife change cost;
TCost =Transportation cost;
$CutCost_{ij}$ =Block Cost;
WM =Gangsaw machine width;
k_i =Block quality;
H_i =Block height;
Wb_i =Block width;
q_j = Customer order quality;
D_j =Total number of customer order demand amount;
Wt_j =Customer order demand width;
Step 1. //Set parameters of algorithm in machine eligibility constraint
δ = Height difference between simultaneously cutting marble blocks;
L_i= Threshold value for decision of trim or leftover for marble block *i*;
Step 2. //Initialize variables
Leftover of marble block $i = Wb_i$;
Unmetdemand=D_j;
Used marble blocks= [ ];
Customer order demand=D_j;

Table 5.1 Calculate fitness procedure for knife dependent setup (continue)

Step 3. //Determination of which marble block will cut
for j=1: length (Number of marble block)
while (Unmet demand ~= 0) // Continue until the customer order demand meet
if $k_i <= q_j$
//There exist marble blocks in stock area to meet the customer order and
create candidate marble block set CMB will be cut
else
// These customer order does not meet so algorithm will terminate
endif
while Leftover of (CMB (marble block(j))) ~ = 0
if length (customer order demand sets) > 0
if $Wt_j * D_j$ (customer order demand (j)) <= Leftover of (CMB(marble block(j)))
Amount of meeting customer order demand (j)=Total number of
customer order demand (j);
Amount of unmet demand = 0;
Trim-loss (marble block(j)) = Leftover of (marble block(j))-amount of meeting
customer order demand(j) *Wt_j (customer order demand (j));
Leftover of (marble block(j)) = Leftover of(marble block(j))-amount of meeting
customer order demand *Wt_j (customer order demand (j));
Cost1 = $CutCost_{ij}$ * amount of meeting customer order demand(j);
else // $Wt_j * D_j$ (customer order demand (j))> Leftover of (CMB(marble block(j)))
Amount of meeting customer order demand(j) =fix(leftover of (marble block(j))
/Wt_j (CMB(marble block(j)));
Amount of unmet demand = D_j-Amount of meeting customer order demand (j);
New demand = Amount of unmet demand; //update demand
Trim-loss (marble block(j)) = Leftover of (marble block(j))-amount of meeting
customer order demand(j) *Wt_j (customer order demand (j));
Leftover of (marble block(j)) = 0;
Cost2 = $CutCost_{ij}$ * amount of meeting customer order demand(j);
endif
if (customer order demand (j)) == 0
Remove this customer order demand from customer order demand set
endif
endif
j=j+1;
endwhile
endwhile
endfor
Cost = Cost1+Cost2;
Total cost = All cost until the customer order set empty
Step 4. //Determination of being cut marble blocks
if Leftover of (marble block(j)) ~= initial leftover of (marble block(j));
Used marble blocks = marble block(j);
endif
Step 5. //Determination of being cut simultaneously marble blocks in gangsaw machine
if Used marble blocks (i) < Used marble blocks (j) // Prevent symmetry in the solution representation
if Wb_i (used marble blocks (i)) +Wb_j (used marble blocks (j)) <= *WM* and
|H_i (used marble blocks (i)) -H_j (used marble blocks (j)) | <= δ
Construct being cut simultaneously marble block sets CSMB
endif
endif
Step 6. // Construct solution representation of Y and calculate number of marble blocks which simultaneously being cut
Step 7. // Calculate number of used knife in each marble blocks to satisfy each customer order demand type and set up occurs when changing the number of knifes
Step 8. // Calculate fitness value
Total Knifechangecost = *KNCost**Total Knifechange;
Total TCost = TCost * number of used marble blocks;
Fitness = Fitness- TCost * number of marble blocks which simultaneously cutted +Total cost+ Total Knifechangecost;

5.4 Application of the Proposed SDS Algorithm to a Real-World Marble Cutting Problem Examples

In order to assess the empirical performance of the proposed SDS algorithm for marble plane cutting problem with knife dependent setup cost, the computational experiments have been carried out on a set of previously generated 5 real problem instances.

As previously mentioned various parameters affect the performance of the SDS algorithm. Therefore, different number of agents is tested in all problem instances with fixed number of iteration number which is equal to 400. Computational results for the problem instances are given in Table 5.2.

The values assigned to the agents affect the computational time of the SDS algorithm which are reported in Table 5.2 for each group of problem instances.

Table 5.2 Computational results for problem instances

Problem	Number of Agent	Proposed SDS Best sol	Computational Time SDS
Instance1	50	27380 TL	6.673211 sec
	100	27050 TL	13.235716 sec
	200	26490 TL	26.406398 sec
Instance2	50	24780 TL	6.698280 sec
	100	24780 TL	12.579302 sec
	200	24780 TL	25.165535 sec
Instance3	50	23900 TL	6.035512 sec
	100	23510 TL	12.175807 sec
	200	23280 TL	24.461931 sec
Instance4	50	20840 TL	5.664228 sec
	100	20840 TL	11.109076 sec
	200	20810 TL	22.289492 sec
Instance5	50	10810 TL	4.309645 sec
	100	10720 TL	7.972833 sec
	200	10720 TL	15.447974 sec

5.5 Chapter Summary

In this chapter, we have studied the integrated cutting and pattern sequencing problem. In the previous chapter, setups have been neglected and cutting orders have been determined randomly. However, setups cost occurs when changing from one cutting pattern to another. In the marble cutting plane problem, setup cost depends on the total number of customer order types in the patterns. Because in each pattern knifes are set according to assigned number of customer demand. We have proposed a SDS based algorithm to solve the problem. The proposed method has been tested on the previously generated real problem instances and results are satisfying and have been solve reasonable time.

CHAPTER SIX
CONCLUSION

6.1 Summary

This dissertation deals with the 1DCSPUL by taking into account the machines eligibility constraints. The problem has been implemented in a real life marble industry. The problem incorporates additional real world constraints, such as machine eligibility constraints, knife portion in cutting process, and transportation constraints. Due to the significant role of the 1DCSP in real life situations, this problem has been studied extensively within the scope of this thesis.

We firstly have developed a MILP model by considering the cutting process of marble plane in the 1DCSPUL in order to formally describe the problem. However, due to the NP-Hard nature of the problem, the proposed MILP model has not been able to solve large-scale problems. Therefore, we have developed metaheuristic based algorithms in order to tackle the problem. We have selected stochastic SDS. Specifically, effective solution procedures based on stochastic diffusion search algorithms have been developed. Computational experiments have been carried out in order to evaluate the performance of the developed MILP and metaheuristic based models.

Although, setups are an important issue in real-world cutting problems, the literature lacks models and applications dealing with the setups. Therefore, the main goal of this dissertation is to consider the CSP and sequencing problem simultaneously in a cutting process. The objective is to minimize transportation, overgrading, and knife dependent setup costs. The main endeavor of this study is to specifically develop SDS based solution approach to solve the one-dimensional marble plane cutting problem with usable leftover by taking into sequence dependent setups cost between cutting marble blocks. Computational experiments are conducted in order to assess the performance of the proposed models. Also, a real-world case instances have been solved and satisfactory results have been obtained.

6.2 Contribution of the Study

In Chapter 3, we have introduced marble cutting plane problem, which is a major problem of marble processing industry, is handled. The one main cutting operation, which is the cutting marble blocks into planes, is considered. MILP model has been developed for the solution of the problem. One of the salient features of this problem is that it does not require a priori enumeration of all possible cutting patterns. The MILP provides us with the formal representation of the problem. The results have showed that efficiency of cutting process can be increased considerably compared to the current industry average. Furthermore, we can conclude that our MILP can be generalized for solving wide variety cutting problems encountered in the marble industry.

In Chapter 4, we have proposed a SDS algorithm. The efficiency of the proposed metaheuristic algorithm is examined by a computational study on a set of real problem instances. The SDS algorithm has been used with the purpose of improving the diversification of the algorithm, which refers to the ability to visit many and different regions of the search space. Moreover, we have used a new neighborhood structure by utilizing lox crossover which tries to preserve as much as possible both the relative positions between genes and the absolute positions relative to the extremities of parents. We have adopted a mechanism which provides the communication between different agents in the proposed algorithm in order to share the information. Since information sharing is an essential issue from the optimization point of view, this neighborhood structure stimulates shared knowledge between the agents.

In Chapter 5, we deal with knife dependent setup cost of one-dimensional marble plane cutting problem with usable leftover which is a specific case of the well-known 1DCSP. Even though the literature in the area of C&P is growing rapidly, majority of the studies focus on standard problems and practical aspects have not issued in detail. In industrial cutting processes, setups cost occur whenever changing from one cutting pattern to another, and the cutting equipment has to be prepared in order to

meet technical requirements of the new pattern. Setups involve the consumption of resources and the loss of production time. Therefore, CSP and sequencing problem should simultaneously be considered in a cutting process. The objective here is to minimize transportation, overgrading, and knife dependent setup costs. The main endeavor of this study is to specifically developed a mathematical model and a SDS based approach to solve the one-dimensional marble plane cutting problem with usable leftover by taking into sequence dependent setups cost between cutting marble blocks. Computational experiments have been conducted to assess the performance of metaheuristic by solving real problem instances and promising results are obtained.

6.3 Future Works

In Chapter 3, we have developed a MILP model for one-dimensional marble plane cutting problem with regard to some characteristics. In further researches, we might extend the proposed MILP so as to solve by taken into account other dimensions such as height and/or thickness.

Especially, the need of developing solution methods by taking into account multiple objectives such as minimization of trim-loss and maximization of profit.

In most of the real-world situations, some of the input parameters of the model coefficients might not be exactly known. For instance, stochastic demands, interval type of quality degree, and unpredicted costs affect robustness and quality of the solution. Therefore, there is a growing attention among researchers to incorporate modelling of all relevant vague or uncertain information involved in a real decision problem. Therefore, modeling uncertainties by using stochastic or/and fuzzy optimization techniques give more correct and acceptable solutions.

REFERENCES

Abuabara, A., & Morabito, R. (2009). Cutting optimization of structural tubes to build agricultural light aircrafts. *Annals of Operations Research, 169*(1), 149-165.

Arbib, C., Marinelli, F., & Ventura, P. (2016). One-dimensional cutting stock with a limited number of open stacks: bounds and solutions from a new integer linear programming model. *International Transactions in Operational Research, 23*(1-2), 47-63.

Armbruster, M. (2002). A solution procedure for a pattern sequencing problem as part of a one-dimensional cutting stock problem in the steel industry. *European Journal of Operational Research, 141*(2), 328-340.

Beattie, P. D., & Bishop, J. M. (1998). Self-localisation in the 'senario'autonomous wheelchair. *Journal of Intelligent and Robotic Systems, 22*(3-4), 255-267.

Belov, G., & Scheithauer, G. (2003). *The number of setups (different patterns) in one-dimensional stock cutting*. Preprint MATH-NM-15-2003, Department of Mathematics, Dresden University of Technology.

Belov, G., & 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), 85-106.

Belov, G., & Scheithauer, G. (2007). Setup and open-stacks minimization in one-dimensional stock cutting. *INFORMS Journal on Computing, 19*(1), 27-35.

Bischoff, E. E., & Wäscher, G. (1995). Cutting and packing. *European Journal of Operational Research, 84*(3), 503-505.

Bishop, J. M., & Torr, P. (1992). The stochastic search network. *In Neural networks for vision, speech and natural language.* Netherlands: Springer

Burkard, R. E., & Zelle, C. (2003). A local search heuristic for the reel cutting problem in paper production. Universität Graz/Technische Universität Graz. SFB F003-Optimierung und Kontrolle.

Cerqueira, G. R. L., & Yanasse, H. H. (2009). A pattern reduction procedure in a one-dimensional cutting stock problem by grouping items according to their demands. *Journal of Computational Interdisciplinary Sciences*, *1*(2), 159-164.

Chauhan, S. S., Martel, A., & D'Amour, S. (2008). Roll assortment optimization in a paper mill: an integer programming approach. *Computers & Operations Research*, *35*(2), 614-627.

Cheng, C. H., Feiring, B. R., & Cheng, T. C. E. (1994). The cutting stock problem—a survey. *International Journal of Production Economics*, *36*(3), 291-305.

Cherri, A. C., Arenales, M. N., & Yanasse, H. H. (2009). The one-dimensional cutting stock problem with usable leftover–A heuristic approach. *European Journal of Operational Research*, *196*(3), 897-908.

Cui, Y., Zhong, C., & Yao, Y. (2015). Pattern-set generation algorithm for the one-dimensional cutting stock problem with setup cost. *European Journal of Operational Research*, *243*(2), 540-546.

de Carvalho, J. V., & Rodrigues, A. G. (1995). An LP-based approach to a two-stage cutting stock problem. *European Journal of Operational Research*, *84*(3), 580-589.

Degraeve, Z., & Vandebroek, M. (1998). A mixed integer programming model for solving a layout problem in the fashion industry. *Management Science*, *44*(3), 301-310.

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

Eiselt, H. A., & Sandblom, C. L. (2000). Heuristic algorithms. *In Integer Programming and Network Models*. Berlin Heidelberg: Springer

Erjavec, J., Tomat, L., & Gradišar, M. (2011, July). Expansion of the basic EOQ model with inclusion of trim-loss costs. *In Proceedings of the 5th international conference on Applied mathematics, simulation, modelling* (pp. 126-128). World Scientific and Engineering Academy and Society (WSEAS).

Falkenauer, E., & Bouffouix, S. (1991, April). A genetic algorithm for job shop. *In Robotics and Automation*, 1991. Proceedings., *1991 IEEE International Conference* on (pp. 824-829). IEEE.

Ferreira, J. S., Neves, M. A., & Castro, P. F. (1990). A two-phase roll cutting problem. *European Journal of Operational Research*, *44*(2), 185-196.

Garraffa, M., Salassa, F., Vancroonenburg, W., Vanden Berghe, G., & Wauters, T. (2016). The one-dimensional cutting stock problem with sequence-dependent cut losses. *International Transactions in Operational Research*, *23*(1-2), 5-24.

Gau, T., & Wäscher, G. (1995). CUTGEN1: A problem generator for the standard one-dimensional cutting stock problem. *European Journal of Operational Research*, *84*(3), 572-579.

Gilmore, P. C., & Gomory, R. E. (1961). A linear programming approach to the cutting-stock problem. *Operations Research*, *9*(6), 849-859.

Gilmore, P. C., & Gomory, R. E. (1963). A linear programming approach to the cutting stock problem-Part II. *Operations Research*, *11*(6), 863-888.

Gilmore, P. C. (1979). Cutting stock, linear programming, knapsacking, dynamic programming and integer programming, some interconnections. *Annals of Discrete Mathematics*, *4*, 217-235.

Gradišar, M., Jesenko, J., & Resinovič, G. (1997). Optimization of roll cutting in clothing industry. *Computers & Operations Research*, *24*(10), 945-953.

Gradišar, M., Resinovič, G., & Kljajić, M. (1999). A hybrid approach for optimization of one-dimensional cutting. *European Journal of Operational Research*, *119*(3), 719-728.

Gradišar, M., Resinovič, G., & Kljajić, M. (2002). Evaluation of algorithms for one-dimensional cutting. *Computers & Operations Research*, *29*(9), 1207-1220.

Gradisar, M., & Trkman, P. (2005). A combined approach to the solution to the general one-dimensional cutting stock problem. *Computers & Operations Research*, *32*(7), 1793-1807.

Gramani, M. C. N., & França, P. M. (2006). The combined cutting stock and lot-sizing problem in industrial processes. *European Journal of Operational Research*, *174*(1), 509-521.

Golfeto, R. R., Moretti, A. C., & Salles Neto, L. L. D. (2009). A genetic symbiotic algorithm applied to the one-dimensional cutting stock problem. *Pesquisa Operacional*, *29*(2), 365-382.

Goulimis, C. (1990). Optimal solutions for the cutting stock problem. *European Journal of Operational Research*, *44*(2), 197-208.

Haessler, R. W., & Sweeney, P. E. (1991). Cutting stock problems and solution procedures. *European Journal of Operational Research, 54*(2), 141-150.

Hajizadeh, I., & Lee, C. G. (2007). Alternative configurations for cutting machines in a tube cutting mill. *European Journal of Operational Research, 183*(3), 1385-1396.

Hebhoub, H., Aoun, H., Belachia, M., Houari, H., & Ghorbel, E. (2011). Use of waste marble aggregates in concrete. *Construction and Building Materials, 25*(3), 1167-1171.

Hinxman, A. I. (1980). The trim-loss and assortment problems: A survey. *European Journal of Operational Research, 5*(1), 8-18.

Holthaus, O. (2002). Decomposition approaches for solving the integer one-dimensional cutting stock problem with different types of standard lengths. *European Journal of Operational Research, 141*(2), 295-312.

Holthaus, O. (2003). On the best number of different standard lengths to stock for one-dimensional assortment problems. *International Journal of Production Economics, 83*(3), 233-246.

Johnston, R. E., & Sadinlija, E. (2004). A new model for complete solutions to one-dimensional cutting stock problems. *European Journal of Operational Research, 153*(1), 176-183.

Kantorovich, L.V. (1939). Mathematical methods of organizing and planning production. *Management Science 6,* 1960, 366-422.

Kim, B. I., Ki, Y., Son, D., Bae, B., & Park, J. S. (2016). An algorithm for a cutting problem in window frame production. *International Journal of Production Research, 54*(14), 4327-4339.

Koch, S., König, S., & Wäscher, G. (2009). Integer linear programming for a cutting problem in the wood-processing industry: A case study. *International Transactions in Operational Research*, *16*(6), 715-726.

Larsen, A., & Bundesen, C. (1996). A template-matching pandemonium recognizes unconstrained handwritten characters with high accuracy. *Memory & cognition*, *24*(2), 136-143.

Ozfirat, P. M. (2012). An integer programming approach for the three-dimensional cutting planning problem of marble processing industry. *The International Journal of Advanced Manufacturing Technology*, *59*(9-12), 1057-1064.

Poldi, K. C., & de Araujo, S. A. (2016). Mathematical models and a heuristic method for the multiperiod one-dimensional cutting stock problem. *Annals of Operations Research*, *238*(1-2), 497-520.

Ragsdale, C. T., & Zobel, C. W. (2004). The ordered cutting stock problem. *Decision Sciences*, *35*(1), 83-100.

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

Rönnqvist, M. (1995). A method for the cutting stock problem with different qualities. *European Journal of Operational Research*, *83*(1), 57-68.

Sardinas, R. Q., Santana, M. R., & Brindis, E. A. (2006). Genetic algorithm-based multi-objective optimization of cutting parameters in turning processes. *Engineering Applications of Artificial Intelligence*, *19*(2), 127-133.

Scheithauer, G. (1991). Algorithms for the container loading problem. *In Operations Research Proceedings* 1992, 445-452.

Scheithauer, G., & Terno, J. (1995). The modified integer round-up property of the one-dimensional cutting stock problem. *European Journal of Operational Research*, *84*(3), 562-571.

Schilling, G., & Georgiadis, M. C. (2002). An algorithm for the determination of optimal cutting patterns. *Computers & Operations Research*, *29*(8), 1041-1058.

Selver, M. A., Akay, O., Ardali, E., Yavuz, A. B., Onal, O., & Ozden, G. (2009). Cascaded and hierarchical neural networks for classifying surface images of marble slabs. Systems, Man, and Cybernetics, Part C: *Applications and Reviews, IEEE Transactions on*, *39*(4), 426-439.

Shahin, A. A., & Salem, O. M. (2004). Using genetic algorithms in solving the one-dimensional cutting stock problem in the construction industry. *Canadian Journal of Civil Engineering*, *31*(2), 321-332.

Sinuany-Stern, Z., & Weiner, I. (1994). The one dimensional cutting stock problem using two objectives. *Journal of the Operational Research Society*, 231-236.

Stadtler, H. (1990). A one-dimensional cutting stock problem in the aluminium industry and its solution. *European Journal of Operational Research*, *44*(2), 209-223.

Suliman, S. M. (2001). Pattern generating procedure for the cutting stock problem. *International Journal of Production Economics*, *74*(1), 293-301.

Sweeney, P. E., & Haessler, R. W. (1990). One-dimensional cutting stock decisions for rolls with multiple quality grades. *European Journal of Operational Research*, *44*(1990), 224-231.

Trkman, P., & Gradisar, M. (2007). One-dimensional cutting stock optimization in consecutive time periods. *European Journal of Operational Research*, *179*(2), 291-301.

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

Umetani, S., Yagiura, M., & Ibaraki, T. (2006). One-dimensional cutting stock problem with a given number of setups: a hybrid approach of metaheuristics and linear programming. *Journal of Mathematical Modelling and Algorithms*, *5*(1), 43-64.

Vahrenkamp, R. (1996). Random search in the one-dimensional cutting stock problem. *European Journal of Operational Research*, *95*(1), 191-200.

Vanderbeck, F. (2000). Exact algorithm for minimising the number of setups in the one-dimensional cutting stock problem. *Operations Research*, *48*(6), 915-926.

Vasko, F. J., Newhart, D. D., & Stott Jr, K. L. (1999). A hierarchical approach for one-dimensional cutting stock problems in the steel industry that maximizes yield and minimizes overgrading. *European Journal of Operational Research*, *114*(1), 72-82.

Wäscher, G. (1990). An LP-based approach to cutting stock problems with multiple objectives. *European Journal of Operational Research*, *44*(2), 175-184.

Wäscher, G., Haußner, H., & Schumann, H. (2007). An improved typology of cutting and packing problems. *European Journal of Operational Research*, *183*(3), 1109-1130.

Westerlund, T., Harjunkoski, I., & Isaksson, J. (1998). Solving a production optimization problem in a paper-converting mill with MILP. *Computers & Chemical Engineering*, *22*(4), 563-570.

Yang, C. T., Sung, T. C., & 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), 502-513.

Yuen, B. J., & Richardson, K. V. (1995). Establishing the optimality of sequencing heuristics for cutting stock problems. *European Journal of Operational Research*, *84*(3), 590-598.