

DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

**MODELLING AND OPTIMIZATION OF ORDER
BATCHING PROBLEM IN WAREHOUSE
SYSTEMS**



by
Çağla CERĞİBOZAN

February, 2019
İZMİR

MODELLING AND OPTIMIZATION OF ORDER BATCHING PROBLEM IN WAREHOUSE SYSTEMS

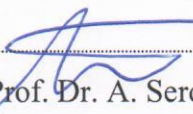
**A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Doctor of
Philosophy in Industrial Engineering, Industrial Engineering Program**

**by
Çağla CERĞİBOZAN**

**February, 2019
İZMİR**

Ph.D. THESIS EXAMINATION RESULT FORM

We have read the thesis entitled “**MODELLING AND OPTIMIZATION OF ORDER BATCHING PROBLEM IN WAREHOUSE SYSTEMS**” completed by **ÇAĞLA CERĞİBOZAN** under supervision of **ASSOC. PROF. DR. A. SERDAR TAŞAN** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Doctor of Philosophy.


Assoc. Prof. Dr. A. Serdar TAŞAN

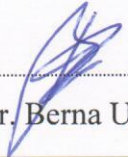
Supervisor


Asst. Prof. Dr. Burcu FELEKOĞLU

Thesis Committee Member


Prof. Dr. Okan TUNA

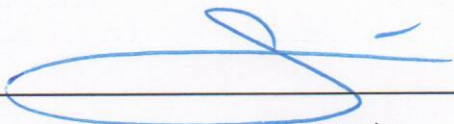
Thesis Committee Member


Prof. Dr. Berna ULUTAŞ

Examining Committee Member


Assoc. Prof. Dr. Yigit KAZANÇOĞLU

Examining Committee Member


Prof. Dr. Kadriye ERTEKİN

Director

Graduate School of Natural and Applied Sciences

ACKNOWLEDGEMENTS

First of all, I would like to express my gratitude to my supervisor Assoc. Prof. Dr. A. Serdar TAŞAN for his support and guidance. I am also thankful to the committee members, Asst. Prof. Dr. Burcu FELEKOĞLU and Prof. Dr. Okan TUNA for their valuable comments and suggestions.

I also would like to thank to Olgun Altar. His support and practical experience was valuable in the implementation of the case study.

All parts of the thesis (conceptual design, research, implementation and writing) are also parts of a process of devotion. I thank all my friends for their friendship and support throughout this process.

I am grateful to my whole family for their pure love, endless support and understanding. I would like to express my deepest thanks to my parents Gülcan CERGİBOZAN and Ali CERGİBOZAN, and my sister Eda CERGİBOZAN. They always brighten my world.

Çağla CERGİBOZAN

MODELLING AND OPTIMIZATION OF ORDER BATCHING PROBLEM IN WAREHOUSE SYSTEMS

ABSTRACT

Order batching problem (OBP) is a combinatorial optimization problem which arises in warehouse order picking process. In OBP, the aim is to find groups of orders and picking routes of these groups to minimize distance travelled by the order picker. This problem is encountered especially in manual order picking systems where the capacity of picking facility is limited. Solving the OBP becomes more important when size of the problem (e.g. number of storage locations, number of aisles, number of customer orders, etc.) is large. Most of the time spent during retrieval of the orders depends on content of the batch and picking route. Therefore, an effective batching and routing approach is essential in reducing the time needed to collect ordered items.

The main objective of this study is identified as developing fast and effective metaheuristic approaches that can solve the OBP in different warehouse configurations. For this purpose, three solution approaches based on genetic algorithm and local search are proposed. The numerical study of proposed algorithms is implemented with generated data sets, and also with instances from literature. Results of the benchmark study demonstrate competitiveness of developed approaches. It is believed that proposed methods will be useful to solve real case applications of the problem in different warehouse configurations. Accordingly, a real case application in a distribution center of a well-known retailer in Turkey is implemented. The case study also includes storage assignment process of incoming products. The results demonstrate that algorithms are practical and useful in real life problems.

Keywords: Order batching problem, genetic algorithms, hybrid metaheuristics, local search

DEPO SİSTEMLERİNDE SİPARİŞ GRUPLAMA PROBLEMİNİN MODELLENMESİ VE OPTİMİZASYONU

ÖZ

Sipariş gruplama problemi (SGP), depo sipariş toplama sürecinde ortaya çıkan bir kombinatoriyal optimizasyon problemidir. SGP’de amaç, sipariş toplama personelinin kat ettiği mesafeyi en aza indirmek için siparişlerin gruplarını ve bu grupların toplama rotalarını bulmaktır. Bu problemle özellikle, toplama aracının kapasitesinin sınırlı olduğu manuel sipariş toplama sistemlerinde karşılaşılmaktadır. Problemin boyutu (örneğin depolama birimlerinin sayısı, koridorların sayısı, müşteri siparişlerinin sayısı vs.) büyük olduğunda SGP’yi çözmek daha önemli hale gelmektedir. Siparişlerin çekilmesi sırasında harcanan zamanın çoğu, grubun içeriğine ve toplama rotasına bağlıdır. Bu nedenle, etkili bir gruplama ve rotalama yaklaşımı, sipariş edilen ürünleri toplamak için gereken zamanı azaltmada önemlidir.

Bu çalışmanın temel amacı, farklı depo konfigürasyonlarında SGP’yi çözebilen hızlı ve etkili metasezgisel yaklaşımlar geliştirmek olarak belirlenmiştir. Bu amaçla, genetik algoritma ve yerel arama tabanlı üç çözüm yaklaşımı önerilmiştir. Önerilen algoritmaların hesaplama çalışması üretilen veri setleri ve ayrıca literatürden örnekler ile gerçekleştirilmiştir. Karşılaştırma çalışmasının sonuçları geliştirilen yaklaşımların rekabet yeteneğini göstermektedir. Önerilen yöntemlerin, farklı depo konfigürasyonlarında problemin gerçek durum uygulamalarını çözmek için kullanışlı olacağına inanılmaktadır. Buna göre, Türkiye’de tanınmış bir perakendecinin dağıtım merkezinde gerçek vaka uygulaması gerçekleştirilmiştir. Vaka çalışması gelen ürünlerin depolama atama işlemini de içermektedir. Sonuçlar, algoritmaların gerçek hayat problemlerinde pratik ve kullanışlı olduğunu göstermektedir.

Anahtar kelimeler: Sipariş gruplama problemi, genetik algoritmalar, melez metasezgiseller, yerel arama

CONTENTS

	Page
Ph.D. THESIS EXAMINATION RESULT FORM	ii
ACKNOWLEDGEMENTS	iii
ABSTRACT	iv
ÖZ	v
LIST OF FIGURES	x
LIST OF TABLES	xii
 CHAPTER ONE – INTRODUCTION	 1
1.1 Order Fulfillment Process	1
1.2 Motivation	4
1.3 Methodology	6
1.4 Outline of the Thesis	7
 CHAPTER TWO – LITERATURE REVIEW ON ORDER PICKING AND BATCHING OPERATIONS	 10
2.1 Introduction to Warehouse Operations	10
2.2 Order Picking Process	10
2.2.1 Simulation Based Studies	14
2.2.2 Exact, Heuristic and Metaheuristic Approaches	15
2.2.3 Review Studies	16
2.3 Order Batching Problem.....	19
2.3.1 Mathematical Formulation	20
2.3.2 Solution Methods for Order Batching Problem.....	22
2.3.2.1 Exact Methods	23
2.3.2.2 Heuristic and Metaheuristic Methods	24
2.3.2.3 Data Mining Approaches	29
2.4 Findings.....	30

2.5 Conclusion.....	40
CHAPTER THREE – PROPOSED METAHEURISTIC APPROACHES FOR THE ORDER BATCHING PROBLEM.....	42
3.1 Introduction	42
3.2 Metaheuristics for the Order Batching Problem	43
3.2.1 Genetic Algorithm	43
3.2.1.1 Solution Representation	45
3.2.1.2 Initialization	46
3.2.1.3 Fitness Value Calculation	46
3.2.1.4 Selection.....	46
3.2.1.5 Recombination	47
3.2.1.6 Mutation.....	48
3.2.1.7 Evaluation and Replacement	48
3.2.2 Hybrid Algorithm with Population Interaction	49
3.2.2.1 Interaction Based on Swarm Optimization Technique	50
3.2.2.2 Hybrid Scheme	52
3.2.3 Local Search Based Algorithm.....	54
3.2.3.1 Flowchart of the Algorithm	55
3.2.3.2 Local Search Heuristics	55
3.2.3.2.1 Swap Heuristic	56
3.2.3.2.2 Move Forward Heuristic	56
3.2.3.2.3 Move Backward Heuristic	56
3.2.3.2.4 2-opt Heuristic.....	57
3.3 Conclusion.....	57
CHAPTER FOUR – COMPUTATIONAL STUDY FOR THE PROPOSED APPROACHES	59
4.1 Introduction	59
4.2 Illustrative Case.....	59

4.2.1 Structure of the Warehouse	60
4.2.2 Routing Strategies	62
4.2.3 Experimental Design for Genetic Algorithm and Hybrid Algorithm.....	66
4.2.4 Design of Parameters for Local Search Based Algorithm.....	70
4.2.5 Computational Study and Results	73
4.3 Benchmark Instances.....	75
4.3.1 Structure of the Warehouse System	76
4.3.2 Computational Study	77
4.3.2.1 Application of the Genetic Algorithm	78
4.3.2.2 Application of the Hybrid Algorithm	79
4.3.2.3 Application of the Local Search Algorithm.....	80
4.3.2.4 Results and Evaluation.....	80
4.4 Conclusion.....	89
 CHAPTER FIVE – CASE STUDY IN A DISTRIBUTION CENTER FROM THE RETAIL SECTOR	 91
5.1 Introduction	91
5.2 Presentation of the Case Study.....	92
5.2.1 Layout of the Distribution Center.....	92
5.2.2 Details of the Current Picking System	96
5.2.3 Proposed Strategies for the Storage Design	98
5.2.4 Metaheuristic Approaches for the Order Picking and Batching Operations.....	101
5.3 Results and Conclusion	103
 CHAPTER SIX – CONCLUSION	 105
6.1 Summary of the Thesis.....	105
6.2 Main Findings	106
6.3 Concluding Remarks	106
6.3.1 Contributions to the Literature	106

6.3.2 Implications for Practice.....	107
6.3.3 Future Research Directions	107
REFERENCES.....	108
APPENDICES	125
APPENDIX-1: Problem Data-1	125
APPENDIX-2: Problem Data-2	126
APPENDIX-3: Problem Data-3	128



LIST OF FIGURES

	Page
Figure 1.1 Order fulfillment process.....	2
Figure 1.2 Research methodology	6
Figure 1.3 Structure of the thesis	8
Figure 2.1 A typical warehouse layout	12
Figure 2.2 Flowchart of the literature review.....	13
Figure 2.3 Problem types of examined studies in the related literature	13
Figure 2.4 Solution methods for order batching related problems.....	39
Figure 2.5 Number of papers for order batching related problems.....	39
Figure 3.1 Genetic algorithm flowchart	44
Figure 3.2 Genetic algorithm pseudo code	45
Figure 3.3 Chromosome representation based on order number	46
Figure 3.4 The order crossover procedure	47
Figure 3.5 Order crossover operation	47
Figure 3.6 Procedure of the 2-opt heuristic.....	48
Figure 3.7 Flowchart of the proposed genetic algorithm	49
Figure 3.8 Pseudo code of PSO.....	51
Figure 3.9 Pseudo code of proposed hybrid algorithm	53
Figure 3.10 Flowchart of the proposed hybrid algorithm	54
Figure 3.11 Flowchart of the proposed local search algorithm.....	55
Figure 3.12 Swap heuristic.....	56
Figure 3.13 Move forward heuristic.....	56
Figure 3.14 Move backward heuristic.....	57
Figure 3.15 2-opt heuristic	57
Figure 4.1 The layout of the warehouse.....	61
Figure 4.2 Random order locations.....	64
Figure 4.3 S-shape routing	64
Figure 4.4 Return routing.....	64
Figure 4.5 Mid-point routing	65
Figure 4.6 Largest Gap routing	65

Figure 4.7 Combined routing	65
Figure 4.8 Optimal routing.....	66
Figure 4.9 Best objective function values along GA iterations for P1.....	68
Figure 4.10 Average objective function values along GA iterations for P1	68
Figure 4.11 Best objective function values along GA iterations for P2.....	68
Figure 4.12 Average objective function values along GA iterations for P2	69
Figure 4.13 Best objective function values along GA iterations for P3.....	69
Figure 4.14 Average objective function values along GA iterations for P3	69
Figure 4.15 Normal probability plot of the standardized effects	71
Figure 4.16 Pareto chart of the standardized effects	71
Figure 4.17 Main effects plot (data means)	72
Figure 4.18 Interaction plot (data means)	72
Figure 4.19 Warehouse layout considered in the study	76
Figure 4.20 Results for genetic algorithm in terms of deviations for ABC1	87
Figure 4.21 Results for genetic algorithm in terms of deviations for Ran1	87
Figure 4.22 Results for hybrid algorithm in terms of deviations for ABC1	88
Figure 4.23 Results for hybrid algorithm in terms of deviations for Ran1	88
Figure 4.24 Results for local search algorithm in terms of deviations for ABC1.....	89
Figure 4.25 Results for local search algorithm in terms of deviations for Ran1	89
Figure 5.1 Course of action for the case study.....	91
Figure 5.2 The layout of the DC	93
Figure 5.3 The U-shape picking within aisle	94
Figure 5.4 An aisle view from the DC	94
Figure 5.5 Receiving area at the DC	95
Figure 5.6 Labeling example.....	95
Figure 5.7 Labeling example of adjacent SKUs	95
Figure 5.8 Current storage area of the DC	97
Figure 5.9 Current layout of the Level 2 storage area.....	98
Figure 5.10 Front view from the depot	99
Figure 5.11 Side view from the depot.....	99
Figure 5.12 Proposed assignment approach for Level 2 storage area.....	100
Figure 5.13 Example locations of ordered SKUs.....	102

LIST OF TABLES

	Page
Table 2.1 Studies on order picking process in the literature	18
Table 2.2 Studies on order batching in the related literature	31
Table 2.3 Detailed information on application environment	37
Table 4.1 Warehouse parameters	61
Table 4.2 Problem classes with related parameter values.....	62
Table 4.3 Parameter levels for the genetic algorithm	67
Table 4.4 Parameter set for the genetic algorithm	67
Table 4.5 Parameter set for the hybrid algorithm	70
Table 4.6 Parameter levels for the local search algorithm.....	71
Table 4.7 Results for S-shape routing strategy	73
Table 4.8 Results for mid-point routing strategy	74
Table 4.9 Results for return routing strategy	75
Table 4.10 Warehouse parameters	78
Table 4.11 Problem sets	78
Table 4.12 Parameter set in benchmark study for the genetic algorithm.....	79
Table 4.13 Parameter set in benchmark study for the hybrid algorithm.....	79
Table 4.14 Parameter set in benchmark study for the local search algorithm	80
Table 4.15 GA results for ABC1 with S-shape strategy	81
Table 4.16 GA results for Ran1 with S-shape strategy.....	81
Table 4.17 Hybrid algorithm results for ABC1 with S-shape strategy	82
Table 4.18 Hybrid algorithm results for Ran1 with S-shape strategy.....	82
Table 4.19 Local search algorithm results for ABC1 with S-shape strategy	83
Table 4.20 Local search algorithm results for Ran1 with S-shape strategy.....	83
Table 4.21 GA results for ABC1 with return and mid-point strategies	84
Table 4.22 GA results for Ran1 with return and mid-point strategies	84
Table 4.23 Hybrid algorithm results for ABC1 with return and mid-point strategies	85
Table 4.24 Hybrid algorithm results for Ran1 with return and mid-point strategies.	85
Table 4.25 Local search algorithm results for ABC1 with return and mid-point strategies.....	86

Table 4.26 Local search algorithm results for Ran1 with return and mid-point strategies.....	86
Table 5.1 Parameters of the DC	97
Table 5.2 Parameters of the Level 2 storage area	99
Table 5.3 An example set of orders	101
Table 5.4 Parameters for the genetic algorithm	102
Table 5.5 Parameters for the hybrid algorithm	102
Table 5.6 Parameters for the local search algorithm.....	103
Table 5.7 Parameters for the genetic algorithm	103



CHAPTER ONE

INTRODUCTION

1.1 Order Fulfillment Process

In a competitive supply chain (SC) system, meeting customer expectations rapidly has become a necessity while ensuring the quality of the product and/or service. Supply chain management (SCM) aims to add value through related processes, and increasing customer service level has more importance than in previous years. SCs have the complexity due to variations in forecast, supply, process, and transportation. Besides, it is a necessity to manage the whole chain effectively; otherwise several problems such as lateness, defective and damaged goods, stock outs etc. will occur.

In today's global competitive environment, awareness about the efficiency of order fulfillment systems is increasing. Particularly in the retail industry, companies try to find novel methods to stay competitive in the age of retail power. Order fulfillment can be defined as the process of all transactions starting with the customer order and continuing with the delivery of the order to the customer. In some cases, order fulfillment concept is used for combination of following operations or some of these operations:

- generating
- packing
- transporting
- servicing after sales of customer orders

The process of order fulfillment (OF) is given in Fig 1.1. According to Lin & Shaw (1998), order fulfillment process (OFP) begins with the receipt of orders from customers, and ends with the delivery of the products. Since the process is composed of different elements and their connections, the OFP is said to be complex. They point out the necessity of integrating the OFP into supply chain networks (SCNs) to improve the OFP.

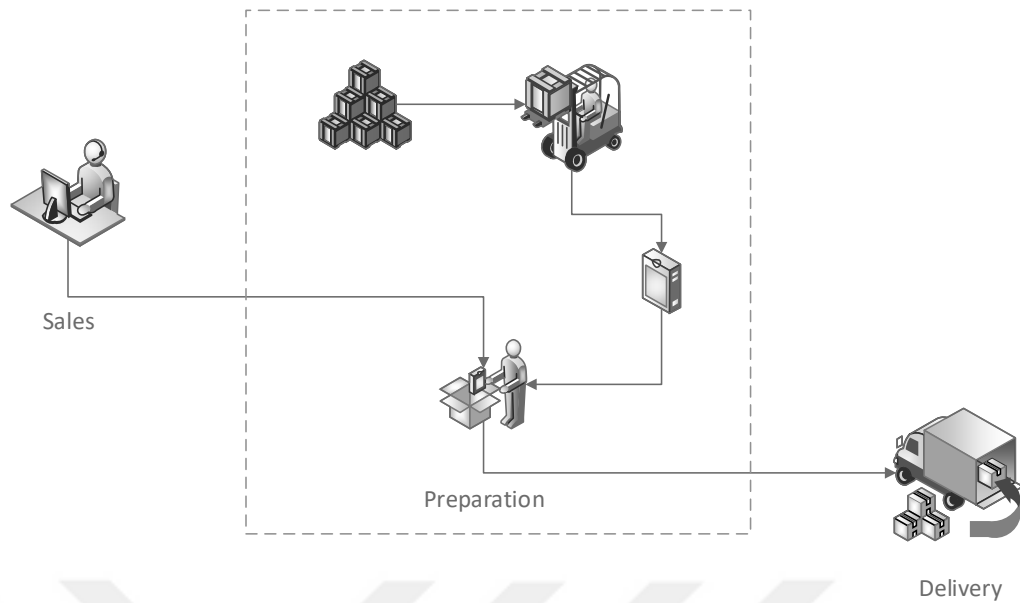


Figure 1.1 Order fulfillment process

In the related literature, OFP has been investigated in many different ways within SCM. Croxton (2003) explains in detail how to implement the OFP within a company and how it can be managed between companies in SC. Pil & Holweg (2004) explore the link between external diversity (for the customer) and internal diversity (for creation of the product); and express that the OF strategy determines the effectiveness of the approaches to be developed against such variability. Chen & Huang (2006) utilize triangular fuzzy numbers (TFNs) to describe the uncertain variables and present an OF analysis model of a SC system in an uncertain environment. Davis-Sramek, Mentzer, & Stank (2008), test a number of hypotheses related with the OFP in order to ensure retailer loyalty. In Davis-Sramek, Germain, & Stank (2010), a basic emphasis is on retailer perceptions that are generally limited to the field of consumer research. Authors investigate how the manufacturer's success in OFP and the level of the service impacts the retailer. In Amer, Luong, Lee, & Ashraf (2008) study design for six sigma (DFSS), and propose a methodology that uses DFSS and also a fuzzy logic technique to optimize the related OFP. In the study of Lin, Hong, Wu, & Wang (2010), it is aimed to present an available-to-promise (ATP) model that enhances the responsiveness of OFPs. Yan, Rabinovich, Dooley, & Evers (2010) examine how fulfillment lead times variations can be administrated successfully in the Internet retailing environment. Simulation study is used to

investigate hypotheses about the links between order fulfillment rules and backlog difference. Authors point out the need to consider both quantity variation and lead time variation when controlling OF systems. Chen et al. (2013) propose a three-stage method to solve the OF problem of on-demand delivery service with large size of orders for a boxed meal case. Authors utilize a geographic information system (GIS) to locate the delivery points selected by the customers. The K-means algorithm is used for clustering customer orders based on locations and the number of vehicles; and a genetic algorithm (GA) approach is utilized for minimization of the total distance.

It can be seen that OFP has become a significant element for business firms to remain competitive. Research on the OFP is mainly focused on a number of areas, and the ideas can be highlighted as follows:

- *It is critical for a firm to determine the role of technology in its processes.* The firm may prefer to use automation techniques widely, or just in a few operations within the OFP.
- *Data sharing is important to respond quickly to changes.* It is significant to set up a data sharing system to keep up with the momentary changes.
- *Speed of implementation is highly essential.* Recently OFP has become a critical element of e-commerce, and firms compete with each other in terms of speed.
- *Because of the implementation costs, simulation studies are used to visualize the process changes.* This is a necessity for costly operations in some cases.

In the light of the studies from the literature, various performance measures can be used for OFP's effectiveness. These performance measures can be indicated as follows:

- Percentage of orders shipped on-time
- Time needed to fulfill the order
- Percentage of returned items
- Operation of the service or article
- Customer satisfaction level on the OFP
- Inventory levels of finished goods

1.2 Motivation

Companies manage their processes in accordance with ever-changing customer expectations and rivalry conditions. A key issue faced by them is how to monitor and control the performance throughout the SC. They tend to increase variety of features in their products to attract customers, but this action often increases complexity and management costs. Hanne & Dornberger (2017) focus on several computational methods that can be used for improving logistics systems performance. According to Manzini (2012), the competition between companies in logistics sector is based upon especially time, cost, and customer service level dimensions. The above-mentioned performance measures denote that most of them have a direct or indirect relation with the time function in OFP. The two main elements in the OFP - picking and transportation - are known to be the most time consuming operations within the process. At this point, warehouse management gains importance as one of the most important activities in a SC.

In recent years, several studies concentrate on design and management of the warehouses and related operations. Detailed design issues of warehouses are investigated in the studies of Baker & Canessa (2009), and Pohl, Meller, & Gue (2009). Gue, Ivanović, & Meller (2012) focus on the arrangement of the aisles in a unit load warehouse. Aisle types are evaluated according to their simplicity for travel between locations. Bartholdi & Hackman (2014) handle the problem of reducing travel with the pick-path optimization. Accordingly, transportation of the goods and items in warehouses has been known as a cost-intensive process, and particular emphasis should be placed on this operation as well as the design policies. Therefore,

order picking operation which requires a substantial effort among warehouse operations will be the main theme in this thesis.

In manual picker to part OP systems, the operational process after receiving an order includes basically preparation and transportation of the order to the customer. At the order preparation level, the most important issues that should be considered in the warehouse configuration are as follows:

- number of stock keeping units (number of different items)
- type of warehouse storage system
- type of warehouse layout and design
- number of order pickers
- type of the tool that order picker use to recognize the items (item recognition system)
- type of the picking/storing device
- number of aisles and shelves
- size of the shelves
- method of taking the order from the stock

Additionally, it is also very important to decide whether quality control will be carried out while receiving the product. The decision is that of applying standard procedure to all incoming products, or a distinct procedure to some suppliers. Therefore, the company will determine the policy with respect to this decision.

The following issues will be investigated in this thesis:

- Importance of the order batching within order picking process
- Type of solution techniques have been applied to the problem
- Gaps in the related literature and future research opportunities
- Fast and effective solution approaches to solve the OBP
- Case study application in a warehouse

1.3 Methodology

The research methodology of this thesis is expressed in Figure 1.2. The main goal of the thesis is to conduct an in-depth analysis on warehouse order picking processes, and develop effective and applicable solution methods for the order batching problem (OBP) encountered within this process.

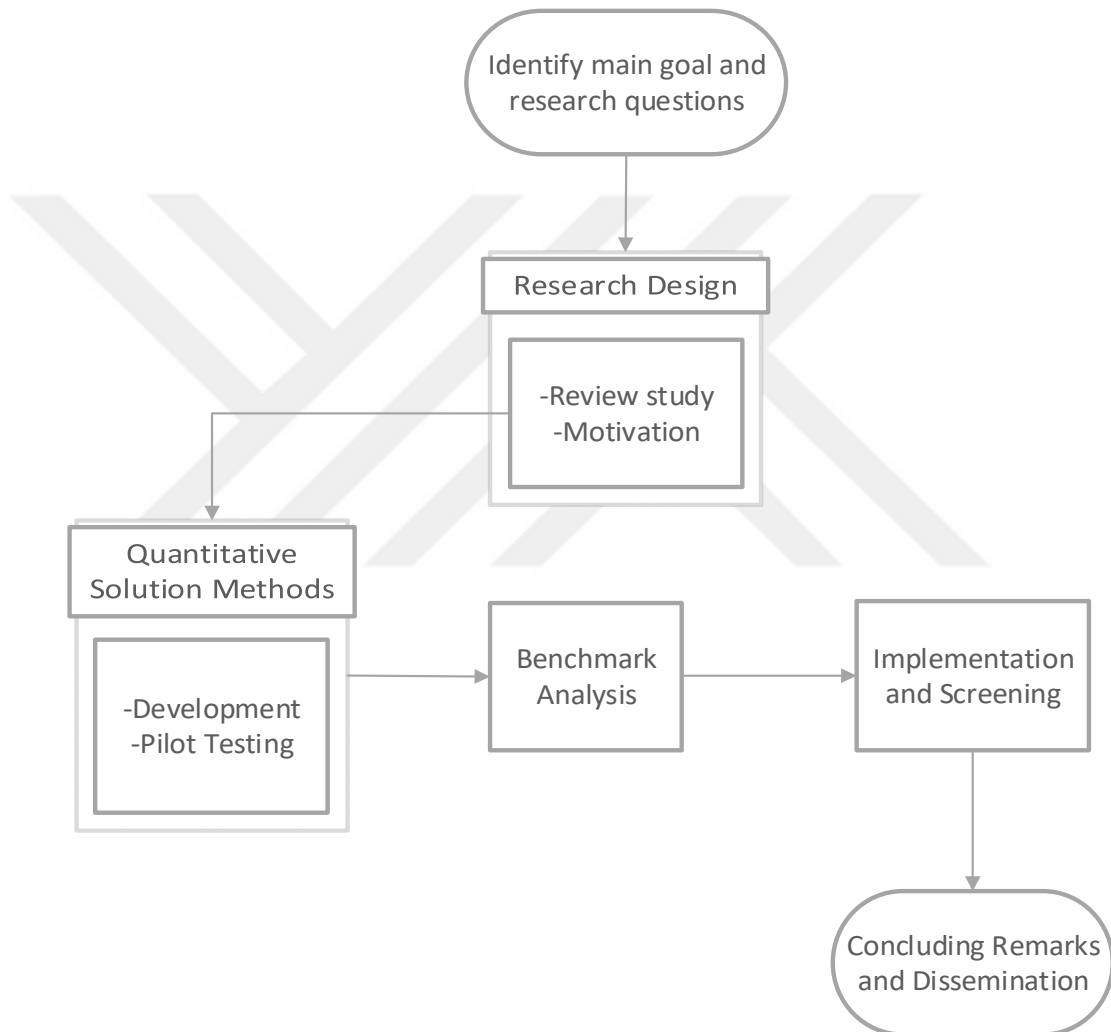


Figure 1.2 Research methodology

After determination of the research questions, the research framework is designed. Review studies and the motivation of the study construct the design of the research approaches that will be carried out through the thesis. Therefore, quantitative solution approaches will first be identified and developed for the problem, and then preliminary testing of the solution approaches will be implemented. The results will

lead to a benchmark analysis that shows the capabilities and effectiveness of the algorithms. After the computational studies, the algorithms will be implemented to a real case study, and the thesis will be finalized with the final results and concluding remarks. In the next section, the outline of the thesis will be given in detail with chapter contents.

1.4 Outline of the Thesis

The general outline of the thesis is presented in Figure 1.3. At first, the framework of the thesis is identified considering the existing logistics literature. The main research field in this thesis is chosen as the warehouse management because of its significance in the logistics process. In Chapter 1, an introduction to the warehousing and related operations in these systems is carried out. Following an in-depth analysis of the literature on warehouse operations, order picking process is identified to encompass the highest time during the order fulfillment process. Consequently, the aim of the thesis is identified as optimizing the order picking process in warehouses.

In Chapter 2, the problem statement is implemented with mathematical models of the OBP. Order picking and batching operations in warehouses are also investigated in this chapter, and an overview of solution approaches is presented to provide general information on this operation. Current state of the research on OBP is examined under exact, heuristic/metaheuristic, and data mining approaches subheadings.

In the last decades, metaheuristic solution approaches have become a topic of great interest. Due to their advantages that could not be ignored, use of metaheuristics in optimization problems is frequently encountered. In this thesis, different solution approaches are proposed for the OBP. In Chapter 3, three metaheuristic algorithms are proposed to solve the OBP: genetic algorithm, hybrid algorithm, and local search based algorithm. These metaheuristic approaches constitute and route the batches in order to find a least cost solution to the problem.

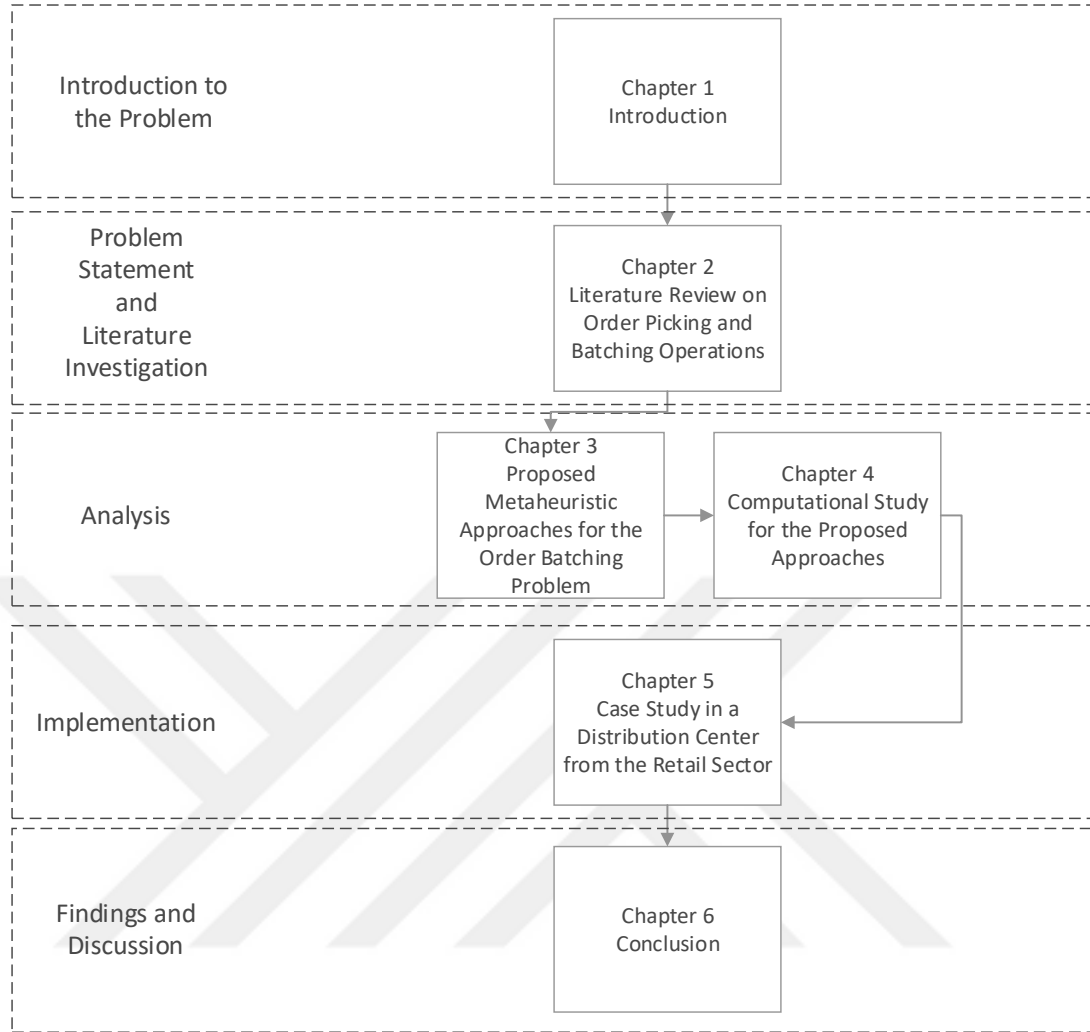


Figure 1.3 Structure of the thesis

In Chapter 4, a detailed computational study with 32 benchmark instances as well as generated problem sets for the OBP are implemented. The benchmark instances are obtained from a database that is commonly used in the related literature. The results are compared in terms of percentage deviations from the best solutions found in the literature. According to results, the performance of the algorithms are evaluated.

In Chapter 5, the developed metaheuristic approaches are applied to a distribution center (DC) of a well-known retailer in Turkey. The DC serves to large number of markets; correspondingly, the total number of processed items is high. The case study helps to understand the dynamics of the whole picking process under some real life constraints. Warehouse layout considered in the case study is a multi-block

layout with receiving, storage and shipping areas. Descriptive information on operations of the DC are given in the study, and after defining and setting all the related information and parameters, the computational study is performed with the example data set obtained from the DC.

Finally, in Chapter 6, the thesis is concluded with the key findings and future research directions. It is believed that the importance of the order picking in warehouse operations will not change or decrease over time. However, as the changing needs arise, many innovations are likely to occur in this process.



CHAPTER TWO

LITERATURE REVIEW ON ORDER PICKING AND BATCHING OPERATIONS

2.1 Introduction to Warehouse Operations

Warehouses are storage facilities used for supplying customer orders and handling the items in the logistics systems. Fundamental warehouse operations can be stated as receiving, storage, order picking (including batching), packing, and shipping (Tompkins & Smith, 1998; Ten Hompel & Schmidt, 2007).

Typical warehouse activities start with the receiving process which corresponds to accepting and checking procedure of the items. Storage process includes assignment of stock keeping unit numbers to items, sorting and holding of items in the storage area. In the order picking process, ordered items are retrieved from their storage locations according to the predetermined picking policy. Packing process consists of packaging and preparation of the orders for shipping. As a final operation, shipping process ensures transportation of the orders to customers. Among these activities, order picking process has a separate importance. This process has storage, batching/zoning (if applicable) and picker routing decisions in itself, and effectiveness of this process is completely dependent on these policies (Petersen & Schmenner, 1999). Therefore, order picking process which is related with the batching operation will be investigated in detail. It should be noted that the earlier version of this section is published in Cergibozan & Tasan (2019).

2.2 Order Picking Process

Order picking is the process of retrieving ordered items from their locations in a warehouse (Petersen, 1997; Petersen & Schmenner, 1999; de Koster, Le-Duc, & Roodbergen, 2007). It is a significant process of a warehouse, since main objective in this operation is cost and/or delivery time minimization. According to de Koster et al. (2007) and Bartholdi & Hackman (2014), cost of order picking is estimated to form

as 55% of the total warehouse operating cost, therefore in this paper the main concern is given especially to this process.

In order picking systems, a key decision is that whether a manual or an automated picking system will be used. Due to results of the research on the related field, it has been determined that manual order picking systems are highly preferred in practice (de Koster, et al., 2007; Henn & Schmid, 2013). These systems can be classified into two according to movement direction: picker-to-parts systems and parts-to-picker systems. Picker-to-parts systems in which the order picker walks or moves with a picking facility to the location of the items to pick up orders are more widespread than parts-to-picker systems. In parts-to-picker systems, items are retrieved by unit load devices (automated storage and retrieval systems) and transported to the picker (de Koster et al., 2007). Basically, picking and batching operations in picker-to-parts systems are investigated in the following sections.

In Figure 2.1, an example of a warehouse layout that involves identical storage units can be seen. The layout consists of storage units, aisles between locations and two cross aisles as front and back aisles, and an input/output point which can be called as depot. The layout corresponds to a low-level picker-to-parts systems layout in which the picker moves two directions to pick up the items. It also has two cross aisles, which makes it a single block layout. If the number of cross aisles increases, the layout becomes multi-block layout (Petersen & Schmenner, 1999).

In the review paper of Rouwenhorst et al. (2000), warehouse design and control problems are classified and examined along strategic, tactical and operational levels. From this point of view, order picking process can be thought as an operational level activity because of its frequency of occurrence. Such a frequent operation will ensure higher productivity if it could be managed effectively.

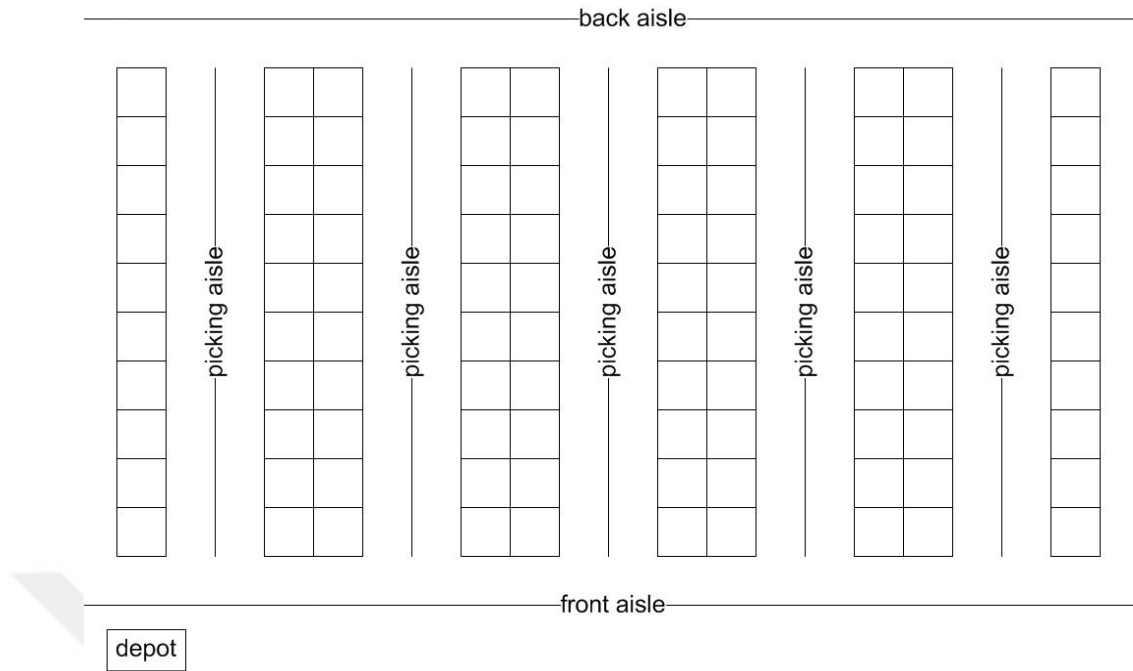


Figure 2.1 A typical warehouse layout

In recent years, research on order picking and batching problems has grown considerably. Selection of the papers in this literature review chapter is made after evaluation of relevance, number of citations, and accessibility features of the search results in the Scholar Google, Scopus, and Web of Science databases. In Figure 2.2, systematic flowchart of the review is given.

Since the database search returns a great number of order picking related studies, studies which emphasize directly on order picking process are examined, and the remarkable points of the studies are summarized.

The aim of this type of search is to provide general information on how this process is known and handled in relatively well-known research articles. The advantage is that it provides quick information to the reader. Besides, it restricts the search around widespread papers; however, an informative paper which is published recently may not be included in the evaluation.

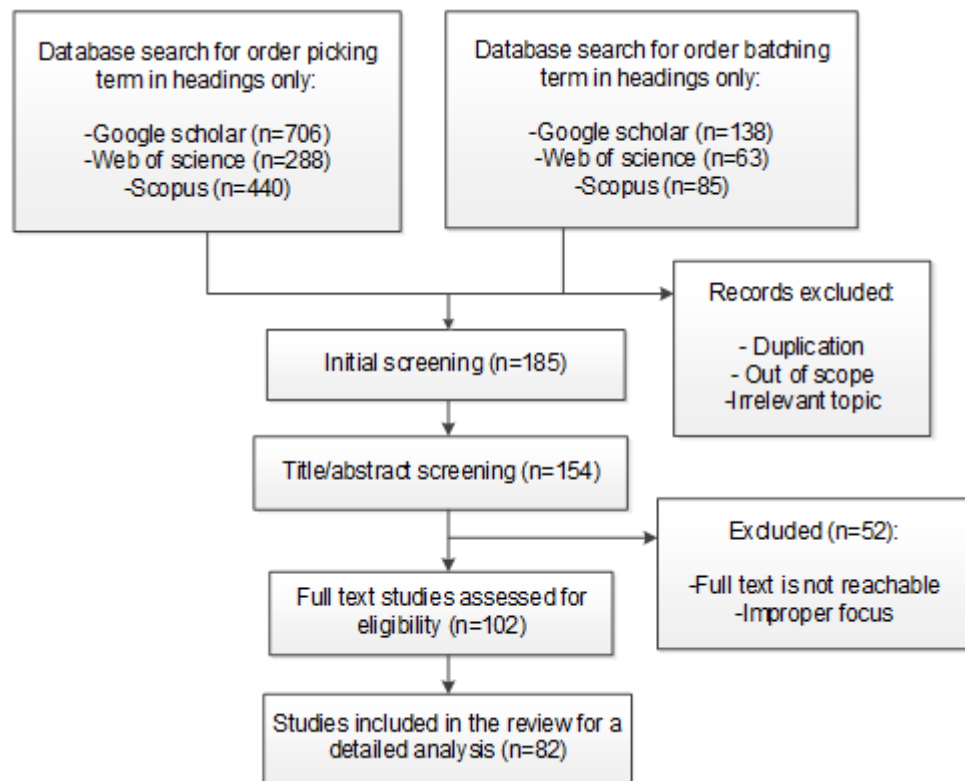


Figure 2.2 Flowchart of the literature review

This chapter draws a frame for the order batching problem (OBP) in warehouse order picking process. Figure 2.3 shows the distribution of the studies according to the problem type. The main problem in 53% of the studies examined in this review is the OBP.

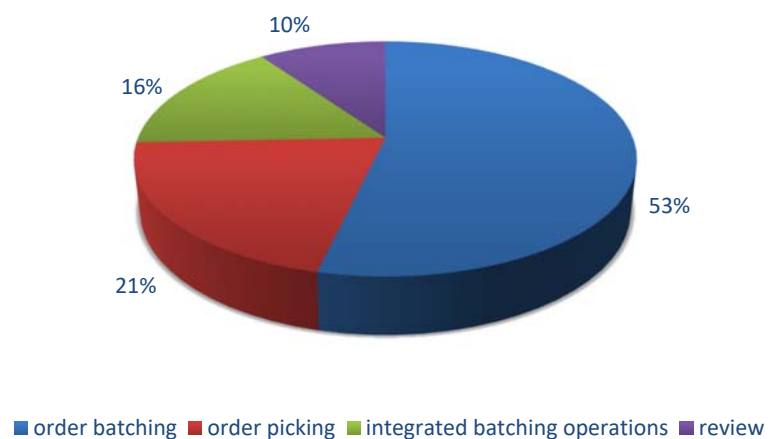


Figure 2.3 Problem types of examined studies in the related literature

The studies on order picking process are classified under the following subheadings, and main characteristics of the studies are summarized.

2.2.1 Simulation Based Studies

At first, simulation based studies draw attention about order picking process. Petersen & Aase (2004) examine the effect of picking, storage, and routing operations on order picker travel. A simulation model that is based on the activities of a distribution center is developed. Different policies are evaluated to find the process decision that yields the highest percent savings compared with the existing basic policies, and sensitivity analyses are implemented. Results indicate order batching yields the highest savings particularly in case of smaller order sizes. It is also found that the use of either a class-based or volume-based storage policy results with approximately the equal savings as batching, while being less sensitive to the average order size. In their case specific simulation study, Dekker, de Koster, Roodbergen, & van Kalleveen (2004) reduce average route length by applying routing and storage policies. Yu & de Koster (2009) study a queuing network model for evaluating the order picking system performance. Queuing network consists of pick zones and conveyor pieces. Their paper considers dynamic batching approach. Results are evaluated after a simulation study, and the algorithm is found as applicable to real cases. The study of Le-Duc & de Koster (2007) is a simulation study which aims to estimate the average throughput time of a random order. In order picking operations, S-shape heuristic is used as a routing heuristic. Schleyer & Gue (2012) examine order picking process by making analyses on the batch size and travel time in the operations of a single block warehouse. Pan, Wu, & Chang (2014) propose a travel time estimation model to evaluate performance of order picking process and find an optimal warehouse layout design. Authors consider class-based storage and use different routing strategies. Hong, Johnson, & Peters (2015) consider bucket brigade order picking systems and investigate effects of different factors on picker blocking delay. In another study of Hong, Johnson, & Peters (2016), authors suggest a MIP model to reduce picker blocking.

2.2.2 Exact, Heuristic and Metaheuristic Approaches

In the literature, there have been several studies that utilize different solution methods including exact methods, heuristics and metaheuristics. Ratliff & Rosenthal (1983) develop a graph based solution procedure for the order picking problem. Their algorithm is found as linear in the number of aisles. In the study of Brynzér & Johansson (1995), authors make a design and evaluation of the picking performance of a kitting process by taking into consideration several case studies. Roodbergen & de Koster (2001) present a branch-and-bound (B&B) algorithm and routing heuristics for the warehouses that have multiple cross aisles. Proposed methods are experimented on different warehouse layouts. Theys, Bräysy, Dullaert, & Raa (2010) bring two solution approaches for the routing operation in warehouses. Authors use Lin–Kernighan–Helsgaun traveling salesman problem (TSP) heuristic and 2-opt local search heuristic with four different picking strategies (S-shape, largest gap, combined, aisle-by-aisle) to explore which combination yields better results among others. In the study of Hsieh & Huang (2011), K-means Batching (KMB) and Self-organisation Map Batching (SOMB) heuristics are suggested for evaluating performance of the order picking systems. Integration of diverse storage assignment, order batching and picker routing policies under order environments with different size are also investigated by the experiments. Results of these heuristics are compared with strict-order picking, association rule and particle swarm optimization (PSO) batching method policies. The study brings a holistic view to an order picking system. Atmaca & Ozturk (2013) suggest a mixed integer programming (MIP) model and a simulated annealing (SA) algorithm for determining the picking policy in an automated storage and retrieval system (AS/RS). Chen, Wang, Xie, & Qi (2016) propose an ant colony optimization (ACO) approach to solve the routing problem of multiple pickers in a congestion case. Ming-Huang Chiang, Lin, & Chen (2014) suggest an association rule mining based approach for order picking operations in warehouses. Modified class-based heuristic (MCBH) and association seed based heuristic (ASBH) are proposed, and an experimental study is implemented with real data set. Results are compared with class based allocation approach (CBAA) and

found effective. Grosse & Glock (2015) examine the learning effect in order picking process with a mathematical model.

2.2.3 Review Studies

Since the emergence of the problem, a number of review studies about warehouse operations that focus on order picking and order batching have been proposed. Van den Berg & Zijm (1999) examine warehousing systems with a classification, and give general and introductory warehousing model examples. Cormier (2005) analyses warehouse operations in many aspects, and batching policies are investigated under the title of throughput capacity models. The study of de Koster et al. (2007) draws a frame for order picking. Study presents the importance and process of order picking in warehouse operations in a broad sense. Several studies from the literature are mentioned in the sections of layout design, storage assignment types, zoning, batching, routing strategies, and order accumulation and sorting methods. As a result of their review study, authors state that future research would focus on optimization problems that include stochastic demands, decision problems that are combinations of one or more picking operations, and problems which consider storage assignment type as a decision for routing operation.

In the review study of Gu, Goetschalckx, & McGinnis (2007), order batching studies are classified according to heuristics in the framework of warehouse operations, and also order closeness metrics are investigated. A comprehensive review on AS/RSs is implemented by Roodbergen & Vis (2009). The study by Henn, Koch, & Wäscher (2012) gives an investigation on the order batching solution methods. After main characteristics of the problem are described, existing solution approaches are examined under two main titles as constructive approaches and metaheuristic approaches. Dynamic order batching, due date related objectives and combination of other warehouse operations are stated as trending topics for further research. The study of Ong & Joseph (2014) reviews twenty studies about order picking improvement approaches. According to their review, order picking methods

are basically examined under four headings in the articles: stock-to-picker systems, storage assignment policy, order batching, and order picking sequencing.

Beside these reviews and taxonomy papers, the research in this literature review section is based on more methodological; and aims to give future research directions according to latest developments and trends. Table 2.1 summarizes selected studies on order picking process.

In the related literature, order picking process is discussed in many aspects. Most of the examined studies suggest simulation based solution methods to evaluate performance of the order picking process, and some studies investigate solution approaches for integrated operations of picking, batching and sorting processes. In addition, metaheuristics and data mining based approaches are also proposed for the problem.

Review studies display a general framework, however it is found that proposed solution approaches are completely problem dependent and particularly investigated within the corresponding study. A significant majority of the examined studies consider 2-dimensional (2D) warehouse layouts. Consequently, it is also seen that there is a lack of general models for application to different warehouse layouts and configurations.

Table 2.1 Studies on order picking process in the literature

Authors	Examined problem	Solution method	Main concern	Synthesized problem	Case study	Warehouse layout	
						2D	3D
Ratliff & Rosenthal (1983)	order picking	graph based	minimize length of the order-picking tour	✓		✓	
Brynzér & Johansson (1995)	order picking	case study based research	find out design and performance issues related with materials kitting and order picking systems		✓	✓	
Roodbergen & de Koster (2001)	picker routing	B&B and heuristics	minimize travel times	✓		✓	
Petersen & Aase (2004)	picking, storage, and routing operations	simulation	improve overall system performance by taking into consideration picking, routing and storage assignment issues		✓	✓	
Dekker et al. (2004)	order picking	simulation	reduce average route length		✓	✓	
Le-Duc & de Koster (2007)	order picking	simulation	estimate the average throughput time of a random order	✓		✓	
Yu & de Koster (2009)	order picking system performance	queuing network model	analyze the mean order throughput time of a random order		✓	✓	
Theys et al. (2010)	picker routing	TSP heuristic, 2-opt	minimize distance travelled	✓		✓	
Hsieh & Huang (2011)	order picking system	KMB and SOMB	minimize total travel distance	✓		✓	
Atmaca & Ozturk (2013)	order picking	MIP and SA	minimize storage allocation, storage retrieval and storage keeping costs	✓	✓	✓	
Schleyer & Gue (2012)	order picking	queuing model	analyze throughput time	✓		✓	

Table 2.1 continues

Authors	Examined problem	Solution method	Main concern	Synthesized problem	Case study	Warehouse layout	
						2D	3D
Pan et al. (2014)	order picking	simulation	estimate the travel time	✓			✓
Ming-Huang Chiang et al. (2014)	order picking	association rule mining based approach	minimize travel distance		✓	✓	
Grosse & Glock (2015)	order picking	mathematical model	model worker learning in order picking	✓	✓	✓	
Hong et al. (2015)	picker blocking	analytical model and simulation	estimate picker blocking delay in bucket brigade picking system	✓		✓	
Hong et al. (2016)	picker blocking	MIP	reduce picker blocking in bucket brigade picking system	✓		✓	
Chen et al. (2016)	picker routing	ACO	minimize total order service time	✓		✓	

2.3 Order Batching Problem

Order batching is a common problem that is encountered during the process of order picking. When several customer orders are demanded, it is expected to pick up all of the orders with minimum cost. The cost function can be stated in terms of distance travelled, elapsed time, etc. When the problem is studied as a maximization problem, the objective function would be customer satisfaction level, system performance, number of orders completed in a time interval, etc.

Order batching problem (OBP) can be described as the problem of finding the batching combination with minimum cost containing a list of customer orders. OBP is composed of the batch construction and batch routing levels. In this problem, each order can have different number and volume of items. The order picker travels the whole storage area with a picking facility which has a limited capacity. At this point, multiple batches must be created to pick up all of the orders. While the number of

orders increases, the number of possible batches grows exponentially. The problem is known to be NP-hard in the strong sense and found as solvable in polynomial time if each batch has at most two orders (Gademann & Van de Velde, 2005).

2.3.1 Mathematical Formulation

The OBP can be formulated as follows with refer to Gademann & Van de Velde (2005), and Henn, Koch, Doerner, Strauss, & Wäscher (2010). Assume that $J = \{1, \dots, n\}$ is the set of customer orders, C is the capacity of the picking device, and c_j is the total capacity of the order j ($j \in J$). Batches related with customer orders are defined by the vector $a_i = (a_{i1}, \dots, a_{in})$. Binary entry a_{ij} shows whether an order j is involved in a batch i ($a_{ij} = 1$) or not ($a_{ij} = 0$). The set I of all feasible batches is characterized such that capacity of the picking vehicle is not violated, and constraint (2.1) displays this feature:

$$\sum_{j \in J} c_j a_{ij} \leq C, \forall i \in I \quad (2.1)$$

Binary decision variables x_i ($i \in I$) define if a batch i is chosen ($x_i = 1$) or not ($x_i = 0$), and d_i represents length of the picking tour on that all orders of a batch i are picked up, then the following mathematical formulation is given:

$$\min \sum_{i \in I} d_i x_i \quad (2.2)$$

$$s. t. \sum_{i \in I} a_{ij} x_i = 1, \forall j \in J \quad (2.3)$$

$$x_i \in \{0,1\}, \forall i \in I \quad (2.4)$$

Objective function (2.2) minimizes total length of the picking tour of all batches. Here, constraints (2.3) and (2.4) ensure that each customer order is included in exactly one of the chosen batches. This formulation is a very basic one, and different objective function and decision variables can be presented in the studies. Additional constraints could be included in the model according to the examined problem. Different models on order batching can be found in Elsayed, Lee, Kim, & Scherer

(1993), Chen & Wu (2005), Henn & Schmid (2013), Kulak, Sahin, & Taner (2012), Chen, Cheng, Chen, & Chan (2015).

Since the previous model considers that the batch formations and the lengths of the picking tours are known, a more detailed mathematical model is needed to formulate the problem. The new mathematical model constructs the batches, and computes the total length of the related picking tour according to a routing scheme. The notation of the proposed model can be claimed as follows with refer to Kulak et al. (2012), Chen et al. (2015), and Öncan (2015). In this mixed integer optimization model, K is the set of customer orders ($k \in K$) and V represents storage locations in the warehouse ($i, j \in V$). S is the subset of the locations ($S \subset V$) and B corresponds to set of batches ($b \in B$). C is the capacity of the picking device. w_k represents weight of the order k , and the distance between location i and j is given by the parameter d_{ij} . Binary parameter a_{ik} describe if an item of order k is picked from location i ($a_{ik} = 1$) or not ($a_{ik} = 0$). Additively, decision variables in the model can be explained as follows. X_k^b is equal to 1 if order k is assigned to batch b ($=0$, otherwise). Y_{ij}^b is equal to 1 if location i is visited right after location j in batch b ($=0$, otherwise). Z_i^b is equal to 1 if location i is visited in batch b ($=0$, otherwise). The set of all feasible batches is characterized such that capacity of the picking device is not violated. Accordingly, the following model formulation can be given:

$$\min \sum_{b \in B} \sum_{i \neq j \in V} d_{ij} Y_{ij}^b \quad (2.5)$$

s. t.

$$\sum_{j \in V, j \neq i} Y_{ij}^b = Z_i^b \quad \forall b \in B, i \in V \quad (2.6)$$

$$\sum_{i \in V, i \neq j} Y_{ij}^b = Z_j^b \quad \forall b \in B, j \in V \quad (2.7)$$

$$\sum_{i \in S, j \in V \setminus S} Y_{ij}^b \geq Z_i^b \quad \forall b \in B, S \subset V \quad (2.8)$$

$$Z_i^b \geq a_{ik} X_k^b \quad \forall b \in B, i \in V, k \in K \quad (2.9)$$

$$\sum_{b \in B} X_k^b = 1 \quad \forall k \in K \quad (2.10)$$

$$\sum_{k \in K} w_k X_k^b \leq C \quad \forall b \in B \quad (2.11)$$

$$X_k^b, Y_{ij}^b, Z_i^b \in \{0,1\}, \quad \forall i, j \in V, b \in B, k \in K \quad (2.12)$$

In the objective function (2.5), total distance of all batches and routes is minimized. Connected with the condition that related location is visited in the route, constraints (2.6) and (2.7) guarantee each storage location has one predecessor and successor, respectively. Constraint (2.8) is the sub tour elimination constraints related with the TSP that ensure solutions consist of only one tour linking all the storage locations in a batch. With constraint (2.9), binary variables for related locations that will be visited are enforced to 1. Constraint (2.10) ensures that each customer order is included in exactly one batch. Constraint (2.11) displays the capacity constraints of the picking device, and (2.12) describes the variable properties.

In the studies on OBP, different assumptions are used in the experiments and the objective for the problem can vary between studies. For example, as an objective function, minimizing total distance of all batches, total or average travel/picking time elapsed, total/average throughput time of orders are considered. If the organization pays attention to its customers, not only minimizing the total cost, but also reaching a high customer satisfaction level is important. At this point, it is a necessity to include a customer based viewpoint in the objective function to ensure and also improve customer satisfaction. In addition, there could be time windows for the orders that should not be disregarded; therefore, order batching problem can have many different characteristics among the studies.

2.3.2 Solution Methods for Order Batching Problem

In the related literature, OBP is mainly examined with exact, heuristic and metaheuristic methods. In the following sections, detailed investigation on these studies will be given. Beside them, some of the studies use queuing network models with different performance measures, some of them employ policies related with

picking operations. Since the batching operation is a complicated and time consuming task, many papers address to fast and least cost batching procedures.

In the study of Elsayed et al. (1993), a penalty function is considered for batching operation in a warehouse with an AS/RS system. Tang & Chew (1997) propose a queuing model for a picker-to-parts system. Authors use time window batching approach and make a comparison between different order batching and location strategies. Van Nieuwenhuyse, de Koster, & Colpaert (2007) study effectiveness of online order batching and sorting operations using a queuing network approach. Authors utilize examples from the literature, and S-shape picking is considered for picker routing. Optimal pick batch size, optimal allocation of workforce to the picking and sorting operations, and comparison of sort while pick (SWP) and pick and sort (PAS) strategies are made through experiments with different scenarios. Parikh & Meller (2008) investigated whether a batch picking or zone picking is suitable for the existing distribution center, and a cost model is proposed to make a selection between picking strategies. A model for fixed time window batching is proposed by Van Nieuwenhuyse & de Koster (2009). In their model, average customer order throughput time is estimated. Hong, Johnson, & Peters (2012) discuss an order batching formulation and heuristic technique convenient for a large-scale order picking case in parallel-aisle picking systems. The solutions are compared with a lower bound developed as a LP relaxation of the batching formulation in the article. It is seen that with the implementation of a simulation study, developed heuristic method outperforms other approaches from the literature or used in practice.

2.3.2.1 Exact Methods

As the exact solution techniques, most of the research relies on applications with small instances. Armstrong, Cook, & Saipet (1979) study the OBP for a liquor distribution center case. The warehouse is a semi-automated system, and authors propose a MIP formulation then experimental study is performed with a 30 order/12 aisle example. Ruben & Jacobs (1999) study the batch picking problem with batch construction heuristics and a simulation model. Gademann, Van Den Berg, & Van

Der Hoff (2001) study the wave picking operation and propose an order batching algorithm that minimizes the maximum lead time of any of the batches. Authors present a B&B algorithm as an exact solution method for the problem and utilize from the 2-opt heuristic as an initial upper bound. The OBP is evaluated as proximity batching, and it is stated by the authors that the problem is NP-hard. In the paper of Gademann & Van de Velde (2005), OBP is investigated with a branch-and-price optimization algorithm, and an iterated descent approximation algorithm is proposed to solve the problem. Bozer & Kile (2008) present a MIP model for the problem and make a comparison with the normalized time savings (NTS) heuristic and R&J heuristic derived from Ruben & Jacobs (1999). Tang, Wang, Liu, & Liu (2011) examine the OBP with a MIP approach which is a combination of Lagrangian relaxation and column generation. A steelmaking and continuous-casting system is studied as a real case, and a number of numerical experiments are implemented to both real data and generated large scale instances. A column generation based exact solution method is developed by Muter & Öncan (2015) to solve OBP. Authors consider different routing strategies in the experiments, and use 90 data sets which include up to 100 orders.

2.3.2.2 Heuristic and Metaheuristic Methods

Since exact methods for the OBP are only practical for instances with small sizes, heuristic approaches are developed. The savings algorithm which was proposed by Clarke & Wright (1964) as a solution method for a vehicle routing problem also constitutes a basis for the OBP. Elsayed & Unal (1989) suggest four heuristics for the batching problem, and compare effectiveness of the heuristics by means of total travel time. Pan & Liu (1995) study seed selection and addition algorithms (SSAA), and also a SL method for the OBP in a storage/retrieval (S/R) system. Experiments are implemented with generated data, and it is also investigated that if shape factor, capacity of the S/R machine, and storage assignment policy have an effect on selection of the order batching algorithm. As a final result, economic convex hull algorithm is found to give best results. Rosenwein (1996) studies the OBP with heuristics, and uses different distance metrics for the assigning process of orders to

batches. Lam, Choy, Ho, & Lee (2014) present a mathematical model combined with a fuzzy logic technique to the batch picking operation.

The *metaheuristics* stand out for solving OBP in recent years. In the study of Albareda-Sambola, Alonso-Ayuso, Molina, & De Blas (2009), a metaheuristic approach which is based on variable neighborhood search (VNS) is proposed for the OBP and tested with four different warehouse configurations. The objective in their algorithm is to minimize total travel time, and the algorithm is compared to other methods in the literature. Henn et al. (2010) present metaheuristic approaches for the problem. Iterated local search (ILS) and a rank based ant system (RBAS) algorithm are proposed. Five problem sets are created for experiments, besides S-shape and largest gap routing strategies are used for picking orders. Results are compared with the following heuristics from the literature: FCFS, a variant of the savings algorithm (C&W(ii)), C&W(ii)+local search, iterative descent from Gademann & Van de Velde (2005). According to the results, in terms of computational time, ILS gives solutions faster than RBAS. Authors also state that it would be a good approach to consider dynamic batching, customer response time and developing faster metaheuristics as future research. Henn & Wäscher (2012) propose tabu search (TS) heuristic and attribute-based hill climber (ABHC) heuristic for the problem in which total length of the picking tours is minimized. In the study, both S-shape and largest gap routing strategies are used. Authors utilize from C&W(ii)+local search, ILS-1 and ILS-2 heuristics for comparison. Online batching and integration of one or more processes in picking operations are pointed out to be trend topics in future studies. Henn (2012a) studies dynamic OBP and an optimization model is proposed. S-shape and largest gap heuristics are used as routing strategies, and FCFS, C&W(ii), ILS heuristics are used for batching options. A competitive analysis is implemented to see the comparison of proposed online algorithms with offline algorithms when they are applied to the online case. For further research, using different types of layouts and assignment methods, and due date related objectives are suggested. In the study of Matusiak, de Koster, Kroon, & Saarinen (2014), authors study the precedence constrained batching problem. An A*-algorithm is used for routing part and a SA approach is used for the batching part of the algorithm. Objective function in the

algorithm is determined as minimization of total routing costs of the batches. Menéndez, Pardo, Sánchez-Oro, & Duarte (2017b) study the min-max OBP, and developed VNS based algorithms to solve the problem. Žulj, Kramer, & Schneider (2018) develop an adaptive large neighborhood search (ALNS) and TS for the OBP, and report that their algorithm is effective at solving problem instances with large sizes. Lin, Kang, Hou, & Cheng (2016) solve the OBP with a PSO algorithm. Li, Huang, & Dai (2017) study the OBP focused on a retailer's warehouse. They propose an ACO algorithm with local search. In the study of Scholz & Wäscher (2017), batching and routing problem is investigated, and an ILS algorithm is proposed. The Kiva mobile fulfilment system is examined in the study of Xiang, Liu, & Miao (2018). The study proposes MIP models for storage assignment and the OBP, and a VNS algorithm for the OBP. Nicolas, Yannick, & Ramzi (2018) examine an automated picking system with vertical lift modules, and the objective is defined as minimization of the total time required to collect all customer orders. The authors use a SA approach.

In the literature, *genetic algorithm based solution approaches* are utilized in several studies to solve the problem. Tsai, Liou, & Huang (2008) study the batch picking problem in which total travel cost with earliness and tardiness penalties is minimized. Authors utilize from seven different problem sets that have 2-dimensional (2D) or 3-dimensional (3D) warehouse layouts, and they suggest a genetic algorithm (GA) approach to solve the problem. Their approach tries to find an effective batch combination for picking with the GA_BATCH algorithm, then travel path in a batch is tried to be minimized by the GA_TSP algorithm. Authors compare their model with two models in which one considers only travel cost and other considers only earliness/tardiness penalty as the objective function. Hsu, Chen, & Chen (2005) present a GA based batching method for the OBP. Their algorithm minimizes total travel distance, and several experiments are implemented for 10 problem sets that are created. Both 2D and 3D warehouse layouts are considered in the experiments. Benchmarks are made with Gibson & Sharp (1992)'s batching method and first come first serve (FCFS) heuristic. Chirici & Wang (2014) also proposed GAs for the problem. Authors made a numerical experiment with data sets

including up to 60 orders, and S-shape routing heuristic is used. Results of item-oriented genetic algorithm (IGA) and group-oriented genetic algorithm (GGA) are compared with C&W(ii) algorithm. According to results, GGA ensures more improvements in the objective in comparison to IGA. Hong & Kim (2017) propose an S-shape route selecting batching model (SRSB), a heuristic procedure and a lower bound algorithm for a parallel aisle picking system. Şahin & Kulak (2013) propose GA based solution methods for the OBP in warehouses. Experiments are implemented to different warehouse configurations. Öncan (2013) also develops a GA for the OBP. Experimental results are compared with the savings algorithm. Considered warehouse layout is 2D, and has the capacity of storage for 200 items. Azadnia, Taheri, Ghadimi, Mat Saman, & Wong (2013) propose a hybrid solution approach for the OBP where customer due dates are considered. Authors use a weighted association rule mining approach to cluster customer orders first, a binary integer programming (BIP) model for constructing batches, and finally a GA for the picker routing phase to minimize total distance of the batches. Koch & Wäscher (2016) develop a grouping GA for the OBP, and make a comparison with the standard item centered GA. The authors state the grouping GA is preferable over the standard GA. A hybrid grouping GA is also proposed by Mutingi & Mbohwa (2017). The performance of the approach is compared with the item centered GA, and evaluated in terms of the solution quality and the computational time.

In recent years, studies on *integrated picking operations* are noteworthy. Won & Olafsson (2005) analyze the effects of integrated order batching and picking activities. Authors propose two heuristic approaches to the order batching and sequencing problem (OBSP) and their model minimizes the weighted average of the total collection time of batches and also holding time of the orders. The study of Kulak et al. (2012) concentrates on integrated approach for order batching and routing. They propose a cluster based seed algorithm and TS algorithm for the problem. Henn (2012b) studies the order batching and sequencing problem with multiple pickers (OBSPMP), and proposes a variable neighborhood descent (VND) and a VNS algorithm to minimize the total tardiness of customer orders. The author benefits from the earliest start date rule for an preliminary solution. S-shape and

largest gap routing heuristics, and a class-based storage assignment strategy are considered. A single-block layout with 900 storage locations (10 picking aisles and 90 storage locations for each aisle) and two cross aisles are used for the experimental study. According to results, VND outperforms VNS especially by means of computing times. Henn & Schmid (2013) present ILS and ABHC heuristics for the order OBSP. The objective is given as minimizing total tardiness of all customer orders, and both S-shape and largest gap routing heuristics are used. Authors make a comparison study with two earliest due date approaches. Extensions of the problem in which priorities of customers are considered, dynamic cases and integrated order picking problems are suggested to taken into account in the further studies. The study of Chen et al. (2015) focuses on integrated order batching, sequencing and routing problem (IOBSRP). Authors suggest a hybrid coded genetic algorithm (HGA) for the order batching and sequencing processes, and an ACO method for the routing process. Proposed nonlinear mixed integer optimization model is a modification of models suggested by Kulak et al. (2012) and Henn & Schmid (2013). Experimental study is implemented with up to 8 orders and 10 items. Cheng, Chen, Chen, & Yoo (2015) propose PSO and ACO based hybrid algorithm to solve order batching and picker routing problem. Zhang, Wang, & Huang (2016) focus on integrated on-line order picking and distribution scheduling problem (IOOPDS), and examine dynamic batching in B2C operations. Authors propose rule based solutions to the problem. Valle, Beasley, & da Cunha (2017) present an IP formulation to the joint order batching and picker routing problem (JOBPRP), and develop a solution that proves optimality for the instances involving up to 20 orders. Scholz, Schubert, & Wäscher (2017) examine the joint order batching, assignment and sequencing, and routing problem (JOBASRP). A mathematical model is presented and the problem is solved with a VND algorithm. Menéndez, Bustillo, Pardo & Duarte (2017a) examine the OBSP with general variable neighborhood search (GVNS) approach. Zhang, Wang, Chan, & Ruan (2017) propose rule based solution approach for the online OBSP where picking is implemented by multiple pickers. In the study of Matusiak, de Koster, & Saarinen (2017), batching operations are analyzed via a joint batching and generalized assignment problem (BatchGAP) model, and an adaptive large neighborhood search algorithm (ALNS) is proposed. Van Gils, Ramaekers, Braekers,

Depaire, & Caris (2018) conduct simulation experiments on integration of the storage, batching, picking and routing activities, and a real case study is implemented.

2.3.2.3 Data Mining Approaches

The increasing focus on data mining methods helps research on arrangement of the data according to relationships to be increased. The study of Chen & Wu (2005) brings data mining and integer programming approaches together for the OBP. The paper concentrates on maximizing order associations in a batch. Experiments are implemented in a 2D warehouse layout with generated problem sets which have order sizes that range from 40 to 300, and the S-shape routing strategy is used in computations. Hwang & Kim (2005) develop a cluster analysis based batching algorithm and use three routing strategies for picking orders: traversal, return, and midpoint routing. In the study of Chen, Huang, Chen, & Wu (2005), a data mining method based on association rules is developed for batching orders in a parallel-aisle warehouse. Ho & Tseng (2006) propose several batching methods based on seed-order selection rules (SOSR) and accompanying-order selection rules (AOSR) for the problem. Experiments are implemented under two route-planning methods (largest gap and largest gap with SA improvement), and two aisle-picking-frequency distributions. Random generated test examples are used in the computations, and minimization of total order-picking travel distance is accepted as the objective. Best combination of these rules is found with the help of the ANOVA results. Ho, Su, & Shi (2008) continue the study of Ho & Tseng (2006). Authors present some new rules and add a new distance/area based rules. Their study also reveals the interaction between SOSR and AOSR. Authors use total travel distance as the performance measure and state that examining the problem as a single problem -not many sub problems as in the study- is suggested as a further research. A clustering approach for the order batching operation is suggested by Hsieh & Fan (2010). In their paper, SOMB approach is experimented to maximize average picking vehicle utility and minimize total order picking distances.

2.4 Findings

The output of this literature review is given in this section. Table 2.2 demonstrates studies from the literature examining order batching, and 57 papers together with their objectives and characteristics are mentioned. It has to be noted that types of warehouse layout are indicated as 2D or 3D according to the data sets used. It is also clear that case studies are not implemented in most of the papers, and 3D layouts which can be thought as warehouses with high racks are rarely encountered. Consequently, the lack of such studies in the literature has come to light.

According to studies, it is found that OBP is examined with many approaches. In the related literature MIP models, branch-and-price, B&B algorithms are mainly used as exact techniques. Heuristic approaches such as savings algorithm, priority based heuristics are utilized to provide good solutions in a short time interval. In many studies, several metaheuristic approaches including GA, ant systems, VNS, SA, TS and also association rule mining based solution approaches take place (Cergibozan & Tasan, 2019).

Table 2.3 shows the programming language, order and picker information used in the studies focused on the OBP. To the best of our knowledge, a library of standard test data does not exist in the OBP literature. In addition, solution environments and computation platforms (e.g. hardware and software) differ between the studies. Consequently, making a comparison between the studies by means of computation times would not be a proper approach in this situation. However, it is thought that the information in Table 2.3 will be helpful for generation of standard data sets for the OBP in the future studies (Cergibozan & Tasan, 2019).

Table 2.2 Studies on order batching in the related literature

Authors	Examined problem	Solution method	Main concern	Synthesized problem	Case study	Warehouse layout	
						2D	3D
Armstrong et al. (1979)	order batching	MIP	minimize the total time required to complete all batches		✓	✓	
Elsayed & Unal (1989)	order batching	heuristics	minimize total travel time	✓		✓	✓
Elsayed et al. (1993)	batching and sequencing	simulation	minimize earliness and tardiness penalties	✓		✓	✓
Pan & Liu (1995)	order batching	SSAA	minimize total distance travelled by the S/R machine	✓		✓	
Rosenwein (1996)	order batching	heuristics	minimize total distance travelled	✓		✓	
Tang & Chew (1997)	order batching and storage assignment	queuing model	measuring the average time an order spends in the system (minimize expected delay time of all customer orders)	✓		✓	
Ruben & Jacobs (1999)	batch picking	heuristics and simulation model	measure performance in terms of total distance, total man hours, percent idle time and capacity utilization	✓		✓	
Gademann et al. (2001)	order batching	B&B, 2-opt	minimize the maximum lead time of any of the batches	✓		✓	

Table 2.2 continues

Authors	Examined problem	Solution method	Main concern	Synthesized problem	Case study	Warehouse layout	
						2D	3D
Hsu et al. (2005)	order batching	GA	minimize total travel distance	✓		✓	✓
Gademann & Van de Velde (2005)	order batching	branch and price, iterated descent algorithms	minimize total travel time	✓		✓	
Chen & Wu (2005)	order batching	data mining and integer programming	maximize order associations in a batch	✓		✓	
Won & Olafsson (2005)	order batching and sequencing	heuristics	minimize the weighted average of the total picking time of tours or batches and the holding time of the orders in the batch	✓		✓	✓
Hwang & Kim (2005)	order batching	cluster analysis based batching algorithm	minimize the total travel distance of the picker	✓		✓	
Chen et al. (2005)	order batching	association rule based data mining	minimize travel distance	✓		✓	
Ho & Tseng (2006)	order batching	SOSR and AOSR	minimize total order-picking travel distance	✓		✓	
Van Nieuwenhuysen et al. (2007)	online order batching and sorting	queuing network approach	estimate average throughput time of a customer order	✓		✓	
Bozer & Kile (2008)	order batching	MIP model and heuristics	minimize the total distance required to pick all the orders	✓		✓	

Table 2.2 continues

Authors	Examined problem	Solution method	Main concern	Synthesized problem	Case study	Warehouse layout	
						2D	3D
Tsai et al. (2008)	batch picking	GA	minimize total travel cost with earliness and tardiness penalties	✓		✓	✓
Parikh & Meller (2008)	order batching	cost model	estimate cost of alternative picking strategies	✓		✓	
Ho et al. (2008)	order batching	new SOSR and AOSR	minimize total travel distance	✓		✓	
Albareda-Sambola et al. (2009)	order batching	VNS	minimize total travel time	✓		✓	
Van Nieuwenhuysen & de Koster (2009)	time window batching	simulation	estimate average customer order throughput time	✓		✓	
Henn et al. (2010)	order batching	ILS, RBAS	minimize total length of all tours	✓		✓	
Hsieh & Fan (2010)	order batching	clustering approach (SOMB)	minimize total order picking distances	✓		✓	
Tang et al. (2011)	order batching	MIP	minimize costs including assignment cost, inventory holding cost and penalty cost		✓	✓	
Hong et al. (2012)	order batching	heuristics with a simulation model	minimize the total travel distance	✓		✓	
Henn & Wäscher (2012)	order batching	TS, ABHC	minimize total length of the picking tours	✓		✓	

Table 2.2 continues

Authors	Examined problem	Solution method	Main concern	Synthesized problem	Case study	Warehouse layout	
						2D	3D
Henn (2012a)	dynamic order batching	optimization model and heuristics	minimize the maximum completion time of the customer orders	✓		✓	
Henn (2012b)	OBSPMP	VND, VNS	minimize the total tardiness of customer orders	✓		✓	
Kulak et al. (2012)	order batching and routing	cluster based seed algorithm and TS	minimize the total travel distance over all batches and tours	✓		✓	
Henn & Schmid (2013)	OBSP	ILS, ABHC	minimize total tardiness of all customer orders	✓		✓	
Chirici & Wang (2014)	order batching	GA	minimize total length of the picking tours	✓		✓	
Azadnia et al. (2013)	order batching	weighted association rule mining, BIP, GA	minimize total tardiness of customer orders	✓			✓
Öncan (2013)	order batching	GA	minimize total travel length of pickers	✓		✓	
Şahin & Kulak (2013)	order batching	GA	minimize total travel distance	✓		✓	✓
Matusiak et al. (2014)	precedence constrained batching	SA	minimize total routing costs of the batches		✓		✓

Table 2.2 continues

Authors	Examined problem	Solution method	Main concern	Synthesized problem	Case study	Warehouse layout	
						2D	3D
Lam et al. (2014)	batch picking	mathematical model with fuzzy logic	minimize the total travel distance to pick all orders		✓	✓	
Chen et al. (2015)	IOBSRP	HGA and ACO	minimize total tardiness of customer orders	✓		✓	
Muter & Öncan (2015)	order batching	column generation	minimize total travel cost of the pickers	✓		✓	
Cheng et al. (2015)	order batching and picker routing	PSO, ACO	minimize total travel distance	✓	✓	✓	
Koch & Wäscher (2016)	order batching	grouping GA	minimize total length of all tours	✓		✓	
Lin et al. (2016)	order batching	PSO	minimize the total picker Manhattan routing distance	✓	✓	✓	
Zhang et al. (2016)	IOOPDS	rule based solutions	minimize service time, maximize the number of delivered orders	✓		✓	
Hong & Kim (2017)	order batching	SRSB model, heuristics	minimize travel distance of the picker	✓		✓	
Menéndez et al. (2017a)	OBSP	GVNS	minimize total tardiness of all orders	✓		✓	
Mutingi & Mbohwa (2017)	order batching	grouping GA	minimize the total tour length to pick the requested items	✓		✓	

Table 2.2 continues

Authors	Examined problem	Solution method	Main concern	Synthesized problem	Case study	Warehouse layout	
						2D	3D
Li et al. (2017)	order batching and picker routing	ACO	minimize the picking distance		✓		✓
Scholz & Wäscher (2017)	order batching and picker routing	ILS	minimize total length of all picker tours	✓		✓	
Menéndez et al. (2017b)	min-max OBP	VNS	minimize total travel distance	✓		✓	
Zhang et al. (2017)	online order batching and sequencing problem	rule based solutions	minimize the turnover time of all orders	✓		✓	
Matusiak et al. (2017)	BatchGAP	ALNS	minimize the sum of costs of all pickers	✓	✓	✓	
Valle et al. (2017)	JOBPRP	IP model	minimize total length	✓		✓	✓
Scholz et al. (2017)	JOBASRP	mathematical model, VND	minimize total tardiness	✓		✓	
Xiang et al. (2018)	storage assignment and order batching	MIP models and VNS	minimize the total product similarity in all pods, and number of visits of pods	✓		✓	
Nicolas et al. (2018)	order batching with VLMs	mathematical model, SA	minimize the total time needed to collect all orders	✓	✓		✓
Žulj et al. (2018)	OBP	ALNS, TS	minimize total length of all tours	✓		✓	
Van Gils et al. (2018)	order picking	simulation	increase efficiency by combining related processes		✓	✓	

Table 2.3 Detailed information on application environment

Authors	Computation environment	Order information		
		Maximum number of orders	Maximum number of items in order/total	Maximum capacity of the picker
Armstrong et al. (1979)	NA	30	-	-
Elsayed & Unal (1989)	Fortran 77	52	-	350
Elsayed et al. (1993)	NA	8	-/524	1200
Pan & Liu (1995)	Turbo Pascal 6.0	50	10/-	200
Rosenwein (1996)	Fortran	1000	6/-	50
Tang & Chew (1997)	NA	-	4/-	-
Ruben & Jacobs (1999)	C++, ANOVA	4000	15/-	30
Gademann et al. (2001)	NA	32	20/-	-
Hsu et al. (2005)	Visual C++ 6.0	300	-/400	500
Gademann & Van de Velde (2005)	C and CPLEX 5.0	32	20/-	-
Chen & Wu (2005)	Visual Basic 6.0, OPL	300	-/400	500
Won & Olafsson (2005)	NA	30	5/-	30
Hwang & Kim (2005)	CPLEX 7.0	30	10/-	24
Chen et al. (2005)	NA	300	-/400	500
Ho & Tseng (2006)	ANOVA	250	36/-	80
Van Nieuwenhuyse et al. (2007)	NA	44	4/-	-
Bozer & Kile (2008)	NA	25	5/-	25
Tsai et al. (2008)	NA	250	10/400	50000
Parikh & Meller (2008)	CPLEX	12000	-/36000	75
Ho et al. (2008)	ANOVA	250	-	-
Albareda-Sambola et al. (2009)	C++ 6.0	250	-/1250	150
Van Nieuwenhuyse & de Koster (2009)	NA	48	-	-
Henn et al. (2010)	C++, CPLEX 10.1	60	25/-	75
Hsieh & Fan (2010)	ANOVA	-	50/-	50
Tang et al. (2011)	C++, IBM OSL	1000	25/-	-
Hong et al. (2012)	C, CPLEX and ANOVA	2160	9/-	50
Henn & Wäscher (2012)	C++, CPLEX 10.1	100	25/-	75
Henn (2012a)	C++	120	25/-	75

Table 2.3 continues

Authors	Computation environment	Order information		
		Maximum number of orders	Maximum number of items in order/total	Maximum capacity of the picker
Henn (2012b)	C++, CPLEX 12.0	200	25/-	75
Kulak et al. (2012)	Java 1.5	250	-/509	100
Henn & Schmid (2013)	C++, CPLEX 12.0	80	25/-	75
Chirici & Wang (2014)	NA	60	25/-	75
Azadnia et al. (2013)	MATLAB 7.9.0, Java 1.5, CPLEX 12.4	150	10/200	550
Öncan (2013)	C++	100	10 /-	48
Şahin & Kulak (2013)	Visual Studio, ANOVA	300	-/1049	40
Matusiak et al. (2014)	CPLEX 12.4	1536	50/-	-
Lam et al. (2014)	MATLAB	20	-/100	100
Chen et al. (2015)	Visual Basic	8	10/-	5
Muter & Öncan (2015)	C++, CPLEX 12.5, Concert 2.5	100	10/-	48
Cheng et al. (2015)	MATLAB, CPLEX	200	10/-	5
Koch & Wäscher (2016)	C++	60	25/-	75
Lin et al. (2016)	C++	100	16/-	4 orders
Zhang et al. (2016)	MATLAB	600	5/-	60
Hong & Kim (2017)	C , CPLEX	500	2.02/-	20
Menéndez et al. (2017a)	Java 7	80	25/-	75
Mutingi & Mbohwa (2017)	NA	60	25/-	75
Li et al. (2017)	MATLAB	10000	12/-	100
Scholz & Wäscher (2017)	C++	80	25/-	75
Menéndez et al. (2017b)	Java 7	100	25/-	75
Zhang et al. (2017)	MATLAB	480	5/-	75
Matusiak et al. (2017)	C++	2056	NA	NA
Valle et al. (2017)	C++, CPLEX 12.6	5000	NA	40
Scholz et al. (2017)	C++	200	25/-	75
Xiang et al. (2018)	Java, CPLEX 12.6	100	4/-	NA
Nicolas et al. (2018)	CPLEX	192	NA	NA
Žulj et al. (2018)	Java	100	25/-	75
		600	5/-	15
Van Gils et al. (2018)	ANOVA	26	5/-	40

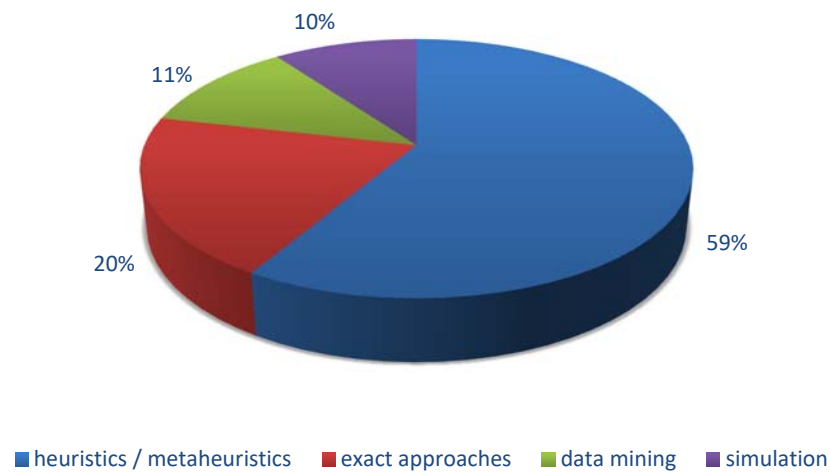


Figure 2.4 Solution methods for order batching related problems

Figure 2.4 displays the percentage values of the solution methods for order batching related problems. Almost one-half of the studies are based on heuristic and metaheuristic solution techniques. In particular, it is seen that solution approaches based on data mining and metaheuristics has gained quite attention in recent years. Therefore, the research on this problem can be said to be focused on finding optimal or near optimal solutions with use of relatively fast methods.

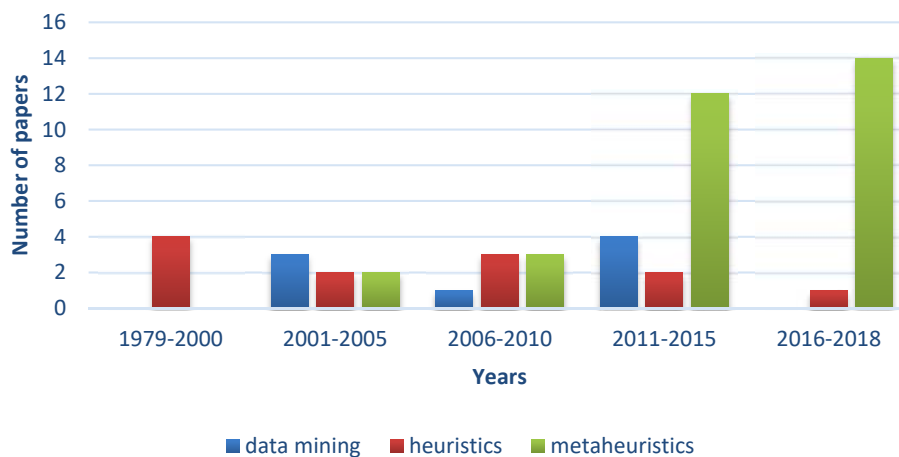


Figure 2.5 Number of papers for order batching related problems

Figure 2.5 gives an idea about number of papers related with order batching. Among the studies in Table 2.2, 51 studies which correspond to 85% investigate data mining, heuristic and metaheuristic methods. In addition, real life applications of this

problem have become more complex and frequently encountered. Therefore, the complexity of the problem requires much more attention and fast solution approaches in today's competitive environment. Accordingly, Figure 2.5 reveals that there is an increasing trend on particularly metaheuristic solution methods for this problem.

2.5 Conclusion

This chapter draws a framework for the order batching problem (OBP) in warehouse order picking process. According to examined studies, a number of conclusions are addressed. First of all, it is thought that future research would concentrate on especially integrated warehouse operations such as combination of the picking, batching and routing stages of customer orders. Because of the complex and combinatorial structure of such an integrated problem, development of effective and efficient metaheuristic approaches is essential.

Another finding is that, storing and batching interrelated items is an important issue for the effectiveness of the batching operation, therefore a number of papers investigate data mining approaches within batching operation. Besides, decisions on determination of storage locations and layout design can also be considered as part of the integrated problem.

In many real word case situations, customers hope to have their orders within a certain period of time. However, customer due dates are only examined in a few papers, and there is a lack of studies concentrating on this type of problems. Since the point in question is the priorities of the customers, due date related objectives are likely to increase in future studies. Additionally, integration of the batching operation with the distribution system would be an effective decision, since there could be a waste of time during the transportation process.

During the order fulfilment process, some characteristics of the orders can change dynamically, such as quantity and type of the ordered items. Consequently, an online (dynamic) batching method will be the suitable approach to apply in these situations.

In addition, there has not been standard data set for the OBP in the literature. Such a test library will be helpful to measure computational efficiencies of the algorithms.

It is predicted that in the future studies on warehouse optimization, most reactive companies to their customers will be the most successful ones in this area. Thereby, every effort that makes delivery process faster and adaptive becomes the most trending topic in the current state of warehouse management.



CHAPTER THREE

PROPOSED METAHEURISTIC APPROACHES FOR THE ORDER BATCHING PROBLEM

3.1 Introduction

In the context of optimization, the number of articles focusing on metaheuristics has increased considerably over the years. The reason for that is the exact solution methods are not particularly useful in solving large scale optimization problems, where a rapid solution is needed. Heuristic methods are often utilized to achieve an effective solution within a short time interval. Problem specific structures of heuristics and some other advanced requirements which appear in development process of algorithms have accelerated the emergence of metaheuristics. Metaheuristics can also be interpreted as problem independent solution approaches that bring heuristic techniques together in order to use key features of them. In light of these definitions, a *metaheuristic* can be expressed as algorithmic solution method that includes notions that guide heuristic techniques for the aim of finding the best solution (Cergibozan, 2013). The advantages of the metaheuristics among other approaches have led the remarkable development of these methods over the last decades. The major advantages of metaheuristics can be stated as follows:

- The ability to solve the problem rapidly
- Utilization of different search techniques within
- Ease of implementation to several types of problems

Since their emergence in the optimization concept, several metaheuristic methods have been developed to solve optimization problems. In general, the method that will be used to solve a problem depends on the distinctive features of the considered problem. It can be conveniently said that the same approach is valid when choosing a metaheuristic technique to apply. Burke & Kendall (2005) present several search techniques that can be used in optimization. Doerner et al. (2007) address several metaheuristic methods developed for complex optimization problems. Talbi (2013)

focuses on the hybrid structures for metaheuristics, and a detailed analysis is presented. Classification schemes on hybrid metaheuristics can be found in the study of Raidl, Puchinger, & Blum (2010). There is still a wide research area to develop metaheuristic based solution approach to optimization problems. Particularly, it is worth to say that metaheuristics are useful for finding effective solutions in a limited time. However, if we have the opportunity to tolerate the time somewhat, it is better to use a hybrid technique that contains an exact solution approach in some level.

In this chapter, three metaheuristic techniques for the OBP are introduced: a genetic algorithm approach, a hybrid technique based on population interaction, and a local search based approach. All of these algorithms have their own specifications, e.g. the hybridization structure has the advantage of maintaining the information exchange via the swarm intelligence technique, a number of neighborhood moves will be utilized in the local search algorithm. After the computational study, performance of the algorithms at solving the OBP will be evaluated.

3.2 Metaheuristics for the Order Batching Problem

Exact solution methods are not applicable when the large size orders come into question and herein some metaheuristic solution approaches are used to find near optimal solutions. In this study, we propose three metaheuristic approaches for the OBP: a genetic algorithm that utilizes from a local search, a hybrid algorithm based on interaction of different populations, and a local search algorithm. In the following, the main features and structures of these algorithms are explained in detail.

3.2.1 Genetic Algorithm

Genetic algorithms are a type of evolutionary algorithms and reflect the process of evolution for solving optimization problems (Holland, 1975; Gen & Cheng, 2000). The genetic process is based on the principle of changing a population along iterations. The process starts with a preliminary population of candidate solutions, improves the solutions, and results with the best solution in the population. Figure

3.1 demonstrates the general algorithmic flow of a genetic algorithm. In this population-based method, individual that yields the best objective function value for the corresponding problem is selected as the best solution (Sastry, Goldberg, & Kendall, 2005).

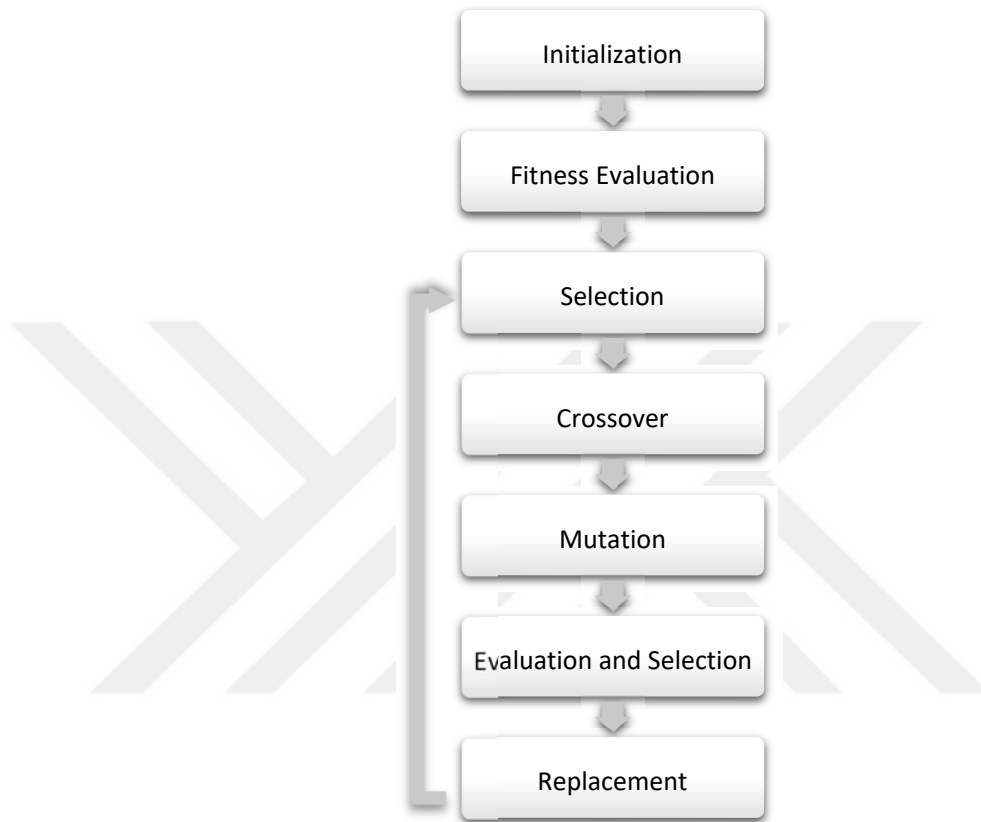


Figure 3.1 Genetic algorithm flowchart

In Figure 3.2, pseudo code of a genetic algorithm is displayed. Here, t_{ga} means iteration number for the genetic algorithm. Accordingly, $P(t_{ga})$ and $C(t_{ga})$ corresponds to the population and child chromosome in t_{ga} iteration, respectively. After determining parameters, data and related encoding scheme, algorithm is initialized with a population. The individuals which can be called as chromosomes in the population are evaluated in accordance with their fitness values for the related problem. From this point, the iterations of the genetic algorithm start. Selection, recombination and mutation operations are applied sequentially. The resulting child chromosome and the parent chromosomes on hand are evaluated together, and the selection operation is implemented to form the new population. These operations are continued until the termination condition is reached.

In this study, a genetic algorithm that includes a local search component is developed to solve the OBP. The details of the proposed algorithm can be briefly represented as follows. Firstly, the representation is made according to order number, this can be thought as a permutation representation. The initialization is implemented by use of several chromosomes as the starting population. The fitness values of the chromosomes are evaluated taking the characteristics of the OBP into consideration. In the selection stage, roulette wheel method is used. After that, chromosomes are recombined according to the order crossover technique. The swap operation is applied to the chromosomes in the mutation stage, and the resulting chromosome is evaluated. The elitist selection scheme is employed, and after mutation, two opt local search heuristic is implemented according to a rate. Reaching the termination condition is enough for proposed algorithm to be terminated.

Algorithm 1: Genetic algorithm

```

input: problem data, related parameters
output: the best solution in the population
begin
   $t_{ga} \leftarrow 0$ ; //  $t_{ga}$ : generation number
  initialize  $P(t_{ga})$  by encoding procedure; //  $P(t_{ga})$ : population
  evaluate  $P(t_{ga})$  by decoding procedure and keep the best solution;
  while (not terminating condition) do
    select parents from  $P(t_{ga})$  by selection procedure;
    produce  $C(t_{ga})$  from  $P(t_{ga})$  by crossover procedure; //  $C(t_{ga})$ : offspring
    produce  $C(t_{ga})$  from  $P(t_{ga})$  by mutation procedure;
    evaluate  $C(t_{ga})$  by decoding procedure & update the best solution;
    select  $P(t_{ga}+1)$  from  $P(t_{ga})$  and  $C(t_{ga})$  by selection procedure;
     $t_{ga} \leftarrow t_{ga} + 1$ ;
  end
  output: the best solution in the population
end;

```

Figure 3.2 Genetic algorithm pseudo code

In the following subsections, details of the operators and procedures in the algorithm are explained.

3.2.1.1 Solution Representation

Encoding type of the individual is determined according to the structure of the problem. In OBP, solutions are represented as a permutation vector of the orders, and individuals correspond to chromosomes. See the related example shown in Figure

3.3. The picking sequence of the orders starts with Order No 1, and ends with Order No 3.

1	5	9	2	6	4	10	8	7	3
---	---	---	---	---	---	----	---	---	---

Figure 3.3 Chromosome representation based on order number

3.2.1.2 Initialization

In the algorithm, initial population contains random generated individuals, and population size (p) differs according to the data sets used. Besides, population size is arranged to provide a balance between investigating the solution area deeply and reducing computational time.

3.2.1.3 Fitness Value Calculation

Solution qualities of the members in the population are calculated in this stage. Considering that OBP is a minimization problem, fitness of a member is calculated in accordance with the characteristics of this type of problems. In the algorithm, a proportional fitness value is used (see Eq. (3.1)). $fitness(i)$ and td_i correspond to fitness value and total travel distance of i th solution, respectively.

$$fitness(i) = 1 / (td_i / \sum_{i=1}^p td_i) \quad (3.1)$$

3.2.1.4 Selection

Candidates for crossover operation are determined in the selection stage. There are several strategies that can be used for this aim (e.g. tournament selection, random selection, linear ranking). Roulette wheel selection (RWS), which is one of these strategies, is determined as the selection procedure in the algorithm. In RWS, the degree of fitness to the objective demonstrates the likelihood of selection. Eq. (3.2) corresponds to probability of selection. Parents, two individuals with the highest values of $ps(i)$, are chosen.

$$ps(i) = fitness(i) / \sum_{i=1}^p fitness(i) \quad (3.2)$$

3.2.1.5 Recombination

In recombination stage, which is also named as crossover, parents are recombined according to a strategy. Different crossover methods (e.g. m-point, partial mapped, cycle, position-based) can be used for permutation representation (Gen & Cheng, 2000).

Procedure 1: Order crossover	
1. Determine a random segment from one of the parents	
2. Copy this segment to corresponding situation in the child chromosome	
3. Remove nodes that take place in this segment from the second parent	
4. Put the remaining and unassigned nodes to the empty spaces in the child chromosome sequentially	

Figure 3.4 The order crossover procedure

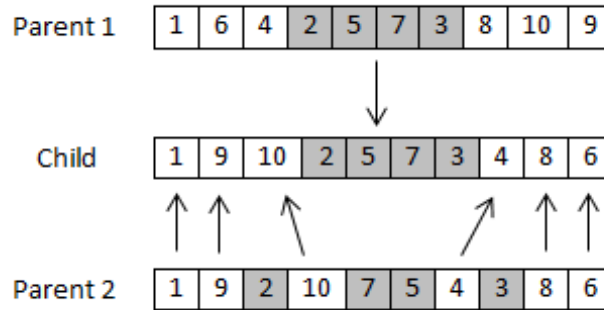


Figure 3.5 Order crossover operation with refer to Gen & Cheng (2000)

The order crossover method explained in Figure 3.4 is powerful for creating diverse chromosomes (see also Figure 3.5). Therefore, this procedure is applied to the parents chosen from the population.

3.2.1.6 Mutation

After implementing the crossover operation to the parents, mutation operation is applied to the child in accordance to a ratio. As the mutation procedure, swap operation that is also known as the 2-exchange operation is employed in the algorithm; and positions of two selected nodes are replaced with each other as the name suggests.

3.2.1.7 Evaluation and Replacement

In this stage, solution qualities of parents and child are evaluated. There are various replacement strategies such as adding child to the population without evaluation, or choosing best two of three candidates, and adding them to the population. Latter, which is also applied in this study, is known as elitist strategy.

After mutation and evaluation stages, we use a local search method to improve solution of the child if it is selected to be added to the population. Thus, a “local best solution” will be investigated through this heuristic. In two-opt local search technique, two selected points are replaced with each other and construct a new route, such that the order between these two points is reversed (Johnson & McGeoch, 1997; Zäpfel, Braune, & Bögl, 2010). As it generates neighboring but somewhat different chromosomes around the solution of the child, this local search technique is chosen as an improvement procedure and performed iteratively. Figure 3.6 display the 2-opt procedure applied in the algorithm.

Procedure 2: 2-opt heuristic
1. Determine two random nodes in the child
2. Reverse the nodes between these two nodes
3. Create the new chromosome with other points in place

Figure 3.6 Procedure of the 2-opt heuristic

The termination condition of the proposed GA is determined as a fixed period of time. Therefore, the algorithm is terminated after a certain number of iterations.

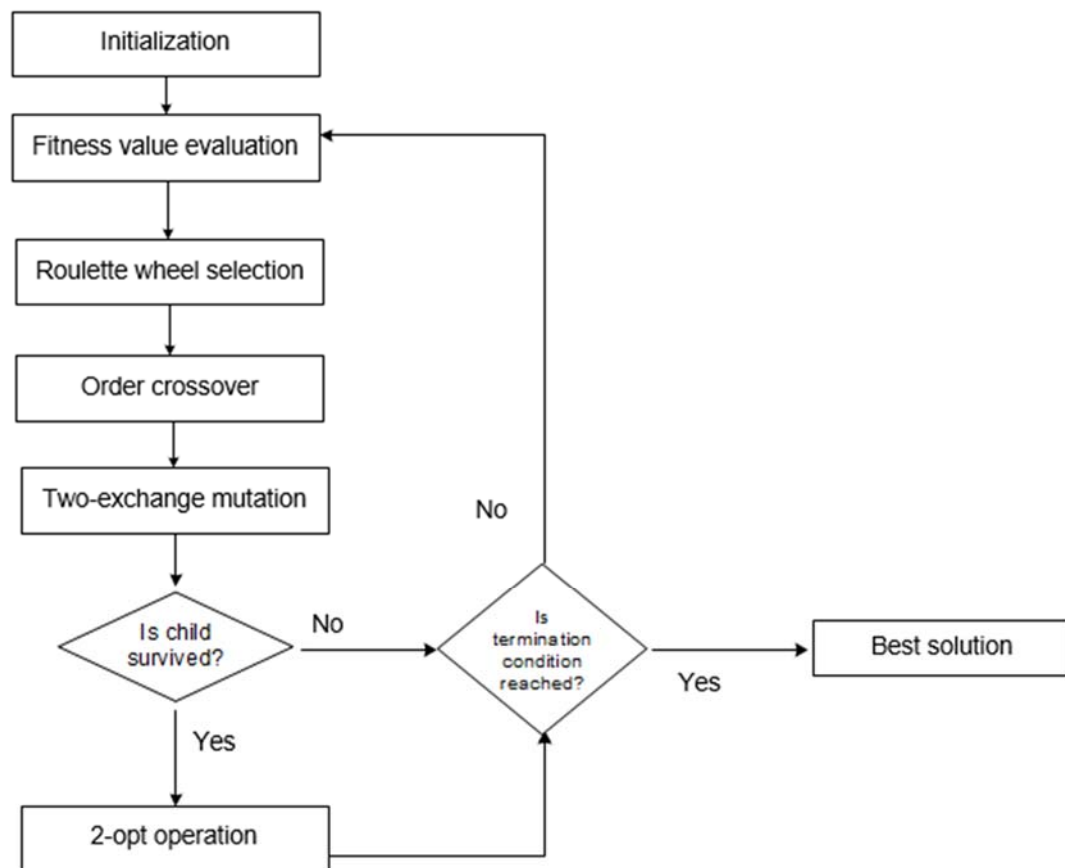


Figure 3.7 Flowchart of the proposed genetic algorithm

In Figure 3.7, the proposed genetic algorithm is displayed as a flowchart. The proposed approach has the advantage of the local search operation inside. Application of the powerful 2-opt heuristic to the child chromosome improves the solution of the child, that lead a constant improvement on the fitness values across the population indirectly.

3.2.2 Hybrid Algorithm with Population Interaction

The second algorithm for the OBP is an interaction based hybrid genetic algorithm. Proposed algorithm includes different search schemes within. Unlike the previous genetic algorithm, this algorithm uses interaction of several populations throughout the iterations.

3.2.2.1 Interaction Based on Swarm Optimization Technique

Studies on swarm behavior in the field of optimization are quite common. The intelligence behind these techniques is that individuals in the swarm are in constant communication, and are directed towards the desired target. In particle swarm optimization (PSO), members of the swarm make an information exchange between each other (Kennedy & Eberhart, 1995; Merkle & Middendorf, 2005). Here, members also called as particles are accepted as different solutions. Each member knows the global best solution in the swarm up to that time, and searches the solution area according to the supplementary information on hand. In addition to the global best solution, the member evaluates its own objective function value on that time, own best solution, and related velocity. All particles in the swarm try to improve the best objective function value continuously (Coello, Dehuri, & Ghosh, 2009).

Suppose that $v_k(t)$ is the velocity of the particle k at iteration t , $x_k(t)$ is position of the particle k at iteration t . $pbest_k(t)$ and $gbest(t)$ are the best solutions of the particle k and the swarm up to iteration t , respectively. w is inertia factor, b_1 and b_2 are acceleration coefficients, r_1 and r_2 are uniform random numbers between $[0,1]$. The new velocity and position of the particle are determined by Eq. (3.3) and Eq. (3.4), respectively.

$$v_k(t + 1) = w \cdot v_k(t) + b_1 \cdot r_1 \cdot (pbest_k(t) - x_k(t)) + b_2 \cdot r_2 \cdot (gbest(t) - x_k(t)) \quad (3.3)$$

$$x_k(t + 1) = x_k(t) + v_k(t + 1) \quad (3.4)$$

PSO is originally developed for continuous optimization problems and when the problem is discrete, the equations above are inappropriate for application. In the literature, several studies examine PSO for discrete optimization problems, and some of them will be mentioned here. Hu, Eberhart, & Shi (2003) studied PSO for the n -queens problem. In their study, velocities are defined in terms of probability of changing two elements of the permutation vector, and two-exchange (swap) move is applied as mutation operation. Wang, Huang, Zhou, & Pang (2003) used swap operator and swap sequence as the velocity update procedure for solving TSP.

Machado & Lopes (2005) developed a hybrid algorithm based on PSO, GA and a local search for TSP. Goldberg, Goldberg, & De Souza (2008) proposed PSO algorithms for TSP, and made a comparison between their results and some of other algorithms in the literature. The main idea of their research is based on again distinct velocity operators. Authors use inversion neighborhood, which is the basic 2-opt move, and Lin-Kernighan neighborhood as two local search methods for their algorithms. In these studies, it is seen that solution methods vary across the studies. As the velocity update procedure, some of these methods use transformation of the Eq. (3.3) to a binary case, some of them use crossover operation (Settles & Soule, 2005; Pant, Thangaraj, & Abraham, 2007; Hao, Wang, & Huang, 2007), or a swap operation is applied in some studies. In this study, velocity update procedure is determined as the order crossover operation, which generates quite good new chromosomes. According to this communication procedure, particles exchange information with each other iteratively.

Algorithm 2: Particle swarm optimization algorithm input: $f(\mathbf{x})$, $\mathbf{v}_k(0)$, $\mathbf{pbest}_k(0)$, $\mathbf{gbest}(0)$, b_1, b_2, r_1, r_2 output: the best solution begin $t \leftarrow 0$; initialize $\mathbf{x}_k(t)$ by encoding procedure; evaluate $\mathbf{x}_k(t)$ by decoding procedure and keep the best solution; while (not terminating condition) do for each particle $\mathbf{x}_k$ in swarm do $\mathbf{v}_k(t+1) = w(t)\mathbf{v}_k(t) + b_1 r_1 (\mathbf{pbest}_k(t) - \mathbf{x}_k(t)) + b_2 r_2 (\mathbf{gbest}(t) - \mathbf{x}_k(t))$; $\mathbf{x}_k(t+1) = \mathbf{x}_k(t) + \mathbf{v}_k(t+1)$; evaluate $\mathbf{x}_k(t+1)$ by decoding procedure; if $f(\mathbf{x}_k(t+1)) < f(\mathbf{pbest}_k(t))$; update $\mathbf{pbest}_k(t) = \mathbf{x}_k(t+1)$; end; $\mathbf{gbest}(t+1) = \text{argmin}\{f(\mathbf{pbest}_k(t)), f(\mathbf{gbest}(t))\}$ & update the best solution; $t \leftarrow t+1$; end; end; output: the best solution $\mathbf{gbest}$; end;

Figure 3.8 Pseudo code of PSO

In Figure 3.8 the pseudo code of the PSO algorithm can be seen. The algorithm attempts to reach the best solution among the swarm by constantly evaluating the personal best and global best solutions. After the initial data and the parameters are set, the algorithm is started with a swarm. The speed and position of each particle k in the swarm is determined. This value is updated if the current solution value of the particle, $x_k(t)$, is better than its personal best solution, $pbest_k(t)$. The global best solution, $gbest$, is selected from the best solutions of all individuals, and is constantly updated. Once a specific condition is reached, the algorithm is terminated.

In the following section, developed interaction based hybrid algorithm is explained in detail. Note that, the hybrid scheme is based on the swarm optimization technique, which is considered as PSO in proposed algorithm. The reason of selection PSO is that after genetic operations, a population that contains both good solutions from the genetic algorithm and randomly generated new individuals was considered to be useful in terms of exploration of diversity. Therefore, it is aimed with the hybrid algorithm to perform both diversification and intensification processes which are two major subjects in metaheuristics.

3.2.2.2 Hybrid Scheme

Proposed hybrid algorithm is expressed in Figure 3.9 in detail. Algorithm starts with the GA, and implements GA operations iteratively. In the evaluation stage, if child is chosen to be added to the population, 2-opt heuristic is performed according to the 2-opt rate. After GA operations, PSO procedure starts according to a rate.

The swarm contains random generated chromosomes and a number of best solutions in the GA population so far. Position of a particle is accepted as permutation representation of the chromosome. Velocity and position update procedures are implemented via the order crossover operation as mentioned earlier. In this operation, we use two crossover parameters, $p1$ (crossover rate with particle best), and $p2$ (crossover rate with global best). The crossover operation is applied to the particle according to the rates of these parameters.

Algorithm 3: Hybrid algorithm
input: problem data, PSO and GA parameters output: the best solution in GA population begin $t_{ga} \leftarrow 0$; initialize $\mathbf{x}_k(t_{ga})$ by encoding procedure; //population $P(t_{ga})=[\mathbf{x}_k(t_{ga})]$ evaluate $\mathbf{x}_k(t_{ga})$ by decoding procedure and keep the best solution; repeat produce $C(t_{ga})$ from $P(t_{ga})$ by order crossover procedure; produce $C(t_{ga})$ from $P(t_{ga})$ by two exchange mutation procedure; evaluate $C(t_{ga})$ by decoding procedure; if child is survived local search heuristic by 2-opt procedure; end; update the best solution; select $P(t_{ga}+1)$ from $P(t_{ga})$ and $C(t_{ga})$ by selection procedure; if PSO condition satisfied begin $t_{ps0} \leftarrow 0$; initialize PSO swarm while (not terminating condition) do for each particle $\mathbf{x}_k$ in swarm do update position $\mathbf{x}_k(t_{ps0}+1)$ by order crossover procedure; if $f(\mathbf{x}_k(t_{ps0}+1)) < f(pbest_k(t_{ps0}))$ then update $pbest_k(t_{ps0}) = \mathbf{x}_k(t+1)$; end; $gbest(t_{ps0}+1) = \operatorname{argmin} \{f(pbest_k(t_{ps0}), gbest(t_{ps0}))\}$ and update; $t_{ps0} \leftarrow t_{ps0} + 1$; end; end; output: the best solution $gbest$; end; add $gbest$ to $P(t_{ga}+1)$ by removing the worst solution from $P(t_{ga}+1)$ end; $t_{ga} \leftarrow t_{ga}+1$; until (the GA terminating condition met) output: the best solution in GA population; end;

Figure 3.9 Pseudo code of proposed hybrid algorithm

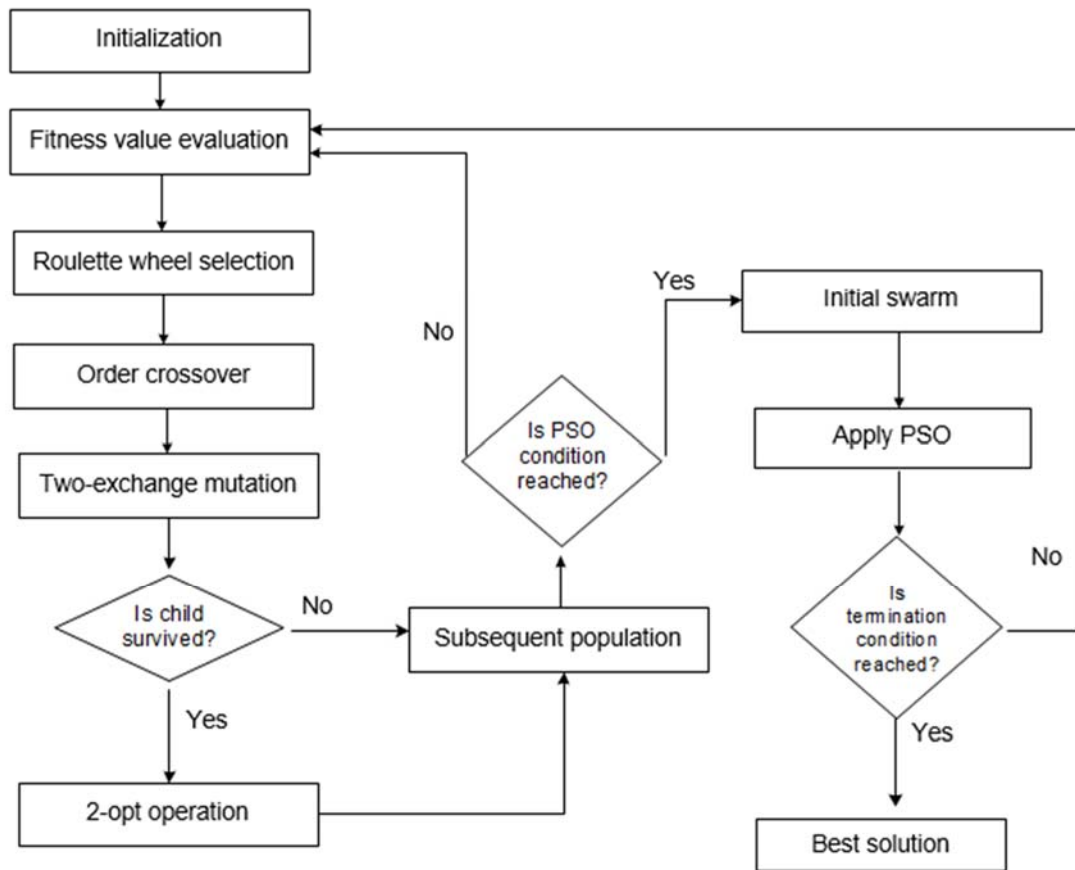


Figure 3.10 Flowchart of the proposed hybrid algorithm

Figure 3.10 also displays the flowchart of the proposed hybrid algorithm. After implementation of the GA operations sequentially, PSO algorithm starts the communication scheme between the chromosomes. The benefit of this scheme is that it evaluates neighborhoods of the chromosomes, and updates the global best solution by considering personal best solutions of them in each iteration.

3.2.3 Local Search Based Algorithm

As it is stated previously, the exact solution methods seem to be impractical when problem size increases; therefore, heuristic and metaheuristic solution approaches are used to find effective solutions in a reasonable time. In this section, the OBP is investigated with a local search based algorithm. Details of the local search algorithm are described in the next subsections.

3.2.3.1 Flowchart of the Algorithm

The flowchart of the proposed algorithm is displayed in Figure 3.11. Algorithm starts with a number of initial solutions. Several local search techniques are utilized in the algorithm, and these heuristics are implemented in order.

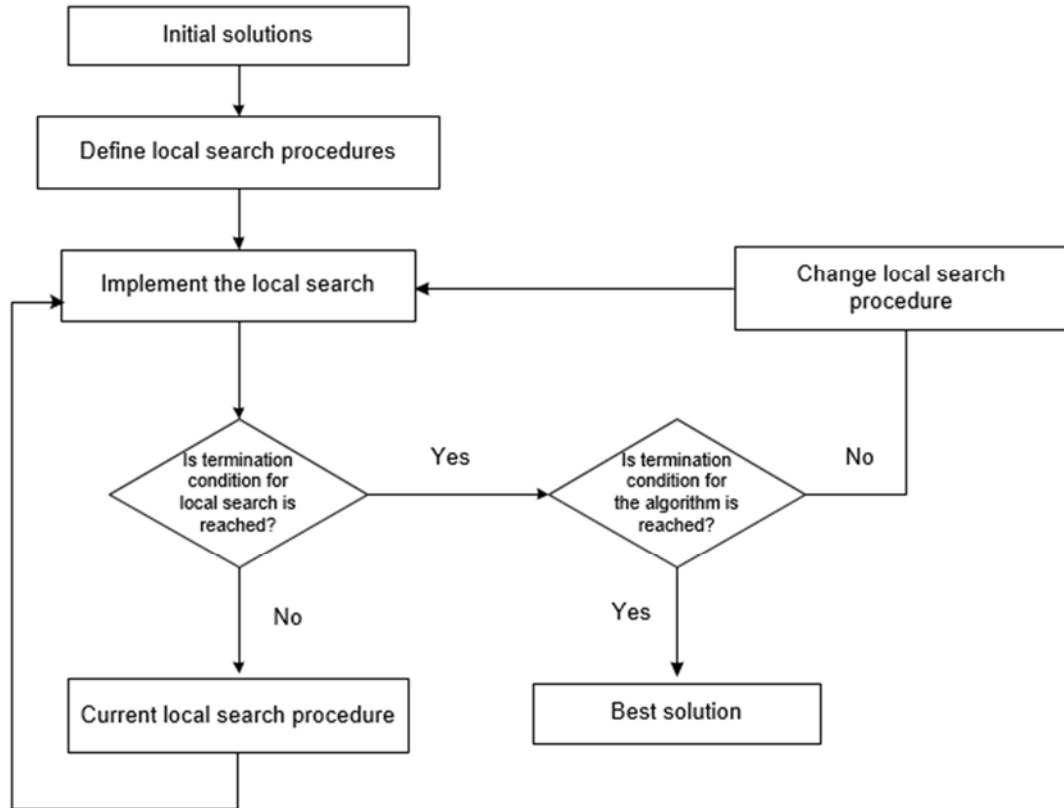


Figure 3.11 Flowchart of the proposed local search algorithm

The current heuristic procedure is used until the finishing state of the heuristic is reached. After that, the next heuristic is implemented. When the termination condition is reached, the algorithm yields the best solution so far.

3.2.3.2 Local Search Heuristics

The algorithm consists of a number of local search heuristics. First of all, customer orders are evaluated as solution components at the beginning of the

algorithm. Algorithm is initialized with a number of solutions that indicate random permutations of the orders. The result of a solution is identified as l_i is total length of the i th chromosome, which corresponds to total distance travelled while picking all of the batches. In the following, the local search techniques that are used in this study are explained in detail.

3.2.3.2.1 Swap Heuristic. In this heuristic, two nodes from the solution element are chosen, and exchanged with each other (see Figure 3.12).

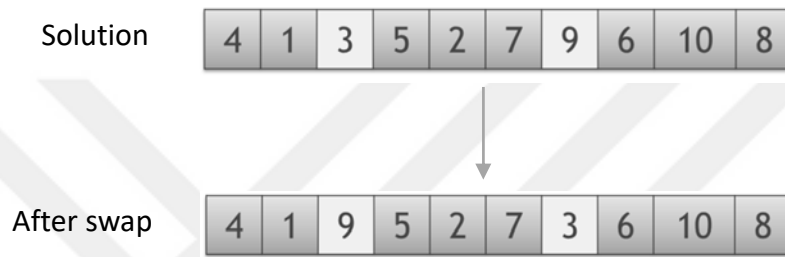


Figure 3.12 Swap heuristic

3.2.3.2.2 Move Forward Heuristic. One node is chosen from the solution randomly, and this node is assigned to a further point from its location in the solution. Figure 3.13 represents the move forward heuristic.

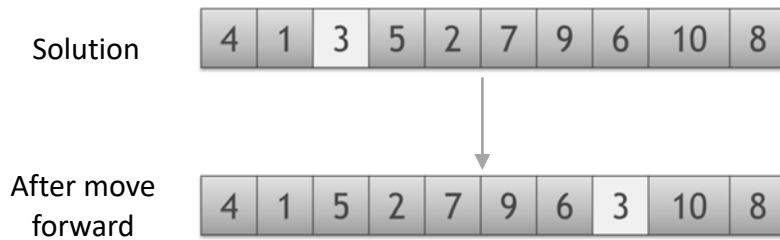


Figure 3.13 Move forward heuristic

3.2.3.2.3 Move Backward Heuristic. This heuristic is another example of move heuristics. In this heuristic, one node is chosen from the solution randomly, and this node is assigned to a previous point from its location in the solution (see Figure 3.14).

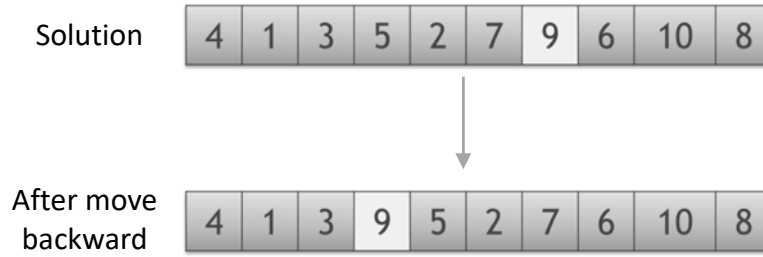


Figure 3.14 Move backward heuristic

3.2.3.2.4 2-opt Heuristic. As explained previously, in 2-opt local search, two points are chosen and replaced with each other; while the part between these two points are reversed in the new permutation. The modification in the solution element via the 2-opt move is displayed in Figure 3.15.



Figure 3.15 2-opt heuristic

3.3 Conclusion

In this chapter, three metaheuristic approaches are developed and tested to solve the OBP. Genetic algorithm approach uses a local search inside, and searches the solution area iteratively. Secondly, a hybrid approach that contains genetic algorithm and PSO based interaction scheme is presented with its specific features. Finally, a local search based algorithm is developed for the OBP. A detailed comparison on the performances of these approaches is made in the next chapter.

Each of the suggested methods may have a distinct advantage and disadvantage, but it is believed that they all contribute to the literature. It is seen that the problem can be modeled with these methods, and all of these solution methods can be used to solve the problem.

As it is stated in the previous chapter, solution methods from the literature do not seem to be comparable in real terms. From the studies it is seen that authors compared their algorithms with other algorithms by making similar assumptions and using similar warehouse layouts.

In the next chapter, computational experiments on problem sets with different sizes and characteristics are implemented. In addition, the benchmark study of proposed algorithms is conducted.



CHAPTER FOUR

COMPUTATIONAL STUDY FOR THE PROPOSED APPROACHES

4.1 Introduction

The main purpose of this study is identified as developing fast and effective metaheuristic approaches that can solve the OBP in different warehouse configurations. For this purpose, three metaheuristic algorithms are developed in Chapter 3. In this chapter, the algorithms are applied to the illustrative case of the OBP and also tested on benchmark instances of the OBP. Firstly, the illustrative case is presented as well as the generated problem sets. Secondly, the benchmark instances are described. The instance set is chosen among the most cited studies in the related field. Then, corresponding warehouse layout is presented with its characteristics. Warehouse layout considered in this study is taken from a study which concentrates on metaheuristics for the problem. Experimental design study is implemented to the parameters, and a computational study is performed both for the problem sets and the instances taken from the corresponding study. The performances of the approaches are evaluated after obtaining the results. According to the computations, the algorithms find effective solutions to the OBP in a reasonable time; and performances of developed algorithms seem competitive among the other approaches.

4.2 Illustrative Case

This section presents the illustrative case of the OBP. In the following sections, the warehouse layout, routing strategies and related information about the warehouse picking operation are described. Parameter design and experimental studies related with the proposed algorithms are implemented in order to test performances of the algorithms.

4.2.1 Structure of the Warehouse

The warehouse layout used in the experiments can be seen in Figure 4.1. The warehouse is a single-block warehouse with three parallel picking aisles and two cross aisles. The depot is the origin of the warehouse and can be seen in the bottom left corner of the layout. Distances between storage locations are computed according to their (x,y) coordinates with the consideration of only vertical and horizontal movements of the picker and the picking device. The picking operation is implemented by walking the distance between two storage locations which are at the same y coordinate. The layout corresponds to layout of a low-level picker to parts system where the picker moves two directions to pick up the items. Other assumptions used in the experiments are given below:

- Picker to part warehouse system is considered,
- Only one order picker retrieves orders from their storage locations,
- Storage locations are of identical size,
- Only one storage location is assigned for each item type,
- Orders are known and indivisible, and do not change during the picking process (static batching),
- Capacity of the picking facility is higher than the volume of each customer order.

It is important to point out that, for each individual, assignment of the orders to batches is made considering the capacity of the picking facility. For example, if total volume of the orders until that moment is lower than or equal to the total capacity of the picking facility, and then the assignment of the order is implemented; otherwise the corresponding order is added to a new batch.

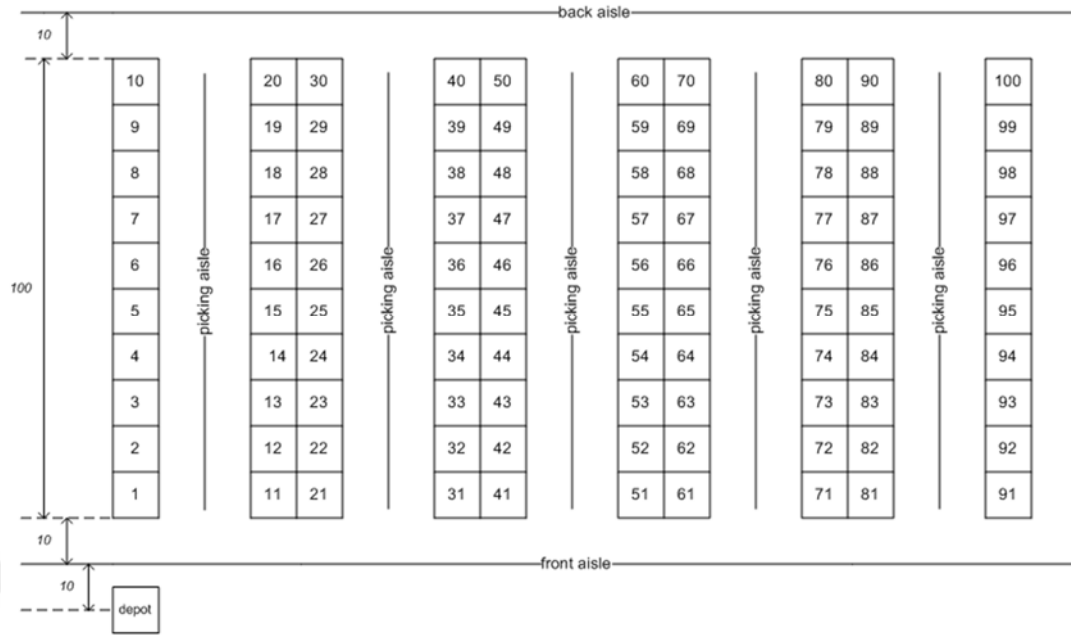


Figure 4.1 The layout of the warehouse

All warehouse parameters are displayed in Table 4.1. These parameters are number of picking aisles and total storage locations, length and width values of the storage units and aisles, and remaining distance information related with the depot and aisles.

Table 4.1 Warehouse parameters

Parameter	Value
number of picking aisles	5
number of total storage locations	100
length of a storage unit	10
aisle width	20
distance between depot and the front cross aisle	10
vertical distance between cross aisles and first storage locations	10

Problem classes used in the experiments are presented in Table 4.2 with the corresponding parameter values. Three different problem sets as P1, P2, and P3 are generated to evaluate the performance of the algorithms. The number of orders in

these problem sets ranges from 10 to 100, and an average of 10 items is assigned to each order. The number of items per order is uniformly distributed within the interval [5, 15]. Accordingly, the total volume of the orders and corresponding information on minimum number of batches are given in Table 4.2. The capacity of the picking vehicle is fixed at 20 units of items for each problem set.

Table 4.2 Problem classes with related parameter values

Parameter	P1	P2	P3
number of orders	10	50	100
average number of items	10	10	10
total volume of orders	100	500	1000
capacity of the picking vehicle	20	20	20
minimum number of batches	5	25	50

The order information of P1, P2 and P3 problem sets can be found in Appendix-1, Appendix-2, and Appendix-3, respectively. The information is displayed as a 0-1 matrix showing that the value corresponding to an item in an order row is 1 if the item is included in that order, and 0 otherwise.

4.2.2 Routing Strategies

For an order picking operation, determining the route of the order picker who is responsible for picking the items from different locations is an important problem to concentrate. From Figure 4.2 to Figure 4.8, examples of routing strategies for a warehouse with single block are illustrated by use of the *Interactive Warehouse* which can be found in Roodbergen (2018). The methods are S-shape, return, mid-point, largest gap, combined, and optimal strategies. These routing methods are examples for single-block warehouses, and they can also be used for multiple-block warehouses (Roodbergen & de Koster, 2001; Petersen, Aase, & Heiser, 2004; de Koster, Le-Duc, & Roodbergen, 2007). In the following, brief definitions on these routing strategies are given.

Constituting a picking route is as important as batching. In practice, the order picker tends to follow the route that looks the easiest. Therefore, if the necessary information is not given about the importance of the route, he/she will follow the easiest route, based on own experiences.

In the *S-shape (traversal)* heuristic, order picker enters the respective aisle and passes it entirely if an item in the order list will be picked up from that aisle. If the last item that will be picked is in the last aisle, then the picker will return to the front aisle.

In the *return* strategy, the picker returns and exits from that aisle as soon as he/she picks the item. This heuristic can be expressed as one of the simplest strategies to be understood and be applied.

The *mid-point* heuristic ensures that the picker enters until the middle of the aisle to pick up the articles, and returns from the same aisle, except for the first and the last aisles.

Largest gap strategy limits the picker not to travel long distances among an aisle. In this heuristic, three distances are considered within the aisle: the distance from the front aisle to the nearest ordered item, the distance from the back aisle to the nearest ordered item, and the largest distance between two adjacent ordered items. The picker evaluates these distances, and chooses not to travel the largest distance (gap) among them.

Combined strategy is a mix of traversal and return heuristics. After picking all items of an aisle, the picker decides whether to pass the next aisle from the back aisle, or to follow a return route.

Optimal heuristic by Ratliff & Rosenthal (1983) gives a least cost solution for a picking route; however, it can be a complicated route to follow for the picker.

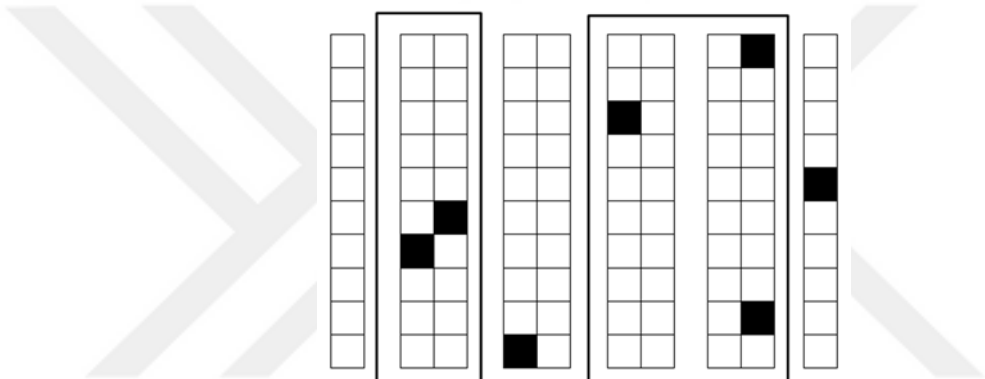


Figure 4.3 S-shape routing



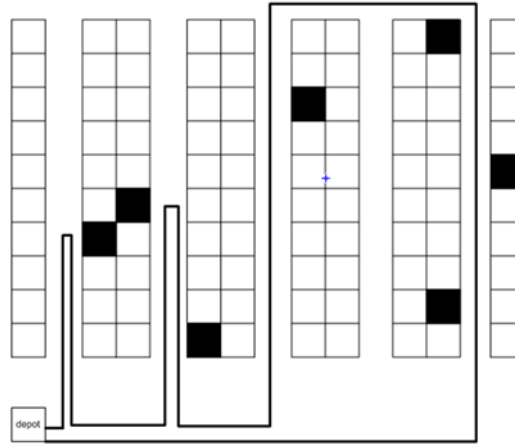


Figure 4.8 Optimal routing

In this chapter, the order batching problem is examined with the proposed approaches, and the experimental study is performed with three routing strategies: S-shape, mid-point and return. The reason of this selection is that, order pickers may resist and may not follow the route if they think that the given route is complicated or not feasible. Consequently, the most understandable and easy to implement routing strategies are chosen. These three strategies stand out in this respect (Le-Duc, & Roodbergen, 2007; Franzke, Grosse, Glock, & Elbert, 2017).

4.2.3 Experimental Design for Genetic Algorithm and Hybrid Algorithm

The first study is implemented with the proposed GA approach. Table 4.3 displays the parameter levels for the proposed algorithms. Since inappropriate tuning of parameters may result with a local optimum, parameters are constructed according to the sizes of the data sets in the computational study.

As the problem size grows, computation time of the algorithm clearly increases; consequently, a parameter study is implemented, and final parameter settings for GA are presented in Table 4.4.

Table 4.3 Parameter levels for the genetic algorithm

Parameter	Level
population size (p)	[50, 200]
crossover ratio	[0.7, 0.75, 0.8]
genetic replications	[100, 1500]
mutation ratio	[0.1, 0.2]
mutation replications	[20, 50]
local search ratio	[0.5, 0.75]

Table 4.4 Parameter set for the genetic algorithm

Parameter	P1	P2	P3
population size (p)	50	100	200
crossover ratio	0.8	0.8	0.8
genetic replications	500	1500	1500
mutation ratio	0.1	0.1	0.1
mutation replications	20	20	20
local search ratio	0.75	0.75	0.75

In the parameter study; all parameters except one are fixed at their initial values first, and then the best parameter value of the one is determined by experiments. The setting is finished when all parameter values are fixed (Ropke & Pisinger, 2006). With these parameter levels, it is aimed to find a trade-off point that provides effective solutions in a reasonable time.

The number of individuals ranges from 50 to 200. The settings on crossover and mutation rates are prevalent and frequently encountered in the literature. The number of runs is determined as 500 and 1500 for the GA according to the convergence of best solutions in the experimental study.

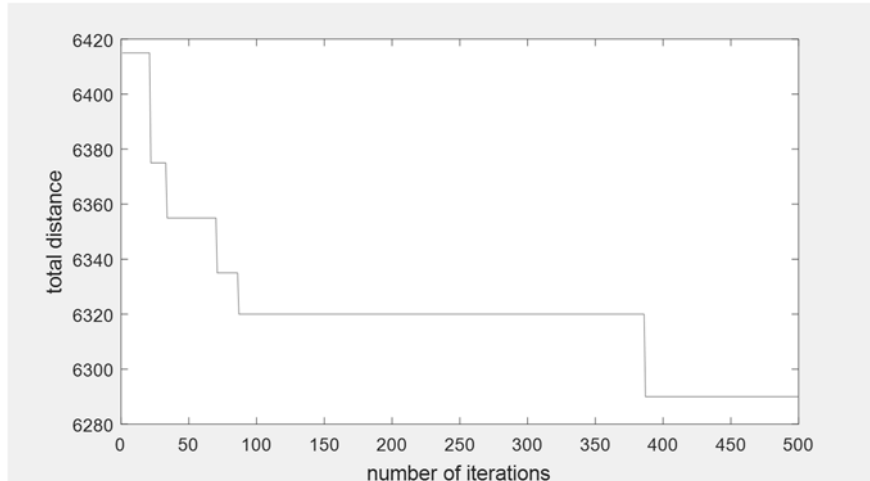


Figure 4.9 Best objective function values along GA iterations for P1

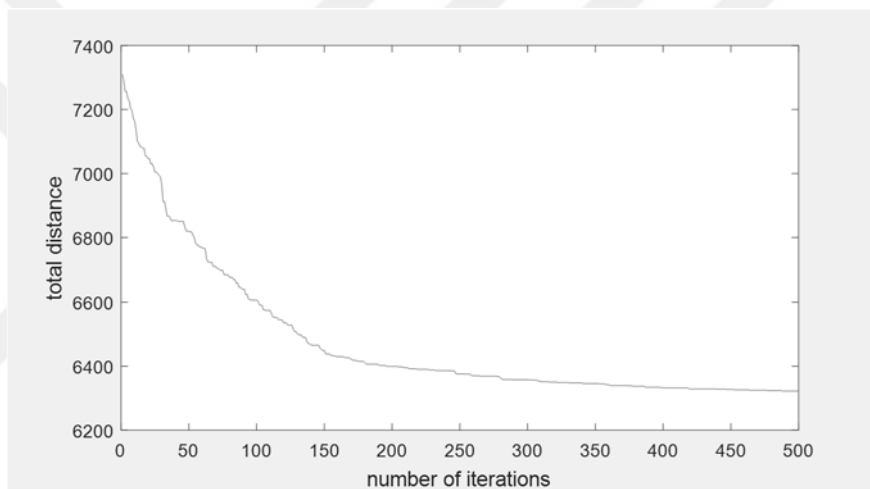


Figure 4.10 Average objective function values along GA iterations for P1

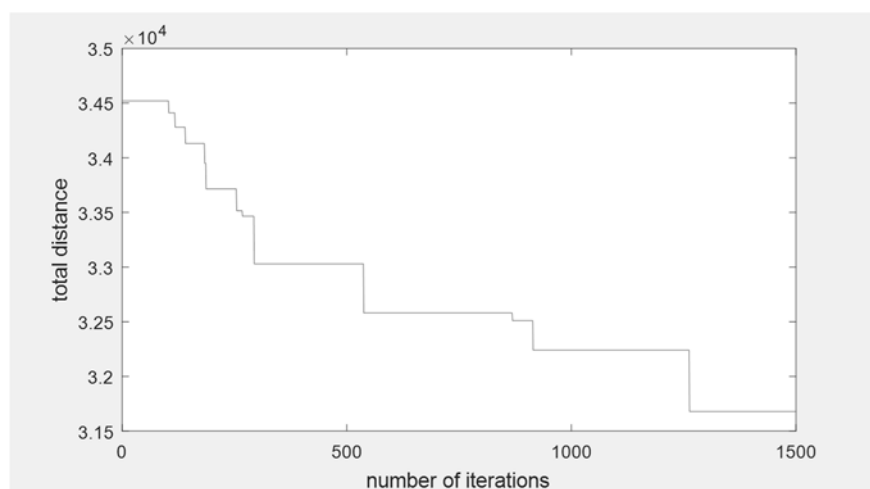


Figure 4.11 Best objective function values along GA iterations for P2

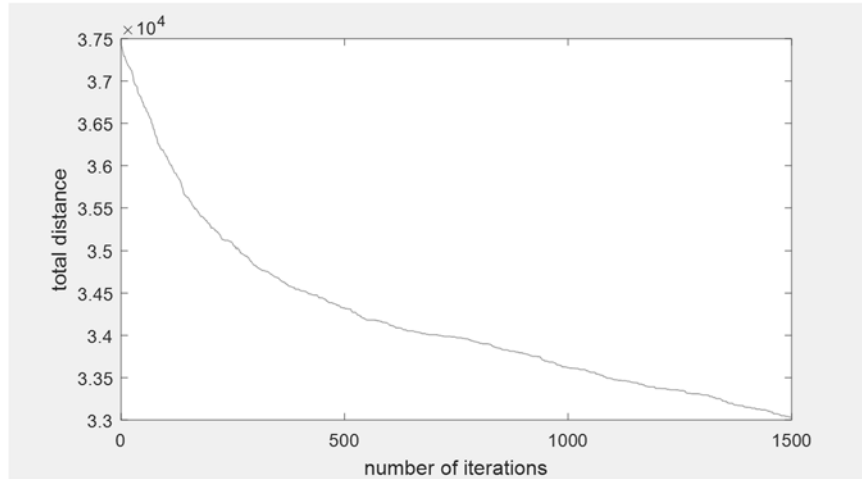


Figure 4.12 Average objective function values along GA iterations for P2

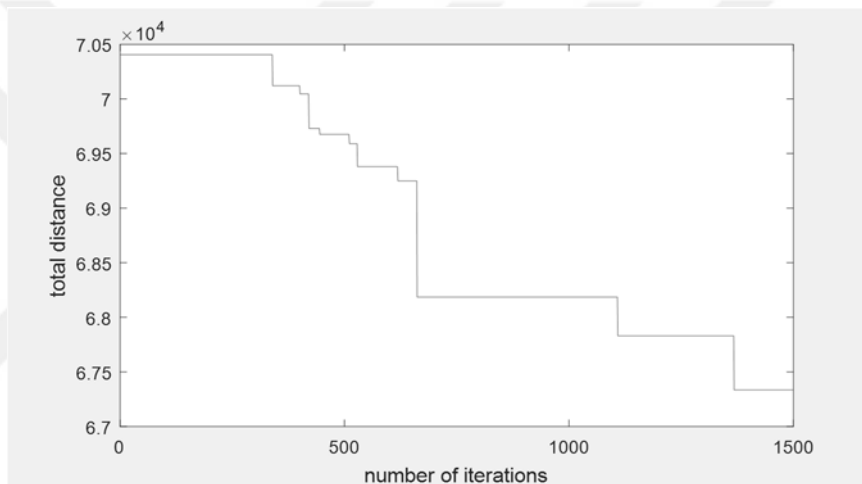


Figure 4.13 Best objective function values along GA iterations for P3

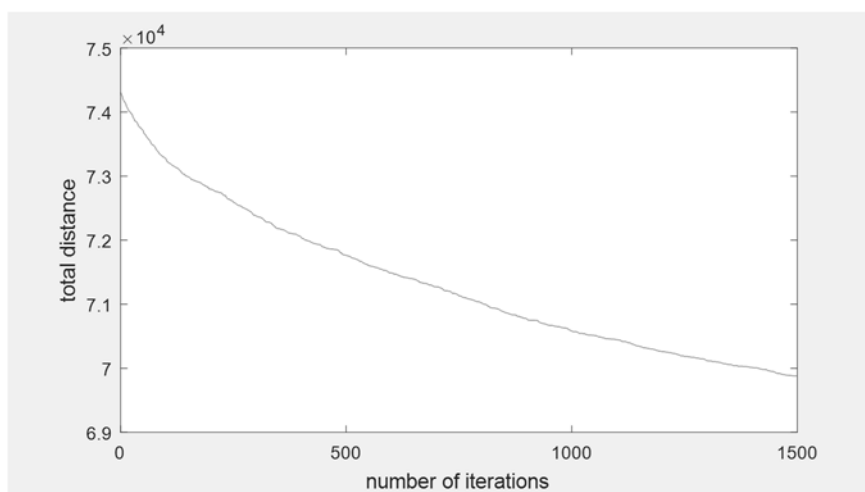


Figure 4.14 Average objective function values along GA iterations for P3

From Figure 4.9 to 4.14, the convergence graph for best and average objective function values along the GA iterations is displayed for the P1, P2 and P3 data set, respectively. The graphs indicate the convergence for the P1 data set, and it can be seen in the X-axis that the convergence occurs around the level of 500. Whereas the convergence is seen at the level of 1500 iterations in the P2 and P3 data sets. For this reason, GA iterations are set at these values.

Table 4.5 Parameter set for the hybrid algorithm

Parameter	Value
PSO ratio	0.1
PSO replications	100
p_1 (crossover rate with particle best)	0.35
p_2 (crossover rate with global best)	0.35

In the second experimental study, the parameters for the proposed hybrid algorithm are determined. In Table 4.5, the final parameter setting for the hybrid algorithm is given. Note that the genetic parameters inside the hybrid algorithm remain same as in the proposed genetic algorithm. In accordance with the ratio given, the PSO procedure is executed. The PSO crossover rates p_1 and p_2 are determined equally.

4.2.4 Design of Parameters for Local Search Based Algorithm

In this section, the experimental study is performed for the proposed local search based algorithm. Since the performance of the algorithm varies according to the parameters inside, design of the values of these parameters helps to increase the effectiveness of the algorithm. For this purpose, factorial design technique is used to decide significant factors in reducing the total distance travelled. In Table 4.6, two levels for the parameters are listed. Levels are determined as 50 and 300 for the number of initial solutions. The characteristics of the swap, move forward and move backward heuristics in the algorithm are seen as similar, therefore these heuristics are accepted as one group, and their levels are set to be the same. However, the 2-opt

heuristic requires more computational effort compared to the other heuristics, thus its upper level is fixed at a lower value not to increase computational time further. Consequently, with three factors and two levels, full factorial experimental design is implemented with two replicates. The experiments are implemented with the P2 data set, and MINITAB software is used for the analysis.

Table 4.6 Parameter levels for the local search algorithm

Parameter	Level 1	Level 2
number of initial solutions	50	300
number of first group of local search (LS1)	20	100
number of 2-opt (LS2)	75	250

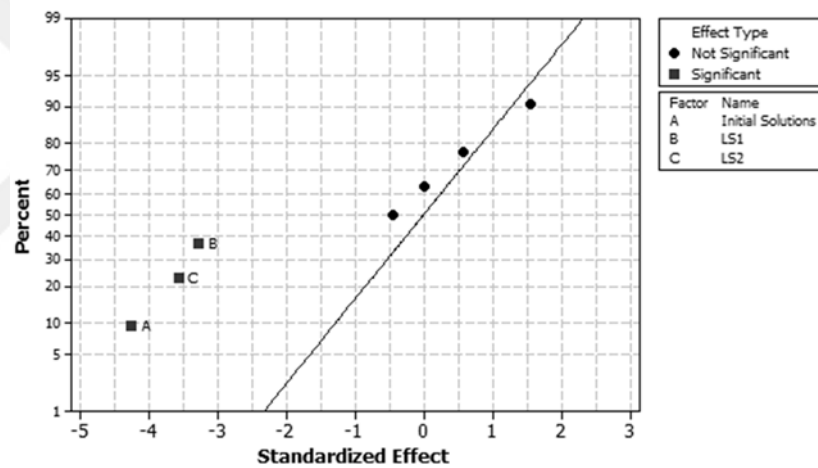


Figure 4.15 Normal probability plot of the standardized effects

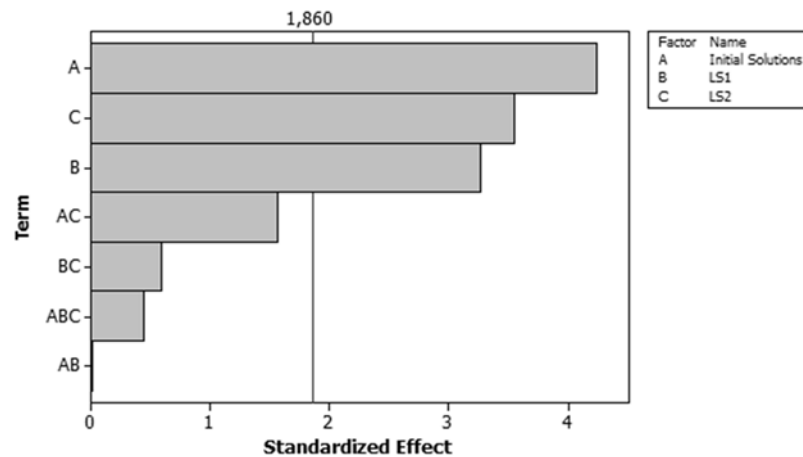


Figure 4.16 Pareto chart of the standardized effects

The advantage of the full factorial technique is that each combination of the parameters is evaluated and the response is recorded to visualize the interactions among the factors. Figure 4.15 and Figure 4.16 display the normal probability plot and Pareto chart of the standardized effects. Three parameters have significant effects on the objective function value.

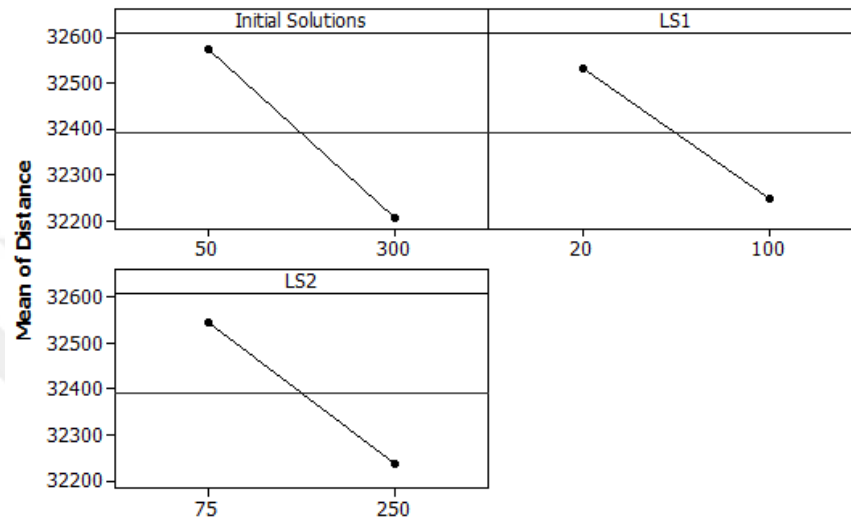


Figure 4.17 Main effects plot (data means)

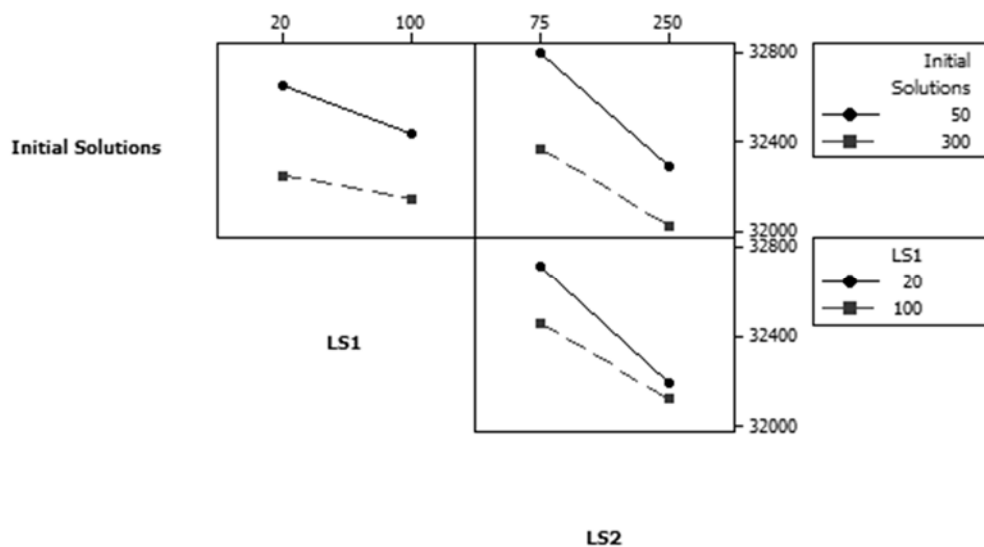


Figure 4.18 Interaction plot (data means)

As it can be seen from Figure 4.17, number of initial solutions iterations has higher main effect than other parameters. Interaction plot (see Figure 4.18) also illustrate that values for Initial Solutions=300, LS1=100, LS2=250 should be chosen

as the parameter settings for the experiments, since they yield lower values for the total distance. After setting these parameters, computational study is implemented.

4.2.5 Computational Study and Results

In the implementation, the algorithms are coded in MATLAB, and experimented on a computer with Intel® Core™ i7-3632QM, 2.2 GHz Processor and 8 GB RAM. After the computations for each of the proposed algorithms are implemented, corresponding results and average computation times are recorded.

Table 4.7 Results for S-shape routing strategy

Method		Best solutions (length unit)	Average	Worst	Dev. %	Computation time (in seconds)
Genetic algorithm	P1	6290	6290	6290	0	18.03
	P2	31680	32370	33085	2.18	139.7
	P3	65935	67481	68490	2.34	428.5
Hybrid algorithm	P1	6290	6290	6290	0	38.6
	P2	31680	32045	33720	1.15	294.4
	P3	66010	67224	68780	1.84	574.6
Local search algorithm	P1	6290	6290	6290	0	11.23
	P2	31980	32960	33220	3.06	26.07
	P3	66960	68240	69100	1.91	89.77
First come first serve (FCFS)	P1	10730	-	-	-	1.2
	P2	56180	-	-	-	1.4
	P3	112670	-	-	-	1.8

In Table 4.7, Table 4.8 and Table 4.9 results of the computational study for S-shape, midpoint and return strategies are presented, respectively. Algorithms are run for 10 times, and then best, worst, and average solutions found are given. Calculation of Dev. %, percentage deviation of the average solution to the best solution, is displayed in Eq. (4.1).

$$Dev.\% = ((average - best)/best) \times 100 \quad (4.1)$$

Table 4.8 Results for mid-point routing strategy

Method		Best solutions (length unit)	Average	Worst	Dev. %	Computation time (in seconds)
Genetic algorithm	P1	6220	6400	6580	2.89	16.1
	P2	34180	35110	36040	2.72	132.2
	P3	69260	70860	72460	2.31	145.9
Hybrid algorithm	P1	6160	6240	6480	1.3	39.8
	P2	34280	35250	36220	2.83	258.6
	P3	70080	71620	73160	2.2	572.4
Local search algorithm	P1	6140	6250	6360	1.79	12.6
	P2	34940	35880	36820	2.69	24.1
	P3	69760	71470	73180	2.45	82.5
First come first serve (FCFS)	P1	12380	-	-	-	1.1
	P2	58880	-	-	-	1.2
	P3	118020	-	-	-	1.9

After implementation of the algorithm, first come first serve (FCFS) rule is implemented and included in the benchmark study. In FCFS heuristic, orders are picked up according to their arrival sequence; therefore, first received order will be in the first batch, second order will be the next order that will be picked up, and so on. According to results, proposed algorithms give obviously better results than FCFS.

Table 4.9 Results for return routing strategy

Method		Best solutions (length unit)	Average	Worst	Dev. %	Computation time (in seconds)
Genetic algorithm	P1	5235	5336	5740	1.93	14.9
	P2	34200	34930	36100	2.13	126.5
	P3	61345	61795	62070	0.73	386.8
Hybrid algorithm	P1	5260	5296	5740	0.68	47.4
	P2	34200	34930	36100	2.13	273.9
	P3	60595	61343	61990	1.23	544.8
Local search algorithm	P1	5280	5360	5840	1.52	10.6
	P2	35680	35970	36260	0.81	23.7
	P3	72720	73334	73820	0.84	79.4
First come first serve (FCFS)	P1	12640	-	-	-	1.1
	P2	59180	-	-	-	1.2
	P3	118440	-	-	-	1.9

The average computation times for the algorithms are seen as reasonable. In the computational study, it is aimed to find a trade-off point while searching in a wide solution area (e.g. high population size and large number of GA iterations) to be able to find good solutions.

4.3 Benchmark Instances

In Chapter 2, it has been stated that there are no standard data sets in the related literature. Considering the work done in recent years, the data sets that were cited multiple times are utilized in the benchmark study.

The experimental study of proposed GA, hybrid GA and local search algorithms is performed with benchmark instances used in the related literature. The algorithms are tested with two different data sets of Henn & Wäscher (2012) which can be

accessed from Optsicom Project (2018). In the following sections, the information on warehouse layout, orders, and picker capacity are given.

4.3.1 Structure of the Warehouse System

In this study, a single block layout is used in the experimental study. Considered warehouse has the same characteristics with the one used in the study of Henn & Wäscher (2012). In Figure 4.19, the warehouse layout which is considered in this study can be seen.

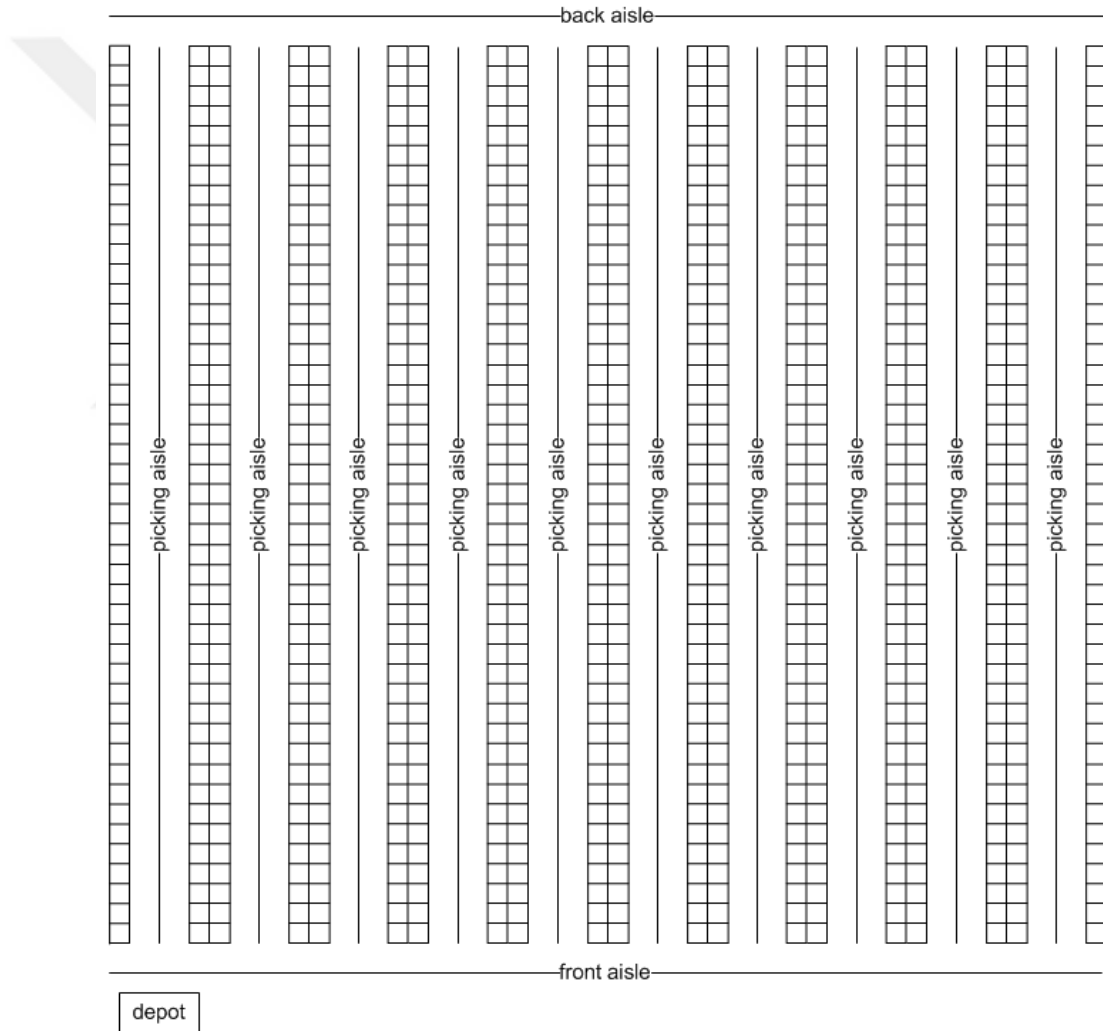


Figure 4.19 Warehouse layout considered in the study

The warehouse layout used in the computational experiments is a single block warehouse with 900 storage locations. It consists of identical storage units, parallel

aisles between locations, two cross aisles as front and back aisles, and an input/output point which can be called as depot. The layout corresponds to a low-level picker-to-parts systems layout in which the picker moves two directions to pick up the items.

In addition to the layout information, a set of assumptions are needed to execute the computations. Following assumptions are determined and used in the experiments:

- Distances between storage locations are computed according to their (x,y) coordinates,
- Order picker can pick up orders from both sides of an aisle,
- Picker-to-part system is considered,
- One order picker is available,
- Storage units are identical,
- Each item has a distinct storage area,
- Static order batching is considered,
- The volume of an order does not exceed the picker capacity.

4.3.2 Computational Study

The computational study of the proposed approaches is implemented in this section. Sequentially, proposed genetic algorithm, hybrid algorithm and the local search approach are applied to the benchmark instances.

Table 4.10 displays the related warehouse configuration. It includes measures and distances related with the aisles and storage locations in the layout. The picker walks in the middle of the aisle, so he/she can pick the orders without making additional movements. The distance between two aisles is 5 units. The length of an aisle including the distance between two cross aisles (front and back aisles) are given as 47 units.

Table 4.10 Warehouse parameters (Henn & Wäscher, 2012)

Parameter	Value
number of picking aisles	10
number of storage locations on each side of an aisle	45
number of total storage locations	900
length of a storage unit	1
width of a storage unit	1.5
aisle width	2
distance between depot and front cross aisle	0.5

Table 4.11 Problem sets (Henn & Wäscher, 2012)

Method	ABC	Random
number of orders	40, 60, 80, 100	40, 60, 80, 100
routing scheme	S-shape	S-shape
average number of articles per customer order	15	15
variance for the number of articles per customer order	10	10
capacity of the picking device (number of items)	30, 45, 60, 75	30, 45, 60, 75

Performances of proposed algorithms are tested with two different data sets of Henn & Wäscher (2012), and related information is given in Table 4.11. Two storage procedures are considered in the data sets; ABC storage and random storage of the items. Number of orders varies from 40 to 100, and capacity of the picker varies from 30 to 75 items. Therefore 32 instances in total are experimented. Three routing strategies are used in the application.

4.3.2.1 Application of the Genetic Algorithm

As mentioned earlier, GA is an evolution based metaheuristic approach, and many studies are focused on development of this method for solving optimization problems since emergence of this conceptual analogy (Gen, Cheng, & Lin, 2008). The GA process that is developed to solve the OBP is explained previously in Chapter 3.

Table 4.12 Parameter set in benchmark study for the genetic algorithm

Parameter	Value
population size (p)	100
crossover ratio	0.75
genetic replications	250
mutation ratio	0.1
mutation replications	20
local search ratio	0.75

Before the implementation of the proposed GA, the parameter set is determined. Since the performance of the algorithm changes according to the parameters, design of the values of these parameters is important to increase the effectiveness. In addition, GA operators and parameter values are selected so as to explore the solution area in a comprehensive way (Cergibozan, 2013). Therefore, an experimental study on the parameters of the proposed algorithm is implemented. Table 4.12 displays the resulting GA parameters for the algorithm.

4.3.2.2 Application of the Hybrid Algorithm

To take advantage of the features of the GA based hybrid scheme, determination of the accurate levels for the parameters are significant. Table 4.13 demonstrates the parameter set for the proposed hybrid algorithm.

Table 4.13 Parameter set in benchmark study for the hybrid algorithm

Parameter	Value
PSO ratio	0.1
PSO replications	100
$p1$ (crossover rate with particle best)	0.35
$p2$ (crossover rate with global best)	0.35

4.3.2.3 Application of the Local Search Algorithm

The parameter set applied in the proposed local search algorithm is displayed in Table 4.14.

Table 4.14 Parameter set in benchmark study for the local search algorithm

Parameters	Value
number of initial solutions	300
number of first group of local search (L1)	100
number of 2-opt (L2)	250

These values are fixed at the same levels used in the illustrative example. The levels of these parameters are seen as best tradeoff points considering the computation time and solution quality.

4.3.2.4 Results and Evaluation

After parameter design, computational experiments are implemented and the results are obtained. Computations of the proposed algorithms with different routing strategies are performed for 32 instances; and best, worst and average tour lengths with corresponding running times are given. The BKS column represents the best known solution with refer to Žulj et al. (2018).

According to the GA results (see Table 4.15, Table 4.16), it is seen that the algorithm is able to reach good solutions in reasonable times. The results are found as consistent and competitive.

The solution quality of the proposed hybrid method can be seen from Table 4.17, Table 4.18. The computation time of the algorithm seems reasonable. It can be derived from the results that the PSO based interaction structure improves the solution quality in most of the instances; however, the computation time is increased in comparison to GA results. This is because the PSO scheme within the algorithm is

an additional component and requires lots of search moves between the individuals in the population.

Table 4.15 GA results for ABC1 with S-shape strategy

Order size	Picker capacity	BKS	Best solutions (length unit)	Average	Worst	Dev	Time (in seconds)
40	30	7874	8478	9039	10262	0.08	9.2
	45	5029	5608	6246	6588	0.12	
	60	4479	5026	5156	5225	0.12	
	75	3336	3658	3852	3933	0.1	
60	30	12896	14800	16188	17406	0.15	27.4
	45	7786	9423	9805	10364	0.21	
	60	6103	7126	7277	7601	0.17	
	75	4241	4981	4986	4991	0.17	
80	30	17268	18760	19354	20858	0.09	71.8
	45	10920	12662	13528	14250	0.16	
	60	7661	9212	9502	9736	0.2	
	75	6330	7699	7741	7774	0.22	
100	30	17789	19833	22208	23480	0.11	188.4
	45	11406	14252	14963	15748	0.25	
	60	8889	11057	11729	12130	0.24	
	75	7505	9449	9634	9726	0.26	

Table 4.16 GA results for Ran1 with S-shape strategy

Order size	Picker capacity	BKS	Best solutions (length unit)	Average	Worst	Dev	Time (in seconds)
40	30	10751	12008	12370	12588	0.12	11.3
	45	6759	7556	7627	7692	0.12	
	60	5846	5947	6538	6795	0.02	
	75	4639	4680	5019	5225	0.01	
60	30	16755	19418	19505	19540	0.16	28.2
	45	10282	12012	12335	12643	0.17	
	60	8246	9637	9683	9751	0.17	
	75	5783	6334	6416	6524	0.1	
80	30	22711	26326	26584	26934	0.16	74.3
	45	15261	17651	17813	17978	0.16	
	60	10283	11565	11814	12303	0.12	
	75	8377	9715	9731	9759	0.16	
100	30	24027	29256	29348	29440	0.22	192.1
	45	15493	18843	19082	19320	0.22	
	60	11880	14112	14193	14271	0.19	
	75	9696	11465	11545	11682	0.18	

Table 4.17 Hybrid algorithm results for ABC1 with S-shape strategy

Order size	Picker capacity	BKS	Best solutions (length unit)	Average	Worst	Dev	Time (in seconds)
40	30	7874	8478	9049	10186	0.08	16.4
	45	5029	5608	6133	6545	0.12	
	60	4479	4890	5147	5389	0.09	
	75	3336	3906	4041	4215	0.17	
60	30	12896	14533	15787	17024	0.13	43.6
	45	7786	9423	9909	10362	0.21	
	60	6103	7306	7679	8070	0.2	
	75	4241	5126	5303	5398	0.21	
80	30	17268	18880	20741	22308	0.09	92.7
	45	10920	13219	13905	14256	0.21	
	60	7661	8988	9698	10159	0.17	
	75	6330	8038	8203	8333	0.27	
100	30	17789	19833	22634	25036	0.11	209.1
	45	11406	14252	15194	15842	0.25	
	60	8889	11057	11711	12117	0.24	
	75	7505	9449	9645	9756	0.26	

Table 4.18 Hybrid algorithm results for Ran1 with S-shape strategy

Order size	Picker capacity	BKS	Best solutions (length unit)	Average	Worst	Dev	Time (in seconds)
40	30	10751	12008	12192	12299	0.12	15.8
	45	6759	7583	7702	7826	0.12	
	60	5846	6739	6776.3	6804	0.15	
	75	4639	5103	5148.2	5181	0.1	
60	30	16755	19473	19503	19550	0.16	41.4
	45	10282	12097	12277	12557	0.18	
	60	8246	9637	9694	9751	0.17	
	75	5783	6334	6416.3	6524	0.1	
80	30	22711	26326	26670	27137	0.16	93.5
	45	15261	17651	17772	17978	0.16	
	60	10283	11565	12166	12863	0.12	
	75	8377	9715	9810.3	9965	0.16	
100	30	24027	29256	29470	29684	0.22	214.2
	45	15493	18260	18552	18843	0.18	
	60	11880	14112	14193	14271	0.19	
	75	9696	11465	11487	11526	0.18	

In Table 4.19 and Table 4.20, computational results of the proposed local search algorithm are given.

Table 4.19 Local search algorithm results for ABC1 with S-shape strategy

Order size	Picker capacity	BKS	Best solutions (length unit)	Average	Worst	Dev	Time (in seconds)
40	30	7874	8478	8934	9141	0.08	8.7
	45	5029	5877	5889	5914	0.17	
	60	4479	5173	5230	5327	0.15	
	75	3336	3801	3854	3934	0.14	
60	30	12896	14987	15217	15543	0.16	22
	45	7786	9423	9427	9431	0.21	
	60	6103	7318	7344	7369	0.2	
	75	4241	5092	5095	5098	0.2	
80	30	17268	20135	20224	20312	0.17	60.5
	45	10920	13499	13511	13522	0.24	
	60	7661	9414	9442	9469	0.23	
	75	6330	7639	7656	7673	0.21	
100	30	17789	22469	22529	22588	0.26	159.3
	45	11406	14252	14358	14411	0.25	
	60	8889	11057	11188	11318	0.24	
	75	7505	9449	9471	9492	0.26	

Table 4.20 Local search algorithm results for Ran1 with S-shape strategy

Order size	Picker capacity	BKS	Best solutions (length unit)	Average	Worst	Dev	Time (in seconds)
40	30	10751	11746	12092	12182	0.09	8.2
	45	6759	7434	7588	7776	0.1	
	60	5846	6401	6414.5	6419	0.09	
	75	4639	5160	5180.5	5194	0.11	
60	30	16755	18445	18723	19556	0.1	21.4
	45	10282	11993	12267	12515	0.17	
	60	8246	9145	9178.7	9246	0.11	
	75	5783	6339	6392.3	6447	0.1	
80	30	22711	25430	25430	25430	0.12	57.1
	45	15261	17265	17458	17651	0.13	
	60	10283	11516	11698	12012	0.12	
	75	8377	9646	9646	9646	0.15	
100	30	24027	28601	29159	29717	0.19	153.9
	45	15493	18336	18535	18734	0.18	
	60	11880	13964	14050	14112	0.18	
	75	9696	11458	11514	11622	0.18	

Although the algorithm uses basic local search techniques inside, it has the advantage of both running time and solution quality in comparison to other algorithms. Therefore, it can be stated that the local search based algorithm may be preferred among the proposed techniques.

Table 4.21 GA results for ABC1 with return and mid-point strategies

Order size	Picker capacity	Best solutions return	Average	Best solutions mid-point	Average	Time (in seconds)
40	30	7272	7298	7802	7888	8.6
	45	4569	4620	5647	5687	
	60	4364	4373	4943	4973	
	75	3321	3340	3505	3540	
60	30	11376	11407	13499	13668	26.1
	45	7441	7482	8466	8506	
	60	6251	6284	6648	6729	
	75	4068	4075	4753	4778	
80	30	15112	15222	17242	17583	73.5
	45	10877	10883	12145	12277	
	60	7678	7714	8685	8721	
	75	6111	6247	7091	7099	
100	30	16639	16732	19103	19191	182.9
	45	11186	11200	13299	13330	
	60	8885	8922	10541	10586	
	75	7430	7460	8656	8696	

Table 4.22 GA results for Ran1 with return and mid-point strategies

Order size	Picker capacity	Best solutions return	Average	Best solutions mid-point	Average	Time (in seconds)
40	30	8842	8877	10914	11177	9.1
	45	5611	5726	7150	7155	
	60	5284	5349	6614	6702	
	75	4074	4128	4539	4605	
60	30	14355	14355	17707	17736	26.8
	45	9026	9111	11426	11456	
	60	7233	7285	8879	8901	
	75	5390	5431	6365	6451	
80	30	18682	18813	24206	24252	76.2
	45	13431	13527	17319	17337	
	60	9184	9308	11377	11466	
	75	7874	7889	9124	9159	
100	30	20441	20476	26179	26429	187.3
	45	13799	13819	18091	18116	
	60	10926	10926	13299	13350	
	75	9253	9301	11649	11677	

From Table 4.21 to Table 4.26, proposed genetic, hybrid and local search algorithms are compared by using return and mid-point routing strategies.

Table 4.23 Hybrid algorithm results for ABC1 with return and mid-point strategies

Order size	Picker capacity	Best solutions return	Average	Best solutions mid-point	Average	Time (in seconds)
40	30	7058	7115	7762	7829	14.4
	45	4519	4519	5576	5611	
	60	4369	4399	4949	4972	
	75	3326	3353	3427	3487	
60	30	11484	11506	13352	13423	42.1
	45	7376	7404	8320	8456	
	60	6194	6299	6771	6790	
	75	4051	4100	4893	4930	
80	30	15138	15168	17656	17690	95.6
	45	10690	10816	12146	12212	
	60	7650	7662	8565	8636	
	75	6197	6267	7123	7171	
100	30	16638	16745	19159	19345	218.4
	45	10981	11112	13375	13460	
	60	8966	8973	10406	10478	
	75	7471	7572	8672	8806	

Table 4.24 Hybrid algorithm results for Ran1 with return and mid-point strategies

Order size	Picker capacity	Best solutions return	Average	Best solutions mid-point	Average	Time (in seconds)
40	30	8843	8866	10900	11086	15.9
	45	5731	5767	7119	7232	
	60	5329	5349	6869	6890	
	75	4003	4034	4710	4757	
60	30	14219	14232	17862	17898	43.7
	45	9067	9130	11448	11467	
	60	7427	7442	8924	9069	
	75	5442	5479	6252	6394	
80	30	18574	18594	23950	24036	92.3
	45	13322	13468	17394	17425	
	60	9262	9270	11523	11574	
	75	7879	7952	9122	9202	
100	30	20280	20363	26166	26199	221.5
	45	13779	13831	17990	18056	
	60	10941	10953	13267	13563	
	75	9159	9200	11496	11555	

Since there have not been any best solution values of these data sets with return and mid-point strategies, these results are given for making a comparison between both strategies.

Table 4.25 Local search algorithm results for ABC1 with return and mid-point strategies

Order size	Picker capacity	Best solutions return	Average	Best solutions mid-point	Average	Time (in seconds)
40	30	7052	7212	8089	8257	7.6
	45	4814	4822	5981	6039	
	60	4501	4549	5167	5258	
	75	3377	3411	3838	3843	
60	30	12136	12249	14199	14224	19.4
	45	7552	7571	9067	9081	
	60	6481	6492	7171	7178	
	75	4180	4186	5154	5223	
80	30	15216	15519	17781	17950	58.1
	45	10988	11014	12815	12844	
	60	7911	7913	9256	9270	
	75	6411	6445	7562	7624	
100	30	16786	16857	20114	20176	157.6
	45	11134	11216	14116	14238	
	60	9086	9141	10960	10990	
	75	7740	7745	9147	9353	

Table 4.26 Local search algorithm results for Ran1 with return and mid-point strategies

Order size	Picker capacity	Best solutions return	Average	Best solutions mid-point	Average	Time (in seconds)
40	30	9054	9129	11821	11905	7.9
	45	5811	5820	7564	7569	
	60	5469	5477	7321	7339	
	75	4176	4178	5006	5059	
60	30	14513	14594	18621	18737	21.3
	45	9322	9365	11868	11903	
	60	7478	7493	9492	9500	
	75	5613	5654	6627	6681	
80	30	18717	18762	24880	25168	58.7
	45	13875	13883	17939	17982	
	60	9451	9499	12080	12103	
	75	7983	8034	9772	9781	
100	30	20338	20484	27089	27364	156.2
	45	14003	14179	18675	18722	
	60	11015	11020	14151	14201	
	75	9462	9466	12089	12170	

The computation times of the algorithms are reasonable. According to the results, it is seen that these approaches can be used for solving the OBP in different warehouse configurations and picking systems.

From Figure 4.20 to Figure 4.25, the graphical representations of the results are shown.

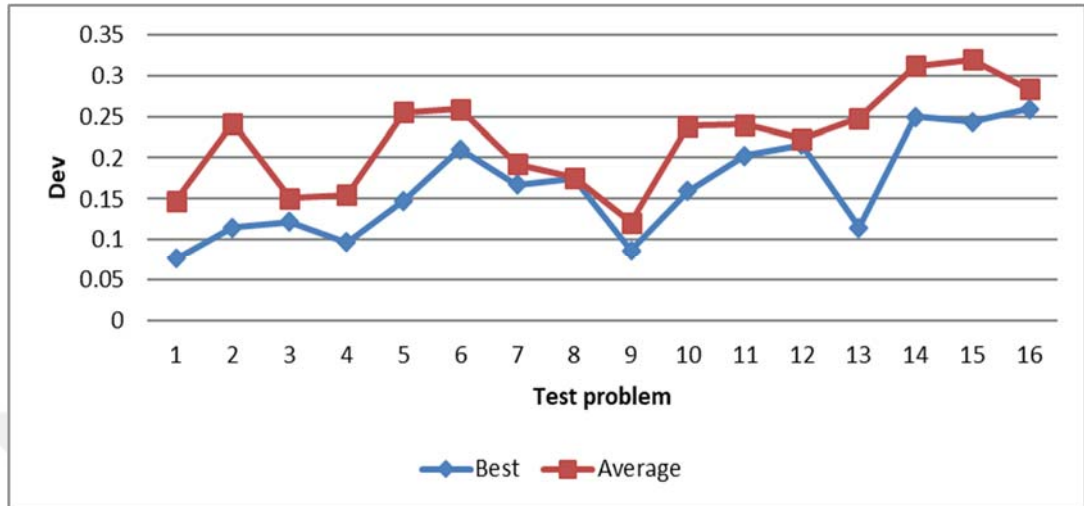


Figure 4.20 Results for genetic algorithm in terms of deviations for ABC1

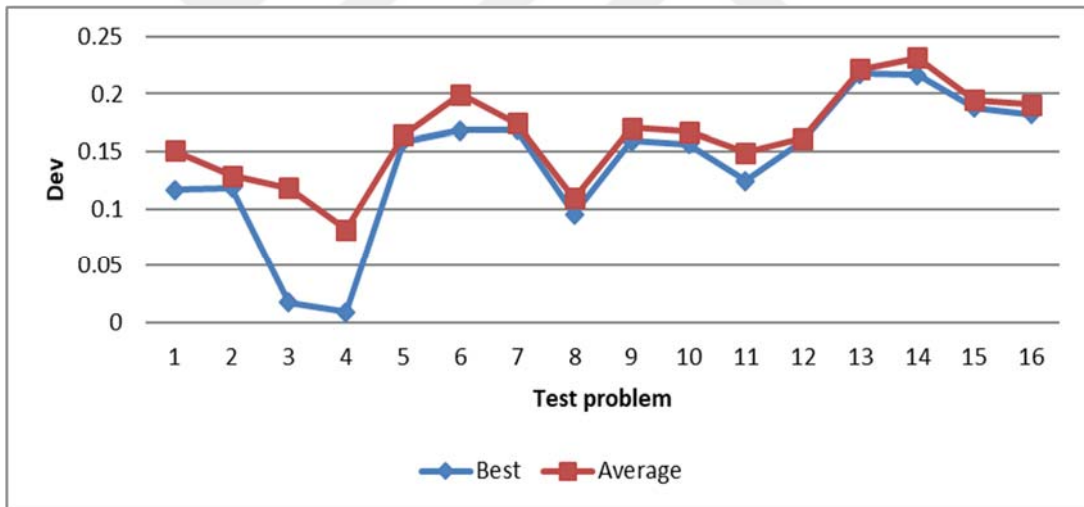


Figure 4.21 Results for genetic algorithm in terms of deviations for Ran1

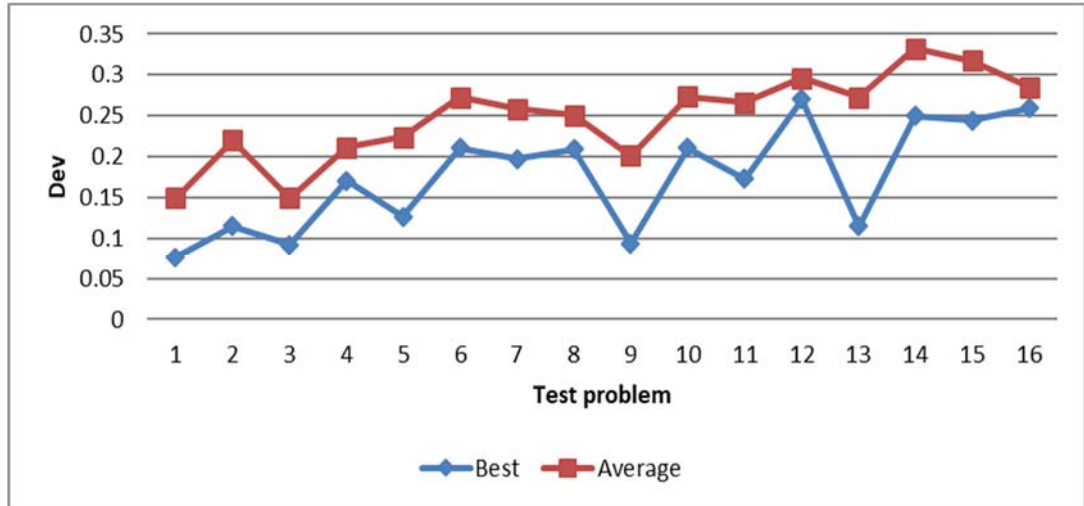


Figure 4.22 Results for hybrid algorithm in terms of deviations for ABC1

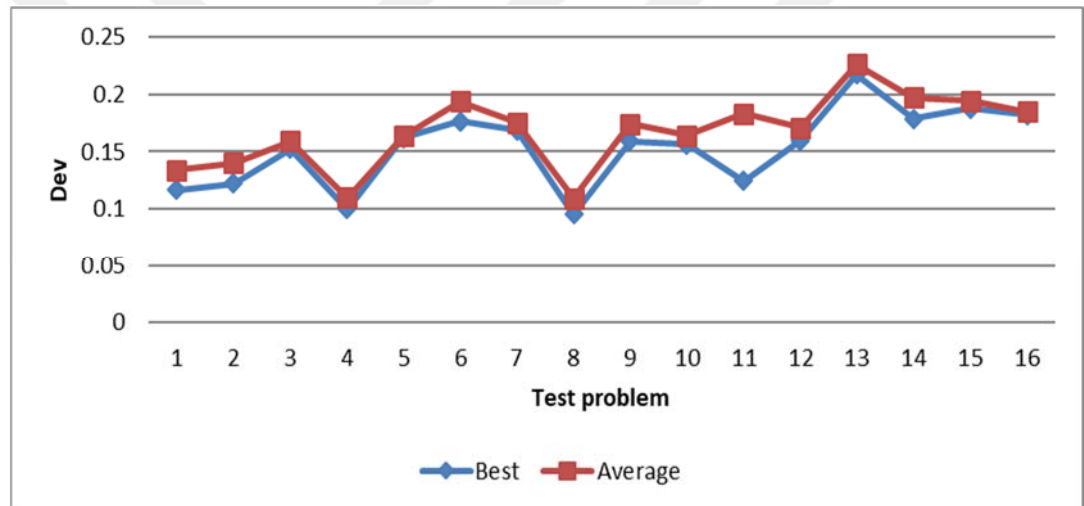


Figure 4.23 Results for hybrid algorithm in terms of deviations for Ran1

These figures represent the percentage deviations of the algorithms for S-shape strategy. Best line means the deviation related with the best found solution with the algorithm; and average line displays the deviation related with the average found solution with the algorithm. In general, the genetic, hybrid and local search algorithm results are steady in consideration with each instance.

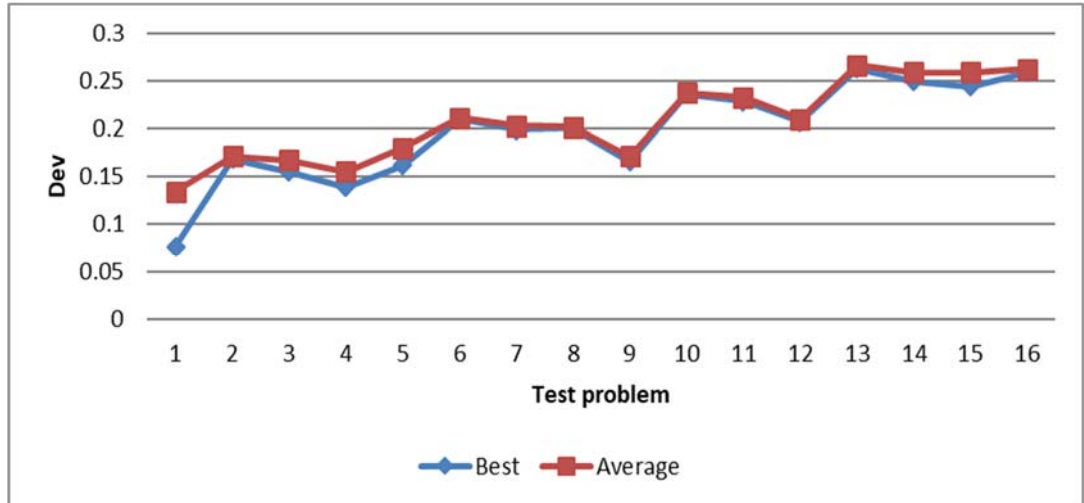


Figure 4.24 Results for local search algorithm in terms of deviations for ABC1

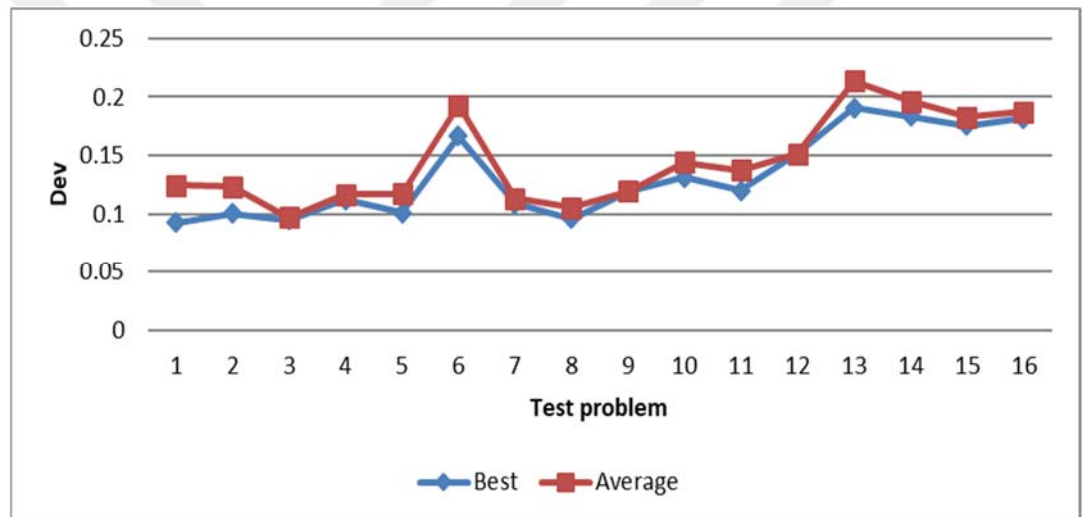


Figure 4.25 Results for local search algorithm in terms of deviations for Ran1

4.4 Conclusion

In this study, the OBP is investigated with two genetic algorithm based solution approaches, and a local search based approach. At first, the algorithms are experimented with an illustrative case of OBP. The effectiveness of the algorithms is also compared against the FCFS heuristic. Results of the algorithms show that they are capable of generating good solutions for the problem in a short time period.

After the first experimental study with generated data sets, developed algorithms are evaluated with the benchmark data obtained from the related literature. According to the results, developed approaches are able to find reasonable solutions to OBP. The computation times of the algorithms are relatively small. It is seen from the experiments that the more complex the algorithm is, the longer the calculation time will be.

The aim of the computational study is to display the applicability and effectiveness of developed algorithms. However, algorithms are hardly comparable with the metaheuristic approaches proposed in other studies by means of computation times. As it is stated previously in Chapter 2, the reason of that is each study uses a different computation environment. Algorithms are needed to be run under the same conditions to compare the time, which seems impossible. In addition, the available benchmark examples are limited in the literature. The number of studies using the same benchmark instances is too few, and the assumptions used in these studies may differ. This situation may result with different solutions that also yield different objective function values. Therefore, the study is concluded with the key findings on proposed algorithms.

Additionally, the initial cluster of solutions in the algorithms is generated randomly. Constructing initial solutions based on a specific approach is also another solution technique, but it is not examined here. This will probably be a further research of this study.

In the next chapter, a real case study in the distribution center of a well-known retailer in Turkey is implemented. The case study includes a layout design study and optimization approaches for the OP process. The main goal of the case study is to test the proposed algorithms and apply them in a real operating system.

CHAPTER FIVE

CASE STUDY IN A DISTRIBUTION CENTER FROM THE RETAIL SECTOR

5.1 Introduction

Real-life applications of the optimization problems can be quite different from theoretical work. We may need to work with some constraints that were previously assumed or ignored. Therefore, the main goal throughout the implementation process will be understanding the structure of the layout and developing the most appropriate solution. It is seen that the proposed algorithms for the OBP give effective results after analysis of the computational studies in Chapter 4. In this study, the warehouse operations in the distribution center (DC) of a well-known retailer in Turkey will be studied. In the following sections, the layout of the DC is presented with its characteristics, the order and picking information of the current system is described considering the warehouse layout and routing scheme. The application project includes both layout design and picking optimization. Finally, the study is concluded with the future research directions.



Figure 5.1 Course of action for the case study

The implementation process in the DC is shown in Figure 5.1. At first, we identified the structure of the layout. This stage includes determination of the physical boundaries and dimensions of the DC, number of storage locations, widths and lengths of the storage units, number of aisles and their positioning, location of the receiving and shipment areas, number of receiving and shipment docks, and other basic measurements and within the DC. Since the DC is operated coordinative with the top management, hard constraints may occur within operational processes. Consequently, after analysis of the layout, we determined the procedures, policies

and related constraints that cannot be changed, e.g. movement type of the picking process, layout of the DC, and assignment of the articles in the main storage area. These constraints clearly give a direction to the optimization approaches that will be developed. It is essential at this point that the proposed method should be acceptable, easy to understand and can be applied by the whole participants within the DC. Therefore, the most convenient approach is applied to tackle the order batching and picking problem within the DC in the light of the whole process.

5.2 Presentation of the Case Study

In this study, a DC of a well-known retail store in Turkey is examined for the case study. The company is one of the leading companies in the retailing sector in Turkey, and has a total of 2078 stores in all provinces at Turkey. The company continuously aims to be the leading company in the sector by developing innovative and sustainable approaches. It seeks to achieve its goal through customer-focused and environmentally friendly service, a wide range of products and appropriate pricing strategies.

The DC is one of 13 DCs of the company, and it is located in Kemalpaşa district, Izmir. The DC in Kemalpaşa serves the stores in Izmir and Aegean Region with the wholesale operations and distribution operations. Details of the DC's layout are given in the following sections.

5.2.1 Layout of the Distribution Center

In Figure 5.2, the general view of the layout is displayed. The layout consists of receiving and shipping docks, storage areas and office buildings.

The storage policy of the DC can be explained as a “frequent item based” approach. The ABC storage system is used in this layout; therefore, the most frequently ordered items are located in the nearest storage locations to the shipment area. The depot has approximately 7600 stock keeping units (SKUs) that are located

at different storage locations. The capacity of the picking vehicle is determined by means of the volume related with the ordered items. In other words, the dimension of the item is evaluated while filling the pallet.

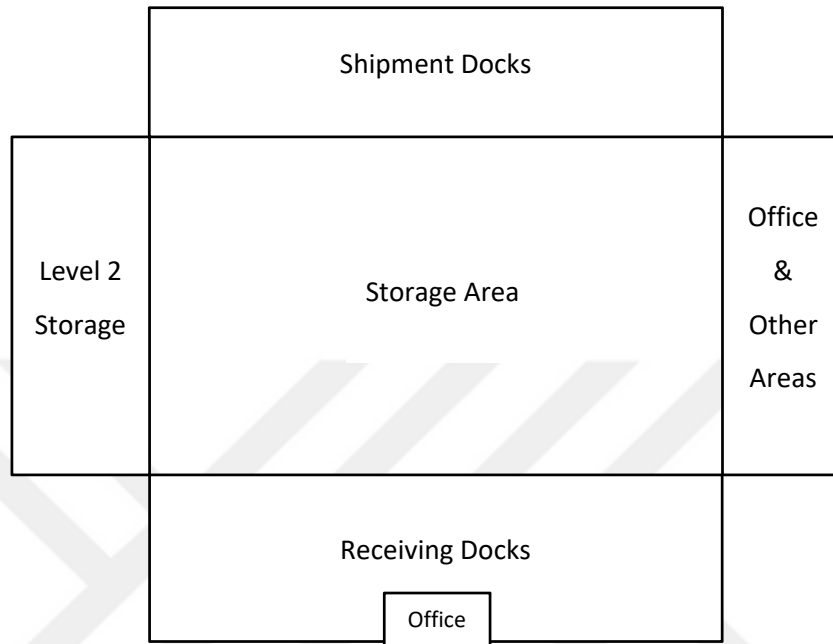


Figure 5.2 The layout of the DC

The picking route of the items is given in Figure 5.3, and can be described as follows:

1. At first, the ordered items are identified with their size.
2. The picker starts from the receiving area to pick up the items. The nearest item to the front aisle are picked first, then the second item in the same rack (if available) of the same aisle are picked, and so on.
3. In the same aisle, the U-shape picking scheme is used. When crossing the aisle, the nearest aisle is entered.

Note that the U-shape picking scheme is the version of the return strategy where the picker does not pass the opposite rack unless he picks all the ordered items in the related rack.

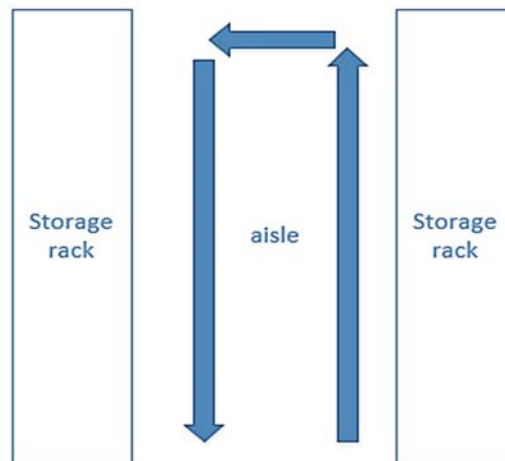


Figure 5.3 The U-shape picking within aisle

The total number of the pickers is 25, and the performance of a picker is measured as the total number of the items picked. Daily and hourly target values are assigned for each order picker; therefore, the picker tries to reach these predetermined targets, e.g. number of orders picked on-time. From Figure 5.4 to Figure 5.7, some images from the DC can be seen.



Figure 5.4 An aisle view from the DC (Personal archive, 2018)



Figure 5.5 Receiving area at the DC (Personal archive, 2018)



Figure 5.6 Labeling example (Personal archive, 2018)



Figure 5.7 Labeling example of adjacent SKUs (Personal archive, 2018)

The performance of the current picking system is screened by the managers in terms of total picking time; therefore, proposed genetic algorithm based approaches

are applied to the case study to evaluate the current performance of the system. Also the local search based metaheuristic technique is used to solve this problem at the DC.

5.2.2 Details of the Current Picking System

In the DC, current state of the layout and the information on orders are as follows:

- Distances between storage locations are computed according to their (x,y) coordinates with the consideration of only vertical and horizontal movements of the picker and the picking vehicle,
- Order picker moves 4 footsteps while passing to the opposite side of the same aisle,
- Picker to part warehouse system is considered,
- Only one storage location is assigned for each item type,
- Orders are known and divisible, and do not change during the picking process,
- Capacity of the picking device is higher than the volume of each customer order.

It is important to point out that, for each individual, assignment of the items to batches is implemented until the total capacity of the picking device is not violated; otherwise a new batch is filled. The layout parameters of the DC are shown in Table 5.1.

The current layout of the DC is shown in Figure 5.8. The DC has a multi-block warehouse layout with approximately 7600 storage locations. The layout of the DC consists of 19 parallel aisles including non-identical storage units; three cross aisles as front, back, and middle cross aisles; and receiving and shipment areas that are located in front of and behind the depot, respectively.

Table 5.1 Parameters of the DC

Parameter	Value
number of picking aisles	19
number of storage locations on each side of an aisle	200
number of total storage locations	7600
length of a storage unit	2 footsteps
width of a storage unit	2 footsteps
aisle width	4 footsteps
distance between receiving point and front cross aisle	5 footsteps

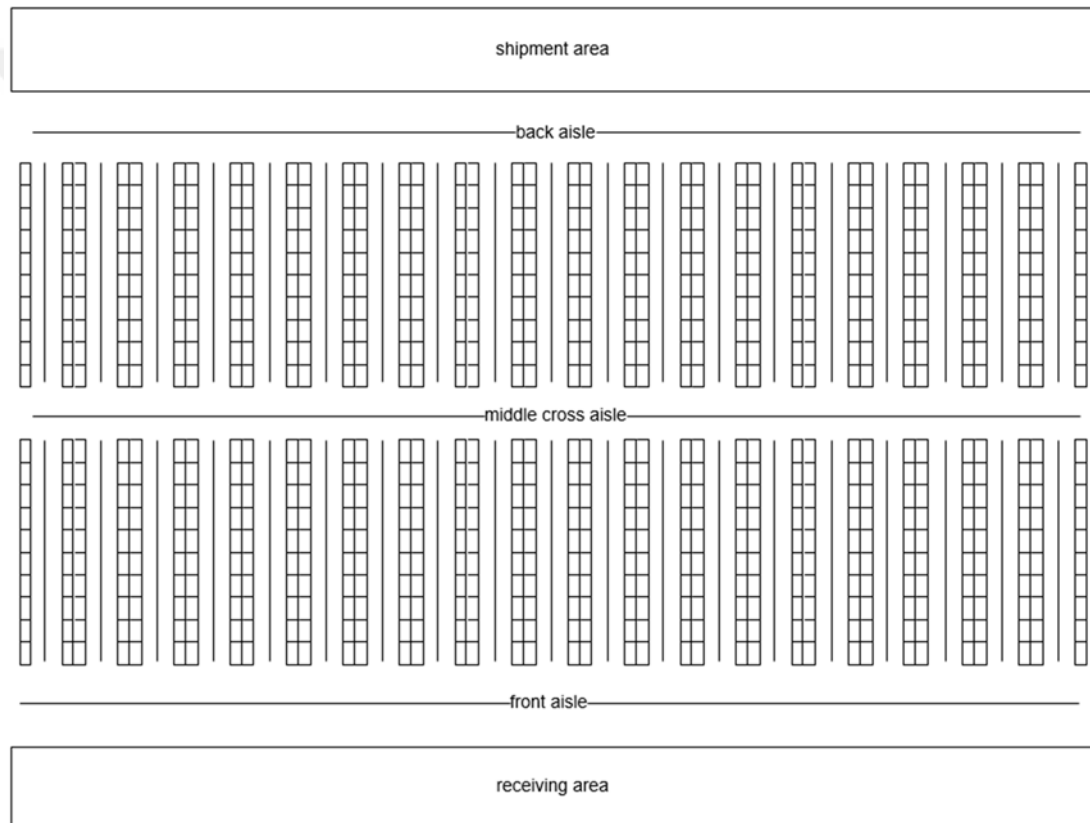


Figure 5.8 Current storage area of the DC

The layout corresponds to a low-level picker-to-parts system where the picker moves two directions to pick up the items. Note that the upper levels of the racks are used only for temporary storage of the items. The planning of the stock unit levels is made independently from the DC; therefore, the received stock units are located in the nearest available locations in the DC.

5.2.3 Proposed Strategies for the Storage Design

While the optimization process is in progress, seasonal demands with high rates of consumption become an issue that is needed to be solved. The Level 2 storage area is decided to be used to store these items by the managers of the DC, and the current layout of the Level 2 is given in Figure 5.9. The Level 2 layout consists of nine distinct racks, and three aisles. The depot is placed in the middle of the layout. Currently, the storage area is not in use; and it is requested that future SKUs should be placed in this area appropriately. The main aim is to position the products with higher demands to the closest shelves to the depot point.

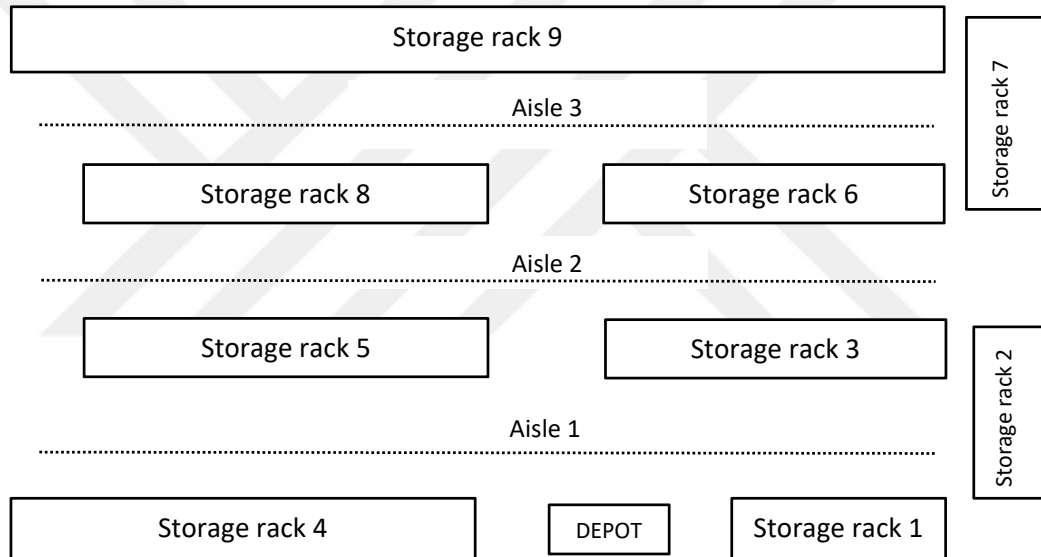


Figure 5.9 Current layout of the Level 2 storage area

The layout parameters of Level 2 area are expressed in Table 5.2. At first, the values of these parameters are recorded after measuring the related distances step by step, and then the storage design is implemented. Images from Level 2 storage area is given in Figure 5.10 and Figure 5.11.

Table 5.2 Parameters of the Level 2 storage area

Parameter	Value
number of picking aisles	3
number of identical storage units	58
number of storage locations within a storage unit	3
number of total SKU locations	174
length of a storage unit	160 cm
width of a storage unit	270 cm
aisle width 1	630 cm
aisle width 2	350 cm
aisle width 3	330 cm
distance between depot and the nearest storage location	700 cm



Figure 5.10 Front view from the depot (Personal archive, 2018)



Figure 5.11 Side view from the depot (Personal archive, 2018)

To assign the incoming products to the empty racks in the Level 2 area, the ABC storage analysis is applied. ABC analysis is a classification technique for the

inventory (Ivanov, Tsipoulanidis, & Schönberger, 2017). This technique is based on the Pareto principle and evaluation of the inventory according to the demand amounts. It is believed that 80% of the demand corresponds to 20% of total items. In this approach, the most demanded items are called as A-level items; the subsequent ones are called as B-level and C-level items, respectively.

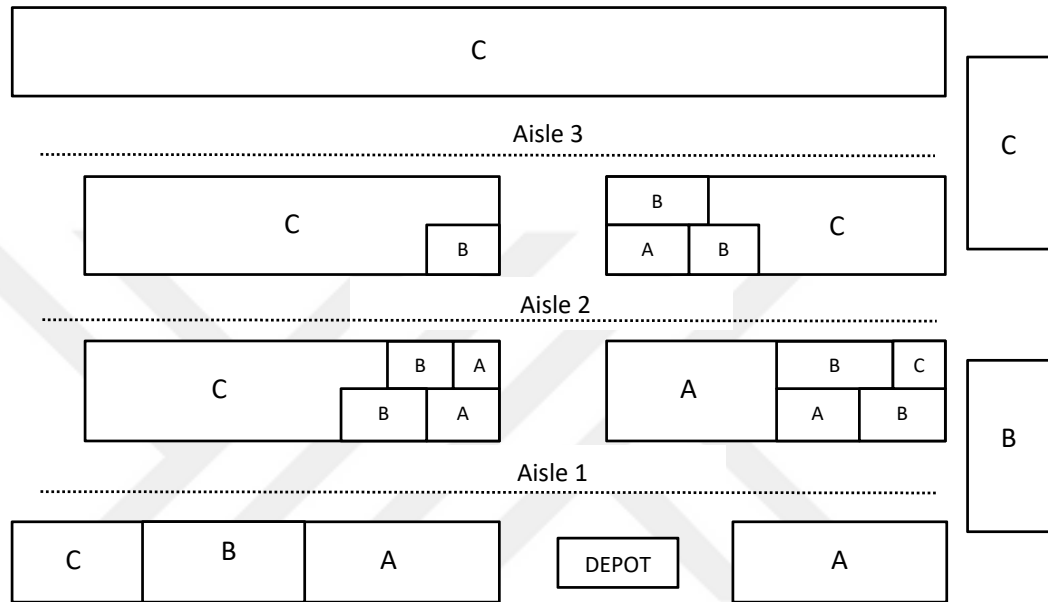


Figure 5.12 Proposed assignment approach for Level 2 storage area

In class based storage systems, the most demanded items are placed in the closest storage units to depot and these locations are expected to be visited frequently. Therefore, the total distance the picker moves from/to depot will be reduced. Consequently, the advantage of this technique is that the picking process will be performed in a shorter time.

The application stage starts after the volume information of the incoming products arrives. The real study comprises of determination of product labelling, addressing and product placement to storage racks. The placement operation is implemented according to the aisle lengths and product labels. The resulting storage assignment can be seen in Figure 5.12. The A, B and C class items are located in related areas.

5.2.4 Metaheuristic Approaches for the Order Picking and Batching Operations

The main aim in the case study is to optimize the picking process in the DC. Currently, picking operations are implemented by a number of workers whom are assigned to different zones. In the case study, the order information data of one picking zone is obtained; therefore, the picking optimization is implemented only for this picking zone.

The related data belongs to single day orders from several customers. Pickers follow a predetermined route to pick up the orders. Current routing strategy is previously described as U-shape within an aisle. Locations of items from an example order are shown in Figure 5.13.

The proposed GA, hybrid and local search based algorithms are analyzed with the data obtained from the DC. The example data set includes a large number of orders with order and picker performance details. Table 5.3 displays the parameters related with the order and picking information in the DC. The subset column represents the considered case of the order picker for one day.

Table 5.3 An example set of orders

Parameter	Total	Subset
number of ordered items	3723	235
routing scheme	U-shape within aisle	U-shape within aisle
average number of items per customer order	32	17
number of customers	118	14
number of pickers	6	1

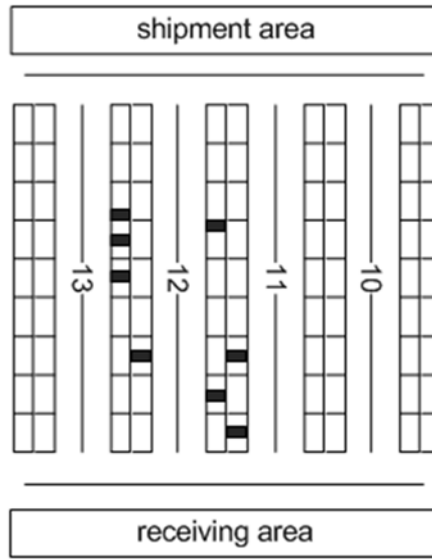


Figure 5.13 Example locations of ordered SKUs

In the implementation, MATLAB software is utilized. Table 5.4, Table 5.5 and Table 5.6 show the parameter levels for proposed algorithms. Related experimental study and computations of the proposed algorithms are implemented to the case in the DC. The algorithms have been executed a number of times, and final computational results are given in Table 5.7.

Table 5.4 Parameters for the genetic algorithm

Parameter	Value
population size (p)	50
crossover ratio	0.8
genetic replications	50
mutation ratio	0.1
mutation replications	20
local search ratio	0.75

Table 5.5 Parameters for the hybrid algorithm

Parameter	Value
PSO ratio	0.1
PSO replications	50
p_1 (crossover rate with particle best)	0.35
p_2 (crossover rate with global best)	0.35

Table 5.6 Parameters for the local search algorithm

Parameter	Value
number of initial solutions	50
number of first group of local search (L1)	20
number of 2-opt (L2)	20

Table 5.7 Comparison of the results for the case study

Method	S-shape routing strategy	Mid-point routing strategy	Return routing strategy	Average computation time (in seconds)
Genetic algorithm	58.4	62.7	59.1	9.08
Hybrid algorithm	58.4	62.7	58.8	10.16
Local search algorithm	60.7	65.9	59.4	8.12
Current picking method	278	278	278	-

According to the results, the algorithms are clearly superior to the current picking routine in terms of computational time. The results also show that algorithms give comparable solutions with each other. Note that the information is based on real data and the length units are measured by footsteps. The solution of the current method includes only the time spent while picking the orders. When the footstep measure is converted to the total time spent by the picker, the picker spends approximately 278 minutes to collect the items. However, proposed algorithms yields effective and efficient results to the problem in a very short time.

5.3 Results and Conclusion

In this chapter, order picking operations of a well-known retailer's DC are chosen for the case study of the warehouse optimization. The first study in the DC is the implementation of a class based storage assignment of the related SKUs in the Level

2 storage area. The application is finalized in the DC. According to the feedbacks from both manager and staff, this study is appreciated, and used by the order pickers right after the conclusion.

In the second study, developed GA, hybrid and the local search based algorithms are applied to the picking operations of the DC, and the performances of the algorithms are compared with the current performance of the picking system. According to result of the computational study, the algorithms are highly effective at solving the picking problem in the DC. According to the results, the algorithms decrease picking time significantly, and they also provide considerable cost saving in labor costs. The DC has no strict picking policies; therefore, it provides flexibility in the implementation of the proposed algorithms.

The operation system of the DC is basically quantity based inventory tracking. The DC measures performances in terms of error rates in the ordered items and also in target values of the pickers. The future study of this case study will be determination of the optimal strategy to measure the picker performance. Decisions on determination of storage locations for all items and layout design can also be considered in the future. Especially, data mining techniques will be useful in arrangement of the layout of the DC, e.g. association rules of the items.

CHAPTER SIX

CONCLUSION

6.1 Summary of the Thesis

The main objective of this thesis is identified as developing fast and effective solution approaches that can solve problems encountered in warehouse order picking process. When the subject is focused, it is seen that the most important and time consuming problem in the warehouses is the problem of order batching and picking with a convenient routing approach. For this reason, the main focus of the study is identified as the order batching problem (OBP).

At first, a comprehensive literature investigation on order picking and batching operations is implemented. Since the database search is resulted with very high number of studies, the search is narrowed according to some criteria. After the filtering stage, the studies on hand are focused, and analyzed in detail.

The literature review has provided a framework to the thesis. In the light of the information obtained, appropriate methods are designed to solve the OBP. Two genetic algorithm based approaches and a local search based algorithm that includes several heuristic search techniques are developed. The performances of the algorithms are first evaluated by applying them on sample problem sets. Afterwards, computational experiments are carried out with the instances from the studies that have been cited most frequently in the literature.

Making a real application besides theoretical solutions is one of the goals in this thesis. In this direction, a real case study is implemented in one of the leading companies in the retail sector. Basic operations and processes at the distribution center (DC) of the company are examined, and important constraints and policies are defined. After analyzing the structure of the order picking processes, the issues that need improvement are identified, and appropriate solution approaches are developed.

6.2 Main Findings

The literature review on order picking and batching operations provides a framework to the thesis. As it can be seen from the findings section in Chapter 2, the number of metaheuristic solution approaches for the OBP has increased significantly in recent years. Consequently, three metaheuristics are developed to solve the OBP. The performances of the algorithms are tested via the experimental studies in Chapter 4. In the computational study, it is seen that the algorithms are competitive and yield good results. The effectiveness of the algorithms triggered the implementation process of a real case study. Therefore, the application is conducted in a well-known retailer's DC. As the first case, the storage assignment of incoming products to a predetermined storage area is implemented. A class based storage policy is developed and applied to the case. The result of this case shows that the picker movements in the storage area are minimized. Secondly, a computational study is implemented to test performances of the proposed algorithms. Total picking time of the picker is decreased substantially according to the results of the proposed algorithms. The results which rely on proposed solution methods for the OBP and related suggestions are favorably received by the company.

6.3 Concluding Remarks

As it can be seen from the constantly evolving technology worldwide, speed and efficiency in the supply chain processes is needed more than ever before. When the studies in the literature are investigated, it is understood that the OBP maintains its importance. Any inefficiency emerges in the warehouse operations will cause the bullwhip effect in the process. Therefore, an effective order picking process is important for both company and customer satisfaction.

6.3.1 Contributions to the Literature

This thesis aims to provide solutions to OBP and similar problems and contribute to the literature. The literature review on the OBP is a comprehensive study

particularly on the latest solution approaches for the problem, and gives insights on solution environments of the studies. The metaheuristic solution approaches developed in this study are evaluated with generated data sets and benchmark instances from literature. According to the results of the computational study, it is seen that developed solution approaches are applicable, competitive and yield good results.

6.3.2 Implications for Practice

In real life situations, the assumptions related with the operations in the picking systems are important issue to concentrate. Time and cost related evaluations are thought to be the core decision factors to measure the performance of the picking system. Therefore, simulation based evaluations may take place in some level of the application process.

6.3.3 Future Research Directions

Research topics in the future may be on systems that can adapt to technological developments and rapidly changing demands. Ergonomics, due date related or sustainability related objectives may be considered in further studies. In this competitive environment, it is a necessity for companies to be able to adapt to changing situations. Therefore, companies that follow technological developments and whose processes are efficient, will be one step ahead of other competitors. In addition, companies that would like to keep customer satisfaction at a high level should continually strive to understand and meet customers' demands. These approaches and efforts are equally important not only for companies but also for all customers and stakeholders who receive services. The expectations of everyone in the process can be stated as the main purpose of these studies.

REFERENCES

- Albareda-Sambola, M., Alonso-Ayuso, A., Molina, E., & De Blas, C. S. (2009). Variable neighborhood search for order batching in a warehouse. *Asia-Pacific Journal of Operational Research*, 26(5), 655-683.
- Amer, Y., Luong, L., Lee, S.-H., & Ashraf, M.A. (2008). Optimizing order fulfillment using design or six sigma and fuzzy logic. *International Journal of Management Science and Engineering Management*, 3(2), 83-99.
- Armstrong, R. D., Cook, W. D., & Saipie, A. L. (1979). Optimal batching in a semi-automated order picking system. *Journal of the operational research society*, 30(8), 711-720.
- Atmaca, E., & Ozturk, A. (2013). Defining order picking policy: A storage assignment model and a simulated annealing solution in AS/RS systems. *Applied Mathematical Modelling*, 37(7), 5069-5079.
- Azadnia, A. H., Taheri, S., Ghadimi, P., Mat Saman, M. Z., & Wong, K. Y. (2013). Order batching in warehouses by minimizing total tardiness: A hybrid approach of weighted association rule mining and genetic algorithms. *The Scientific World Journal*, Article ID 246578.
- Baker, P., & Canessa, M. (2009). Warehouse design: A structured approach. *European Journal of Operational Research*, 193(2), 425-436.
- Bartholdi III, J. J., & Hackman, S. T. (2014). *Warehouse & distribution science: release 0.96*. The Supply Chain and Logistics Institute, School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA. Retrieved January 15, 2016, from <http://www2.isye.gatech.edu/~jjb/wh/book/editions/wh-sci-0.96.pdf>.

- Bozer, Y. A., & Kile, J. W. (2008). Order batching in walk-and-pick order picking systems. *International Journal of Production Research*, 46(7), 1887-1909.
- Brynzér, H., & Johansson, M. I. (1995). Design and performance of kitting and order picking systems. *International Journal of Production Economics*, 41(1-3), 115-125.
- Burke, E. K., & Kendall, G. (2005). *Search methodologies. Introductory tutorials in optimization and decision support techniques*. New York: Springer Science+ Business Media.
- Cergibozan, Ç. (2013). *Meta-heuristic solution approaches for traveling salesman and traveling repairman problems*. M.Sc Thesis, Dokuz Eylül University, İzmir.
- Cergibozan, Ç., & Tasan, A. S. (2019). Order batching operations: an overview of classification, solution techniques, and future research. *Journal of Intelligent Manufacturing*, 30, 335-349.
- Chen, C.-T., & Huang, S.-F. (2006). Order-fulfillment ability analysis in the supply-chain system with fuzzy operation times. *International Journal of Production Economics*, 101, 185-193.
- Chen, F., Wang, H., Xie, Y., & Qi, C. (2016). An ACO-based online routing method for multiple order pickers with congestion consideration in warehouse. *Journal of Intelligent Manufacturing*, 27(2), 389-408.
- Chen, M.-C., Huang, C.-L., Chen, K.-Y., & Wu, H.-P. (2005). Aggregation of orders in distribution centers using data mining. *Expert Systems with Applications*, 28(3), 453-460.
- Chen, M.-C., & Wu, H.-P. (2005). An association-based clustering approach to order batching considering customer demand patterns. *Omega*, 33(4), 333-343.

- Chen, R.-C., Shieh, C.-H., Chan, K.-T., Chiu, S.-Y., Fan, J.-Y., Chang, Y.-T., & Ma, N.-J. (2013). A systematic approach to order fulfillment of on-demand delivery service for bento industry. *Procedia Computer Science*, 17, 96-103.
- Chen, T.-L., Cheng, C.-Y., Chen, Y.-Y., & Chan, L.-K. (2015). An efficient hybrid algorithm for integrated order batching, sequencing and routing problem. *International Journal of Production Economics*, 159, 158-167.
- Cheng, C.-Y., Chen, Y.-Y., Chen, T.-L., & Yoo, J.J.-W. (2015). Using a hybrid approach based on the particle swarm optimization and ant colony optimization to solve a joint order batching and picker routing problem. *International Journal of Production Economics*, 170, 805-814.
- Chirici, L., & Wang, K.-S. (2014). Tackling the storage problem through genetic algorithms. *Advances in Manufacturing*, 2(3), 203-211.
- Clarke, G., & Wright, J. W. (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, 12(4), 568-581.
- Coello, C. C., Dehuri, S., & Ghosh, S. (Eds.). (2009). *Swarm intelligence for multi-objective problems in data mining*. Berlin Heidelberg: Springer-Verlag.
- Cormier, G. (2005). Operational research methods for efficient warehousing. In A. Langevin & D. Riopel (Eds.), *Logistics systems: Design and optimization*, 93-122. New York: Springer.
- Croxton, K. (2003). The Order fulfillment process. *The International Journal of Logistics Management*, 14(1), 19-32.

- Davis-Sramek, B., Germain, R., & Stank, T. P. (2010). The impact of order fulfillment service on retailer merchandising decisions in the consumer durables industry. *Journal of Business Logistics*, 31(2), 215-230.
- Davis-Sramek, B., Mentzer, J. T., & Stank, T. P. (2008). Creating consumer durable retailer customer loyalty through order fulfillment service operations. *Journal of Operations Management*, 26(6), 781-797.
- de Koster, R., Le-Duc, T., & Roodbergen, K. (2007). Design and control of warehouse order picking: A literature review. *European Journal of Operational Research*, 182(2), 481-501.
- Dekker, R., de Koster, R., Roodbergen, K. J., & van Kalleveen, H. (2004). Improving order-picking response time at Ankor's warehouse. *Interfaces*, 34(4), 303-313.
- Doerner, K. F., Gendreau, M., Greistorfer, P., Gutjahr, W., Hartl, R. F., & Reimann, M. (Eds.). (2007). *Metaheuristics: Progress in complex systems optimization*, 39. US: Springer.
- Elsayed, E. A., Lee, M.-K., Kim, S., & Scherer, E. (1993). Sequencing and batching procedures for minimizing earliness and tardiness penalty of order retrievals. *International Journal of Production Research*, 31(3), 727-738.
- Elsayed, E. A., & Unal, O. I. (1989). Order batching algorithms and travel-time estimation for automated storage/retrieval systems. *International Journal of Production Research*, 27(7), 1097-1114.
- Franzke, T., Grosse, E. H., Glock, C. H., & Elbert, R. (2017). An investigation of the effects of storage assignment and picker routing on the occurrence of picker blocking in manual picker-to-parts warehouses. *The International Journal of Logistics Management*, 28(3), 841-863.

- Gademann, N., & Van de Velde, S. (2005). Order batching to minimize total travel time in a parallel-aisle warehouse. *IIE Transactions*, 37(1), 63-75.
- Gademann, N., Van Den Berg, J. P., & Van Der Hoff, H. H. (2001). An order batching algorithm for wave picking in a parallel-aisle warehouse. *IIE Transactions*, 33(5), 385-398.
- Gen, M., & Cheng, R. (2000). *Genetic algorithms and engineering design*. New York: John Wiley & Sons.
- Gen, M., Cheng, R. & Lin, L. (2008). *Network models and optimization: Multiple objective genetic algorithm approach*. London: Springer.
- Gibson, D. R., & Sharp, G. P. (1992). Order batching procedures. *European Journal of Operational Research*, 58(1), 57-67.
- Goldbarg, E. F. G., Goldbarg, M. C., & De Souza, G. R. (2008). Particle swarm optimization algorithm for the traveling salesman problem. In F. Greco (Ed.), *Travelling salesman problem*, 75-96. Vienna, Austria: InTech.
- Grosse, E. H., & Glock, C. H. (2015). The effect of worker learning on manual order picking processes. *International Journal of Production Economics*, 170, 882-890.
- Gu, J., Goetschalckx, M., & McGinnis, L. F. (2007). Research on warehouse operation: A comprehensive review. *European Journal of Operational Research*, 177(1), 1-21.
- Gue, K. R., Ivanović, G., & Meller, R. D. (2012). A unit-load warehouse with multiple pickup and deposit points and non-traditional aisles. *Transportation Research Part E: Logistics and Transportation Review*, 48(4), 795-806.

- Hanne, T., & Dornberger, R. (2017). *Computational intelligence in logistics and supply chain management*. Switzerland: Springer.
- Hao, Z. F., Wang, Z. G., & Huang, H. (2007). A particle swarm optimization algorithm with crossover operator. In *International Conference on Machine Learning and Cybernetics*, 1036-1040. IEEE.
- Henn, S. (2012a). Algorithms for on-line order batching in an order picking warehouse. *Computers & Operations Research*, 39(11), 2549-2563.
- Henn, S. (2012b). Order batching and sequencing for the minimization of the total tardiness in picker-to-part warehouses. *Flexible Services and Manufacturing Journal*, 27(1), 86-114.
- Henn, S., Koch, S., Doerner, K.F., Strauss, C., & Wäscher, G. (2010). Metaheuristics for the order batching problem in manual order picking systems. *Business Research*, 3(1), 82-105.
- Henn, S., Koch, S., & Wäscher, G. (2012). Order batching in order picking warehouses: A survey of solution approaches. In R. Manzini (Ed.), *Warehousing in the global supply chain: Advanced models, tools and applications for storage systems*, 105-137. London: Springer.
- Henn, S., & Schmid, V. (2013). Metaheuristics for order batching and sequencing in manual order picking systems. *Computers & Industrial Engineering*, 66(2), 338-351.
- Henn, S., & Wäscher, G. (2012). Tabu search heuristics for the order batching problem in manual order picking systems. *European Journal of Operational Research*, 222(3), 484-494.

- Ho, Y.-C., Su, T.-S., & Shi, Z.-B. (2008). Order-batching methods for an order-picking warehouse with two cross aisles. *Computers & Industrial Engineering*, 55(2), 321-347.
- Ho, Y.-C., & Tseng, Y.-Y. (2006). A study on order-batching methods of order-picking in a distribution centre with two cross-aisles. *International Journal of Production Research*, 44(17), 3391-3417.
- Holland, J. H. (1975). *Adaptation in natural and artificial systems*. Michigan: University of Michigan Press.
- Hong, S., Johnson, A. L. & Peters, B. A. (2012). Large-scale order batching in parallel-aisle picking systems. *IIE Transactions*, 44(2), 88-106.
- Hong, S., Johnson, A. L. & Peters, B. A. (2015). Quantifying picker blocking in a bucket brigade order picking system. *International Journal of Production Economics*, 170, 862-873.
- Hong, S., Johnson, A. L., & Peters, B. A. (2016). Order batching in a bucket brigade order picking system considering picker blocking. *Flexible Services and Manufacturing Journal*, 28(3), 425-441.
- Hong, S., & Kim, Y. (2017). A route-selecting order batching model with the S-shape routes in a parallel-aisle order picking system. *European Journal of Operational Research*, 257(1), 185-196.
- Hsieh, L.-F., & Fan, C.-Y. (2010). A new SOM batching heuristic for order picking systems. In *IEEE 17th International Conference on Industrial Engineering and Engineering Management (IE&EM 2010)*, 1478-1481. IEEE.

- Hsieh, L.-F., & Huang, Y.-C. (2011). New batch construction heuristics to optimise the performance of order picking systems. *International Journal of Production Economics*, 131(2), 618-630.
- Hsu, C.-M., Chen, K.-Y., & Chen, M.-C. (2005). Batching orders in warehouses by minimizing travel distance with genetic algorithms. *Computers in Industry*, 56(2), 169-178.
- Hu, X., Eberhart, R. C., & Shi, Y. (2003). Swarm intelligence for permutation optimization: a case study of n-queens problem. In *Swarm Intelligence Symposium, SIS2003. Proceedings of the 2003 IEEE*, 243-246.
- Hwang, H., & Kim, D. G. (2005). Order-batching heuristics based on cluster analysis in a low-level picker-to-part warehousing system. *International Journal of Production Research*, 43(17), 3657-3670.
- Ivanov, D., Tsipoulanidis, A., & Schönberger, J. (2017). Global supply chain and operations management: *A decision-oriented introduction to the creation of value*. Switzerland: Springer.
- Johnson, D. S., & McGeoch, L. A. (1997). The traveling salesman problem: A case study in local optimization. In E. H. L. Aarts & J. K. Lenstra, (Eds.), *Local search in combinatorial optimization*, 215-310. London: John Wiley & Sons.
- Kennedy, J., & Eberhart, R. C. (1995). Particle swarm optimization. In *Proceedings of the 1995 IEEE international conference on neural networks (Perth, Australia)*, 1942-1948. New Jersey: IEEE Service Center.
- Koch, S., & Wäscher, G. (2016). A grouping genetic algorithm for the Order Batching Problem in distribution warehouses. *Journal of Business Economics*, 86(1-2), 131-153.

- Kulak, O., Sahin, Y., & Taner, M. E. (2012). Joint order batching and picker routing in single and multiple-cross-aisle warehouses using cluster-based tabu search algorithms. *Flexible Services and Manufacturing Journal*, 24(1), 52-80.
- Lam, C. H., Choy, K. L., Ho, G. T., & Lee, C. K. M. (2014). An order-picking operations system for managing the batching activities in a warehouse. *International Journal of Systems Science*, 45(6), 1283-1295.
- Le-Duc, T., & de Koster, R. M. (2007). Travel time estimation and order batching in a 2-block warehouse. *European Journal of Operational Research*, 176(1), 374-388.
- Li, J., Huang, R., & Dai, J. B. (2017). Joint optimisation of order batching and picker routing in the online retailer's warehouse in China. *International Journal of Production Research*, 55(2), 447-461.
- Lin, C. C., Kang, J. R., Hou, C. C., & Cheng, C. Y. (2016). Joint order batching and picker Manhattan routing problem. *Computers & Industrial Engineering*, 95, 164-174.
- Lin, F.-R., & Shaw, M.J. (1998). Reengineering the order fulfillment process in supply chain networks. *The International Journal of Flexible Manufacturing Systems*, 10, 197-229.
- Lin, J. T., Hong, I.-H., Wu, C.-H., & Wang, K.-S. (2010). A model for batch available-to-promise in order fulfillment processes for TFT-LCD production chains. *Computers & Industrial Engineering*, 59, 720-729.
- Machado, T.R., & Lopes, H.S. (2005). A hybrid particle swarm optimization model for the traveling salesman problem. In B. Ribeiro, R.F. Albrecht, & A. Dobnikar (eds.): *Adaptive and natural computing algorithms*, 255-258. Wien: Springer-Verlag.

- Manzini, R. (Ed.) (2012). *Warehousing in the global supply chain: Advanced models, tools and applications for storage systems*. London: Springer-Verlag.
- Matusiak, M., de Koster, R., Kroon, L., & Saarinen, J. (2014). A fast simulated annealing method for batching precedence-constrained customer orders in a warehouse. *European Journal of Operational Research*, 236(3), 968-977.
- Matusiak, M., de Koster, R., & Saarinen, J. (2017). Utilizing individual picker skills to improve order batching in a warehouse. *European Journal of Operational Research*, 263(3), 888-899.
- Menéndez, B., Bustillo, M., Pardo, E. G., & Duarte, A. (2017a). General variable neighborhood search for the order batching and sequencing problem. *European Journal of Operational Research*, 263(1), 82-93.
- Menéndez, B., Pardo, E. G., Sánchez-Oro, J., & Duarte, A. (2017b). Parallel variable neighborhood search for the min-max order batching problem. *International Transactions in Operational Research*, 24(3), 635-662.
- Merkle, D., & Middendorf, M. (2005). Swarm intelligence. In E. K. Burke, & G. Kendall, (Eds.), *Search Methodologies - Introductory tutorials in optimization and decision support techniques*, 401-435. New York: Springer.
- Muter, İ., & Öncan, T. (2015). An exact solution approach for the order batching problem. *IIE Transactions*, 47(7), 728-738.
- Mutingi, M., & Mbohwa, C. (2017). Optimizing order batching in order picking systems: Hybrid grouping genetic algorithm. In *Grouping Genetic Algorithms*, 121-140. Cham: Springer.

- Ming-Huang Chiang, D., Lin, C. P., & Chen, M. C. (2014). Data mining based storage assignment heuristics for travel distance reduction. *Expert Systems*, 31(1), 81-90.
- Nicolas, L., Yannick, F., & Ramzi, H. (2018). Order batching in an automated warehouse with several vertical lift modules: Optimization and experiments with real data. *European Journal of Operational Research*, 267(3), 958-976.
- Ong, J. O., & Joseph, D. T. (2014). A review of order picking improvement methods. *J@Ti (Teknik Industri)*, 9(3), 135-138.
- Opticom Project (2018). Retrieved February 2, 2018, from <http://www.opticom.es/obp/>.
- Öncan, T. (2013). A genetic algorithm for the order batching problem in low-level picker-to-part warehouse systems. In *Proceedings of the International MultiConference of Engineers and Computer Scientists (IMECS 2013)*, 19-24. Hong Kong.
- Öncan., T. (2015). MILP formulations and an iterated local search algorithm with tabu thresholding for the order batching problem. *European Journal of Operational Research*, 243(1), 142-155.
- Pan, C. H., & Liu, S. Y. (1995). A comparative study of order batching algorithms. *Omega*, 23(6), 691-700.
- Pan, J. C. H., Wu, M. H., & Chang, W. L. (2014). A travel time estimation model for a high-level picker-to-part system with class-based storage policies. *European Journal of Operational Research*, 237(3), 1054-1066.

- Pant, M., Thangaraj, R., & Abraham, A. (2007). A new pso algorithm with crossover operator for global optimization problems. In E. Corchado et al., (Eds.), *Innovations in Hybrid Intelligent Systems*, 215-222. Berlin Heidelberg: Springer.
- Parikh, P. J., & Meller, R. D. (2008). Selecting between batch and zone order picking strategies in a distribution center. *Transportation Research Part E: Logistics and Transportation Review*, 44(5), 696-719.
- Petersen, C. G. (1997). An evaluation of order picking routeing policies. *International Journal of Operations & Production Management*, 17(11), 1098-1111.
- Petersen, C. G., & Aase G. (2004). A comparison of picking, storage, and routing policies in manual order picking. *International Journal of Production Economics*, 92(1), 11-19.
- Petersen, C. G., Aase, G. R., & Heiser, D. R. (2004). Improving order-picking performance through the implementation of class-based storage. *International Journal of Physical Distribution & Logistics Management*, 34(7), 534-544.
- Petersen, C. G., & Schmenner, R. W. (1999). An evaluation of routing and volume-based storage policies in an order picking operation. *Decision Sciences*, 30, 481-501.
- Pil, F. K., & Holweg, M. (2004). Linking product variety to order-fulfillment strategies. *Interfaces*, 34(5), 394-403.
- Pohl, L. M., Meller, R. D., & Gue, K. R. (2009). An analysis of dual-command operations in common warehouse designs. *Transportation Research Part E: Logistics and Transportation Review*, 45(3), 367-379.

- Raidl, G. R., Puchinger, J., & Blum, C. (2010). Metaheuristic hybrids. In *Handbook of metaheuristics*, 469-496. Boston, MA: Springer.
- Ratliff, H. D., & Rosenthal, A. S. (1983). Order-picking in a rectangular warehouse: a solvable case of the traveling salesman problem. *Operations Research*, 31(3), 507-521.
- Roodbergen, K. J. (2018). *Interactive warehouse*. Retrieved October 9, 2018, from <http://www.roodbergen.com/warehouse/index.php>.
- Roodbergen, K. J., & Vis, I. F. A. (2009). A survey of literature on automated storage and retrieval systems. *European Journal of Operational Research*, 194(2), 343-362.
- Roodbergen, K. J., & de Koster, R. (2001). Routing methods for warehouses with multiple cross aisles. *International Journal of Production Research*, 39(9), 1865-1883.
- Ropke, S., & Pisinger, D. (2006). An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science*, 40 (4), 455-472.
- Rosenwein, M. B. (1996). A comparison of heuristics for the problem of batching orders for warehouse selection. *International Journal of Production Research*, 34(3), 657-664.
- Rouwenhorst, B., Reuter, B., Stockrahm, V., Van Houtum, G. J., Mantel, R. J., & Zijm, W. H. M. (2000). Warehouse design and control: Framework and literature review. *European Journal of Operational Research*, 122(3), 515-533.

- Ruben, R. A., & Jacobs, F. R. (1999). Batch construction heuristics and storage assignment strategies for walk/ride and pick systems. *Management Science*, 45(4), 575-596.
- Sastry, K., Goldberg, D., & Kendall, G. (2005). Genetic algorithms. In *Search methodologies*, 97-125. Boston, MA: Springer.
- Schleyer, M., & Gue, K. (2012). Throughput time distribution analysis for a one-block warehouse. *Transportation Research Part E: Logistics and Transportation Review*, 48(3), 652-666.
- Scholz, A., Schubert, D., & Wäscher, G. (2017). Order picking with multiple pickers and due dates–Simultaneous solution of order batching, batch assignment and sequencing, and picker routing problems. *European Journal of Operational Research*, 263(2), 461-478.
- Scholz, A., & Wäscher, G. (2017). Order batching and picker routing in manual order picking systems: the benefits of integrated routing. *Central European Journal of Operations Research*, 25(2), 491-520.
- Settles, M., & Soule, T. (2005). Breeding swarms: a GA/PSO hybrid. In *Proceedings of the 7th annual conference on genetic and evolutionary computation*, 161-168. ACM.
- Şahin, Y., & Kulak, O. (2013). Depo operasyonlarının planlanması için genetik algoritma esaslı modeller [Genetic algorithm based approaches for planning warehouse operations]. *International Journal of Alanya Faculty of Business*, 5(3), 141-153. Retrieved January 15, 2016, from <http://alanyadergi.akdeniz.edu.tr/dinamik/164/446.pdf>.
- Talbi, E. G. (Ed.). (2013). *Hybrid metaheuristics*. Berlin Heidelberg: Springer-Verlag.

- Tang, L. C., & Chew, E. P. (1997). Order picking systems: batching and storage assignment strategies. *Computers & Industrial Engineering*, 33(3-4), 817-820.
- Tang, L., Wang, G., Liu, J., & Liu, J. (2011). A combination of Lagrangian relaxation and column generation for order batching in steelmaking and continuous-casting production. *Naval Research Logistics (NRL)*, 58(4), 370-388.
- Ten Hompel, M., & Schmidt, T. (2007). *Warehouse management: Automation and organisation of warehouse and order picking systems*. Berlin Heidelberg: Springer-Verlag.
- Theys, C., Bräysy, O., Dullaert, W., & Raa, B. (2010). Using a TSP heuristic for routing order pickers in warehouses. *European Journal of Operational Research*, 200(3), 755-763.
- Tompkins, J. A., & Smith, J. D. (1998). *The warehouse management handbook*. Raleigh, NC: Tompkins.
- Tsai, C.-Y., Liou, J. J. H., & Huang, T.-M. (2008). Using a multiple-GA method to solve the batch picking problem: considering travel distance and order due time. *International Journal of Production Research*, 46(22), 6533-6555.
- Valle, C. A., Beasley, J. E., & da Cunha, A. S. (2017). Optimally solving the joint order batching and picker routing problem. *European Journal of Operational Research*, 262(3), 817-834.
- Van den Berg, J. P., & Zijm, W. H. M. (1999). Models for warehouse management: Classification and examples. *International Journal of Production Economics*, 59(1-3), 519-528.

- Van Gils, T., Ramaekers, K., Braekers, K., Depaire, B., & Caris, A. (2018). Increasing order picking efficiency by integrating storage, batching, zone picking, and routing policy decisions. *International Journal of Production Economics*, 197, 243-261.
- Van Nieuwenhuyse, I., & de Koster, R., (2009). Evaluating order throughput time in 2-block warehouses with time window batching. *International Journal of Production Economics*, 121(2), 654-664.
- Van Nieuwenhuyse, I., de Koster, R., & Colpaert, J. (2007). *Order batching in multi-server pick-and-sort warehouses (Research Report KBI 0731)*. Retrieved January 15, 2016, from https://lirias.kuleuven.be/bitstream/123456789/175475/1/KBI_0731_research.
- Wang, K.-P., Huang, L., Zhou, C.-G., & Pang, W. (2003). Particle swarm optimization for traveling salesman problem. In *Proceedings of the Second International Conference on Machine Learning and Cybernetics, Xi'an*.
- Won, J., & Olafsson, S. (2005). Joint order batching and order picking in warehouse operations. *International Journal of Production Research*, 43(7), 1427-1442.
- Xiang, X., Liu, C., & Miao, L. (2018). Storage assignment and order batching problem in Kiva mobile fulfilment system. *Engineering Optimization*, 50(11), 1941-1962.
- Yan, T., Rabinovich, E., Dooley, K., & Evers, P.T. (2010). Managing backlog variation in order fulfillment: The case of Internet retailers. *International Journal of Production Economics*, 128, 261-268.
- Yu, M., & de Koster, R. (2009). The impact of order batching and picking area zoning on order picking system performance. *European Journal of Operational Research*, 198(2), 480-490.

- Zäpfel, G., Braune, R., & Bögl M. (2010). *Metaheuristic search concepts: A tutorial with applications to production and logistics*. Berlin Heidelberg: Springer-Verlag.
- Zhang, J., Wang, X., Chan, F. T., & Ruan, J. (2017). On-line order batching and sequencing problem with multiple pickers: A hybrid rule-based algorithm. *Applied Mathematical Modelling*, 45, 271-284.
- Zhang, J., Wang, X., & Huang, K. (2016). Integrated on-line scheduling of order batching and delivery under B2C e-commerce. *Computers & Industrial Engineering*, 94, 280-289.
- Žulj, I., Kramer, S., & Schneider, M. (2018). A hybrid of adaptive large neighborhood search and tabu search for the order-batching problem. *European Journal of Operational Research*, 264(2), 653-664.

APPENDICES

APPENDIX-1: Problem Data-1

Table A.1 Problem Data-1

item	order number									
	1	2	3	4	5	6	7	8	9	10
1	0	0	0	0	0	0	0	0	0	0
2	0	1	0	0	0	1	0	0	0	0
3	0	0	0	0	1	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0	1
5	0	0	0	0	0	0	1	0	1	0
6	0	0	1	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	1	0	0	0	1	1
9	0	0	0	1	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0
11	1	0	0	0	0	0	0	0	0	0
12	0	0	0	1	0	0	0	0	1	0
13	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	1	0	0	0	1	0
15	0	0	0	0	0	0	0	1	0	0
16	0	0	1	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	1	1	0	0	0	0
19	0	0	0	0	0	0	0	0	0	1
20	0	1	0	0	0	0	0	0	0	0
21	0	0	0	1	0	0	0	1	0	0
22	0	0	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0
24	0	0	0	0	1	0	0	0	0	0
25	1	0	1	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	0	0
27	0	0	0	0	1	0	0	0	0	0
28	0	1	0	0	0	0	1	1	0	0
29	0	0	0	0	0	0	0	1	0	0
30	0	0	0	0	0	0	0	0	1	0
31	0	0	0	0	0	1	0	0	0	0
32	0	0	0	0	0	0	0	0	0	1
33	0	0	0	0	1	0	0	0	0	0
34	1	0	0	0	0	0	0	0	0	1
35	0	0	1	0	0	0	0	0	0	0
36	0	0	0	0	1	0	0	0	1	0
37	1	0	0	0	0	0	0	0	1	0
38	0	0	0	0	0	0	1	0	0	1
39	0	0	0	0	0	0	0	0	0	0
40	0	0	0	0	0	0	0	1	0	0
41	0	1	0	0	0	0	0	0	0	0
42	0	0	0	0	0	0	0	0	0	0
43	0	0	0	0	1	0	1	1	0	0
44	0	0	0	0	0	0	0	0	0	0
45	0	0	1	0	0	0	0	0	0	0
46	0	0	0	1	0	1	0	0	0	0
47	0	0	0	0	0	0	0	0	0	0
48	1	0	1	0	0	0	0	0	1	0
49	0	0	0	0	0	0	0	0	0	1
50	0	1	0	0	0	0	0	0	0	0
51	0	0	0	0	0	0	1	0	0	0
52	1	0	0	0	0	0	0	0	0	0
53	1	0	0	0	0	0	0	0	0	0
54	0	0	0	0	1	0	0	0	0	0
55	0	0	0	0	0	0	0	0	1	0
56	0	0	0	1	0	0	0	0	0	0
57	0	0	0	0	0	0	0	0	0	0
58	0	0	0	0	0	0	1	0	0	0
59	1	0	0	0	0	0	0	0	0	0
60	0	0	0	0	0	0	0	0	0	0
61	0	0	0	0	0	0	0	0	0	1
62	0	0	0	0	0	0	0	0	0	0
63	1	0	0	0	0	1	0	1	0	0
64	0	0	0	0	0	0	0	0	0	0
65	1	0	0	0	0	0	0	0	1	0
66	0	0	0	0	0	0	1	0	0	0
67	0	0	0	0	0	0	0	0	0	0
68	0	0	0	1	0	0	0	0	0	0
69	0	0	0	0	0	0	0	0	0	0
70	0	0	0	0	0	0	0	0	0	0
71	0	1	0	1	0	0	0	0	0	0
72	0	0	0	0	0	0	0	0	1	0
73	0	1	0	0	0	0	0	0	0	0
74	0	0	0	0	0	0	0	0	0	0
75	0	0	0	0	0	0	0	0	0	0
76	0	0	0	0	0	0	0	0	0	0
77	0	0	0	0	0	0	0	0	0	0
78	0	0	0	1	0	0	0	1	0	0
79	0	0	0	0	0	0	0	0	0	0
80	1	0	0	0	0	0	0	0	1	0
81	0	0	0	0	0	1	0	0	0	0
82	0	0	0	1	0	0	0	0	0	0
83	0	0	0	0	1	0	0	0	0	0
84	0	0	0	0	0	1	0	0	0	0
85	0	0	0	1	0	0	0	0	0	0
86	0	0	0	0	0	0	0	0	0	0
87	0	1	0	0	0	0	0	1	0	0
88	0	0	0	0	0	0	0	1	0	0
89	0	0	0	0	0	1	0	0	0	0
90	0	0	0	0	0	0	0	0	0	0
91	0	0	0	0	0	0	0	0	0	0
92	0	1	0	0	0	0	0	0	0	0
93	0	0	0	0	0	0	0	0	0	0
94	0	0	0	1	0	0	0	0	0	0
95	0	0	0	0	0	0	0	0	0	1
96	0	0	0	0	1	0	0	0	0	0
97	0	0	0	0	0	0	0	0	0	0
98	0	0	0	0	0	0	0	0	1	0
99	0	1	0	0	0	0	0	0	0	0
100	0	0	0	0	1	0	0	0	1	0

APPENDIX-2: Problem Data-2

Table A.2 Problem Data-2

	order number																																																	
item	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
2	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
4	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0
5	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	
7	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0
8	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	1	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	0	0	0	0	0	1	0	
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	
12	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	1	0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	1	1	0	0	0	1	0	1	0	0	0	0	1	0	0	0	0	
13	0	0	0	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0	
14	0	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1	1	0	1	0	0	1	0	0	0	0	1	0	0	0	1	0	1	0	0	1	0	0	0	
15	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	
16	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	
17	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	
18	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	
19	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	1	1	0	0	1	0	0	1	0	0	1	0	1	0	0	0	1	0	0	1	0	1	0	1	0
20	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	
21	0	0	1	0	0	0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
22	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	1	0	0	0	0	1	0	
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
25	0	0	0	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	0	0	
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
28	1	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
29	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0
30	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	
31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
32	1	0	0	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
33	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
35	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	
36	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	1	0	1	0	1
37	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
38	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	1	0	1	0	
39	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
40	0	1	0	1	0	0	0	0	0	0	0																																							

APPENDIX-2

Table A.2 continues

Table A.2 continues

	order number																																																		
item	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	
51	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
52	1	0	0	0	0	0	1	0	0	1	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
53	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0
54	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	
55	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	
56	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0	0	0	0	1	0	0	0	1	1	1	0	0	0	0	0		
57	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0		
58	0	0	0	0	0	0	1	1	0	0	0	0	0	1	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
59	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	1	0	0	0	0	
60	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
61	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	
62	0	0	0	0	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
63	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	
64	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	
65	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0	1	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	
66	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0		
67	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0		
68	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0		
69	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
70	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	
71	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0		
72	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
73	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0		
74	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
75	0	0	0	0	0	1	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0		
76	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0		
77	1	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1		
78	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0		
79	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0		
80	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	
81	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
82	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
83	0	0	0	0	0	0	1	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	
84	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0		
85	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	
86	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
87	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0		
88	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
89	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0			

APPENDIX-3: Problem Data-3

Table A.3 Problem Data-3

[illegible]

APPENDIX-3

Table A.3 continues

Table A.3 continues

item	order number																																																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50		
51	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0		
52	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0		
53	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
54	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	1	0	1	0	1	0
55	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
56	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1
57	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
58	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0
59	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
60	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
61	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
62	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
63	0	0	0	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
64	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
65	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
66	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
67	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	0	1	0	1	1	0
68	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
69	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	1	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0
70	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0	0
71	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	1	1	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	1	0	0
72	0	0	0	0	0	1	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
73	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0
74	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1	0	0	0	0	0
75	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
76	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1	1	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0
77	0	0	1	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
78	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
79	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
80	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
81	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
82	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	1	0	1	0	0	0	0	0	0	0	0	1	0	0	0	1	1	1	0	0	0	0	0
83	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
84	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
85	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1	0	0	0	0	0	0	0	0
86	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	1	0	0	1	1	1	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1
87	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0
88	0	0	0	0	0	0	0	0	0	0	0	0	0	1																																						

APPENDIX-3

Table A.3 continues

Table A.3 continues

[illegible]

APPENDIX-3

Table A.3 continues

Table A.3 continues

[illegible]