YAŞAR UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

MASTER THESIS

A MULTI-COMPARTMENT VEHICLE ROUTING PROBLEM FOR INCOMPATIBLE PRODUCTS

Bahar TAŞAR

Thesis Co-Advisor: Assoc. Prof. Dr. Deniz TÜRSEL ELİİYİ
Thesis Co-Advisor: Prof. Dr. Levent KANDİLLER

Department of Industrial Engineering

Presentation Date: 29.04.2016

Bornova-İZMİR
2016

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of master of science.

Assoc. Prof. Dr. Deniz TÜRSEL ELİİYİ (Supervisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of master of science.

Prof. Dr. Levent KANDİLLER (Supervisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of master of science.

Assoc. Prof. Dr. Pınar MIZRAK ÖZFIRAT

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of master of science.

Assist. Prof. Dr. Adalet ÖNER

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of master of science.

Assist. Prof. Dr. Uğur ELİİYİ

Prof. Dr. Cüneyt GÜZELİŞ
Director of the Graduate School

ABSTRACT

A MULTI-COMPARTMENT VEHICLE ROUTING PROBLEM

FOR INCOMPATIBLE PRODUCTS

TAŞAR, Bahar
MSc in Industrial Engineering
Co-Supervisor: Assoc. Prof. Dr. Deniz TÜRSEL ELİİYİ
Co-Supervisor: Prof. Dr. Levent KANDİLLER
April 2016, 88 pages

This thesis focuses on a special category of distribution problems for the case of incompatible products. To satisfy different type of demands with minimum logistics costs, incompatible products are carried on the same vehicle but in different compartments. The scope of this study is to explore new mathematical models for the corresponding Multi-Compartment Vehicle Routing Problem (MCVRP) and its variants. While there exists a vast amount of Vehicle Routing Problem (VRP) literature covering several variants, the MCVRP is still open for research. Our study is motivated by a real life instance of a livestock feed distribution system, where each livestock farm demands one type of feed from a single depot. We consider some variants of the MCVRP, as multiple trips of vehicles and the splitting of demand. A taxonomic framework for VRP literature is also suggested. A general mathematical model, and its variants are formulated. A computational experiment is designed for testing the performance of the developed models. Exact solution schemes are evaluated for small sized problem instances, whereas heuristic algorithms are proposed for larger instances. Our results indicate that the proposed methodology is applicable to real life logistics problems such as food, fuel and other chemical distribution.

Keywords: Multi compartment vehicle routing problem, split delivery, multi-trip, mathematical modeling, heuristics.

ÖZET

KARIŞAMAYAN ÜRÜNLER İÇİN ÇOK KOMPARTIMANLI ARAÇ ROTALAMA PROBLEMİ

Bahar TAŞAR
Yüksek Lisans, Endüstri Mühendisliği Bölümü
Tez Danışmanı: Doç. Dr. Deniz TÜRSEL ELİİYİ
Tez Danışmanı: Prof. Dr. Levent KANDİLLER
April 2016, 88 sayfa

Bu tez dağıtım problemlerinin özel bir kategorisi olan karışamayan ürünlere odaklanmıştır. En az lojistik maliyeti ile farklı tip talepleri karşılamak için, karışamayan ürünler aynı araçta fakat farklı kompartımanlarda taşınmaktadır. Tez kapsamında Çok Kompartımanlı Araç Rotalama Problemi (ÇKARP) ve varyantları için yeni matematiksel modeller önerilmektedir. Birçok varyantı olan Araç Rotalama Problemi (ARP)'nin çok geniş bir literatürü mevcut olduğu halde, ÇKARP alanı hala araştırmaya açıktır. Tez kapsamında ARP literatürü için bir taksonomik çerçeve önerilmiştir. Çalışmamız bir gerçek hayat problemi olan, her çiftliğin tek bir depodan tek tip yem talep ettiği bir canlı hayvan yem dağıtım sisteminden yola çıkarak ortaya çıkmıştır ve ÇKARP'nin varyantları olan çoklu seferleri ve bölünmüş dağıtımları göz önüne almaktadır. Genel bir matematiksel model ve varyantları formüle edilmiştir. Geliştirilen matematiksel modellerin performanslarını test etmek için sayısal bir deney tasarlanmıştır. Büyük boyutlu problemler için sezgisel yöntemler önerilirken, kesin çözüm planları küçük boyutlu problem örnekleri için değerlendirilmiştir. Sonuçlarımız geliştirilen yöntemlerin gıda, yakıt ve diğer kimyasal dağıtım gibi gerçek hayat problemleri için de uygulanabileceğini göstermektedir.

Anahtar sözcükler: Çok kompartımanlı araç rotalama problemi, bölünmüş dağıtım, çoklu sefer, matematiksel modelleme, sezgisel yöntemler.

ACKNOWLEDGEMENTS

I would like to thank to my supervisors Prof. Dr. Levent Kandiller and Assoc. Prof. Dr. Deniz Türsel Elliyi for their support, advices, valuable and important comments on my thesis. Their guidance helped me in all the time of research and writing of this thesis.

I would like to thank to Gündüz İlsever, manager of Çamlı Yem Besicilik, for his help on my thesis study. He gave the inspiration of the thesis subject and he shared all the data from the real life.

I thankfully acknowledge toTUBITAK - Turkish Technological and Scientific Research Institute for supporting me as a scholarship student during my master period, (Program: 2210).

I would also like to sincerely thank my parents, brother and spouse. They were always there supporting me and encouraging me with their best wishes.

Bahar TAŞAR
İzmir, 2016

TEXT OF OATH

I declare and honestly confirm that my study, titled "A Multi-Compartment Vehicle Routing Problem for Incompatible Products" and presented as a Master's Thesis, has been written without applying to any assistance inconsistent with scientific ethics and traditions, that all sources from which I have benefited are listed in the bibliography, and that I have benefited from these sources by means of making references.

TABLE OF CONTENTS

INDEX OF FIGURES

INDEX OF TABLES

INDEX OF SYMBOLS AND ABBREVIATIONS

Abbreviations	Explanations
VRP	Vehicle Routing Problem
CVRP	Capacitated Vehicle Routing Problem
MCVRP	Multi-Compartment Vehicle Routing Problem
SDVRP	Split Delivery Vehicle Routing Problem
TSP	Traveling Salesman Problem
RRT	Record and Record Travel
ABHC	Attribute Based and Hill Climber
RSH	Random Start Heuristic
SSH	Smart Start Heuristic
ALB	Assembly line balancing
STW	Soft time window
HTW	Hard time window
TW’	Without time window
PD	Pickup and delivery
BH	Backhaul
LH	Linehaul
BH&LH	Backhaul and linehaul
SD	Split delivery
SD’	Without split delivery
ST	Single trip
MT	Multi-trip
IF	Identical fleet
HF	Heterogeneous fleet
FC	Fixed capacity
FLC	Flexible capacity
SC	Single compartment

FMC	Fixed multi-compartment
FLMC	Flexible multi-compartment
MSTC	Minimization of total travel cost
MSTT	Minimization of total travel time
MSTL	Minimization of total travel length
MSV	Minimization of total vehicle
MSWT	Minimization of total waiting time
MMTT	Minimization of maximum travel time
MMTL	Minimization of maximum travel length
LBTT	Load balancing of travel time
LBTL	Load balancing of travel length
SP	Single product
MCP	Multi-compatible product
MICP	Multi-incompatible product
STP	Single time period
MTP	Multi-time period

1 INTRODUCTION

Supply chain management involves managing materials and information flows between suppliers, manufacturers, wholesalers/retailers and customers. Within the supply chain network, logistics has a critical role to sustain the continuity of the system flows. That is why industries focus on improving their distribution processes to decrease their costs while satisfying their customer demands.

Vehicle Routing Problem (VRP) arises from the logistics field, and deals with the distribution of goods to customers. It is a generalization of the Traveling Salesman Problem (TSP). In TSP, a vehicle visits customers to satisfy their demands by minimizing the transportation cost. If total customer demand exceeds vehicle capacity, transportation requires more than one vehicle and the problem becomes VRP, which is a combinatorial optimization problem dealing with the assignment of products to vehicles and the routing of the vehicles in such a way that a desired objective is optimized. Since the problem is NP-Hard (Lenstra and Rinnooy Kan, 1981), heuristic methods have been developed to find near-optimal solutions.

A well-studied form of the VRP is the Capacitated Vehicle Routing Problem (CVRP). The vehicles start from a single depot to satisfy the demands of a set of customers and return to the depot. The problem is constrained by vehicle capacity, while meeting customer demand with minimum total distance traveled. The variants of CVRP are obtained by relaxing some constraints. The concept of multi-compartment VRP (MCVRP) includes separated compartments carrying incompatible products for one or more customer demand. The problem aims to meet different types of customer demands in the same vehicle, through seeking minimum route cost.

A real life problem provided a motivation for this thesis work. Çamlı Yem Besicilik Inc. is a leading group of companies in Turkish food chain sector. The company runs a wide supply chain network in the turkey-breeding, organic dairy breeding, aquaculture and manure sector. The feed production occurs in the facilities located in Pınarbaşı.

We first analyzed the current system, and developed solution methodologies to improve operational decisions. The distribution of feeds is provided with special vehicles having several compartments. The fleet is considered to be homogenous. A compartment on a vehicle can carry only one type of feed at a time. The problem is to deliver the turkey feeds from a single depot via 6 vehicles to 60 contracted farms. Each vehicle has 4 compartments with 4 tons capacities. The type of feeds changes with respect to turkey's growing periods. In each growing period, only one type of feed is used. So, each farm demands a single type of feed. Each compartment can carry only one type of feed but different compartments on the same truck can carry different feed types. Moreover, the same type of demands from different farms can be transported in the same compartment. The weekly plan is currently made manually by one dispatcher without taking distances into consideration. The aim is to meet daily demands in a timely manner at minimum cost.

Motivated by the distribution system explained above, we have formulated mathematical models for several variants of the problem, and solved these on a commercial solver. As the problem sizes gets large in practical applications, we have also developed a heuristic for the corresponding MCVRP on hand.

The rest of this thesis is organized as follows. The problem is defined and the related literature is reviewed in Chapter 2. Chapter 3 includes the mathematical model formulation and solution approaches for the studied problem. The computational study is presented in Chapter 4. We finally provide an overall summary of the thesis, and address possible future research directions in Chapter 5.

2 PROBLEM ENVIRONMENT

In VRP, the goods (products) are kept in storage at one or more locations (depots). The distribution of products from the depot to end users (customer) is provided via resources (vehicles). VRP tries to satisfy customer demands from one or more depots by a given set of vehicles and selecting the appropriate routes within a given time period. Main components of the VRP are described below.

The **road network** is described through a graph whose arcs represent the road sections and whose vertices correspond to the road junctions, which includes the depot and customer locations.

The **customers** must be served and their demands must be satisfied in given time period. The characteristics of the customers are listed below:

- Customer's location in the road graph;
- Amount of goods (demand), possibly of different types, which must be delivered or collected;
- Periods of the day (time windows) during which the customer can be served;
- Durations required to deliver or collect the goods at the customer location (loading or unloading times) possibly dependent on the vehicle type;
- Subset of the available vehicles that can be used to serve the customer.

The **depots** are the stations where vehicles load their goods. Transportation of goods is performed by using a fleet of **vehicles** whose composition and size can be fixed, or defined according to the requirements of the customers. The vehicle characteristics are listed below:

- Home depot of the vehicle and the possibility to end service at a depot other than home;
- Capacity of the vehicle, expressed as the maximum weight, volume or number of boxes that the vehicle can load;
- Possible subdivision of the vehicle into compartments, each characterized by its capacity and by the types of goods that can be carried;

- Devices available for loading and unloading operations;
- Costs associated with utilization of the vehicle (per distance unit, etc.) (Toth, 2002).

The **drivers** operate the vehicles by satisfying several constraints required by union contracts and company regulations (Toth, 2002). VRP solution methods have been designed with respect to desired objectives. Although VRP has several objectives, the most commonly used objective is to minimize the total cost, as the distribution of goods is a necessary but non-value added activity. The typical set of objectives used is given below:

- Each movement of the vehicles brings fuel cost, and the total travel cost is minimized by choosing the shortest distance routes.
- In the case that the vehicles are rented with respect to time, total travel time can be minimized.
- If the vehicles have purchasing, driver and depreciation costs, an objective can be the minimization of the number of vehicles used.
- The waiting time of the customers is minimized in some humanitarian logistics problems.
- The maximum tour length or duration can be minimized.
- The tour lengths or times should be balanced for drivers and vehicle depreciations. So the range of minimum and maximum tour measurement unit should be minimized to have load balancing.
- Some constraints can be soft and the penalties of exceeding these constraints' RHSs can be minimized.

These objectives can be separated or combined in the objective function.

In the way of deciding on the objective, there are some restrictions and requirements. The main necessity of the problem is that each customer demand is satisfied. The finite resource is the capacity; the number of vehicles and vehicles' sizes is limited. Therefore, capacity is the main constraint. Tour constraints provide the continuity and the connection of routes and vehicles.

A well-studied form of the VRP is the Capacitated Vehicle Routing Problem (CVRP). The vehicles start from a single depot to satisfy the demands of a set of customers and return to the depot. The problem is constrained by the vehicle capacity, while meeting customer demand with minimum total distance traveled. The variants of CVRP are obtained by relaxing some constraints of the CVRP.

VRP types can be divided to four main classes with respect to the constraints to describe the problem structure such as operational policy, vehicle, product and period as summarized in **Figure 2.1**. Operational policy dimension represents problem characteristics and constraints regarding system configuration and operating principles. In a VRP having time windows, each customer is associated with a time interval for delivery. VRP problems can be classified as ones having no time windows (TW'), soft time windows that can be violated with a penalty (STW) (Bin and Fu, 2003), and hard time windows where no violation is possible (HTW) (Chang et al., 2009). Pickup and Delivery (PD) (Savelsbergh and Sol, 1995, Chuah and Yingling, 2005, Rais et al., 2014), in which vehicles pick up products from certain customers and deliver them to their destinations is another category in the operational policy dimension. Other possibilities of carrying include Backhaul (BH) (Goetschalckx and Jacobs-Blecha, 1989) where vehicles pick up from customers and deliver to the depot, Linehaul (LH) where vehicles deliver demands to customers, and Backhaul and Linehaul (BH&LH) as a hybrid policy. If the demand of a customer can be satisfied by more than one vehicle, VRP with Split Delivery (SD) is considered (Dror and Trudeau, 1989 and 1990, Laporte et al., 1999, Fleischmann et al., 2004, Archetti et al., 2005), otherwise the problem does not allow any split delivery (SD') (Fu, 2002, Gendreau et al., 1996a). Single Trip (ST) VRP and Multi-Trip (MT) VRP (Salhi, 1987) allow single or several trips per vehicle, respectively.

VRP solutions take shape according to their desired objectives. The problem can have various objectives, but most of the objectives in literature focus on the minimization of the total distribution cost or time, as the distribution of goods is a necessary but non-value adding activity. Therefore, minisum type objectives are most commonly used, which include the minimization of the total distribution cost (MSTC), total distribution time (MSTT) or length (MSTL). Minimizing the total waiting time of the customers (MSWT) is also considered to increase customer satisfaction. Minimax type objectives as minimizing the maximum travel time (MMTT) or travel length (MMTL) can be useful in balancing, as well as load balancing objectives (LBTT and LBTL) used for fairness between drivers and balancing vehicle depreciations (Lee and Ueng 1999, Corberan et al., 2002). Other objectives can also be considered depending on the problem structure, such as nonmonetary objectives in humanitarian logistics problems.

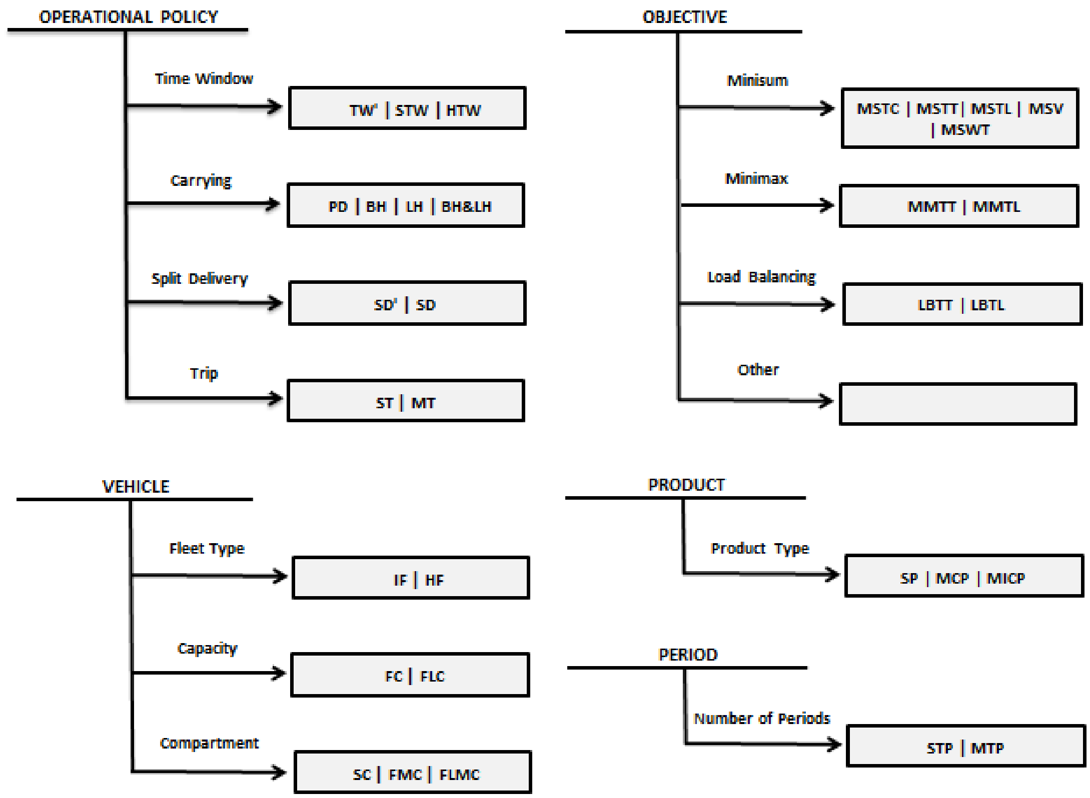

Figure 2.1 Taxonomic Review of VRP Literature

In the vehicle dimension, identical or heterogeneous vehicles (Rego and Roucairol, 1995, Cordeau and Laporte, 2001) determine the fleet type. The fixed capacitated VRP (FC) is the most studied in the VRP literature (Jaw et al., 1986, Angelelli and Speranzo, 2002). While FC class has fixed capacities of vehicles, in flexible fleet size (FLC) (Letchford and Englese, 1998), the number of vehicles or vehicle capacities can be increased. Vehicles can consist of a single compartment (SC), fixed size multiple compartments (FMC), or flexible-size multiple compartments (FLMC) (Derigs, 2011). In another dimension, the distribution may involve a single product (SP), multiple compatible (MCP) or incompatible (MICP) products. In terms of short or long term planning, the planning horizon may involve a single time period (STP) or multiple time periods (MTP).

VRP has a vast literature, but taxonomic studies are not many. Some of the VRP taxonomic studies in the literature can be listed as Reisman (1992), Bodin (1975), Bodin and Golden (1981), Desrochers et al. (1990), Laporte and Osman (1995), Desrochers et al. (1999), Eksioglu et al. (2009). We propose a methodology for classifying VRPs through **Figure 1**. The problem characteristics are determined by the operational policy. Objective, vehicle and product properties change the problem structure and solution methodology. Our taxonomy proposal is inspired from Kendall's Notation (Kendall, 1953). The classes are obtained by subclasses and each class features are separated by reagents. The notation covering five segments is described in **Figure 2.2**.

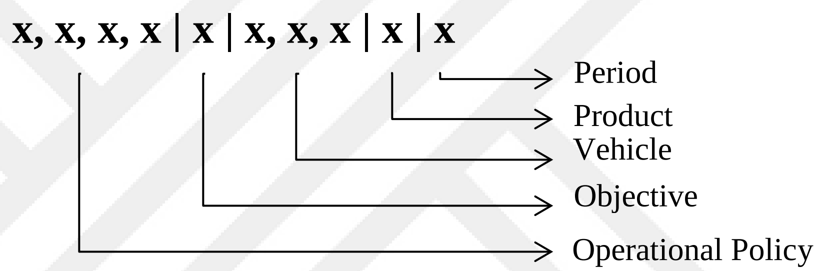

Figure 2.2 Proposed Taxonomy

For instance, MCVRP for Incompatible Products is symbolized as,

TW', LH, SD, MT | MSTT | IF, FC, FMC | MICP | STP.

2.1 Literature Review

The VRP problem was proposed by Dantzig and Ramser (1959) as a generalization of the well-known TSP. Clarke and Wright (1964) were among the first to develop a heuristic algorithm for this combinatorial optimization problem.

The split delivery VRP (SDVRP) was introduced by Dror and Trudeau in 1989. They indicated that by allowing split deliveries, some savings are made, heuristic and exact methods are generated. Archetti et al. (2006) developed a mixed integer program (MIP) for the SDVRP. They determined a threshold (k) for the quantity delivered on a route (k-SDVRP) and solved by using tabu search algorithm setting the length of the tabu list and the maximum number of iterations (SPLITABU). Boudia et al. (2007) introduced a memetic algorithm with population management for the split delivery VRP. In this metaheuristic, they used the Genetic Algorithm with local search for moves of split deliveries. Moreover, they applied the tabu search inspired from Archetti et al. (2006), Chen et al. (2007) developed a heuristic that combines endpoint mixed integer programming and record to record travel algorithm. They used the Clark and Write algorithm for the starting solution which is improved with endpoint integer programming. First of all, they found recorded best feasible solution until the time limit as the initial solution with the endpoint integer programming. Secondly, the solution was used as the new run of the program. And the final solution was calculated by post processing of the second solution with a variable length record to record travel algorithm in a reasonable time. Mota et al. (2007) presented scatter search methodology as a metaheuristic procedure for the split delivery VRP and objective was to minimize the total number of vehicles. They used two heuristic algorithms to split the demands such as Big Tour (Lin and Kernighan, 1973) and parallel Savings Algorithm (Clarke and Wright, 1964). To improve the solutions, interchanging one customer from a route with another client in another route for the non-split customers and two-split changes (specification of k-split changes) were used.

Derigs et al. (2009) employed the standard local search-based metaheuristics with inter-tour and intra-tour exchange operators for SDVRP. First, they applied some local moves such as, 2-Opt (Potvin and Rousseau, 1995), Exchange and Relocate (Savelsbergh, 1992). These moves were applied to find the best sequence of the deliveries within a tour as the initial solution. In the improvement phase, customers were sequenced in random order and tours are improved by 2-Opt. Tours were considered as TSPs and if a customer could not be served completely in the current tour, demand was splitted and the remaining demand was met in the next tour. They also tested their solutions with different metaheuristics such as Simulated Annealing (SA) (Kirkpatrick et al, 1983), Record to Record Travel (RRT) (Dueck, 1993), Attribute Based and Hill Climber heuristic (ABHC) (Whittley and Smith 2004). Wilck IV and Cavalier (2012) studied on a construction heuristic for the SDVRP. Partitioning procedure was used to assign customers to vehicle routes and there are three control steps in their algorithm. Vehicles completed their routes and unsatisfied demand of customers was assigned to the routes in Control 1. In Control 2 capacity level was determined and if it is feasible the algorithm proceeds to Control 3. If the assigned demand was greater than the capacity level in Control 1, Control 3 step occurred iteratively. Control 3 selected the number of customers closest to previous selected customers, adds the routes. After all demands were assigned to the vehicle routes, a TSP procedure were applied for each of the vehicles. Belenguer et al. (2000) proposed a cutting plane approach and some valid inequalities. They developed a lower bound and shown the quality of computational results. To minimize the total number of vehicles, Lee et al. (2006) studied the shortest path approach formulated with dynamic programming. Since the SDVRP is NP-Hard, they solved only the small size instances optimally. Jin et al. (2007) also minimized the number of vehicles and they used exact approaches as integer programming with valid inequalities for the clustering phase by minimizing the total clustering cost, and they solved the TSP (the routing phase).

The first multi-trip VRP (MTVRP) was introduced by Salhi (1987), the pairs of routes were matched with vehicles, and the number of trips was restricted two. Fleischmann (1990) developed a constructive heuristic with Savings Algorithm (Clarke and Wright 1964) for routing and Bin Packing for assignment part of the problem. Taillard et al. (1996) first generated a large set of vehicle routes by tabu search algorithm and selected the subset of the routes. These were matched with working days by using a Bin Packing heuristic. In those years, Brandao and Mercer (1997) worked on the same problem with time window issue. They proposed a constructive heuristic as a combination of nearest neighbor and insertion. Salhi and Petch (2007) proposed a hybrid Genetic Algorithm for minimizing the maximum overtime under fixed number of vehicles. They used chromosome injection, and cloning and crossover and mutation operators. For smaller VRP sub-problems, Savings Algorithm was used and complete set of vehicle trips are found by using Bin Packing heuristics. Azi et al. (2007) proposed an exact algorithm to solve single vehicle for multi-usage of vehicle under time window condition. All non-dominated feasible routes were generated firstly. In second, the potential routes were selected and sequenced according to vehicles working period.

Azi et al. (2010) introduced a branch and price algorithm. Lower bounds were computed by solving the linear programming relaxation. The pricing sub-problems were shortest path problems with resource constraints. In the problem of fixed size fleet, vehicles could not meet all customer demands and profit maximization determined which customers will be satisfied. Macedo et al. (2011) proposed an exact algorithm for the problem with time windows and multi-trip. The algorithm was iterative and it relied on a pseudo-polynomial network flow model. The nodes represented time instants, and the arcs represented feasible vehicle routes. They tested their algorithm on benchmark instances and their approach was found to be efficient according to literature. Hernandez et al. (2014) worked on the multi-trip Vehicle Routing Problem with time windows. They provided an exact two-phase algorithm, such as listing the customers that match the maximum trip duration and choosing a best set of trips by using branch and price. They proposed a set covering formulation as the column generation master problem, where columns represented trips and the sub-problem selected appropriate timing for trips.

The concept of multi-compartment VRP (MCVRP) includes separated compartments that have incompatible products for one or more customer demands. It aims to meet different types of customer demands by the same vehicle, through seeking the set of minimum route cost. VRP has been widely studied in the literature but there is not so much research on VRP. Some of the recent MCVRP studies are described in **Table 2.1** using our taxonomic classes.

Table 2.1 Recent MCVRP Literature

Authors	Year	Problem
Van der Bruggen et al.	1995	TW', LH, SD', ST | MMTC | HF, FC, FMC | MICP | STP
Avella et al.	2004	TW', LH, SD', ST | MMTL | IF, FC, FMC | MICP | STP
Suprayogi et al.	2006	TW', LH, SD, MT | MSV, MMTT, LBTT | IF, FC, FMC | MICP | STP
El Fallahi et al.	2008	TW', LH, SD, ST | MMTL | IF, FLC, FMC | MICP | STP
Derigs et al.	2011	TW', LH, SD', ST | MMTL | IF, FC, FMC | MICP | STP
Mendoza et al.	2010	TW', LH, SD', ST | MMTL | IF, FLC, FMC | MICP | STP
Surjandari et al.	2011	TW', LH, SD, ST | MMTL | HF, FC, FMC | MICP | MTP
Benantar and Oufi	2012	TW, LH, SD', ST | MMTL | HF, FC, FMC | MICP | STP
Asawarungsaengkul et al.	2012	TW', LH, SD, ST | MMTL | HF, FC, FMC | MICP | STP
Lahyani et al.	2014	TW', LH, SD, ST | MSV, MMTL | HF, FC, FMC | MICP | MTP

Van der Bruggen et al. (1995) studied on multi-compartment assignment problem for a large oil company. They considered the number and size of trucks and driver schedules. They first decided which customers are assigned to which depots and created vehicle routes and driver shifts with delivery schedules. Avella et al. (2004) introduced a multi-period routing problem and each compartment has to be full or empty. Once the assignments are made, the routing phase that corresponds to a TSP that can be solved individually for each vehicle. They developed a greedy heuristic to solve the problem. Suprayogi et al. (2007) worked on a multi-objective delivery problem in Indonesia and East Timor. They used sequential insertion and local search algorithms to obtain solutions.

Fallahi et al. (2008) discussed on the distribution of cattle food to farms and proposed the memetic algorithm and tabu search for solution of the MCVRP with split delivery for different type of products. They assumed that each compartment was dedicated to a single product, where a product may denote different types of goods that have common properties. The option to bring different products ordered by a customer using multiple vehicles was also allowed.

Derigs et al. (2011) introduced food and petrol distribution examples and a general model formulation for VRP. They added the multi-compartment vehicle constraints for incompatible products with extensions. In their problem, each compartment could carry any product, but different products could not be mixed in one compartment. A customer may place several orders, each referring to more than one single product. To solve their problem, they showed some heuristics such as, local search, large neighborhood search, adaptive search, construction heuristics and metaheuristics. They also discussed the flexible multi-compartment VRP.

Mendoza et al. (2010) proposed a memetic algorithm with genetic operators and local search procedures for MCVRP where customer demands were stochastic. Surjandari et al. (2011) introduced a Petrol Station Replenishment Problem (PSRP) in Indonesia, in which vehicles carried incompatible products and customer demands should be in multiples of compartment capacity. They proposed a mathematical model and a tabu search algorithm as the solution methodology. Benantar and Oufi (2012) also studied on MCVRP under time window constraints and proposed construction heuristics as well as tabu search. Asawarungsaengkul et al. (2012) proposed two different splitting patterns for assigning the products to compartments in MCVRP. They used Savings Algorithm for large size problem instances. Finally, Lahyani et al. (2014) studied on collection of olive oil with heterogeneous multi-compartment vehicles with split delivery for different type of products and they solved the problem by using a branch and cut algorithm.

In our study, we consider a MCVRP for Incompatible Products, which can be denoted as TW’,LH,SD,MT | MSTC | IF,FC,FMC | MICP | STP based on the developed taxonomy. Our problem involves separated compartments that can carry incompatible products for one or more customers while minimizing total cost.

3 METHODOLOGY

In this thesis, the basic, multi-trip and split delivery models as operation decisions of the MCVRP's solution are formulated. After the mathematical models are built and solved for small size data, the heuristic methods are constructed to obtain the solutions of large size instances. Moreover, to test the heuristic solutions' performances, some lower bounds are generated. The following sections describe the specifics of our methodology.

3.1 Formulation

As it was explained in the first chapter, the distribution system has a fleet of homogenous vehicles having several compartments, where a compartment on a vehicle can carry only one type of feed at a time. Each farm demands a single type of feed, and the same type of demands from different farms can be transported in the same compartment. The aim is to meet daily demands in a timely manner at minimum cost. We propose a basic mixed integer programming model formulation for this MCVRP. All decision variables are binary except the ones related to time. The extensions of the problem can be easily adapted using the basic model formulation in this section.

3.1.1 Assumptions

The problem is solved under following the set of assumptions:

A1	There is a single depot.
A2	There are enough products for each type at the depot.
A3	All vehicles are available and fully loaded at the depot.
A4	Fleet vehicles are identical.
A5	There is no breakdown.
A6	All vehicles have the same average speed.
A7	There is an 8 hour shift for all vehicles.
A8	Each compartment on a vehicle is dedicated to one type of product.
A9	Loading and unloading times are included in the traveling time.
A10	There is no customer priority.

A11 Each customer can order only one type of product.

A12 Customer demands cannot exceed a vehicle capacity.

A13 The same type of product from different customers can be mixed in the same compartment.

The following are the indices and parameters used in the formulation:

k : Vehicle index (k =1,…,K)

l : Compartment (silo) index (l=1,…,L)

i,j : Customer indices (i,j=1,…,N)

p : Product index (p=1,…,P)

f_k : Fixed cost for vehicle k (TL)

C_{lk} : Capacity of compartment l of vehicle k (tons)

D_{ip} : Demand of customer i for product type p (tons)

s_{ij} : Distance from customer i to customer j (km)

$\propto$: Fuel cost (TL/km)

T_k : Time capacity of vehicle k (min)

T_{ij} : Time from customer i to customer j (min)

We define our decision variables below.

$$x_{ki}=\begin{cases}1, \text{ if vehicle } k \text{ serves customer } i\\ 0, \text{ otherwise}\end{cases}$$

$$y_k=\begin{cases}1, \text{ if vehicle } k \text{ is used}\\ 0, \text{ otherwise}\end{cases}$$

$$u_{ij}=\begin{cases}1, \text{ if route from customer } i \text{ to customer } j \text{ is used}\\ 0, \text{ otherwise}\end{cases}$$

$$z_i=\begin{cases}1, \text{ if route from depot to customer } i \text{ is used}\\ 0, \text{ otherwise}\end{cases}$$

$v_i = \begin{cases} 1, \text{ if route from customer } i \text{ to depot is used} \\ 0, \text{ otherwise} \end{cases}$

$a_{lkp} = \begin{cases} 1, \text{ if compartment } l \text{ of vehicle } k \text{ is used for product type } p \\ 0, \text{ otherwise} \end{cases}$

t_{ij}^k=The time taken by vehicle k from customer i to customer j

3.1.2 Mathematical Model

Based on the above definitions, the mathematical model is as follows:

$$Min \sum_{k=1}^{K} f_k y_k + \alpha \left\{ \sum_{i=1}^{N} \left[(s_{i0}\, v_i) + (s_{0i} z_i) + \sum_{j=1}^{N} (s_{ij} u_{ij}) \right] \right\} \quad (1)$$

Subject to:

$$\sum_{i=1}^{N} x_{ki} \le N y_k \qquad \forall k = 1,\dots,K \quad (2)$$

$$\sum_{i=1}^{N} D_{ip} x_{ki} \le \sum C_{lk} a_{lkp} \qquad \forall k = 1,\dots,K \quad \forall p = 1,\dots,P \quad (3)$$

$$\sum_{p=1}^{P} a_{lkp} \le y_k \qquad \forall l = 1,\dots,L \quad \forall k = 1,\dots,K \quad (4)$$

$$\sum_{k=1}^{K} x_{ki} = 1 \qquad \forall i = 1,\dots,N \quad (5)$$

$$z_j + \sum_{i=1}^{N} u_{ij} = 1 \qquad \forall j = 1,\dots,N \quad i \ne j \quad (6)$$

$$v_i + \sum_{j=1}^{N} u_{ij} = 1 \qquad \forall i = 1,\dots,N \quad i \ne j \quad (7)$$

$$\sum_{i=1}^{N} z_i = \sum_{k=1}^{K} y_k \quad (8)$$

$$\sum_{i=1}^{N} v_i = \sum_{k=1}^{K} y_k \quad (9)$$

$$x_{ki} - (1 - u_{ij}) \le x_{kj} \qquad \forall k = 1,\dots,K \quad \forall i, j = 1,\dots,N \quad i \ne j \quad (10)$$

$$x_{kj} - (1 - u_{ij}) \le x_{ki} \qquad \forall k = 1,\dots,K \quad \forall i, j = 1,\dots,N \quad i \ne j \quad (11)$$

$$t_{ij}^k \ge T_{ij} - T_{ij}(2 - x_{ki} - u_{ij}) \qquad \forall k = 1,\dots,K \quad \forall i, j = 1,\dots,N \quad i \ne j \quad (12)$$

$$t_{0j}^k \ge T_{0j} - T_{0j}(2 - x_{kj} - z_j) \qquad \forall k = 1,\dots,K \quad \forall j = 1,\dots,N \quad (13)$$

$$t_{i0}^k \ge T_{i0} - T_{i0}(2 - x_{ki} - v_i) \qquad \forall k = 1,\dots,K \quad \forall i = 1,\dots,N \quad (14)$$

$$\sum_{k=1}^{K} t_{ij}^k \le T_{ij} u_{ij} \qquad \forall i, j = 1,\dots,N \quad i \ne j \quad (15)$$

$$\sum_{k=1}^{K} t_{0j}^k \le T_{0j} z_j \qquad \forall j = 1,\dots,N \quad (16)$$

$$\sum_{k=1}^{K} t_{i0}^k \le T_{i0} v_i \qquad \forall i = 1,\dots,N \quad (17)$$

$$\sum_{j=1}^{N} t_{0j}^k + \sum_{i=1}^{N} t_{i0}^k + \sum_{i=1}^{N} \sum_{j=1}^{N} t_{ij}^k \le T_k \qquad \forall k = 1,\dots,K \quad (18)$$

$$u_{ij} + u_{ji} \leq 1 \qquad \forall i, j = 1,\ldots,N \;\; i \neq j \qquad (19)$$

$$u_{ij} + u_{ji} + u_{im} + u_{jm} + u_{mi} + u_{mj} \leq 2 \qquad \forall i, j, m = 1,\ldots,N \;\; i \neq j \neq m \qquad (20)$$

$$x_{ki}, y_k, z_i, v_i, u_{ij} \in \{0,1\} \qquad \forall k = 1,\ldots,K \;\; \forall i, j = 1,\ldots,N \;\; i \neq j \qquad (21)$$

$$a_{lkp} \in \{0,1\} \qquad \forall l = 1,\ldots,L \;\; \forall k = 1,\ldots,K \;\; \forall p = 1,\ldots,P \qquad (22)$$

$$t_{ij}^k, \mathrm{t}_{0j}^k, t_{i0}^k \geq 0 \qquad \forall k = 1,\ldots,K \;\; \forall i, j = 1,\ldots,N \;\; i \neq j \qquad (23)$$

The objective function in (1) minimizes the total transportation cost including the fixed costs of the vehicles and the fuel costs. Constraint set (2) ensures that a vehicle can serve customers only if it is in use. Constraint set (3) ensures that the amount of any product cannot exceed the capacity of the assigned silo. Constraint set (4) ensures that there can be only one type of product in one silo of a vehicle. Constraint set (5) ensures that a customer can be served by only one vehicle, while constraint sets (6) and (7) guarantee flow balance. Constraint sets (8) and (9) ensure that the total number of departures (arrivals) from (to) the depot is equal to the total number of vehicles used. Constraint sets (10) and (11) ensure that customers on the same route are visited by the same vehicle, guaranteeing continuity of the routes. Constraint set (12) is used to form feasible routes in terms of travel times between nodes. Constraint sets (13) and (14) determine the times and routes between depot and a customer. Constraint sets (15), (16) and (17) relate time and route variables. While constraint set (18) ensures time limits for the routes, constraint set (19) and (20) enforces sub-tour elimination for only two and three-customer sub-tours. We add these constraints to prevent the sub-tours initially, since they are of cubic number. However, we add constraints for 4 and more customers whenever we detect sub-tours in the solution, as will be explained later. Finally, constraint sets (21), (22) and (23) dictate the structures and sign restrictions of the decision variables. As the simple VRP is NP-hard, so is our problem. The size of the model is defined with binary and continuous variables, and number of constraints in **Table 3.1**.

Table 3.1 Size of the Model

Binary Variables	Continuous Variables	Constraints
$\binom{N}{2} + K(N + PL + 1) + 2N$	$K\left(\binom{N}{2} + 2N\right)$	$15N + 9K + L + P$

3.1.3 Extensions

The basic model is extended with some modifications and additions. We propose two main extensions as multi-trips for vehicles and split delivery for customers. In addition, there can be an improvement for objective value by relaxing the time overtime constraint. The extensions are independent from each other.

Overtime:

Currently, all drivers work T_k-hour shifts. For evaluating the possibility of allowing overtime at a higher wage rate, a new nonnegative continuous decision variable is defined as follows:

o_k: overtime of vehicle k (min)

Constraint (18) is updated to include the overtime for each vehicle:

$$\sum_{j=1}^{N} t_{0j}^{k} + \sum_{i=1}^{N} t_{i0}^{k} + \sum_{i=1}^{N} \sum_{j=1}^{N} t_{ij}^{k} \le T_k + o_k \qquad \forall k = 1, \ldots, K \tag{18'}$$

The overtime of each vehicle is penalized in the objective function:

$$Min \sum_{k=1}^{K} (f_k y_k + c_o o_k) + \alpha \sum_{i=1}^{N} ((s_{i0}\, v_i) + (s_{0i} z_i) + \sum_{j=1}^{N} (s_{ij} u_{ij})) \tag{1'}$$

where $c_o = \dfrac{f_k}{T_k}$ is defined as the unit cost of overtime, computed as the fixed cost of a vehicle per minute.

Multiple Trips:

This extension of the basic model allows at most R trips per vehicle, with multiple visits to the depot during the working day. For this purpose, we define a trip index r = 1,…, R, and the number of vehicles in the model is replaced by k = 1,…,K, K+1,…,2K,…,2K+1,…,RK. With this representation, k+1,…,2K represent the second trips of the original K vehicles. A new constraint is formed to ensure that a vehicle can make its next tour only if it performs its previous one, as follows.

$$y_{rK+k} \leq y_{(r-1)K+k} \qquad \forall k = 1,...,K\,,\ \forall r = 1,...,R \qquad (24)$$

To satisfy the time constraint for the same vehicle in different trips, constraint (18) is updated as (18’’):

$$\sum_{r=1}^{R}\sum_{j=1}^{N} t_{0j}^{(r-1)K+k} + \sum_{r=1}^{R}\sum_{i=1}^{N} t_{i0}^{(r-1)K+k} + \sum_{r=1}^{R}\sum_{i=1}^{N}\sum_{j=1}^{N} t_{ij}^{(r-1)K+k} \leq T_k \ \ \forall k = 1,...,K \ (18’’)$$

Split Delivery:

When split delivery is allowed, a customer’s demand can be met through visits of several vehicles. In such a case, Constraint set (5) should be omitted from the model. To allow split delivery, a vehicle index is added to decision variables *u*, *z* and *v*. Constraint (6) and (7) are modified. The following constraint (25) is added to the model to ensure flow balance through the routes of the vehicles.

$$\mathrm{z}_{kj} + \sum_{i=1}^{N} u_{kij} = x_{kj} \qquad \forall k = 1,...,K \ \ \forall j = 1,...,N \ \ i \neq j \qquad (6’)$$

$$\mathrm{v}_{ki} + \sum_{i=1}^{N} u_{kij} = x_{ki} \qquad \forall k = 1,...,K \ \ \forall i = 1,...,N \ \ i \neq j \qquad (7’)$$

$$\sum_{i=1}^{N} u_{kji} + v_{ki} = \sum_{i=1}^{N} u_{kij} + z_{kj} \qquad \forall k = 1,...,K \ \ \forall j = 1,...,N \ \ i \neq j \qquad (25)$$

Another change handles the fulfillment of customer demand through the definition of a new continuous nonnegative decision variable q_{ki} between 0 and 1, representing the percentage of customer *i*’s demand met by vehicle *k*. Constraint set (3) is then replaced by (3’) to accommodate this change. Additional constraint set (26) ensures that a vehicle can satisfy a demand percentage of a customer only if it visits that customer. Demand of each customer is met via additional constraint set (27), and the new decision variable is defined by constraint set (28).

$$\sum_{i=1}^{N} D_{ip} q_{ki} \leq \sum_{l=1}^{L} C_{lk} a_{lkp} \qquad \forall k = 1,...,\mathrm{K} \ \ \forall p = 1,...,\mathrm{P} \qquad (3’)$$

$$\mathrm{q}_{ki} \leq x_{ki} \qquad \forall k = 1,...,K \ \ \forall i = 1,...,N \qquad (26)$$

$$\sum_{k=1}^{K} q_{ki} = 1 \qquad \forall i = 1,...,N \qquad (27)$$

$$\mathrm{q}_{ki} \geq 0 \qquad \forall k = 1,...,K \ \ \forall i = 1,...,N \qquad (28)$$

Constraint sets (15), (16) and (17) are updated as below to handle the relations between vehicle visits and corresponding time.

$$t_{ij}^{k} \le T_{ij} u_{kij} \qquad \forall i, j = 1,\ldots,N \;\; i \ne j \qquad (15')$$

$$t_{0j}^{k} \le T_{0j} z_{kj} \qquad \forall j = 1,\ldots,N \qquad (16')$$

$$t_{i0}^{k} \le T_{i0} v_{ki} \qquad \forall i = 1,\ldots,N \qquad (17')$$

3.2 Solution Approach

VRP is a fundamental problem in combinatorial optimization. The problem can be modeled and solved with general purpose solvers such as Lingo, Excel-Solver, OPL, etc. However, as it is known as an NP-Hard problem, it is hard especially to treat the sub-tour elimination constraints. Therefore, heuristic approaches are needed for real life problem instances. Many heuristic and metaheuristic approaches have been developed specifically for VRP by seeking good solutions in reasonably small time periods.

We first developed a mathematical model and solved it optimally for small sized test problems by using IBM ILOG OPL CPLEX 12.6 solver. The formulation that we discussed in Section 3.1 is solved only with sub-tour elimination constraints for two and three customers. Existent sub-tours for more customers are checked at the end of each solution. If any sub-tour occurs, the proper sub-tour constraints are activated.

Figure 3.1 illustrates an example how sub-tours are eliminated for more than three customers. Customers 4, 6, 7 and 10 have a closed loop, and there is no connection with the depot. To break the loop, the following constraint is appended to the formulation:

$$u_{4,6} + u_{4,7} + u_{4,10} + u_{6,4} + u_{6,7} + u_{6,10} + u_{7,4} + u_{7,6} + u_{7,10} + u_{10,4} + u_{10,6} + u_{10,7} \le 3$$

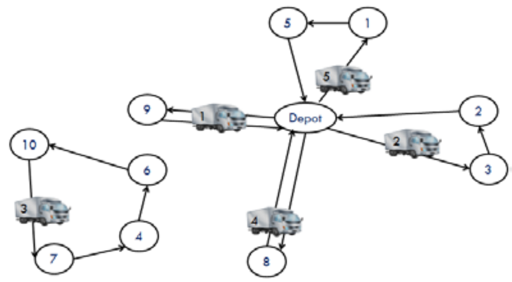

Figure 3.1 Example of sub-tour elimination

We could not find an optimal solution for the basic model after 15 customers, and after 10 customers for multi trips and split delivery models in a reasonable time. Therefore, some construction and improvement heuristics are developed and coded in C++ integrated with Visual Studio Express. The heuristic approaches for the basic, multi-trip and split delivery cases are explained below. We use a sample problem to illustrate all steps. In the problem instance there are 10 customers and their demands and product types are given in **Table 3.2**. The vehicle fixed cost is 400TL. The symmetric distance matrix is given in **Table 3.3**. Each vehicle has 4 compartments, each with a capacity of 4 tons.

Table 3.2 Example Customer Demands

Customers	1	2	3	4	5	6	7	8	9	10
Demand (tons)	12	12	1	15	12	9	9	11	2	7
Product Type	3	3	1	3	1	4	2	2	1	4

Table 3.3 Example Distance Matrix

Customers	1	2	3	4	5	6	7	8	9	10
1	-	156.76	167.26	97.95	123.92	16.47	38.56	142.31	111.46	6.27
2	156.76	-	185.08	180.33	176.10	164.64	152.48	152.73	102.10	154.74
3	167.26	185.08	-	79.33	46.92	156.36	200.44	33.64	88.40	172.09
4	97.95	180.33	79.33	-	32.52	83.94	135.37	68.79	83.47	103.75
5	123.92	176.10	46.92	32.52	-	111.65	159.31	40.34	74.14	129.21
6	16.47	164.64	156.36	83.94	111.65	-	54.76	133.50	108.51	22.72
7	38.56	152.48	200.44	135.37	159.31	54.76	-	172.82	133.50	32.34
8	142.31	152.73	33.64	68.79	40.34	133.50	172.82	-	54.76	146.53
9	111.46	102.10	88.40	83.47	74.14	108.51	133.50	54.76	-	113.82
10	6.27	154.74	172.09	103.75	129.21	22.72	32.34	146.53	113.82	-

3.2.1 Basic Model

Initially, customer demands are divided by compartment capacity to find a bound for the needed number of compartments. For instance; if a customer has 13 tons of a product, it should have ($\lceil 13/4 \rceil =$) 4 compartments. After initialization step, customer demands are converted to the needed number of silos as shown in **Table 3.4**.

Table 3.4 Initialization of Customer Demands and Product Types

Customers	1	2	3	4	5	6	7	8	9	10
Demand (tons)	12	12	1	15	12	9	9	11	2	7
Demand (#of silos)	3	3	1	4	3	3	3	3	1	2
Product Type	3	3	1	3	1	4	2	2	1	4

Construction Step:

We have four different construction heuristics to build the initial solutions. The parameters and variables that are used in the procedures are given below;

n : number of customers
x : selected customer
d_x : selected customer demand
c_k : current vehicle load
v_x : the vehicle of customer x
r : number of iterations

a. Random Start Heuristic 1 (RSH_1)

Customers are assigned to a vehicle one by one randomly, as long as there is enough capacity in the vehicle. If no more customers can be assigned to the vehicle, a new vehicle is opened. The procedure continues until all customers are assigned to vehicles.

```
procedure   RSH_1(v)
   do
      do
         x = customer is selected randomly from (n)
         if ( capacity and time constraints are satisfied after
              customer x is loaded into current vehicle )
               v_x ← k
               c_k ← c_k + d_x
               n ← n + 1
         end if
         r ← r + 1
      while ( current vehicle load is less than vehicle capacity
              and iteration limit is not reached )
      k ← k + 1
      r ← 0
      c_k ← 0
   while ( not all customers are assigned to vehicles )
   return   v
endprocedure
```

At the end of RSH_1, vehicles are assigned to the customers randomly, and each vehicle has a route as shown in **Table 3.5**. The objective value of our example RSH_1 construction solution is 5149.37 TL.

Table 3.5 RSH_1's Solution

Customers	1	2	3	4	5	6	7	8	9	10
Demand (#of silos)	3	3	1	4	3	3	3	3	1	2
Product Type	3	3	1	3	1	4	2	2	1	4
Vehicle	5	1	2	6	2	3	8	7	1	4

b. Random Start Heuristic 2 (RSH_2)

In this heuristic, the customers are paired randomly. If the selected pair's demand can fit into a single vehicle, they are both assigned; else they are released. After all possible pairs are assigned; each remaining single customer is assigned to an individual vehicle.

$procedure \quad RSH\_2(v)$

do

do

$x_1 = \text{first customer is selected randomly from } (n)$

$x_2 = \text{second customer is selected randomly from } (n)$

if capacity and time constraints are satisfied after customer x_1 and x_2 are loaded into current vehicle

$v_{x_1} \leftarrow k$

$v_{x_2} \leftarrow k$

$c_k \leftarrow c_k + d_{x_1} + d_{x_2}$

$n \leftarrow n + 2$

$end\ if$

$r \leftarrow r + 1$

$while \begin{pmatrix} \text{current vehicle load is less than vehicle capacity} \\ \text{and iteration limit is not reached} \end{pmatrix}$

$k \leftarrow k + 1$

$r \leftarrow 0$

$c_k \leftarrow 0$

$r_2 \leftarrow r_2 + 1$

$while \begin{pmatrix} \text{not all customers are assigned to vehicles} \\ \text{and second iteration limit is not reached} \end{pmatrix}$

$return \quad v$

$endprocedure$

At the end of the RSH_2, vehicles are assigned to the customers two by two randomly and each vehicle has a route shown in **Table 3.6**. Objective value of RSH_2 construction solution is 5231.18 TL.

Table 3.6 RSH_2's Solution

Customers	1	2	3	4	5	6	7	8	9	10
Demand (#of silos)	3	3	1	4	3	3	3	3	1	2
Product Type	3	3	1	3	1	4	2	2	1	4
Vehicle	3	4	1	5	1	2	6	7	2	8

c. Smart Start Heuristic 1 (SSH_1)

SSH_1 construction heuristic is similar to the nearest neighborhood algorithm for TSP. The first customer of a vehicle is the closest customer from the depot and the next customer in the sequence is always the closest to the current customer. It continues by satisfying the capacity constraint. When the capacity is exceeded, a new vehicle is opened and the same procedure is used.

$procedure \quad SSH\_1(v)$

$\quad do$

$\quad\quad do$

$\quad\quad\quad x = \text{nearst customer from current position}$

$\quad\quad\quad if \begin{array}{l}\text{capacity and time constraints are satisfied after} \\ \text{customer x is loaded into current vehicle}\end{array}$

$\quad\quad\quad\quad v_x = k$

$\quad\quad\quad\quad c_k = c_k + d_x$

$\quad\quad\quad\quad n = n + 1$

$\quad\quad\quad end\ if$

$\quad\quad while(\text{current vehicle load is less than vehicle capacity})$

$\quad\quad k = k + 1$

$\quad\quad c_k = 0$

$\quad while(\text{not all customers are assigned to vehicles})$

$\quad return \quad v$

$endprocedure$

At the end of the SSH_1, vehicles are assigned to the customers according to nearest neighborhood search and each vehicle has a route as shown in **Table 3.7**. Objective value of SSH_1 construction solution is 5162.53 TL.

Table 3.7 SSH_1's Solution

Customers	1	2	3	4	5	6	7	8	9	10
Demand (#of silos)	3	3	1	4	3	3	3	3	1	2
Product Type	3	3	1	3	1	4	2	2	1	4
Vehicle	5	8	3	2	3	4	7	1	1	6

d. Smart Start Heuristic 2 (SSH_2)

SSH_2 heuristic aims to minimize the total number of vehicles. Firstly, customers are sorted in decreasing order according to their demands in terms of compartments. The algorithm starts to put away the customers to a vehicle and until the vehicle is full. When there is no customer demand that can be loaded, a new vehicle is opened and algorithm returns the beginning of the array.

procedure $SSH\_2(v)$
δ {customer demands are in decreasing order}
do
do
$x =$ customer is selected successively according to sorted demands
if capacity and time constraints are satisfied after customer x is loaded into current vehicle
$v_x = k$
$c_k = c_k + d_x$
$n = n + 1$
end if
while(current vehicle load is less than vehicle capacity)
$k = k + 1$
$c_k = 0$
while(not all customers are assigned to vehicles)
return v
endprocedure

At the end of the SSH_2, the vehicles are assigned to the customers according to minimum usage by compressing the demands to the vehicles and each vehicle has a route as shown in **Table 3.8**. The objective value is 5162.53 TL.

Table 3.8 SSH_2's Solution

Customers	1	2	3	4	5	6	7	8	9	10
Demand (#of silos)	3	3	1	4	3	3	3	3	1	2
Product Type	3	3	1	3	1	4	2	2	1	4
Vehicle	1	2	3	4	5	6	7	8	2	3

Improvement Step:

The construction heuristics provide initial solutions indicating how customers are assigned to vehicles and they determine the routes of the vehicles. At the end of the each construction heuristic, the following Variable Neighborhood Search (VNS) algorithm is used to improve the route of each vehicle.

$procedure \quad VNS(\pi)$
$k = 1 \qquad k_{\max} = 2$
$N_1(\pi) = Swap \qquad N_2(\pi) = Insertion$
$\quad do$
$\qquad \pi_1 = N_k(\pi)$
$\qquad if \quad (\mathrm{f}(\pi_1) < \mathrm{f}(\pi))$
$\qquad\quad \pi \leftarrow \pi_1$
$\qquad\quad k \leftarrow 1$
$\qquad else$
$\qquad\quad k \leftarrow k + 1$
$\qquad end\ if$
$\quad while(k \le k_{\max}$ and iteration number is less than number of customers$)$
$\quad return \quad \pi$
$endprocedure$

The improvement is based on the following idea. At the beginning of the improvement the customer demands are aggregated into number of compartments. So, there could be empty places inside the compartments. By combining different customer demands, the number of vehicles and also route costs can be decreased. In our improvement heuristic, all computations are made in terms of tons, that are, the actual demands.

In the first step of the improvement algorithm, the vehicles are sorted in increasing order according to their total current capacities. The first customer demand is successively removed from current location, the algorithm checks whether there exists a vehicle whose remaining capacity for the same type of product is greater and equal to the customer's demand. The displacement is started from the first customer of the first vehicle and the procedure tries to find a room in the last vehicle. As it is not easy to close the vehicle that has the maximum capacity, the aim is to maximize the total amount of demand served by that vehicle. It is tried to start processing from the vehicle with the minimum capacity usage. If there is not enough capacity in the last vehicle, other vehicles are tried in the backward manner.

The vehicle wanted to be closed/removed from schedule, is called “chosen vehicle”. The selected customer in chosen vehicle is named as “chosen customer” and the vehicle that is tried to be put away is called “found vehicle”. If there is enough capacity for the chosen customer in found vehicle and the product type is proper, the customer (demand) is directly assigned to that vehicle. The initial and changed route costs are recalculated for both chosen and found vehicles. The new routes are improved by using VNS. The difference of initial and changed cost for chosen vehicle is kept as “saving” and the difference between initial and changed cost for found vehicle is called “loss”. The remaining capacity is updated and the next customer of the same vehicle will be processed. After all customers that could be removed in the same vehicle are processed, the total losses are computed..

If the summation of saving and loss is positive, the value is subtracted from the objective value. It means that the change yields a better route and chosen customer remains in the found vehicle route. If the result is negative and the vehicle is empty, all negative changes are added and compared with the vehicle fixed cost. If vehicle fixed cost is grater, negative changes remain and the difference is subtracted from the objective value. If the vehicle cost is less than negative changes, all negative movements for chosen vehicle are withdrawn. Finally, if the vehicle is not empty, negative changes are also withdrawn for all movements of the chosen vehicle. Such movement of the customers constitutes a myopic solution. Some movement decisions depend on the other customers on the same route. Therefore, a second chance is given to improvement step to recover some gaps, and the algorithm has two replications in such a way that the current solution is recalled for a second run when the first improvement phase is ended.

All of our construction algorithms except RSH_2 have quadratic worst-case time complexities as $O(N^2)$, where N is the number of customers. The algorithmic complexity of RSH_2 is defined as $O\left(\binom{N}{2}^2\right)$. The order of VNS algorithm is also $O(N^2)$. At the beginning of the improvement step, the vehicles are sorted with the bubble sort method, of complexity $O(N^2)$. The customers are switched among the vehicles, and the complexity of this operation is $O(KN^3)$ where K represents the number of vehicles. Hence, the overall complexity of the improvement becomes $O(KN^3)$.

The vehicles' time constraints (where the time limit is 8 hours for each vehicle) are considered in all construction steps of the heuristic algorithms. But in the improvement phase, we first ignore the time constraint to provide flexibility of customer changes between vehicles. At the end of the improvement step, tour completion times are calculated, and if there is a violation, the same improvement algorithm with time-controlled version is applied. To illustrate the improvement phase, the solution steps on the example problem are shown in **Table 3.9**. The current total cost is recorded as 5149.37 TL with 8 vehicles in **Table 3.9**. **Table 3.10** indicates the vehicles that contain the assigned customers.

Table 3.9 Vehicle Assignment to the Customers - RSH_1

Customers	1	2	3	4	5	6	7	8	9	10
Demand (#of silos)	3	3	1	4	3	3	3	3	1	2
Product Type	3	3	1	3	1	4	2	2	1	4
Vehicle	5	1	2	6	2	3	8	7	1	4

Vehicles sorted in decreasing order according to their capacities are shown in **Table 3.11**.

Table 3.10 Initial Vehicle Routes - RSH_1

Vehicles	1	2	3	4	5	6	7	8
Customers	2,9	3,5	6	10	1	4	8	7

Table 3.11 Vehicle Routes with Sorted Vehicle Capacities - RSH_1

Vehicles	1	2	3	4	5	6	7	8
Customers	10	6	7	8	1	3,5	2,9	4

The first step of the improvement phase starts with the first vehicle. The vehicle visits customer 10 in the initial solution. The customer is tried to be removed from the first vehicle, and another vehicle is searched starting from the last vehicle in the order. There is enough capacity and proper product type in vehicle 2, so customer 10 switches to vehicle 2. The route of vehicle 2 is changed; routing cost increases by an amount of 44.30 TL with increasing number of customers. Otherwise, the routing cost of vehicle 1 decreases 226.62 TL. As a result, vehicle 1 becomes empty, it is closed with 182.32 TL gain and a saving of 400TL fixed cost. As a result, the objective value decreases from 5149.37 TL to 4567.05 TL with 7 vehicles as shown in **Table 3.12**.

Table 3.12 Vehicle Routes with Dischargement of Vehicle 1- RSH_1

Vehicles	1	2	3	4	5	6	7	8
Customers	-	6,**10**	7	8	1	3,5	2,9	4

The same procedure is applied for vehicles 2, 3, 4 and 5 but customers that are in these vehicles could not find any proper place to be moved. In processing vehicle 6, customer 3 is switched to vehicle 7. The routing cost of vehicle 6 decreases by an amount of 112.45 TL, while the routing cost of vehicle 7 increases by 205.79 TL. The loss for routing is 93.34 TL. If there is no customer remaining in vehicle 6, it can be closed if the routing cost is less than the vehicle's fixed cost. Because customer 5 could not move to another vehicle, there is a loss and customer 3 goes back to vehicle 6 as shown in **Table 3.13**.

Table 3.13 Vehicle Routes with Dischargement of Vehicle 6 - RSH_1

Vehicles	1	2	3	4	5	6	7	8
Customers	-	6,10	7	8	1	5	2,9,**3**	4

A similar case is observed for vehicle 7. Customer 9 changes its place to vehicle 6. The routing cost of vehicle 7 decreases by 22.76 TL when the routing cost of vehicle 6 increases by 41.74 TL. The loss for routing is 18.98 TL. If there is no customer left in vehicle 7, it can be closed because the routing cost is less than vehicle's fixed cost. Since there is a net loss, customer 9 goes back to vehicle 7 shown as in **Table 3.14**.

Table 3.14 Vehicle Routes with Dischargement of Vehicle 7- RSH_1

Vehicles	1	2	3	4	5	6	7	8
Customers	-	6,10	7	8	1	5,3,**9**	2	4

At the end of the first iteration, the results of the remaining vehicle routes are shown in **Table 3.15** and vehicle assignment to customers is given in **Table 3.16;** objective value remains as 4567.05 TL.

Table 3.15 Vehicle Routes - RSH_1

Vehicles	1	2	3	4	5	6	7
Customers	6,10	7	8	1	5,3	2,9	4

Table 3.16 Vehicle Routes - RSH_1 -Improvement

Customers	1	2	3	4	5	6	7	8	9	10
Demand (#of silos)	3	3	1	4	3	3	3	3	1	2
Product Type	3	3	1	3	1	4	2	2	1	4
Vehicle	4	6	5	7	5	1	2	3	6	1

After 1000 replications, the final objective value for the basic model is found as 4542.98 TL with 7 vehicles as is given in **Table 3.17.**

Table 3.17 Final Vehicle Routes - RSH_1 -Improvement

Customers	1	2	3	4	5	6	7	8	9	10
Demand (#of silos)	3	3	1	4	3	3	3	3	1	2
Product Type	3	3	1	3	1	4	2	2	1	4
Vehicle	3	5	5	7	4	1	2	6	4	1

As well as basic mode, the multi-trip and split delivery models have the same construction heuristics, but they differ in the improvement phase and thereafter.

3.2.2 Multi-Trip Model

Multi-trip model has similar improvement steps as compared with the basic model, but the total tour completion times of two determined vehicles (it is assumed that a vehicle can have at most two trips in an 8 hour time limit) should be less than 8 hours. As these two vehicles actually represent two trips of the same vehicle, the fixed cost of one of the two determined vehicles is subtracted from the objective function. All construction and improvement steps are the same as the basic model; but at end of the improvement step, the tour completion times are calculated and sorted in decreasing order and two vehicles with total tour completion time less than 8 hours are named as the same vehicle. Therefore, one vehicle fixed cost is subtracted from the objective value.

At the end of the basic model, the computed tour completion times are given in **Table 3.18**.

Table 3.18 Vehicles' Tour Completion Times - RSH_1 -Improvement

Vehicles	1	2	3	4	5	6	7
Route Times	175	211	134	154	207	264	114

The completion times are sorted in decreasing order as given in **Table 3.19** and paired up by considering the time constraint that is 480 minutes (8 hour shift) as repeated in **Table 3.20**. It is possible to decrease the number of vehicles from 7 to 4 by allowing multiple tours, and the objective value becomes 3342.98 TL.

Table 3.19 Tour Completion Times in Decreasing Order - RSH_1-Improvement

Vehicles	6	2	5	1	4	3	7
Route Times	264	211	207	175	154	134	114

Table 3.20 Vehicle Combinations - RSH_1- Improvement

Vehicles	6	2	5	1	4	3	7
Route Times	264	211	207	175	154	134	114

3.2.3 Split Delivery Model

In split delivery model, we assume that one or two vehicles can meet one customer demand. In the case that there are two vehicles visiting a customer, percentage of a customer demand served by one of the vehicles should be greater than 25%, called as the minimum splitting ratio. The heuristic algorithm has the same construction and improvement steps. However, at the end of the improvement phase, each vehicle is tried to be closed. According to split delivery assumptions, if some customers remain in a vehicle at the end of the improvement phase of that vehicle, these customers' demands are split into two parts and sent to the two different vehicles.

To split a customer demand, the minimum splitting ratio rule is applied. For example, if the first found vehicle's remaining capacity for the same type of product is less than the 25% of the chosen customer demand in the chosen vehicle, it is skipped. If the first found vehicle's remaining capacity for the same type of product is greater than 75% of the chosen customer demand in the chosen vehicle, 75% of the chosen customer's demand in the chosen vehicle is loaded to the first found vehicle. However, the first found vehicle's remaining capacity for the same type of product is greater than 25% and less than 75% of the chosen customer's demand, all remaining capacity of the first found vehicle is used. At the end of the splitting procedure, if the chosen vehicle cannot be closed, all customers that cause the increasing in routing costs are returned back to their original vehicles.

The split procedure is applied after each replication of the basic problem. To illustrate the procedure, we use the same problem data. The first step of the split algorithm is to sort the vehicle capacities in increasing order. The improvement step output for **Table 3.15** is given in **Table 3.21.**

Table 3.21 Vehicle Routes with Sorted Vehicle Capacities - RSH_1_Split

Vehicles	1	2	3	4	5	6	7
Customers	7	8	1	5,3	2,9	4	6,10

The algorithm first tries to discharge vehicle 1. The actual demand of customer 7 is 9 tons in terms of 2^{nd} product type and also there is no place for customer 7 to change its vehicle. The remaining capacity of each vehicle is calculated, and the remaining capacity of vehicle 2 is 1 ton for 2^{nd} product type and it has 1 empty compartment (4 tons) that can be proper for all types of products. Moreover, the remaining capacity of vehicle 3 is also equivalent to an empty compartment (4 tons). So, the demand of customer 7 is divided into two parts; 5 tons of the demand is placed into vehicle 2, and 5 tons goes to vehicle 3 as shown in **Table 3.22.**

Table 3.22 Vehicle Routes with Dischargement of Vehicle 1 - RSH_1 - Split

Vehicles	1	2	3	4	5	6	7
Customers	-	8,7	1,7	5,3	2,9	4	6,10

The replacement of customer 7 yields a decreased route cost for vehicle 1, while an increased cost for vehicles 2 and 3. The routing cost of vehicle 1 decreases by an amount of 295.80 TL, whereas vehicle 2's cost increases 295.62 TL and vehicle 3's is 93.39 TL, leading to a loss of 93.21 TL. However, vehicle 1 can be closed, and the fixed cost could be saved as 400TL. Therefore, the movement of customer 7 generates net savings and the new objective value becomes 4260.26 TL with 6 vehicles. For other vehicles, there is no improvement; the replacement of customers causes only losses and/or vehicles cannot be closed. The final routes are given in **Table 3.23**.

Table 3.23 Final Vehicle Routes - RSH_1_Split

Vehicles	1	2	3	4	5	6
Customers	8,7	1,7	5,3	2,9	4	6,10

3.3 Lower Bound

Our basic model can be solved up to 15 customers and we could not find optimal solutions for 20 customers and more. For split delivery and multi-trip models can only be solved up to 15 customers. Since the problem is NP-Hard, some combinatorial optimization methods and some smart approaches can provide feasible solutions as lower bounds. To test our heuristic model for large size problem instances, we compute some lower bound values that consist of routing cost and fixed cost.

In basic model, the lower bound of the total fixed cost is computed by multiplying the total number of vehicles with the fixed cost of a vehicle. To find the number of vehicles, the total amount of demand for each type of product is divided into the compartment capacity, and the upper integer represents the total number of required compartments for product type. The total number of needed compartments is divided into the number of compartments that are located in the vehicles; this upper bound value gives the total number of required vehicles. This number is compared with the number of customers whose demands are more than half of the vehicle capacity. As each of these customers should be visited by only one vehicle, the highest value is taken. An example to find the lower bound for number of vehicles for the same sample data in **Table 3.2** is shown below:

$$\#Vehicles = \frac{\sum_p \frac{\sum_i Dip}{C}}{Number\ of\ compartments} = \frac{\frac{(1+12+2)}{4}+\frac{(9+11)}{4}+\frac{(12+12+15)}{4}+\frac{(9+7)}{4}}{4} = 6$$

Next, the demands of customers higher than 8 tons are counted, and 7 such customers are found. Hence, minimum 7 vehicles are needed. As this check gives the higher value, the fixed cost of the problem is equal to 2800TL (7x400TL).

In the multi-trip model, the lower bound of the fixed cost is computed by dividing the number of vehicles found in basic model by two if the number is even; and if it is odd, the number is increased by one and then divided by two. For the split-delivery model, the lower bound of the fixed cost is the same as basic model, but without the check for high demand customers. There are three different lower bounding methods for the routing part, as explained below.

Lower Bound-1:

The first lower bound method tries to find the minimum distribution distance by seeking the customers' neighbors individually. The procedure starts from the first customer and selects two different closest customers that are nearest. This is applied for each customer, and each customer's connection nodes consist of the two closest neighbors, as shown in **Figure 3.2**.

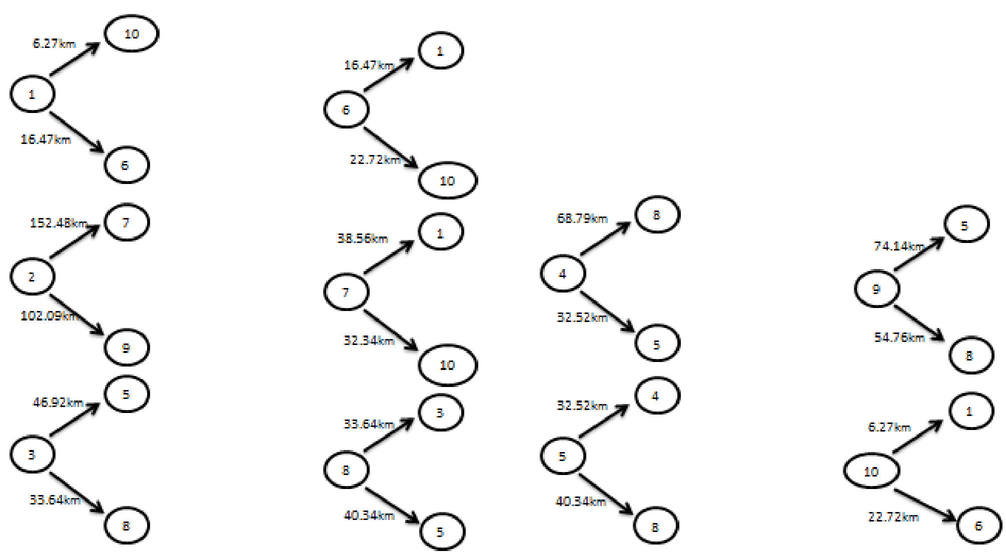

Figure 3.2 Customers Neighbors - LB-1

After these connections are included in the route, there is one entrance and one exit, and a customer can be a neighbor for two different customers. So, the customer can be counted twice. After the neighbor customers are found, multiple counts are eliminated. If a customer is counted more than twice, minimum two distances for that customer are kept and the rest are erased as illustrated in **Figure 3.3**. After the multiple counts are eliminated, the new situation is shown in **Figure 3.4**.

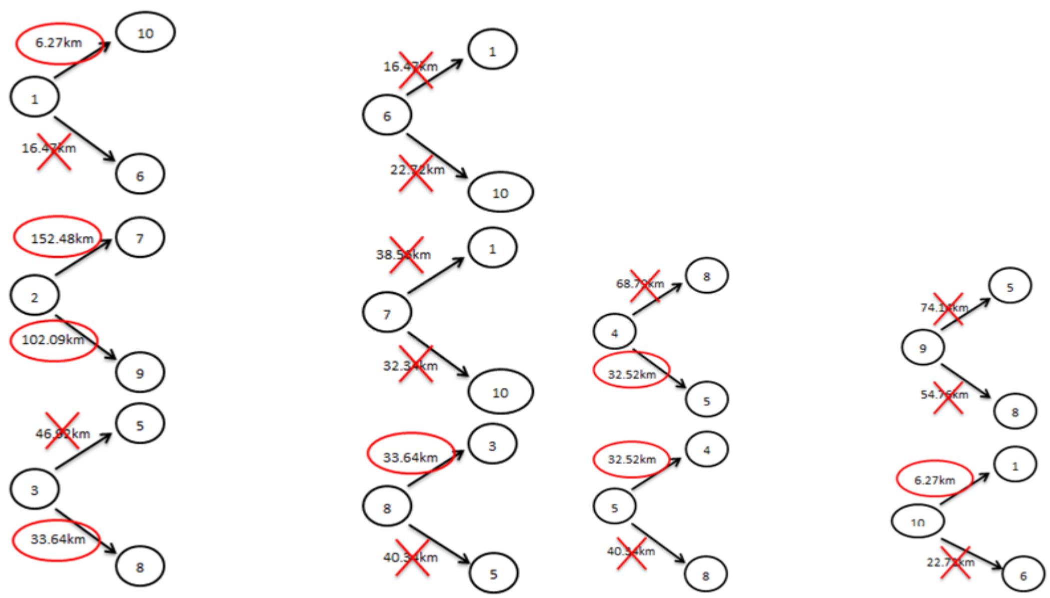

Figure 3.3 Marking Multiple Counts - LB-1

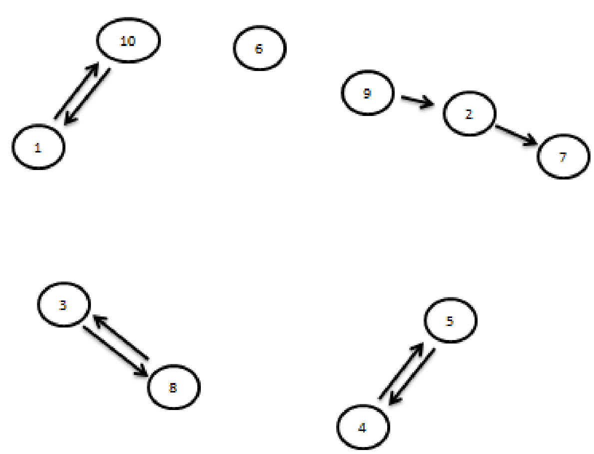

Figure 3.4 Eliminating Multiple Counts - LB-1

The next step is to enforce the depot connection arcs. The number of entrances/exits from/to depot should be equal to lower bound of the number of vehicles (7 vehicles), and for each vehicle the closest customers from the depot are selected as starting nodes. Selected customers' previous connections are deleted, and their all arcs are connected to depot, as shown in **Figure 3.5**. The total of all arc distances are multiplied by fuel cost and the result gives a lower bound for the routing cost.

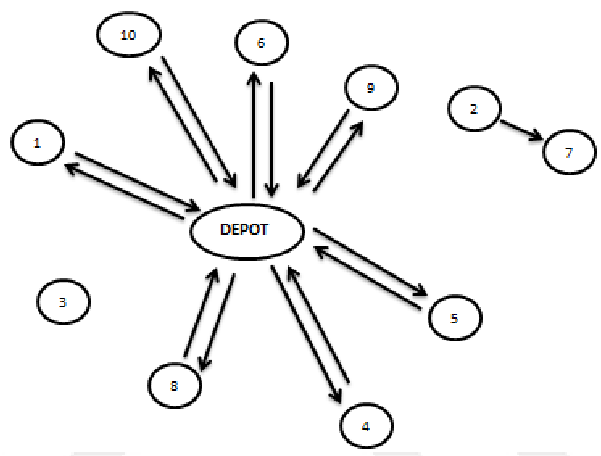

Figure 3.5 Customer-Depot Connections - LB-1

The total distances of the connected arcs on the network is equal to 1065.35 km and the total routing cost is found as 1491.49 TL by multiplying total distance with the 1.4 TL/km fuel cost. The objective value is equal to 4291.49TL (routing cost + fixed cost, 1491.49TL + 2800TL).

Lower Bound-2:

The second lower bound is found by solving an assignment problem without depot and sub-tour elimination constraints. The example problem is solved in CPLEX OPL and solution is visualized in **Figure 3.6**.

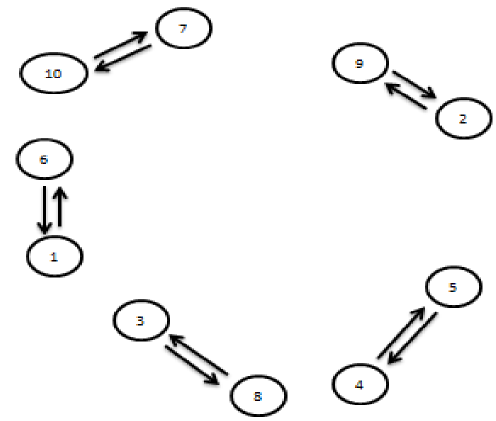

Figure 3.6 Solution of Assignment Problem - LB-2

To add the depot connections on the assignment solution, the closest 7 customers from the depot are selected. Selected customers' connection arcs are deleted and they are connected to the depot as shown in **Figure 3.7**. The total of all distances are multiplied by the fuel cost, and the result gives a lower bound for the routing cost. In our example, the total distance of the connected arcs is 921.87 km, and the total routing cost is found as 1278.01 TL by multiplying total distance with the 1.4 TL/km fuel cost. The lower bound for the objective value becomes 4078.01TL (routing cost + fixed cost, 1278.01TL + 2800TL).

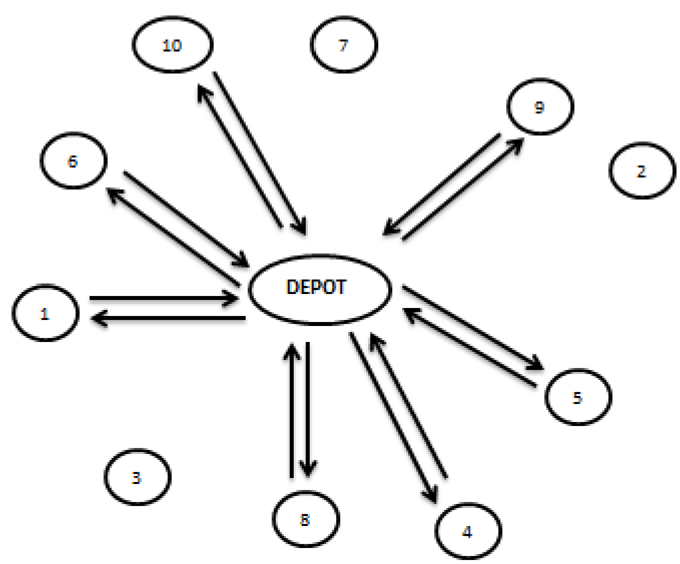

Figure 3.7 Customer-Depot Connections - LB2

Lower Bound-3:

The last lower bound is obtained by solving the original model on CPLEX OPL, and taking the best solution after 5 minutes as the lower bound. As the example has only 10 customers, it gives the optimal solution on CLPLEX OPL in less than 5 minutes. So, this lower bound gives the optimal for the example as 4542.98 TL. As it will be depicted in the next chapter, this lower bound performs very well for larger problem instances.

4 COMPUTATIONAL STUDY

4.1 Experimental Design

To test our methodology for different kinds of problems, we generated 10 random problem instances for each combination of factor levels. The parameter values and how they are generated are explained below:

Number of customers:

The number of customers directly affects the problem size. The levels for number of customers are determined as 10, 15, 20, 50 and 100. First, 100 customers' data are generated for each combination and the data for less number of customers are obtained by taking the related portion.

Product type:

We assume that there are 4 different types of products and each customer can receive only one type of them. The product types are assigned to customers uniformly.

Demand:

The demand of the customers cannot exceed the total vehicle capacity which is assumed to be 16. Three different ranges are decided for the customer demands. In the first demand range D[1,16] (Uniformly distributed between 1 and 16), a customer can demand as high as the vehicle capacity. To decrease the variance without changing the mean of the demand distribution among customers, the second demand range is defined as D[4,12]. The third level has less amount of demands, and also low variance, that is D[1,8]. This level allows that at least two customer demands are carried in a vehicle.

Coordinates:

Customers are located in a large square (-200000m; +200000m) and the depot is fixed at the origin (0, 0). We divide the area in 16 equal square regions. Some regions have many customers, and some of them have a few numbers of customers. To reflect the realistic scatter of customers, the area is divided into low and high density regions. The number of low density regions is three times that of the high density regions, but the high density regions are 4 times denser than low density regions: All in all, there are 12 low density regions and 4 high density regions, as shown in **Figure 4.1**.

After the regions are fixed, customers are created randomly with respect to densities. The number of customers in the low density region is calculated by $\left\lceil \frac{total\ number\ of\ customers}{28\ (=4x4+12)} \right\rceil$. The denominator comes from the number of high density regions times the multiplier (4) adding number of low density regions.

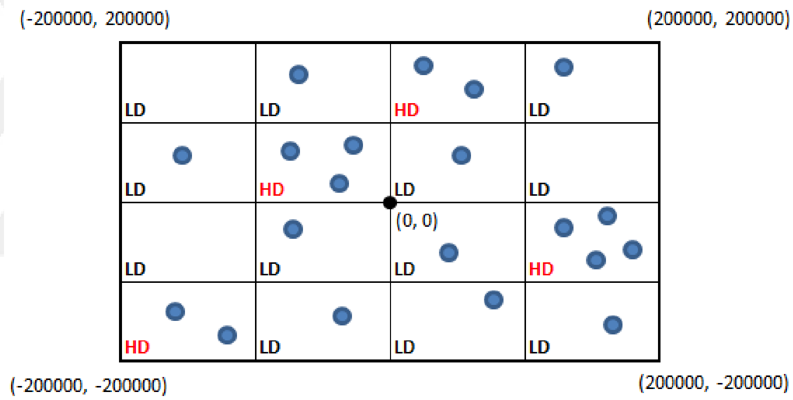

Figure 4.1 Regions and customers on the area

Number of vehicles:

The number of vehicles is not a given parameter. The total amount of demand is divided by the vehicle capacity to yield the lower bound (V). The second level is found by adding 1 to first level, and the third level (V+1) is found incrementally one more time (V+2).

The pilot experimentation is designed to evaluate the performance for the basic model including several factor levels to generate data sets. **Table 4.1** presents the factors and their levels used in the experimentation. The number of customers is set as 10 and 15, because a general solver can obtain the optimal values for these sizes. As explained above, the locations of the customers are generated uniformly over a 200,000 by 200,000 grid, and Euclidian distances are computed between each pair of nodes. The single depot is located at (0,0) coordinates. The demand of each customer is generated from a discrete uniform distribution, as explained, as well as the number of vehicles. The fuel cost is assumed as α= 1.4 TL/km in all settings, while the fixed cost of using a vehicle on a working day is set as 200 TL and 400 TL at two levels.

Table 4.1 Factors and Levels of Instances

Factors	Levels
Number of customers	10, 15
Demand	U[1,8], U[1,16], U[4,12]
Number of vehicles	*V*, *V*+1, *V*+2
Fixed cost of vehicle	200TL, 400TL

Ten instances are generated from each setting, and 360 instances in total. **Table 4.2** presents the computational results of the pilot experiment where each row represents the ten instances of the same setting. Due to the high fixed cost of the vehicles as compared to the fuel cost, the optimal solution does not use any extra vehicles and keeps the minimum feasible number of vehicles used.

Table 4.2 Analysis of Basic Model Run Results

	Avg. Obj.	Max Obj.	Min Obj.	Avg. CPU (sec.)	Max CPU (sec.)	Min CPU (sec.)	Avg. V.	Max V.	Min V.
N10_D1-16_FC400	4203.27	5455.25	3208.92	0.79	1.34	0.56	7	9	5
N10_D1-16_FC200	2903.27	3655.25	2208.92	0.65	1.31	0.30	7	9	5
N15_D1-16_FC400	6064.13	7328.18	4515.45	254.15	691.38	26.92	10	11	7
N15_D1-16_FC200	4224.13	5128.18	3115.45	493.73	2456.03	14.28	10	11	7
N10_D1-8_FC400	2609.85	2941.97	2073.28	1.16	3.52	0.28	4	4	3
N10_D1-8_FC200	1889.85	2141.97	1473.28	1.07	2.24	0.28	4	4	3
N15_D1-8_FC400	3551.98	4237.26	2806.13	330.34	686.00	9.28	5	6	4
N15_D1-8_FC200	2591.98	3037.26	2006.13	524.34	1187.32	12.11	5	6	4
N10_D4-12_FC400	4128.67	5227.14	3508.14	1.44	2.73	0.79	7	8	5
N10_D4-12_FC200	2868.67	3627.14	2452.31	1.62	2.05	0.95	7	8	5
N15_D4-12_FC400	5826.20	7237.26	4963.84	316.57	982.94	10.13	9	11	8
N15_D4-12_FC200	4099.75	5037.26	3363.84	356.22	934.25	8.63	9	11	8

As it can be seen from **Table 4.2**, the least average objective values belongs to D[1,8] settings. The logic is to load as many customer demands as possible into a vehicle. So, the route cost increases while the number of required vehicles with expensive fixed cost decreases. The number of vehicles of D[1,16] and D[4,12] settings are similar. But D[1,16] has high variance, and some vehicles can carry many customers, some of them visits only one customer, and this increases the routing cost. Therefore, D[1,16] problem instances give high objective values. Expectedly, the small amount of demands D[1,8] has significantly higher computation times on average. Because, the vehicles can carry many number of customers' demands, and this increases the possible assignment combinations and causes increasing CPU times. D[4,12] yields the fastest results, as the amount of demands are high and the variance is low. The customer demands are assigned to vehicles easily. The average CPU time of setting D[1,16] is better than D[1,8] due to high demand but worse than D[4,12] due to high variance. As it can be seen in **Table 4.2**, the combinatorial nature of the problem prevails when the number of customers is increased. CPU times and objective values also increase, and most of the problem instances cannot be solved in reasonable times.

To test the quality of the heuristic algorithms, the controllable factors and their levels are determined as shown in **Table 4.3**.

Table 4.3 Factors and Levels for Heuristic Algorithm

Factors	Levels
Number of replications	1, 10, 100, 500, 1000
Number of replications for improvement	1, 2

RSH_1 and RSH_2's construction heuristics are not smart algorithms, and to find the better solutions, the algorithms try many alternative solutions with random selections and assignments. When the number of replications increases, the amount of alternative solutions also increase with increasing the CPU time. **Table 4.4** shows the CPU times and GAP% ($\frac{\text{heuristic result-optimal result}}{\text{optimal result}}$) between optimal and RSH_1 heuristic results.

Table 4.4 Results of RSH_1

	# of Replications									
	1		10		100		500		1000	
	GAP	CPU (sec.)	GAP	CPU (sec.)	GAP	CPU (sec.)	GAP	CPU (sec.)	GAP	CPU (sec.)
N10_D1-16_FC400	0.00%	0.00	0.00%	0.00	0.00%	0.00	0.00%	0.04	0.00%	0.07
N10_D1-16_FC200	0.00%	0.00	0.00%	0.00	0.00%	0.02	0.00%	0.06	0.00%	0.11
N15_D1-16_FC400	4.00%	0.00	2.21%	0.00	1.40%	0.01	0.00%	0.08	0.00%	0.12
N15_D1-16_FC200	6.00%	0.00	3.22%	0.00	2.04%	0.01	0.00%	0.05	0.00%	0.13
N10_D1-8_FC400	14.00%	0.00	3.20%	0.00	0.00%	0.02	0.00%	0.10	0.00%	0.07
N10_D1-8_FC200	19.00%	0.00	4.52%	0.00	0.00%	0.00	0.00%	0.04	0.00%	0.07
N15_D1-8_FC400	18.00%	0.00	10.74%	0.00	1.12%	0.03	0.90%	0.12	0.00%	0.16
N15_D1-8_FC200	17.00%	0.00	10.52%	0.00	1.05%	0.01	0.35%	0.07	0.00%	0.18
N10_D4-12_FC400	0.00%	0.00	0.00%	0.00	0.00%	0.01	0.00%	0.05	0.00%	0.06
N10_D4-12_FC200	0.00%	0.00	0.00%	0.00	0.00%	0.00	0.00%	0.06	0.00%	0.07
N15_D4-12_FC400	3.00%	0.00	0.72%	0.00	0.40%	0.01	0.00%	0.06	0.00%	0.13
N15_D4-12_FC200	4.00%	0.00	1.03%	0.00	0.58%	0.01	0.00%	0.05	0.00%	0.14

As seen in the **Table 4.4**, when the number of replications increases, the GAP% value decreases as expected. We decided to fix the number of replications as 500 and 1000 to have the better solutions.

At the end of the each construction step, to improve the current tours, our improvement step is applied. Our modified VNS algorithm provides the better routes in each movement of the customers for both the chosen vehicle and found vehicle routes. The number of iterations in VNS is determined by the number of customers.

Increasing the number of replications of the improvement step affects solution quality. **Figure 4.2** shows how improvement step replication size changes the solution. Customer X is in vehicle A, customers Y and Z are in vehicle B, and customer T is in vehicle C. In the discharging steps of vehicle A, customer X does not move to vehicle B because of increasing route cost in the first step. After vehicle B tries to discharge all its customers, only customer Y moves brings saving. But at this point, one should consider customer X. If customer Y has not been in vehicle B at the beginning, customer X could come to vehicle B. So, the improvement step should be double checked.

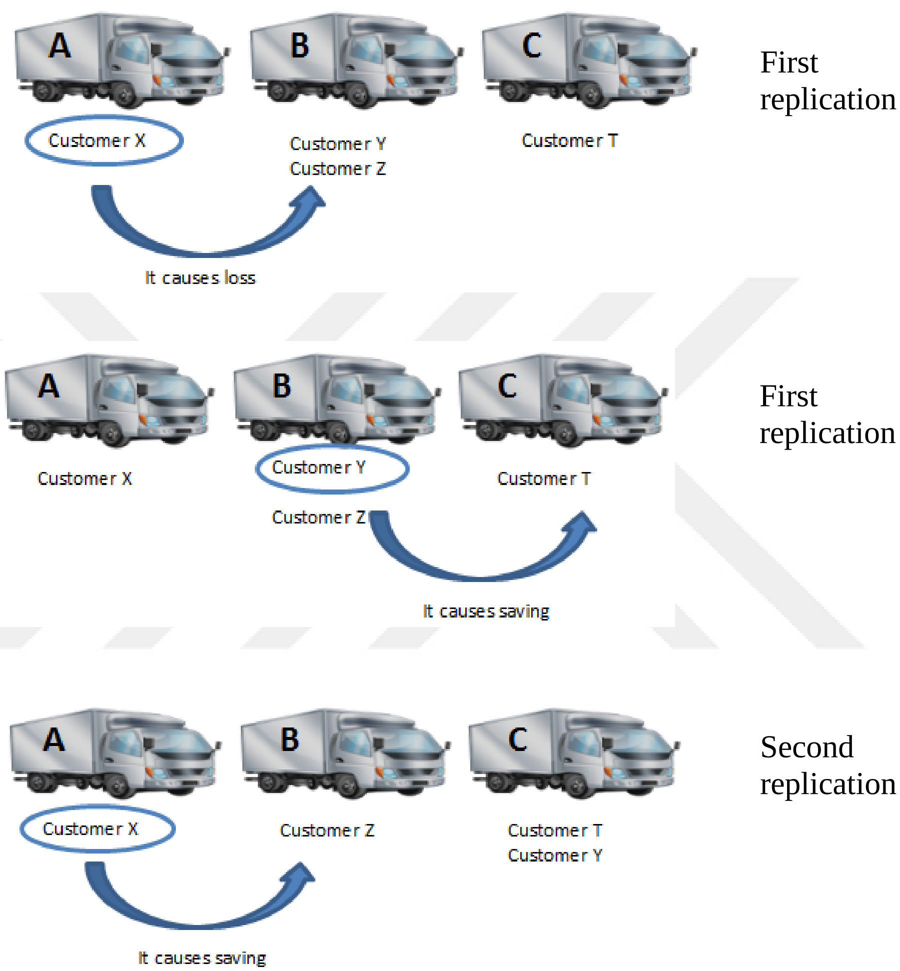

Figure 4.2 Example of Improvement Replication Effect

The results for increasing replication size of the improvement step are shown in **Table 4.5**.

Table 4.5 Results of Replications for Improvement

	Improvement Replication - 1				Improvement Replication - 2			
	500		1000		500		1000	
	GAP	CPU(sec.)	GAP	CPU(sec.)	GAP	CPU(sec.)	GAP	CPU(sec.)
N10_D1-16_FC400	0.00%	0.07	0.00%	0.13	0.00%	0.08	0.00%	0.16
N10_D1-16_FC200	0.00%	0.05	0.00%	0.13	0.00%	0.10	0.00%	0.18
N15_D1-16_FC400	0.00%	0.22	0.00%	0.38	0.00%	0.26	0.00%	0.53
N15_D1-16_FC200	0.00%	0.22	0.00%	0.39	0.00%	0.28	0.00%	0.65
N10_D1-8_FC400	0.00%	0.20	0.00%	0.36	0.00%	0.33	0.00%	0.58
N10_D1-8_FC200	0.00%	0.16	0.00%	0.36	0.00%	0.33	0.00%	0.56
N15_D1-8_FC400	0.02%	0.68	0.00%	1.59	0.01%	1.13	0.00%	2.25
N15_D1-8_FC200	0.01%	0.90	0.00%	1.34	0.01%	1.31	0.00%	2.12
N10_D4-12_FC400	0.00%	0.02	0.00%	0.05	0.00%	0.04	0.00%	0.05
N10_D4-12_FC200	0.00%	0.02	0.00%	0.04	0.00%	0.02	0.00%	0.10
N15_D4-12_FC400	0.00%	0.06	0.00%	0.15	0.00%	0.16	0.00%	0.15
N15_D4-12_FC200	0.00%	0.07	0.00%	0.17	0.00%	0.09	0.00%	0.15

Although the double checking of improvement phase does not create a big difference for 10 and 15 customers, it is useful for larger size problems.

According to the test results of the factor levels for the solution method, **Table 4.6** gives the summary of the selected levels for evaluating the performance of our heuristics.

Table 4.6 Reduced Factor Levels for Heuristics

Factors	Levels
Number of replications	500, 1000
Number of replications for improvement	2

4.2 Computational Analysis

The basic, multi-trip and split delivery models are solved with generated data sets under determined levels on the general purpose solver OPL for optimal solutions, and C++ compiler for heuristic solutions. All runs are made on a PC with Intel® Core™ İ5-3360M CPU @ 2.80GHz and 8.00 GB RAM. The results are summarized as average, maximum and minimum values for 10 problem instances at each setting. The tables for maximum and minimum values are provided in Appendix 1. The average values are shown and discussed in subsections below.

4.2.1 Basic Model

The optimal solutions of the basic model can be obtained up to 20 customers. **Table 4.7** represents the computational results of average optimal values, CPU times, lower bounds and their gaps from optimal value, four heuristic results with GAPs% and their CPU times.

The table shows that, both lower bound-1 and lower bound-2 values are not close to the optimal values. The best lower bound is obtained mostly from lower bound-2. RSH_1 and RSH_2 heuristics reach the optimal values for each problem variants within reasonable times only. Since 1000 replication results give better results with respect to 500 replications, we show the results for 1000 replications from this point onwards.

Table 4.7 Summary of the Basic Model Results for 10 Customers

			AVG					
			N10_D1-16_FC400	N10_D1-16_FC200	N10_D1-8_FC400	N10_D1-8_FC200	N10_D4-12_FC400	N10_D4-12_FC200
OPT			4203.27	2903.27	2609.85	1889.85	4128.67	2868.67
CPU(sec)			0.79	0.65	1.16	1.07	1.44	1.62
LB_1			3606.03	2346.03	2008.06	1288.06	3355.84	2175.84
GAP_LB_1			17.23%	25.14%	28.51%	44.76%	23.27%	32.45%
LB_2			3618.13	2358.13	2129.59	1409.59	3398.46	2218.46
GAP_LB_2			16.62%	24.11%	21.13%	32.07%	21.57%	29.66%
BEST LB			3642.88	2382.88	2139.02	1419.02	3410.75	2230.75
BEST GAP			15.98%	23.09%	20.65%	31.31%	21.14%	28.99%
RSH_1	# of Rep	1000	4203.27	2903.27	2609.85	1889.85	4128.67	2868.67
		GAP	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		CPU (sec)	0.23	0.24	0.74	0.69	0.10	0.12
RSH_2	# of Rep	1000	4203.27	2903.27	2609.85	1891.51	4128.67	2868.67
		GAP	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		CPU (sec)	0.42	0.43	0.70	0.70	0.30	0.30
SSH_1	# of Rep	1	4325.44	2997.70	2844.30	2043.65	4230.18	2970.18
		GAP	0.03%	0.03%	0.09%	0.08%	0.03%	0.04%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
SSH_2	# of Rep	1	4271.62	2971.62	2813.05	2073.05	4239.86	2979.86
		GAP	0.02%	0.03%	0.08%	0.11%	0.03%	0.04%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
BEST UB			4203.27	2903.27	2609.85	1889.85	4128.67	2868.67
BEST GAP			0.00%	0.00%	0.00%	0.00%	0.00%	0.00%

The heuristic results for all customer sizes demonstrate similar characteristics with the optimal results in **Table 4.2**. Decreasing fixed cost from 400TL to 200TL provide positive effect on the objective value. The high demand variance D[1,16] causes increasing costs with respect to low variance D[4.12]. In addition, low demand D[1,8] decreases the objective value under increasing CPU times. When the number of customers is increased to 15, the instances can still be solved by the solver optimally. **Table 4.8** shows the results of optimal, lower bound and heuristic results.

Table 4.8 Summary of the Basic Model results for 15 customers

			AVG					
			N15_D1-16_FC400	**N15_D1-16_FC200**	**N15_D1-8_FC400**	**N15_D1-8_FC200**	**N15_D4-12_FC400**	**N15_D4-12_FC200**
OPT			6064.13	4224.13	3551.98	2591.98	5862.20	4099.75
CPU (sec)			152.15	473.73	330.34	524.34	366.57	296.22
LB_1			4881.85	3161.85	2676.93	1716.93	4562.40	2942.40
GAP_LB_1			24.37%	33.98%	29.30%	45.81%	28.51%	39.55%
LB_2			4949.19	3229.19	2872.09	1912.09	4659.00	3039.00
GAP_LB_2			22.58%	31.00%	20.39%	30.66%	25.84%	35.10%
BEST LB			4949.19	3229.19	2872.09	1912.09	4659.00	3039.00
BEST GAP			22.58%	31.00%	20.39%	30.66%	25.84%	35.10%
RSH_1	**# of Rep**	**1000**	6064.13	4224.13	3566.07	2613.48	5862.20	4102.25
		GAP	0.00%	0.00%	0.37%	0.85%	0.00%	0.07%
		CPU (sec)	0.73	0.76	2.77	2.73	0.29	0.29
RSH_2	**# of Rep**	**1000**	6077.04	4224.13	3580.03	2630.04	5862.20	4099.75
		GAP	0.20%	0.00%	0.78%	1.51%	0.00%	0.00%
		CPU (sec)	0.96	0.97	2.09	2.07	0.63	0.66
SSH_1	**# of Rep**	**1**	6302.62	4410.08	3937.07	2899.32	6171.19	4351.19
		GAP	4.16%	4.44%	11.06%	11.89%	5.41%	6.35%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
SSH_2	**# of Rep**	**1**	6287.39	4447.39	4003.26	3023.26	6210.32	4390.32
		GAP	4.08%	5.87%	12.66%	16.64%	6.05%	7.25%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
BEST UB			6064.13	4224.13	3557.39	2605.96	5862.20	4099.75
BEST GAP			0.00%	0.00%	0.14%	0.52%	0.00%	0.00%

The lower bound gaps increase with increasing number of customers. Also, heuristic result gaps increase a little bit for D[1-8] settings. As there are many possible assignment combinations to get the optimal value, the heuristic could not catch the optimal among these many combinations.

The summary results for 20 customers are given in **Table 4.9**. For 20 customers, we cannot obtain any optimal values, and the heuristic results are compared with the best lower bound ones. As we mentioned before, the lower bound values are not satisfactory to be used instead of the optimal values. Therefore, we can say that the large gaps between the best lower bound and heuristic results occur because of the weakness of the lower bounds. At this point, we use lower bound-3, obtained from the CPLEX solver with a time limit of 5 minutes. The gaps from the optimal values are given by CPLEX, and it has better values with respect to lower bound-1 and lower bound-2. To indicate the difference of lower bound-3 from other two lower bounds, "GAP (w/o CPLEX)" is shown for best gap of lower bound-1 and lower bound-2. For instance, as it seen in **Table 4.9,** "N20_D1-16 FC400" of "BEST GAP" states lower bound-3 and equal to 10.40%. However, the gap between lower bound-3 and the optimal value is shown as 9.81%. It means that the "BEST GAP" would be less if the lower bound-3 were more effective. The value of "GAP (w/o CPLEX)" is worse, because of lower bound-1 and lower bound-2 are weaker than lower bound-3. From this perspective, the performance of the heuristic algorithms can be assessed more clearly.

Table 4.9 Summary of the Basic Model Results for 20 Customers

			AVG					
			N20_D1-16 FC400	**N20_D1-16-FC200**	**N20_D1-8-FC400**	**N20_D1-8-FC200**	**N20_D4-12-FC400**	**N20_D4-12-FC200**
LB_3			7365.21	5011.02	4045.84	2780.99	6883.86	4645.79
GAP_LB_3			9.81%	11.10%	13.74%	18.65%	16.07%	17.24%
LB_1			6608.95	4268.95	3512.20	2252.20	6058.46	3898.46
LB_2			6712.70	4372.70	3784.35	2524.35	6190.68	4030.68
BEST LB			7405.00	5028.19	4045.84	2780.99	6883.86	4645.79
RSH_1	**# of Rep**	**1000**	8166.29	5644.27	4731.62	3486.67	7890.56	5481.78
		GAP	10.40%	12.36%	16.95%	25.43%	14.42%	17.86%
		CPU (sec)	1.46	1.42	5.85	6.19	0.55	0.55
RSH_2	**# of Rep**	**1000**	8167.18	5653.87	4766.04	3462.61	7891.36	5483.30
		GAP	11.31%	14.29%	19.07%	27.19%	16.09%	20.65%
		CPU (sec)	1.21	1.56	5.91	5.81	0.66	0.69
SSH_1	**# of Rep**	**1**	8419.31	5968.45	5188.74	3877.16	8265.51	5805.51
		GAP	15.05%	21.42%	30.21%	43.92%	22.23%	28.78%
		CPU (sec)	0.00	0.00	0.01	0.01	0.00	0.00
SSH_2	**# of Rep**	**1**	8562.89	6002.89	5294.53	4045.63	8299.00	5826.69
		GAP	17.40%	22.56%	33.19%	50.44%	22.85%	29.43%
		CPU (sec)	0.00%	0.00%	0.20%	0.50%	0.00%	0.00%
BEST UB			8163.98	5644.19	4726.71	3448.66	7885.71	5474.65
BEST GAP			10.40%	12.36%	16.95%	24.91%	14.42%	17.86%
GAP (w/o LB_3)			21.84%	29.39%	24.95%	36.86%	27.17%	35.66%

Table 4.10 and **Table 4.11** show the results of 50 and 100 customers for the basic model. CPLEX could not find a solution in 5 minutes, so there is no lower bound-3 value. The gaps and also CPU times increase with the number of customers.

Table 4.10 Summary of the Basic Model Results for 50 Customers

			AVG					
			N50_D1-16_FC400	N50_D1-16_FC200	N50_D1-8_FC400	N50_D1-8_FC200	N50_D4-12_FC400	N50_D4-12_FC200
LB_1			15237.85	9797.85	7953.30	5013.30	14267.32	9147.32
LB_2			15418.79	9980.42	8486.40	5541.80	14496.48	9376.93
BEST LB			15435.94	9996.70	8486.40	5541.80	14501.71	9381.65
RSH_1	**# of Rep**	**1000**	19482.46	13622.82	11812.80	8796.98	19024.30	13422.50
		GAP	26.43%	36.74%	39.47%	59.42%	31.35%	43.51%
		CPU (sec)	32.07	32.53	147.57	147.62	9.33	9.57
RSH_2	**# of Rep**	**1000**	19411.24	13633.43	11794.71	8756.93	19016.03	13422.76
		GAP	25.97%	36.88%	39.30%	58.68%	31.31%	43.47%
		CPU (sec)	24.09	24.37	132.61	133.41	8.22	8.28
SSH_1	**# of Rep**	**1**	20164.71	14187.63	11960.76	8708.81	19909.32	14018.08
		GAP	30.87%	42.48%	41.23%	57.88%	37.50%	49.88%
		CPU (sec)	0.04	0.03	0.14	0.16	0.01	0.00
SSH_2	**# of Rep**	**1**	20500.95	14440.95	12798.17	9705.74	20248.32	14368.32
		GAP	33.07%	45.01%	51.12%	75.77%	39.77%	53.56%
		CPU (sec)	0.01	0.01	0.14	0.14	0.00	0.00
BEST UB			19374.19	13587.46	11701.86	8612.46	18930.74	13379.02
BEST GAP			25.73%	36.41%	38.14%	56.00%	30.71%	43.02%

Table 4.11 Summary of the Basic Model Results for 100 Customers

			AVG					
			N100_D1-16-FC400	N100_D1-16-FC200	N100_D1-8-FC400	N100_D1-8-FC200	N100_D4-12-FC400	N100_D4-12-FC200
LB_1			29546.83	18866.83	15122.83	9402.83	27777.14	17677.14
LB_2			29919.05	19239.21	16088.74	10347.50	28232.56	18129.93
BEST LB			29950.29	19268.77	16088.74	10347.50	28242.59	18138.97
RSH_1	**# of Rep**	**1000**	39328.37	27351.65	24128.48	17655.12	37987.49	26820.95
		GAP	31.61%	42.39%	50.07%	71.02%	34.63%	48.23%
		CPU (sec)	318.83	322.14	1531.01	1948.37	96.12	94.06
RSH_2	**# of Rep**	**1000**	39356.18	27379.98	24103.54	17636.28	37944.45	26858.94
		GAP	31.71%	42.53%	49.94%	70.81%	34.49%	48.46%
		CPU (sec)	257.26	259.35	1417.76	1732.29	77.26	77.36
SSH_1	**# of Rep**	**1**	41749.83	28916.59	25062.22	18145.16	40610.19	28619.56
		GAP	39.69%	50.51%	55.87%	75.76%	43.94%	58.20%
		CPU (sec)	0.29	0.33	1.37	1.39	0.07	0.06
SSH_2	**# of Rep**	**1**	42367.11	30661.91	27132.10	22913.34	39890.24	30162.60
		GAP	41.81%	59.72%	68.56%	121.95%	41.46%	66.84%
		CPU (sec)	0.27	0.28	1.14	1.15	0.11	0.09
BEST UB			38598.23	27293.73	23413.92	17213.80	36551.42	26766.31
BEST GAP			29.13%	42.07%	45.59%	66.84%	29.56%	47.94%

4.2.2 Multi-Trip Model

In multi-trip model, multiple visits to customers are allowed and we assume that a vehicle can have two distinct tours at most. As expected, multi-trip model results for the same problems give better objective values. If two distinct tours are assigned to a single vehicle, one of the fixed costs will be saved. Multi-trip model is more complex than the basic model. So, the CPU times of our general solver are increased, and the solver cannot give an optimal solution for 15 customers in reasonable times. By using the same data set for 10 customers, we solved 10 instances from each setting, and the summary of the results are shown in **Table 4.12.**

Table 4.12 Summary of the Multi-Trip Model Results for 10 Customers

			AVG					
			N10_D1-16_FC400	**N10_D1-16_FC200**	**N10_D1-8_FC400**	**N10_D1-8_FC200**	**N10_D4-12_FC400**	**N10_D4-12_FC200**
OPT			3027.53	2323.27	1991.12	1571.12	2932.46	2252.46
CPU (sec)			40.78	44.20	12.14	14.70	44.83	37.59
LB_1			2446.03	1766.03	1328.06	1008.06	2235.84	1615.84
GAP_LB_1			25.84%	33.93%	52.04%	59.82%	31.13%	39.99%
LB_2			2458.13	1778.13	1449.59	1129.59	2278.46	1658.46
GAP_LB_2			24.93%	32.51%	38.73%	41.96%	28.58%	36.22%
BEST LB			2482.88	1802.88	1459.02	1139.02	2290.75	1670.75
BEST GAP			23.99%	31.17%	37.96%	40.86%	27.88%	35.27%
RSH_1	**# of Rep**	**1000**	3027.53	2323.27	1997.52	1577.52	2954.74	2274.30
		GAP	0.00%	0.00%	0.32%	0.40%	0.63%	0.80%
		CPU (sec)	0.27	0.26	0.70	0.73	0.08	0.07
RSH_2	**# of Rep**	**1000**	3027.53	2323.27	1999.77	1579.77	2954.74	2274.30
		GAP	0.00%	0.00%	0.44%	0.56%	0.63%	0.80%
		CPU (sec)	0.33	0.37	0.59	0.57	0.16	0.17
SSH_1	**# of Rep**	**1**	3165.44	2417.70	2536.77	1896.12	3110.18	2410.18
		GAP	3.86%	3.89%	20.89%	16.87%	5.66%	6.51%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
SSH_2	**# of Rep**	**1**	3151.62	2411.62	2452.61	1892.61	3159.86	2439.86
		GAP	3.51%	3.76%	17.41%	16.26%	6.67%	7.14%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
BEST UB			3043.27	2323.27	1997.52	1577.52	2954.73	2274.30
BEST GAP			0.00%	0.00%	0.32%	0.40%	0.62%	0.80%

When the multi-trip model is compared with the basic model, the multi-trip model lower bounds are worse, but the heuristics have small gaps from optimal values that can be ignorable, and the CPU times are a bit more. Unexpectedly, D[4,12] settings have the highest gaps for multi-trip models. When we consider the heuristic algorithm of the multi-trip model, the combination of vehicles at the end of the improvement phase could not match the two vehicles' routes properly. The general solver tries to combine two vehicles route under time constraints. But, the heuristic algorithm does not try to combine routes until the last part of the solution. The setting is very similar to D[1,16] because of their close mean values and the objective values are also close. However, D[1,16] setting has no gap. The propensity of the D[1,16] is a little bit different.

To show the reason of the gap difference between D[1,16] and D[4,12] models, the following example can be useful. The D[1,16] and D[4,12] problems use similar numbers of vehicles. So, we can assume that the problem of both D[1,16] and D[4,12] heuristic solutions have 6 vehicles and the vehicles routing times (distances) are gives as T(D[1,16])={400, 400, 240, 240, 80, 80} with unbalanced routes due to high variance demands and T(D[4,12])={250, 250, 240, 240, 310, 80}. At the end of the improvement step of each solution, the vehicles' routes are able to combine under the time constraint. The heuristic algorithm combines vehicles such as T*(D[1,16])={480, 480, 480} and T*(D[4,12])={330, 250, 480, 310}. It shows that the heuristic algorithms of the D[1,16] problems is able to combine vehicles easily with respect to D[1,8] and gets nearer to the optimal results.

Table 4.13 Summary of the Multi-Trip Model Results for 15 Customers

			AVG					
			N15_D1-16_FC400	**N15_D1-16_FC200**	**N15_D1-8_FC400**	**N15_D1-8_FC200**	**N15_D4-12_FC400**	**N15_D4-12_FC200**
LB_3			4041.04	3111.46	2446.56	1875.67	3578.26	2709.76
GAP_LB_3			6.28%	8.28%	8.52%	10.72%	12.53%	14.10%
LB_1			3281.85	2361.85	1897.45	1656.93	3002.40	2162.40
LB_2			3349.19	2429.19	2032.09	1791.57	3099.00	2259.00
BEST LB			4041.04	3111.46	2446.56	1976.70	3615.92	2712.85
RSH_1	**# of Rep**	**1000**	4366.47	3392.07	2772.30	2194.07	4192.35	3252.40
		GAP	8.26%	9.34%	13.69%	11.35%	16.30%	20.69%
		CPU (sec)	0.62	0.62	2.25	2.26	0.21	0.21
RSH_2	**# of Rep**	**1000**	4389.02	3384.13	2804.92	2224.45	4192.35	3252.40
		GAP	8.84%	8.98%	14.97%	12.83%	16.30%	20.69%
		CPU (sec)	0.69	0.68	1.93	1.97	0.36	0.36
SSH_1	**# of Rep**	**1**	4582.62	3590.08	3242.00	2502.00	4506.34	3570.06
		GAP	13.85%	15.78%	32.25%	26.21%	24.75%	32.46%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
SSH_2	**# of Rep**	**1**	4572.28	3592.28	3723.31	2883.31	4678.20	3618.20
		GAP	13.91%	16.46%	52.37%	45.89%	29.87%	34.20%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
BEST UB			4366.47	3384.13	2766.81	2188.98	4128.34	3252.40
BEST GAP			8.26%	8.98%	13.47%	11.02%	14.41%	20.69%
BEST GAP (w/o CPLEX)			31.66%	40.93%	38.41%	28.45%	33.39%	44.48%

The basic model could be solved optimally for 15 customers. However the multi-trip model formulation size is more complex than the basic model, hence it could not be solved with 15 customers. We used lower bound-3 instead of optimal values to compare the heuristic results. As seen **Table 4.13**, the gaps between heuristic and lower bound-3 values are close to the gaps between optimal and CPLEX results. This shows that the heuristic gaps would be less if they were compared with the optimal results.

Table 4.14, Table 4.15 and **Table 4.16** show the results for 20, 50 and 100 customers of the multi-trip model. The CPLEX could not find a feasible solution in 5 minutes, so there is no lower bound-3 value. The gaps and also CPU times are increased with increased number of customers. As large sized problems have high CPU times, smart start heuristic solutions may be more reasonable to use in operational decisions.

Table 4.14 Summary of the Multi-Trip Model Results for 20 Customers

			AVG					
			N20_D1-16_FC400	N20_D1-16_FC200	N20_D1-8_FC400	N20_D1-8_FC200	N20_D4-12_FC400	N20_D4-12_FC200
LB_3			4977.84	3706.27	2809.56	2134.13	4533.52	3370.66
GAP_LB_3			16.90%	19.16%	22.62%	26.88%	23.87%	27.42%
LB_1			5008.95	3468.95	2672.20	1832.20	4498.46	3118.46
LB_2			5112.70	3572.70	2944.35	2104.35	4630.68	3250.68
BEST LB			5272.24	3798.63	2981.65	2167.06	4645.07	3373.82
RSH_1	# of Rep	1000	5779.77	4387.75	3728.09	2992.91	5555.28	4286.54
		GAP	9.86%	15.67%	25.55%	38.39%	19.59%	27.01%
		CPU (sec)	1.29	1.29	5.87	5.90	0.49	0.49
RSH_2	# of Rep	1000	5780.65	4398.93	3761.47	2940.47	5558.04	4300.19
		GAP	9.86%	15.98%	26.64%	36.08%	19.67%	27.42%
		CPU (sec)	1.24	1.31	5.38	5.41	0.64	0.64
SSH_1	# of Rep	1	6101.18	4748.45	4268.74	3437.16	6105.51	4725.51
		GAP	15.83%	25.02%	43.97%	58.69%	31.69%	40.21%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
SSH_2	# of Rep	1	6370.79	4862.89	4683.91	3715.00	6179.00	4766.69
		GAP	21.50%	28.42%	57.84%	72.04%	33.17%	41.59%
		CPU (sec)	0.00	0.00	0.01	0.00	0.00	0.00
BEST UB			5777.45	4387.66	3714.31	2918.22	5549.37	4286.54
BEST GAP			9.80%	15.66%	25.07%	34.97%	19.46%	27.01%
GAP (w/o CPLEX)			13.86%	23.68%	27.15%	39.59%	19.84%	31.87%

Table 4.15 Summary of the Multi-Trip Model Results for 50 Customers

			AVG					
			N50_D1-16_FC400	N50_D1-16_FC200	N50_D1-8_FC400	N50_D1-8_FC200	N50_D4-12_FC400	N50_D4-12_FC200
LB_1			9917.85	7137.85	4713.30	3393.30	9267.32	6647.32
LB_2			10098.79	7320.42	5246.40	3921.80	9496.48	6876.93
BEST LB			10115.94	7336.70	5246.40	3921.80	9501.71	6881.65
RSH_1	# of Rep	1000	16085.11	10673.81	9598.22	7395.24	13403.39	10461.16
		GAP (LB)	59.14%	45.85%	87.02%	91.30%	41.51%	52.79%
		CPU (sec)	42.50	39.95	161.79	156.57	9.95	9.68
RSH_2	# of Rep	1000	16309.06	10765.97	9774.84	7474.91	13500.05	10544.09
		GAP (LB)	61.19%	47.10%	90.49%	93.68%	42.54%	53.91%
		CPU (sec)	39.60	36.74	160.52	154.73	9.46	9.08
SSH_1	# of Rep	1	15160.95	11006.33	9146.29	6799.30	13741.36	10818.35
		GAP (LB)	49.71%	50.55%	77.17%	74.32%	44.93%	57.78%
		CPU (sec)	0.05	0.04	0.18	0.19	0.01	0.01
SSH_2	# of Rep	1	15522.40	11345.74	10553.62	8338.61	14259.00	11213.08
		GAP(LB)	53.26%	55.17%	103.74%	113.83%	50.38%	63.59%
		CPU (sec)	0.02	0.03	0.18	0.19	0.00	0.00
BEST UB			13934.55	10427.76	8888.21	6652.51	12953.38	10163.18
BEST GAP			37.67%	42.58%	71.98%	70.47%	36.80%	48.40%

Table 4.16 Summary of the Multi-Trip Model Results for 100 Customers

			AVG					
			N100_D1-16_FC400	N100_D1-16_FC200	N100_D1-8_FC400	N100_D1-8_FC200	N100_D4-12_FC400	N100_D4-12_FC200
LB_1			18946.83	13566.83	9482.83	6582.83	17775.18	12675.18
LB_2			19319.05	13939.21	10448.74	7527.50	18254.57	13150.73
BEST LB			19350.29	13968.77	10448.74	7527.50	18264.60	13159.77
RSH_1	# of Rep	1000	27281.09	21262.75	19320.14	14709.69	26273.90	20606.17
		GAP(LB)	41.01%	52.61%	85.49%	96.18%	44.25%	57.25%
		CPU (sec)	426.08	412.21	1594.80	1831.80	101.09	98.08
RSH_2	# of Rep	1000	27350.35	21325.98	19444.02	14775.08	26277.64	20652.06
		GAP(LB)	41.37%	53.06%	86.67%	97.00%	44.26%	57.63%
		CPU (sec)	370.63	340.89	1508.46	1745.27	83.01	81.01
SSH_1	# of Rep	1	28757.34	22348.64	19532.51	14916.00	27898.58	21848.32
		GAP(LB)	48.60%	60.39%	87.32%	98.41%	53.15%	66.73%
		CPU (sec)	0.30	0.30	1.50	1.50	0.05	0.04
SSH_2	# of Rep	1	29516.02	23098.26	21544.86	16491.91	28691.30	22505.60
		GAP(LB)	52.53%	65.77%	106.87%	119.86%	57.62%	71.89%
		CPU (sec)	0.36	0.25	1.28	1.33	0.07	0.06
BEST UB			27227.37	21202.66	18408.73	13999.02	26148.80	20556.25
BEST GAP			40.74%	52.16%	76.72%	86.63%	43.56%	56.89%

4.2.3 Split Delivery Model

The split delivery model is the most complex model of the three. Some of the customer demands are splitted into two parts and carried on different vehicles. Both optimal and heuristic solutions take more CPU time than the other two models. **Table 4.17** shows the summary results of split delivery model for 10 customers.

Table 4.17 Summary of the Split Delivery Model Results for 10 Customers

			AVG					
			N10_D1-16_FC400	N10_D1-16_FC200	N10_D1-8_FC400	N10_D1-8_FC200	N10_D4-12_FC400	N10_D4-12_FC200
OPT			4038.80	2818.80	2595.59	1875.59	3873.46	2693.46
OPT_GAP			1.98%	4.10%	0.00%	0.00%	2.68%	2.65%
CPU (sec)			921.75	1300.21	216.12	297.90	1012.96	843.32
LB_1			3526.03	2306.03	2008.06	1288.06	3355.84	2175.84
GAP_LB_1			15.22%	23.50%	30.41%	47.82%	15.74%	24.37%
LB_2			3538.13	2318.13	2129.59	1409.59	3398.46	2218.46
GAP_LB_2			14.71%	22.60%	22.88%	34.77%	14.20%	21.85%
BEST LB			3642.88	2318.13	2129.59	1409.59	3398.46	2218.46
BEST GAP			13.97%	21.46%	22.41%	34.01%	13.82%	21.23%
RSH_1	**# of Rep**	**1000**	4088.76	2848.75	2601.57	1881.57	3934.49	2731.40
		GAP	1.21%	1.01%	0.22%	0.31%	1.50%	1.45%
		CPU (sec)	0.25	0.28	0.63	0.85	0.39	0.17
RSH_2	**# of Rep**	**1000**	4088.76	2848.75	2601.57	1881.57	3934.49	2731.40
		GAP	1.21%	1.01%	0.22%	0.31%	1.50%	1.45%
		CPU (sec)	0.24	0.28	0.63	0.84	0.38	0.17
SSH_1	**# of Rep**	**1**	4265.94	2964.59	2730.39	1973.54	4282.17	2845.98
		GAP	5.45%	4.96%	5.12%	5.11%	10.33%	5.69%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
SSH_2	**# of Rep**	**1**	4296.15	2985.56	2749.72	1987.50	4312.76	2867.05
		GAP	6.20%	5.70%	5.87%	5.85%	11.12%	6.47%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
BEST UB			4088.76	2848.75	2601.57	1881.57	3934.49	2731.40
BEST GAP			1.21%	1.01%	0.22%	0.31%	1.50%	1.45%

There is no gap for both basic and multi-trip models with 10 customers, but split delivery heuristic results have small gaps because of problem complexity. Unexpectedly, split delivery gaps of small size problem instances behave differently for D[1-8] demand level. While D[1-8] demand has the highest gaps among other demand types in basic and multi-trip model results, split delivery gaps of small size problem instances have less gaps than other demand types shown as in **Table 4.17, Table 4.18** and **Table 4.19.**

Table 4.18 Summary of the Split Delivery Model Results for 15 Customers

			AVG					
			N15_D1-16_FC400	N15_D1-16_FC200	N15_D1-8_FC400	N15_D1-8_FC200	N15_D4-12_FC400	N15_D4-12_FC200
LB_1			4841.85	3141.85	2676.93	1716.93	4562.40	2942.40
LB_2			4909.19	3209.19	2872.09	1912.09	4659.00	3039.00
LB_3			4951.85	3344.85	3135.32	2222.96	4746.28	3149.00
GAP_LB_3			16.24%	17.85%	9.41%	10.97%	13.12%	19.23%
BEST LB			5019.34	3355.47	3100.33	2184.72	4813.75	3183.52
RSH_1	# of Rep	1000	5873.45	4123.28	3531.92	2614.34	5491.78	3929.83
		GAP	17.24%	23.34%	14.25%	20.26%	14.06%	23.53%
		CPU (sec)	1.21	1.23	3.34	3.27	0.62	0.61
RSH_2	# of Rep	1000	5901.44	4135.05	3578.88	2602.31	5527.98	3931.19
		GAP	17.82%	23.71%	15.66%	19.69%	14.80%	23.54%
		CPU (sec)	1.27	1.29	2.92	2.96	0.75	0.76
SSH_1	# of Rep	1	6199.66	4391.89	3831.87	2921.17	6053.90	4340.98
		GAP	23.80%	31.39%	24.29%	34.83%	26.01%	36.89%
		CPU (sec)	0.00	0.00	0.00	0.10	0.00	0.00
SSH_2	# of Rep	1	6255.03	4455.03	3984.71	3004.71	6074.27	4393.63
		GAP	25.08%	33.54%	28.60%	37.96%	26.34%	38.27%
		CPU (sec)	0.00	0.00	0.00	0.20	0.00	0.00
BEST UB			5868.56	4123.28	3525.93	2591.71	5490.53	3926.83
BEST GAP			17.14%	23.34%	14.02%	19.24%	14.03%	23.42%
BEST GAP (wo CPLEX)			19.94%	29.25%	23.44%	36.73%	17.87%	29.31%

Table 4.19 Summary of the Split Delivery Model Results for 20 Customers

			AVG					
			N20_D1-16_FC400	N20_D1-16_FC200	N20_D1-8_FC400	N20_D1-8_FC200	N20_D4-12_FC400	N20_D4-12_FC200
LB_1			6488.95	4208.95	3512.20	2252.20	6058.46	3898.46
LB_2			6592.70	4312.70	3784.35	2524.35	6190.68	4030.68
LB_3			6506.68	4240.49	3915.37	2628.61	6015.57	3931.81
GAP_LB_3			33.52%	41.97%	19.14%	24.71%	34.95%	46.30%
BEST LB			6650.37	4370.26	3920.10	2642.30	6203.29	4046.02
RSH_1	# of Rep	1000	7840.40	5493.94	4719.15	3455.08	7411.55	5258.19
		GAP	18.20%	26.18%	20.49%	30.93%	19.41%	29.94%
		CPU (sec)	2.78	2.58	8.10	8.33	1.53	1.43
RSH_2	# of Rep	1000	7860.47	5502.89	4707.22	3455.89	7468.75	5242.63
		GAP	18.57%	26.33%	20.21%	31.03%	20.34%	29.54%
		CPU (sec)	2.50	2.64	7.91	7.77	1.55	1.58
SSH_1	# of Rep	1	8294.93	5891.75	5075.74	3785.75	7939.32	5662.36
		GAP	24.96%	35.03%	29.54%	43.52%	28.00%	40.03%
		CPU (sec)	0.00	0.00	0.11	0.01	0.00	0.00
SSH_2	# of Rep	1	8341.50	5901.50	5264.02	4001.08	7878.75	5679.16
		GAP	25.85%	35.57%	34.54%	51.92%	27.07%	40.53%
		CPU (sec)	0.00	0.00	0.01	0.01	0.00	0.00
BEST UB			7803.85	5484.87	4693.37	3439.42	7410.11	5232.46
BEST GAP			17.66%	25.95%	19.86%	30.37%	19.38%	29.31%
BEST GAP (wo CPLEX)			18.66%	27.60%	24.16%	36.45%	19.63%	29.81%

The split delivery heuristic algorithm has a deficiency for splitting customer demands into different vehicles. At the end of the improvement step, to discharge a vehicle, the customer in that vehicle is taken and another vehicle is found to send the customer demands. If the found vehicle's empty capacity is greater than 75% of the chosen customer demand, 75% of the chosen customer demand is sent to the found vehicle. But, when we check the optimal solution, only 60% of the chosen customer demands are sent and the remaining capacity is used for another customer demand. Our heuristic algorithm has myopic view, and can miss better alternatives. However, D[1,8] settings have small demands, and therefore easier to split properly for small size instances.

Table 4.20 and **Table 4.21** show the results for 50 and 100 customers of the split delivery model. CPLEX could not find a solution in 5 minutes. The gaps without lower bound-3, and also CPU times are increase with number of customers.

Table 4.20 Summary of the Split Delivery Model Results for 50 Customers

			AVG					
			N50_D1-16_FC400	**N50_D1-16_FC200**	**N50_D1-8_FC400**	**N50_D1-8_FC200**	**N50_D4-12_FC400**	**N50_D4-12_FC200**
LB_1			15237.85	9797.85	7953.30	5013.30	14267.32	9147.32
LB_2			15418.79	9980.42	8486.40	5541.80	14496.48	9376.93
BEST LB			15435.94	9996.70	8486.40	5541.80	14501.71	9381.65
RSH_1	**# of Rep**	**1000**	19271.78	13442.19	11947.02	8626.43	18283.15	12848.17
		GAP(LB)	25.10%	34.96%	41.12%	56.61%	26.31%	37.46%
		CPU (sec)	67.84	65.21	204.47	211.33	36.09	33.52
RSH_2	**# of Rep**	**1000**	19305.36	13515.15	11952.38	8606.98	18246.55	12776.44
		GAP(LB)	25.40%	35.64%	41.18%	56.22%	26.04%	36.68%
		CPU (sec)	59.82	60.74	190.47	197.93	30.61	30.84
SSH_1	**# of Rep**	**1**	20098.99	14166.54	11835.27	8525.84	19135.17	13786.53
		GAP(LB)	30.37%	42.08%	39.98%	55.37%	32.18%	47.44%
		CPU (sec)	0.06	0.06	0.22	0.23	0.04	0.04
SSH_2	**# of Rep**	**1**	20386.10	14301.99	12992.47	9460.59	19324.25	13673.78
		GAP(LB)	32.28%	43.65%	53.25%	71.40%	33.52%	46.37%
		CPU (sec)	0.06	0.05	0.22	0.24	0.04	0.06
BEST UB			19159.33	13390.20	11693.57	8297.45	18186.72	12705.97
BEST GAP			24.33%	34.41%	38.19%	50.80%	25.62%	35.95%

Table 4.21 Summary of the Split Delivery Model Results for 100 Customers

			AVG					
			N100_D1-16_FC400	**N100_D1-16_FC200**	**N100_D1-8_FC400**	**N100_D1-8_FC200**	**N100_D4-12_FC400**	**N100_D4-12_FC200**
LB_1			29546.83	18866.83	15122.83	9402.83	27777.14	17677.14
LB_2			29919.05	19239.21	16088.74	10347.50	28232.56	18129.93
BEST LB			29950.29	19268.77	16088.74	10347.50	28242.59	18138.97
RSH_1	**# of Rep**	**1000**	38962.86	27483.46	24617.57	18247.91	37394.54	26518.84
		GAP(LB)	30.52%	43.11%	53.47%	76.99%	32.61%	46.53%
		CPU (sec)	802.63	856.97	2453.81	2538.54	411.72	422.08
RSH_2	**# of Rep**	**1000**	39311.27	27431.41	24714.48	18319.41	37067.01	26714.37
		GAP(LB)	31.78%	42.80%	54.05%	77.70%	31.51%	47.58%
		CPU (sec)	656.32	730.35	2080.93	2051.99	302.99	341.86
SSH_1	**# of Rep**	**1**	39564.42	28141.12	23558.52	17285.37	38044.13	27393.76
		GAP(LB)	32.33%	46.29%	46.82%	67.53%	34.85%	51.21%
		CPU (sec)	0.77	0.73	2.71	2.74	0.48	0.46
SSH_2	**# of Rep**	**1**	41112.86	29352.83	25916.10	19632.83	39307.56	28469.44
		GAP(LB)	40.28%	55.90%	64.87%	91.21%	39.42%	56.99%
		CPU (sec)	0.57	0.60	2.46	2.59	0.34	0.40
BEST UB			38550.24	27137.58	23458.18	17088.52	36878.60	26320.99
BEST GAP			28.28%	39.99%	44.08%	62.26%	30.48%	44.46%

4.3 Results and Discussions

Ten randomly generated problem instances with determined parameters are run for the three different MCVRP models. The basic model could be solved 20 customers optimally, and the performance of lower bound-1 and lower bound-2 are tested by these optimal values in **Table 4.22**. As seen on the table, lower bounds have high gaps, while the heuristics gaps that compared with optimal values are almost equal to zero. After 20 customers, the heuristic results have to be compared with lower bounds instead of the optimal values.

After 15 customers, the upper bound gaps increase suddenly due to the lower bound weakness. The lower bound-3 is obtained for 20 customers, and its effect is given in the table. However, after 20 customers, the solver could not find a lower bound within 5 minutes and hence the heuristic gaps increase.

Table 4.22 Performance of Basic Model Heuristics

	BEST LB GAP	BEST UB GAP	UB GAP (wo CPLEX)
N10_D1-16_FC400	15.98%	0.00%	
N10_D1-16_FC200	23.09%	0.00%	
N10_D1-8_FC400	20.65%	0.00%	
N10_D1-8_FC200	31.31%	0.00%	
N10_D4-12_FC400	21.14%	0.00%	
N10_D4-12_FC200	28.99%	0.00%	
N15_D1-16_FC400	22.58%	0.00%	
N15_D1-16_FC200	31.00%	0.00%	
N15_D1-8_FC400	20.39%	0.14%	
N15_D1-8_FC200	30.66%	0.52%	
N15_D4-12_FC400	25.84%	0.00%	
N15_D4-12_FC200	35.10%	0.00%	
N20_D1-16_FC400		10.40%	21.84%
N20_D1-16_FC200		12.36%	29.39%
N20_D1-8_FC400		16.95%	24.95%
N20_D1-8_FC200		24.91%	36.86%
N20_D4-12_FC400		14.42%	27.17%
N20_D4-12_FC200		17.86%	35.66%
N50_D1-16_FC400		25.73%	
N50_D1-16_FC200		36.41%	
N50_D1-8_FC400		38.14%	
N50_D1-8_FC200		56.00%	
N50_D4-12_FC400		30.71%	
N50_D4-12_FC200		43.02%	
N100_D1-16_FC400		29.13%	
N100_D1-16_FC200		42.07%	
N100_D1-8_FC400		45.59%	
N100_D1-8_FC200		66.84%	
N100_D4-12_FC400		29.56%	
N100_D4-12_FC200		47.94%	

The multi-trip model tries to decrease the number of vehicles by using the same vehicle for a second tour. The lower bound of the multi-trip model is almost half of the basic model lower bounds. However, the model size increases and the optimal solutions can be obtained up to 15 customers. Therefore, after 10 customers, we get the heuristic gaps by comparing them with the weak lower bounds as seen in **Table 4.23**. For 15 and 20 customers, lower bound-3 values are acquired from CPLEX at the end of the 5 minutes, and this shows that the actual heuristic gaps might not be as bad as it seems after 20 customers.

Table 4.23 Performance of Multi-Trip Model Heuristics

	BEST LB GAP	BEST UP GAP	UB GAP (wo CPLEX)
N10_D1-16_FC400	23.99%	0.00%	
N10_D1-16_FC200	31.17%	0.00%	
N10_D1-8_FC400	37.96%	0.32%	
N10_D1-8_FC200	40.86%	0.40%	
N10_D4-12_FC400	27.88%	0.62%	
N10_D4-12_FC200	35.27%	0.80%	
N15_D1-16_FC400		8.26%	31.66%
N15_D1-16_FC200		8.98%	40.93%
N15_D1-8_FC400		13.47%	38.41%
N15_D1-8_FC200		11.02%	28.45%
N15_D4-12_FC400		14.41%	33.39%
N15_D4-12_FC200		20.69%	44.48%
N20_D1-16_FC400		9.80%	13.86%
N20_D1-16_FC200		15.66%	23.68%
N20_D1-8_FC400		25.07%	27.15%
N20_D1-8_FC200		34.97%	39.59%
N20_D4-12_FC400		19.46%	19.84%
N20_D4-12_FC200		27.01%	31.87%
N50_D1-16_FC400		37.67%	
N50_D1-16_FC200		42.58%	
N50_D1-8_FC400		71.98%	
N50_D1-8_FC200		70.47%	
N50_D4-12_FC400		36.80%	
N50_D4-12_FC200		48.40%	
N100_D1-16_FC400		40.74%	
N100_D1-16_FC200		52.16%	
N100_D1-8_FC400		76.72%	
N100_D1-8_FC200		86.63%	
N100_D4-12_FC400		43.56%	
N100_D4-12_FC200		56.89%	

According to our assumptions, at most two vehicles can visit a customer. Although splitting of a demand increases the route cost, it decreases the number of vehicles' fixed costs. The split delivery model is harder than the basic and the multi-trip models. **Table 4.24** shows that the heuristic gaps are higher than the other model heuristics.

Since the split delivery route distance cannot be less than the basic model route distance and the minimum number of vehicles should be the same without manual control for both, the maximum lower bound values of the basic model are used for the split delivery model. When we compare the results, their lower bound values are close and since the objective value decreases, the lower bound gaps are decreased.

But our split delivery heuristic performance is not as good as the basic model heuristic.

Table 4.24 Performance of Split Delivery Model Heuristics

	BEST LB GAP	BEST UP GAP	UB GAP (wo CPLEX)
N10_D1-16_FC400	13.97%	1.21%	
N10_D1-16_FC200	21.46%	1.01%	
N10_D1-8_FC400	22.41%	0.22%	
N10_D1-8_FC200	34.01%	0.31%	
N10_D4-12_FC400	13.82%	1.50%	
N10_D4-12_FC200	21.23%	1.45%	
N15_D1-16_FC400		17.14%	19.94%
N15_D1-16_FC200		23.34%	29.25%
N15_D1-8_FC400		14.02%	23.44%
N15_D1-8_FC200		19.24%	36.73%
N15_D4-12_FC400		14.03%	17.87%
N15_D4-12_FC200		23.42%	29.31%
N20_D1-16_FC400		17.66%	18.66%
N20_D1-16_FC200		25.95%	27.60%
N20_D1-8_FC400		19.86%	24.16%
N20_D1-8_FC200		30.37%	36.45%
N20_D4-12_FC400		19.38%	19.63%
N20_D4-12_FC200		29.31%	29.81%
N50_D1-16_FC400		24.33%	
N50_D1-16_FC200		34.41%	
N50_D1-8_FC400		38.19%	
N50_D1-8_FC200		50.80%	
N50_D4-12_FC400		25.62%	
N50_D4-12_FC200		35.95%	
N100_D1-16_FC400		28.28%	
N100_D1-16_FC200		39.99%	
N100_D1-8_FC400		44.08%	
N100_D1-8_FC200		62.26%	
N100_D4-12_FC400		30.48%	
N100_D4-12_FC200		44.46%	

The heuristic algorithms find solutions of small size data in reasonable times. The increasing CPU time is associated with the increasing number of customers. **Figure 4.3** shows the relations between number of customers and CPU times in terms of seconds for the basic model with 200TL and 400TL fixed cost for basic model. Number of customers and CPU time relations charts for multi-trip and split delivery are shown in **Figure 4.4** and **Figure 4.5.** The split delivery heuristics take the longest time. The number of replications of the improvement step, splitting the demands to different vehicles and withdrawal processes increase the CPU time. As seen on the plots, range D[1-8] has the longest CPU times by far after 50 customers, and it has a parabolic curve. The other types of demands have the similar patterns, but take less durations.

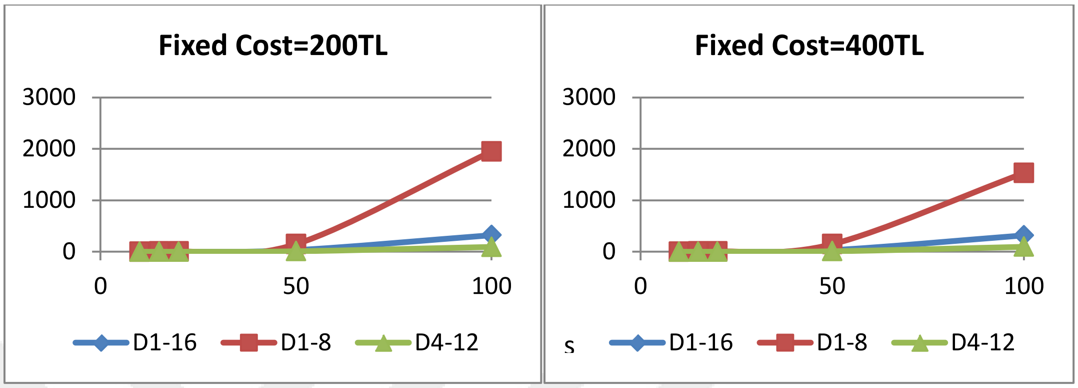

Figure 4.3 Customer-CPU Time Relations of Basic Model Heuristics

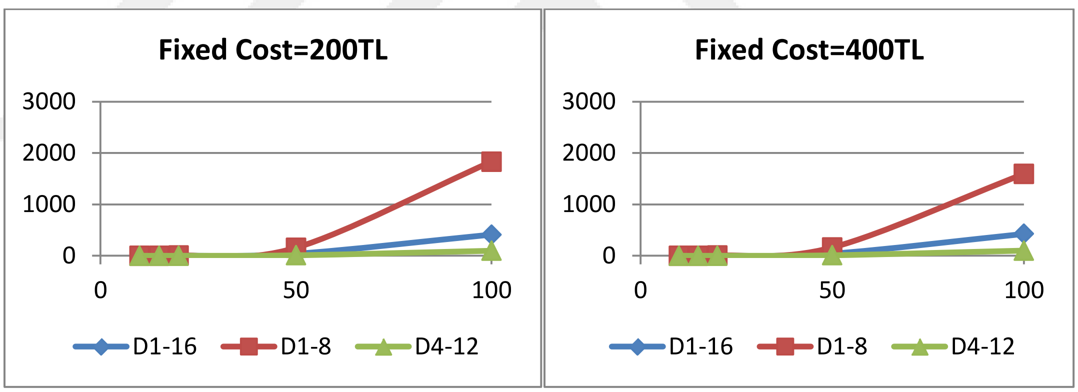

Figure 4.4 Customer- CPU Time Relations of Multi-Trip Model Heuristics

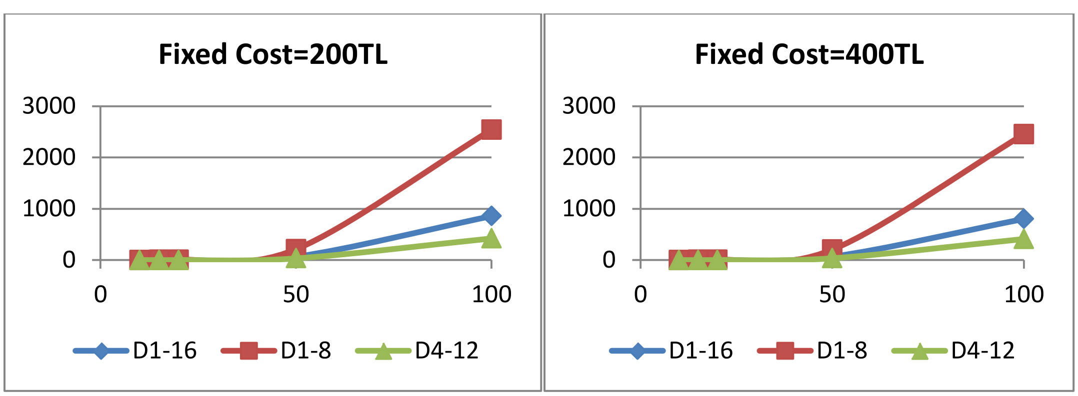

Figure 4.5 Customer-CPU Time Relations of Split Delivery Model Heuristics

Multi-trip and split delivery approaches are operational decisions, and they do not need any investments. It can be clearly seen from **Table 4.25** that multi-trip decision brings higher savings than split delivery. While, split delivery model tries to decrease the vehicle number by increasing route cost, in multi-trip the vehicle fixed cost, decreases by almost half. Expectedly the percentage savings have a relation with the fixed cost of vehicles.

Table 4.25 Savings of Multi-Trip and Split Delivery over Basic Model

Fixed Cost	Number of Customers	Multi-Trip	Split Delivery
400	10	36.56%	2.61%
	15	36.76%	3.74%
	20	36.95%	3.96%
	50	37.84%	4.54%
	100	37.23%	4.79%
200	10	23.88%	2.41%
	15	23.45%	2.61%
	20	24.93%	2.80%
	50	24.98%	3.43%
	100	25.90%	3.83%

5 CONCLUSION

In this thesis, we consider a multi-compartment vehicle routing problem, which is a variant of the vehicle routing problem. We first reviewed the related literature and proposed a taxonomic framework for variants of the problem. We analyzed and modeled the problem. Different operational decisions of product distribution such as multi-trip and split delivery were also examined, and the proposed model formulation was extended. To evaluate the effectiveness of our formulations, we conducted an experimental study. Random data sets were generated with respect to the determined factor levels, such as number of customers, demand type, and fixed cost. We observed that the increasing objective value and CPU time was associated with the increasing number of customers and fixed cost for the exact solutions of the basic model. D[1,8] (uniformly distributed) yielded the least average objective values because of fewer vehicle usages. However, its possible assignment combinations were more than others, and this increased the CPU time. While the resulting number of vehicles for D[1,16] and D[4,12] settings were similar, D[1,16] yielded higher objective values due to high variance, whereas D[4,12] with low variance yielded the fastest results.

Although we got the exact optimal solutions for small size instances, we had to resort to heuristic approaches for larger problem instances. The solution of the heuristic algorithms for the small size instances were found in milliseconds, but CPU time was increased with the increasing number of customers. Our heuristic algorithms were compared with the exact solutions reported by a general purpose solver for small sized randomly generated instances. We studied lower bounds, and checked their percentage gaps from the optimal solutions for small sized instances. These bounds were used for the purpose of reporting gaps of our heuristics in the case of larger instances, for which the solver cannot find a solution in reasonable times.

The heuristic solutions performed excellently, having tiny gaps for small size problems, achieving optimum solutions for the majority of the problem instances. The performance of our lower bounding schemes for the small instances was not quite good. For larger problems, the heuristic results were compared to the lower bounds, and the gaps were computed. Although large gaps were observed for larger problems, this is considered to be mainly due to the poor performance of our lower bounds. We believe that if the lower bound values were tighter, the heuristic gaps would be much more reasonable for larger problem instances.

The extensions of the basic model brought considerable savings. The operational decisions of the multi-trip model decreased the number of vehicle usage, and in turn, the fixed costs. However, the split delivery approach that tries to discharge vehicles by splitting the demands into two parts was not found to be as effective, as it increased the route costs while trying to decrease number of vehicles.

The problem can be extended in various directions. Time window perspective can be added to increase the satisfaction of customers. In this extension, the customers are visited within a given time interval. The intervals cannot be violated in case of hard time windows, or penalties can be included in the form of soft time windows.

Our study is deterministic and static; the customer demands were generated and known beforehand and vehicle routes did not change. When some parameters like demand are partially known or estimated, the problem should be considered as stochastic. In addition, the routes of vehicles can change over time, or new links between customers can be added to the system, in which case the problem becomes a dynamic vehicle routing problem.

As another future research opportunity, the VRP studied in this thesis can be embedded within a more general aggregate production planning framework, where the distribution can be triggered by the reorder points of the customer inventories. In such a case, the problem would involve integrated multi-period production planning and distribution decisions.

Another realistic extension may involve pick-up and deliveries. As farms in turkey breeding will require different feeds in different stages of growth, one customer picked-up leftover feed may be another one’s delivery. In such a case, a vehicle starting customer delivery would also pick up the leftover products from some customers and transit them to others or return them to the depot.

REFERENCES

Angelelli, E., Speranza, M. G., 2002, The Vehicle Routing Problem with Intermediate Facilities. European Journal of Operational Research, 137, 233–247.

Archetti, C., Mansini, R., Speranza, M. G., 2005, Complexity and Reducibility of the Skip Delivery Problem. Transportation Science, 39(2), 182–187.

Archetti, C., Hertz, A., Speranza, M.G., 2006, Metaheuristics For the Team Orienteering Problem. Journal of Heuristics 13 (1), 49–76.

Asawarungsaengkul, K., Rattanamanee, T., Nanthavanij, S., 2012, A Multi-Compartment Vehicle Routing Problem with Splitting of the Order Quantitiy for Delivery of the Multiple Types of Fuel Oils to Gas Stations. The 2nd International Conference on Computer and Management.

Avella, P., Boccia, M, Sforza, A., 2004, Solving a Fuel Delivery Problem by Heuristic and Exact Approaches. European Journal of Operational Research, 152(1), 170-179.

Azi, N., Gendreau, M., and Potvin, J.Y., 2007, An Exact Algorithm For A Single Vehicle Routing Problem With Time Windows And Multiple Routes. European Journal of Operational Research 178, 755–766.

Azi, N., Gendreau, M., and Potvin, J.Y., 2010, An Exact Algorithm For A Vehicle Routing Problem With Time Windows And Multiple Use Of Vehicles. European Journal of Operational Research 202, 756–763.

Bin, S., Fu, Z., 2003, An Improved Genetic Algorithm For Vehicle Routing Problem With Soft Time Windows. Systems Engineering, 21 (6), 12–15.

Belenguer, J.M., Martinez, M.C., Mota, E., 2000, A Lower Bound For The Split Delivery Vehicle Routing Problem. Operations Research 48 (5), 801–810.

Benantar, A., Oufi, R., 2012, Optimization of Vehicle Routes: An Application To Logistic and Transportation of the Fuel Distribution. 9th International Conference on Modeling, Optimization & SIMulation.

Bodin, L., 1975, A Taxonomic Structure For Vehicle Routing And Scheduling Problems. Computers and Urban Society, 1, 11–29.

Bodin, L., Golden, B., 1981, Classification in Vehicle Routing And Scheduling. Networks, 11, 97–108.

Boudia, M., Prins, C., Reghioui, M., 2007, An Effective Memetic Algorithm With Population Management For The Split Delivery Vehicle Routing Problem. In: Hybrid Metaheuristics. LNCS, vol. 4771. Springer, Berlin, Heidelberg, pp. 16–30.

Brandao, J. Mercer, A., 1997, A Tabu Search Algorithm for the Multi-trip Vehicle Routing and Scheduling Problem. European Journal of Operational Research, 100, 180-191

Chang, T. S., Wan, Y. W., Tsang, W., 2009, A Stochastic Dynamic Traveling Salesman Problem With Hard Time Windows. OOI European Journal of Operational Research 198, 748–759.

Chen, S., Golden, B., and Wasil, E. 2007, The Split Delivery Vehicle Routing Problem: Applications, Algorithms, Test Problems, And Computational Results. Networks, 49(4):318–329.

Chuah, K. H., Yingling, J. C., 2005, Routing For A Just-In-Time Supply Pick Up And Delivery System. Transportation Science, 39(3), 328–339.

Clarke, G., Wright, J., 1964, Scheduling of Vehicles From a Central Depot to a Number of Delivery Points. Operations Research, Vol. 12, No. 4, pp.568-581. doi:10.1287/opre.12.4.568.

Corberan, A., Fernandez, E., Laguna, M., Marti, R., 2002, Heuristic Solutions To The Problem of Routing School Buses With Multiple Objectives. J. Oper. Res. Soc. 53(4), 427–435.

Cordeau, J. F., Laporte, G., 2001, A Tabu Search Algorithm For The Site Dependent Vehicle Routing Problem With Time Windows. INFOR, 39(3), 292–298.

Dantzig, G. B., Ramser, J. H., 1959, The Truck Dispatching Problem. Management Science, Vol. 6, No. 1, pp. 80-91. doi:10.1287/mnsc.6.1.80.

Derigs, U., Li, B., Vogel, U., 2009, Local Search-Based Metaheuristics For The Split Delivery Vehicle Routing Problem. Journal of the Operational Research Society, 1356–1364.

Derigs, U., Gottlieb, J., Kalkoff, J., Piesche, M., Rothlauf, F., Vogel, U., 2011, Vehicle Routing With Compartments: Applications, Modelling And Heuristics. OR Spectrum, 33(4), pp. 885-914.

Desrochers, M., Lenstra, J. K., and Savelsbergh, M. W. P., 1990. A Classification Scheme For Vehicle Routing And Scheduling Problems. European Journal of Operational Research, 46, 322–332.

Desrochers, M., Jones, C. V., Lenstra, J. K., Savelsbergh, M. W. P., and Stougie, L., 1999, Towards a Model Algorithm Management System For Vehicle Routing And Scheduling Problems. Decision Support Systems, 25, 109–133.

Dror, M. and Trudeau, P., 1989, Savings by Split Delivery Routing. Transportation Science, Vol. 23, No. 2, pp. 141-145. doi:10.1287/trsc.23.2.141.

Dror, M. and Trudeau, P., 1990, Split Delivery Routing. Naval Research Logistics, Vol. 37, pp. 383-402.

Dueck, G., 1993, New optimization Heuristics: The Great Deluge Algorithm And The Record To Record Travel. Journal of Computational Physics, 104, 86–92

Eksioglu, B., Vural, A. V., Reisman, A., 2009, Survey The Vehicle Routing Problem: A taxonomic review Computers & Industrial Engineering 57, 1472–1483.

El Fallahi, A., Prins, C., Wolfler, C., 2008, A Memetic Algorithm And A Tabu Search For The Multi-Compartment Vehicle Routing Problem. Computers & Operations Research, 35(5), pp.1725-1741.

Fleischmann, B., Gnutzmann, S., and Sandvoss, E., 2004, Dynamic Vehicle Routing Based On Online Traffic Information. Transportation Science, 38(4), 420–433.

Fu, L., 2002, Scheduling Dial-A-Ride Paratransit Under Time Varying, Stochastic Congestion. Transportation Research Part B, 36(6), 485–506.

Gendreau, M., Laporte, G., and Seguin, R., 1996a, A Tabu Search Heuristic For The Vehicle Routing Problem With Stochastic Demand And Customers. Operations Research, 44(3), 469–477.

Goetschalckx, M. and Blecha C. J., 1989, The Vehicle Routing Problem With Backhauls. European Journal of Operational Research, Vol. 42, pp. 39-51.

Hernandez, F., Feillet, D., Giroudeau, R., and Naud, O., 2014, A New Exact Algorithm To Solve The Multi-Trip Vehicle Routing Problem With Time Windows And Limited Duration. 4OR, 12(3), 235–259.

Jaw, J. J., Odoni, A. R., Psaraftis, H. N., and Wilson, N. H., 1986, A Heuristic Algorithm For The Vehicle Advance Request Dial-A-Ride Problem With Time Windows. Transportation Research Part B, 20(3), 243–257.

Jin, M., Liu, K., and Bowden, R.O., 2007, A Two-Stage Algorithm With Valid Inequalities For The Split Delivery Vehicle Routing Problem. International Journal of Production Economics, 105(1): 228–242.

Kendall, D. G., 1953, Stochastic Processes Occurring in the Theory of Queues and their Analysis by the Method of the Imbedded Markov Chain. The Annals of Mathematical Statistics24 (3): 338.

Kirkpatrick, S., Gelatt, C., Vecchi, M., 1983, Optimization By Simulated Annealing. Science 220 (4598), 671–680.

Lahyani, R., C. Coelho, L., Khemakhem, M., Laporte, G., Semet, F., 2014, A Multi-Compartment Vehicle Routing Problem Arising in the Collection of Olive Oil in Tunisia . CIRRELT -17.

Laporte, G., Osman, I. H., 1995, Routing Problems: A bibliography. Annals of Operations Research, 61, 227–262.

Laporte, G., Gendreau, M., Potvin, J.-Y., and Semet, F., 1999, Classical And Modern Heuristics For The Vehicle Routing Problem. Technical report, Les Cahiers du Gerad, G-99-21.

Lee, T., Ueng, J., 1999, A Study Of Vehicle Routing Problems With Load-Balancing. Int. J. Phys. Distribution Logistics Management 29(10), 646–658.

Lee, C., Epelman, M., White III, C., Bozer, Y., 2006, A Shortest Path Approach To The Multiple-Vehicle Routing Problem With Split Pick-Ups. Transportation Research Part B: Methodological 40 (4), 265–284.

Lenstra, J., Rinnooy Kan, A., 1981, Complexity Of Vehicle Routing And Scheduling Problems. Networks vol. 11, pp.221-227.

Letchford, A. N., and Eglese, R. W., 1998, The Rural Postman Problem With Deadline Classes. European Journal of Operational Research, 105, 390–400.

Lin, S., Kernighan, B., 1973, An Effective Heuristic Algorithm For The Traveling Salesman Problem. Operations Research 21 (2), 498–516.

Macedo, R., Alves, C., Val´erio de Carvalho, J. M., Clautiaux, F., Hanafi, S., 2011, Solving The Vehicle Routing Problem With Time Windows And Multiple Routes Exactly Using A Pseudo-Polynomial Model. European Journal of Operational Research 214 (3), 536 – 545.

Mendoza, J. E., Castaniera B., Guéreta C., Medagliab A. L., Velascob N., 2010, A Memetic Algorithm For The Multi-Compartment Vehicle Routing Problem With Stochastic Demands. Computers & Operations Research, 37(11), pp.1886-1898.

Mota, E., Campos, V., Corberan, A., 2007, A New Metaheuristic For The Vehicle Routing Problem With Split Demands. In: Evolutionary Computation in Combinatorial Optimization. LNCS, vol. 4446. Springer, Berlin, Heidelberg, pp. 121–129.

Potvin, J.-Y., Rousseau, J.-M., 1995, An Exchange Heuristic For Routing Problems With Time Windows. Journal of the Operational Research Society 46 (12), 1433–1446.

Rais A., Alvelos F., Carvalho M.S., 2014, New Mixed Integer-Programming Model For The Pickup And Delivery Problem With Transshipment. European Journal of Operational Research Volume 235, Issue 3, pp. 530–539.

Rego, C., and Roucairol, C., 1995, Using Tabu Search For Solving A Dynamic Multiterminal Truck Dispatching Problem. European Journal of Operational Research, 83, 411–429.

Reisman, A., 1992, Management Science Knowledge: It's Creation Generalization And Consolidation. Westport, CT: Quorum Books Publishing Company.

Ribeiro, R., Loureno, H.R., 2001, A Multi-Objective Model For A Multi-Period Distribution Management Problem. In: MIC'2001-4th Metaheuristics International Conference, 97–102.

REFERENCES (continued)

Salhi, S., 1987, The Integration Of Routing Into The Location-Allocation And Vehicle Composition Problems. Ph.D. Thesis, University of Lancaster, pp. 198–208.

Salhi, S., Petch, R.J., 2007, A GA Based Heuristic For The Vehicle Routing Problem With Multiple Trips. Journal ofMathematical Modelling and Algorithms 6 (4), 591–613.

Savelsbergh, M., 1992, The Vehicle Routing Problem With Time Windows: Minimizing Route Duration. ORSA Journal on Computing 4 (2), 146–154.

Savelsbergh, M., Sol, M., 1995, The General Pickup And Delivery Problem Transportation Science, 29 (1), pp. 17–29.

Suprayogi, Setiawan Komara, S., Yamato, H., 2007, A Local Search Technique for Solving a Delivery Problem of Fuel Products. Proceedings of the 2nd International Conference on Operations and Supply Chain Management, 18-20.

Surjandari, I., Rachman, A., Dianawati, F., Wibowo, R.P., 2011, Petrol Delivery Assingment with Multi-Product, Multi-Depot, Split Deliveries and Time Windows. International Journal of Modeling and Optimization Vol.1, No.5.

Toth, P., Vigo, D., 2002, The Vehicle Routing Problem. Philadelphia: Society for Industrial and Applied Mathematics.

Van der Bruggen, L., Gruson, R., Salomon, M., 1995, Reconsidering the Distribution Structure Of Gasoline Products For A Large Oil Company. Eur J Oper Res 81(3):460–473.

Whittley, I., Smith, G., 2004, The Attribute Based Hill Climber. Journal of Mathematical Modelling and Algorithms 3 (2), 167–178.

Wilck IV, J. H. Cavalier, T. M., 2012, A Construction Heuristic for the Split Delivery Vehicle Routing Problem," American Journal of Operations Research, 2, 153-162.

CURRICULUM VITAE

Bahar Taşar was born in İzmir. She received her BS degree in Industrial Engineering with the 1st degree from Yaşar University. Before she joined the M.Sc. Program, she worked as a production planning engineer for 6 months. She is a research and teaching assistant in the Industrial Engineering Department at Yaşar University. She continues her MS degree in Industrial Engineering at Yaşar University. During her master degree period, she has taken various courses such as System Simulation, Optimization Models and Algorithms, Heuristic Optimization, Probabilistic Analysis, Mathematics of Operational Research and Supply Chain Processes and Management. Her research interests include vehicle routing, part formation and clustering.

APPENDIX 1 SOLUTION RESULTS

Heuristic Solutions Results with Minimum and Maximum Values:

Part 1 Basic Model Part 1.1 10 Customers

			MAX					
			N10_D1-16_FC400	N10_D1-16_FC200	N10_D1-8_FC400	N10_D1-8_FC200	N10_D4-12_FC400	N10_D4-12_FC200
OPT			5455.25	3655.25	2941.97	2141.97	5227.14	3627.14
CPU (sec)			1.34	1.31	3.52	2.24	2.73	2.05
LB_1			4718.11	3118.11	2463.79	1663.79	4117.10	2717.10
GAP LB_1			32.00%	40.68%	46.75%	77.57%	37.18%	50.44%
LB_2			4758.44	3158.44	2527.17	1727.17	4162.81	2762.81
GAP LB_2			32.00%	40.68%	36.88%	58.59%	32.67%	42.65%
BEST LB			4758.44	3158.44	2527.17	1727.17	4162.81	2762.81
BEST GAP			32.00%	40.68%	36.88%	58.59%	32.67%	42.65%
RSH_1	# of Rep	1000	5455.25	3655.25	2941.97	2141.97	5227.14	3627.14
		GAP	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		CPU (sec)	0.39	0.38	2.03	2.07	0.27	0.28
RSH_2	# of Rep	1000	5455.25	3655.25	2941.97	2141.97	5227.14	3627.14
		GAP	0.00%	0.00%	0.00%	0.01%	0.00%	0.00%
		CPU (sec)	1.60	1.58	1.81	1.86	1.07	1.03
SSH_1	# of Rep	1	5455.25	3655.25	3340.66	2340.66	5227.14	3627.14
		GAP	0.13	0.12	0.19	0.20	0.06	0.08
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
SSH_2	# of Rep	1	5455.25	3655.25	3525.99	2525.99	5227.14	3627.14
		GAP	0.09%	0.12%	0.27%	0.38%	0.12%	0.16%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
BEST UB			5455.25	3655.25	2941.97	2141.97	5227.14	3627.14
BEST GAP			0.00%	0.00%	0.00%	0.00%	0.00%	0.00%

			MIN					
			N10_D1-16_FC400	N10_D1-16_FC200	N10_D1-8_FC400	N10_D1-8_FC200	N10_D4-12_FC400	N10_D4-12_FC200
OPT			3208.92	2208.92	2073.28	1473.28	3508.14	2452.31
CPU (sec)			0.56	0.30	0.28	0.28	0.79	0.95
LB_1			2667.19	1667.19	1510.45	910.45	2667.19	1667.19
GAP LB_1			5.86%	8.70%	17.39%	27.69%	12.62%	19.13%
LB_2			2758.23	1758.23	1619.37	1019.37	2758.23	1758.23
GAP LB_2			7.53%	11.34%	7.74%	11.84%	12.62%	19.13%
BEST LB			2758.23	1758.23	1619.37	1019.37	2758.23	1758.23
BEST GAP			5.86%	8.70%	7.74%	11.84%	12.62%	19.13%
RSH_1	# of Rep	1000	3208.92	2208.92	2073.28	1473.28	3508.14	2452.31
		GAP	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		CPU (sec)	0.09	0.08	0.22	0.23	0.05	0.05
RSH_2	# of Rep	1000	3208.92	2208.92	2073.28	1489.80	3508.14	2452.31
		GAP	0.00	0.00	0.00	0.00	0.00	0.00
		CPU (sec)	0.07	0.09	0.21	0.22	0.07	0.09
SSH_1	# of Rep	1	3214.59	2214.59	2181.76	1581.76	3714.55	2544.37
		GAP	0.00%	0.00%	0.85%	1.17%	0.00%	0.00%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
SSH_2	# of Rep	1	3484.93	2460.39	2362.40	1762.40	3653.05	2453.05
		GAP	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
BEST UB			3208.92	2208.92	2073.28	1473.28	3508.14	2452.31
BEST GAP			0.00%	0.00%	0.00%	0.00%	0.00%	0.00%

Part 1.2 15 Customers

			MAX					
			N15_D1-16_FC400	N15_D1-16_FC200	N15_D1-8_FC400	N15_D1-8_FC200	N15_D4-12_FC400	N15_D4-12_FC200
OPT			7328.18	5128.18	4237.26	3037.26	7237.26	5037.26
CPU (sec)			475.50	2456.03	686.00	1187.32	1482.94	1334.25
LB_1			6538.01	4338.01	3518.87	2318.87	5258.79	3458.79
GAP LB_1			37.31%	51.38%	43.56%	68.83%	37.63%	52.16%
LB_2			6572.00	4372.00	3617.76	2417.76	5304.51	3504.51
GAP LB_2			34.45%	45.68%	31.98%	48.16%	36.44%	43.74%
BEST LB			6572.00	4372.00	3617.76	2417.76	5304.51	3504.51
BEST GAP			34.45%	45.68%	31.98%	48.16%	36.44%	43.74%
RSH_1	**# of Rep**	**1000**	7328.18	5128.18	4273.30	3088.74	7237.26	5037.26
		GAP	0.00%	0.00%	1.50%	2.13%	0.00%	0.66%
		CPU (sec)	1.57	1.75	6.28	6.06	0.51	0.51
RSH_2	**# of Rep**	**1000**	7328.18	5128.18	4288.74	3088.74	7237.26	5037.26
		GAP	2.03%	0.00%	3.55%	6.41%	0.00%	0.00%
		CPU (sec)	1.82	1.77	5.01	5.02	1.93	2.19
SSH_1	**# of Rep**	**1**	7528.83	5328.83	4318.06	3316.90	7364.51	5164.51
		GAP	11.36%	7.98%	24.15%	23.87%	13.03%	13.25%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
SSH_2	**# of Rep**	**1**	7328.18	5128.18	4757.13	3557.13	7373.99	5173.99
		GAP	11.10%	16.09%	21.39%	22.41%	9.68%	14.09%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
BEST UB			7328.18	5128.18	4273.30	3088.74	7237.26	5037.26
BEST GAP			0.00%	0.00%	0.85%	1.88%	0.00%	0.00%

			MIN					
			N15_D1-16_FC400	N15_D1-16_FC200	N15_D1-8_FC400	N15_D1-8_FC200	N15_D4-12_FC400	N15_D4-12_FC200
OPT			4515.45	3115.45	2806.13	2006.13	4963.84	3363.84
CPU (sec)			6.92	14.28	13.78	15.48	10.13	8.63
LB_1			3779.89	2379.89	2073.03	1273.03	3779.89	2379.89
GAP LB_1			12.09%	18.21%	20.05%	30.43%	15.82%	24.16%
LB_2			3874.90	2474.90	2254.78	1454.78	3874.90	2474.90
GAP LB_2			11.51%	17.30%	11.73%	17.46%	11.04%	16.50%
BEST LB			3874.90	2474.90	2254.78	1454.78	3874.90	2474.90
BEST GAP			11.51%	17.30%	11.73%	17.46%	11.04%	16.50%
RSH_1	**# of Rep**	**1000**	4515.45	3115.45	2806.13	2047.82	4963.84	3363.84
		GAP	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		CPU (sec)	0.15	0.16	0.53	0.51	0.15	0.13
RSH_2	**# of Rep**	**1000**	4515.45	3115.45	2806.13	2006.13	4963.84	3363.84
		GAP	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		CPU (sec)	0.20	0.21	0.67	0.66	0.19	0.18
SSH_1	**# of Rep**	**1**	5028.49	3303.04	2889.76	2089.76	5350.74	3750.74
		GAP	0.46%	0.67%	1.35%	4.17%	0.37%	1.19%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
SSH_2	**# of Rep**	**1**	5016.64	3616.64	3033.65	2233.65	5401.90	3801.90
		GAP	0.00%	0.00%	6.93%	9.56%	1.53%	2.19%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
BEST UB			4515.45	3115.45	2806.13	2006.13	4963.84	3363.84
BEST GAP			0.00%	0.00%	0.00%	0.00%	0.00%	0.00%

Part 1.3 20 Customers

			MAX					
			N20_D1-16_FC400	N20_D1-16_FC200	N20_D1-8_FC400	N20_D1-8_FC200	N20_D4-12_FC400	N20_D4-12_FC200
LB CPLEX			8310.84	5710.84	4522.30	3094.01	7631.13	5257.86
GAP LB CPLEX			14.93%	16.42%	17.97%	22.41%	23.88%	21.83%
LB_1			7512.89	4912.89	3993.54	2593.54	6636.39	4436.39
LB_2			7580.92	4980.92	4257.21	2857.21	6800.94	4600.94
BEST LB			8310.84	5710.84	4522.30	3094.01	7631.13	5257.86
RSH_1	# of Rep	1000	9505.64	6505.64	5354.29	3972.34	8977.01	6177.01
		GAP	17.55%	19.64%	21.85%	32.79%	21.79%	23.03%
		CPU (sec)	2.97	3.09	11.07	11.33	0.91	0.88
RSH_2	# of Rep	1000	9517.51	6517.51	5390.62	3893.30	8977.01	6177.01
		GAP	18.88%	21.68%	23.28%	36.48%	24.90%	27.19%
		CPU (sec)	2.17	4.85	11.27	11.83	1.06	1.08
SSH_1	# of Rep	1	9718.74	6718.74	5786.65	4533.48	9160.37	6447.94
		GAP	21.91%	37.83%	39.10%	65.78%	27.74%	36.47%
		CPU (sec)	0.00	0.00	0.02	0.02	0.00	0.00
SSH_2	# of Rep	1	9564.23	6736.26	6026.35	4450.31	9279.22	6479.22
		GAP	25.69%	34.34%	44.30%	66.72%	28.92%	34.65%
		CPU (sec)	0.00	0.00	1.00	3.00	0.00	0.00
BEST UB			9505.64	6505.64	5354.29	3893.30	8977.01	6177.01
BEST GAP			17.55%	19.64%	21.85%	32.79%	21.79%	23.03%
GAP (w/o CPLEX)			30.58%	41.59%	32.38%	48.33%	39.13%	45.79%

			MIN					
			N20_D1-16_FC400	N20_D1-16_FC200	N20_D1-8_FC400	N20_D1-8_FC200	N20_D4-12_FC400	N20_D4-12_FC200
LB CPLEX			6050.73	3951.85	3288.16	2279.93	6173.37	4172.58
GAP LB CPLEX			4.42%	5.27%	3.75%	10.55%	7.34%	11.92%
LB_1			4864.48	3064.48	2676.76	1676.76	5478.74	3478.74
LB_2			5143.86	3343.86	3012.70	2012.70	5712.59	3646.87
BEST LB			6254.64	4123.59	3288.16	2279.93	6173.37	4172.58
RSH_1	# of Rep	1000	6740.18	4735.50	3876.74	2940.65	6761.45	4700.67
		GAP	4.48%	5.57%	13.04%	20.49%	7.59%	12.66%
		CPU (sec)	0.67	0.67	1.98	3.82	0.30	0.30
RSH_2	# of Rep	1000	6717.00	4734.63	3942.26	2943.93	6761.45	4700.67
		GAP	5.28%	8.96%	15.34%	18.78%	8.48%	14.11%
		CPU (sec)	0.58	0.55	3.46	3.43	0.22	0.41
SSH_1	# of Rep	1	6832.02	4763.72	4223.58	3279.19	7102.19	4902.19
		GAP	6.23%	14.09%	15.26%	29.13%	11.99%	19.33%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
SSH_2	# of Rep	1	6984.86	4984.86	4622.75	3612.96	7474.93	5209.93
		GAP	10.25%	15.49%	20.47%	43.23%	15.06%	19.44%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
BEST UB			6717.00	4734.63	3876.74	2940.65	6761.45	4700.67
BEST GAP			4.48%	5.57%	13.04%	18.78%	7.59%	12.66%
GAP (w/o CPLEX)			14.42%	16.89%	18.73%	25.09%	18.36%	25.56%

Part 1.4 50 Customers

			MAX					
			N50_D1-16_FC400	N50_D1-16_FC200	N50_D1-8_FC400	N50_D1-8_FC200	N50_D4-12_FC400	N50_D4-12_FC200
LB_1			17323.53	11323.53	8908.31	5874.57	15470.89	10226.11
LB_2			17152.03	11496.11	9422.61	6418.47	15838.36	10637.04
BEST LB			17323.53	11496.11	9422.61	6418.47	15838.36	10637.04
RSH_1	# of Rep	1000	21071.20	14808.10	12849.20	9518.55	20452.10	14202.20
		GAP	34.68%	46.35%	50.95%	77.93%	36.93%	52.52%
		CPU (sec)	60.43	63.29	208.29	210.56	15.66	15.76
RSH_2	# of Rep	1000	20979.30	14698.80	12782.40	9595.99	20412.50	14284.80
		GAP	32.65%	47.19%	51.42%	76.60%	36.41%	51.28%
		CPU (sec)	53.80	53.44	195.86	189.63	13.82	14.20
SSH_1	# of Rep	1	21944.00	15390.40	13022.10	9322.06	21105.30	14910.20
		GAP	42.25%	55.17%	52.36%	75.17%	45.31%	60.23%
		CPU (sec)	0.07	0.10	0.24	0.35	0.02	0.02
SSH_2	# of Rep	1	22105.00	15613.40	13871.90	10671.90	22117.50	15517.50
		GAP	41.75%	56.57%	58.82%	90.33%	48.00%	66.25%
		CPU (sec)	0.04	0.04	0.23	0.22	0.01	0.01
BEST UB			20979.30	14698.80	12782.40	9322.06	20412.50	14202.20
BEST GAP			32.65%	46.35%	50.95%	70.52%	36.41%	51.28%

			MIN					
			N50_D1-16_FC400	N50_D1-16_FC200	N50_D1-8_FC400	N50_D1-8_FC200	N50_D4-12_FC400	N50_D4-12_FC200
LB_1			12741.54	8141.54	6895.94	4295.94	12741.54	8141.54
LB_2			13130.44	8537.15	7533.61	4816.61	13067.80	8472.17
BEST LB			13130.44	8537.15	7533.61	4816.61	13067.80	8472.17
RSH_1	# of Rep	1000	16373.30	11824.30	10508.90	7936.50	17317.30	12333.50
		GAP	20.24%	28.81%	32.90%	48.30%	25.81%	33.52%
		CPU (sec)	17.69	17.76	115.40	114.26	6.38	6.76
RSH_2	# of Rep	1000	16520.50	11856.30	10482.80	7866.73	17256.20	12237.10
		GAP	21.10%	27.86%	33.02%	46.42%	23.27%	34.29%
		CPU (sec)	11.89	11.36	96.03	96.26	6.18	6.01
SSH_1	# of Rep	1	17122.90	12582.70	10516.10	7848.20	18534.40	13053.20
		GAP	22.54%	30.05%	32.87%	43.71%	31.65%	39.94%
		CPU (sec)	0.01	0.01	0.10	0.09	0.00	0.00
SSH_2	# of Rep	1	17197.60	12397.60	11569.40	8840.90	17810.00	12810.00
		GAP	24.58%	32.31%	45.16%	65.72%	33.04%	43.57%
		CPU (sec)	0.00	0.00	0.07	0.09	0.00	0.00
BEST UB			16373.30	11824.30	10482.80	7848.20	17256.20	12237.10
BEST GAP			20.24%	27.86%	32.87%	43.71%	23.27%	33.52%

Part 1.5 100 Customers

			MAX					
			N100_D1-16_FC400	N100_D1-16_FC200	N100_D1-8_FC400	N100_D1-8_FC200	N100_D4-12_FC400	N100_D4-12_FC200
LB_1			32735.85	21535.85	16785.66	10785.66	30195.20	19795.20
LB_2			33610.69	22401.26	17822.25	11784.26	31002.14	20590.66
BEST LB			33610.69	22401.26	17822.25	11784.26	31002.14	20590.66
RSH_1	# of Rep	1000	41879.30	29676.70	26181.37	19122.70	40475.40	28840.90
		GAP	49.06%	59.44%	56.17%	79.16%	45.42%	64.43%
		CPU (sec)	586.39	613.97	2031.54	4145.47	151.27	146.67
RSH_2	# of Rep	1000	42022.06	29696.80	26286.10	19061.00	40623.30	28603.50
		GAP	49.66%	60.07%	56.79%	79.95%	46.00%	65.09%
		CPU (sec)	435.74	457.45	2305.15	5415.92	114.79	111.99
SSH_1	# of Rep	1	46224.27	31764.88	28914.71	20867.39	44685.63	31138.17
		GAP	64.62%	76.08%	72.47%	94.81%	60.61%	81.60%
		CPU (sec)	0.53	0.71	1.62	2.18	0.15	0.12
SSH_2	# of Rep	1	48739.45	34652.59	32010.76	32010.76	48747.97	33968.91
		GAP	79.59%	92.09%	103.32%	215.79%	75.21%	98.11%
		CPU (sec)	0.53	0.71	1.62	1.59	0.39	0.27
BEST UB			41854.64	29676.70	26181.37	18894.78	40461.46	28603.50
BEST GAP			49.06%	59.44%	56.17%	76.40%	45.42%	64.43%

			MIN					
			N100_D1-16_FC400	N100_D1-16_FC200	N100_D1-8_FC400	N100_D1-8_FC200	N100_D4-12_FC400	N100_D4-12_FC200
LB_1			25916.07	16316.07	13225.93	8025.93	25312.15	15912.15
LB_2			26344.24	16752.86	14448.93	9218.29	25730.34	16338.12
BEST LB			26344.24	16752.86	14448.93	9218.29	25730.34	16338.12
RSH_1	# of Rep	1000	35216.40	25092.60	21468.80	16249.80	34915.20	24886.70
		GAP	24.60%	32.48%	42.29%	59.59%	30.56%	39.21%
		CPU (sec)	166.59	175.25	1092.26	1030.31	59.96	56.26
RSH_2	# of Rep	1000	35315.10	24825.70	21446.00	16143.00	34943.60	24899.50
		GAP	23.85%	32.57%	40.47%	58.86%	28.55%	38.91%
		CPU (sec)	143.06	149.67	1086.38	1084.16	57.29	60.22
SSH_1	# of Rep	1	35196.10	24917.40	20916.90	15617.90	35911.90	25311.90
		GAP	27.30%	33.86%	34.74%	45.81%	32.28%	41.86%
		CPU (sec)	0.11	0.06	1.14	1.10	0.02	0.02
SSH_2	# of Rep	1	32010.76	26723.00	19483.80	17402.50	32010.76	26657.00
		GAP	9.42%	38.73%	25.63%	63.14%	11.04%	44.87%
		CPU (sec)	0.11	0.06	0.19	0.25	0.02	0.02
BEST UB			32010.76	24825.70	19483.80	15617.90	32010.76	24886.70
BEST GAP			9.42%	32.48%	25.63%	45.81%	11.04%	38.91%

Part 2.1 10 Customers

			MAX					
			N10_D1-16_FC400	N10_D1-16_FC200	N10_D1-8_FC400	N10_D1-8_FC200	N10_D4-12_FC400	N10_D4-12_FC200
OPT			3855.25	2855.25	2541.97	1941.97	3855.25	2855.25
CPU (sec)			100.47	91.72	34.00	40.38	91.73	95.90
LB_1			3118.11	2318.11	1663.79	1376.61	2917.10	2117.10
GAP_LB_1			50.10%	56.71%	98.96%	96.58%	49.01%	54.81%
LB_2			3158.44	2358.44	1727.17	1521.89	2962.81	2162.81
GAP_LB_2)			50.10%	56.71%	73.19%	70.45%	41.31%	45.93%
BEST LB			3158.44	2358.44	1727.17	1521.89	2962.81	2162.81
BEST GAP			50.10%	56.71%	73.19%	70.45%	34.26%	45.93%
RSH_1	# of Rep	1000	3855.25	2855.25	2541.97	1941.97	4027.14	3027.14
		GAP	0.00%	0.00%	3.19%	3.98%	4.27%	5.68%
		CPU (sec)	0.78	0.86	2.15	2.25	0.15	0.12
RSH_2	# of Rep	1000	3855.25	2855.25	2541.97	1941.97	4027.14	3027.14
		GAP	0.00%	0.00%	3.19%	3.98%	4.27%	5.68%
		CPU (sec)	1.05	1.41	1.15	1.15	0.41	0.49
SSH_1	# of Rep	1	3855.25	2855.25	2890.34	2112.48	4027.10	3027.14
		GAP	15.48%	12.71%	32.37%	27.49%	15.70%	17.13%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
SSH_2	# of Rep	1	3855.25	2855.25	3525.99	2525.99	4027.14	3027.14
		GAP	14.67%	13.24%	41.03%	33.53%	28.50%	29.53%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
BEST UB			3855.25	2855.25	2541.97	1941.97	4027.10	3027.14
BEST GAP			0.00%	0.00%	3.19%	3.98%	4.27%	5.68%

			MIN					
			N10_D1-16_FC400	N10_D1-16_FC200	N10_D1-8_FC400	N10_D1-8_FC200	N10_D4-12_FC400	N10_D4-12_FC200
OPT			2408.92	1808.92	1673.28	1273.28	2452.31	1852.31
CPU (sec)			9.40	13.05	1.64	1.89	14.30	12.74
LB_1			1867.19	1267.19	976.61	710.45	1867.19	1267.19
GAP_LB_1			8.13%	10.98%	27.69%	12.09%	19.13%	25.77%
LB_2			1958.23	1358.23	1121.89	819.37	1958.23	1358.23
GAP_LB_2)			10.58%	14.48%	11.84%	1.39%	19.13%	25.77%
BEST LB			1958.23	1358.23	1121.89	819.37	1958.23	1358.23
BEST GAP			8.13%	10.98%	11.84%	1.39%	19.13%	25.77%
RSH_1	# of Rep	1000	2408.92	1808.92	1673.28	1273.28	2452.31	1852.31
		GAP	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		CPU (sec)	0.07	0.05	0.21	0.19	0.04	0.04
RSH_2	# of Rep	1000	2408.92	1808.92	1673.28	1273.28	2452.31	1852.31
		GAP	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		CPU (sec)	0.06	0.09	0.18	0.18	0.08	0.07
SSH_1	# of Rep	1	2414.59	1814.59	1781.76	1381.76	2544.37	1944.37
		GAP	0.00%	0.00%	0.97%	1.27%	0.00%	0.00%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
SSH_2	# of Rep	1	2460.39	1860.39	1962.40	1562.40	2453.05	1853.05
		GAP	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
BEST UB			2408.92	1808.92	1673.28	1273.28	2452.31	1852.31
BEST GAP			0.00%	0.00%	0.00%	0.00%	0.00%	0.00%

Part 2.2 15 Customers

			MAX					
			N15_D1-16_FC400	N15_D1-16_FC200	N15_D1-8_FC400	N15_D1-8_FC200	N15_D4-12_FC400	N15_D4-12_FC200
LB_CPLEX			5173.75	3899.21	2806.11	2048.07	4166.37	3177.43
GAP_LB_CPLEX			17.52%	17.61%	16.53%	22.48%	20.93%	23.36%
LB_1			4538.01	3338.01	2423.55	2089.91	3658.79	2658.79
LB_2			4572.00	3372.00	2423.55	2306.40	3704.51	2704.51
BEST LB			5173.75	3899.21	2806.11	2306.40	4166.37	3177.43
RSH_1	# of Rep	1000	5328.18	4128.18	3073.30	2488.74	5237.26	4037.26
		GAP	21.56%	16.04%	32.89%	21.52%	26.42%	41.01%
		CPU (sec)	1.35	1.32	3.41	3.48	0.40	0.41
RSH_2	# of Rep	1000	5328.18	4128.18	3136.67	2513.97	5237.26	4037.26
		GAP	24.23%	16.04%	33.32%	24.81%	26.42%	41.01%
		CPU (sec)	1.78	1.73	2.98	3.07	0.68	0.67
SSH_1	# of Rep	1	5528.83	4328.83	4230.56	3030.56	6013.63	4553.20
		GAP	23.64%	25.42%	63.45%	41.33%	45.47%	52.37%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
SSH_2	# of Rep	1	5328.18	4128.18	4746.00	3546.00	5407.56	4207.56
		GAP	32.59%	36.38%	83.36%	63.44%	40.11%	46.05%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
BEST UB			5328.18	4128.18	3073.30	2488.74	5237.26	4037.26
BEST GAP			21.56%	16.04%	32.89%	21.52%	26.42%	41.01%
GAP (w/o CPLEX)			51.11%	70.62%	83.24%	87.28%	50.09%	59.50%

			MIN					
			N15_D1-16_FC400	N15_D1-16_FC200	N15_D1-8_FC400	N15_D1-8_FC200	N15_D4-12_FC400	N15_D4-12_FC200
LB_CPLEX			3105.01	2211.91	1836.73	1439.26	2855.38	2259.24
GAP_LB_CPLEX			0.53%	0.81%	0.20%	0.78%	5.08%	6.83%
LB_1			2579.89	1779.89	1273.03	873.03	2579.89	1779.89
LB_2			2674.90	1874.90	1454.78	1054.78	2674.90	1874.90
BEST LB			3105.01	2211.91	1836.73	1439.26	2981.26	2259.24
RSH_1	# of Rep	1000	3315.45	2515.45	2006.13	1647.82	3363.84	2563.84
		GAP	2.98%	1.05%	0.99%	3.91%	5.35%	7.97%
		CPU (sec)	0.18	0.19	1.41	1.41	0.12	0.12
RSH_2	# of Rep	1000	3315.45	2436.13	2006.13	1606.13	3363.84	2563.84
		GAP	2.98%	1.05%	0.99%	3.51%	5.35%	7.97%
		CPU (sec)	0.17	0.20	1.25	1.27	0.19	0.18
SSH_1	# of Rep	1	3828.49	2703.04	2089.76	1689.76	3558.48	2950.74
		GAP	6.86%	7.14%	9.69%	10.10%	5.20%	12.28%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
SSH_2	# of Rep	1	3816.64	3016.64	2633.65	2033.65	3801.90	3001.90
		GAP	2.98%	5.87%	24.19%	25.78%	11.42%	12.80%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
BEST UB			3315.45	2436.13	2006.13	1606.13	3363.84	2563.84
BEST GAP			2.98%	1.05%	0.99%	3.51%	5.20%	7.97%
GAP (w/o CPLEX)			16.54%	22.43%	16.93%	3.51%	16.50%	22.94%

Part 2.3 20 Customers

			MAX					
			N20_D1-16_FC400	N20_D1-16_FC200	N20_D1-8_FC400	N20_D1-8_FC200	N20_D4-12_FC400	N20_D4-12_FC200
LB_CPLEX			5781.04	4338.15	3240.95	2440.76	5134.68	3899.56
GAP_LB_CPLEX			26.47%	27.02%	29.42%	35.10%	32.64%	34.98%
LB_1			5868.17	4068.17	3193.54	2193.54	5036.39	3636.39
LB_2			5980.92	4180.92	3457.21	2457.21	5200.94	3800.94
BEST LB			5980.92	4338.15	3457.21	2457.21	5200.94	3899.56
RSH_1	# of Rep	1000	6705.64	5105.64	4063.06	3322.41	6177.01	4832.84
		GAP	16.10%	27.38%	42.80%	51.79%	29.49%	35.31%
		CPU (sec)	2.49	2.51	9.24	9.45	0.80	0.76
RSH_2	# of Rep	1000	6717.51	5117.51	4161.84	3245.36	6178.03	4832.84
		GAP	15.85%	27.35%	45.80%	55.47%	28.62%	35.31%
		CPU (sec)	3.33	3.49	8.59	8.84	0.89	0.93
SSH_1	# of Rep	1	6918.74	5372.88	5027.73	4533.48	7047.94	5447.94
		GAP	21.91%	41.14%	73.84%	90.05%	57.82%	59.07%
		CPU (sec)	0.00	0.00	0.01	0.01	0.00	0.00
SSH_2	# of Rep	1	7248.72	5648.72	5450.31	4250.31	7208.50	5285.43
		GAP	35.11%	42.77%	90.27%	98.01%	52.11%	59.57%
		CPU (sec)	0.00	0.00	0.02	0.01	0.00	0.00
BEST UB			6705.64	5105.64	4063.06	3245.36	6177.01	4832.84
BEST GAP			15.85%	27.35%	42.80%	49.98%	28.62%	35.31%
GAP (w/o CPLEX)			33.10%	46.81%	56.80%	70.14%	28.62%	40.62%

			MIN					
			N20_D1-16_FC400	N20_D1-16_FC200	N20_D1-8_FC400	N20_D1-8_FC200	N20_D4-12_FC400	N20_D4-12_FC200
LB_CPLEX			3847.88	2847.13	2429.54	1829.54	3918.20	2929.73
GAP_LB_CPLEX			9.89%	9.33%	10.11%	20.42%	14.64%	17.26%
LB_1			3264.48	2264.48	1876.76	1276.76	3878.74	2678.74
LB_2			3543.86	2543.86	2212.70	1612.70	4112.59	2846.87
BEST LB			4082.76	2932.67	2429.54	1829.54	4112.59	2929.73
RSH_1	# of Rep	1000	4740.18	3619.91	3046.48	2444.18	4761.45	3700.67
		GAP	4.12%	2.73%	9.73%	23.66%	7.85%	17.71%
		CPU (sec)	0.69	0.69	3.95	3.95	0.32	0.33
RSH_2	# of Rep	1000	4717.00	3634.25	3069.58	2372.80	4780.44	3700.67
		GAP	4.12%	3.14%	10.56%	20.05%	7.85%	17.71%
		CPU (sec)	0.54	0.55	3.45	3.55	0.43	0.39
SSH_1	# of Rep	1	4832.02	3763.72	3466.70	2879.19	5102.19	3902.19
		GAP	10.19%	11.16%	13.57%	27.23%	12.29%	24.12%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
SSH_2	# of Rep	1	5384.86	3856.42	4012.96	3212.96	5209.93	4009.93
		GAP	8.49%	9.45%	39.60%	55.72%	14.66%	27.55%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
BEST UB			4717.00	3619.91	3046.48	2372.80	4761.45	3700.67
BEST GAP			4.12%	2.73%	9.73%	20.05%	7.85%	17.71%
GAP (w/o CPLEX)			4.12%	2.73%	9.73%	20.05%	7.85%	17.71%

Part 2.4 50 Customers

			MAX					
			N50_D1-16_FC400	N50_D1-16_FC200	N50_D1-8_FC400	N50_D1-8_FC200	N50_D4-12_FC400	N50_D4-12_FC200
LB_1			11323.53	8323.53	6074.57	4474.57	10270.89	7626.11
LB_2			11497.00	8696.11	6622.61	5018.47	10638.36	8037.04
BEST LB			11497.00	8696.11	6622.61	5018.47	10638.36	8037.04
RSH_1	# of Rep	1000	35595.50	11808.10	10786.00	8586.16	14697.70	11602.20
		GAP	256.17%	61.65%	156.98%	147.88%	51.35%	67.80%
		CPU (sec)	104.12	93.36	249.61	216.96	17.67	17.53
RSH_2	# of Rep	1000	36307.41	11859.85	11040.60	8398.07	14538.30	11684.80
		GAP	263.30%	64.88%	162.12%	152.84%	54.38%	68.11%
		CPU (sec)	113.49	101.77	272.08	233.48	19.26	19.11
SSH_1	# of Rep	1	20614.14	12390.40	11058.20	7843.19	15431.25	12110.20
		GAP	90.87%	69.30%	122.01%	110.04%	58.45%	75.78%
		CPU (sec)	0.10	0.06	0.30	0.28	0.02	0.01
SSH_2	# of Rep	1	20955.17	12813.40	13071.90	10271.90	16935.81	13042.44
		GAP	94.03%	70.99%	132.68%	150.76%	73.69%	91.23%
		CPU (sec)	0.07	0.05	0.24	0.23	0.00	0.00
BEST UB			16129.63	11698.80	10496.90	7824.28	14538.30	11602.20
BEST GAP			56.38%	55.48%	116.74%	98.87%	51.02%	67.80%

			MIN					
			N50_D1-16_FC400	N50_D1-16_FC200	N50_D1-8_FC400	N50_D1-8_FC200	N50_D4-12_FC400	N50_D4-12_FC200
LB_1			8341.54	5941.54	3322.10	2722.10	8341.54	5941.54
LB_2			8730.44	6247.77	3890.09	3280.15	8667.80	6126.25
BEST LB			8730.44	6247.77	3890.09	3280.15	8667.80	6126.25
RSH_1	# of Rep	1000	11083.24	8388.26	8546.75	6098.62	12679.70	9730.59
		GAP	25.32%	34.26%	61.42%	59.71%	26.84%	32.98%
		CPU (sec)	23.66	22.95	97.03	100.67	3.34	2.25
RSH_2	# of Rep	1000	11304.91	8556.02	8717.68	6220.60	12695.80	9837.09
		GAP	27.82%	34.53%	64.27%	62.90%	29.38%	35.63%
		CPU (sec)	17.76	16.45	105.76	109.73	3.64	2.45
SSH_1	# of Rep	1	12516.73	9265.74	6570.08	4869.84	10965.84	8722.76
		GAP	36.40%	23.94%	38.19%	45.65%	12.09%	24.95%
		CPU (sec)	0.02	0.02	0.13	0.12	0.00	0.00
SSH_2	# of Rep	1	12797.60	9612.71	8181.79	5874.39	11368.01	9127.19
		GAP	41.02%	28.59%	64.36%	75.70%	16.20%	30.74%
		CPU (sec)	0.00	0.00	0.11	0.12	0.00	0.00
BEST UB			11083.24	8388.26	6570.08	4869.84	10965.84	8722.76
BEST GAP			25.32%	23.94%	38.19%	45.65%	12.09%	24.95%

Part 2.5 100 Customers

			MAX					
			N100_D1-16_FC400	N100_D1-16_FC200	N100_D1-8_FC400	N100_D1-8_FC200	N100_D4-12_FC400	N100_D4-12_FC200
LB_1			21535.85	15935.85	10785.66	7785.66	19795.20	14595.20
LB_2			22410.69	16801.26	11822.25	8784.26	20602.14	15390.66
BEST LB			22410.69	16801.26	11822.25	8784.26	20602.14	15390.66
RSH_1	# of Rep	1000	32095.82	24086.23	21813.60	17322.70	29274.50	23303.80
		GAP	54.64%	79.07%	100.55%	123.48%	59.56%	68.14%
		CPU (sec)	1060.41	923.46	2292.62	4665.54	168.26	164.31
RSH_2	# of Rep	1000	32320.49	24254.83	21883.10	17428.50	28836.90	23003.50
		GAP	55.72%	80.32%	100.72%	122.46%	60.68%	68.90%
		CPU (sec)	1102.82	960.40	2384.33	4852.16	174.99	170.88
SSH_1	# of Rep	1	35229.34	26437.77	22976.54	17905.80	29808.90	23608.90
		GAP	69.74%	96.55%	113.87%	124.42%	75.14%	84.10%
		CPU (sec)	0.55	0.67	1.72	2.24	0.08	0.08
SSH_2	# of Rep	1	35552.54	26680.31	25261.80	19895.80	31462.80	25062.80
		GAP	71.29%	98.36%	124.92%	149.20%	76.75%	92.70%
		CPU (sec)	1.35	0.75	1.72	2.24	0.18	0.18
BEST UB			32095.82	24086.23	20932.86	16067.72	28836.90	23003.50
BEST GAP			54.64%	79.07%	94.85%	104.58%	59.56%	68.14%

			MIN					
			N100_D1-16_FC400	N100_D1-16_FC200	N100_D1-8_FC400	N100_D1-8_FC200	N100_D4-12_FC400	N100_D4-12_FC200
LB_1			16316.07	11516.07	8025.93	5425.93	16092.57	11292.57
LB_2			16744.24	11952.86	9248.93	6618.29	16750.46	11946.11
BEST LB			16744.24	11952.86	9248.93	6618.29	16750.46	11946.11
RSH_1	# of Rep	1000	21909.89	16759.25	16945.62	11974.67	24462.10	19041.69
		GAP	30.85%	39.96%	52.86%	71.47%	28.23%	35.14%
		CPU (sec)	245.34	256.04	859.03	984.30	33.23	19.10
RSH_2	# of Rep	1000	22063.25	16876.57	17064.24	12058.49	24633.33	19174.98
		GAP	31.77%	40.94%	53.93%	72.67%	29.13%	36.08%
		CPU (sec)	143.89	150.46	893.39	1023.67	34.56	19.87
SSH_1	# of Rep	1	24048.95	18395.46	16425.30	12940.00	25511.90	20111.90
		GAP	33.80%	40.38%	52.37%	63.73%	40.75%	48.33%
		CPU (sec)	0.08	0.09	1.05	1.12	0.03	0.03
SSH_2	# of Rep	1	24269.58	18564.23	18770.66	13264.34	26857.00	21092.48
		GAP	41.35%	50.45%	69.33%	89.94%	42.04%	49.69%
		CPU (sec)	0.08	0.04	0.04	0.27	0.03	0.03
BEST UB			21909.89	16759.25	16425.30	11974.67	24462.10	19041.69
BEST GAP			30.85%	39.96%	52.37%	63.73%	28.23%	35.14%

Part 3.1 10 Customers

			MAX					
			N10_D1-16_FC400	N10_D1-16_FC200	N10_D1-8_FC400	N10_D1-8_FC200	N10_D4-12_FC400	N10_D4-12_FC200
OPT			5029.52	3429.52	2941.97	2141.97	4557.74	3157.74
OPT_GAP			8.77%	11.99%	0.00%	0.00%	13.07%	11.94%
CPU (sec)			2410.04	2712.89	814.79	966.92	2199.98	2300.10
LB_1			4718.11	3118.11	2463.79	1663.79	4117.10	2717.10
GAP_LB_1			20.31%	32.49%	44.02%	73.04%	21.09%	32.96%
LB_2			4758.44	3158.44	2527.17	1727.17	4162.81	2762.81
GAP_LB_2			19.88%	31.13%	37.11%	54.63%	17.11%	26.55%
BEST LB			4758.44	3158.44	2527.17	1727.17	4162.81	2762.81
BEST GAP			19.88%	31.13%	37.11%	54.63%	17.11%	26.55%
RSH_1	**# of Rep**	**1000**	5057.10	3457.10	2941.97	2141.97	4877.00	3246.06
		GAP	7.87%	7.32%	1.29%	1.79%	7.00%	3.93%
		CPU (sec)	0.41	0.39	0.87	1.23	1.39	0.40
RSH_2	**# of Rep**	**1000**	5057.10	3457.10	2941.97	2141.97	4877.00	3246.06
		GAP	7.87%	7.32%	1.29%	1.79%	7.00%	3.93%
		CPU (sec)	0.39	0.40	0.87	1.18	1.33	0.38
SSH_1	**# of Rep**	**1**	5562.81	3802.81	3125.45	2245.45	5958.41	3343.44
		GAP	12.18%	11.62%	10.00%	10.00%	50.00%	10.00%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
SSH_2	**# of Rep**	**1**	5607.31	3833.23	3150.46	2263.42	6006.07	3370.19
		GAP	12.74%	12.17%	10.88%	10.88%	51.20%	10.88%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
BEST UB			5057.10	3457.10	2941.97	2141.97	4877.00	3246.06
BEST GAP			7.87%	7.32%	1.29%	1.79%	7.00%	3.93%

			MIN					
			N10_D1-16_FC400	N10_D1-16_FC200	N10_D1-8_FC400	N10_D1-8_FC200	N10_D4-12_FC400	N10_D4-12_FC200
OPT			3208.92	2208.92	2073.28	1473.28	3216.76	2216.76
OPT_GAP			0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
CPU(sec)			112.18	423.53	18.99	21.92	170.61	164.32
LB_1			2667.19	1667.19	1510.45	910.45	2667.19	1667.19
GAP_LB_1			0.07	0.10	0.17	0.25	0.11	0.16
LB_2			2758.23	1758.23	1619.37	1019.37	2758.23	1758.23
GAP_LB_2			5.70%	8.58%	8.09%	11.84%	9.32%	13.85%
BEST LB			2758.23	1758.23	1619.37	1019.37	2758.23	1758.23
BEST GAP			5.70%	8.58%	8.09%	11.84%	9.32%	13.85%
RSH_1	**# of Rep**	**1000**	3208.92	2208.92	2073.28	1473.28	3272.83	2272.83
		GAP	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		CPU (sec)	0.13	0.17	0.23	0.41	0.12	0.08
RSH_2	**# of Rep**	**1000**	3208.92	2208.92	2073.28	1473.28	3272.83	2272.83
		GAP	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
		CPU (sec)	0.12	0.16	0.23	0.39	0.12	0.08
SSH_1	**# of Rep**	**1**	3337.28	2297.28	2156.21	1532.21	3403.74	2363.74
		GAP	3.00%	1.38%	3.00%	3.00%	3.00%	3.00%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
SSH_2	**# of Rep**	**1**	3363.98	2315.66	2173.46	1544.47	3430.97	2382.65
		GAP	3.51%	1.88%	3.82%	3.82%	3.82%	3.82%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
BEST UB			3208.92	2208.92	2073.28	1473.28	3272.83	2272.83
BEST GAP			0.00%	0.00%	0.00%	0.00%	0.00%	0.00%

Part 3.2 15 Customers

			MAX					
			N15_D1-16_FC400	N15_D1-16_FC200	N15_D1-8_FC400	N15_D1-8_FC200	N15_D4-12_FC400	N15_D4-12_FC200
LB_1			6538.01	4338.01	3518.87	2318.87	5258.79	3458.79
LB_2			6572.00	4372.00	3617.76	2417.76	5304.51	3504.51
LB_CPLEX			5951.69	3966.89	3842.39	2653.14	5462.77	3698.42
GAP_LB_CPLEX			24.52%	27.09%	16.53%	21.27%	18.20%	26.95%
BEST LB			6572.00	4372.00	3842.39	2653.14	5462.77	3698.42
RSH_1	# of Rep	1000	7298.88	5098.88	4237.26	3088.74	6437.60	4635.25
		GAP	26.28%	37.04%	27.95%	42.23%	17.84%	35.59%
		CPU (sec)	2.14	1.93	3.94	4.01	0.97	1.13
RSH_2	# of Rep	1000	7328.18	5106.21	4289.98	3129.79	6478.57	4678.57
		GAP	28.74%	37.04%	27.95%	43.25%	18.59%	35.60%
		CPU (sec)	2.78	2.42	3.50	4.00	1.19	1.11
SSH_1	# of Rep	1	7528.83	5328.83	4313.97	3298.10	7177.69	5048.47
		GAP	39.26%	54.35%	44.84%	72.01%	42.71%	54.66%
		CPU (sec)	0.00	0.00	0.01	1.00	0.00	0.00
SSH_2	# of Rep	1	7328.18	5128.18	4731.08	3531.08	7047.04	5240.56
		GAP	34.32%	45.48%	46.23%	69.84%	36.62%	55.23%
		CPU (sec)	0.00	0.00	0.00	1.00	0.00	0.00
BEST UB			7298.88	5098.88	4237.26	3088.74	6437.60	4635.25
BEST GAP			26.28%	37.04%	27.95%	42.23%	17.84%	35.59%
GAP (w/o CPLEX)			27.15%	41.73%	37.24%	60.42%	27.11%	41.32%

			MIN					
			N15_D1-16_FC400	N15_D1-16_FC200	N15_D1-8_FC400	N15_D1-8_FC200	N15_D4-12_FC400	N15_D4-12_FC200
LB_1			3779.89	2379.89	2073.03	1273.03	3779.89	2379.89
LB_2			3874.90	2474.90	2254.78	1454.78	3874.90	2474.90
LB_CPLEX			4007.40	2587.38	2552.08	1771.59	3872.96	2540.94
GAP_LB_CPLEX			9.81%	12.62%	5.27%	5.58%	5.58%	11.41%
BEST LB			4007.40	2587.38	2552.08	1771.59	3874.90	2540.94
RSH_1	# of Rep	1000	4539.37	3115.45	2806.13	2060.34	4515.36	3124.30
		GAP	11.06%	14.19%	6.74%	8.98%	4.48%	14.62%
		CPU (sec)	0.38	0.35	2.78	2.29	0.33	0.31
RSH_2	# of Rep	1000	4539.37	3115.45	2857.99	2060.34	4502.91	3102.91
		GAP	11.51%	14.19%	11.04%	10.58%	8.22%	14.62%
		CPU (sec)	0.37	0.36	2.12	2.11	0.33	0.33
SSH_1	# of Rep	1	4803.93	3326.23	2889.76	2089.76	5350.74	3750.74
		GAP	14.56%	18.23%	12.27%	17.55%	17.54%	24.37%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
SSH_2	# of Rep	1	4806.97	3406.97	3033.65	2233.65	5294.03	3694.03
		GAP	11.51%	17.30%	17.37%	19.25%	19.28%	27.71%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
BEST UB			4539.37	3115.45	2806.13	2060.34	4502.91	3102.91
BEST GAP			11.06%	14.19%	6.74%	8.98%	4.48%	14.62%
GAP (w/o CPLEX)			11.06%	16.63%	9.60%	18.57%	4.48%	16.29%

Part 3.3 20 Customers

			MAX					
			N20_D1-16_FC400	N20_D1-16_FC200	N20_D1-8_FC400	N20_D1-8_FC200	N20_D4-12_FC400	N20_D4-12_FC200
LB_1			7512.89	4912.89	3993.54	2593.54	6636.39	4436.39
LB_2			7580.92	4980.92	4257.21	2857.21	6800.94	4600.94
LB_CPLEX			7612.68	5015.50	4430.46	3013.29	6829.55	4615.08
GAP_LB_CPLEX			43.04%	49.00%	26.57%	34.63%	46.62%	56.16%
BEST LB			7612.68	5015.50	4430.46	3013.29	6829.55	4615.08
RSH_1	# of Rep	1000	8751.12	6151.12	5285.45	3933.45	8210.16	5868.70
		GAP	27.62%	39.74%	25.05%	40.48%	29.02%	41.79%
		CPU (sec)	4.93	4.31	10.35	11.05	2.29	2.05
RSH_2	# of Rep	1000	8751.12	6151.12	5227.94	3853.53	8216.42	5842.79
		GAP	29.01%	39.21%	26.02%	39.67%	29.02%	39.79%
		CPU (sec)	4.78	5.09	9.94	10.29	2.34	2.36
SSH_1	# of Rep	1	9176.66	6576.66	5811.20	4442.07	8944.92	6181.56
		GAP	31.52%	53.42%	35.36%	56.04%	40.85%	50.28%
		CPU (sec)	0.00	0.01	1.00	0.02	0.00	0.00
SSH_2	# of Rep	1	9085.87	6466.33	5856.47	4627.00	8672.28	6276.32
		GAP	35.22%	49.95%	48.30%	71.26%	36.55%	51.64%
		CPU (sec)	0.00	0.00	0.03	0.02	0.00	0.00
BEST UB			8751.12	6151.12	5227.94	3853.53	8210.16	5804.45
BEST GAP			27.62%	39.21%	25.05%	39.67%	29.02%	39.79%
GAP (w/o CPLEX)			28.88%	41.06%	31.09%	45.71%	29.26%	39.79%

			MIN					
			N20_D1-16_FC400	N20_D1-16_FC200	N20_D1-8_FC400	N20_D1-8_FC200	N20_D4-12_FC400	N20_D4-12_FC200
LB_1			4864.48	3064.48	2676.76	1676.76	5478.74	3478.74
LB_2			5143.86	3343.86	3012.70	2012.70	5712.59	3646.87
LB_CPLEX			5194.77	3388.36	3187.53	2162.51	5185.29	3384.89
GAP_LB_CPLEX			19.62%	31.68%	13.02%	19.76%	28.77%	33.18%
BEST LB			5194.77	3388.36	3187.53	2162.51	5712.59	3646.87
RSH_1	# of Rep	1000	6629.48	4646.54	3943.15	2932.42	6482.34	4482.34
		GAP	12.58%	18.43%	16.11%	22.19%	12.80%	19.26%
		CPU (sec)	1.41	1.11	6.70	6.41	1.08	0.95
SSH_1	# of Rep	1000	6701.69	4612.46	3943.16	2924.65	6467.91	4467.91
		GAP	14.08%	17.56%	14.75%	22.50%	12.61%	19.34%
		CPU (sec)	0.90	0.86	6.11	5.71	1.00	1.02
RSH_2	# of Rep	1	6832.02	4763.72	4314.50	3076.26	6655.17	4655.17
		GAP	20.15%	25.32%	24.74%	29.45%	15.87%	24.34%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
SSH_2	# of Rep	1	6915.67	4715.67	4447.20	3447.20	7024.68	5024.68
		GAP	16.75%	20.19%	25.02%	37.36%	17.23%	25.81%
		CPU (sec)	0.00	0.00	0.00	0.00	0.00	0.00
BEST UB			6629.48	4612.46	3943.15	2924.65	6467.91	4467.91
BEST GAP			12.58%	17.56%	14.75%	22.19%	12.61%	19.26%
GAP (w/o CPLEX)			13.05%	17.56%	17.99%	26.95%	12.61%	19.34%

Part 3.4 50 Customers

			MAX					
			N50_D1-16_FC400	N50_D1-16_FC200	N50_D1-8_FC400	N50_D1-8_FC200	N50_D4-12_FC400	N50_D4-12_FC200
LB_1			17323.53	11323.53	8908.31	5874.57	15470.89	10226.11
LB_2			17152.03	11496.11	9422.61	6418.47	15838.36	10637.04
BEST LB			17323.53	11496.11	9422.61	6418.47	15838.36	10637.04
RSH_1	# of Rep	1000	20871.17	14599.90	13249.22	9823.92	19207.30	14010.00
		GAP	31.87%	40.27%	50.59%	75.57%	32.10%	49.98%
		CPU (sec)	103.83	115.51	260.23	262.68	62.94	65.30
RSH_2	# of Rep	1000	20871.17	14727.75	13433.18	9683.17	19408.90	13983.90
		GAP	36.51%	41.22%	51.67%	75.73%	32.40%	48.51%
		CPU (sec)	105.09	120.61	229.70	270.07	53.11	54.31
SSH_1	# of Rep	1	22871.17	15240.85	12701.15	9310.79	20754.33	15170.05
		GAP	38.68%	51.04%	52.15%	79.78%	37.50%	59.71%
		CPU (sec)	0.10	0.09	0.28	0.30	0.08	0.10
SSH_2	# of Rep	1	23271.17	15614.30	15402.57	11848.21	20811.90	15411.90
		GAP	39.50%	52.17%	73.90%	102.10%	42.61%	62.99%
		CPU (sec)	0.09	0.12	0.27	0.26	0.08	0.10
BEST UB			20871.17	14573.11	12701.15	9310.79	19207.30	13983.90
BEST GAP			31.46%	40.27%	50.21%	74.96%	32.10%	48.13%

			MIN					
			N50_D1-16_FC400	N50_D1-16_FC200	N50_D1-8_FC400	N50_D1-8_FC200	N50_D4-12_FC400	N50_D4-12_FC200
LB_1			12741.54	8141.54	6895.94	4295.94	12741.54	8141.54
LB_2			13130.44	8537.15	7533.61	4816.61	13067.80	8472.17
BEST LB			13130.44	8537.15	7533.61	4816.61	13067.80	8472.17
RSH_1	# of Rep	1000	16475.20	11732.90	10372.60	7488.48	17114.50	11304.87
		GAP	16.43%	15.65%	24.56%	27.35%	17.30%	17.99%
		CPU (sec)	40.78	43.71	169.33	169.32	19.72	19.67
RSH_2	# of Rep	1000	16449.80	11780.50	10441.10	7504.34	16639.30	11035.73
		GAP	16.43%	19.42%	25.15%	29.23%	17.04%	15.18%
		CPU (sec)	38.42	39.70	163.55	130.00	14.73	17.04
SSH_1	# of Rep	1	17246.90	11759.23	10906.20	7192.48	17968.40	12021.64
		GAP	19.80%	28.59%	22.83%	22.32%	21.72%	25.47%
		CPU (sec)	0.04	0.05	0.20	0.17	0.03	0.02
SSH_2	# of Rep	1	17281.10	12481.10	11043.61	7994.27	17622.20	12401.73
		GAP	19.07%	23.04%	31.12%	36.61%	23.48%	29.44%
		CPU (sec)	0.03	0.03	0.18	0.18	0.03	0.04
BEST UB			16449.80	11732.90	10372.60	7192.48	16639.30	11035.73
BEST GAP			16.43%	15.65%	22.83%	22.32%	17.04%	15.18%

Part 3.5 100 Customers

			MAX					
			N100_D1-16_FC400	N100_D1-16_FC200	N100_D1-8_FC400	N100_D1-8_FC200	N100_D4-12_FC400	N100_D4-12_FC200
LB_1			32735.85	21535.85	16785.66	10785.66	30195.20	19795.20
LB_2			33610.69	22401.26	17822.25	11784.26	31002.14	20590.66
BEST LB			37069.90	22401.26	17822.25	11784.26	31002.14	20590.66
RSH_1	# of Rep	1000	46007.81	32513.14	29326.33	21779.16	41906.10	29651.42
		GAP	55.69%	67.20%	81.66%	106.56%	48.24%	65.09%
		CPU (sec)	1188.78	1487.24	3027.55	3344.93	669.48	758.57
RSH_2	# of Rep	1000	45602.69	32643.82	29597.60	21525.06	40959.31	28721.13
		GAP	64.87%	64.52%	83.34%	110.56%	54.62%	65.60%
		CPU (sec)	1086.84	1304.16	2438.48	2452.66	409.25	567.66
SSH_1	# of Rep	1	47956.68	32357.52	27433.35	20273.45	44872.13	32898.86
		GAP	80.00%	60.31%	69.93%	89.55%	54.32%	72.37%
		CPU (sec)	1.47	1.37	3.34	5.38	1.28	1.08
SSH_2	# of Rep	1	49077.62	34226.77	35032.07	27239.29	44567.23	31222.74
		GAP	86.29%	104.30%	136.72%	184.24%	65.77%	82.56%
		CPU (sec)	1.28	1.29	3.84	4.05	0.51	0.83
BEST UB			45602.69	32357.52	27433.35	19138.36	40959.31	28637.66
BEST GAP			48.17%	60.31%	69.93%	81.25%	48.24%	61.85%

			MIN					
			N100_D1-16_FC400	N100_D1-16_FC200	N100_D1-8_FC400	N100_D1-8_FC200	N100_D4-12_FC400	N100_D4-12_FC200
LB_1			25916.07	16316.07	13225.93	8025.93	25312.15	15912.15
LB_2			26344.24	16752.86	14448.93	9218.29	25730.34	16338.12
BEST LB			26344.24	16752.86	14448.93	9218.29	25730.34	16338.12
RSH_1	# of Rep	1000	34798.40	24795.70	21529.50	16234.90	34560.60	24737.22
		GAP	18.37%	32.50%	40.64%	57.28%	24.23%	38.11%
		CPU (sec)	462.18	425.22	1964.71	2116.98	208.28	213.70
RSH_2	# of Rep	1000	34871.90	24693.90	21678.10	16168.50	34581.10	24885.90
		GAP	20.13%	30.86%	38.74%	54.70%	21.61%	38.61%
		CPU (sec)	365.61	372.25	1890.16	1508.55	157.10	169.57
SSH_1	# of Rep	1	34743.20	25120.60	20794.50	15282.40	34813.70	24482.12
		GAP	21.94%	31.07%	32.47%	45.81%	22.55%	38.95%
		CPU (sec)	0.45	0.34	2.06	1.30	0.18	0.26
SSH_2	# of Rep	1	36047.10	25463.58	22835.90	17381.42	35272.20	25500.22
		GAP	25.31%	34.60%	36.50%	56.79%	28.50%	42.32%
		CPU (sec)	0.26	0.17	1.92	1.70	0.25	0.19
BEST UB			34743.20	24693.90	20794.50	15282.40	34560.60	24482.12
BEST GAP			18.37%	30.86%	32.47%	45.81%	21.61%	38.11%