A TWO STAGE SOLUTION APPROACH TO SPARE PARTS DISTRIBUTION UNDER A SPECIAL COST STRUCTURE

A THESIS

SUBMITTED TO THE DEPARTMENT OF INDUSTRIAL ENGINEERING

AND THE INSTITUTE OF ENGINEERING AND SCIENCE

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF SCIENCE

By

Esra Koca

July, 2010

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

Assoc. Prof. Emre Alper Yıldırım (Advisor)

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

Asst. Prof. Osman Alp

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

Asst. Prof. İbrahim Körpeoğlu

Approved for the Institute of Engineering and Science:

Prof. Levent Onural
Director of the Institute

ABSTRACT

A TWO STAGE SOLUTION APPROACH TO SPARE PARTS DISTRIBUTION UNDER A SPECIAL COST STRUCTURE

Esra Koca
M.S. in Industrial Engineering
Supervisor: Assoc. Prof. Emre Alper Yıldırım
July, 2010

In this thesis, we consider a multicommodity distribution problem. We assume that there is a central depot which houses a number of different types of items. There is a finite number of geographically dispersed demand points which place orders for these items on a daily basis. The demand of these demand points should be satisfied from this central depot. We assume that a finite number of identical trucks with predetermined destinations are used for the distribution of the items from the central depot to each demand point. The demand of each demand point can be split among several trucks and a single truck is allowed to visit several demand points. Our objective is to satisfy the demand of each demand point with the minimum total distribution cost while respecting the capacity of each truck. The cost structure is dictated by the final destinations of trucks used in the distribution of the items and the set of demand points visited by each truck. We propose two different solution approaches. The first approach, called the Direct Approach, is aimed at solving the problem directly using a mixed integer linear programming formulation. Since the Direct Approach becomes computationally infeasible for real-life problems, we propose a so-called Hierarchical Approach that is aimed at solving the problem in two stages using an aggregation followed by a disaggregation scheme. We study the properties of the solutions computed with the Hierarchical Approach. We perform extensive computational studies on a data set adapted from a major automotive manufacturing company in Turkey in an attempt to compare the performances of the two approaches. Our results reveal that the Hierarchical Approach significantly outperforms the Direct Approach on the vast majority of the instances.

Keywords: multicommodity distribution, transportation, logistics.

ÖZET

ÖZEL MALİYET YAPISI ALTINDA YEDEK PARÇA DAĞITIMI İÇİN İKİ AŞAMALI ÇÖZÜM YÖNTEMİ

Esra Koca
Endüstri Mühendisliği, Yüksek Lisans
Tez Yöneticisi: Doç. Dr. Emre Alper Yıldırım
Temmuz, 2010

Bu tez çalışmasında çok ürünlü bir dağıtım problemi üzerine çalıştık. Farklı çeşitte parçaların bulunduğu merkez depodan istek noktalarına kamyonlarla dağıtım yapılmaktadır. İstek noktalarının herbir parça çeşidi için günlük talepleri bu kamyonlar ile sağlanmalıdır. Bir istek noktasının talebi birden fazla kamyonla sağlanabilir. Benzer sekilde bir kamyon birden fazla istek noktasının talebini taşıyabilir. Belirli sayıda son durağı bilinen türdeş kamyonlar olduğunu varsaydık. Taşımacılık maliyeti olarak iki çeşit ücret vardır: son durak ücreti ve ugrama ücreti. Her bir istek noktası için belirli miktarda bir son durak ücreti vardır. Eğer bir kamyon taşımacılıkta kullanılıyorsa o kamyonun son surak ücreti ödenmelidir. Aynı şekilde, her bir istek noktası ikilisi için de belirli bir uğrama ücreti vardır. Eğer bir kamyon son durağı dışında bir istek noktasına da parça taşıyorsa o istek noktası için uğrama ücreti ödenmelidir. Problem her bir istek noktasının her bir parça çeşidi için taleplerini kamyon kapasitelerine uygun bir şekilde en az taşımacılık maliyeti ile saglamaktır. Problemi çözmek için iki farklı çözüm yöntemi geliştirdik: Doğrudan Çözüm Yöntemi ve Aşamalı Çözüm Yöntemi. Bu iki çözüm yönteminin sonuçlarını analiz edip karşılaştırdık. Türkiye'nin önde gelen otomobil üreticilerinden birisinden elde ettiğimiz günlük veriler ile çözüm yöntemlerimizi test ettik. Elde ettiğimiz sonuçlara göre aşamalı çözüm yönteminin bu problemi çözmek için daha efektif bir cözüm yöntemi olduğu sonucuna vardık.

Anahtar sözcükler: çok ürünlü dağıtım, taşımacılık, lojistik.

Acknowledgement

I would like to express my deepest and most sincere gratitude to my advisor, Assoc. Prof. Emre Alper Yıldırım for his invaluable trust, support, guidance and motivation during my graduate study. I feel lucky to have such a great advisor and mentor.

I am indebted to my dissertation committee, Asst. Prof. Osman Alp and Asst. Prof. İbrahim Körpeoğlu for accepting to read and review this thesis and for their recommendations.

I would like to express my deepest gratitude to my family for their perpetual love and trust. It is invaluable for me to feel that they are always proud of me. I am especially grateful to my brother Ender Koca for his support and love.

I wish to express my special thanks to my friend and house mate Esra Çelik and my intimate friend Ece Zeliha Demirci, for their great friendship and everlasting support. We are the friends from our first day in Bilkent and I believe that our friendship will be forever.

I am grateful to my friends Feyza Kazanç, Hatice Çalık, Ayşe Çelikbaş, Burak Paç, Efe Burak Bozkaya and Can Öz for their valuable friendship and support.

I would like to thank all my friends and I feel very lucky to have so many great people around me. I sincerely apologize from all of my friends whose names are not stated here.

I am also grateful to TÜBİTAK for their support during my graduate study.

Finally, I would like to thank Rıza Bozoklar, Sinan Südütemiz, Bülent Ercan and Hasan Yonar for their support.

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

Logistics is the management of the flow of material, service, information and capital between the origin point and the consumption point in order to satisfy requirements of the consumers. It consists of integration of information, transportation, warehousing, inventory, material handling and packaging. Today logistics is one of the important functions in business.

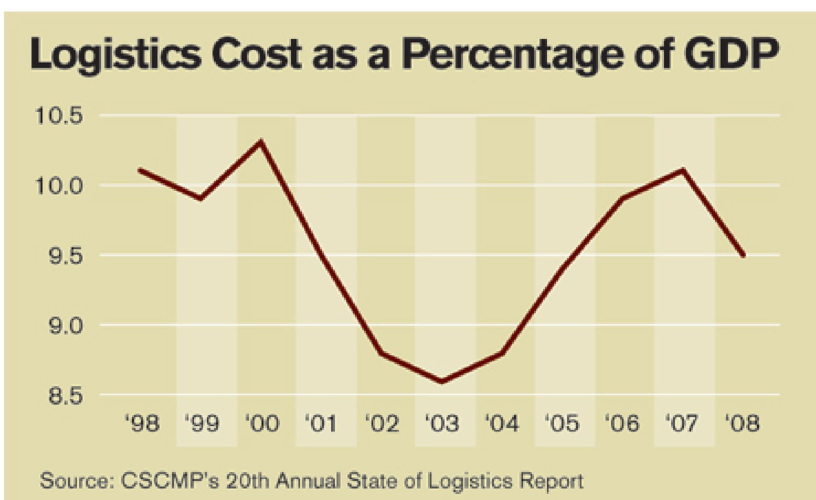

Figure 1.1: Logistics Cost as a Percentage of GDP

The significance of logistics can be understood by examining annual logistics costs of the countries. Cost of logistics always constitutes a large percentage of Gross Domestic Product (GDP) in the U.S. As seen in Figure 1.1, in the last ten years, logistics cost in each year in the U.S. has corresponded to more than 8.5 percent of that year's GDP. This situation makes logistics an important part of the economy.

According to the 20^{th} Annual State of Logistics Report of Council of Supply Chain Management Professionals (CSCMP), U.S. business logistics cost was $1,344 billion which is equal to 9.4 percent of U.S. GDP in 2008. In addition, as seen in Figure 1.2, transportation cost was $872 billion which is about 65 percent of the total logistics cost. This makes transportation as the most important part of logistics functions.

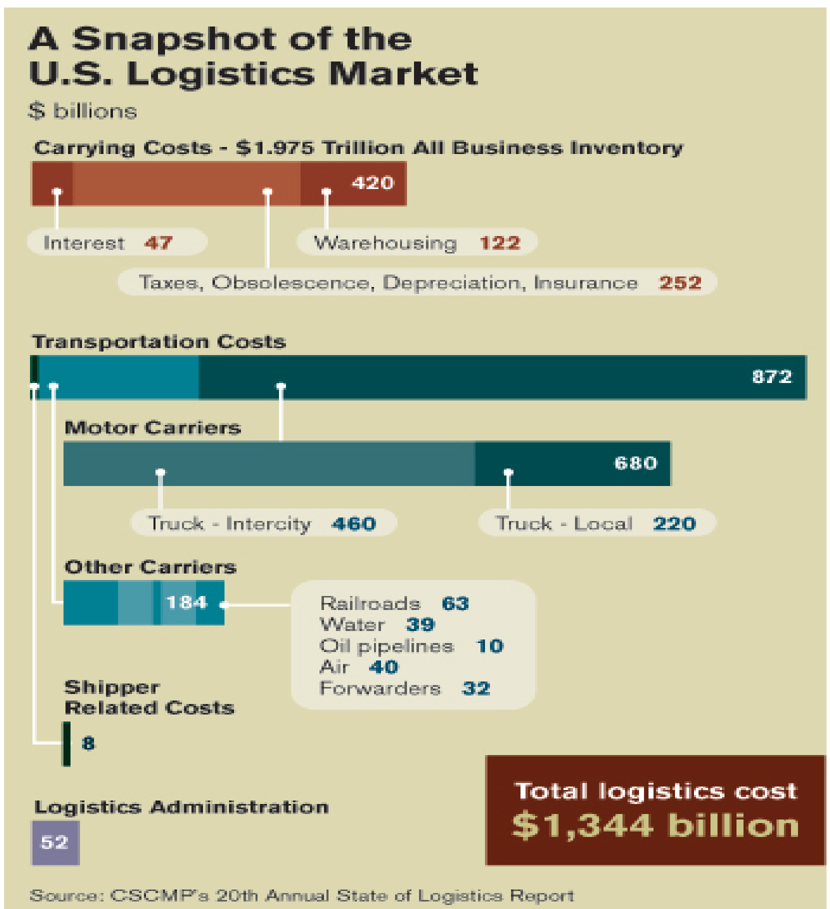

Figure 1.2: Distribution of Logistics Costs of U.S. in 2008

A similar situation is valid for Turkey. Transportation costs and incomes of Turkey between the years 2000 and 2007 can be seen in Figure 1.3 [48]. Transportation cost of Turkey in 2007 was $6.268 billion USD whereas its GDP in 2007 was \$663.419 billion USD. Therefore, Turkey's transportation cost in 2007 is equal to 9.45 percent of its GDP. In addition, the transportation income of Turkey in 2007 was 6.104 billion USD. According to the logistics report for 2007 which is prepared by UTIKAD, the Freight Forwarders and Logistics Service Providers Association in Turkey, respecting the data of Central Bank of the Republic of Turkey, the reason of this situation is that Turkey could not take advantage of its own logistics resources in the international trades. Therefore, there is an effort to reduce the transportation costs and improve the logistics activities in Turkey.

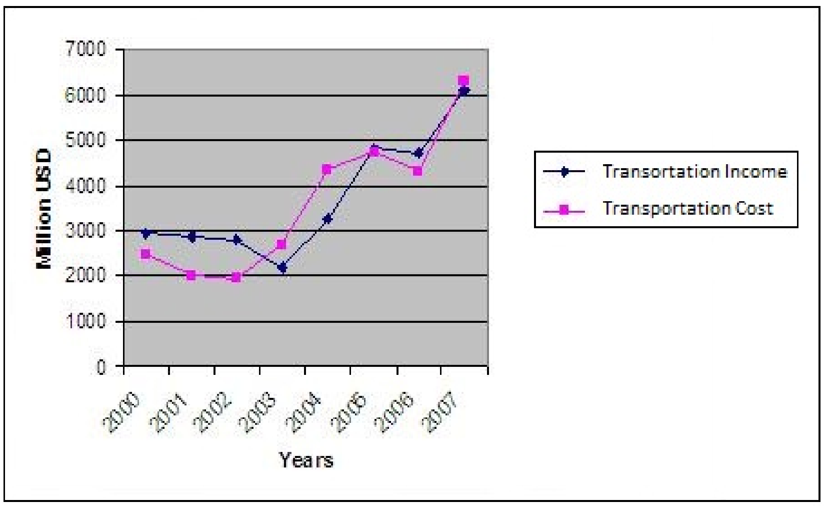

Figure 1.3: Transportation Costs and Incomes of Turkey Between 2000 and 2007

The importance of the logistics does not only lie on its high costs. Due to globalization, competition has increased and companies try to survive in the new global business market by improving their productivity and customer service. Therefore, quick delivery of goods to customers is important for companies in order to stay in the market. Quick deliveries is possible if the companies manage their logistics well.

Another aspect that requires companies to manage their logistics well is the environmental issues. An effective transportation means less fuel consumption and less environmental pollution. Moreover as fuel costs are high, it is also helpful for companies to reduce their costs.

As explained above, it is necessary and beneficial for companies to manage their logistics well. Transportation is one of the most important parts of the logistics and it should be done in a cost-efficient way for the sake of the companies. The problem we study in this thesis is inspired by a logistics problem of one of the major automotive companies in Turkey. The company has a warehouse for spare parts in its facility, and there are retailers all around Turkey that are served by this warehouse. The company outsources its logistics and the distribution of the spare parts to the retailers is done by trucks of the logistics company. Each day demands of some or all of the retailers for the spare parts should be satisfied by the trucks. The problem is to satisfy the demand of each retailer for each spare part type with the minimum transportation cost. Although the problem is inspired by an automotive company, many logistics problems can be cast in this setting. There is a central depot in which there are different types of items that can be ordered by any one of the demand points. Distribution of the items from the central depot to the demand points is done by trucks. A truck can carry items for more than one demand point and a demand point's demand can be satisfied by more than one truck. The problem is to satisfy the daily demand of each demand point for each item type with the minimum transportation cost. Many logistics problems can be cast in this setting. Consequently, many real life distribution problems can be solved by the solution method proposed for this problem.

The problem we study in this thesis is not a classical routing problem. The main difference between our problem and a routing problem is the cost structure. There are two types of costs in our problem: final destination costs and visiting costs. There is a fixed number of trucks each having certain final destinations and final destination costs. A truck may carry demand for demand points other than its final destination. In this situation, an additional visiting cost is paid for each visit of the truck to the demand points. The visiting cost depends on the final destination of the truck and the demand point that is visited. When the final destination of a truck and the demand points that will be visited are known, the route that the truck will follow is fixed. Therefore, we do not need to route the trucks that will be used; we need to choose the trucks that will be used and decide how to satisfy demand of each demand point for each item type with the minimum transportation cost. Therefore, the problem we study is not a classical routing problem, it is more similar to a capacity allocation problem. Each truck has a limited capacity and a fixed final destination cost. We need to determine the trucks that will be used in the distribution and allocate capacities of the trucks to the demand of each demand point with minimum cost.

In the next section, we provide a review of the literature on the distribution problems.

1.1 Literature Survey

We review the literature in order to determine the problems related to our problem and the solution techniques presented for solving them. We find out that the problems that are related to distribution problems are the Traveling Salesman Problem (TSP), Multi-Traveling Salesman Problem (mTSP) and Vehicle Routing Problem (VRP). In this section, we briefly give the definitions of these problems and the solutions techniques developed for each problem type.

TSP is the problem of finding a path through a weighted graph that starts and ends at the same vertex and visits every other vertex in the graph exactly once so that the total weight of the path is minimized. Although TSP related problems were treated in 1800s by the mathematicians W.R. Hamilton and Thomas Kirkman, the general form of the TSP was first studied starting in the 1930s by mathematicians Karl Menger in Vienna and Harvard. TSP is an NP-Hard Problem but many of the special cases of the TSP can be solved efficiently in polynomial time. Dantzig, Fulkerson and Johnson [13], solved a 49-city problem with the linear programming approach. They used subtour restrictions in their solution approach. In the survey of Bellmore and Nemhauser [7], several exact and approximate solution methods for the TSP are reviewed. There are three fundamentally different solution generation ways: tour-to-tour improvement, based on finding a better tour that is a neighbor of the present tour; tour building, based on building a sequence by successively including other nodes into the present sequence until a tour is obtained; subtour elimination, starting from an optimal solution to the assignment problem under the matrix C, subtours are eliminated iteratively until a tour is obtained. All procedures of the tour-to-tour improvement are approximate. Exact tour building algorithms are dynamic programming [6],[25],[28], the branch and bound algorithms of Little et al. [36], and Hatfield and Pierce [27]. Exact subtour elimination methods are integer linear programming [13], [8], [37], [38], the branch and bound algorithm of Eastman [17], and the Gilmore-Gomory method [23]. There are also partitioning and decomposition methods used by Held and Karp [28] and Karg and Thompson [31] in order to obtain approximate solutions to the TSP.

There exists several families of heuristics for the TSP. These can be classified into three categories: constructive heuristics, improvement heuristics, and composite heuristics consisting of a tour construction phase followed by an improvement phase. Renaud, Boctor and Laporte [41], introduced a fast composite heuristic for the symmetric TSP. In addition, well solvable special cases of the TSP are researched in the survey by Burkard, Deineko, Van Dal, Van Der Veen and Woeginger [10].

A generalization of the TSP is the multiple traveling salesman problem (mTSP). In mTSP there are m salesmen located at a single vertex and the problem is to determine tours for each of the m salesmen through a weighted graph that starts and ends at the same vertex and visits every other vertex in the graph exactly once so that the total weight of the tours is minimized [5]. Compared to the TSP, the mTSP is more adequate to model real life situations, since it is capable of handling more than one salesman. Bektas [30], reviews the literature and describes exact and heuristic solution approaches proposed for the problem. There are different types of integer programming formulations for the mTSP: in the assignment-based formulations, subtour elimination constraints (SECs) are used in order to get a proper solution for the problem. Several types of SECs are proposed in Datzig et. al.[13], Miller et. al.[38], Gavish [21], Kara and Bektas [30]. Laporte and Nobert [32], presented two different formulations for the mTSP. A k-degree centre tree-based formulation and a flow based formulation are due to Christofides et al. [11]. Laporte and Nobert [32] proposed an exact solution algorithm based on the relaxation of the some of the constraints of the problem and they introduce SECs iteratively. Ali and Kernington [1] proposed a branch and bound algorithm for the asymmetric mTSP. Gavish and Srikanth [22] attempt to solve a large scale mTSP and another exact solution method is proposed by Gromicho et al.[26].

mTSP can be considered as a relaxation of the VRP, which deals with designing a set of vehicle routes of least cost in such a way that each customer is visited exactly once by exactly one vehicle, the total demand of any route does not exceed the vehicle capacity and all the routes start and end at the depot. Most of the distribution and logistics problems are modeled as the VRP. There exists a broad literature on this problem and consequently there are many exact and approximate solution algorithms.

The VRP was first introduced by Dantzig and Ramser [14], but they called it "truck dispatching problem". In this paper, an approach for obtaining a near optimal solution was proposed. Five years later, Clarke and Wright [12], modified the solution approach of Dantzig and Ramser and developed an effective greedy heuristic. The algorithm of Clarke and Wright, called "savings algorithms", first creates vehicle routes containing the depot and one other vertex and then merges the routes according to the largest saving in the total cost. Several improvements to these algorithms have been proposed by Gaskell [20], Yellow [47], Golden et al. [24], Paessens [40], and Nelson et al. [39].

Another heuristic for solving the VRP is the sweep algorithm which is proposed in a book by Wren [45] and a paper by Wren and Holliday [46]. In this algorithm clusters are initially formed by rotating a ray centered at the depot and then TSP for each cluster is solved. An extension of the sweep algorithm which is called the Petal algorithm is another heuristic that is developed by Balinski and Quandt [4]. This algorithm first generates routes which are called "petals" and then selects the routes that will be used among them by solving a set partitioning problem. Several improvements to the petals algorithms are proposed by Foster and Ryan [19], Ryan et al. [43] and Renaud et al. [42].

The two phase method of Fisher and Jaikumar [18] is called the cluster first, route second algorithm. In this algorithm customers are first allocated to clusters first and then TSP is solved for each cluster. A generalized assignment problem is solved to form clusters. Bramel and Simchi-Levi [9] developed a two phase heuristic in which seeds are determined by solutions of the capacitated location problems and the remaining vertices are gradually added into their allotted route in the second stage. Most of the classical and modern heuristics and exact approaches developed for the VRP can be found in the surveys of Laporte [33], Laporte et al. [34] and Toth and Vigo [44].

There are several variations of VRP like VRP with capacity restrictions (CVRP), time windows (VRPTW), pick-up and delivery (VRPPD), time dependent travel times, uncertain demand and messy cost functions. However, none of these variations are suitable for our problem. In the TSP, mTSP, and VRP, it is assumed that each vertex, or customer, should be visited once in total whereas in our problem there is no restriction on the number of visits to any demand point. The demand of any demand point may be split into several trucks in our problem. However, recently a new VRP variant is introduced to the VRP literature which is called the Split Delivery Vehicle Routing Problem (SDVRP) by Dror and Trudeau [15]. The SDVRP allows the delivery to a demand point to be split between two or more vehicles. In many cases, allowing split deliveries yields savings in both the total distance traveled and the number of vehicles required.

There are few exact algorithms in the SDVRP literature. Dror et al.[16] solved the problem with a mixed integer programming approach using several valid inequalities. Their method optimally solves small instances of the problem with up to 10 demand points. They used heuristic methods rather than an exact solution method to obtain feasible solutions. Lee et al. [35] developed a dynamic programming model with finite state and action spaces. Their largest instance consists of nine demand points and six vehicles. Archetti et al.[3],[2] performed the worst case analysis for the SDVRP and developed a tabu search algorithm for the SDVRP, respectively. The SDVRP in not as widely studied as other variants of the VRP, like CVRP, VRPTW, VRPPD. Exact algorithms in the literature can only solve small SDVRP instances.

Jin, Liu and Bowden [29] developed a two stage algorithm with valid inequalities for the SDVRP. They adopt the formulation of Dror et al.[16] and Lee et al. [35]. Their solution approach gives good results for the problem.

The problem we study is similar to the SDVRP. However, in SDVRP it is assumed that the cost structure is symmetric and has the triangular property whereas we do not restrict the cost in our problem in any way. Therefore, the problem we study is different than their problem. In addition, as explained in the previous section, the cost structure of the problem makes it different from a routing problem. Therefore, we do not use their algorithm but develop our specific solution approach.

1.2 Thesis Overview

The rest of the thesis can be summarized as follows. In the next chapter, we define our problem and present the solution approaches we develop in order to solve the problem. In Chapter 3, the solutions of the models that are introduced in Chapter 2 are analyzed and compared. Numerical results and several comparisons are presented in Chapter 4. Finally, we present our concluding remarks in Chapter 5.

Chapter 2

Problem Definition and Optimization Models

In the previous chapter, we gave a short description of the problem we studied in this thesis. In this chapter, in Section 2.1 we formally define the problem. After that, we present the solution approaches developed to solve the problem. We developed two different solution approaches: The Direct Approach and The Hierarchical Approach. In Section 2.2, we explain these solution approaches in detail introducing the models used in these solution approaches.

2.1 Problem Definition

Assume that there is a central depot from which a number of items should be distributed to demand points so as to satisfy their demand. Distribution of items is carried out by trucks and the goal is to minimize the total distribution cost. There are M demand points whose demand should be satisfied by the central depot.

A truck that will be used in the distribution may satisfy more than one point's demand and the demand of any one of the points may be satisfied by more than one truck. Each truck that is used in the distribution will have a fixed final destination which is also one of the demand points.

We assume that the transportation costs are primarily determined by the final destination of a truck. For each demand point j, $j = 1, \dots, M$, there is a final destination cost of f_j. If a truck whose final destination is the demand point j used in the distribution of the items, then f_j is paid as the final destination cost of that truck. If a truck whose final destination is the demand point j, $j = 1, \dots, M$, also carries load for the demand point i, then $s_{i,j}$, $i = 1, \dots, M$, $i \neq j$, will be paid for the visit of that truck to the demand point i.

The shortest path between the depot and the demand point j, $j = 1, \dots, M$ is assumed to be known. If a truck whose final destination is the demand point j, $j = 1, \dots, M$ is used in the distribution, then this shortest path will be used. Therefore, when the final destination of a truck is known, then the route that it should follow becomes fixed. Since the final destination of each truck is known, all of these trucks have certain routes to follow. In addition, for each demand point pair (i, j), $i = 1, \dots, M$, $j = 1, \dots, M$, $i \neq j$, there is a fixed route to be followed for the visit of a truck whose final destination is the demand point j to the demand point i. Therefore, if we know the truck that will be used and the demand points it will visit, then the route it will follow becomes fixed. Consequently, the problem under consideration is different from a typical routing problem.

A visit of a truck whose final destination is the demand point j, $j = 1, \dots, M$, to the demand point i, $i = 1, \dots, M$, $i \neq j$ is carried out by the smallest possible deviation from the route of the truck. The value of $s_{i,j}$, $i = 1, \dots, M$, $j = 1, \dots, M$, depends on the position of i with respect to the shortest path between the depot and the demand point j. Therefore, $s_{i,j}$ is not symmetric, in general. There may be some $i = 1, \dots, M$, $j = 1, \dots, M$, $i \neq j$ such that $s_{i,j} \neq s_{j,i}$. In addition, $s_{i,j}$ does not directly depend on the length of the deviation from the shortest path to j. It can be seen the cost for the combination of the extra time passed, additional distance traveled, extra fuel consumed, etc. for the visit of the truck to i.

If $s_{i,j} > f_i$, then $s_{i,j} + f_j > f_i + f_j$. Consequently, a truck whose final destination is the demand point j does not visit the demand point i in any optimal solution as it is obvious that sending one truck to the demand point i and one truck to the demand point j costs less. In this case, we let $s_{i,j} = +\infty$.

Both f_j, $j = 1, \ldots, M$, and $s_{i,j}$, $i = 1, \ldots, M$, $j = 1, \ldots, M$, $i \neq j$ are assumed to be known. This cost structure can be represented by defining an $M \times M$ cost matrix C whose elements are composed of visiting and final destination costs such that:

$$C_{i,j} = \begin{cases} s_{i,j} & \text{if } i \neq j, \\ f_j & \text{if } i = j, \end{cases}$$

for $i = 1, \ldots, M$, $j = 1, \ldots, M$. The cost matrix C is an input to our problem.

We assume that costs are additive and that they are independent of one another. If a truck whose final destination is the demand point j visits only demand points i and k on the way, then the total cost for that truck will be $f_j + s_{i,j} + s_{k,j}$.

We assume that there is a fixed number of trucks whose final destination is the demand point j, $j = 1, \ldots, M$ and we denote the number of trucks whose final destination is the demand point j by t_j, $j = 1, \ldots, M$. We represent the set of these trucks by $T_j = \{1, \ldots, t_j\}$. Overall, there are $T = \sum_{j=1}^{M} t_j$ trucks. We assume that all of the trucks are identical.

There are N items that can be ordered by the demand points. The demand of each demand point for each item type is denoted by $D_{i,l}$ for $i = 1, \ldots, M$ and $l = 1, \ldots, N$. In order to determine the capacity of a truck allocated to one unit of a single item, we use the measure of a truckload. The capacity of a truck that is occupied by one unit of item type l, $l = 1, \ldots, N$ is denoted by w_l so that $0 < w_l \leq 1$. $w_l = 1/n_l$, where n_l is the largest number of item type l that can be loaded into a truck for each $l = 1, \ldots, N$.

We assume that any combination of items can be loaded into trucks as long as their total volume is less then or equal to one truckload. This assumption seems restrictive since in fact it depends on the shapes of the items. There may be some combinations of the items that cannot be loaded into trucks despite the fact that their total volume is less than or equal to one truckload. The geometry of the items may affect the number of items that can be loaded into a truck. However, this restriction can be avoided by expanding the assumption if necessary. Without this assumption, the problem turns into a 3-D bin packing problem, which is a strongly NP-hard problem.

Our main goal is to satisfy the demand of each demand point for each item type with the minimum transportation cost while respecting the capacity of trucks. As stated above, as each truck has a certain final destination, each truck has a certain route and we do not need to determine routes for the trucks. We only need to decide which trucks to use and how to allocate demands of the demand points to these trucks so that the total transportation cost is minimized. The actual problem is to allocate demands of the demand points to trucks so as to minimize the total transportation cost. Therefore, our problem is more similar to a resource allocation problem.

The problem was inspired by a major automotive manufacturer in Turkey. The automotive company outsources its logistics from a logistics company and this company uses the cost structure explained above. The company has a warehouse for spare parts in its facility, and there are retailers all around Turkey that are served by this warehouse. The distribution of spare parts to the retailers is performed by trucks of the logistics company. For each retailer, there is a fixed route to be followed. For instance, if the final destination of a truck and retailers that it will visit are known, then the route that the truck should follow becomes fixed. The cost structure is determined by these fixed routes.

Many logistics problems can be cast in this setting. For instance, the central depot may represent the warehouse of a manufacturing facility and demand points may represent distribution centers; or the central depot may be a distribution center of an automobile company and the demand points may be retailers. The problem introduced in the chapter is so general that many real life problems can be solved by the solution method proposed for this problem.

In the following section, we introduce our optimization models that will form a basis for our solution approach.

2.2 Optimization Models

In this section, we present several optimization models for the problem defined in Section 2.1. In Section 2.2.1, we develop a mixed integer linear programming model for the problem introduced in Section 2.1. However, for large instances of the problem, the model becomes difficult to solve. In Section 2.2.2, we introduce our two stage solution approach in an attempt to solve the problem in an efficient way.

2.2.1 Direct Approach

The Direct Approach is an attempt to solve the problem introduced in Section 2.1 using a mixed integer linear programming model developed in this section. The purpose of this model is to select the trucks that will be used and decide how to satisfy demands of all the demand points with the minimum transportation cost while respecting the truck capacities.

Despite the fact that all of the parameters used for this formulation are introduced in Section 2.1, we can summarize our parameters as follows:

M : Number of demand points
N : Number of different types of items
T_j : Set of trucks whose final destination is the demand point j, $j = 1, \ldots, M$
t_j : Number of trucks whose final destination is the demand point j, $j = 1, \ldots, M$
f_j : Final destination cost for each truck in T_j, $j = 1, \ldots, M$
$s_{i,j}$: Cost of visiting demand point i for each truck in T_j, $i = 1, \ldots, M, j = 1, \ldots, M$, $i \neq j$
$D_{i,l}$: The daily demand (in number of units) of demand point i for item type l,

$i = 1, \ldots, M$, $l = 1, \ldots, N$
w_l: Capacity of a truck that is occupied by one unit of item type l, $l = 1, \ldots, N$.

We next define the decision variables:

$$z_{j,k} = \begin{cases} 1 & \text{if truck } k \in T_j \text{ is used ;} \\ 0 & \text{otherwise,} \end{cases} \qquad j = 1, \ldots, M,\ k \in T_j.$$

$$x_{i,j,k} = \begin{cases} 1 & \text{if truck } k \in T_j \text{ carries load for the demand point } i; \\ 0 & \text{otherwise,} \end{cases}$$

$$i = 1, \ldots, M,\ j = 1, \ldots, M,\ k \in T_j.$$

$d_{i,j,k,l}$ = Demand (in number of units) of the demand point i for item type l that is satisfied by truck $k \in T_j$, $i = 1, \ldots, M$, $j = 1, \ldots, M$, $k \in T_j$, $l = 1, \ldots, N$.

Using the parameters and decision variables defined above, we formulate the following mixed integer programming model, which we refer to as (IP):

$$
\begin{aligned}
\text{(IP)} \qquad \min \quad & \sum_{j=1}^{M}\sum_{k\in T_j} f_j z_{j,k} + \sum_{i=1}^{M}\sum_{j=1}^{M}\sum_{k\in T_j} s_{i,j}x_{i,j,k} \\
\text{s.t.} \quad & \\
\sum_{i=1}^{M} x_{i,j,k} &\leq M z_{j,k}, \quad j=1,\ldots,M, k\in T_j, && (2.1)\\
\sum_{l=1}^{N} w_l d_{i,j,k,l} &\leq x_{i,j,k}, \quad i=1,...,M, j=1,...,M, k\in T_j && (2.2)\\
\sum_{i=1}^{M}\sum_{l=1}^{N} w_l d_{i,j,k,l} &\leq z_{j,k}, \quad j=1,\ldots,M, k\in T_j, && (2.3)\\
\sum_{j=1}^{M}\sum_{k\in T_j} d_{i,j,k,l} &= D_{i,l}, \quad i=1,\ldots,M, l=1,\ldots,N, && (2.4)\\
d_{i,j,k,l} \geq 0 \text{ and integer}, & \quad i=1,...,M, j=1,...,M, k\in T_j, l=1,..,N && (2.5)\\
x_{i,j,k} &\in \{0,1\}, \quad i=1,...,M, j=1,...,M, k\in T_j, && (2.6)\\
z_{j,k} &\in \{0,1\}, \quad j=1,\ldots,M, k\in T_j, && (2.7)
\end{aligned}
$$

The objective function is the total cost that arises from the cost of final destinations of used trucks and visiting cost for the demand points they will visit.

If a truck is not used, it cannot satisfy the demand of any demand point, which is ensured by the constraint (2.1). Similarly, if any part of the demand of a demand point is not allocated to a truck, then that truck cannot carry any item for that point. On the other hand, even if some part of the demand of a demand point is allocated to a truck, total volume of the items carried by the truck for the demand point cannot exceed the truck capacity. This is guaranteed by the constraint (2.2). The constraint (2.3) is the capacity constraint that should be satisfied for each truck. The constraint (2.4) ensures the satisfaction of the demand of each demand point for each item type.

The remaining constraints (2.5), (2.6) and (2.7) define the range of values that the decision variables can take.

(IP) can be used to solve small instances of the problem. However, when the problem instance gets larger, the model becomes increasingly difficult to solve since both the number of constraints and number of variables of (IP) is $O(M^2NT)$ where $T = \sum_{j=1}^{M} T_j$. The number of constraints of (IP) is $3MT + 2M^2T + MN + M^2NT$ and the number of variables of (IP) is $MT(M + MN + 1)$. In addition, all the variables are either binary variables or integer variables, which makes the problem more difficult to solve. There are $MT(M + 1)$ binary variables and M^2NT integer variables in this model.

Since the larger instances of the problem may not be solved by the Direct Approach, we developed a new solution approach, called the Hierarchical Approach. We explain this solution approach in the next subsection.

2.2.2 Hierarchical Approach

As the number of integer and binary variables in (IP) quickly increases with N, M and T, the model becomes difficult to solve for large instances of the problem.

We present the Hierarchical Approach in an attempt to solve larger instances of the problem in a more effective way. In an attempt to decrease the number of discrete variables and constraints, we first ignore the different type of items. We aggregate the demand of each demand point. We achieve this by interpreting the total demand of each demand point in terms of truckload so that $E_i = \displaystyle\sum_{l=1}^{N} w_l D_{i,l}$, for $i = 1, \ldots, M$. Next, we solve a problem to satisfy the demand (in truckload) of each demand point with the minimum transportation cost while respecting truck capacities. This is the first stage. The important difference between the first stage and the Direct Approach is the ignorance of the different type of items in the first stage. The new aggregate items are assumed to be divisible and the integrality is therefore ignored in the first stage. This yields a decrease in the number of variables and the number of constraints by a factor of O(N). In addition, as the demand in truckload can be allocated to trucks in any proportion, there is no integrality restriction on the allocated demands of each demand point in the first stage. These results make the first stage problem easier to solve in comparison with the Direct Approach. Therefore, we solve a relatively easier and smaller problem in the first stage.

The solution from the first stage gives us information about two crucial points. The first one is the trucks that will be used in the distribution. The second one is the demand points that each truck will visit. According to this information, in the second stage, we solve the problem of satisfying the demand of each demand point for each item type with the trucks that will visit it. Note that this problem decomposes naturally. Demand of a demand point can be satisfied by a finite number of trucks. Similarly, a truck can satisfy the demand of a finite number of demand points. In addition, as the demand points are geographically dispersed, there may be demand points that are not related to each other. Therefore, we divide the problem into subproblems after the first stage solution is obtained. This is called the clustering stage. A cluster consists of all the trucks that will visit at least one of the demand points of the cluster. In addition, all the demand points that are visited by at least one of the trucks of the cluster also belong to the same cluster with the trucks. Therefore, solution of one subproblem does not depend on another one. The subproblem is to satisfy the demand of each demand point in the cluster by the trucks that will visit it within the same cluster. At the end of the clustering stage, we have several subproblems to solve and in the second stage, we solve each subproblem separately. In the second stage, we disaggregate the demand of each demand point. We do not ignore different type of items. We solve the problem with respect to the solution of the first stage. At the end of the second stage, we determine how to distribute the items into trucks so that the demand of each demand point is satisfied.

The main idea of the Hierarchical Approach is the demand aggregation and disaggregation. Demand aggregation makes the first stage problem easier to solve as explained above. However, the demand aggregation also has a drawback. We ignore the integrality of the items while considering the demand in truckload. Therefore, we may allocate the aggregated demand in truckload into trucks in such a way that it may not be possible to allocate the same demand for items into the same number of trucks. Consider the following example:

Example 2.1: Suppose that there are two demand points, Point 1 and Point 2 and there are two item types, Item 1 and Item 2. One unit of Item 1 occupies 0.2 truckload and one unit of Item 2 occupies 0.3 truckload. The demand of each demand point for each item type is given in the following table:

$D_{i,l}$	Item 1	Item 2
Point 1	1	3
Point 2	0	3

The aggregated demand of Point 1 is $E_{Point1} = 1.1$ truckload and the aggregated demand of Point 2 is $E_{Point2} = 0.9$ truckload. This aggregated demand can be allocated to two trucks. One truckload of the demand of Point 1 can be allocated to Truck 1, and the rest of the demand of Point 1 and all of the demand of Point 2 can be allocated to Truck 2. However, it is not possible to allocate demand of the demand points for each item type into two trucks after disaggregating the demand.

As shown in Example 2.1, demand aggregation may lead us to a demand allocation which is not possible for the disaggregated demand. This is the disadvantage of the demand aggregation. Therefore, in the second stage, we allow excess capacity usage in the trucks, but we penalize it in the objective function.

In the following subsections, we explain the stages of the Hierarchical Approach in detail.

2.2.2.1 First Stage

The main purpose of the first stage is to determine the trucks that will be used in order to satisfy the demand of the demand points with the minimum transportation cost. When the model is solved to optimality, trucks that will be used and capacities that should be allocated on each truck to each demand point they will visit are determined with minimum total cost.

As in the Direct Approach, we initially assume that there are $T = \sum_{i=1}^{M} t_i$ trucks available. Next, we compute the total demand of each demand point in terms of truckload,i.e., we let $E_i = \sum_{l=1}^{N} w_l D_{i,l}$ for $i = 1, \ldots, M$. In the first stage, we try to satisfy the demand of each demand point with minimum cost. In other words, we ignore different types of items and we only allocate total demand E_i of each demand point $i = 1, \ldots, M$ into trucks. As the total demand E_i can be allocated into trucks in any proportion, we also ignore the integrality of items in the first stage. Consequently, from the solution of the first stage we only obtain the set of trucks that will be used and the capacities that should be reserved for each demand point in each truck.

The parameters of the first stage are defined as follows:

M : Number of demand points
N : Number of different types of items
T : Total number of trucks
T_j : Set of trucks whose final destination is the demand point j, $j = 1, \ldots, M$
t_j : Number of trucks whose final destination is the demand point j, $j = 1, \ldots, M$
f_j : Final destination cost for each truck in T_j, $j = 1, \ldots, M$
$s_{i,j}$: Cost of visiting the demand point i for each truck in T_j, $i = 1, \ldots, M, j = 1, \ldots, M$
E_i : Demand (in truckload) of the demand point i, $i = 1, \ldots, M$
w_l : Capacity of a truck that is occupied by a unit of item type l, $l = 1, \ldots, N$

Variables of the first stage are defined as follows:

$$z_{j,k} = \begin{cases} 1 & \text{if truck } k \in T_j \text{ is used;} \\ 0 & \text{otherwise,} \end{cases} \qquad j = 1, \ldots, M,\ k \in T_j.$$

$$x_{i,j,k} = \begin{cases} 1 & \text{if truck } k \in T_j \text{ carries load for the demand point } i; \\ 0 & \text{otherwise,} \end{cases}$$

$$i = 1, \ldots, M,\ j = 1, \ldots, M,\ k \in T_j.$$

$y_{i,j,k}$ = Demand of the demand point i (in truckload) that is satisfied by truck $k \in T_j$, $i = 1, \ldots, M, j = 1, \ldots, M$, $k \in T_j$.

For the first stage of the Hierarchical Approach we derive the following model which we refer to as the First Stage Model (FSM):

$$
\begin{aligned}
\text{(FSM)} \qquad & \min \quad \sum_{j=1}^{M} \sum_{k \in T_j} f_j z_{j,k} + \sum_{i=1}^{M} \sum_{j=1}^{M} \sum_{k \in T_j} s_{i,j} x_{i,j,k} \\
& \text{s.t.}
\end{aligned}
$$

$$
\begin{aligned}
\sum_{i=1}^{M} x_{i,j,k} &\leq M z_{j,k}, \quad j = 1, \ldots, M, k \in T_j && (2.8)\\
y_{i,j,k} &\leq x_{i,j,k}, i = 1, \ldots, M, j = 1, \ldots, M, k \in T_j && (2.9)\\
\sum_{i=1}^{M} y_{i,j,k} &\leq z_{j,k}, \quad j = 1, \ldots, M, k \in T_j && (2.10)\\
\sum_{j=1}^{M} \sum_{k \in T_j} y_{i,j,k} &= E_i, \quad i = 1, \ldots, M, && (2.11)\\
y_{i,j,k} \geq 0 & \qquad i = 1, \ldots, M, j = 1, \ldots, M, k \in T_j, && (2.12)\\
x_{i,j,k} \in \{0, 1\}, & \qquad i = 1, \ldots, M, j = 1, \ldots, M, k \in T_j, && (2.13)\\
z_{j,k} \in \{0, 1\}, & \qquad j = 1, \ldots, M, k \in T_j. && (2.14)
\end{aligned}
$$

The objective function is the total cost that arises from the cost of final destinations of the trucks used and visiting cost of the demand points they will visit.

If a truck is not used, then it cannot carry any load for any of the demand point. Similarly, if any part of the demand of a demand point is not allocated to a truck, then that truck cannot carry any load for that point. These are ensured by the constraints (2.8), (2.9), respectively. Total volume of the load carried by a truck cannot exceed the truck capacity. This is guaranteed by the constraints (2.10). The constraints (2.11) serve for the demand satisfaction of all the demand points.

The remaining constraints (2.12), (2.13) and (2.14) define the range of values the decision variables can take.

2.2.2.2 Intermediate Stage: Clustering

In the first stage, we solve the first stage model for the aggregated demand of each demand point. At the end of the first stage, the trucks that will be used and the demand points they will visit are determined. The allocated capacities on each truck for each demand point are also decided. However, items demanded by the demand points are not allocated to the trucks. Therefore, we need to disaggregate the aggregated demand of each demand point in terms of different items. In the second stage, we disaggregate the demand respecting the solution from the first stage. However, this problem naturally decomposes into smaller problems. In this stage, we determine these subproblems.

We construct a graph with respect to the solution of the first stage. Consider a bipartite graph in which vertices represent trucks and demand points. If a truck visits a demand point, then there exists an edge between the vertices that represent that truck and that demand point. This graph is a bipartite graph as there are no edge between any two trucks or any two demand points. For example, suppose that the first stage reveals that n trucks will be used in order to satisfy the demand of m demand points. Consider the following graph in Figure 2.1. According to this graph, Truck 1 and Truck 2 carry load for Demand Point 1 and Truck 3 satisfies the demand of Demand Point 2 and Demand Point 3.

In the resulting graph, we find out connected components, i.e., we divide the graph into clusters or subgraphs such that there exist no edge between any pair of the clusters. For example, according to the graph in Figure 2.1, Truck 1, Truck 2 and Point 1 form the first cluster. Point 2, Point 3 and Truck 3 form the second cluster and so on.

Note that a demand point appears only in one cluster and similarly a truck belongs only to one cluster. In addition, trucks that will carry load for a demand point are in the same cluster with the demand point and the demand points

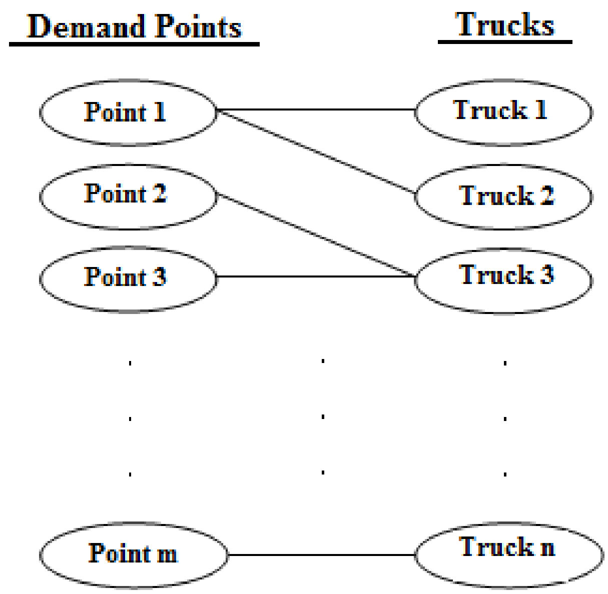

Figure 2.1: The Bipartite Graph

that are visited by the same truck are also in the same cluster with the truck. Satisfying the demand of all the demand points in a cluster with the trucks that will visit them is a subproblem for each cluster. Solution of a subproblem does not depend on the solution of another one.

We solve the second stage for each cluster individually. For our example, the first subproblem is to satisfy the demand of Point 1 using only Truck 1 and Truck 2. The second subproblem is to satisfy the demand of the demand points Point 2 and Point 3 using Truck 3. We continue until there is no cluster left.

After the first stage solution is obtained, we find the clusters in the clustering stage and then solve the second stage for each cluster separately. In the next section, we explain the second stage of the solution approach.

2.2.2.3 Second Stage

After the clustering stage is completed, we solve the second stage for each cluster separately as the corresponding subproblems can be solved independently.

In the second stage, we solve the subproblems for each cluster. A subproblem for a specific cluster is to satisfy the demand of the demand points in the cluster for each item type using only the trucks that will visit them. At the end of this stage, we find how to satisfy the demand of each demand point for each item type. However, as we ignore the integrality of items in the first stage, it may not be possible to find a solution to the second stage that respects the first stage solution (see Example 2.1). Therefore, in the second stage, our objective is to find a feasible solution respecting the solution from the first stage as closely as possible. In order to achieve this objective, we develop two different models which are called the Second Stage Model-1 (SSM-1) and Second Stage Model-2 (SSM-2).

As the solution of the first stage becomes an input for the second stage, notations that we use for some of the variables of the first stage and parameters of the second stage will be the same. The parameters of the second stage are as follows:

M' : Number of demand points in the cluster
T' : Number of trucks in the cluster
R' : Set of demand points in the cluster; $|R'| = M'$,
T'_j : Set of trucks whose final destination is the demand point j, $j = 1, \ldots, M'$
t'_j : Number of trucks whose final destination is the demand point j, $j = 1, \ldots, M'$
N : Number of different types of items
w_l : Capacity of a truck that is occupied by a unit of item type l, $l = 1, \ldots, N$
$D_{i,l}$: Demand of demand point i for item type l, $i = 1, \ldots, M'$, $l = 1, \ldots, N$

The following are the optimal solutions returned by (FSM) and used in the second stage:

$$x^*_{i,j,k} = \begin{cases} 1 & \text{if truck } k \in T'_j \text{ carries load for the demand point } i; \\ 0 & \text{otherwise,} \end{cases}$$

$$i \in R',\ j \in R', k \in T'_j.$$

$y^*_{i,j,k}$: Demand (in truckload) of demand point i that will be satisfied by truck $k \in T'_j$, for $i \in R'$, $j \in R'$, $k \in T'_j$.

In (SSM-1), we find a solution such that the deviation from the solution of the first stage is minimized. In the first stage, we allocate the demand (in truckload) of the demand points into the used trucks. In (SSM-1), we try to find a solution that is as close as possible to the first stage solution.

The variables of (SSM-1) are as follows:

$d_{i,j,k,l}$: Number of units of item type l in truck $k \in T'_j$ for the demand point i, $i \in R'$, $j \in R'$, $k \in T'_j$, $l = 1, \ldots, N$.

$p_{i,j,k}$: Capacity used more than the allocated capacity for the demand point i in truck $k \in T'_j$, $i \in R'$, $j \in R'$, $k \in T'_j$.

$q_{i,j,k}$: Capacity used less than the allocated capacity for the demand point i in truck $k \in T'_j$, $i \in R'$, $j \in R'$, $k \in T'_j$.

$c_{j,k}$: Additional capacity used in truck $k \in T'_j$, $j \in R'$, $k \in T'_j$.

(SSM-1) is formulated as the following:

$$
\begin{aligned}
&\text{(SSM-1)} \qquad \min \quad \sum_{j\in R'}\sum_{k\in T'_j}\left(\sum_{i\in R'}(p_{i,j,k}+q_{i,j,k})+c_{j,k}\right)\\
&\text{s.t.}
\end{aligned}
$$

$$
\begin{aligned}
\sum_{j\in R'}\sum_{k\in T'_j} x^*_{i,j,k}d_{i,j,k,l} &= D_{i,l}, \quad i\in R', \quad l=1,..,N, && (2.15)\\
\sum_{i\in R'}\sum_{l=1}^{N} w_l d_{i,j,k,l} &\le 1+c_{j,k} \quad j\in R', \quad k\in T'_j && (2.16)\\
\sum_{l=1}^{N} w_l d_{i,j,k,l} &\le y^*_{i,j,k}+p_{i,j,k}, \quad i,j\in R', \quad k\in T'_j && (2.17)\\
\sum_{l=1}^{N} w_l d_{i,j,k,l} &\ge y^*_{i,j,k}-q_{i,j,k}, \quad i,j\in R', \quad k\in T'_j && (2.18)\\
d_{i,j,k,l}\ge 0 \text{ and integer}, &\quad i,j\in R', \quad k\in T'_j, \quad l=1,..,N, && (2.19)\\
p_{i,j,k}\ge 0, &\quad i,j\in R', \quad k\in T'_j, && (2.20)\\
q_{i,j,k}\ge 0, &\quad i,j\in R', \quad k\in T'_j, && (2.21)\\
c_{j,k}\ge 0, &\quad j\in R', \quad k\in T'_j. && (2.22)
\end{aligned}
$$

The constraint (2.15) ensures the demand satisfaction of each demand point for each item type. The constraint (2.16) finds the excess in capacity usage, $c_{j,k}$, in each truck $k \in T'_j$ for $j = 1, \ldots, M'$ in order to satisfy the demand of each demand point. The constraints (2.17) and (2.18) find the positive and negative deviation from the allocated capacity $p_{i,j,k}$ and $q_{i,j,k}$ in truck $k \in T_j$, $j = 1, \ldots, M'$ for each demand point $i = 1, \ldots, M'$, respectively.

The remaining constraints (2.19), (2.20), (2.21) and (2.22) define the ranges of values that each decision variable can take.

We define the excess capacity usage in trucks using the variables $c_{j,k} \ge 0$, $j \in R', k \in T'_j$ and the constraint (2.16). We need to do this as it may not be possible to satisfy the demand of each demand point with the trucks that will visit them (see Example 2.1). Therefore, we allow the excess capacity usage in the trucks, however we penalize the excess capacity in the objective. Since it is a minimization problem, the model tries to make the excess usage as small as possible.

If a solution that is consistent with the first stage solution is found, then

$$\sum_{j\in R'}\sum_{k\in T'_j}\left(\sum_{i\in R'}(p_{i,j,k}+q_{i,j,k})+c_{j,k}\right)=0.$$

This means that the actual capacity in each truck that is allocated to each demand point is exactly the same as the first stage solution. However, this may not always be possible since the integrality restriction is ignored in the first stage. The objective function is chosen to ensure smallest deviation from the first stage solution.

We propose another model for the second stage, called the Second Stage Model-2 (SSM-2). In (SSM-2), we only minimize the total infeasibility, total excess in the capacities of trucks used without taking into account the allocated capacities in each truck for each demand point from the first stage solution.

(SSM-2) is formulated as follows:

$$\text{(SSM-2)} \qquad \min \quad \sum_{j\in R'}\sum_{k\in T'_j} c_{j,k}$$

$$\begin{aligned}
&\text{s.t.} && && \\
\sum_{j\in R'}\sum_{k\in T'_j} x^*_{i,j,k} d_{i,j,k,l} &= D_{i,l}, && i\in R', \quad l=1,\ldots,N, && (2.23)\\
\sum_{i\in R'}\sum_{l=1}^{N} w_l d_{i,j,k,l} &\le 1+c_{j,k} && j\in R', \quad k\in T'_j && (2.24)\\
d_{i,j,k,l}\ge 0 \text{ and integer}, & && i,j\in R', k\in T'_j, \quad l=1,\ldots,N, && (2.25)\\
c_{j,k} &\ge 0, && j\in R', \quad k\in T'_j. && (2.26)
\end{aligned}$$

The constraint (2.23) ensures the demand satisfaction of each demand point for each item type. The constraint (2.24) finds the excess in capacity usage, $c_{j,k}$, in each truck $k\in T'_j$ for $j=1,\ldots,M'$ in order to satisfy the demand of each demand point.

The remaining constraints (2.25) and (2.26) define the range of values that each decision variable can take.

The main purpose of the second stage is to find a feasible solution respecting the solution of the first stage as closely as possible. However, we may ignore the capacity allocations in each truck. There may be another combination of the allocated capacities in the trucks without changing the total transportation cost while satisfying the demand of all the demand points. Moreover, any solution to (SSM-1) is also a solution for (SSM-2). If it is best to stick to the solution of the first stage, (SSM-2) also does it since the first stage solution is also a feasible solution for (SSM-2). Therefore, we ignore the constraints (2.17) and (2.18) of (SSM-1) in (SSM-2). Our main purpose in (SSM-2) is to find a feasible solution such that no truck is used more than its capacity. Therefore, we again allow the excess capacity usage in the trucks by the same reason explained before and we now minimize only the total excess capacity used in the trucks. If it is possible to satisfy the demand of each demand point by the trucks with no excess capacity, (SSM-2) finds a solution with an objective function value of zero. If it is not possible, then (SSM-2) finds a solution with excess capacity in trucks as small as possible.

We develop (SSM-2) since it is easier to solve than (SSM-1) since it has M'^2T' fewer constraints and variables than (SSM-1). In addition, the set of feasible points of (SSM-1) is a subset of the set of feasible points of (SSM-2).

When the Hierarchical Approach finds a solution to the problem introduced at the beginning of the chapter, it does it in a significantly shorter time than the Direct Approach. In addition, it may also solve the problem instances that cannot be solved by the Direct Approach. However, by the Hierarchical Approach we have a risk to find a solution that is infeasible for the original problem since the excess capacity usage in the trucks is allowed in the Hierarchical Approach. If there exists $k \in T'_j$, $j = 1, \ldots, M'$, such that $c_{j,k} > 0$ in any of the subproblems, then the solution of the Hierarchical Approach is not feasible for the original problem. In the next chapter, we analyze the solution of this approach and find an upper bound for $c_{j,k}$, $j = 1, \ldots, M'$, $k \in T'_j$. Our goal is to find an upper bound on the measure of infeasibility of the solution we obtain from the second stage.

Chapter 3

Analysis of The Optimization Models

In the previous chapter, we introduce the Hierarchical Approach and the Direct Approach. In this chapter, we compare these two different solution approaches. As discussed in the previous chapter, the Hierarchical Approach may give a solution that is not feasible for the original problem. In this chapter, we give the conditions in order for the solution of the Hierarchical Approach to be an optimal solution of the original problem and then we analyze the quality of solutions of the Hierarchical Approach. We find an upper bound on the the maximum infeasibility residual for the solution of the Hierarchical Approach.

In the second stage of the Hierarchical Approach, we solve each subproblem separately. At the end of the second stage, we obtain a solution for each subproblem. The solution of each subproblem gives us information about how to satisfy the demand of the demand points in the corresponding cluster. Consequently, a solution of the original problem introduced in Chapter 2 is given by the combination of the solutions of each subproblem. We call the combination of the solutions of each subproblem of the second stage "the solution of the Hierarchical Approach" in the rest of the thesis.

We next discuss several properties of the Hierarchical Approach.

Lemma 1 *The optimal value of (FSM) is less than or equal to the optimal value of (IP).*

Proof. It suffices to show that any optimal solution of (IP) is a feasible solution of the first stage of the Hierarchical Approach. Let $\overline{d}_{i,j,k,l}$ be an optimal solution of (IP). Then we can compute $\overline{y}_{i,j,k} = \sum_{l=1}^{N} w_l \overline{d}_{i,j,k,l}$ for each $i = 1, \ldots, M$, $j = 1, \ldots, M$, $k \in T_j$. Since

$$\sum_{i=1}^{M} \sum_{l=1}^{N} w_l \overline{d}_{i,j,k,l} \leq 1$$

by (2.3) for each $j = 1, \ldots, M$, $k \in T_j$, we have

$$\sum_{i=1}^{M} \overline{y}_{i,j,k} \leq 1$$

for each $j = 1, \ldots, M$, $k \in T_j$. In addition, if we multiply both sides of the equation

$$\sum_{j=1}^{M} \sum_{k \in T_j} \overline{d}_{i,j,k,l} = D_{i,l}$$

by w_l and sum up both sides for all $l = 1, \ldots, N$, we obtain

$$\sum_{j=1}^{M} \sum_{k \in T_j} \overline{y}_{i,j,k} = E_i$$

for all $i = 1, \ldots, M$. This shows that any optimal solution of (IP) can be transformed into a feasible solution for the first stage of the Hierarchical Approach. Therefore, any optimal solution of the Hierarchical Approach has a cost which is less than or equal to the cost of solution of (IP) since both problems have the same objective function value. $\square$

As indicated in Chapter 2, it is easier to solve (SSM-2) than solving (SSM-1). Therefore, we henceforth assume that we use (SSM-2) in the second stage of the Hierarchical Approach.

In the next proposition, we present a sufficient condition in order for the solution of the Hierarchical Approach to be an optimal solution for (IP).

Proposition 1 *If the optimal objective function value of the second stage of the Hierarchical Approach is equal to zero for each cluster, then any optimal solution of the Hierarchical Approach is also an optimal solution of the original problem.*

Proof. If $c_{j,k} = 0$ for all $j = 1, \ldots, M$, $k \in T_j$, then all trucks are used without any excess in their capacities. Therefore this solution is also a feasible solution for (IP). By Lemma 1, this solution is an optimal solution for (IP). $\square$

Proposition 1 gives a sufficient condition for the optimality of the solution of the Hierarchical Approach. However, this condition may not be satisfied in general. As we allow the excess capacity usage in trucks in the second stage of the Hierarchical Approach, it may find a solution in which there exists at least one truck that is used more than its capacity. Therefore, the solution of the Hierarchical Approach may be infeasible for the original problem.

Next, we find an upper bound on the possible excess capacity usage for a truck in an optimal solution of the Hierarchical Approach.

Consider a cluster in which there are M' demand points $\{R_1, \ldots, R_{M'}\}$ and T' trucks $\{T_1, \ldots, T_{T'}\}$. Each truck has a capacity of one truckload.

Since there are T' trucks, the total capacity of all trucks in the cluster is T'. We can assume that the total demand of all demand points in the cluster is equal to the total capacity of all trucks in the cluster. If this is not the case, we can add a dummy demand point with demand (T' - total demand of all demand points) and satisfy this assumption.

Let

d_j^+ : positive deviation from the truck capacity of one truckload for truck j

d_j^- : negative deviation from the truck capacity of one truckload for truck j

d_j : net deviation from the truck capacity of one truckload for truck j, $j = 1, \ldots, T'$.

Then,

$$\sum_{j=1}^{T'} d_j = 0 \tag{3.1}$$

where

$$d_j = d_j^+ - d_j^-, \tag{3.2}$$

$$\min\left\{d_j^+, d_j^-\right\} = 0. \tag{3.3}$$

The objective of the second stage of the Hierarchical Approach is to minimize the total excess capacity used in the trucks, i.e.,

$$\min \sum_{j=1}^{T'} d_j^+. \tag{3.4}$$

Let $w = \max_{l \in B'} w_l$ where

B' : set of items in the cluster; $B' \subset \{1, \ldots, N\}$,

w_l : Truck capacity (in terms of truckload) occupied by one unit of item type l, $l \in B'$.

The next lemma establishes an upper bound on the largest net deviation in any truck for an optimal solution of (SSM-2).

Lemma 2 *We have*

$$\max_{j=1,\ldots,T'} d_j \leq w. \tag{3.5}$$

Proof. If $T' = 1$, by the equation (3.1), $d_1 = 0$ and (3.5) is satisfied. Let $T' > 1$. Suppose to the contrary that

$$\max_{j=1,\ldots,T'} d_j > w. \tag{3.6}$$

Assume without loss of generality that $d_1 > w$. Then, it follows from the equation (3.1) that

$$\sum_{j=2}^{T'} d_j = -d_1 < -w \tag{3.7}$$

and

$$\min_{j=2,\ldots,T'} d_j \leq \frac{\sum_{j=2}^{T'} d_j}{T'-1} < \frac{-w}{T'-1} \tag{3.8}$$

since the minimum of a finite set of real numbers is less than or equal to the average of the set. Let $\arg\min_{j=2,\ldots,T'} d_j = k$. Note that $k \neq 1$.

Consider the subgraph in which demand points and trucks are represented by nodes and edges connect demand points and trucks if a truck satisfies all or some proportion of the demand of a demand point (see Figure 2.1 for an example). This subgraph is a bipartite graph as there are no edges between any two trucks or any two demand points. We claim that we can construct a feasible solution of (SSM-2) with a smaller objective function value. We achieve this by redistributing items among trucks.

There exists a path between any two trucks in this graph and such a path includes at least one demand point. In other words, we can find a path from T_1 to T_j, for any $j \in \{1, \ldots, T'\}$. Let the path from T_1 to T_k be $P = \{T_1, R_1, T_2, R_2, \ldots, T_{k-1}, R_{k-1}, T_k\}$. We can take an item, say item l_1, that is ordered by R_1 and reload it to the next truck in the path after R_1 which is T_2. As $\{T_1, R_1, T_2\} \subset P$, R_1 is connected to both of T_1 and T_2. This implies that both of the trucks carry items for R_1. Therefore, by taking one item from T_1 and reloading it to T_2 we do not change the total cost. By this process, we may change the loads of each truck but do not affect the demand satisfaction of the demand points. Then, we take an item, say item l_2 from T_2 which is ordered by R_2, and reload it to the next truck T_3 in the path. We repeat this process until we find a feasible solution for (SSM-2) with a smaller objective function value.

Let Δd_j^+ be the net change in the objective function value corresponding to the truck T_j for $j = 1, \dots, T'$.

Since $d_1 > w$, by our assumption, we still have $d_1 > 0$ after we remove an item from T_1. Therefore

$$\Delta d_1^+ = -w_{l_1}. \tag{3.9}$$

When we add item l_1 to T_2, we may observe the following different cases. We treat each case separately.

- **Case I,** if $d_2 < 0$ and $w_{l_1} + d_2 < 0$, then

  $$d_2^+ = \Delta d_2^+ = 0 \tag{3.10}$$

  and then

  $$\sum_{i=1}^{T'} \Delta d_i^+ = \Delta d_1^+ + \Delta d_2^+ + 0 = -w_{l_1} + 0 = -w_{l_1} < 0, \tag{3.11}$$

  $$\sum_{i=1}^{T'} (d_i^+ + \Delta d_i^+) < \sum_{i=1}^{T'} d_i^+. \tag{3.12}$$

  We obtain a solution which has a better objective function value, which is a contradiction.

- **Case II,** if $d_2 < 0$ and $w_{l_1} + d_2 \geq 0$, then

  $$d_2^+ = \Delta d_2^+ = w_{l_1} + d_2 < w_{l_1} \leq w, \tag{3.13}$$

  and then

  $$\sum_{i=1}^{T'} \Delta d_i^+ = \Delta d_1^+ + \Delta d_2^+ + 0 < -w_{l_1} + w_{l_1} = 0, \tag{3.14}$$

  $$\sum_{i=1}^{T'} (d_i^+ + \Delta d_i^+) < \sum_{i=1}^{T'} d_i^+. \tag{3.15}$$

We obtain a solution which has a better objective function value, which is a contradiction.

- **Case III,** if $d_2 \geq 0$, then

$$\Delta d_2^+ = w_{l_1}, \tag{3.16}$$

$$d_2^+ \geq w_{l_1}, \tag{3.17}$$

and then

$$\sum_{i=1}^{T'} \Delta d_i^+ = \Delta d_1^+ + \Delta d_2^+ + 0 = -w_{l_1} + w_{l_1} = 0, \tag{3.18}$$

$$\sum_{i=1}^{T'} (d_i^+ + \Delta d_i^+) = \sum_{i=1}^{T'} d_i^+. \tag{3.19}$$

If one of the first two cases happens, then we reach our goal and obtain a contradiction. If the third case is realized, then we remove one item, item w_{l_2} from T_2 and reload it to the next truck in the path, T_3. Then,

$$\Delta d_2^+ \leq w_{l_1} - w_{l_2}. \tag{3.20}$$

For T_3, we repeat the same analysis with T_2. If one of the first two cases is realized, then we reach our goal. Otherwise, we continue the process, which eventually ends when we remove item $w_{l_{k-1}}$ from T_{k-1} and reload it to T_k. As

$$d_k < \frac{-w}{T' - 1}, \tag{3.21}$$

$$d_k^+ = 0, \tag{3.22}$$

and when we load item $w_{l_{k-1}}$ to T_k, we have

$$\Delta d_k^+ < w_{l_{k-1}} - \frac{w}{T' - 1}. \tag{3.23}$$

In addition, in this case, the third case should have been realized for all the trucks $T_j, j \in \{2, \ldots, k-1\}$ since otherwise the process would have stopped earlier. Then,

$$\begin{array}{rcl}
\Delta d_1^+ & = & -w_{l_1}, \\
\Delta d_2^+ & \leq & w_{l_1} - w_{l_2}, \\
\Delta d_3^+ & \leq & w_{l_2} - w_{l_3}, \\
& . & \\
& . & \\
& . & \\
\Delta d_k^+ & < & w_{l_{k-1}} - \dfrac{w}{T'-1}.
\end{array}$$

When we sum up all the changes, we obtain

$$\sum_{i=1}^{T'} \Delta d_i^+ < \frac{-w}{T'-1} < 0, \tag{3.24}$$

$$\sum_{i=1}^{T'} (d_i^+ + \Delta d_i^+) < \sum_{i=1}^{T'} d_i^+. \tag{3.25}$$

By this process, we obtain a feasible solution which has a better objective function value. If there still exists a truck with $d_j > w$, $j \in \{1, \ldots, T'\}$ in any cluster of the second stage, we can repeat the same. This is a contradiction. Therefore,

$$\max_{j=1,\ldots,T'} d_j \leq w. \tag{3.26}$$

$\square$

In the next theorem, we show that we can further improve the upper bound in Lemma 2.

Theorem 1 *Consider a cluster in which there are M' demand points $\{R_1, \ldots, R_{M'}\}$ and T' trucks $\{T_1, \ldots, T_{T'}\}$. Each truck has a capacity of one truckload.*

Let $w = \max_{l \in B'} w_l$ where

B' : set of items in the cluster; $B' \subset \{1, \ldots, N\}$,

w_l : Truck capacity (in terms of truckload) occupied by one unit of item type $l \in B'$.

Then,

$$\max_{i=1,\ldots,T'} d_i \leq w \left(1 - \frac{1}{T'}\right), \tag{3.27}$$

where

d_j *: net deviation from the truck capacity of one truckload for truck* T_j, $j = 1, \ldots, T'$.

Proof. We assume that the total demand of all the demand points is equal to T'. If this is not the case, then we can satisfy this condition by defining a dummy demand point as in the previous proof.

Let

d_j^+ : positive deviation from the truck capacity of one truckload for truck T_j

d_j^- : negative deviation from the truck capacity of one truckload for truck T_j

Δd_j^+ : net change in the objective function value caused by truck T_j

for $j = 1, \ldots, T'$.

If $T' = 1$, by the equation (3.1), $d_1 = d_1^+ = 0$ and (3.27) is satisfied.

If $T' \geq 2$, we again proceed by contradiction. Suppose, without loss of generality, that

$$d_1 = d_1^+ > w \left(1 - \frac{1}{T'}\right) > 0. \tag{3.28}$$

Then by (3.1),

$$\sum_{j=2}^{T'} d_j = -d_1 = -d_1^+ < -w \left(1 - \frac{1}{T'}\right) = -w\frac{T'-1}{T'} < 0, \tag{3.29}$$

and

$$\min_{j=2,\ldots,T'} d_j < -\frac{w}{T'} < 0, \tag{3.30}$$

since the minimum of a finite set of real numbers is less than or equal to the average of the set. Let $\arg\min_{j=2,\ldots,T'} d_j = k$. Note that $k \neq 1$.

We will use the same procedure as in the previous proof. We remove l_1 from T_1, reload it to T_2; take l_2 from T_2 and reload it to T_3 and do this until we obtain a solution that has a better objective function value. In contrast, this time there are two different cases for T_1:

Case 1: $w_{l_1} \leq d_1^+$

This case is equivalent to the previous proof since in the previous proof, it is assumed that

$$w < d_1^+ \tag{3.31}$$

and as

$$w = \max_{l \in B'} w_l \geq w_{l_1}, \tag{3.32}$$

in the previous proof

$$w_{l_1} \leq d_1^+. \tag{3.33}$$

Then, we achieve our goal either in a truck $T_j, j \in \{2, \ldots, k-1\}$ or when we reach T_k. The cases where we achieve our goal before T_k are the same as the previous proof. If we achieve our goal in T_k, then

$$\begin{array}{rcl} \Delta d_1^+ & = & -w_{l_1}, \\ \Delta d_2^+ & \leq & w_{l_1} - w_{l_2}, \\ & . & \\ & . & \\ & . & \\ \Delta d_k^+ & \leq & w_{k-1} + d_k < w_{k-1} - \dfrac{w}{T'}. \end{array}$$

When we sum up Δd_i^+ values over all $i = 1, \ldots, k$ values, we obtain

$$\sum_{i=1}^{T'} \Delta d_i^+ < -\frac{w}{T'} < 0, \tag{3.34}$$

$$\sum_{i=1}^{T'} (d_i^+ + \Delta d_i^+) < \sum_{i=1}^{T'} d_i^+. \tag{3.35}$$

This is a contradiction as we find a better solution.

Case 2: $w_{l_1} > d_1^+ > w(1 - \frac{1}{T'})$

For this case, $\Delta d_1^+ = -d_1^+ < -w_{l_1}$. If one of the first two cases of the previous proof is realized in the truck T_j, $j \in \{2, \ldots, k-1\}$, then

$$\begin{array}{rcl} \Delta d_1^+ & < & -w_{l_1}, \\ \Delta d_2^+ & \leq & w_{l_1} - w_{l_2}, \\ & . & \\ & . & \\ & . & \\ \Delta d_j^+ & \leq & w_{j-1} + d_j. \end{array}$$

When we sum up Δd_i^+ over all $i = 1, \ldots, j$ values, we obtain

$$\sum_{i=1}^{T'} \Delta d_i^+ < d_j < 0, \tag{3.36}$$

$$\sum_{i=1}^{T'} (d_i^+ + \Delta d_i^+) < \sum_{i=1}^{T'} d_i^+. \tag{3.37}$$

This is a contradiction as we find a better solution.

If we reach the truck T_k, then

$$\begin{array}{rcl} \Delta d_1^+ & < & -w_{l_1}, \\ \Delta d_2^+ & \leq & w_{l_1} - w_{l_2}, \\ & . & \\ & . & \\ & . & \\ \Delta d_k^+ & \leq & w_{k-1} + d_k < w_{k-1} - \dfrac{w}{T'}. \end{array}$$

When we sum up Δd_i^+ over all $i = 1, \ldots, k$ values, we obtain

$$\sum_{i=1}^{T'} \Delta d_i^+ < -\frac{w}{T'} < 0, \tag{3.38}$$

$$\sum_{i=1}^{T'} (d_i^+ + \Delta d_i^+) < \sum_{i=1}^{T'} d_i^+. \tag{3.39}$$

Since we obtain a feasible solution with a smaller objective function value, this is a contradiction. □

Remark: The upper bound $w\left(1-\frac{1}{T'}\right)$ for $\max_{j=1,\ldots,T'} d_j$ is tight as illustrated by the following example.

Example 3.1: Assume that there is only one type of item, $B' = \{1\}$, which occupies $w = w_1 = \frac{2}{3}$ of the truck capacity. There is one demand point, $M' = 1$ and it has a demand of three units of the item, $D_{1,1} = 3$. As the total demand is two truckloads, the Hierarchical Approach finds a solution in which two trucks visit the demand point. Therefore, $T' = 2$. However, according to the second stage of the solution approach,

$$d_1 = \tfrac{1}{3} = \tfrac{2}{3}\left(1-\tfrac{1}{2}\right) = w\left(1-\tfrac{1}{T'}\right) \text{ and}$$
$$d_2 = -\tfrac{1}{3}.$$

Therefore, the upper bound $w\left(1-\frac{1}{T'}\right)$ cannot be improved.

Consequently, despite the fact that the Hierarchical Approach may give a solution that is not feasible for the original problem, there is a tight upper bound on the the maximum infeasibility residual for the solution of the Hierarchical Approach.

We can take advantage of this upper bound for obtaining a feasible solution for the original problem. Suppose that, in the first stage of the Hierarchical Approach, we assume that all the trucks have a capacity of $(1-w)$ truckload. Then, we need to replace $z_{j,k}$ with $(1-w)$ in the constraints (2.10) for all $j = 1,\ldots,M$, $k \in T_j$ and solve (FSM) with the modified constraints. After the solution of (FSM) is obtained and the clustering stage is completed, we solve (SSM-2) for all the clusters by assuming that the trucks have a capacity of one truckload. Since we know that the excess capacity cannot be more than $w(1-\frac{1}{T'}) \leq w$, $\forall T'$, by Theorem 3, it is obvious that there can not be an excess capacity usage in any of the trucks in a solution found by the modified solution approach. Consequently, the modified Hierarchical Approach certainly finds a feasible solution for the original problem. Therefore, we can ensure that a feasible solution is computed by utilizing the upper bound in Theorem 3. However since we assume reduced capacities in the first stage, this solution may result in a higher cost.

Alternatively, we can solve the problem for only the clusters in which there exists some extra capacity usage in some of the trucks. For this alternative, we need to solve (FSM) for only the demand points in the cluster by assuming that all the trucks have a capacity of $(1 - w)$ truckload. In this way, we can find a better feasible solution from the first alternative, since this time we assume only some of the trucks have less capacity. In addition, if the cluster is small sized, then we can use the Direct Approach for solving the problem of satisfying the demand of each demand point in the cluster with the trucks having a capacity of one truckload. This alternative, gives a better feasible solution from the first two ones.

In this chapter, we analyzed the quality of solutions of the Hierarchical Approach and presented a sufficient condition in order for the solution of the Hierarchical Approach to be an optimal solution of the original problem. Next, we obtained a tight upper bound on the the maximum infeasibility residual for the solution of the Hierarchical Approach. In the next chapter, we introduce our computational study for the Hierarchical Approach and the Direct Approach and we compare the solution approaches based on the computational results.

Chapter 4

Computational Results

In this chapter, we present the computational results of the two solution approaches introduced in Chapter 2. We used GAMS 22.3, Microsoft Office Excel 2007 and Visual Basic 6.5 in the implementation of the solution approaches. In addition, CPLEX 10 is used as the solver in GAMS 22.3. All the computations were performed on a computer that has 4 GB DDR2 RAM, Intel Core 2 Duo 2.53 GHz P8700 processor and 32-bit Windows 7 operating system.

We designed an Excel-based user friendly interface on Excel. The user enters the number of items ordered by each demand point for each item type in an Excel sheet in which there exists a column for each demand point and a row for each item type and clicks a button that activates the Visual Basic code. All the necessary data is exported to different sheets in a format such that GAMS can use them as input and then GAMS is called by Visual Basic. Next, GAMS starts to solve the models with the input data. We used `xlimport`, a procedure that imports data from a spreadsheet into a GAMS program, and `Gdxxrw`, a GAMS utility to read Excel spreadsheet data, in order to get the input data from Excel sheets. We used the following options in each of the GAMS models:

`reslim`	=	43200
`optcr`	=	0.00
`nodefileind`	=	3
`solvefinal`	=	0
`names`	=	0

The first option guarantees that the running time for a GAMS model can be at most 12 hours. If the optimal solution cannot be obtained after 12 hours, then GAMS gives the best feasible solution obtained up to that time. In the second option, relative optimality criteria is set to 0 which means that GAMS will terminate when it finds an optimal solution unless another termination criterion is satisfied. In the third option, we specify how to handle the node files during the MIP processing. By default, CPLEX transfers nodes to node files when in-memory set is larger than 128 MBytes and it keeps the resulting node files in compressed form in memory. By setting `nodefileind` to 3, node files are stored on the disk in compressed form. This provides more memory that can be used. The last two options are modified in order to avoid an 'Out of Memory' error. As we are interested in the primal values of the solution, we do not need to see the marginal values of variables and there is no need to solve the final problem for the fixed values of the variables. This is guaranteed by the fourth option. The last option prevents GAMS names for the variables and equations to be loaded into CPLEX and this leads to less memory usage.

After GAMS terminates, we transfer the solution found to an Excel spreadsheet. We use `xldump`, a procedure that can be used to export data from a GAMS program to a spreadsheet, in order to achieve this. Then, we arrange the solution such that there exists an Excel sheet for each truck that will be used. In each sheet, the demand points that will be visited by the truck that corresponds to that sheet and the number of items that will be carried to these demand points are specified. This makes the solution easy to understand. The flow chart for the solution approaches can be seen in Figure (4.1).

We used the data adapted from one of the major automobile companies in

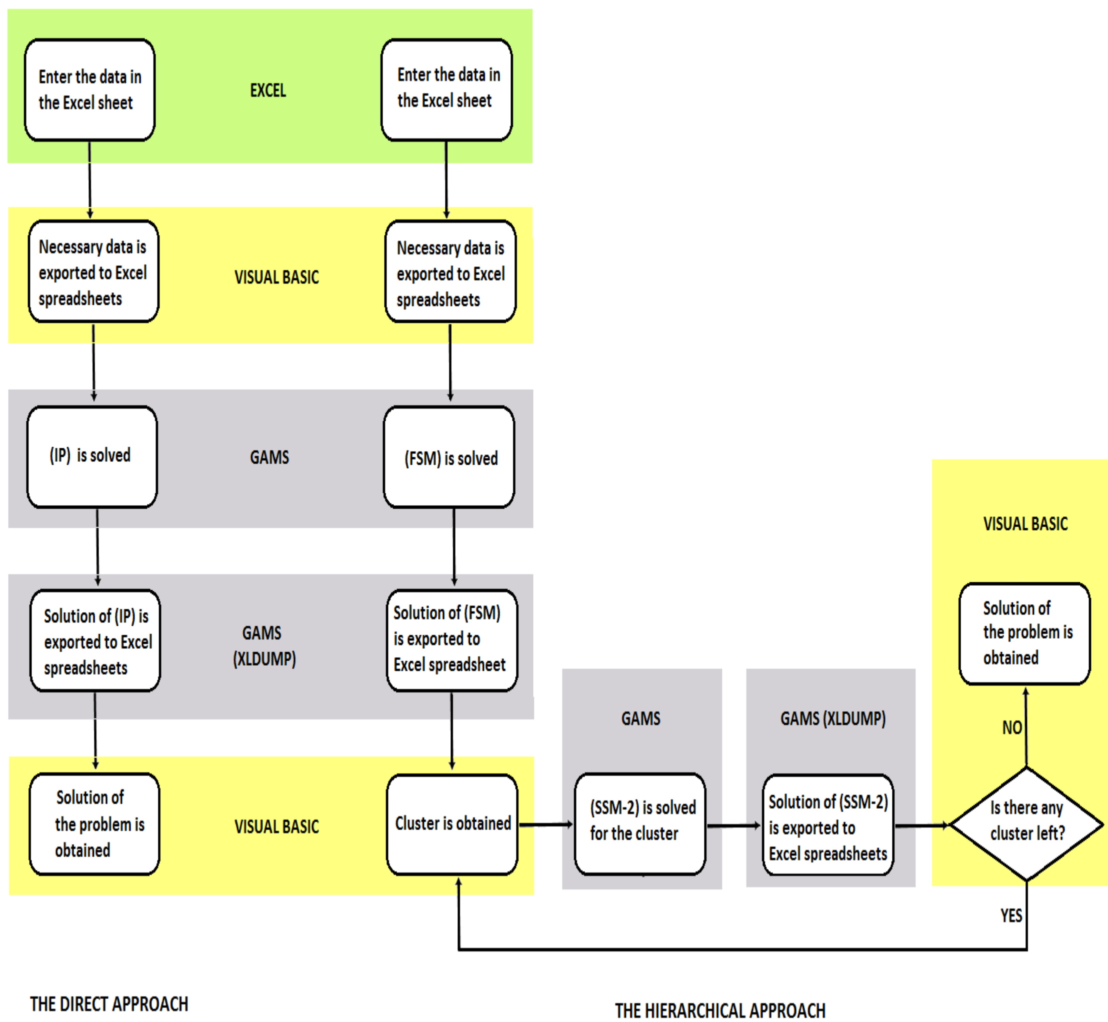

Figure 4.1: Flowchart of the solution approaches

Turkey in order to compare the two different solution approaches. There are 63 demand points and 71 different types of items. However, not all of the demand points place an order each day. Similarly, some of the items may not be demanded in some days.

Daily demands of the demand points for items are satisfied by trucks. Trucks are outsourced from a logistics company. According to the contract between the companies, there is no limit on the number of trucks; the company gets as many trucks as necessary each day. Therefore, there is no limit on the number of trucks.

However, the solution approaches we introduced in Chapter 2, assume that there are t_j trucks whose final destination is the demand point j for each demand point j, $j = 1, \ldots, M$ where t_j, $j = 1, \ldots, M$, is a known, fixed number. Therefore, we need to fix the number of trucks before the models are solved. In order to achieve this, we need to find sufficiently large values for each t_j, $j = 1, \ldots, M$. The following part shows how large t_j should be so that an optimal solution to the problem can be found by the solution approaches introduced in Chapter 2.

Assume that there is a truck whose final destination is the demand point j. We assume that, demand point i can be visited by a truck whose final destination is the demand point j if and only if $s_{i,j} < \infty$. As we stated in Chapter 2, if $s_{i,j} > f_i$, then it is obvious that sending a truck to demand point i costs less than a visit of the truck whose final destination is the demand point j to i. In this case, we let $s_{i,j} = \infty$. Therefore, the demand point i can be visited by a truck whose final destination is the demand point j if and only if $s_{i,j} \leq f_i$. Let $R_j \subseteq R = \{1, \ldots, M\}$ be the set of demand points that can be visited by the truck whose final destination is the demand point j. Observe that once the final destination of a truck is determined, then the set of demand points that can be visited by this truck becomes available; therefore R_j is known for each $j \in R$. As a result, we can define

$$R_j = \{i \in R : s_{i,j} \leq f_i, \quad i \neq j\}$$

for all $j \in R$.

A truck whose final destination is the demand point j can carry demand of demand points $i \in R_j \cup \{j\}$. Consequently, there may be a solution such that the trucks whose final destination is the demand point j satisfy demand of all the demand points in $R_j \cup \{j\}$. Therefore, t_j, the number of trucks whose final destination is the demand point j, should be large enough to be able to satisfy the demand of j and demand points in R_j. Therefore, in order to find the sufficient number of trucks for each demand point j, we need to consider demand of all the demand points in $R_j \cup \{j\}$, $j \in R$.

In order to find the sufficient number of trucks t_j for each j, we use the algorithm described in Algorithm 1.

Algorithm 1 Algorithm For Finding Sufficient Number of Trucks

```
(Input) R_j, set of demand points for each demand point j, j ∈ R;
D_{i,l}, demand of demand point i for item type l, i ∈ R, l = 1,...,N;
w_l, volume (in truckload) of each item type for l = 1,...,N
(Output) t_j, sufficient number of trucks for satisfying demand of all the points
in R_j ∪ {j}, j ∈ R;
t'_i, sufficient number of trucks for satisfying demand of the demand point i,
i ∈ R
for i ∈ R do
  t'_i ← 0
  if Σ_{l=1}^{N} D_{i,l} > 0 then
    t'_i ← 1
    B_1 ← 0
    for l := 1 to N do
      if D_{i,l} > 0 then
        for n := 1 to D_{i,l} do
          if {k ∈ {1,...,t'_i} : w_l + B_k ≤ 1} = ∅ then
            t'_i ← t'_i + 1
            B_{t'_i} ← w_l
          else
            k* ← arg min{k ∈ {1,...,t'_i} : w_l + B_k ≤ 1}
            B_{k*} ← w_l + B_{k*}
for j ∈ R do
  t_j ← t'_j
  for i ∈ R_j do
    t_j ← t_j + t'_i
```

Satisfying the demand of a demand point for each item type with the minimum number of trucks resembles a bin packing problem. In our problem, trucks corresponds to bins and items ordered by the demand point correspond to the items that will be placed into bins. The problem is to find the minimum number of bins for packing all the items. Therefore, in Algorithm 1, we modified a greedy algorithm for the bin packing problem.

In this algorithm, for each demand point $i = 1, \ldots, M$, the demand of i for each item type l is placed into trucks. When all the demand of i is placed into trucks, the number of trucks necessary for satisfying the demand of demand point

i is found and is denoted by t'_i. Note that t'_i is an upper bound for the minimum number of trucks that can satisfy the demand of demand point i as this is a feasible solution for the bin packing problem for demand point i. After t'_i is found for all $i = 1, \ldots, M$, then $t_j = t'_j + \sum_{i \in R_j} t'_i$ is computed and t_j gives us the sufficient number of trucks for each demand point $j = 1, \ldots, M$.

In the following lemma, we show that Algorithm 1 finds the sufficient number of trucks for each demand point so at most t_j trucks are used for each demand point $j = 1, \ldots, M$ in all of the optimal solutions.

Lemma 3 *There cannot be an optimal solution to the problem introduced in Chapter 2 in which more than t_j trucks are used for any demand point j, $j = 1, \ldots, M$, where t_j denotes the number of trucks that have demand point j as the final destination which is found by Algorithm 1.*

Proof. Let $R = \{1, \ldots, M\}$ be the set of demand points. Suppose that for each demand point $j \in R$, the bin packing problem is solved. Let t_j^b be the optimal solution of the bin packing problem for the demand point $j \in R$. Note that t_j^b is the minimum number of trucks necessary for satisfying the demand of point j. Let $t_j^{b^*} = t_j^b + \sum_{i \in R_j} t_i^b$ be the number of trucks whose final destination is the demand point j. Note that $t'_j \geq t_j^b$ for all $j \in R$ and consequently $t_j \geq t_j^{b^*}$. Therefore, if we can show that $t_j^{b^*}$ trucks are sufficient for each demand point $j \in R$, then it is obvious that t_j trucks are also sufficient.

Suppose, for a contradiction, that there exists an optimal solution that uses more than $t_j^{b^*}$ trucks for some $j \in R$. Let $t_j^{o^*}$ be the number of used trucks whose final destination is the demand point j in the optimal solution and let $n_{i,j}$ be the number of visits to point $i \in R_j$ in the optimal solution by trucks whose final destination is the demand point j. Note that $\{j\} \cup R_j$ consists of all the demand points that can be served by a truck whose final destination is the demand point j. Therefore, there may be two cases in the optimal solution: $t_j^{o^*}$ trucks satisfy all the demand of all the demand points in the set $\{j\} \cup R_j$ or $t_j^{o^*}$ trucks satisfy some portion of demand of some of the points in $\{j\} \cup R_j$.

First Case: $t_j^{o^*}$ *trucks satisfy all the demand of all the demand points in*

the set $\{j\} \cup R_j$.
We know that there exists a solution in which $t_j^{b^*} = t_j^b + \sum_{i \in R_j} t_i^b$ trucks whose final destination is the demand point j are used such that t_i^b trucks satisfy demand of point $i \in R_j$ and t_j^b trucks satisfy the demand of point j. The cost of this solution is

$$f_j t_j^{b^*} + \sum_{i \in R_j} t_i^b s_{i,j}.$$

However, by our assumption $t_j^{b^*} < t_j^{o^*}$. As t_i^b is the minimum number of trucks necessary for satisfying the demand of $i \in R_j$, $n_{i,j} \geq t_i^b$ for all $i \in R_j$. Consequently,

$$f_j t_j^{b^*} + \sum_{i \in R_j} t_i^b s_{i,j} < f_j t_j^{o^*} + \sum_{i \in R_j} n_{i,j} s_{i,j},$$

which is a contradiction since satisfying all the demand of all the demand points in the set $\{j\} \cup R_j$ using $t_j^{b^*} < t_j^{o^*}$ trucks costs less.

Second Case: $t_j^{o^*}$ *trucks satisfy some portion of demand of some of the points in* $\{j\} \cup R_j$.
Let $\bar{R}_j \subset \{j\} \cup R_j$ be the set of demand points which are visited by trucks whose final destination is the demand point j in the optimal solution. Similarly, let $D_{i,l,j} \leq D_{i,l}$, $i \in \bar{R}_j$, $l = 1, \ldots, N$, $j \in R$, be the number of type l items demanded by point i and carried by trucks whose final destination is the demand point j in the optimal solution. Suppose that we solve the bin packing problem for satisfying the demand of points in $\bar{R}_j$ where the demand of each point for each item type is $D_{i,l,j}$, $i \in \bar{R}_j$, $l = 1, \ldots, N$. Let the solution of the bin packing problem for each demand point $i \in \bar{R}_j$ be $\bar{t}_i^b$ and let $\bar{t}_j^{b^*} = \sum_{i \in \bar{R}_j} \bar{t}_i^b$. It is obvious that $\bar{t}_i^b \leq t_i^b$ for all $i \in \bar{R}_j$ and consequently $\bar{t}_j^{b^*} \leq t_j^{b^*}$. The bin packing solution gives a solution to the problem such that $\bar{t}_i^b$ trucks satisfy the demand of points in $\bar{R}_j$ where demand is $D_{i,l,j}$. Then, the cost of this solution is

$$f_j \bar{t}_j^{b^*} + \sum_{i \in \bar{R}_j \setminus \{j\}} \bar{t}_i^b s_{i,j}$$

and as a result of the argument above

$$f_j \bar{t}_j^{b^*} + \sum_{i \in \bar{R}_j \setminus \{j\}} \bar{t}_i^b s_{i,j} \leq f_j t_j^{b^*} + \sum_{i \in \bar{R}_j \setminus \{j\}} t_i^b s_{i,j}.$$

By the first case we know that

$$f_j t_j^{b^*} + \sum_{i \in \bar{R}_j \setminus \{j\}} t_i^b s_{i,j} < f_j t_j^{o^*} + \sum_{i \in \bar{R}_j \setminus \{j\}} n_{i,j} s_{i,j},$$

therefore

$$f_j \bar{t}_j^{b^*} + \sum_{i \in \bar{R}_j \setminus \{j\}} \bar{t}_i^b s_{i,j} < f_j t_j^{o^*} + \sum_{i \in \bar{R}_j \setminus \{j\}} n_{i,j} s_{i,j}$$

which is a contradiction.

As a consequence, there cannot be an optimal solution to the problem that uses more than $t_j^{b^*}$ trucks for some $j \in R$. Since $t_j \geq t_j^{b^*}$ for all $j \in R$, Algorithm 1 finds a sufficient number of trucks for each demand point. □

The following lemma shows that solution of the Algorithm 1 cannot be improved.

Lemma 4 *Solution of the Algorithm 1 is tight, i.e., there exists an instance of the problem for which there may be an optimal solution to a problem that uses exactly t_j trucks for some $j \in R$.*

Proof. Consider the following example:

Suppose that $R = \{A, B, C\}$ and consider the cost table below. Assume that in the table, $s_{j,j}$ corresponds to f_j for each $j = A, B, C$ and the other cells indicate the visiting costs. For instance, $f_A = 10$ and $s_{B,A} = 5$. Therefore, final destination cost of each truck whose final destination is the demand point A is 10 units and a visit of a truck whose final destination is the demand point A to the demand point B costs 5 units.

$s_{i,j}$	j		
i	A	B	C
A	10	∞	∞
B	5	20	∞
C	5	∞	20

Then, $R_A = \{B, C\}$ and $R_B = R_C = \emptyset$. Assume that each demand point has exactly one truckload of demand. Then, Algorithm 1 gives the following results:

$$t'_A = 1 \; ; \; t'_B = 1 \; ; \; t'_C = 1;$$
$$t_A = 1 + 1 + 1 = 3 \; ; \; t_B = 1 \; ; \; t_C = 1.$$

The optimal solution of this problem is to send 3 trucks whose final destination is the demand point A. The first truck goes directly to point A (costs 10 units). The second truck first visits point B then goes to point A (costs $5 + 10 = 15$ units). Finally, the last one first visits point C then goes to point A (costs $5 + 10 = 15$ units); and the total cost of the optimal solution is $10 + 15 + 15 = 40$ units.

Notice that $t_A = 3$ and all of them are used in the optimal solution. Therefore, the upper bounded solution resulting from Algorithm 1 cannot be improved. $\square$

Therefore, initially we create t_j trucks for each demand point $j = 1, \ldots, M$ and assign the demand point j to these trucks as the final destination. This procedure does not cut off the optimal solution as shown by Lemma 4. In addition, t_j, $j = 1, \ldots, M$ cannot be improved as shown by Lemma 4.

As the company provided us only with the data of 11 days, we performed all the computational studies on this data. The data we used can be summarized in the following table. In addition, the detailed data can be seen in Appendix A.1 - A.24. The number of trucks defined for each problem in Table 4.1 is computed by Algorithm 1.

We applied the two different solution approaches for solving the problem for

Table 4.1: Input Data

Day #	# of Demand Points	# of Item Types	# of Trucks Defined
Day 1	14	45	77
Day 2	11	50	39
Day 3	12	47	46
Day 4	17	49	101
Day 5	10	37	24
Day 6	19	44	60
Day 7	23	49	163
Day 8	10	49	40
Day 9	7	39	26
Day 10	8	41	34
Day 11	27	48	169

the data of each day. In Table 4.2, the number of discrete and continuous variables, total number of variables and number of constraints for the first stage model and for each subproblem of the second stage are given. As it can be seen from the table, the first stage problem is a larger problem compared to the subproblems of the second stage in terms of the number of variables and the number of constraints. In the first stage, we solve a relatively larger problem and in the second stage we solve several small subproblems.

We prefer to compare the model statistics of (IP) and (FSM), as subproblems of the second stage are very small problems compared to the first stage problem. In Table 4.3, total number of variables, number of discrete variables and number of constraints of (IP) and (FSM) are presented. As illustrated in the table, the first stage problem is also a smaller problem than the problem of the Direct Approach in terms of the number constraints and variables. In (IP), only the variable that corresponds to the objective function value is a continuous variable, all of the other variables are discrete variables. As it can be seen from the table, there is a big difference between the number of discrete variables of (IP) and (FSM). This is a serious disadvantage for the Direct Approach. In addition, the number of constraints of (IP) is larger than the number of constraints of (FSM) in all of the days. Therefore, in both of the stages of the Hierarchical Approach, relatively smaller and easier problems are solved. This makes the Hierarchical

Table 4.2: The Hierarchical Approach Statistics

Day #	**# of**	**First Stage Model**	**Second Stage Model-2**													
			Cluster #													
			1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**
1	Discrete Variables	16170	360	45	720	90	90	90	90							
	Continuous Variables	15093	5	2	5	2	2	3	2							
	Variables (in Total)	31263	365	47	725	92	92	93	92							
	Constraints	17263	95	47	185	92	92	48	92							
2	Discrete Variables	5148	50	300	150	50	50	300	50	50	50					
	Continuous Variables	4720	2	4	4	2	2	4	2	2	2					
	Variables(in Total)	9868	52	304	154	52	52	304	52	52	52					
	Constraints	5589	52	104	54	52	52	104	52	52	52					
3	Discrete Variables	7176	940	47	188	564	94									
	Continuous Variables	6625	6	2	3	5	2									
	Variables (in Total)	13801	946	49	191	569	96									
	Constraints	7741	194	49	97	146	96									
4	Discrete Variables	30906	588	441	441	392	98	98	98							
	Continuous Variables	29190	4	4	4	3	3	3	3							
	Variables(in Total)	60096	592	445	445	395	101	101	101							
	Constraints	32641	200	151	151	199	52	52	52							
5	Discrete Variables	2640	37	111	148	37	37	74								
	Continuous Variables	2401	2	2	3	2	2	2								
	Variables (in Total)	5041	39	113	151	39	39	76								
	Constraints	2891	39	113	77	39	39	76								
6	Discrete Variables	22800	264	88	528	88	88	44	44	132	44	88				
	Continuous Variables	21661	4	2	5	2	2	2	2	2	2	2				
	Variables (in Total)	44461	268	90	533	90	90	46	46	134	46	90				
	Constraints	23960	92	90	137	90	90	46	46	134	46	90				

Table 4.2 - The Hierarchical Approach Statistics (Cont'd)

Day #	**# of**	**First Stage Model**	**Second Stage Model-2**													
			Cluster #													
			1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**
7	Discrete Variables	89976	98	196	392	196	98	98	294	294	392	98				
	Continuous Variables	86228	2	3	5	3	2	2	4	3	3	2				
	Variables(in Total)	176204	100	199	397	199	100	100	298	297	395	100				
	Constraints	93749	100	101	103	101	100	100	102	150	199	100				
8	Discrete Variables	4400	196	49	147	588	196	49	49							
	Continuous Variables	4001	5	2	2	7	5	2	2							
	Variables (in Total)	8401	201	51	149	595	201	51	51							
	Constraints	4811	54	51	147	105	54	51	51							
9	Discrete Variables	1456	39	39	234	78	78	39								
	Continuous Variables	1275	2	2	4	3	3	2								
	Variables (in Total)	2731	41	41	238	81	81	41								
	Constraints	1646	41	41	82	42	42	41								
10	Discrete Variables	2448	164	41	164	615										
	Continuous Variables	2177	3	2	3	6										
	Variables (in Total)	4625	167	43	167	621										
	Constraints	2729	85	43	85	129										
11	Discrete Variables	127764	48	192	48	144	144	192	288	288	96	96	144	144	48	48
	Continuous Variables	123202	2	3	2	2	2	5	4	4	2	2	2	2	2	2
	Variables (in Total)	250966	50	195	50	146	146	197	292	292	98	98	146	146	50	50
	Constraints	132355	50	99	50	146	146	53	100	100	98	98	146	146	50	50

Approach more preferable over the Direct Approach.

Table 4.3: Direct Approach vs. Hierarchical Approach Statistics

Day #	**# of Discrete Variables**		**Total # of Variables**		**# of Constraints**	
	(IP)	(FSM)	(IP)	(FSM)	(IP)	(FSM)
1	695310	16170	695311	31263	17879	17263
2	241098	5148	241099	9868	6128	5589
3	318504	7176	318505	13801	8293	7741
4	1461167	30906	1461168	60096	33457	32641
5	91440	2640	91441	5041	3251	2891
6	975840	22800	975841	44461	24777	23960
7	4315099	89976	4315100	176204	94853	93749
8	200400	4400	200401	8401	5291	4811
9	51142	1456	51143	2731	1912	1646
10	91664	2448	91665	4625	3049	2729
11	6041412	127764	6041413	250966	133624	132355

In Table 4.4, the results obtained from the two solution approaches are summarized. Resource usage reported for the Hierarchical Approach is the total time necessary for solving the first stage and each of the subproblems of the second stage. As it can be seen in the table, the resource usage of the Direct Approach is significantly larger than the resource usage of the Hierarchical Approach. This is an expected situation as the problem sizes are very different from each other. As indicated before, relatively smaller and easier to solve problems are solved in both of the stages of the Hierarchical Approach.

It can be observed from the table that maximum infeasibility residual is 0 for all the days. By Proposition 1, this implies the Hierarchical Approach finds feasible solutions for all days. For the data of 11 days, there are only two days, Day 7 and Day 11, that cannot be solved to optimality by the Hierarchical Approach within the limited time. However, these days cannot be solved to optimality by the Direct Approach either. Nevertheless, for Day 7, the Hierarchical Approach finds a better solution with a smaller cost in the same time. Note that, the Hierarchical Approach finds a solution in which 19 trucks are used in total, whereas the Direct Approach finds a solution in which 20 trucks are used. For Day 11, the two solution approaches find solutions with the same cost. For the other

Table 4.4: Direct Approach vs. Hierarchical Approach Results

Day #	**Hierarchical Approach**				**Direct Approach**		
	Cost	Resource Usage (sec.)	# of Trucks Used	Maximum Infeasibility Residual	Cost	Resource Usage (sec.)	# of Trucks Used
1	8236.2	52	14	0	8236.2	43201*	14
2	8221.71	22	15	0	8221.71	1709	15
3	10308.240	3	13	0	10308.240	43201*	13
4	11141.19	843	17	0	11141.19	43202*	17
5	4276.35	9	7	0	4276.35	108	7
6	10141.40	2427	15	0	10141.40	43201*	15
7	12887.65	43202*	19	0	12910.49	43204*	20
8	7704.08	2	18	0	7704.08	14909	18
9	3353.59	18	10	0	3353.59	43201*	10
10	3956.86	1	10	0	3956.86	43201*	10
11	13613.84	43201*	22	0	13613.84	43205*	22

* Terminated due to time limit.

days, all the days except the day 7 and 11, the Hierarchical Approach finds the optimal solutions with impressive resource usages. Moreover, note that, the cost of the solution of the Hierarchical Approach is less than or equal to the cost of the solution of the Direct Approach for each day. Therefore, we can conclude that the solution of the Hierarchical Approach is at least as good as the solution of the Direct Approach for each day on this data set.

The Direct Approach cannot solve the problem to optimality for the data of 8 days in the limited time. The results reported for these days are the best solutions that are found before the time limit is exceeded. It finds optimal solutions only for the days 2, 5 and 8. However, as seen from the table, there is a big difference between the solution times for the solution approaches for these days . The Hierarchical Approach finds the optimal solutions in significantly less time. Moreover, note that the sizes of the problem for these days are very small compared to the other days. Therefore, the Direct Approach can be used for solving the problem for small sizes whereas the Hierarchical Approach performs well also for larger instances of the problem.

Although the Direct Approach can not solve the problem to optimality for the data of days 1,3,4,5,6,9,10,11, note that costs of the solutions found by both of the solution approaches are the same for these days. Since the Hierarchical Approach finds the optimal solutions for these days, the Direct Approach also finds the optimal solutions for these days. However, CPLEX cannot verify that an optimal solution is found because of the size of the (IP).

In summary, our computational study shows that the Hierarchical Approach is a better solution approach than the Direct Approach in many aspects such as the resource usage and the quality of the solution found in the limited time. For small instances of the problem the Direct Approach can be used. However, for larger instances of the problem it is better to use the Hierarchical Approach.

Chapter 5

Conclusion and Future Research

In this thesis, we consider a multicommodity distribution problem. We assume that there is a central depot which houses a number of different types of items. There is a finite number of geographically dispersed demand points which place orders for these items on a daily basis. The demand of these demand points should be satisfied from this central depot. We assume that a finite number of identical trucks with predetermined destinations are used for the distribution of the items from the central depot to each demand point. The demand of each demand point can be split among several trucks and a single truck is allowed to visit several demand points. The problem is to satisfy the demand of each demand point with the minimum total distribution cost while respecting the capacity of each truck. The cost structure is dictated by the final destinations of trucks used in the distribution of the items and the set of demand points visited by each truck.

Since the cost structure of the problem makes it different from the VRP, we developed two different solution approaches in an attempt to solve the problem.

The Direct Approach solves the problem with a mixed integer linear programming model which is called (IP). The purpose of this model is to select the trucks that will be used and decide how to satisfy demands of all the demand points with the minimum transportation cost while respecting the truck capacities. As the number of integer and binary variables in (IP) quickly increases with the number of demand points, item types and trucks, the model becomes difficult to solve for larger instances of the problem.

We developed the Hierarchical Approach in an attempt to solve larger instances of the problem in a more effective way. The Hierarchical Approach consists of two main stages and an intermediate stage. In the first stage, we aggregate the demand of each demand point and we solve a problem to satisfy the aggregated demand of each demand point with the minimum transportation cost while respecting the truck capacities. By the solution of the first stage we determine the trucks that will be used in the distribution and the demand points that each truck will visit. According to this information, we divide the problem into subproblems in the intermediate stage which is called the clustering stage. We partition the set of the demand points and the used trucks into clusters so that there is no relation between any of the clusters. In the second stage, we solve each subproblem separately. We solve the problem of satisfying the demand of all the demand points in a cluster with the trucks that will visit them for each cluster. At the end of the second stage, we determine how to distribute the items into trucks so that the demand of each demand point is satisfied. As a drawback of the demand aggregation and disaggregation, we allow the excess capacity usage in the trucks in the second stage. Therefore, the Hierarchical Approach may find a solution that is not feasible for the original problem. However, we penalize it in the objective function. We showed that maximum infeasibility of the solution we obtain by the Hierarchical Approach is less than or equal to the maximum truck capacity that can be occupied by one unit of the ordered items. Then, we further improved this upper bound and found a tight bound on the maximum infeasibility residual for each truck.

We tested the solution approaches with real data set obtained from a major automotive company in Turkey. As the company provided us with only data of 11 days, we performed all the computational studies on this data. We used GAMS 22.3, CPLEX 10 as the solver in Gams 22.3, Microsoft Office Excel 2007 and Visual Basic 6.5 in the implementation of the solution approaches. We conclude that the Hierarchical Approach is more efficient than the Direct Approach in many aspects.

Although the Hierarchical Approach may find an infeasible solution for the original problem, it found a feasible solution in our computational study in all of the days. For the data of 11 days, there were only two days that could not be solved to optimality by the Hierarchical Approach within the limited time. However, these days also could not be solved to optimality by the Direct Approach. For the other days, the Hierarchical Approach found the optimal solutions with impressive resource usages. On the other hand, the Direct Approach could not solve the problem to optimality for the data of 8 days in the limited time.

As a conclusion, our computational study showed that the Hierarchical Approach is more effective than the Direct Approach in many aspects such as the resource usage and the quality of the solution found in the limited time. It is better to use the Hierarchical Approach in order to solve the problem as it less time consuming and it gives better solutions than the Direct Approach.

Several variations in the problem may be handled with small modifications in the solution approaches. For instance, we assumed that the trucks are identical. However, if the trucks are not identical, the constraints related to the truck capacity can be modified easily. In addition, we believe that more work can be done in an attempt to improve the solution approaches. For instance, if there exists an excess capacity usage in some of the trucks then maybe the problem can be solved for the trucks with less capacities. This would prevent an excess capacity usage in any of the trucks. However, this also may lead to a worse solution.

A further research may be an extension to the Hierarchical Approach by obtaining the clusters first. If the clusters can be obtained first somehow, then the solution to the problem can be found more easily. Then, the solution approach resembles the cluster first, route second methods. However, obtaining the clusters first is also an important problem in the literature.

Another further research may be to relax the assumption that any combination of items can be loaded into trucks as long as their total volume is less then or equal to one truckload. Without this assumption, the problem turns into the 3-D Bin Packing Problem which is a strongly NP-Hard problem. However, the problem may be more realistic since we do not ignore the geometry of the items anymore.

Bibliography

[1] Ali Al and J.L. Kernington, The Asymmetric m-Traveling Salesmen Problem: A Duality Based Branch-and-Bound Algortihm, *Discrete Applied Mathematics* 13, 259-276, 1986.

[2] C. Archetti, A. Hertz, M.G. Speranza, A Tabu Search Algorithm for the Split Delivery Vehicle Routing Problem, *Transportation Science* 40, 64-73, 2006.

[3] C. Archetti, M.W.P. Savelsbergh, M.G. Speranza, Worst-Case Analysis for Split Delivery Vehicle Routing Problem, *Transportation Science* 40(2), 226-234, 2006.

[4] M.L. Balinski, R.E. Quandt, On an Integer Program for a Delivery Program, *Operations Research* 12, 300-304, 1964.

[5] T. Bektaş, The Multiple Salesman Problem: An Overview of Formulations and Solution Procedures, *Omega: The International Journal of Management Science* 34, 209-219, 2006.

[6] R. Bellman, Dynamic Programming Treatment of the Traveling Salesman Problem, *Journal of the ACM* 9, 61-63, 1962.

[7] M. Bellmore, G.L.Nemhauser, The Traveling Salesman Problem: A Survey, *Operations Research*, Vol.16, No. 3, 538-558, May-June 1968.

[8] F. Bock, Mathematical Programming Solution of Traveling Salesman Examples, *Recent Advances in Mathematical Programming*, (R.L. Graves and P. Wolfe, eds.), 339-341 McGrraw-Hill, New York, 1963.

[9] J.B. Bramel, D. Simchi-Levi, A Location Based Heuristic for General Routing Problems, *Operations Research* 43, 649-660, 1995.

[10] R. E. Burkard, V. G. Deineko, R. Van Dal, J. A. A. Van Der Veen , G. J. Woeginger, Well-Solvable Special Cases of the Traveling Salesman: A Survey, *SIAM Review*, Vol. 40, No. 3, 496-546, Sep., 1998.

[11] N. Christofides, A. Mingozzi, P. Toth, Exact Algorithms for the Vehicle Routing Problem, Based on Spanning Tree and Shortest Path Relaxations, *Mathematical Programming* 20, 255-282,1981.

[12] G. Clarke, J. W. Wright, Scheduling of Vehicles from a Central Depot to a number of Delivery Points, *Operations Research* 12, 568-581, 1964.

[13] G. Dantzig, R. Fulkerson, S. Johnson, Solution of a Large-Scale Traveling-Salesman Problem, *Journal of the Operations Research Society of America*, Vol.2, No.4, 393-410, Nov., 1954.

[14] G. B. Dantzig, J. H. Ramser, The Truck Dispatching Problem, *Management Science*, 6, 80-91, 1959.

[15] M. Dror, P. Trudeau, Savings by Split Delivery Routing, *Transportation Science* 23, 141-145, 1989.

[16] M. Dror, G. Laporte, P. Trudeau, Vehicle Routing with Split Deliveries, *Discrete Applied Mathematics* 50, 239-254, 1994.

[17] W.L. Eastman, Linear Programming with Pattern Constraints, Ph.D. Dissertation, Harvard, 1958.

[18] M.L. Fisher, R. Jaikumar, A Generalized assignment Heuristic for Vehicle Routing, *Networks* 11, 109-124, 1981.

[19] , B.A. Foster, D.M. Ryan, An Integer Programming Approach to the Vehicle Scheduling Problem, *Operations Research* 27, 367-384, 1976.

[20] T.J. Gaskell, Bases for Vehicle Fleet Scheduling, *The Operations Research Society*, Vol. 18, No. 3, pages: 281-296, Sep., 1967.

[21] B. Gavish, A Note on the Formulation of the M-Salesman Traveling Salesman Problem, *Management Science*, Vol. 22, No. 6, 704-705, 1976.

[22] B. Gavish, K. Srikanth, An Optimal Solution Method for Large-Scale Multiple Traveling Salesmen Problems, *Operation Research* 34(5), 698-717, 1986.

[23] P.C. Gilmore and R.E. Gomory, Sequencing a One State Variable Machine: A Solvable Case of the Traveling Salesman Problem, *Operations Research* 12, 655-679,1964.

[24] B. L. Golden, T. L. Magnanti, H. Q. Nguyen, Implementing Vehicle Routing Algorithms, *Networks* 7, 113-148, 1977.

[25] R. H. Gonzales, Solution to the Traveling Salesman Problem by Dynamic Programming on the Hypercube, Technical Report No. 18, Operation Research Center, MIT, 1962.

[26] J. Gromicho, J. Paixao, I. Branco, Exact Solution of Multiple Traveling Salesmen Problem, In: Mustafa Akgül, et. al., editors. *Combinatorial Optimization*, NATO ASI Series, Vol.F82, Berlin: Springer; 291-291, 1992.

[27] D.J. Hatfield and J.F. Pierce, Production Sequencing by Combinatorial Programming, IBM Cambridge Scientific Center, Cambridge, Mass. , 1966.

[28] M. Held and R. M. Karp, A Dynamic Programming Approach to Sequencing Problems, *SIAM* 10, 196-210, 1962.

[29] M. Jin, K. Liu, R.O. Bowden, A Two-Stage Algorithm with Valid Inequalities for the Split Delivery Vehicle Routing Problem, *International Journal of Production Economics* 105, 228-242, 2007.

[30] İ. Kara, T. Bektaş, Integer Linear Programming Formulations of Multiple Salesman Problems and Its Variations, *European Journal of Operations Research* 174, 1449-1458, 2006.

[31] L.L. Karg and G.L. Thompson, A Heuristic Approach to Solving Traveling Salesman Problems, *Management Science* 10, 225-248, 1964.

[32] G. Laporte, Y. Nobert, A Cutting Planes Algorithm for the m-Salesman Problem, *The Journal of the Operations Research Society*, Vol. 31, No. 11, 1017-1023, Nov., 1980.

[33] G. Laporte, The Vehicle Routing Problem: An Overview of Exact and Approximate Algorithms, *European Journal of Operations Research* 59, 345-358, 1992.

[34] G. Laporte, M. Gendreau, J. Potvin, F. Semet, Classical and Modern Heuristics for the Vehicle Routing Problem, *International Transactions in Operations Research* 7, 285-300, 2000.

[35] C.G. Lee, M. Epelman, C.C. White III, Y.A. Bozer, A Shortest Path Approach to the Multiple-Vehicle Routing Problem with Split Pick-Ups, *Transportation Researh - Part B; Methodological* 40(4), 265-284, 2006.

[36] J.D.C. Little, K.G. Murty, D.W. Sweeney, C. Karel, A Algorithm for the Traveling Salesman Problem, *Recent Advences in Mathematical Programming* (R.L. Graves and P. wolfe, eds.), 311-318, 1963.

[37] G.T. Martin, An Accelerated Euclidean Algorithm for Integer Linear Programming, *Recent Advances in Mathematical Programming*, (R.L. Graves and P. Wolfe, eds.), 311-318, 1963.

[38] C. E. Miller, A. W. Tucker, R. A. Zemlin, Integer Programming Formulation Traveling Salesman Problem, *Journal of ACM* 2, 393-410, 1960.

[39] M.D. Nelson, K. E. Nygard, J. H. Griffin, W. E. Shreve, Implementation Techniques for the Vehicle Routing Problem, *Computers & Operations Research* 12, 273-283, 1985.

[40] H. Paessens, The Savings Algorithm for the Vehicle Routing Problem, *European Journal of Operation Research* 34, 336-344, 1988.

[41] J. Renaud, F.F. Boctor, G. Laporte, A Fast Composite Heuristic for the Symmetric Traveling Salesman Problem, *INFORMS Journal on Computing*, Vol. 8, No.2, 134-143, Spring 1996.

[42] J. Renaud, F.F. Boctor, G. Laporte, An Improved Petal Heuristic for the Vehicle Routing Problem, *Journal of the Operations Research Society* 47, 329-336, 1996b.

[43] D.M. Ryan, C. Hjorring, F. Glover, Extensions of the Petal Method for Vehicle Routing, *Journal of the Operations Research Society* 44, 289-296,1993.

[44] P. Toth, D. Vigo, Models, Relaxations and Exact Approaches for the Capacitated Vehicle Routing Problem, *Discrete Applied Mathematics* 123, 487-512, 2002.

[45] A. Wren, Computers in Transport Planning and Operation, Ian Allan, London, 1971.

[46] A. Wren, A. Holliday, Computer Scheduling of Vehicles from One or More Depots to a Number of Delivery Points, *Operations Research Quarterly* 23, 333-344, 1972.

[47] P. Yellow, A Computational Modification to the Savings Method of Vehicle Scheduling, *Operations Research Quarterly* 21, 281-283, 1970.

[48] UTIKAD, The International Transportation and Logistics Sector Analysis Report, 2007, ¡http://www.utikad.org.tr¿.

Appendix A

Data Used in the Computational Study

Table A.1: Capacity of a truck that is occupied by one unit of an item

l	w_l	l	w_l	l	w_l
1	0.00694	25	0.01389	49	0.00139
2	0.01042	26	0.01389	50	0.00042
3	0.00521	27	0.00833	51	0.00077
4	0.00694	28	0.00694	52	0.00144
5	0.00463	29	0.00833	53	0.00139
6	0.00521	30	0.01042	54	0.02778
7	0.00278	31	0.01389	55	0.05952
8	0.00417	32	0.00595	56	0.00083
9	0.00208	33	0.00347	57	0.01389
10	0.02083	34	0.01389	58	0.00167
11	0.04167	35	0.04167	59	0.00014
12	0.00833	36	0.04167	60	0.00042
13	0.00417	37	0.00069	61	0.00038
14	0.00521	38	0.00052	62	0.00595
15	0.00521	39	0.00052	63	0.00116
16	0.00521	40	0.00417	64	0.00042
17	0.00208	41	0.00833	65	0.00298
18	0.00208	42	0.00694	66	0.00219
19	0.01042	43	0.00260	67	0.00104
20	0.00833	44	0.00091	68	0.01042
21	0.01042	45	0.00144	69	0.00833
22	0.00694	46	0.01389	70	0.00042
23	0.00347	47	0.02083	71	0.04167
24	0.00595	48	0.00595		

Table A.2: Cost Matrix*

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
1	774.78	∞	∞	44.43	∞	298.96	∞	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	∞	∞
2	∞	285.88	∞	∞	∞	∞	∞	∞	∞	∞	∞	202	∞	∞	208.06	44.43	44.43	∞	∞
3	44.43	∞	1097.85	44.43	∞	298.96	∞	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	∞	∞
4	∞	∞	∞	375.97	∞	∞	∞	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	∞	∞
5	∞	∞	∞	343.4	1413.75	∞	44.43	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	264.62	∞
6	∞	∞	∞	44.43	∞	714.69	∞	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	∞	∞
7	∞	∞	∞	343.4	∞	∞	436.02	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	∞	∞
8	∞	∞	∞	44.43	∞	∞	∞	760.32	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	∞	∞
9	∞	∞	∞	∞	∞	∞	∞	∞	518.9	∞	44.43	62.62	∞	∞	208.06	44.43	44.43	∞	∞
10	∞	74.74	∞	∞	∞	∞	∞	∞	∞	578.95	∞	202	∞	305.02	208.06	44.43	44.43	∞	44.43
11	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	278.67	62.62	∞	∞	208.06	44.43	44.43	∞	∞
12	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	255.84	∞	∞	208.06	44.43	44.43	∞	∞
13	∞	∞	∞	∞	∞	∞	∞	∞	131.3	∞	44.43	62.62	676.25	∞	208.06	44.43	44.43	∞	∞
14	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	202	∞	480.45	208.06	44.43	44.43	∞	∞
15	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	224.61	44.43	44.43	∞	∞
16	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	67.27	31.24	∞	∞
17	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	31.24	87.34	∞	∞
18	∞	∞	∞	343.4	∞	∞	44.43	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	714.69	∞
19	∞	74.74	∞	∞	∞	∞	∞	∞	∞	∞	∞	202	∞	305.02	208.06	44.43	44.43	∞	527.3
20	∞	∞	∞	343.4	∞	∞	44.43	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	264.62	∞
21	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	62.62	∞	∞	208.06	44.43	44.43	∞	∞
22	44.43	∞	387.84	44.43	∞	298.96	∞	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	∞	∞
23	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	202	∞	∞	208.06	44.43	44.43	∞	∞
24	∞	74.74	∞	∞	∞	∞	∞	∞	∞	44.43	∞	202	∞	305.02	208.06	44.43	44.43	∞	44.43
25	∞	∞	∞	44.43	∞	44.43	∞	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	∞	∞
26	∞	∞	∞	343.4	∞	∞	44.43	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	264.62	∞
27	∞	∞	∞	343.4	∞	∞	44.43	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	264.62	∞
28	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	∞	∞
29	44.43	∞	∞	44.43	∞	298.96	∞	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	∞	∞
30	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	165.64	44.43	∞	∞	208.06	44.43	44.43	∞	∞
31	44.43	∞	∞	44.43	∞	298.96	∞	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	∞	∞
32	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	∞	∞

Table A.3: Cost Matrix(Cont'd)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
33	∞	74.74	∞	∞	∞	∞	∞	∞	∞	∞	∞	202	∞	∞	208.06	44.43	44.43	∞	∞
34	∞	74.74	∞	∞	∞	∞	∞	∞	∞	∞	∞	202	∞	∞	208.06	44.43	44.43	∞	∞
35	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	62.62	∞	∞	208.06	44.43	44.43	∞	∞
36	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	62.62	∞	∞	208.06	44.43	44.43	∞	∞
37	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	208.06	44.43	44.43	∞	∞
38	∞	74.74	∞	∞	∞	∞	∞	∞	∞	∞	∞	202	∞	∞	208.06	44.43	44.43	∞	∞
39	44.43	∞	∞	44.43	∞	298.96	∞	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	∞	∞
40	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	62.62	∞	∞	208.06	44.43	44.43	∞	∞
41	44.43	∞	∞	44.43	∞	298.96	∞	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	∞	∞
42	∞	∞	∞	44.43	∞	44.43	∞	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	∞	∞
43	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	∞	∞
44	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	202	∞	90.9	208.06	44.43	44.43	∞	∞
45	∞	∞	∞	44.43	∞	44.43	∞	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	∞	∞
46	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	62.62	∞	∞	208.06	44.43	44.43	∞	∞
47	∞	∞	∞	44.43	∞	298.96	∞	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	∞	∞
48	∞	∞	∞	44.43	∞	44.43	∞	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	∞	∞
49	44.43	∞	∞	44.43	∞	298.96	∞	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	∞	∞
50	∞	∞	∞	343.4	∞	∞	44.43	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	264.62	∞
51	∞	∞	∞	∞	∞	∞	∞	∞	44.43	∞	44.43	62.62	353.5	∞	208.06	44.43	44.43	∞	∞
52	∞	∞	∞	343.4	∞	∞	44.43	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	264.62	∞
53	∞	∞	∞	343.4	∞	∞	44.43	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	264.62	∞
54	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	202	∞	44.43	208.06	44.43	44.43	∞	∞
55	∞	∞	∞	343.4	∞	∞	44.43	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	264.62	∞
56	44.43	∞	387.84	44.43	∞	298.96	∞	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	∞	∞
57	∞	∞	∞	343.4	∞	∞	44.43	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	264.62	∞
58	∞	∞	∞	343.4	∞	∞	44.43	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	264.62	∞
59	∞	∞	∞	460.56	∞	∞	∞	∞	∞	∞	∞	202	∞	∞	117.16	44.43	44.43	∞	∞
60	∞	∞	∞	343.4	44.43	∞	44.43	∞	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	264.62	∞
61	∞	∞	∞	∞	∞	∞	∞	∞	44.43	∞	44.43	62.62	∞	∞	208.06	44.43	44.43	∞	∞
62	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	202	∞	44.43	208.06	44.43	44.43	∞	∞
63	∞	∞	∞	44.43	∞	∞	∞	44.43	∞	∞	∞	202	∞	∞	44.43	44.43	44.43	∞	∞

Table A.4: Cost Matrix(Cont'd)

	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38
1	∞	∞	∞	∞	∞	∞	∞	∞	92.92	∞	∞	∞	341.38	∞	∞	∞	∞	∞	∞
2	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	103.02
3	∞	∞	∞	∞	∞	∞	∞	∞	92.92	137.36	∞	290.88	341.38	∞	∞	∞	∞	∞	∞
4	∞	∞	∞	∞	∞	∞	∞	∞	92.92	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
5	232.3	∞	∞	∞	∞	∞	44.43	44.43	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
6	∞	∞	∞	∞	∞	∞	∞	∞	92.92	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
7	∞	∞	∞	∞	∞	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
8	∞	∞	∞	∞	∞	∞	∞	∞	92.92	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞
9	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	44.43	∞	∞
10	∞	∞	∞	214.12	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	44.43	∞	∞	44.43	44.43
11	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
12	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
13	∞	401.98	∞	∞	∞	∞	∞	∞	∞	∞	474.7	∞	∞	∞	∞	44.43	44.43	∞	∞
14	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	103.02
15	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
16	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
17	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
18	∞	∞	∞	∞	∞	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
19	∞	∞	∞	214.12	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	44.43	∞	∞	44.43	44.43
20	783.15	∞	∞	∞	∞	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
21	∞	518.9	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	169.68	169.68	∞	∞
22	∞	∞	1346.48	∞	∞	∞	∞	∞	92.92	44.43	∞	290.88	341.38	∞	∞	∞	∞	∞	∞
23	∞	∞	∞	396.37	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	103.02
24	∞	∞	∞	214.12	654.62	∞	∞	∞	∞	∞	317.14	∞	∞	44.43	44.43	∞	∞	44.43	44.43
25	∞	∞	∞	∞	∞	1300.83	∞	∞	92.92	∞	∞	∞	341.38	∞	∞	∞	∞	∞	∞
26	232.3	∞	∞	∞	∞	∞	1233.57	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
27	232.3	∞	∞	∞	∞	∞	44.43	1369.32	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
28	∞	∞	∞	∞	∞	∞	∞	∞	263.05	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
29	∞	∞	∞	∞	∞	∞	∞	∞	92.92	992.14	∞	∞	341.38	∞	∞	∞	∞	∞	∞
30	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	594.56	∞	∞	∞	∞	∞	∞	∞	∞
31	∞	∞	∞	∞	∞	∞	∞	∞	92.92	∞	∞	992.14	341.38	∞	∞	∞	∞	∞	∞
32	∞	∞	∞	∞	∞	∞	∞	∞	92.92	∞	∞	∞	744.72	∞	∞	∞	∞	∞	∞

Table A.5: Cost Matrix(Cont'd)

	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38
33	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	293.08	∞	∞	∞	44.43	22.82
34	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	323.11	∞	∞	44.43	44.43
35	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	345.92	∞	∞	∞
36	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	375.97	∞	∞
37	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	240.22	∞
38	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	293.08
39	∞	∞	∞	∞	∞	∞	∞	∞	92.92	∞	∞	∞	341.38	∞	∞	∞	∞	∞	∞
40	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	44.43	∞	∞
41	∞	∞	∞	∞	∞	∞	∞	∞	92.92	∞	∞	∞	341.38	∞	∞	∞	∞	∞	∞
42	∞	∞	∞	∞	∞	∞	∞	∞	92.92	∞	∞	∞	341.38	∞	∞	∞	∞	∞	∞
43	∞	∞	∞	∞	∞	∞	∞	∞	92.92	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
44	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	103.02
45	∞	∞	∞	∞	∞	∞	∞	∞	92.92	∞	∞	∞	341.38	∞	∞	∞	∞	∞	∞
46	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
47	∞	∞	∞	∞	∞	∞	∞	∞	92.92	∞	∞	∞	341.38	∞	∞	∞	∞	∞	∞
48	∞	∞	∞	∞	∞	∞	∞	∞	92.92	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
49	∞	∞	∞	∞	∞	∞	∞	∞	92.92	44.43	∞	290.88	341.38	∞	∞	∞	∞	∞	∞
50	44.43	∞	∞	∞	∞	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
51	∞	401.98	∞	∞	∞	∞	∞	∞	∞	∞	474.7	∞	∞	∞	∞	44.43	44.43	∞	∞
52	44.43	∞	∞	∞	∞	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
53	44.43	∞	∞	∞	∞	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
54	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	103.02
55	232.3	∞	∞	∞	∞	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
56	∞	∞	∞	∞	∞	∞	∞	∞	92.92	44.43	∞	290.88	341.38	∞	∞	∞	∞	∞	∞
57	232.3	∞	∞	∞	∞	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
58	44.43	∞	∞	∞	∞	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
59	∞	24.24	∞	∞	∞	∞	∞	∞	210.08	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
60	232.3	∞	∞	∞	∞	∞	44.43	44.43	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
61	∞	401.98	∞	∞	∞	∞	∞	∞	∞	∞	474.7	∞	∞	∞	∞	44.43	44.43	∞	∞
62	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	103.02
63	∞	∞	∞	∞	∞	∞	∞	∞	92.92	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞

Table A.6: Cost Matrix(Cont'd)

	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57
1	∞	∞	∞	∞	44.43	∞	∞	∞	70.7	450.46	∞	∞	∞	∞	∞	∞	∞	∞	∞
2	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
3	171.7	∞	46.46	∞	44.43	∞	∞	∞	70.7	450.06	222.2	∞	∞	∞	∞	∞	∞	∞	∞
4	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
5	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	∞	123.22
6	∞	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
7	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
8	∞	∞	∞	∞	450.46	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
9	∞	70.7	∞	∞	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
10	∞	∞	∞	∞	∞	351.48	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
11	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
12	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
13	∞	44.43	∞	∞	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
14	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
15	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
16	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
17	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
18	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
19	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
20	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
21	∞	371.68	∞	∞	∞	∞	∞	96.96	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
22	171.7	∞	96.96	∞	44.43	∞	∞	∞	70.7	450.06	44.43	∞	∞	∞	∞	∞	∞	44.43	∞
23	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
24	∞	∞	∞	∞	∞	351.48	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
25	∞	∞	373.7	44.43	44.43	∞	44.43	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞
26	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	∞	123.22
27	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	∞	123.22
28	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
29	171.7	∞	∞	∞	44.43	∞	∞	∞	70.7	450.06	∞	∞	∞	∞	∞	∞	∞	∞	∞
30	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
31	44.43	∞	∞	∞	44.43	∞	∞	∞	70.7	450.06	∞	∞	∞	∞	∞	∞	∞	∞	∞
32	∞	∞	∞	∞	450.46	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞

Table A.7: Cost Matrix(Cont'd)

	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57
33	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
34	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
35	∞	∞	∞	∞	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
36	∞	∞	∞	∞	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
37	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
38	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
39	880.44	∞	∞	∞	44.43	∞	∞	∞	70.7	450.06	∞	∞	∞	∞	∞	∞	∞	∞	∞
40	∞	511.7	∞	∞	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
41	171.7	∞	992.14	∞	44.43	∞	∞	∞	70.7	450.06	∞	∞	∞	∞	∞	∞	∞	∞	∞
42	∞	∞	∞	760.32	44.43	∞	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞
43	∞	∞	∞	∞	601.78	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
44	∞	∞	∞	∞	∞	571.74	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
45	∞	∞	373.7	44.43	44.43	∞	1187.93	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞
46	∞	∞	∞	∞	∞	∞	∞	345.92	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
47	∞	∞	∞	∞	44.43	∞	∞	∞	774.73	450.46	∞	∞	∞	∞	∞	∞	∞	∞	∞
48	∞	∞	∞	∞	44.43	∞	∞	∞	∞	744.72	∞	∞	∞	∞	∞	∞	∞	∞	∞
49	171.7	∞	∞	∞	44.43	∞	∞	∞	70.7	450.06	1060.6	∞	∞	∞	∞	∞	∞	∞	∞
50	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	1045	∞	∞	44.43	∞	∞	∞	∞
51	∞	70.7	∞	∞	∞	∞	∞	44.43	∞	∞	∞	∞	714.69	∞	∞	∞	∞	∞	∞
52	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	∞	1346.48	44.43	∞	∞	∞	∞
53	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	947.71	∞	∞	∞	∞
54	∞	∞	∞	∞	∞	137.36	∞	∞	∞	∞	∞	∞	∞	∞	∞	1083.43	∞	∞	∞
55	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	1097.85	∞	123.22
56	171.7	∞	96.96	∞	44.43	∞	∞	∞	70.7	450.06	44.43	∞	∞	∞	∞	∞	∞	1159.11	∞
57	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	947.71
58	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	∞	∞	44.43	∞	∞	∞	∞
59	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
60	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	44.43	∞	123.22
61	∞	70.7	∞	∞	∞	∞	∞	44.43	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
62	∞	∞	∞	∞	∞	137.36	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
63	∞	∞	∞	∞	450.46	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
64	∞	∞	∞	∞	450.46	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞

Table A.8: Cost Matrix(Cont'd)

	58	59	60	61	62	63
1	∞	∞	∞	∞	∞	∞
2	∞	∞	∞	∞	∞	∞
3	∞	∞	∞	∞	∞	∞
4	∞	∞	∞	∞	∞	∞
5	∞	∞	∞	∞	∞	∞
6	∞	∞	∞	∞	∞	∞
7	∞	∞	∞	∞	∞	∞
8	∞	∞	∞	∞	∞	∞
9	∞	∞	∞	∞	∞	∞
10	∞	∞	∞	∞	∞	∞
11	∞	∞	∞	∞	∞	∞
12	∞	∞	∞	∞	∞	∞
13	∞	∞	∞	274.72	∞	∞
14	∞	∞	∞	∞	∞	∞
15	∞	∞	∞	∞	∞	∞
16	∞	∞	∞	∞	∞	∞
17	∞	∞	∞	∞	∞	∞
18	∞	∞	∞	∞	∞	∞
19	∞	∞	∞	∞	∞	∞
20	∞	∞	∞	∞	∞	∞
21	∞	∞	∞	∞	∞	∞
22	∞	∞	∞	∞	∞	∞
23	∞	∞	∞	∞	∞	∞
24	∞	∞	∞	∞	∞	∞
25	∞	∞	∞	∞	∞	∞
26	∞	∞	∞	∞	∞	∞
27	∞	∞	∞	∞	∞	∞
28	∞	∞	∞	∞	∞	∞
29	∞	∞	∞	∞	∞	∞
30	∞	∞	∞	∞	∞	∞
31	∞	∞	∞	∞	∞	∞
32	∞	∞	∞	∞	∞	∞
33	∞	∞	∞	∞	∞	∞
34	∞	∞	∞	∞	∞	∞
35	∞	∞	∞	∞	∞	∞
36	∞	∞	∞	∞	∞	∞
37	∞	∞	∞	∞	∞	∞
38	∞	∞	∞	∞	∞	∞
39	∞	∞	∞	∞	∞	∞
40	∞	∞	∞	∞	∞	∞
41	∞	∞	∞	∞	∞	∞
42	∞	∞	∞	∞	∞	∞
43	∞	∞	∞	∞	∞	∞
44	∞	∞	∞	∞	∞	∞
45	∞	∞	∞	∞	∞	∞
46	∞	∞	∞	∞	∞	∞
47	∞	∞	∞	∞	∞	∞
48	∞	∞	∞	∞	∞	∞
49	∞	∞	∞	∞	∞	∞
50	∞	∞	∞	∞	∞	∞
51	∞	∞	∞	44.43	∞	∞
52	44.43	∞	∞	∞	∞	∞
53	∞	∞	∞	∞	∞	∞
54	∞	∞	∞	∞	44.43	∞
55	∞	∞	∞	∞	∞	∞
56	∞	∞	∞	∞	∞	∞
57	∞	∞	∞	∞	∞	∞
58	1293.62	∞	∞	∞	∞	∞
59	∞	616.19	∞	∞	∞	∞
60	∞	∞	1609.54	∞	∞	∞
61	∞	∞	∞	616.19	∞	∞
62	∞	∞	∞	∞	783.15	∞
63	∞	∞	∞	∞	∞	861.61

* The rows correspond to the final destinations and the columns correspond to the visited demand points.

Table A.9: Data of Day 1

Demand Point	Item Type																									
	1	2	3	5	7	8	9	10	11	12	13	14	15	17	18	19	20	21	22	24	25	26	27	28	29	30
1	11	6			1	16		11	15	2		7	2	14	5	5	2				1	2		11	11	13
2	6	1	2	2	5	2		4	5		10	5			10	4	3	3	1		1	3		3	2	
7	28	4	2	3	2	5		21	21				2	2	8			3	2	1	2	1	8	5	12	4
8	10	2			12		40	3	4							1					1					
15		1								1					1										2	
16	4	1						6	4			4	2	2	1	2			3				3	9	3	3
17	1				1																					
27								1				2	2	17	1	1	20				1	1		1	1	1
28	1	1			1																				2	2
34		1									5	2			17	5	4	2				1	8	6	4	4
35						11		4	5													1				
36	2							1				1				1										
47	11	1			11	8		10	13																	
55	1	1				2						1			2											

Table A.10: Data of Day 1(Cont'd)

Demand Point	Item Type																		
	31	32	35	36	37	38	39	40	41	46	47	49	52	54	56	64	65	68	69
1			2	5		24	145	10	8	1						7		3	
2	1	10	1	1			60	16			1								
7	3	2	6	6	102	5	48	69	4	3	2	1	1		7				
8	1		1	2			2	11						1					2
15					1														
16	2	11						42		1									
17				1															
27		15						9	2	2									
28	2			1				5											
34	3	13	3				904	24		1									
35			2			5	81	38											
36				1	1			1											
47				10		27	7										1		
55		2	1	3	5			4											

Table A.11: Data of Day 2

Demand Point	Item Type																								
	1	2	3	4	5	7	8	9	10	11	12	13	14	15	17	18	19	20	21	22	24	25	26	27	28
1	4				5	31	4	1	10	11						2	2								
3	4	2				8	11		4	5				22				20		1	1				
7	23	2			1	3	2	6	25	36						1				1					
10	8	1				7	1		8	8															
11	1	2				1	3	17	10	9							1						1		
19	3	2				1			2	4	3		4				1								
21	11	5		1					2	3	1	20			2	0			2	1	1	2	2	3	1
33	5			1				1	9	9	1		9	54	2	11	8	5			1		5	31	8
34	5	21					1	2	6	8						1				2	1				
37	21	5			2	2	1		6	14							1								
47	9	3	5				15		5	10	2			2			1	10		1					2

Demand Point	Item Type																								
	29	30	31	32	33	34	35	36	37	38	39	40	41	46	47	52	55	56	60	62	63	64	65	66	67
1							1		3		58										1				
3								11	10		1							4							
7					4		3	6	36	21	240					2	2	9					1	2	
10										20															
11								1		13	1	20	2	1								6			
19	5	2					2	1			26	6													
21	3			2		1	1	3	16	15	16	8	4							1					
33	20	16	3	4			2	2	107		49	3		2											1
34							1	1		9	25								10						
37										1	1												1		
47	3	3						5			122	1	4	2	2				5						

Table A.12: Data of Day 3

Demand	Item Type																						
Point	1	2	3	6	7	8	9	10	11	12	13	14	15	17	18	19	20	21	22	23	24	25	26
1			6				7	2	4				24	40			30						
2	17	2			2	3	4	6	6			5						2	1				
8	2	1					11	3	3							1							
16			1							3		4	8	10	1	4	22	2	2			1	2
22	3	2						3	3		15		4	6	7	1	6			1	1	2	3
27	7	11			5	6	3	7	14			4					2						1
33	7	2		1		1	1	7	22								6					1	1
34	43	15			1	3	9	1	4			1				1	1	3	1			1	2
35	1							1	2	4		2				3							
37							8		1		20						2	1					
43	5	3	2		1		4	2	5	1	10	5	6	2	1		14	2					2
46	2	6			1			1	2														2

Demand	Item Type																							
Point	27	28	29	30	31	32	35	36	37	38	39	40	41	42	46	48	49	50	54	63	64	65	67	70
1						8	2																	
2	1			2			2		16			6	4		1		18	2						
8								1				4		219					1					
16	10		8	8	2		1	1				18												
22			2	2	1	12		12	6		150	1												
27						9	1			15	3	1			2						7			
33						6	10	3	4	4		68		5		2				1			2	10
34	9	9	4	2		1		1				1			1		26							
35		2	4	1		22	2		5			6												
37						10		1				14												
43	4	6	4	1	1	7		4	2		43	15	2									1		
46	2		1		1			1	35		16	5			3									

Table A.13: Data of Day 4

Demand	Item Type																							
Point	1	2	3	4	5	7	8	9	10	11	12	13	14	15	17	18	19	20	21	22	24	25	26	27
3	55	7				41		6	6	9										1				
7	12	5					1	20	11	16			12	2	2		3	7			1		2	2
10	12	1			3	1		4	9	8								19			1			
11		2				7	1	11	7	3														
12	1	2				1		2	1	2	1		2			1								2
16	20	3			2	3			6	10						1	2							
21	16	2					4	10	6	3						1							1	
22	24	4				1			6	7											1			
29	2	7				1		4	2	3														
33	5	3				1		17	6	13		5			2			6				1	1	2
34	3	9		1		1	2	7	19	16		10	1				1	4			1		1	
35	18	3	4			16	29	30	16	14		5	2		3		1	6	1			1		2
43	18						1		4	2							1				1			
47	11								3	4													1	
52	15	1				1		2	6	3		5					4	3				2		
58	1	1				2				2	1	20					2							
59	2					4	1	1	2	4	1				4	1	1	3	3				2	

Table A.14: Data of Day 4 (Cont'd)

Demand Point	Item Type																								
	28	29	30	31	32	33	35	36	37	38	39	40	41	42	44	46	47	49	50	54	58	63	65	68	71
3									7	20	71							20	4	1	1				
7		2	2	1			2	3	5	4	50	38		20			1	53	2	2					
10		3			5			3			4			2						2			1		2
11							1				1							20	1						
12	1	2			1		1		26		62	7				3									
16							2	5	18	16	10				21			21	2						
21	2							1						5											
22							1	1	4		9									1					
29								1	21									18							1
33	1	1			1		1	5		10		142				2		25		4					1
34	2	4		1	1	1	2	4		22	5	15				3		20		6		1			
35				1			1	3		31		30		10	30									3	
43								2		5					8			51							1
47	2						1	1	3			21	4					10							1
52					3						1	5	2					50	2						
58									13		1	1													
59					4		1	2			5	6				1									

Table A.15: Data of Day 5

Demand Point	Item Type																		
	1	2	3	7	8	9	10	11	12	13	14	15	17	19	20	21	24	25	26
1	15	4			14	6	7	5										1	
2							5	2						1			1		
7								2		10	3					1			
27	13						4	3						1			1		
33	14			1		13	11	6	1		2	4		2		1			1
34	1	4		1		1	2	6	2		8		2		2		1		3
37	7			1		4	5	2											
43								3											
47	23	1				2	5	3											
55			2			2			1		1			1	1	1			

Demand Point	Item Type																		
	27	28	29	30	32	35	36	37	38	39	40	42	44	46	49	54	58	71	
1	2	1					3	5	3		2		15		44	1			
2							2		3	70		4					1	1	
7					3	1					28			1					
27							1			1	11		6		20				
33		2	2	3	1	2	6		32	10	8			1	20	7		2	
34	2	3	3	2			3			2	30			2					
37							2											1	
43																			
47					6	3	2				20								
55			1					19			8			2	7				

Table A.16: Data of Day 6

Demand Point	Item Type																					
	1	2	3	5	6	7	9	10	11	12	13	14	15	16	17	18	19	20	21	22	24	25
1	2	1																				
2	1										20		4		18			8				1
7	1				1				2	1		1			2	1	1	102		2		1
13	3	4				2		3	3							2	2			1		
14	9	1					1	4	4			1			4	1		1				
21	2		2			1		1	2		30				2						1	
23	1					1	1		1			1				1	1					
24		1				1	12	3	2				2					2				1
29	2	3				1		2	2		5				2		1	5	1			4
30	2	1																				
33	2	1			3			1	3	11		5	26					35	20		2	
35											80							39				
37	1	1	2				1			2		2		2	6		1	3				
40	4	1				1	2	1		1		1				1						
42			1							5							2	48	10			
44	2	1				1			2			1						1				1
52	1	1		1	6	1		1	1				2		2	1	1	5				1
58	1	1		6		1									4	2		10	3		1	
59		3				1					10											

Table A.17: Data of Day 6(Cont'd)

Demand Point	Item Type																					
	26	27	28	29	30	31	32	35	36	37	38	39	40	41	46	48	49	56	57	58	59	61
1									4	5		5	1						1			
2		1		1	2			1	1				1		1							
7	3	5	2	17	2	2		8	5	3	6	5	112									
13	1		5	1	2		1	1		10			8		5							
14	1		2	2	1	1	1	4	4	9		2	5	2	3		30					
21			2					2	6	2	7		4									
23			1						2				6		1							
24		2			1			1	5	16			7		1		10					
29	2		1		2		6	1				1	3		1	1					1	
30									3				11									
33		10	22	30	3	4	3	4	11			75	30									1
35		11		13			20															
37	3	3	1	3	1	4	1	4	4				8	4	2	1		1				
40		2	2	3				1	4	14		2	3					2				
42		8	5	8	4	5																
44		1							2				5		2							
52			3		1				6		5		1							1		
58					1		1	3	7				5		3							
59			2					1	1						1							

Table A.18: Data of Day 7

Demand Point	Item Type																							
	1	2	3	4	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	24	25	26
1	1	2							1	2								1						
3	41	3	3			60			4	4						22			31	10			1	10
6						1																		
7	92	6			90	1			1	7	5		5	10	2	54	2	7	1	4	2		3	4
15	2	1									1						1	1			1			2
16	2	4		1						1	3		2	4				1	1					1
17	1	2				1				1								1					1	
19	1	2							1	3												1		
20	1	1									1						1	1						1
22	11		2		4				2	1								1	11	2				
25	1	1								1								1	1					
28	3								2	2			4				2							
33	3	5	5			1			1	4	2	45	11	1		21	10	3	4	1	4	1	1	3
34	7	4							2	2	5	5	14	20		6		1	10	1		1		2
35	1	1	1						3	5		20	5	2		4		1	3	3			1	1
36	1	3								2			1					1		2	1	1	1	
41		1									1								1					
42	46	5				14			6	10								1						
45	10	3				2	1	2	2	3					1		1		1					
46	1	1							1				1											
47	1					1				1									27					1
53	1	3				1		2		3	2		1				1	2			1			1
56	1					1																		

Table A.19: Data of Day 7(Cont'd)

Demand Point	Item Type																								
	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	45	46	49	50	51	52	53	54
1									1	3		5													
3	2	3				25			1	2				3	14										
6									1	1	15								1						
7	21	5	7	7	6	2			11	7	2	30		14	12		2							1	6
15	1								1	1	4		15	3		1			1						
16	1	1	1	1		10			2	1				3											
17	3		1		1	1			1	7	2		10	5					3	20		1	1		
19	2	3							5	3				9			1			4					1
20			1	3						3	6		5							2		1			
22	1	5							3	4		5	1	28			1								
25		1		3			1		1	2	4		28	5					1	14					
28	2	1			1					3	11		20	4					4	12					
33	1	1	9	4	7	1		2	16	7	26		6	33	2		2		2	4					
34	1	5	9	1		5			4	4	13		23	6			2		3	2					
35	6	6		4	3				2	4				24											
36		2							3	4			1	9			1		1	12					
41										1	1			3											
42												23	144			10	1			60	4				
45		3				5			1	1	25		30	11		1	1		2	10					
46		1							1	1	5	1		1			1								
47	1	9		3	1	11		1	3	1				8											
53	3	2	1	1	2	1			1	2			15	6				4	2						
56										2			10												

Table A.20: Data of Day 8

Demand	Item Type																							
Point	1	2	3	4	6	7	8	9	10	11	12	13	14	15	17	18	19	20	21	22	23	24	25	26	
1			1			1				3		10	3					1					2		
7	8	12						1	41	36															
16	13	1	2						4	7		20					2							2	
31	2															1	1								
33	52	9						4	21	27	3	40		6			1		2				2	2	
34	28	8			4	6		1	11	12			18	12	7		6	4		2		1	2	3	
35	30	5	8	1	2		4		18	27	5	5	18	22	4		2	1	2		5	1	2	1	
39	1	1				1			1	1								1							
42		1	4						4	2			3	12	10		3			2					
47	10								12	7							1								

Demand	Item Type																							
Point	27	28	29	30	31	32	33	35	36	37	38	39	40	41	42	43	46	48	49	50	52	53	58	67	71
1		6				10			3				3			2									
7								3	9	43	55								30				1		
16	1	3						2		2	13	90	5	4			4	3							
31								1	1	1			10												
33	21	3	15	13	4		4	6	3		28	30	44	6			1					5		2	
34	6	11	12	4	4			4	4	20	14	9	53				1	1	42		1				
35	7	1	16	5	2			8	7	6	24		21			1	1			4				1	
39		1	1			1			2			1	2												
42	11	3	1	1		1		2	1				27	2			3								2
47								1		8	29				12					5					

Table A.21: Data of Day 9

Demand	Item Type																		
	1	2	4	7	8	9	10	11	12	13	14	15	17	18	19	20	21	25	26	
2	1	1	1				3	5			6		2		1	3	1	1		
16	1						5	7			1					13				
33	5	3		1			13	15	10	5	14			3	3	1		1	2	
34	9				4		14	14			22									
35	12	7		24	4		10	11	3		14	4	14		3	22	2	1		
37	1	2				1	1	2	1		4				2			1		
42	2	3		35			12	9												

Demand	Item Type																		
	27	28	29	30	31	32	35	36	37	38	39	40	46	47	48	49	50	56	60	67
2	2	2	1				3	1	17	5	30	5	1	1		11				
16			2		1			2	2		20	2				4			8	
33		1	15		5	1	10	4	2			26		1	8	9	2			
34		22		1	3		2					67	2	2	1		7			1
35			10	4	11		3						2	3		5				
37	3		3		1		1	1	12		1	6			1	2		4		
42							1									25			5	

Table A.22: Data of Day 10

Demand	Item Type																				
Point	1	2	3	4	5	7	8	10	11	12	13	14	15	17	19	20	21	22	24	25	
7	2	2							4				2		3	8	1	2			
8	1	2				1		1				13	8	10	3	2	1				
16	21	3	2			21		3	8							3					
21	11	4	1			1		4	7				2	6	1	3		1	1		
33	20	6						2	6		5	11		2	16	1					
34	3	3			5	2	5	10	26						1	7				2	
35		3	1	1				1	2	3	5	2	2		1					1	
37	5	2	2			1	2	3	5	5	5			4		6	1		1	1	

Demand	Item Type																				
Point	26	27	28	29	30	31	32	34	35	36	37	38	39	40	41	43	46	47	48	49	50
7		7	5	4	2		2		3	5				27				1			
8		3	25	11	4	2	3		1	3				1						2	
16	2	1				2			2	3				22							
21	2	1	1	2		1	1			4	4	2		8	2	1				10	1
33		1		2		4	11		4	22				2	8	1	6		2		
34				2			1	1	2	6	7	24		6						11	1
35		4		3	1	2			1	16				11				1			
37	3		1			2	2		1	3	11		1	13	2			1		3	6

Table A.23: Data of Day 11

Demand Point	Item Type																							
	1	2	3	4	6	7	8	9	10	11	12	13	14	15	17	18	19	20	21	22	24	25	26	27
1	5	3							4	5	2	5	3				2						1	
2	6	3							5	7	1							16				1		2
6		1				1		2	2															
7	8	7		1		1	4		12	16		5	2				1			2			3	
8	7	3		1		8	3		2	2							2	1						
14	3	2	2			1			3	4	1				2	1		2						
15	1	1							2	1														
16		1	5						5	7			4	6	14		1							2
22	1	2				1	1		4	4			4	2	2			1			1	1	1	1
23	3	2								2	1		2				2				1			
28	1	1							1	2			2					2					1	
29	2	1							3	4											1			
31	1	1							2	2	3		2											
33	51	8	2	1		1	2	2	27	29	10	10	29	4	5		4	10	2	1		1	2	10
34	8	10		1		3	3		15	22	5		6				4	1			1	2	1	1
35	2	7				12		2	11	19	2		8	2	1	1		1		3		1	2	5
37	2			1		1			2	3	1		1											
39	1	1							1	1			1			1					1			
42	1	4			2				4	7								20		1				
44	2	1							3	2	4		3					1						
47	4	5							3	8			1	2	4									1
48	2	2				1					1							1			1			
54	1	1				2			1	1								2						
56	3								1	1														
58	8				5	2			4	4	2			2	8		1	5	1			1		
62	1	2				2			1				1				1				1			
63	1								2	2		5							1				2	2

Table A.24: Data of Day 11 (Cont'd)

Demand	Item Type																							
Point	28	29	30	31	32	33	34	35	36	37	38	39	40	41	43	46	47	48	49	50	54	58	60	67
1		3							2	16		5	7		1	1	1		9	1				
2	2				1				6			50	4				1	1	2	2				
6					1														3					
7	3	2	1						8	43	18	2				1			11	4			2	
8			2		1			1			15		11			3		3	160	2				
14		1	1						2			1	8		1				4				5	
15										6			1			1			4					
16		9	2							23	50			2				5	60					
22			1		1				3	3		4	2	2		1				1				
23		1	3	2					1				1		1	2		1					1	
28	1	2	2	2	2			1		4			4			2			4					
29									1	11	16								2				1	
31		4			2					1		5	2			1								
33	12	32	8	3	1		1	1	8	15	13		29				1	1	12	1	2	1		4
34	4	8	4		3	4		2	8	1	5	4	37			7			56	2				
35	5	13	5		2				11	2	20		15			5		1	9	3	1			2
37		1	1		1				1	1			1			3			4				1	
39												7							1					
42		5						1	2			190								1				
44		4						1		2			11			1				1				
47				2					3		5	32	3			1				1		1		
48		1										27											2	
54			1			1			2	21			1			5		1	5					
56																			4					
58	1		1	1	1			2	2	5		3	7	2		1			61			1		
62	1							1	1	2			2			1			10					
63		1	1						2			4				1		3	4					