A LAGRANGEAN HEURISTIC FOR THE TWO-STAGE MODULAR CAPACITATED FACILITY LOCATION PROBLEM

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

SELİM SEVİNÇ

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
INDUSTRIAL ENGINEERING

MAY 2008

Approval of the thesis:

A LAGRANGEAN HEURISTIC FOR THE TWO-STAGE MODULAR CAPACITATED FACILITY LOCATION PROBLEM

submitted by **SELİM SEVİNÇ** in partial fulfillment of the requirements for the degree of **Master of Science in Industrial Engineering, Middle East Technical University** by,

Prof. Dr. Canan Özgen ____________________
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Nur Evin Özdemirel ____________________
Head of Department, **Industrial Engineering**

Assist. Prof. Dr. Sedef Meral ____________________
Supervisor, **Industrial Engineering Dept., METU**

Examining Committee Members:

Assoc. Dr. Canan SEPİL ____________________
Industrial Engineering Dept., METU

Assist. Prof. Dr. Sedef MERAL ____________________
Industrial Engineering Dept., METU

Assist. Prof. Dr. Serhan DURAN ____________________
Industrial Engineering Dept., METU

Prof. Dr. Serpil EROL ____________________
Industrial Engineering Dept., Gazi University

Bora KAT, MS ____________________
Industrial Engineer, TÜBİTAK

Date: ____________________

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name: Selim SEVİNÇ

Signature :

ABSTRACT

A LAGRANGEAN HEURISTIC FOR THE TWO-STAGE MODULAR CAPACITATED FACILITY LOCATION PROBLEM

Sevinç, Selim
M.S., Department of Industrial Engineering
Supervisor: Assist. Prof. Dr. Sedef Meral

May 2008, 157 pages

In this study, a Lagrangean heuristic based on Lagrangean relaxation and subgradient optimization is proposed for the two-stage modular capacitated facility location problem. The objective is to minimize the cost of locating and operating plants and warehouses, plus the cost of transporting goods at both echelons to satisfy the demand of customers. The difference of our study from the two-stage capacitated facility location problem is the existence of multiple capacity levels as a candidate for each plant in the problem. Each capacity level has a minimum production capacity which has to be satisfied to open the relevant capacity level. Obviously, a single capacity level can be selected for an opened facility location. In the second echelon, the warehouses are capacitated and have unique fixed and variable costs for opening and operating. Multiple sourcing is allowed in both transportation echelons.

Firstly, we develop a mixed integer linear programming model for the two-stage modular capacitated facility location problem. Then we develop a Lagrangean heuristic to solve the problem efficiently. Our Lagrangean heuristic consists of three main components: Lagrangean relaxation, subgradient optimization and a primal heuristic. Lagrangean relaxation is employed for obtaining the lower bound, subgradient optimization is used for updating the Lagrange multipliers at each iteration, and finally a three-stage primal heuristic is created for generating the upper bound solutions.

At the first stage of the upper bound heuristic, global feasibility of the plants and warehouses is inspected and a greedy heuristic is executed, if there is a global infeasibility. At the next stage, an allocation heuristic is used to assign customers to warehouses and warehouses to plants sequentially. At the final stage of the upper bound heuristic, local feasibilities of the plants are investigated and infeasible capacity levels are adjusted if necessary.

In order to show the efficiency of the developed heuristic, we have tested our heuristic on 280 problem instances generated randomly but systematically. The results of the experiments show that the developed heuristic is efficient and effective in terms of solution quality and computational effort especially for large instances.

Keywords: Network Design, Modular Capacity, Facility Location, Lagrangean Relaxation, Lagrangean Heuristic, Subgradient Optimization.

ÖZ

İKİ SEVİYELİ MODÜLER KAPASİTELİ TESİS YERLEŞİMİ PROBLEMİ İÇİN BİR LAGRANGE SEZGİSELİ

Sevinç, Selim
Yüksek Lisans, Endüstri Mühendisliği Bölümü
Tez Yöneticisi: Y. Doç. Dr. Sedef Meral

Mayıs 2008, 157 Sayfa

Bu çalışmada iki seviyeli modüler kapasiteli bir tesis yerleşim problemi için Lagrange gevşetimi ve altgradyan optimizasyonu tekniklerine dayanan bir Lagrange sezgiseli sunulmuştur. Problemin amacı, fabrika ve depoların yerleştirilmesi ve işletilmesi ile müşterilerin taleplerini karşılamak üzere ürünlerin fabrikalardan müşterilere taşınması sonucu ortaya çıkan maliyetlerin toplamını asgariye indirmektir. Çalışmamızın kapasiteli tesis yerleşimi probleminden farkı, her fabrika yeri seçeneği için birden fazla kapasite seviyesi seçeneğinin bulunmasıdır. Ayrıca her kapasite seviyesinin açılabilmesi için sağlanması gereken farklı bir minimum üretim kapasitesi bulunmaktadır. Açılacak bir tesis mekanı için sadece bir kapasite seviyesi belirlenebilir. Problemin ikinci seviyesinde, fabrikalardan farklı olarak her depo için sadece tek bir kapasite seviyesi vardır ve açılıp işletilebilmesi için bir sabit ve

değişken maliyete sahiptir. Her iki seviyede de çok kaynaklılığa izin verilmiştir; yani bir depo birden fazla fabrikadan ürün alabilirken, aynı şekilde ikinci seviyedeki bir müşterinin talebi, gerekli durumlarda birden fazla depodan karşılanabilir.

Bu çalışmada söz konusu iki seviyeli modüler kapasiteli tesis yerleşimi problemi için bir karmaşık tamsayılı programlama modeli geliştirilmiş ve sonra bu problemi etkin bir şekilde çözmek için bir Lagrange sezgiseli önerilmiştir. Bu Lagrange sezgiseli; Lagrange gevşetimi, altgradyan optimizasyonu ve üç-aşamalı bir ana sezgiselden oluşmaktadır. Lagrange sezgiselinde, Lagrange gevşetimi alt sınırı bulmakta, altgradyan optimizasyonu her yinelemede Lagrange çarpanlarını güncellemekte ve üç-aşamalı sezgisel de problemin üst sınırını bulmakta kullanılır.

Üç-aşamalı sezgiselin birinci aşamasında, fabrikaların ve depoların toplam fizibiliteleri kontrol edilmekte ve toplam fizibilitenin olmaması durumunda bir açgözlü sezgisel çalıştırılmaktadır. İkinci aşamada ise tahsis sezgiseli önce müşterileri depolara ve sonra da depoları fabrikalara atamaktadır. Üst sınır sezgiselinin son aşamasında, her fabrikanın lokal fizibilitesi kontrol edilmekte ve yerel fizibilitesi olmayan fabrikaların kapasite seviyelerini ayarlamaktadır.

Geliştirilen Lagrange sezgiselinin etkinliğini göstermek için rassal fakat sistematik bir biçimde oluşturulan 280 test problem örneği yaratıldı ve sezgisel yaklaşım bu örnekler üzerinde denendi. Deneylerin sonuçları, geliştirilen sezgiselin çözüm kalitesi ve hesaplama güçlüğü açısından özellikle büyük problemlerde verimli ve etkili olduğunu gösterir.

Anahtar Kelimeler: Serim Tasarımı, Modüler Kapasite, Tesis Yerleşimi, Lagrange Sezgiseli, Lagrange Gevşetmesi, Altgradyan Optimizasyonu.

To My Mom…

ACKNOWLEDGMENTS

I would like to express my deepest appreciation to my supervisor, Assist. Prof. Dr. Sedef Meral for her invaluable support and patient guidance throughout the study.

I particularly would like to thank Dr. Tolga Bektaş from University of Southampton, School of Management, for all his precious advices, comments and suggestions throughout the study.

I also would like to specify my thankfulness to my cousin Hande İrfan for her careful revision and reduction of the thesis report.

I would like to express my deepest gratitude and my love to my father for his never-ending patience, support and encouragement.

Also, I give my thanks to my dear friends Alper Karaduman, Şakir Karakaya and Fatih Altınel for their continuous support.

Finally, I would like to thank Prof. Dr. Bilgin Kaftanoğlu and Prof. Dr. Ömer Saatçioğlu, whose encouragement and motivation considerably guided me to start graduate education in METU IE.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

In the `80s, rough competition in global markets, the introduction of products with short life cycles, and high expectations of customers have forced companies to discover new manufacturing technologies and strategies that allow them to reduce costs and be more competitive in global markets. Strategies such as just-in-time manufacturing, total quality management, kaizen and similar became popular and many resources were invested in implementing these strategies.

In recent years as Simchi-Levi et al. (1999) stated, it has become clear that many companies have reduced manufacturing costs as far as practically possible. Many of these companies focus on and invest in their supply chains since they discovered that creating an effective and efficient supply chain is the next step to increase the profit and market share. Moreover, recent developments in communications and transportation technologies have motivated the evolution of the supply chain. The information and communication systems have been widely developed to provide access to whole information at all stages of the supply chain. Finally, the design of the new transportation modes and significant improvement of the existing ones increase the complexity of the logistic systems.

A supply chain consists of suppliers, plants, warehouses, distribution centers, retailers as well as the raw materials, work-in-process inventory, and finished goods that flow between the facilities. In a supply chain, raw materials are obtained from suppliers, and then goods are produced in one or more plants. Produced goods are dispatched to warehouses for intermediate storage and finally served to retailers or directly to the customers. In order to improve service quality as well as reduce costs, more effective supply chain strategies and the interactions between the components of the supply chain have to be taken into consideration.

Developing and implementing effective strategies in supply chain is called “supply chain management” which focuses on the efficient integration of suppliers, manufacturers, warehouses and stores and encompasses the firm’s activities at many levels, from the strategic level through the tactical to the operational level. Simchi-Levi et al. (1999) also define supply chain management as a set of approaches utilized to effectively integrate suppliers, manufacturers, warehouses and stores, so that merchandise is produced and distributed at the right quantities, to the right locations, and at the right time, in order to minimize system-wide costs while satisfying service level requirements.

The system-wide supply chain costs include the costs of movement, storage, and management of all type of goods both within and between the components of the supply chain. However, inefficient supply chains have additional costs besides these costs stemming from superfluous inventories, excessive transported items, etc. Therefore, there are many opportunities to cut down costs in the less efficient supply chain. In fact, Simchi-Levi et al. (1999) state that experts believe that grocery industry can save about 10% of its annual operating cost by using more efficient supply chain strategies.

The supply chain is composed of a wide range of organizational activities from the strategic to the operational level. The strategic level deals with the decisions that have a long lasting effect on the firm, covers the decisions concerning the number, location and the capacity of manufacturing plants and warehouses. Tactical level deals with the decisions which have a lifetime of three months to a year, and covers purchasing and production decisions, inventory policies and the decisions concerning transportation. The operational level refers to the daily decisions such as scheduling, routing and truck loading.

Distribution network design problem is one of the key issues of supply chain management that deals with determining the best location and capacities of the facilities and the distribution pattern of the goods in order to satisfy the demands of the customers. Distribution network design focuses on the issues related to plant, warehouse and retailer locations. The decisions given within the scope of the distribution network problem are the combination of various strategic and tactical level supply chain decisions, because their implications are significant and long lasting. These decisions may be required due to the changes in the demand patterns, new markets, or previously given wrong decisions. Incorrect decisions of such kind may cause irreversible loss of the overall efficiency of the firm. Therefore, in order to design an efficient strategic plan, it is necessary to plan carefully before making these decisions.

The distribution network design problems are defined within the context of a number of production plants that are supplying warehouses, which, in turn distribute these goods to the customers based on their demand. It is required to select the best set of plants and warehouses from a set of potential alternatives, and determine the amount of product flow and inventory among them so that the objective of the company is optimized.

The choice of the distribution network can be used to achieve a variety of supply chain objectives ranging from low cost to high responsiveness. Distribution means the movement, transfer, or disbursement of goods from the point of production to the point of consumption. In supply chain networks, distribution occurs between every pair of stages; raw materials and components are moved from suppliers to manufacturers, and final products are moved from the manufacturer to the consumer. Distribution is one of the critical points of the overall profitability of a firm, because it directly affects both costs and responsiveness.

Beside the distribution decisions, using warehouses in distribution also affects the costs and responsiveness of a company. According to Amiri (2006), an important strategic issue related to the design and operation of a physical distribution network in a supply chain system is the determination of the best sites for intermediate stocking points, or warehouses. The use of warehouses provides a company with flexibility to respond to the changes in the marketplace and can result in significant cost savings due to economies of scale in transportation or shipping costs.

The network design problems belong to the strategic level of decision making. Jang et al. (2002) mention that the decisions made for network design determine the number and the locations of raw material suppliers, manufacturing plants and intermediate inventory warehouses, select the distribution channel from suppliers to customers and identify the transportation volume among the distributed facilities. Numerous papers including Pirkul et al. (1998), Amiri (2006), and Jang et al. (2002) have dealt with the design problem of supply chain networks.

In this study, we have considered the problem of designing a distribution network that involves simultaneously determining both the best sites of the plants and the distribution facilities and also the best strategy for distributing the products from the plants to the warehouses and from the warehouses to the customers. A common objective in designing such a distribution network is to determine the least cost of the system such that all the demands of all the customers are satisfied. This usually involves making trade-offs between the cost components of the system such as opening and operating cost of the plants and warehouses as well as inbound and outbound transportation costs.

Our aim in this study is to propose solution approaches for a special type of distribution network design problem (DNDP), which is called the two stage modular capacitated facility location problem (TSMCFLP). For this purpose, we have first reviewed the literature covering both the exact and heuristic methods for similar problems and proposed Lagrangean Relaxation and decomposition based heuristics for our TSMCFLP environment. Finally, the proposed heuristics are tested on a number of randomly generated test problems in several instances and the results are compared to the optimal solutions of the problem.

The ensuring exposition is structured as follows:

In Chapter 2, a review of DNDP literature is presented and related work is classified with respect to the capacity limitation, number of echelons and product variety.

In Chapter 3, the problem context is described and a mathematical formulation of the TSMCFLP is presented. The parameters, decision variables and the constraints are explained in detail. Main assumptions considered throughout the study are also presented in this chapter.

In Chapter 4, the solution procedures for TSMCFLP are defined. Before that the Lagrangean relaxation and the subgradient optimization methods and also the implementation of these methods to our problem are discussed. Then the developed heuristic solution for our TSMCFLP environment is explained in detail.

In Chapter 5, the experimental design, data generation for the test problems and the problem instances are defined. The performance measures which have been used to evaluate the quality of the solution are presented and the results of the experiments are discussed.

Finally in Chapter 6, the conclusions are stated along with the suggestions for further research.

CHAPTER 2

REVIEW OF THE RELATED LITERATURE

The distribution network design problem (DNDP) consists of determining the best way to transfer goods from the supply to the demand points by minimizing the overall costs through determining the structure of the network that is relevant to the number and location of different types of facilities. DNDP has long been studied in the operations research literature. DNDPs cover a broad range from the simple single-commodity deterministic linear models to the multi-commodity nonlinear stochastic versions. Solution approaches include heuristics, optimization, simulation and some rather recent and innovative hybrid procedures which integrate multiple solution approaches.

A great deal of research exists in developing the mathematical models and the exact solution techniques for the DNDP which date back to 1970s. An important problem class in this area is the facility location problem. In the broadest sense, the term 'facility' refers to plants, warehouses, distribution centers, retailer outlets, schools, hospitals, etc. In this type of problem, a finite number of candidate location sites for the facilities are given. The problem consists of opening facilities on the set of candidate sites in such a way that the sum of the fixed costs of opening facilities and the variable costs of satisfying the known customer demands from the facilities is minimized.

The facility location problem can be classified into different subcategories depending on the characteristics of the problem that is dealt with. In our literature review section, the main focus is on the heuristics and the exact solutions; hence the considerable work on the simulation studies is neglected. The distribution network location models can be broadly classified according to:

i. Distribution network (capacitated or uncapacitated)
ii. Number of echelons or levels (single or multiple)
iii. Number of commodities (single or multiple)
iv. Cost structure (linear or nonlinear)
v. Planning horizon (static or dynamic)
vi. Pattern of demand (deterministic or stochastic)
vii. Additional side constraints (e.g. single-sourcing, choice from a candidate subset, client matching)

In our literature review, we have classified the studies according to the first two aspects listed above. However, the other five aspects of the problem are examined for each study. Besides, the algorithms in these studies and their solution effectiveness are discussed.

Previous research studies on the facility and demand allocation problems are well surveyed by Brandeau and Chiu (1989) and Avella et al. (1998) among others.

Avella et al. (1998) present the state of art and the future trends in the location analysis. The issues discussed include the modeling aspects in discrete location theory, the influence of the distance function, the relation between discrete, network and continuous location, heuristic techniques, the state of technology and undesirable facility location.

Vidal & Goetschalckx (1997) present a literature review of strategic production-distribution models. The review studies the optimization models and focuses on the identification of the relevant factors to be included in the formulations, and the specific characteristics of the solution methods.

Cornuejols et al. (1991) compare the approaches proposed in the literature for the capacitated plant location problem. The comparison is based on the theoretical and computational results, with the main emphasis placed on the relaxations.

2.1 Uncapacitated Facility Location Problems

In the uncapacitated facility location problem, each facility is assumed to have no limit on its capacity. In this case, due to the uncapacitated structure of the model, each demand point is supplied by only one facility that has the least transportation cost.

2.1.1 Simple (Uncapacitated, Single-Echelon) Facility Location Problems (SFLP)

The simplest case in the class of location models is SFLP that has a single commodity with unlimited capacity, a single transportation echelon and linear costs. This type of problem is based on the tradeoff between the fixed and variable costs.

Numerous approaches have been proposed for solving the SFLP. The earliest attempts were through the use of heuristics. The Kuehn and Hamburger's (1963) "pairwise interchange or bump and shift routine" is a kind of generic standard against which the following algorithms were compared. An early attempt to optimize the SFLP is a branch-and-bound procedure proposed by Efroymson et al. (1966). Khumawala (1972) has also made notable contributions to the efficient solution of the SFLP principally in the development of the efficient branching rules for the branch-and-bound procedure.

Erlenkotter (1978) reports impressive computational success with the SFLP using a dual based procedure, called the dual ascent method which is based on linear programming dual formulation of the problem. Instead of solving the problem directly, Erlenkotter solves a condensed dual in which the dual of the SFLP is reduced to a form involving only the multipliers corresponding to the constraints.

Dual ascent procedure starts with an initial dual solution and adjusts the multipliers incrementally in such a way that complementary slackness violations are reduced. The algorithm terminates when no further adjustments are possible. A simple ascent and adjustment procedure proposes optimal dual solutions, which in turn often correspond to optimal integer primal solutions. If not, a branch-and-bound procedure completes the solution.

Van Roy and Erlenkotter (1982) formulate a particular dynamic facility location problem, where time-staged establishment of facilities at different locations is considered. Opening of new facilities and closing of existing ones is allowed. For solving the problem, a branch-and-bound procedure incorporating a dual ascent method that extends the approach developed by Erlenkotter (1978) for the static uncapacitated problems is proposed.

Klincewicz et al. (1986) describe a branch-and-bound algorithm for a generalization of the classic uncapacitated facility location problem (UFLP), in which customers need multiple products. They call the new problem as the multi-product uncapacitated facility location problem. The lower bound of the problem is obtained by solving an uncapacitated facility location problem for each product using a dual ascent procedure. They also describe a heuristic branch-and-bound procedure in which the solutions to the subproblems at a given node may not generate a true lower bound. Feasible solutions are generated based on the superposition and a drop heuristic.

Klincewicz and Luss, (1987) propose a dual based algorithm for the multi-product uncapacitated facility location problem. In this problem, in addition to the fixed cost for opening a facility, there is an added fixed cost for handling a particular product. The dual ascent and dual adjustment procedures generate a feasible solution to the dual of the linear programming relaxation of the problem. This procedure can be used either as a stand-alone heuristic or can be incorporated with a branch-and-bound heuristic.

Guignard (1988) proposes a model to strengthen the separable Lagrangean relaxation (equivalent to LP relaxation) of the uncapacitated plant location problem by using Bender's inequalities generated during a Lagrangean dual ascent procedure. These inequalities can be used as knapsack constraints in the Lagrangean relaxation.

Korkel (1989) modifies the primal-dual version of Erkenkotter's exact algorithm to get an improved procedure which is called the multi-ascent method. Experiments show that the new method significantly improves the empirically verified average case efficiency of the dual ascent algorithm, especially for large-scale instances.

Klose (1998) presents a branch-and-bound algorithm for solving an uncapacitated facility location problem with an aggregate capacity constraint. The algorithm is based on Lagrangean relaxation and subgradient optimization for the lower bounds and a simple Lagrangean heuristic to produce feasible solutions and penalties to reduce the problem size.

Holmberg (1999) studies the exact solution methods for the uncapacitated facility location problems where the transportation costs are nonlinear and convex. In order to enable the formulation of the problem as an extended linear pure zero-one location model, an exact linearization of the costs is made. In order to obtain an exact solution of the problem, a dual ascent and adjustment method within a branch-and-bound framework is used and this solution is compared to a modified version of Benders' decomposition, which has been found to be the most efficient in this class.

Gourdin et al. (2000) study a particular type of the uncapacitated facility location problem where two clients allocated to the same facility are matched. The allocation cost is calculated as either the cost of a return trip between the facility and the client, or the length of a tour containing the facility and the two clients. They develop a greedy heuristic and a branch-and-cut algorithm, and describe several separation algorithms. The computational results confirm the efficiency of the proposed approach.

Ghosh (2003) develops a neighborhood search heuristic based on tabu search and complete local search with a memory to solve large instances of the uncapacitated facility location problems.

Barahona and Chudak (2005) investigate the solution of large scale instances of the capacitated and uncapacitated facility location problems. They develop a heuristic to approximately solve the problems, providing a feasible solution together with a lower bound on the optimum. The method is based on the volume algorithm to solve the linear programming relaxation to the problem, together with the variants of randomized rounding to obtain feasible solutions. The volume algorithm is an extension of the subgradient method introduced by Held et al. (1974) to produce primal solutions. However, this algorithm is devised for the simplest facility location models of the single echelon, single commodity, uncapacitated models.

Resende and Werneck (2006) present a hybrid multi-start heuristic for the uncapacitated facility location problem based on a very successful method that was originally developed for the p-median problem by them. The results show that the developed algorithm is the best algorithm found so far for obtaining near-optimal or optimal solutions for the large, heterogeneous uncapacitated facility location problem.

Beltran-Royo et al. (2007) develop a new approach for solving uncapacitated facility location problems, based on semi-Lagrangean relaxation (SLR) that has been introduced by Beltran et al. (2006) for solving p-median problems. They propose two different approaches for solving the Lagrangean dual problem which are proximal ACCPM and dual multi-ascent method, and show that using SLR has some advantages for the uncapacitated facility location problems. They can solve many unsolved problem instances in the literature; however it is proved that the algorithm is not as efficient as the recent meta-heuristics like hybrid multi-start heuristic. Nevertheless, it provides better solutions compared to the other Lagrangean based heuristics developed for the uncapacitated facility location problems due to the additional constraint.

2.1.2 Uncapacitated, Multi-Echelon Facility Location Problems

An extension of the SFLP, in which two echelons of facilities are involved, is called the two-echelon uncapacitated facility location problem. In this problem, the deliveries are made from the first-echelon uncapacitated facilities such as plants or depots to the customer via the second-echelon uncapacitated facilities such as the warehouses. The objective is to determine the number and the location of facilities in the echelons, the flow of products between the facilities in different echelons and the assignment of the customers to the facilities in the second echelon.

Ro and Tcha (1984) develop a branch-and-bound algorithm for solving the two-level uncapacitated facility location problem with some side constraints where commodities are delivered from plants to customers either directly or indirectly via warehouses. Side-constraints in this study represent the adjunct relationship of some warehouses to a certain plant. The proposed branch-and-bound procedure employs a set of new mechanisms for lower bounds and simplifications which are obtained by exploiting the submodularity of the objective function and the special structure of the side-constraints.

Narula and Ogbu (1985) formulate and solve an uncapacitated two-level hierarchical location-allocation problem where a certain number of first and second level facilities are to be located, with the objective of minimizing the total weighted travel distance. The solution method is based on Lagrangean relaxation and subgradient optimization.

It is interesting that there is not any research about the uncapacitated, multi-echelon facility location problem between 1985 and 2007. Actually, in their two studies Tragantalerngsak et al. (1997) and (2000) deal with a problem that has uncapacitated plants, but they call their study as the two-echelon capacitated facility location problem because of the existence of the capacitated warehouses. Therefore, these two studies are reviewed within the multi-echelon capacitated facility location problem class.

Marin (2007) presents a mixed integer formulation based on twice indexed transportation variables, which reduce the number of coefficients and variables in the model and performs an analysis of several Lagrangean relaxations for the two-stage uncapacitated facility location problem. The upper bound of the problem is obtained by solving two separate uncapacitated facility location problems for each echelon in every 10 iterations.

2.2 Capacitated Facility Location Problems (CPLP)

When each facility has a limited capacity, the problem is referred to as the capacitated facility location problem. According to Cournuejols et al. (1991), the state of art of solving the capacitated facility location problem is unknown in the sense that no one has known a heuristic that always works well in practice. Part of the reason is that the linear programming relaxation is known not to be tight both theoretically and experimentally.

2.2.1 Capacitated, Single Echelon Facility Location Problems

In the last two decades, many papers have been proposed on solving the capacitated plant location by using both approximate and exact solutions like Barcelo and Casanovas (1984), Aikens (1985), and Beasley (1993). Researchers have worked on developing both heuristic and exact algorithms. Exact algorithms can solve medium sized problems within reasonable computer time while heuristics are required to solve realistic sized problems.

Cournuejols et al. (1991) studied the relaxations of the capacitated facility location problem from three angles: inequalities among the corresponding bounds, computational experiments and complexity. Based on the computational results, they recommend the Lagrangean relaxation based heuristics for solving large scales of capacitated facility location problems.

Beasley (1993) presents a framework for developing Lagrangean heuristics, based upon Lagrangean relaxation and subgradient optimization for the location problems. The computational results are given for four different location problems which are p-median, uncapacitated warehouse location, capacitated warehouse location with and without single-sourcing constraints. Beasley reports that the developed framework is robust and gives good quality solutions for each of the location problems.

Lee (1993) studies multi-commodity, multi-type facility location problem with a choice of various facility types where several different products are required by customers. Each facility type offers a different capacity on a particular product with different fixed set-up costs. In the formulation of the studied problem, in addition to the fixed cost of opening a facility, there is an added fixed cost incurred if an open facility is equipped to handle a particular product. The solution algorithm unifies Benders' decomposition and Lagrangean relaxation into a single framework that involves successive solution to a Benders' primal subproblem and a Lagrangean dual subproblem.

Jayaraman (1998) studies the capacitated warehouse location problem that involves locating a given number of capacitated warehouses in order to satisfy customer demands for different products. A Lagrangean relaxation-based procedure is developed for solving the logistics design problem. Then an effective heuristic solution procedure that is used in conjunction with the Lagrangean problem is discussed. The computational results on a wide variety of problems are reported and these results indicate that the feasible solution procedure consistently provides stable solutions to the problem. Moreover, the heuristic performs well in terms of both approximations to optimality and solution times regardless of the problem structure.

Bornstein and Campelo (2004) propose an Add/Drop heuristic algorithm for the capacitated facility location problem based on the dominance criteria between the fixed and variable costs. The computational results show that it is able to tackle large scale problems obtaining almost always near optimal solutions at very low cost.

Klose and Görtz (2007) present a branch-and-price algorithm for the capacitated facility location problem. The approach is based on relaxing the demand constraints in a Lagrangean manner, and a hybrid mixture of subgradient optimization and a weighted decomposition method is applied for solving the master problem. Furthermore, the column generation procedure is embedded in a branch-and-price algorithm for computing optimal solutions to the CFLP. The proposed branch-and-price method usually performs better than a branch-and-cut method (CPLEX) based on the LP relaxation of the original problem formulation as well as a branch-and-bound method based on Lagrangean relaxation and subgradient optimization.

Sankaran (2007) presents two sets of results pertaining to the solution of capacitated facility location problems that are large, especially with regard to the number of customers. One set of results relates to customer aggregation, while another set of results concerns the judicious selection of variable-upper-bounding (VUB) constraints to include in the initial integer-programming formulation.

2.2.2 Single-Source, Capacitated Facility Location Model

The single-source, capacitated facility location problem is a special case of the capacitated facility location problem in which it could only be supplied to each customer from exactly one facility. This problem has been studied by several authors, including Barcelo and Casanovas (1984), Sridharan (1993), Klincewicz and Luss (1986) and Beasley (1993).

Barcelo and Casanovas (1984) propose a Lagrangean relaxation heuristic where the demand constraints are dualized. The heuristic consists of two stages: plant selection and assignment. The plant selection stage terminates when the total capacity of the open plants just exceeds the total demand. Then either an interchange procedure is adopted to select an improved list of open plants or the assignment stage starts where the assignment of customers to open plants is performed by the regret heuristic.

Barcelo et al. (1991) present an algorithm for the capacitated plant location problem based on a formulation obtained by adding auxiliary variables, which couples the allocation variables. By relaxing the coupling constraints two separate subproblems are obtained, which is the basis for the variable splitting approach.

Sridharan (1993) considers the Lagrangean relaxations of the capacitated plant location problem with the single-source constraints. The paper proposes a Lagrangean heuristic, based on the Lagrangean relaxation, subgradient optimization and a primal heuristic to obtain a feasible solution.

Cortinal and Captivo (2003) study a Lagrangean heuristic combined with search methods, namely with local and tabu search to obtain the upperbound of the problem. The computational results show that Lagrangean heuristic combined with tabu search performs quite well, even for some large instances. Among the two heuristics proposed, it is indicated that the tabu search performed better than the local search for the test instances in the literature.

Chen and Ting (2007) develop a multiple ant colony system and a hybrid algorithm, which combines Lagrangean heuristic and ant colony system to solve the single-source capacitated facility location problem. The performances of the proposed methods are compared with the other heuristic algorithms in the literature. The computational results demonstrate that both proposed heuristics are effective and efficient for the problem.

2.2.3 Multi-Echelon, Capacitated Facility Location

A further extension of the location problem is the two-echelon facility location problem where a two-stage distribution process is considered with deliveries being made from first echelon facilities to second echelon facilities and from there to customers. The capacitated plant location problem has been solved by using both approximate and exact methods.

Many models in the literature that are concerned with material procurement, production and distribution activities treat each stage of the supply chain as separate systems and ignore complex supply chain interactions. In order to take more interactions into account in the supply chain, the two-stage facility location problem is further extended to incorporate vendors who have fixed the plant locations and supply raw materials to the production plants. These types of models are referred as the multi-echelon facility location problems in the literature.

For the first time in the literature, Geoffrion and Graves (1974) formulated a multi-commodity, capacitated single-period facility location problem as a mixed integer linear programming model. In order to solve the problem a solution technique based on Benders' decomposition is developed, implemented an applied to a real problem instance.

Tragantalerngsak et al. (1997) study the two-echelon, single-source, capacitated facility location problem. Each facility in the second echelon has a limited capacity and can be supplied by only one facility in the first echelon, which is uncapacitated. Similarly, in the second echelon each customer is serviced by only one facility. The number and location of facilities at both echelons, and the allocation of customers to the second echelon facilities are to be determined simultaneously. They propose a mathematical model for this problem and consider six heuristics based on Lagrangean relaxation for its solution. The subgradient optimization procedure is employed for updating Lagrange multipliers. The results indicate that the lowerbounds, which are obtained from the heuristic where the demand satisfaction constraints are relaxed, have a duality gap which is one third of the one obtained from the traditional linear programming relaxation. Also it is stated that the overall solution time for the heuristics are less than the time to solve the LP relaxation.

Pirkul and Jayaraman (1998) present the distribution network strategic design problem and discuss the transportation and distribution issues that exist for the multi-commodity, multi-level logistics problem. They develop a mixed integer programming model for the plant and warehouse location problem to minimize the total distribution and transportation costs and the fixed costs of opening and operating plants and warehouses. They employ Lagrangean relaxation, subgradient optimization and a primal heuristic to provide an effective feasible solution. The combination of these methods, called as Lagrangean heuristics, performs well in terms of tightness of the gap between the upper and lowerbound and provides good quality results in terms of the computational time regardless of problem size and structure.

Barbarosoğlu and Özgür (1999) deal with the hierarchical design problem of an integrated model of production and distribution functions in a two-echelon system. In order to solve the large scale problem, the Lagrangean relaxation is used to decouple the imbedded distribution and production subproblems, and subgradient optimization is implemented to coordinate the information flow between these in a hierarchical manner. A forward heuristic designed to solve the distribution subproblem is placed in the top level to restrict the solution of the production subproblem in the lower level.

Mazzola and Neebe (1999) present exact and heuristic solution procedures for the multi-commodity capacitated facility location problem (MPCFLP) in which the demand for a number of different product families must be supplied from a set of facility sites and each site offers a choice of facility types exhibiting different capacities. They define a branch-and-bound algorithm for the MPCFLP that utilizes bounds formed by a Lagrangean relaxation which decomposes the problem into uncapacitated facility location (UFL) subproblems and easily solvable 0-1 knapsack subproblems. The UFL subproblems are solved by the dual-based procedure of Erlenkotter. They also present a subgradient optimization based heuristic for the MPCFLP. The heuristic is seen to be extremely effective, generating result for the test problems that on average within 2% of optimality, and the branch-and-bound algorithm is to successfully solve all test problems.

Klose (1999) proposes a heuristic solution procedure for the two-stage capacitated facility location problem with single-source constraints. The approach is based on linear programming, and iteratively refines the LP formulation using valid inequalities and facets for various relaxations of the problem. After each re-optimization step, a feasible solution is obtained from the current fractional solution using different heuristics to determine the set of open depots and simple re-assignment procedures to find a feasible customer assignment. The computational results show that this method is able to compute near-optimal solutions and useful lowerbounds for the two-stage capacitated facility location problem in short computation time, even in the case of larger problem instances.

Marin and Pelegrin (1999) formulate the two-stage capacitated facility location problem in two different ways according to the decision variables used. In the first model, twice-indexed transportation variables are employed whereas the three-indexed variables are employed in the second model. They propose several relaxations based on Lagrangean relaxation for each model type, and compare the performance of these relaxations. For each relaxation, the subgradient optimization and a simple primal heuristic for generating the upperbound is used. The results show that the model with the twice-indexed variable is more appropriate for large-scale instances and among the twice-indexed models, the model relaxing the demand satisfaction constraints provides better solutions.

Hinojosa et al. (2000) deal with a facility location problem where the two-echelon facilities are located by selecting the time periods. The model intends to minimize the total cost for meeting the demands for all the products specified over the planning horizon at various customer locations while satisfying the capacity requirements of the production plants and intermediate warehouses. A Lagrangean relaxation is proposed to solve the problem, with a heuristic procedure that generates the feasible solutions for the original problem using the lowerbound results of the relaxed problem.

Tragantalerngsak et al. (2000) consider a two-echelon capacitated facility location problem with single-sourcing constraints at both echelons. This means each facility in the second echelon has a limited capacity and can be supplied by only one facility in the first echelon. Each customer is also serviced by only one facility in the second echelon. The number and the location of all facilities at both echelons and the allocation of customers to the second level facilities are to be determined simultaneously. A Lagrangean relaxation that employs branch-and-bound algorithm is proposed for the solution which indicates that the method is efficient.

Klose (2000) develops a Lagrangean relax-and-cut procedure for the two-stage, single-source, capacitated facility location problem. The approach is based on relaxing the plant and depot capacity constraints; thus the resulting Lagrangean subproblem is an aggregate capacitated plant location problem and can be solved efficiently by the branch-and-bound method based on dual ascent and subgradient optimization. Feasible solutions are obtained employing reassignment heuristics. The lowerbound is further improved by adding valid inequalities, which cuts off a near-optimal fractional solution of the primal master problem.

Pirkul and Jayaraman (2001) study an integrated logistics model for locating production and distribution facilities in a multi-echelon, multi-commodity environment. Both facility types are capacitated and the numbers of the opened facilities for each level are fixed to a predefined value. Only at the third echelon there is a single-sourcing constraint, besides multiple sourcing is allowed at the first and second echelons. They provide a Lagrangean relaxation and subgradient optimization based solution procedure.

Jang et al. (2002) propose a supply network with a global bill of material. The supply network management system is made up of four different modules which are the supply network optimization module, the planning module for production and distribution operations from raw material suppliers to customers, the model management module and finally the data management module. First two modules are solved by Lagrangean relaxation and a genetic algorithm, respectively. In the supply network optimization module, a model similar to the one that of Pirkul and Jayaraman (2001) is introduced. Based on the solution, an integrated planning module for the production and distribution operations covering raw material suppliers to customers is solved.

Elhedhli and Goffin (2005) propose a solution methodology for a production – distribution problem that is based on Lagrangean relaxation, interior point methods and branch-and-bound. Lagrangean relaxation is applied in a two-level hierarchy; branch-and-bound is based on a Lagrangean lowerbound and column generation, while interior point methods are used within a cutting plane context. Unlike the classical Lagrangean approach, the study devises a two-level hierarchy of Lagrangean relaxation, where the constraints are relaxed sequentially, rather than simultaneously, provides better bounds for the original problem.

In the study of Dias et al. (2007), the dynamic location problem with opening, closure and reopening of facilities is formulated and an efficient primal-dual heuristic that computes both upper and lower limits to its optimal solution is described. The problem considers the possibility of re-configuring any location more than once over the planning horizon. A primal-dual heuristic based on the study of Erlenkotter (1978) generates good-quality solutions, and calculates tight lowerbounds for the optimal objective function value. A branch-and-bound procedure that enables to optimize the problem is also described and tested over the same set of randomly generated problems.

2.3 Multi-Capacitated Facility Location Problems

The modular capacitated facility location problem (MCFLP) is an extension of single echelon CFLP in which multiple types of facilities with different sizes and operating costs are considered as possible alternatives in every facility location. In other words, in each plant facility there is more than one alternative that has different production capacities and operating costs. But only one (or none) of these alternatives can be installed to a facility location. The assumption of the traditional CFLP is one alternative facility for a location and this actually restricts its practical application, because in real life, most of the time decision makers have possibility to choose among the several types of technologies for a plant.

The MCFLP is a recently developed subject in logistics management and there are only a few papers about this subject due to the complexity of the problem. The MCFLP is quite difficult to solve because there are more binary variables compared to the CFLP. Broek et al. (2006) address to the multiple capacity levels only in variable costs. It means that the fixed cost of establishing a facility to a location is the equivalent for every possible capacity level which is called as the production volumes between the breakthroughs in this class of problems, but the operating costs differ at each breakthrough. Conversely, in the study of Amiri (2006), the fixed costs are different while the variable costs are equivalent for each capacity level. Other papers define different fixed and variable costs for each capacity level, as defined also in our study.

The “modular capacity” term refers to the candidate capacity levels of a facility and was introduced to the literature for the first time by Correia and Captivo (2003). Their study was triggered by a problem that arose in the location of health care facilities in Portugal. The authors realized that this kind of service should be built in the modules of a certain size which had a determined number of structures like consulting rooms, waiting rooms, and also staff rooms. It was wiser to install one or more modules to the locations that had higher patient intensity. If two modules were installed to a location, structures like rooms and machines had to be doubled, but it was not necessary to double the entire staff. This assumption actually, explains the increase in the fixed cost, and the decrease in the variable cost at higher capacity levels. Practical use of this modular structure can be found either in public service such as schools, waste management facilities, fire department structures or private services like warehouses, manufacturing plants and distribution centers etc…

Holmberg and Ling (1997) are the first researchers to introduce the multiple capacity concepts in logistic problems. They define their problem environments as the “facility location problem with staircase costs”. The staircase cost function in the model is presented as a finite piecewise linear increasing function with a finite set of discontinuities, each corresponding to a capacity level of a facility. In their study, a Lagrangean heuristic based on Lagrangean relaxation, subgradient optimization and a transportation problem heuristic which leads to the primal feasible solutions developed in order to deal with this problem. In order to compare the results of the Lagrangean heuristic, ADD heuristic developed by Jacobsen (1983), and improved by Domschake and Drexl (1985), is modified to handle the multiple capacity levels. The developed Lagrangean heuristic yields better results than the ADD heuristic and proved itself as quite an efficient method for solving the facility location problem with staircase costs.

Harkness and ReVelle (2002) developed an exact algorithm based on the study of Homberg and Ling (1997) to solve the staircase cost facility location problem (SCFLP). The exact algorithm consists of four parts which are Lagrangean relaxation model for finding proper lowerbounds, subgradient optimization for updating the Lagrange multipliers, problem reduction algorithm for fixing some facilities as open or close based on the techniques presented in the study of Christofides and Beasley (1983) and Beasley (1988), and a branch-and-bound algorithm for solving the reduced problems. They pointed out that the number of alternative capacity levels for a facility location is a key factor determining the performance of Lagrangean heuristic on nearly all measures, whereas the number of alternative facility location plays relatively a minor role. The cost parameters are presented as the other factor significantly related to the performance of Lagrangean relaxation.

Correia and Captivo (2003) generalize the SCFLP presented by Holmberg and Ling (1997) as the MCFLP and propose three different mixed integer linear programming models to compare. They stated that solving the problem could be quite difficult due to the large number of variables and constraints. Lagrangean relaxation is employed to obtain an effective lowerbound while a primal feasible heuristic based on the study of Beasley (1988) is adapted to obtain an upperbound. The Lagrange multipliers are updated using subgradient optimization technique presented by Herd et al. (1974). After the relaxation, emerging minimum cost flow problem is solved using the

Relax-IV algorithm that is introduced by Bertsegas and Tseng (1994). As a result, the presented heuristic leads to the satisfactory results regarding the average gaps and the execution time.

Amiri (2006) addresses to the distribution network design problem in a supply chain system that involves locating production plants and distribution warehouses and determining the best strategy for distributing the product from the plants to the warehouses and from the warehouses to customers. The goal is to select the optimum numbers, locations and the capacities for plants and warehouses to open to satisfy all customer demand at the minimum cost. In the study, all the plants and warehouses are multi-capacitated and multiple sourcing is allowed between all the facilities and customers. A mixed integer programming model and a Lagrangean relaxation with subgradient optimization based heuristic is developed. The results of the experiments indicate that the proposed heuristic procedure produces good feasible solutions when compared to the optimal/best available ones.

Correia and Captivo (2006) extended their previous work by adding single-sourcing constraints to the problem. Again a Lagrangean relaxation theme similar to their earlier work is used to solve the problem. But this time due to the complexity of the problem, even the relaxed subproblems are still very hard to solve. In order to obtain a proper lowerbound, they also relax the integrality constraint of the subproblems and solve the rest of the problem as in their previous work. A primal heuristic enhanced by tabu search and local search is developed during Lagrangean heuristic for obtaining good feasible solutions.

Broek et al. (2006) developed a model as an industrial application for the slaughterhouse industry of Norway. They dealt with a specific problem instance that Norwegian Meat Co-operative faced in determining the locations with production capacities of slaughterhouses and in the allocation of animals in different farming districts which had to be served. The authors observed that the slaughterhouse industry had economies of scale in the production facilities. In order to reflect the economies of scale in the model, they constructed their average cost function as convex and monotonically decreasing function with respect to the volume increase. As mentioned before, no fixed cost for establishing a facility was introduced to the model; all the computations were made by using the average cost function which is convex, continuous but nonlinear. The continuous cost function allowed the authors to employ a model which differs from the one that of Holmberg and Ling (1997) and Correia and Captivo (2003). Since LP relaxation generated poor results, the authors developed a Lagrangean heuristic containing Lagrangean relaxation, subgradient optimization and a greedy heuristic for generating tight upper and lowerbound. The heuristic is quite effective and results a 1% gap between the upper and lowerbound for the Norwegian Meat Co-operative problem.

CHAPTER 3

MATHEMATICAL FORMULATION OF THE TWO-STAGE MODULAR CAPACITATED FACILITY LOCATION PROBLEM

In this chapter firstly we have defined the modular capacity concept and our problem environment. Then we have discussed the assumptions of the model and presented a mixed integer programming model that fits the problem environment and the given assumptions. We have also explained the notations that are used during the study. Finally the requirement of developing a heuristic for the model has been discussed.

3.1 Problem Environment

The problem considered here is an integrated logistics model for locating the manufacturing and distribution facilities in a two-stage supply chain environment. Designing such logistic systems requires two essential decisions, one strategic; deciding where to locate the plants and warehouses, and the other is tactical; determining the distribution pattern from the plants to the customers via warehouses.

In this study, we address the distribution network design problem in a supply chain system that both locates the manufacturing plants and warehouses and determines the best pattern for distributing the goods from the plants to the warehouses and from the warehouses to the customers where multi-levels of capacities are available in the manufacturing plants with different fixed opening costs and different variable operating costs. The aim in our study is selecting the best set of plant and warehouse locations and plant capacity levels to install in order to satisfy the demand of the customers in a way that the overall distribution network cost is minimized.

In the literature, it is found that in general, both the capacitated and the uncapacitated facility location problems are solved by assigning a single fixed and/or a single variable cost for each facility location without considering the annual production amount of the facility. However, in our study, a new approach has been presented for modeling the distribution network in which the fixed and variable costs of opening the facilities are determined based on their opened capacity levels which are related to the planned annual production of the facility. In our model, the fixed opening cost increases non-monotonically where the production volume of the plant also increases regardless of the opening cost. On the other hand, the variable cost of producing a product decreases while the capacity level increases. The essential idea underlying this model is to better represent the real-life nature of the problem. The total cost function that encompasses the fixed opening and the variable operating cost has staircase steps that progressively become longer and flatter as the candidate facility increase in size denotes economies of scale as in real instances. An example of the staircase cost function is illustrated in Figure 3.1 below.

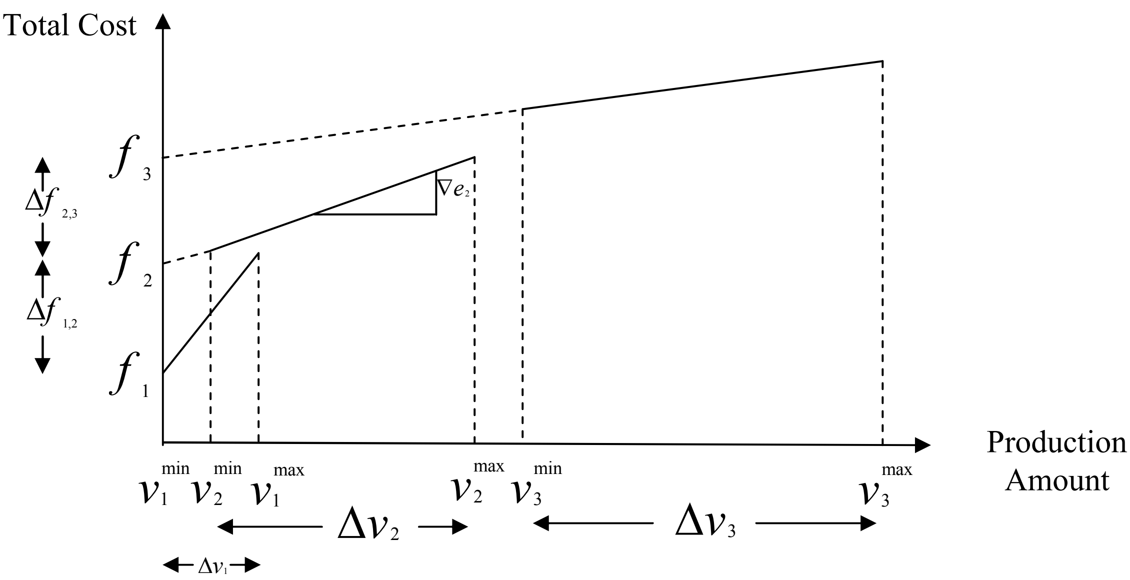

Figure 3.1 The Staircase Cost Function

In Figure 3.1, $v_l^{\min}$ and $v_l^{\max}$ represent the possible minimum and maximum production amounts of the plant, respectively in l^{th} capacity level. f_l denotes the fixed cost of opening the plant in l^{th} capacity level. As interpreted, the slope of this line at any point, ∇e_l gives the unit operating cost at the relevant capacity. It should be noted that as mentioned before, in general $\Delta v_l > \Delta v_{l-1}$ and $\nabla e_l < \nabla e_{l-1}$.

As can be seen in Figure 3.1, two capacity levels may overlap at certain points. It means at some points, two different capacity levels can produce the same production amount with different costs. Due to our objective, the capacity level with the higher total cost is never to be selected, hence to make the problem simpler, the capacity level that has higher costs can be prevented to produce at this volume. Thereby at most, one cost value is assigned to a specific production amount. Additionally, some production amounts may not be covered by any of the capacity levels. It means that a specific plant may not produce in some production amounts. In this situation, no cost function is available for these production amounts.

For better understandability, we can refer to the capacity levels as different production technologies. For example, let us assume that, the first capacity level is a universal lathe. Its fixed cost is low, but its variable cost is high and the production capacity is limited. Then the second capacity level refers to a numerical control lathe. Its fixed cost is higher than the universal lathe, but its variable cost is lower while its production volume is larger. The last capacity level can be conceived as a CNC lathe. Its fixed cost is the highest, its production amount is also the highest and the variable cost is the lowest among the alternatives.

Due to the capacitated nature of the system, only one plant or warehouse cannot satisfy the whole demand of customers. Consequently, at least two plants have to be opened in order to satisfy the total demand. Actually, the minimum number of plants required to be opened may be higher than two in many of the instances. This number is determined by the system itself according to the number of candidate plant locations and the maximum capacities in the problem. The warehouses are also capacitated, but differing from the plants, there is only one capacity level for a warehouse. There is also a required number of opened warehouses in order to satisfy the overall demand. The warehouses have only two costs; one of them is the fixed opening cost which represents the cost of installing a warehouse considering its capacity and the other cost is the variable operating cost that is fixed and do not vary according to the quantity of the goods handled in the warehouse. As the variable cost does not vary according to the capacity of the warehouse, it could be incorporated into the variable cost of transporting the goods from the warehouses to the customers.

As a final note for cost determination, we assume that, the fixed cost of operating a plant includes all the building, machinery, equipment and managerial costs to run a plant at a specified capacity level. Similarly for the warehouses, the fixed costs are assumed to include the building and some small-scale machinery such as crane, forklift leasing or rent, storage management cost and personnel wages to run a warehouse. Each facility has a different fixed and variable cost, which is reasonable because the costs may differ according to the region of the location site.

At both echelons multiple sourcing is allowed in our problem environment. In other words, the opened warehouses can be supplied from one or more facility. Similarly, each customer can be served by multiple warehouses. Based on these characteristics of the problem environment, the visual representation of the distribution network studied is shown in Figure 3.2.

As stated in the literature review chapter, it is clear that the modular capacitated facility location problems are modeled in single echelon environment. To our knowledge, modular capacitated facility problem has never been studied in the two-echelon environment in the literature so far. This study contributes to the literature with a unique model that extends the modular capacitated facility location problem to the two echelon environment, while introducing the capacitated warehouses to the model.

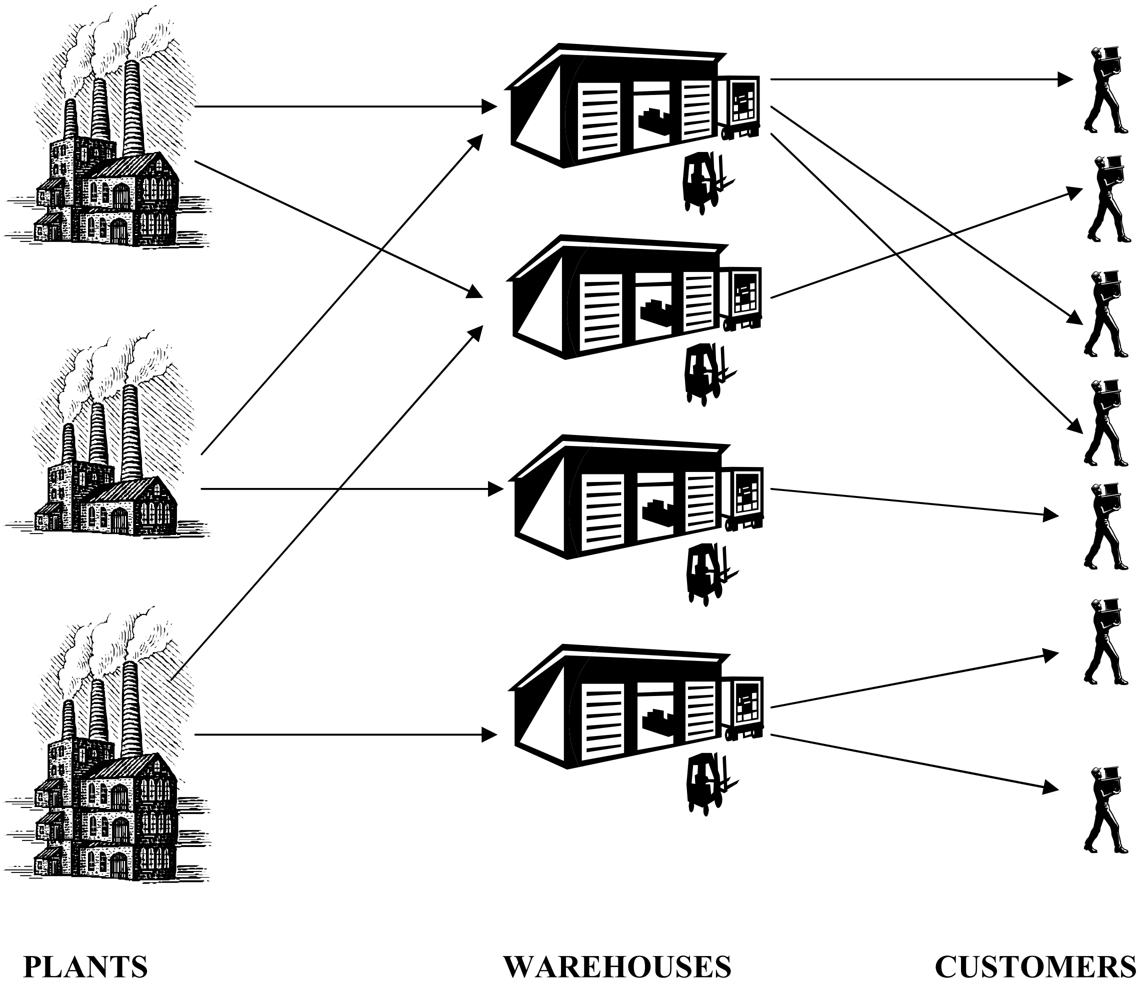

Figure 3.2 Graphical Representation of the Distribution Network

3.2 Model Formulation

In this section of our study, we present a mixed integer linear programming model for the TSMCFLP. The main aim of the model is to select the production plants and warehouses from a number of candidate sites and determine the capacity levels of the opened plants so that the annual total cost of the distribution network is minimized. The solution of our mixed integer model demonstrates the locations of the opened plants and warehouses and the capacities of the opened plants. The results also represent the distribution pattern of the network from the plants to the customers via the warehouses.

3.2.1 Assumptions of the Model

Main assumptions of the model are as follows:

1. The values of the following parameters are deterministic and known.
    - Customer locations and their annual demands
    - Candidate locations for plants and warehouses
    - Unit transportation cost of distributing goods from plants to warehouses
    - Unit transportation cost of distributing goods from warehouses to customers
    - Unit production cost of plants in a specific capacity level
    - Unit handling cost of warehouses
    - Annual fixed cost of opening plants at a specific capacity level
    - Annual fixed cost of opening a warehouse
    - Maximum and minimum annual production amounts of each capacity of a plant.
    - Maximum annual handling capacity of warehouses
2. Each plant has more than one candidate capacity level that determines the maximum and the minimum production amounts of the facility.
3. If a plant is opened in a specified capacity level, the distributed goods from this plant to all the warehouses neither exceed the maximum capacity nor be less than the minimum capacity of that capacity level.
4. Some capacity levels may overlap in some production amounts, and some production amounts may not be covered by any of the capacity level of a plant. If the capacity levels overlap, then the capacity level that has the lower total cost in the relevant production amount is always preferred to the higher cost one.
5. Only one capacity level can be selected for a plant location.
6. Warehouses are capacitated, and if opened, limited amounts can be supplied by and handled.
7. The number of the opened warehouses cannot exceed the pre-specified number which is determined by the decision makers.
8. Demands of the customers must be fully satisfied.

9. A customer can be assigned to one or more opened warehouses (multiple-sourcing).
10. An opened warehouse can be supplied by more than one opened plants (multiple-sourcing).
11. It is not allowed to transport products among facilities of the same type. That is, the shipment from a plant to another plant, the shipment from a warehouse to another warehouse, and the shipment from a customer to another customer is not permitted.
12. Reverse transportation is not allowed. In other words, warehouses can not supply plants and similarly customers can not supply warehouses.
13. There is only one type of product. This may be either a real product or some kind of an aggregated product covering more than one real product.
14. All the volumes of production, handling and transportation have to be integer values. The fractional volumes of production, handling or transportation are not allowed.
15. All minimum and maximum capacities of plants and warehouses have to be integer.

3.2.2 Notation of the Model

The following notation is used in the mixed integer linear programming model of the problem:

	PLANTS $\longrightarrow$	WAREHOUSES $\longrightarrow$	CUSTOMERS
Indices:	i	j	k
Sets :	I	J	K

I : Set of potential plant locations

$I = \{1, 2, \ldots, i, \ldots, |I|\}$

J : Set of potential warehouse locations

$J = \{1, 2, \ldots, j, \ldots, |J|\}$

K : Set of customers

$K = \{1, 2, \dots, k, \dots, |K|\}$

L : Set of potential capacity levels

$L = \{1, 2, \dots, l, \dots, |L|\}$

3.2.3 Parameters of the Model

The parameters of the mixed integer linear programming model are as follows:

d_k : Annual demand of Customer $k \in K$

b_{ij} : Unit transportation cost from Plant $i \in I$ to Warehouse $j \in J$

c_{jk} : Unit transportation and handling cost from Warehouse $j \in J$ to Customer $k \in K$

e_{il} : Unit production cost of Plant $i \in I$ at Level $l \in L$

f_{il} : Annual fixed cost of opening and operating Plant $i \in I$ at Level $l \in L$

g_j : Annual fixed cost of opening and operating Warehouse $j \in J$

$v_{il}^{\max}$: Maximum production capacity of Plant $i \in I$ at Level $l \in L$

$v_{il}^{\min}$: Minimum production capacity of Plant $i \in I$ at Level $l \in L$

w_j : Maximum handling capacity of Warehouse $j \in J$

$R_{\max}$: Number of maximum allowed warehouses

3.2.4 Decision Variables of the Model

The decision variables that will be determined by the model are cited below:

x_{ijl} :Total annual amount supplied from Plant $i \in I$ at Level $l \in L$ to Warehouse $j \in J$

z_{jk} :Total annual amount supplied from Warehouse $j \in J$ to Customer $k \in K$

$$q_{il} = \begin{cases} 1 & \text{If Plant } i \in I \text{ is opened at Level } l \in L \\ 0 & \text{Otherwise} \end{cases}$$

$$r_j = \begin{cases} 1 & \text{If Warehouse } j \in J \text{ is opened} \\ 0 & \text{Otherwise} \end{cases}$$

3.2.5 Original Problem

The TSMCFLP is formulated as a mixed integer linear programming problem below. From now on, this model will be called as the "Original Problem" and denoted by *P*.

$$P = Min \sum_{i \in I} \sum_{j \in J} \sum_{l \in L} \left(b_{ij} + e_{il} \right) x_{ijl} + \sum_{j \in J} \sum_{k \in K} c_{jk} z_{jk} + \sum_{i \in I} \sum_{l \in L} f_{il} q_{il} + \sum_{j \in J} g_j r_j$$

Subject to

$$\sum_{j \in J} x_{ijl} \leq v_{il}^{\max} q_{il} \qquad \forall i \in I \; and \; \forall l \in L \qquad (1)$$

$$v_{il}^{\min} q_{il} \leq \sum_{j \in J} x_{ijl} \qquad \forall i \in I \; and \; \forall l \in L \qquad (2)$$

$$\sum_{l \in L} q_{il} \leq 1 \qquad \forall i \in I \qquad (3)$$

$$\sum_{k \in K} z_{jk} \leq w_j r_j \qquad \forall j \in J \qquad (4)$$

$$\sum_{j \in J} z_{jk} = d_k \qquad \forall k \in K \qquad (5)$$

$$\sum_{j \in J} r_j \leq R_{\max} \qquad (6)$$

$$\sum_{k \in K} z_{jk} \leq \sum_{i \in I} \sum_{l \in L} x_{ijl} \qquad \forall j \in J \qquad (7)$$

$$x_{ijl} \geq 0 \; \; and \; Integer \qquad \forall i \in I, \; \forall j \in J \; \; and \; \; \forall l \in L \qquad (8)$$

$$z_{jk} \geq 0 \; \; and \; Integer \qquad \forall j \in J \; \; and \; \; \forall k \in K \qquad (9)$$

$$q_{il} \in \{0,1\} \qquad \forall i \in I \; \; and \; \; \forall l \in L \qquad (10)$$

$$r_j \in \{0,1\} \qquad \forall j \in J \qquad (11)$$

The objective of the model is to minimize the sum of the variable and the fixed cost components simultaneously. The variable costs include the costs of producing the goods, supplying goods from the plants to the warehouses, and from the warehouses to the customers. Moreover, the fixed costs include the annual costs of opening the plants and warehouses. Thus the objective function consists of four parts. The first part is the total cost of production and first echelon transportation, the second part is the total cost of handling and second echelon transportation, the third part is the total cost of the opened plants and the last part is the total cost of the opened warehouses.

First two constraint sets are the capacity constraints of the plants. The constraint set (1), also known as the capacity constraints, implies the following two situations: if a plant is opened at a location with specific capacity level, the total supplied product from this plant cannot exceed the maximum capacity and if a specific capacity level is not opened, then no product can be supplied from there.

The constraint set (2) ensures that if a facility is opened at the location $i \in I$, the facility has to provide products at least at the amount of the minimum requirement of the corresponding capacity level $l \in L$. In the literature, this class of constraints is called as the "minimum supply requirements" which are similar to the standard capacity constraints, but work in the opposite direction. For an opened plant, while the capacity constraints restraint the total amount supplied from above, minimum supply requirements restraint the total amount supplied from below.

The constraint set (3) guarantees that more than one capacity level cannot be opened in a possible plant location site.

The constraint set (4) is the capacity constraint set of the warehouses and ensures that a warehouse can serve customers if and only if the warehouse is opened. The constraint set (4) also restricts the amount of supplied goods from the opened warehouses to the customers.

The constraint set (5), which is also known as the demand satisfaction constraint, ensures that the demand of each customer has to be satisfied completely by one or more warehouses.

The constraint set (6) limits the number of the opened warehouses to a pre-specified number. This constraint is inherited from the p-median location problems. Various researchers including Cornuejols et al. (1977) and Pirkul and Jayaraman (1998) use the same constraint in their studies.

The constraint set (7) is the flow conservation constraint set and works as a balancer between the inbound and outbound amount of the warehouses. It guarantees that the total amount of the shipped product from an opened warehouse to all the customers cannot exceed the total amount of the supplied product of the same warehouse from all of the plants. This constraint set can be written in equality form but it is written in inequality form in order to decrease the complexity of the problem and increase the efficiency of the branch-and-bound procedures. The solution of the problem is the same in both equality and inequality forms, because the objective function always enforces the right-hand side to be equal to the left-hand side of the constraint set.

The constraint sets (8) and (9) are the non-negativity and integrality constraints and ensure that the amounts of the transported products from a plant at any capacity level to a warehouse and from a warehouse to a customer are integer.

The constraint sets (10) and (11) enforce the integrity on the relevant binary variables.

The "Original Problem", P, is a mixed integer linear programming problem which is an extension of the two-stage capacitated facility location problem (TSCFLP). The TSCFLP is shown to be NP-Hard by Mirchandandi and Francis (1990). TSMCFLP is an extension of TSCFLP and a special case of TSMCFLP, where there is only one available capacity level, is also a TSCFLP. This reflection proves that our problem is also NP-Hard. Thus; solving even the medium sized TSMCFLPs via using the commercial optimization packages is limited. In our experimental study, there are problem instances with over 160,000 variables and 1200 constraints. It is unlikely that commercial optimizers can find the optimal solution for instances like this in reasonable computational effort. Even then, buying these solvers for commercial purposes is expensive. Therefore, we have developed a heuristic method to solve problem P based on a well-established Lagrangean relaxation technique which has been employed successfully in various facility location problems in the studies by Klincewicz and Luss (1986), Barcelo et al. (1991), Beasley (1993), Tragantalerngsak et al. (1997), Pirkul and Jayaraman (1998), Mazzola and Neebe (1999), Jang et al. (2002), Amiri (2006), Marin (2007) following the pioneered studies of Held and Karp (1971) and Geoffrion (1972). Implementing Lagrangean relaxation requires computationally less effort and yields reasonably effective results compared to the optimal solution techniques.

In the following chapter, we discussed our Lagrangean relaxation based approach together with the subgradient optimization algorithm to update the Lagrange multipliers. The previous studies about these techniques are discussed as well.

CHAPTER 4

SOLUTION APPROACH

Lagrangean heuristic is one of the today's indispensable techniques to solve the combinatorial optimization problems. The heuristic consists of three parts; the first part is a technique to generate a lowerbound for the problem. The relaxed problem always has an objective function value that is less than or equal to the optimal solution of the original problem for the minimization case, because the Lagrangean relaxation contains less constraints than the original problem. Therefore, Lagrangean relaxation can be used (actually has to be used in order to call the procedure as Lagrangean heuristic) to generate a lowerbound.

The second part of the heuristic is the primal heuristic that is used for obtaining a proper upperbound. Most of the time, the relaxation of the original problem yields infeasible solutions for the original problem; hence a primal heuristic based on the results of the relaxed problem is needed to construct a feasible solution, that is, an upperbound for the original problem. This upperbound heuristic is the distinctive part of the Lagrangean heuristic from the exact Lagrangean relaxation methods in which, an optimization technique such as branch-and-bound is used in order to close the gap between the solution of the relaxed problem and the optimal solution of the original problem. On the other hand, a primal heuristic is employed in Lagrangean heuristic technique in order to find an acceptable solution in a reasonable computer effort.

The last part is the procedure to update the Lagrange multipliers. Calculating the Lagrangean relaxation lowerbound requires the solution of a concave nondifferentiable optimization problem, because two decision variables which are the Lagrange multipliers and the decision variables of the original problem are multiplied in objective function. Solving this kind of problem may be harder than solving the original problem. In order to deal with this complexity, the values of the Lagrange multipliers are determined in a separate problem. It is quite difficult to find the best values for the Lagrange multipliers so that the overall heuristic is repeated until the efficient Lagrange multipliers are acquired. Various algorithms such as subgradient optimization (Poljak 1969), volume algorithm (Barahona and Chudak, 2005), bundle methods (Crainic et al., 2001), multiplier adjustment methods (Erlenkotter, 1978) can be used to update the Lagrange multipliers. In our study, subgradient optimization has been selected for updating the Lagrange multipliers due to the easy adaptation and less computational effort it requires.

These three components of the Lagrangean heuristic are explained in detail in the following sections.

4.1 Lagrangean Relaxation

Lagrangean relaxation has been used as an effective algorithm for generating lowerbounds for both exact algorithms like branch-and-bound and Lagrangean heuristics for solving the combinatorial optimization problems. The Lagrangean relaxation of a mixed integer problem is obtained through relaxing a set of constraints from the original problem and attaching these constraints into the objective function by penalizing them with proper weights. Agar and Salhi (1998) express that the Lagrangean relaxation is inspired from an important observation that the formulation of many hard combinatorial problems consists of an easy problem that become difficult by the addition of a set of constraints.

The problem after relaxing the hard constraints is called as the relaxed problem which is easier to solve due to its special structure. An illustration of the Lagrangean relaxation is as follows:

$$Z = Minimize\ Cx$$

Subject to

$$Ax = b \tag{12}$$

$$Bx \leq d \tag{13}$$

$$x \geq 0 \text{ and integer} \tag{14}$$

where A and B are coefficient matrices and Constraint set (12) represents the easy constraints and Constraint set (13) represents the hard constraints which make the whole problem difficult to solve or decompose. If the hard constraints are somehow excluded from the constraint set, the problem becomes much more easier to solve. The Lagrangean relaxation works exactly this way. The hard constraints are penalized with Lagrange multipliers, $\lambda \geq 0$, and added to the objective function as follows:

$$Z_{LR} = Minimize\ Cx + \lambda\left(Bx - d\right)$$

Subject to

$$Ax = b \tag{12}$$

$$x \geq 0 \text{ and integer} \tag{14}$$

$$\lambda \geq 0 \tag{15}$$

The optimal objective value of the Lagrangean relaxation problem with the optimal set of Lagrange multipliers provides a lowerbound for the optimal solution to the original minimization problem. It is a lowerbound, because some constraints of the original problem are omitted from the constraint set that gives rise to the enlargement of the feasible region.

4.1.1 Issues of Lagrangean Relaxation

According to Beasley (1995), before reaching a proper and efficient lowerbound using Lagrangean relaxation, there are two major issues that have to be dealt with. These issues can be categorized as:

1. Tactical Issue: How will the optimal Lagrangean multiplier values that give the maximum objective function value of the relaxed problem be found?

2. Strategic Issue: Which sets of constraints should be chosen to relax for the best lowerbound?

4.1.1.1 Tactical Issue of Lagrangean Relaxation

Fisher (1981) emphasizes that the objective function of a Lagrangean relaxation for a mixed integer programming problem is differentiable almost every where, but it is generally nondifferentiable at the optimal point. Hence, to be able to find a near-optimal solution, the problem of finding the optimal values of the Lagrange multipliers has to be detached from the relaxed problem and be solved as a separate problem. The aim of the new problem is finding the best Lagrange multipliers for the Lagrangean relaxation problem that maximizes its objective function value. This problem is called as the Lagrangean dual problem. In our illustration, the Lagrangean dual problem is formulated as follows:

$$Z^D = \underset{\lambda \geq 0}{Max}\left(Z_{LR}(\lambda^t)\right)$$

where Z_{LR} is the Lagrangean relaxation problem with given (known) x variables which is also called as oracle. Tragantalerngsak et al. (1997) express that to solve the Lagrangean dual problem, standard ascent methods based on the gradients of the problem cannot be employed due to the nondifferentiability of the problem. However, there are a lot of alternatives to update the Lagrange multipliers that use the subgradients instead of gradients, such as subgradient optimization, volume algorithm, bundle methods, steepest ascent methods and multiplier adjustment methods.

Subgradient optimization is an iterative technique which is as an extension of the gradient optimization developed to solve the nondifferentiable functions. As stated in Fumero (2001), although many other techniques with stronger convergence properties have been developed, the subgradient optimization and its variants seem to have wider acceptability among researchers and continue to remain as one of the most effective and useful techniques for solving the dual problems, especially when large scale applications are considered. Subgradient optimization used in Lagrangean heuristics is so common that quite a few people believe that the subgradient optimization is a mutual part of the Lagrangean relaxation algorithm. The wide employment of the subgradient optimization in Lagrangean relaxation is due to the simplicity of the algorithm's structure.

More recent algorithms provide outstanding theoretical convergence performances, but they cannot carry their theoretical convergence results into real life applications for large scale problems or problems with complex structures, that is because usually searching for a descent direction in recently developed methods is computationally inefficient. For example, bundle methods require the solution of a quadratic problem to find the descent direction in each iteration. On the other hand, subgradient optimization is both simple to use and computationally efficient in calculating the descent direction. The subgradient optimization has a zig-zag pattern that wastes time, but it gains back this lost time in computing the descent direction. In our study, we have employed a modified subgradient algorithm to update the Lagrangean multipliers. The details of the modified subgradient optimization are explained in the following sections.

4.1.1.2 Strategic Issue of Lagrangean Relaxation

In the illustration of Lagrangean relaxation, the constraints are identified as hard and easy constraints without having any difficulties. But in real life, determining the hard constraints to be relaxed is not so easy. The relaxation of a different set of constraints yields solutions with different qualities regarding the tightness of the solution and the computational effort.

For example, if we go on with to the previous illustration, we can see that there are quite a few candidates for relaxation. We can relax either (12) or (13) individually, or both. Constraint set (14) cannot be relaxed in a Lagrangean relaxation fashion, because actually this constraint set is not an equality or inequality, furthermore, it does not have a Lagrange multiplier/dual variable. However, it can still be removed from the model and replaced with a constraint set that ensures the relevant variables take value between zero and one. This class of relaxation is called as linear programming relaxation (LP relaxation). LP relaxation is an easier way of solving the problem, but as Beasley (1995) states, it always yields worse results compared to the results of the Lagrangean relaxation.

There are two innovative alternatives for relaxation that may not be seen at first glance. The first one is called as Lagrangean decomposition and it relies on assigning different decision variables for each constraint set by adding a binding constraint set that guarantees the value of the new decision variable to be equal to the older one, and then relaxing the binding constraint of these two variables as shown in the illustration below.

For our example, let us replace the decision variables of the second constraint with a new variable “y” and add a binding constraint. The model becomes:

$$Z = Minimize\, Cx$$

Subject to

$$Ax = b \tag{12}$$

$$By \leq d \tag{16}$$

$$x = y \tag{17}$$

$$x \geq 0 \text{ and integer} \tag{14}$$

$$y \geq 0 \text{ and integer} \tag{18}$$

After relaxing the constraint set (17), the Lagrangean relaxation of the model becomes:

$$Z_{LP'} = Minimize\ Cx + \lambda\left(x - y\right)$$

Subject to

$Ax = b$ (12)

$By \leq d$ (16)

$x \geq 0$ and integer (14)

$y \geq 0$ and integer (18)

λ unrestricted (19)

The other innovative relaxation is called the semi-Lagrangean relaxation and introduced by Beltran et al. (2006). It exploits one or more equality constraints in the problem and relaxes only one side of the equation. To show the logic of the semi-relaxation, let us recall our illustration. The constraint set (12) actually is a combination of two different constraint sets shown below:

$Ax \geq b$ (12a)

$Ax \leq b$ (12b)

By replacing the constraint set (12) with (12a) and (12b), we get the following model:

$$Z = Minimize\, Cx$$

Subject to

$Ax \geq b$ (12a)

$Ax \leq b$ (12b)

$Bx \leq d$ (13)

$x \geq 0$ and integer (14)

Now we obtain two more constraint sets that can be relaxed. When we relax constraint set (12a), as Beltran et al. (2006) state, we acquire a relaxed model for the problem as follows:

$$Z = Minimize\ Cx + \lambda(b - Ax)$$

Subject to

$Ax \le b$ (12b)

$Bx \le d$ (13)

$x \ge 0$ and integer (14)

As cited above, there are a lot of candidates for relaxation even in a model which has only a few constraint sets. Geoffrion and Mc Bride (1978) show that generally a relaxation which gives a tighter bound requires longer computational time, whereas an easily solvable relaxation problem is likely to give poor results. Therefore, researchers who are willing to choose the best relaxation set are facing with a trade-off between the computational effort required and the quality of the bounds. As Trangantalerngsak et al. (1997) state, the ease of the solution depends on the methods available for solving the subproblem.

4.1.2 Lagrangean Relaxation of TSMCFLP

In our problem, there are seven constraint sets, excluding the binary and non-negativity constraints ranging from (1) to (7) that lead to $2^7 = 128$ possible relaxations even without considering the innovative ways of relaxation mentioned above. In practice, as Beasley (1995) pointes, most of them are not worth considering. As for our problem relaxing the constraint set (1) yields the problem to two-stage uncapacitated facility location problem with minimum supply requirements and capacitated warehouses; relaxing the constraint set (2) results in two-stage, capacitated facility location problem with staircase costs, relaxing the constraint set (3) generates a two-stage capacitated facility location problem with minimum supply requirements. On the other hand, relaxing the other three constraints, (4), (5) and (6), by themselves does not change the structure of the problem. Relaxing the constraint set (4) results in two-stage, modular capacitated facility location problem with uncapacitated warehouses, relaxing the constraint set (5) results in two-stage, modular capacitated facility location problem without a specific demand and the constraint set (6) generates a two-stage, modular capacitated facility location problem without a restriction on the number of warehouse to be opened. All of the relaxed problems generated so far are still non-decomposable and hard to solve. On the other hand, relaxing the constraint set (7) with the vector of Lagrange multipliers $\alpha = \left(\alpha_1, \alpha_2, ..., \alpha_j, ..., \alpha_{|J|}\right)$, seems to be quite promising:

$$P^{LR}(\alpha) = Min \sum_{i\in I}\sum_{j\in J}\sum_{l\in L}\left(b_{ij} + e_{il}\right)x_{ijl} + \sum_{j\in J}\sum_{k\in K} c_{jk} z_{jk} + \sum_{i\in I}\sum_{l\in L} f_{il} q_{il} + \sum_{j\in J} g_j r_j +$$
$$\sum_{j\in J}\alpha_j\left(\sum_{k\in K} z_{jk} - \sum_{i\in I}\sum_{l\in L} x_{ijl}\right)$$

Subject to

$\sum_{j\in J} x_{ijl} \le v_{il}^{\max} q_{il}$	$\forall i \in I \; and \; \forall l \in L$	(1)
$v_{il}^{\min} q_{il} \le \sum_{j\in J} x_{ijl}$	$\forall i \in I \; and \; \forall l \in L$	(2)
$\sum_{l\in L} q_{il} \le 1$	$\forall i \in I$	(3)
$\sum_{k\in K} z_{jk} \le w_j r_j$	$\forall j \in J$	(4)
$\sum_{j\in J} z_{jk} = d_k$	$\forall k \in K$	(5)
$\sum_{j\in J} r_j \le R_{\max}$		(6)
$x_{ijl} \ge 0 \;\; and \; Integer$	$\forall i \in I, \; \forall j \in J \;\; and \;\; \forall l \in L$	(8)
$z_{jk} \ge 0 \;\; and \; Integer$	$\forall j \in J \;\; and \;\; \forall k \in K$	(9)
$q_{il} \in \{0,1\}$	$\forall i \in I \;\; and \;\; \forall l \in L$	(10)
$r_j \in \{0,1\}$	$\forall j \in J$	(11)

After relaxation of the constraint set (7), the relaxed problem, P^{LR} can be decomposed into two separate problems for each echelon. The first echelon is a relaxed single-echelon, modular capacitated facility location problem, whereas the second echelon part is a single-echelon, capacitated warehouse location problem. From now on, these problems will be called as main subproblem P^{LR1} and main subproblem P^{LR2}, respectively.

The main subproblems P^{LR1} and P^{LR2} are as follows:

The main subproblem P^{LR1}:

$$P^{LR1}(\alpha) = Min \sum_{i \in I} \sum_{j \in J} \sum_{l \in L} \left(b_{ij} + e_{il} - \alpha_j \right) x_{ijl} + \sum_{i \in I} \sum_{l \in L} f_{il} q_{il}$$

Subject to

$$\sum_{j \in J} x_{ijl} \leq v_{il}^{\max} q_{il} \qquad \forall i \in I \; and \; \forall l \in L \qquad (1)$$

$$v_{il}^{\min} q_{il} \leq \sum_{j \in J} x_{ijl} \qquad \forall i \in I \; and \; \forall l \in L \qquad (2)$$

$$\sum_{l \in L} q_{il} \leq 1 \qquad \forall i \in I \qquad (3)$$

$$x_{ijl} \geq 0 \;\; and \; Integer \qquad \forall i \in I, \; \forall j \in J \;\; and \;\; \forall l \in L \qquad (8)$$

$$q_{il} \in \{0,1\} \qquad \forall i \in I \;\; and \;\; \forall l \in L \qquad (10)$$

The main subproblem P^{LR2}:

$$P^{LR2}(\alpha) = Min \sum_{j \in J} \sum_{k \in K} \left(c_{jk} + \alpha_j \right) z_{jk} + \sum_{j \in J} g_j r_j$$

Subject to

$$\sum_{k \in K} z_{jk} \le w_j r_j \qquad \forall j \in J \qquad (4)$$

$$\sum_{j \in J} z_{jk} = d_k \qquad \forall k \in K \qquad (5)$$

$$\sum_{j \in J} r_j \le R_{\max} \qquad (6)$$

$$z_{jk} \ge 0 \ \ and\ Integer \qquad \forall j \in J \ \ and\ \ \forall k \in K \qquad (9)$$

$$r_j \in \{0,1\} \qquad \forall j \in J \qquad (11)$$

It should be noted that the main subproblem P^{LR1} can be decomposed for each plant location candidate, $i \in I$. However, the main subproblem P^{LR2} is still a difficult problem to solve using the exact methods. Hence al least one constraint set of the second main subproblem must be relaxed in order to solve it more efficiently. The constraint set (6) cannot be the relaxed constraint, because it does not exploit the structure of the main subproblem. We can choose the constraint set (4) or (5) or both to relax or we can use the other innovative relaxations.

Fisher (1981) emphasizes that the tightness of bounds after choosing different constraint sets to relax is completely problem-specific and largely empirical. In order to find the lowest gap possible, Beasley (1995) recommends investigation of the previous studies about Lagrangean relaxation related to the study. The previous studies that have similar structures with our problem are Tragantalerngsak et al. (1997, 2000), Pirkul and Jayaraman (1998, 2001), Marin and Pelegrin (1999), Klose (2000), Jang et al. (2002), Elhedhli and Goffin (2005) and Amiri (2006).

Many of the previous studies which compare different Lagrangean relaxations about the two stage problems suggest researchers to relax the similar constraints to the constraint set (5) and (7) simultaneously to obtain the "best" problem considering both the computational burden and the quality of the results of the relaxation. Another option is relaxing the constraint set (4) instead of (5) and solving the remaining uncapacitated warehouse location problem by employing an efficient algorithm such as multi-ascent method of Körkel (1989) or hybrid multistart heuristic of Resende and Wernck (2006).

In our study, we have decided to relax the constraint sets (5) and (7). After relaxing these constraints with Lagrange multipliers α and β respectively, the relaxed problem becomes as follows:

$$P^{LR}(\alpha,\beta) = Min \sum_{i\in I}\sum_{j\in J}\sum_{l\in L}\left(b_{ij}+e_{il}\right)x_{ijl} + \sum_{j\in J}\sum_{k\in K} c_{jk}z_{jk} + \sum_{i\in I}\sum_{l\in L} f_{il}q_{il} + \sum_{j\in J} g_j r_j +$$

$$\sum_{j\in J}\alpha_j\left(\sum_{k\in K} z_{jk} - \sum_{i\in I}\sum_{l\in L} x_{ijl}\right) + \sum_{k\in K}\beta_k\left(d_k - \sum_{j\in J} z_{jk}\right)$$

Subject to

$$\sum_{j\in J} x_{ijl} \le v_{il}^{\max} q_{il} \qquad \forall i \in I \; and \; \forall l \in L \qquad (1)$$

$$v_{il}^{\min} q_{il} \le \sum_{j\in J} x_{ijl} \qquad \forall i \in I \; and \; \forall l \in L \qquad (2)$$

$$\sum_{l\in L} q_{il} \le 1 \qquad \forall i \in I \qquad (3)$$

$$\sum_{k\in K} z_{jk} \le w_j r_j \qquad \forall j \in J \qquad (4)$$

$$\sum_{j\in J} r_j \le R_{\max} \qquad (6)$$

$$x_{ijl} \ge 0 \;\; and \; Integer \qquad \forall i \in I, \; \forall j \in J \;\; and \;\; \forall l \in L \qquad (8)$$

$$z_{jk} \ge 0 \;\; and \; Integer \qquad \forall j \in J \;\; and \;\; \forall k \in K \qquad (9)$$

$$q_{il} \in \{0,1\} \qquad \forall i \in I \;\; and \;\; \forall l \in L \qquad (10)$$

$$r_j \in \{0,1\} \qquad \forall j \in J \qquad (11)$$

It should be noted that we have already mentioned about dividing the problem into two independent subproblems P^{LR1} and P^{LR2} if we relax the constraint set (7). This structure still holds, because the constraint set (7) is still in our relaxed set. But due to the second relaxed constraint set (5), the main subproblems are slightly changed. The main subproblems P^{LR1} and P^{LR2} are now as follows:

The main subproblem P^{LR1}:

$$P^{LR1}(\alpha) = Min \sum_{i \in I} \sum_{j \in J} \sum_{l \in L} \left(b_{ij} + e_{il} - \alpha_j\right) x_{ijl} + \sum_{i \in I} \sum_{l \in L} f_{il} q_{il}$$

Subject to

$$\sum_{j \in J} x_{ijl} \leq v_{il}^{\max} q_{il} \qquad \forall i \in I \; and \; \forall l \in L \qquad (1)$$

$$v_{il}^{\min} q_{il} \leq \sum_{j \in J} x_{ijl} \qquad \forall i \in I \; and \; \forall l \in L \qquad (2)$$

$$\sum_{l \in L} q_{il} \leq 1 \qquad \forall i \in I \qquad (3)$$

$$x_{ijl} \geq 0 \;\; and \; Integer \qquad \forall i \in I, \; \forall j \in J \;\; and \;\; \forall l \in L \qquad (8)$$

$$q_{il} \in \{0,1\} \qquad \forall i \in I \;\; and \;\; \forall l \in L \qquad (10)$$

The main subproblem P^{LR2}:

$$P^{LR2}(\alpha, \beta) = Min \sum_{j \in J} \sum_{k \in K} \left(c_{jk} + \alpha_j\right) z_{jk} + \sum_{j \in J} g_j r_j + \sum_{k \in K} \beta_k \left(d_k - \sum_{j \in J} z_{jk} \right)$$

Subject to

$$\sum_{k \in K} z_{jk} \leq w_j r_j \qquad \forall j \in J \qquad (4)$$

$$\sum_{j \in J} r_j \leq R_{\max} \qquad (6)$$

$$z_{jk} \geq 0 \;\; and \; Integer \qquad \forall j \in J \;\; and \;\; \forall k \in K \qquad (9)$$

$$r_j \in \{0,1\} \qquad \forall j \in J \qquad (11)$$

Thus, this stage, our Lagrangean model has been decomposed into two independent main subproblems. The sum of the values of these main subproblems gives us the solution of the relaxed problem, P^{LR}, at each iteration. The mathematical expression of the value of the relaxed problem at iteration t with Lagrange multiplier vectors α^t *and* β^t is as follows:

$$P^{LR}\left(\alpha^t,\beta^t\right)=P^{LR1}\left(\alpha^t\right)+P^{LR2}\left(\alpha^t,\beta^t\right)$$

It should be noted that we have already discussed finding the best Lagrange multipliers: We have to turn this problem into an iterative approach in which at each iteration Lagrangean multipliers are updated using the subgradient optimization and the subproblems are solved given the predetermined values of Lagrange multipliers. Therefore we can exclude the constant expression "$\sum_{k\in K}\beta_k d_k$" from the objective function of the second main subproblem, P^{LR2}, and add this expression while calculating the value of the relaxed problem at each iteration. Then the value of the relaxed problem is as follows:

$$P^{LR}\left(\alpha^t,\beta^t\right)=P^{LR1}\left(\alpha^t\right)+P^{LR2}\left(\alpha^t,\beta^t\right)+\sum_{k\in K}\beta_k^t d_k$$

The Lagrangean dual problem which is also called as the lowerbound problem can be expressed as the maximum value of the relaxed problem among t iterations.

$$LB\left(\alpha^*,\beta^*\right)=\underset{t}{Max}\left\{P^{LR}\left(\alpha^t,\beta^t\right)\right\}$$

In the literature, adding additional inequalities that are called valid inequalities is recommended in order to increase the value of the Lagrangean dual problem. Valid inequalities are actually redundant for the original model, but become useful after relaxing some constraint sets in the model; that means the valid inequalities can only reveal the undercover properties that disappear from the model after relaxing some constrains. They divide the feasible region of the problem into two parts and omit the part that does not contain the optimal solution of the original problem. This point is very important; an inequality (or equality) may be treated as a valid inequality if and only if the value of the Lagrangean dual problem is still less than or equal to the optimal value of the original problem and relaxed problem is still an easily solvable problem.

For example, we cannot limit the number of the opened warehouses unless we solve the optimal problem with this limitation. Otherwise the lowerbound may climb higher than the objective function value of the original problem and lead us to a wrong solution.

As Klein Haneveld and van der Vlerk (2000) state, in general, it is difficult to find strong valid inequalities that result in a substantial reduction of the computational time and better bounds, simultaneously. For our study, it is also quite difficult to find good valid inequalities, because many of the valid inequalities that are generated for the capacitated facility location problem do not hold in our case or are redundant due to the structure of the problem. For each of the main subproblems, we can add a constraint that limits the minimum number of opened facilities. These constraints may be expressed as:

$$\sum_{i \in I} \sum_{l \in L} q_{il} \geq Q_{\min} \tag{20}$$

$$\sum_{j \in J} r_j \geq R_{\min} \tag{21}$$

where $Q_{\min}$ and $R_{\min}$ values in the constraints (20) and (21) are the minimum number of opened capacity levels and warehouses respectively, in order to satisfy the overall demand. To calculate the value of $Q_{\min}$, firstly the highest available capacities of the plants are sorted in non-increasing order. Let us assume that v_{il}^{n} represents the n^{th} biggest maximum capacity among all plants, then the new set satisfies the following inequality:

$$v_{il}^{1} \geq v_{il}^{2} \geq \ldots \geq v_{il}^{n} \geq \ldots \geq v_{il}^{|N|}, \quad where|N| = |I|$$

Then the value of $Q_{\min}$ has to satisfy the following condition:

$$\sum_{n=1}^{Q_{\min}-1} v_{il}^{n} \leq \sum_{k \in K} d_k \leq \sum_{n=1}^{Q_{\min}} v_{il}^{n}$$

This condition means that at least $Q_{\min}$ opened plants can satisfy the total demand of the customers. The value of $R_{\min}$ is calculated in a similar way. The capacities of the warehouses are sorted in non-decreasing order with the same index, *n*. Hence the n^{th} biggest warehouse capacity is denoted by w_j^n. This set also has to satisfy the following condition:

$$w_{j}^{1} \geq w_{j}^{2} \geq \ldots \geq w_{j}^{n} \geq \ldots \geq w_{j}^{|N|}, \quad where|N| = |J|$$

The value of $R_{\min}$ is the minimum number of opened warehouses which satisfies the inequality below:

$$\sum_{n=1}^{R_{\min}-1} w_{j}^{n} \leq \sum_{k \in K} d_k \leq \sum_{n=1}^{R_{\min}} w_{j}^{n}$$

Differing from the capacitated warehouse location problem that we have dealt with in the second main subproblem, actually the constraint (20) is not a strong inequality for the first main subproblem, because in the capacitated warehouse location problem the difference between the fixed cost of the plant which has the minimum production capacity and the fixed cost of the plant which has the maximum production capacity is small in quantity, on the other hand, in the modular capacitated facility location problem the fixed costs of the first level capacities are much lower than the fixed costs of highest capacities, as expected. Regardless, valid inequalities (20) and (21) are included in the model, because they have no significant computational cost, but have improvement on the value of the relaxed problem.

There is also another valid inequality set for each echelon that can be added into our model. In our model, the decision variables x_{ijl} *and* z_{jk} are already restricted by the maximum production capacities of the capacity levels and the maximum handling capacities of the warehouses by the constraint sets (1) and (4), respectively. It means that we limit these decision variables by the constraints about their origin point (departing site). In addition to that, we can restrict these variables using their destination point (arrival site). It is clear that the destination point of the x_{ijl} variable is the warehouse *j* and the destination point of the z_{jk} variable is the customer *k*. These destination points have their own capacity restrictions that we can use for restricting the relative variables. As a result, the x_{ijl} variable has to be less than or equal to both $v_{il}^{\max}$ and w_j. Similarly, the z_{jk} variable has to be less than or equal to both w_j and d_k. Thus the constraint sets (8) and (9) of our relaxed problem can be rearranged as below:

$$0 \le x_{ijl} \le Min\left(v_{il}^{\max}, w_j\right) \, and \text{ integer} \qquad \forall i \in I, \forall j \in J, \text{and } \forall l \in L \qquad (8)$$

$$0 \le z_{jk} \le Min\left(d_k, w_j\right) \, and \text{ integer} \qquad \forall j \in J \text{ and } \forall k \in K \qquad (9)$$

4.1.2.1 Solution Methodology for the First Main Subproblem

After adding these valid inequalities into our model, the main subproblems are ready to be solved separately. The first main subproblem and its solving methodology are as follows:

$$P^{LR1}(\alpha) = Min \sum_{i \in I} \sum_{j \in J} \sum_{l \in L} \left(b_{ij} + e_{il} - \alpha_j \right) x_{ijl} + \sum_{i \in I} \sum_{l \in L} f_{il} q_{il}$$

Subject to

$$\sum_{j \in J} x_{ijl} \le v_{il}^{\max} q_{il} \qquad \forall i \in I \text{ and } \forall l \in L \qquad (1)$$

$$v_{il}^{\min} q_{il} \le \sum_{j \in J} x_{ijl} \qquad \forall i \in I \text{ and } \forall l \in L \qquad (2)$$

$$\sum_{l \in L} q_{il} \le 1 \qquad \forall i \in I \qquad (3)$$

$$Q_{\min} \le \sum_{i \in I} \sum_{l \in L} q_{il} \qquad (20)$$

$$0 \le x_{ijl} \le Min\left(v_{il}^{\max}, w_j \right) \text{ and Integer} \qquad \forall i \in I,\ \forall j \in J \text{ and } \forall l \in L \qquad (8)$$

$$q_{il} \in \{0,1\} \qquad \forall i \in I \text{ and } \forall l \in L \qquad (10)$$

In the literature there is an efficient solution methodology for the first main subproblem P^{LR1}. It can easily be seen that the main subproblem can be decomposed into subproblems for each plant location $i \in I$ and solved separately after omitting the constraint (20), as no hard constraints that bind the capacity levels of the plants remain. It should be noted that we isolate the constraint (20) from the model, but at the end of the solution, while determining the opened warehouses, we attach back this constraint and give our final decision considering this constraint. Details of attaching these constraint sets back to the model are explained in further sections.

Since the decision variable q_{il} is binary, a plant in a specific location i with a specific capacity level l can either be opened ($q_{il} = 1$) or remain closed ($q_{il} = 0$). If q_{il} is "remain closed", there is no production, so all related x_{ijl} variables and the objective function value of the subproblem are equal to zero. On the other hand, if it is decided to be opened, by setting q_{il} to one, the problem turns into the following problem:

$$P_{i'l'}^{LR1}(\alpha) = Min\sum_{j\in J}\left(b_{i'j} + e_{i'l'} - \alpha_j\right)x_{i'jl'} + f_{i'l'}$$

Subject to

$$\sum_{j\in J} x_{i'jl'} \le v_{i'l'}^{\max} \qquad (1)$$

$$v_{il}^{\min} \le \sum_{j\in J} x_{ijl} \qquad (2)$$

$$0 \le x_{i'jl'} \le Min\left(v_{i'l'}^{\max}, w_j\right) \text{ and } Integer \qquad \forall j \in J \qquad (8)$$

The constraint set (3) is eliminated, because this constraint is satisfied automatically by setting only one q_{il} equal to one at each time. This problem is actually a special type of a knapsack problem. The structure of the knapsack is very special because it is the combination of the following four knapsack problems: it is a minimization knapsack problem (MinKP) since the objective function of the problem is minimization. On the other hand, it is an unweighted knapsack problem due to the equal weights of the decision variables. Also it is an interval knapsack problem (I-KP) on account of existence of additional lower bound limit on the total weight of a knapsack and finally it is a bounded knapsack problem (BKP) because of the structure of the x_{ijl} variables. Detailed information about these knapsack problem classes can be found in Martello and Toth (1990), Zhou (2006) and Babaioff et al., (2007).

Although three of these knapsack problems (MinKP, I-KP and BKP) stated above are NP-hard in the given references above, our subproblem can be solved optimally and very efficiently using a simple algorithm, due to the nature of the unweighted knapsack problem type. It can be seen that in our knapsack problem all weights of the decision variables are the same and equal to one which is aliquot of all upper and lower threshold values. This structure provides our knapsack problem with the integrality property.

Magee and Glover (1996) state that a constraint set has the integrality property if the linear problem formed by adding any linear objective function is guaranteed to have an optimal integer solution. Thus, the objective function value of the LP relaxation of our knapsack problem also gives the optimal solution of the problem as far as the integrality property holds.

To solve this subproblem, P_{il}^{LR}, for iteration *t*, first we have to calculate the objective function coefficient of each x_{ijl} variable, that is cc_{ijl}, for known values of α_j^t and then select the minimum cc_{ijl} value for each *il* pair, that is $\overline{cc}_{ijl}$:

$$cc_{ijl} = b_{ij} + e_{il} - \alpha_j^t \qquad \forall i \in I, \forall j \in J \; and \; \forall l \in L$$

$$\overline{cc}_{il} = \underset{j \in J}{Min}\left\{cc_{ijl}\right\} \qquad \forall i \in I \; and \; \forall l \in L$$

The x_{ijl} variable that has the minimum cc_{ijl} value for an *il* pair is the best variable to minimize the objective function of the subproblem, P_{il}^{LR1}, and therefore selected to satisfy the constraints of the objective function. If $\overline{cc}_{il}$ value is negative (Case 1), then transporting goods to the relevant warehouse as much as possible is the most reasonable action. But the amount that can be transported is limited either by the capacity of the plant or the capacity of the supplied warehouse, thus the value of x_{ijl} variable that has the minimum cc_{ijl} value is set to $Min\left(v_{il}^{\max}, w_j\right)$. On the other hand, if $\overline{cc}_{il}$ has a positive value (Case 2), it means that transporting goods does not improve the objective function, hence setting x_{ijl} variable as small as possible is the best way. Due to the modular capacitated structure, there is a minimum supply requirement, $v_{il}^{\min}$ for each capacity level. Therefore, we can not just set the value of the relevant x_{ijl} to zero. In order to satisfy the constraint set (2), the x_{ijl} variable that has the minimum cc_{ijl} value is set to $Min\left(v_{il}^{\min}, w_j\right)$ even if its $\overline{cc}_{il}$ is positive.

If the value of w_j is less than the value of $v_{il}^{\max}$ for the first case or less than $v_{il}^{\min}$ for the second case, then the maximum or minimum capacity requirement is not satisfied completely. Let us assume that v_{il}^{maxrem} is the remaining amount for reaching the maximum capacity of a capacity level after assigning a value to variable x_{ijl} that has the minimum and negative cc_{ijl}, and v_{il}^{minreq} is the required amount for reaching the minimum capacity of a capacity level after assigning a value to variable x_{ijl} that has the minimum but positive cc_{ijl}.

For the first case, we select the variable x_{ijl} that has the second minimum cc_{ijl} value and assign a value $Min\left(v_{il}^{\max}, w_j\right)$ if the value of cc_{ijl} is still negative. This procedure is repeated until the maximum capacity of a capacity level is full or all remaining cc_{ijl} variables are positive and production capacity is above the $v_{il}^{\min}$.

For the second case, we know that all remaining cc_{ijl} values are positive. Assigning a value to a x_{ijl} variable that has positive cc_{ijl} is not a desirable move, but we have to keep assigning values to satisfy the minimum requirement constraint of the capacity level. Thus, we select the next x_{ijl} variable that has the minimum cc_{ijl} value among the unassigned variables and assign a value $Min\left(v_{il}^{\max}, w_j\right)$. We have to repeat this until the total production of the plant reaches the minimum limit of its capacity level.

The unassigned x_{ijl} variables actually have a worse effect on objective function of the subproblem, P_{il}^{LR1} compared to the assigned ones, and hence they are assigned to zero. After determining the values of all x_{ijl} variables, the total cost of opening a plant for a capacity level is calculated as:

$$tc_{il} = \sum_{j \in J} cc_{ijl} x_{ijl} + f_{il}\,.$$

When we calculate the total opening cost, tc_{il}, for each capacity level of each plant, our main subproblem reduces to determining the values of q_{il} variables from the following pure integer model:

$$P^{LR1}\left(\alpha\right) = Min \sum_{i \in I} \sum_{l \in L} tc_{il} q_{il}$$

Subject to

$$\sum_{l \in L} q_{il} \leq 1 \qquad \forall i \in I \qquad (3)$$

$$Q_{\min} \leq \sum_{i \in I} \sum_{l \in L} q_{il} \qquad (20)$$

$$q_{il} \in \{0,1\} \qquad \forall i \in I \text{ and } \forall l \in L \qquad (10)$$

Solving this problem is also very easy. The constraint set (3) forces that at most one capacity may be opened for a candidate plant location. Due to the fact that there is no additional constraint for selecting a capacity level for a candidate plant location, the capacity level that has the best (minimum for our instance) total opening cost, tc_{il}, value becomes the only candidate for a location. Hence the best possible candidate for a location is determined by inspecting the tc_{il} values of the relevant capacity levels. Let us assume that $\overline{tc_i}$ is the total opening cost that has the minimum opening and operating cost among all the capacity levels for plant location *i*, then it can be expressed as:

$$\overline{tc_i} = \underset{l \in L}{Min}\left\{tc_{il}\right\}.$$

At this stage, our main subproblem P^{LR1} becomes much easier to solve. If $l' \in L$ is the best candidate capacity level for plant location *i*, the main subproblem is as follows:

$$P^{LR1}(\alpha) = Min \sum_{i \in I} \overline{tc}_i \, q_{il'}$$

Subject to

$$Q_{\min} \leq \sum_{i \in I} \sum_{l \in L} q_{il} \tag{20}$$

$$q_{il} \in \{0,1\} \qquad \forall i \in I \; and \; \forall l \in L \tag{10}$$

Before solving this subproblem, we need to introduce a new set, M, which is the non-decreasing ordered set of $\overline{tc_i}$ values. Let us assume that $m \in M$ is the index of this new set M and q_{il}^{m} is the decision variable for the relevant $\overline{tc_i^{m}}$ value, then the set M fulfills the inequality below:

$$\overline{tc}_i^1 \leq \overline{tc}_i^2 \leq \ldots \leq \overline{tc}_i^m \leq \ldots \leq \overline{tc}_i, \quad where |M| = |I|$$

In order to satisfy the constraint (20), at least $Q_{\min}$ many q_{il} variables starting from the beginning of the order have to be set equal to one, even if some of them may have positive coefficients in the objective function. After opening $Q_{\min}$ many plants, if there still negative $\overline{tc_i}$ values, then the associated variables $q_{il[m]}$ are set equal to one as well. If not, no more q_{il} variables are set to one. This can be mathematically described as follows:

$$q_{il[m]} = \begin{cases} 1 & if \;\; m \leq Q_{\min} \text{ or } \overline{tc}_{i[m]} < 0 \\ 0 & \text{Otherwise} \end{cases} \qquad \forall \, m = 1, 2, \ldots, |I|$$

Since the opened plants and capacity levels are determined, we can now calculate the final values of x_{ijl} variables and the value of the objective function of the main subproblem P^{LR1}. As stated before, if a capacity level of a plant is decided as "remain closed", then no transportation occurs from this level. Therefore, the values of x_{ijl} variables are set to zero, if the relevant q_{il} variable is equal to zero; otherwise the values of x_{ijl} variables preserve their former values that are calculated in the previous stages of the algorithm.

Finally, the objective function value of the main subproblem P^{LR1} is calculated by putting either the values of x_{ijl} and q_{il} variables and their coefficients into the objective function of the subproblem or the $\overline{tc_i}$ values of the opened plants. If $\hat{x}_{ijl}$ and $\hat{q}_{il}$ represent the previously selected values of the relevant variables, then the value of our main subproblem is calculated as follows:

$$P^{LR1}(\alpha) = \sum_{i \in I} \sum_{j \in J} \sum_{l \in L} \left(b_{ij} + e_{il} - \alpha_j\right) \hat{x}_{ijl} + \sum_{i \in I} \sum_{l \in L} f_{il} \hat{q}_{il}$$

$$= \sum_{i \in I} \sum_{l \in L} \overline{tc_i}\, \hat{q}_{il}$$

The pseudo-code of the solution algorithm is as follows:

START
FOR Each plant location i and capacity level l
  SET Maximum available capacity, MC = $v_{il}^{\max}$ and
    Minimum required capacity, MR= $v_{il}^{\min}$
  Calculate the objective function coefficient of x_{ijl} variables for each warehouse j (cc_{ijl})
  WHILE MR > 0
      Select the unassigned x_{ijl} variable that has the minimum cc_{ijl} value
      Assign relevant $x_{ijl} = Minimum\left(MR, w_j\right)$
      Update MR = MR - x_{ijl}
  END WHILE
  Select the unassigned warehouse that has the minimum cc_{ijl} value
  WHILE Selected cc_{ijl} < 0 **AND** MC > 0
      Assign relevant $x_{ijl} = Minimum\left(MC, w_j\right)$
      Update MC = MC - x_{ijl}
      Select next warehouse that has the minimum cc_{ijl} value

END WHILE
Calculate total opening cost of relevant capacity level (tc_{il})
END FOR
FOR Each Plant location
SET Capacity level that has the minimum cost as $\overline{tc_i} = \underset{l \in L}{Minimum}\left(tc_{il}\right)$
IF $\overline{tc_i} < 0$ **THEN**
Open relevant capacity level
END IF
END FOR
WHILE Number of opened capacity levels < $Q_{\min}$
Open unopened capacity level that has the minimum $\overline{tc_i}$ value
END WHILE
FOR Each closed plant and capacity level
SET All relevant x_{ijl} variables to 0
END FOR
Calculate the value of $P^{LR1}\left(\alpha\right)$
STOP

4.1.2.2 Solution Methodology for the Second Main Subproblem

The second main subproblem P^{LR2} is also solved in a similar fashion, but this time there is no capacity selection procedure as there is in the main subproblem P^{LR1}, because there is only one capacity limit for each warehouse. The mathematical model of the second main subproblem is as follows:

$$P^{LR2}\left(\alpha,\beta\right) = Min\sum_{j\in J}\sum_{k\in K}\left(c_{jk}+\alpha_j-\beta_k\right)z_{jk}+\sum_{j\in J}g_j r_j$$

Subject to

$$\sum_{k\in K} z_{jk} \le w_j r_j \qquad \forall j \in J \tag{4}$$

$$\sum_{j\in J} r_j \le R_{\max} \tag{6}$$

$$\sum_{j\in J} r_j \ge R_{\min} \tag{21}$$

$$0 \le z_{jk} \le Min\left(d_k, w_j\right) \, and \text{ integer} \qquad \forall j \in J \text{ and } \forall k \in K \tag{9}$$

$$r_j \in \{0,1\} \qquad \forall j \in J \tag{11}$$

We can also solve this main subproblem for each warehouse separately by enforcing the constraint (6). We solve each subproblem corresponding to a warehouse without considering the hard constraint (6), but at the end, the model is forced to fit this neglected constraint in order to get the optimal solution for the main subproblem P^{LR2}. The mixed integer programming model of the subproblem P_j^{LR2} corresponding to warehouse j is as follows:

$$P_j^{LR2}(\alpha,\beta) = Min\sum_{k\in K}\left(c_{jk}+\alpha_j-\beta_k\right)z_{jk}+g_j r_j$$

Subject to

$$\sum_{k\in K} z_{jk} \le w_j r_j \tag{4}$$

$$0 \le z_{jk} \le Min\left(d_k, w_j\right) and \text{ integer} \qquad \forall k \in K \tag{9}$$

$$r_j \in \{0,1\} \tag{11}$$

In the subproblem P_j^{LR2}, the decision variable r_j is binary variable. Either the warehouse is decided to be opened thus the corresponding r_j variable is set to one, or it is decided to be "remain closed" by setting r_j variable equal to zero. If it is decided to be "remain closed", then handling products in this warehouse or supplying goods for satisfying the demand of a customer is not allowed. Thus if r_j is equal to zero, all the related z_{jk} variables are also set to zero. But if it is decided to be opened, our problem then is to determine the valuable customers that provide the minimum objective function. If we assume that $r_{j'}$ is equal to one, then we end up with the following knapsack problem for this opened warehouse:

$$P_{j'}^{LR2}(\alpha,\beta) = Min \sum_{k \in K} \left(c_{j'k} + \alpha_{j'} - \beta_k\right) z_{j'k} + g_{j'}$$

Subject to

$$\sum_{k \in K} z_{j'k} \le w_{j'} \tag{4}$$

$$0 \le z_{j'k} \le Min\left(d_k, w_{j'}\right) and \text{ integer} \qquad \forall k \in K \tag{9}$$

This problem is a minimization type, unweighted, bounded knapsack problem and as the first subproblem, and due to its special structure that is explained in the first main subproblem, this subproblem has integrality property, too. Therefore, the value of the LP relaxation of this special knapsack problem equals its optimal value because of the integrality property.

In order to determine the best customers to serve for a warehouse, the coefficients of the objective function are calculated for each customer. Let us assume that ccc_{jk} is the cost coefficient of the customer *k* supplied from the warehouse *j*, and $\overline{ccc_j}$ is the lowest ccc_{jk} value, corresponding to the most valuable customer supplied from warehouse *j*. Then their values are calculated as follows:

$$ccc_{jk} = c_{jk} + \alpha_j - \beta_k$$

$$\overline{ccc_j} = \underset{k \in K}{Min}\left\{ccc_{jk}\right\}$$

The z_{jk} variable that has the minimum ccc_{jk} value, $\overline{ccc_j}$ for a warehouse *j* is the best variable so as to minimize the objective function of the subproblem, P_j^{LR2}. If the value of $\overline{ccc_j}$ is greater than or equal to zero, it means that there is no valuable customer for this warehouse. In other words, if $\overline{ccc_j}$ is positive, serving a customer worsens the objective function value of the problem. In a situation like this, the best action is not serving any customer from this warehouse *j* and setting all z_{jk} variables equal to zero.

If one or more customers that have negative ccc_{jk} values exist, we have to select the more valuable ones among these customers, because the handling capacity of the warehouse is limited and may not serve all these valuable customers. In order to determine the more valuable customers, we have to re-sort the ccc_{jk} variables of a warehouse in a non-decreasing order. Let *m* be the index for the position in the order, then $ccc_{jk[m]}$ values have to satisfy the following condition:

$$ccc_{jk[1]} \le ccc_{jk[2]} \le \ldots \le ccc_{jk[m]} \le \ldots \le ccc_{jk[|K|]}, \quad where\ m = 1, 2, \ldots, |K|$$

The first customer is the most valuable customer for the relevant warehouse *j*. Hence this customer has to be served in the first place, if $ccc_{jk[1]}$ is negative. The transported amount between this warehouse-customer pair is determined by considering their capacities. So the z_{jk} variable is equal to either the capacity of this warehouse w_j, or the demand of the customer, d_k. If w_j is less than or equal to d_k, then the variable z_{jk} is set equal to w_j, otherwise the variable z_{jk} is set equal to d_k. If the transported amount is equal to the demand of the customer, it means that the warehouse still has the capacity to serve other customers. Hence other customers can be accepted for serving according to the ordering of the customers, until the capacity of the warehouse is full or all the remaining customers have positive ccc_{jk} values. Let us assume that w_j^{rem} is the remaining capacity of the warehouse j after serving some customers that have negative ccc_{jk} values. The transported amounts between the warehouse and the selected customers to be served are either equal to the remaining capacity of the warehouse, w_j^{rem} or equal to the demand of the customer d_k.

The unassigned customers are worthless customers for the related warehouse and the z_{jk} variables related to these customers are set equal to zero. After determining all values of z_{jk} variables of a warehouse, the total cost of opening this warehouse is calculated as follows:

$$wc_j = \sum_{k \in K} ccc_{jk} z_{jk} + g_j$$

When we calculate the total opening cost wc_j, for each warehouse separately, our main subproblem turns into determining the values of r_j which can be formulated as a pure integer model:

$$P^{LR2}(\alpha, \beta) = Min \sum_{j \in J} wc_j r_j$$

Subject to

$$\sum_{j \in J} r_j \leq R_{\max} \tag{6}$$

$$\sum_{j \in J} r_j \geq R_{\min} \tag{21}$$

$$r_j \in \{0,1\} \qquad \forall j \in J \tag{11}$$

Now the problem becomes much easier to solve. The constraint sets (6) and (21) restrict the number of opened warehouses in a tight interval. In order to satisfy these constraints, the number of the opened plants has to be greater than or equal to $R_{\min}$ and less than or equal to $R_{\max}$. In order to solve this problem we have to re-sort the wc_j values in a non-decreasing way where *m* is the index for showing the position in the ordering:

$$wc_{j[1]} \leq wc_{j[2]} \leq \ldots \leq wc_{j[m]} \leq \ldots \leq wc_{j[|J|]}, \quad where\ m = 1, 2, \ldots, |J|.$$

Starting from the beginning of this ordering at least $R_{\min}$ many warehouses have to be opened without considering their $wc_{j[m]}$ values. Then, if there still exists negative $wc_{j[m]}$ values, the relevant warehouses are also opened until the number of the total opened warehouses is equal to $R_{\max}$. The mathematical expression for this selection is as follows:

$$r_j = \begin{cases} 1 & \text{if } m \le R_{\min} \text{ or } \left(wc_{j[m]} < 0 \text{ and } m \le R_{\min} \right) \\ 0 & \text{Otherwise} \end{cases} \qquad \forall\, m = 1, 2, \ldots, |J|$$

Since the opened warehouses are determined, we are free to calculate the final values of z_{jk} variables and the value of the objective function of the main subproblem P^{LR2}. The values of z_{jk} variables are set to zero, if the related r_j variable is equal to zero. If the r_j variable is equal to one, then the related z_{jk} variables preserve their former values that are calculated in the previous stages of the algorithm.

Finally the objective function value of the main subproblem P^{LR2} is calculated by putting the previously found values of r_j and z_{jk} variables, $\hat{r}_j$ and $\hat{z}_{jk}$, into the objective function of the subproblem or only multiplying the wc_j values of the opened warehouses.

$$\begin{aligned} P^{LR2}(\alpha, \beta) &= \sum_{j \in J} \sum_{k \in K} \left(c_{jk} + \alpha_j - \beta_k \right) \hat{z}_{jk} + \sum_{j \in J} g_j \hat{r}_j \\ &= \sum_{j \in J} wc_j \hat{r}_j \end{aligned}$$

The pseudo-code of the solution algorithm is as follows:

START
FOR Each warehouse
    SET Maximum Capacity, MC, equal to w_j
    Calculate the objective function coefficient of z_{jk} variables for each customer (ccc_{jk})
    Select the unassigned customer that has the minimum ccc_{jk} value
    WHILE Selected ccc_{jk} < 0 **AND** MC > 0
        Assign relevant z_{jk} = Minimum (MC, d_k)
        Update MC = MC - z_{jk}
        Select the next customer that has the minimum ccc_{jk} value
    END WHILE
    Calculate total opening cost of relevant capacity level (wc_j)
    IF wc_j < 0 **THEN**
        Open the relevant warehouse
    END IF
END FOR
WHILE Number of opened warehouses < $R_{\min}$
        Open unopened warehouse that has the minimum wc_j value
END WHILE
WHILE Number of opened capacity levels > $R_{\max}$
        Close the opened warehouse that has the maximum wc_j value
END WHILE
FOR Each closed warehouse
    SET All relevant z_{jk} variables to 0
END FOR
Calculate the value of $P^{LR2}\left(\alpha,\beta\right)$
STOP

4.1.2.3 Solution Methodology of the Lagrangean Dual Problem

Even though no optimization software is used for solving these main subproblems, our procedure yields optimal results for all subproblems by exploiting their special structure. Otherwise, if we employ a solution procedure that cannot yield optimal solutions for subproblems, then:

i. it generates infeasible results that have lower objective function value than the optimal solution
ii. it generates feasible results that have higher objective function value than the optimal solution

iii. it generates mixed results that both contains lower and higher objective function value than the optimal solution.

The second and the third solution procedures are not acceptable in Lagrangean relaxation for generating a proper lowerbound, because if they are employed in the solution, the lowerbound may exceed the optimal solution of the original problem P. The first solution procedure can be employed in Lagrangean relaxation, especially if the main subproblems are still difficult to solve, but it is not recommended unless it is mandatory, because employing it yields bigger gaps between the lower and upperbounds.

As mentioned before, the Lagrangean dual problem is the problem of finding the maximum value of the relaxed problem with given Lagrange multiplier values which are separately calculated by the subgradient optimization. Due to some properties of the subgradient optimization that are explained in the next section, the lowerbound value found in each iteration is not monotonically increasing or decreasing. It means that the solution of the Lagrangean dual problem is not always the value of the Lagrangean relaxation of the last iteration, because of the zigzagging pattern of the subgradient optimization. If $P^{LR}\left(\alpha^t, \beta^t\right)$ denotes the objective function value of the Lagrangean relaxation problem for TSMCFLP at iteration t with known α^t *and* β^t values, then it is calculated as:

$$P^{LR}\left(\alpha^t, \beta^t\right) = P^{LR1}\left(\alpha^t\right) + P^{LR2}\left(\alpha^t, \beta^t\right) + \sum_{k \in K} \beta_k^t .$$

And the solution of the Lagrangean dual problem, also called as the lowerbound value of the original problem, is determined by selecting the maximum $P^{LR}\left(\alpha^t, \beta^t\right)$ value among the others. Let LB be the lowerbound of our problem among $t \in T$ iterations, then its value is found as follows:

$$LB = \underset{t \in T}{Max}\left\{P^{LR}\left(\alpha^t, \beta^t\right)\right\} .$$

4.2 Primal Heuristic

Usually, the results of the relaxed problem may turn out to be infeasible for the original problem. Even if they are feasible, they may not be the optimal solution of the original problem. Thus an algorithm has to be employed in order to find the optimal or a near-optimal solution for the problem. In early studies of Lagrangean relaxation, after solving the Lagrangean dual problem, the remaining gap between the optimal solution and the solution of the Lagrangean dual problem, which is also called as the duality gap, is closed by using a combinatorial optimization technique such as the branch-and-bound, branch-and-price or branch-and-cut.

Normally, the branch-and-bound and similar procedures use the solution of the linear programming (LP) relaxation of the model as an initial solution. Early stage studies had discovered that using Lagrangean relaxation instead of LP relaxation usually yields much better initial solutions and started to treat the Lagrangean relaxation as an initiation phase of an exact algorithm. For the first time, Pirkul (1987) proposes an interactive primal heuristic that fixes some decision variables of the original problem with the results obtained in the relaxed problem and determines the rest of the decision variables in the problem by an efficient heuristic. This is an interactive approach, because unlike the branch-and-bound procedure that is executed after the termination of the Lagrangean dual problem, the new primal heuristic is executed repeatedly in each iteration, after finding the solution of the relaxed problem.

Later, it is seen that the Lagrangean relaxation not only provides an initial solution of an exact algorithm, but also an important part of the efficient heuristic which is called as the Lagrangean heuristics. Due to its nature, this heuristic does not seek the optimal solution, but a near-optimal solution. The optimal results may not be obtained using Lagrangean heuristics. On the other hand, using heuristic instead of exact algorithms has serious advantages; the computational effort and the solution time is decreased significantly and obtaining both an upperbound and a lowerbound in an iteration provides us to check the quality of the solution in the current state.

The most important part of the Lagrangean heuristic is definitely the primal heuristic component, because only its results are always feasible for the original problem, hence only its results can be applicable in real world. Pirkul and Jarayaman (1998) extend the primal heuristic of Pirkul (1987) to the two-stage location problem. Since then the variants of this primal heuristic have been used in similar studies. For the reason that there is no alternative heuristic in the literature competing with heuristic presented by Pirkul (1987)), we also use a similar heuristic for generating feasible results.

As mentioned before, the distribution network design problems are location-allocation type problems. In this type of problems, the locations of the opened facilities are determined and the customers are allocated to the opened facilities simultaneously. Deciding locations and allocations simultaneously, makes this problem much difficult to solve. On the other hand, allocation of customers to the facilities formerly determined is a less difficult problem than the simultaneous problem. Pirkul's (1987) primal heuristic is based on this logic. In the heuristic, the location decisions are obtained from the solution of the relaxed problem. In other words, the given decisions about the opened facilities in the relaxed problem are transferred into the primal heuristic so that the heuristic only deals with the allocation problem of the original problem.

In Pirkul's (1987) heuristic only the locations of the facilities are obtained from the lowerbound, but not their capacities. The capacities of the facilities are determined after executing the allocation heuristic, which is inapplicable for our problem, because we have more than one available maximum and minimum capacity levels each with a unique cost. For this reason, we obtain not only the locations of the opened plants but also the selected capacity levels for the opened plants from the lowerbound solution. This makes our primal heuristic more complicated than Pirkul (1987) and Pirkul and Jarayaman (1998).

The proposed primal heuristic consists of three phases. First, a greedy heuristic is developed for making the infeasible location decisions feasible. Secondly, the allocation problem is solved efficiently using an allocation heuristic, and finally at the third phase feasibility is checked for each plant. Details of these three phases are presented below.

4.2.1 Greedy Heuristic

The greedy heuristic is a heuristic that always takes the immediate best or local optimum solution while finding an answer at each stage with the hope of finding the global optimum. Greedy algorithms find the overall or global optimal solution for some optimization problems, but may find suboptimal solutions for some instances of other problems. The heuristic selects the best choice in a situation and then solves the problems that arise later. The choice made by a greedy algorithm may depend on prior decisions made but not on the future choices or all the solutions to the subproblem. It iteratively makes one greedy choice after another, reducing each given problem into a smaller one. In other words, a greedy algorithm never attempts an improvement on the solutions found. In spite of its drawbacks, the greedy heuristic is widely used in the previous studies; it has the advantages of being extremely fast and producing reasonably efficient solutions.

4.2.1.1 Plant Greedy Heuristic

We use a greedy algorithm to generate feasible solutions for the location problem which is not in the scope of the primal heuristic. We have already mentioned that in the primal heuristic, the location problem is removed from the original problem and the opened capacities and warehouses are obtained from the solution of the relaxed problem. However, the plant and warehouse decisions may be infeasible for the original problem, because the demand satisfaction constraints are dualized in the relaxed problem. Hence, at any iteration, the total capacity of the opened plants with specific capacity levels can be in one of the three different cases:

(1) $\sum_{i\in I}\sum_{l\in L} v_{il}^{\min} q_{il} \le \sum_{k\in K} d_k \le \sum_{i\in I}\sum_{l\in L} v_{il}^{\max} q_{il}$

(2) $\sum_{i\in I}\sum_{l\in L} v_{il}^{\max} q_{il} < \sum_{k\in K} d_k$

(3) $\sum_{i\in I}\sum_{l\in L} v_{il}^{\min} q_{il} > \sum_{k\in K} d_k$

CASE 1: In the first case, the total minimum capacity of the opened plants is less than the total demand and the total maximum capacity of the opened plants is greater than or equal to the total demand. In this case, the global feasibility of the original problem is achieved. We call this situation as "global feasibility", because this feasibility deals with the total capacities. In addition to the global infeasibility case there is another infeasibility that we may encounter after solving the allocation heuristic, which we call as local infeasibility.

Local infeasibility emerges if one or more plants distribute less than their minimum supply constraints. Local infeasibility is checked in the third phase of the heuristic procedure and explained in detail in further sections.

If the result of the lowerbound is in the state of global feasibility, then there is nothing we should do. The allocation heuristic can be executed directly for making feasible allocations for the original problem.

CASE 2: In the second case, the total maximum capacity is less than the total demand. It means that the total production capacity is not enough for satisfying the demand of all customers. In order to satisfy the whole demand, one or more plants may be opened or capacity levels of the opened plants may be increased or both of these remedies may be implemented. In order to decide the set of plants/capacity levels that are decided to be opened, we calculate the opening cost OC_{il}, for each possible alternative. For the plants which do not have an opened capacity level, we calculate the opening cost only for the first capacity level; in other words, only the opening cost of the first capacity level OC_{i1} is considered as a possible alternative if no capacity level is already opened in a specific location. For this type of locations opening cost is formed by the fixed cost of opening the first capacity level of the location and variable cost of operating this capacity level at the allowed maximum:

$$OC_{i'1} > 0 \quad if \; q_{i'l} = 0 \quad \forall l \in L$$
$$OC_{i'1} = f_{i'1} + e_{i'1} v_{i'1}^{\max}$$

If there is an opened capacity level for a plant and the current capacity level is not the highest capacity level of this plant, then OC_{il} is calculated for only the next capacity level. If the next capacity level is decided to be opened, then the previous capacity level of the plant has to be closed. Therefore, the opening cost is calculated as the fixed cost of opening the relevant capacity level and variable cost of operating at the maximum level for this capacity minus the fixed and variable cost of operating at the lower capacity level.

$$OC_{i'l'} > 0 \quad if \; q_{i'l'-1} = 1 \; and \; l' \neq 1$$
$$OC_{i'l'} = \left[f_{i'l'} + e_{i'l'} \left(v_{i'l'}^{\max} - v_{i'l'}^{\min} \right) \right] - \left[f_{i'l'-1} + e_{i'l'-1} \left(v_{i'l'-1}^{\max} - v_{i'l'-1}^{\min} \right) \right]$$

If the capacity level/plant that has the minimum opening cost is high enough to satisfy the demand shortage of customers, then it is selected as the opened plant/capacity level. Otherwise we develop three alternative solutions in an attempt to obtain the best solution. We execute the three solutions whenever the capacity level/plant that has minimum opening cost is not high enough to satisfy the demand shortage of customers. After execution, the solution that has the lowest total opening cost is selected and the plants/capacity levels are opened with respect to the results of the selected solution.

These three alternative solutions can be described as follows:

- opening a plant/capacity level that is big enough to satisfy the demand shortage

- opening a plenty of plants/capacities that have the minimum ratio of opening cost, rOC_{il}, until the total capacity of these plants/capacities satisfies the demand shortage

- opening a plenty of plants/capacities until the total capacity of these plants/capacities satisfies the demand shortage according to the minimum opening cost, OC_{il}.

For calculating the ratio of the opening cost, rOC_{il}, we divide OC_{il} values to $\left(v_{il}^{\max} - v_{il}^{\min}\right)$.

$$rOC_{il} = \frac{OC_{il}}{v_{il}^{\max} - v_{il}^{\min}}$$

In order to calculate the total opening cost, $\overline{OC}_2$, of the capacity set determined according to rOC_{il} values of the capacities, we have to re-sort the rOC_{il} values in a non-decreasing order. Let *m* be the index that indicates the position in the ordering, then the ordering is as follows:

$$rOC_{il[1]} \leq rOC_{il[2]} \leq \ldots \leq rOC_{il[m]} \leq \ldots \leq rOC_{il[|M|]} \qquad where\,|M| \leq |I|$$

Similarly for calculating the total opening cost, $\overline{OC}_3$, of the capacity set determined with respect to OC_{il} values of the capacities, we have to re-sort the OC_{il} values in a non-decreasing order. Let *n* be the index that indicates the position in the ordering, then the ordering is as follows:

$$OC_{il[1]} \leq OC_{il[2]} \leq \ldots \leq OC_{il[n]} \leq \ldots \leq OC_{il[|N|]} \qquad where\,|N| \leq |I|$$

Then the costs of these solution alternatives $\overline{OC}_1, \overline{OC}_2$ *and* $\overline{OC}_3$ are calculated as follows:

$$\overline{OC}_1 = \underset{i \in I, l \in L \ni q_{il}=0 \, and \, v_{il}^{\max} - v_{il-1}^{\max} \geq Shortage}{Min} \left\{ OC_{il} \right\} .$$

$$\overline{OC}_2 = \sum_{m \in M} rOC_{il[m]} \left(v_{il[m]}^{\max} - v_{il[m]}^{\min} \right) \qquad where \sum_{m \in M} \left(v_{il[m]}^{\max} - v_{il[m]}^{\max} \right) \geq shortage.$$

$$\overline{OC}_3 = \sum_{n \in N} OC_{il[n]} \qquad where \sum_{n \in N} \left(v_{il[n]}^{\max} - v_{il[n]}^{\max} \right) \geq shortage.$$

It should be noted that, while calculating $\overline{OC}_2$ *and* $\overline{OC}_3$, if a capacity level is selected to be opened, then automatically the opening cost, OC_{il}, for the next capacity level and its ratio, rOC_{il}, is calculated, unless the recently opened capacity level is the highest capacity level of the plant. The rOC_{il} *and* OC_{il} values are re-sorted considering the new values. Furthermore, after determining the sets of opening plants for each alternative solution, a post optimization procedure is employed and the total opening cost is tried to be decreased by removing one or more unnecessary plants.

After calculating the costs of these three alternatives, $\overline{OC}_1, \overline{OC}_2$ *and* $\overline{OC}_3$, the one with the lowest cost is selected and the plants/capacity levels which are decided to be opened in the selected alternative are opened.

CASE 3: In this case, the total minimum capacity of the opened plants is greater than the overall demand. It shows there is excess capacity in the solution. This surplus capacity leads to local infeasibilities in one or more plants if allocation heuristic is executed without making any change. In order to prevent the local infeasibilities, one or more plants/capacity levels have to be closed. In order to determine the plants/capacity levels that are to be closed, a heuristic is employed similar to the one in case 2. But this time, instead of calculating the opening cost for the closed plants/capacities, the closing profit CP_{il} is calculated for the opened capacity levels. If the capacity level is the first level, then $CP_{i'1}$ is calculated as:

$$CP_{i'1} > 0 \quad if\ q_{i'1} = 1$$
$$CP_{i'1} = f_{i'1} + e_{i'1} v_{i'1}^{\max} \quad .$$

If the opened capacity level of a plant is higher than the first level, the value of CP_{il} is calculated as the cost of closing current level minus the cost of opening the prior capacity level:

$$CP_{i'l'} > 0 \quad if\ q_{i'l'} = 1\ and\ l' \neq 1$$
$$CP_{i'l'} = \left[f_{i'l'} + e_{i'l'} \left(v_{i'l'}^{\max} - v_{i'l'}^{\min} \right) \right] - \left[f_{i'l'-1} + e_{i'l'-1} \left(v_{i'l'-1}^{\max} - v_{i'l'-1}^{\min} \right) \right] \ .$$

If the capacity level that has the highest closing profit is big enough to eliminate the surplus, then it is closed. Otherwise the set of plants or capacity levels that are to be closed is selected in the same way as the Case 2, but this time the plants/capacity levels that have the maximum rCP_{il} or CP_{il} are chosen for closing.

$$\overline{CP}_1 = \underset{i,l \ni v_{il}^{\max} - v_{il-1}^{\max} \geq Surplus}{Max} \left\{ CP_{il} \right\}$$

$$\overline{CP}_2 = \sum_{m \in M} rCP_{il}^{m} \left(v_{il[m]}^{\max} - v_{il[m]}^{\min} \right) \qquad where \sum_{m \in M} \left(v_{il[m]}^{\min} - v_{il[m-1]}^{\min} \right) \geq surplus$$

$$\overline{CP}_3 = \sum_{n \in N} CP_{il[n]} \qquad where \sum_{n \in N} \left(v_{il[n]}^{\min} - v_{il[n-1]}^{\min} \right) \geq surplus$$

The pseudo-code of the solution algorithm is follows:

```
START
Calculate Total Maximum Capacity AND Total Minimum Capacity
IF      Total Maximum Capacity < Total Demand THEN
        SET     Shortage = Total Demand - Total Maximum Capacity
        FOR     Each plant location
                IF      No Capacity level is opened in the plant THEN
```

Calculate opening cost (OC_{il}) for the first capacity level which is not in tabu list

ELSE

Calculate opening cost (OC_{il}) for upgrading opened capacity to next capacity level which is not in tabu list

END IF

END FOR

IF Capacity level that has the minimum OC_{il} is big enough to satisfy the Shortage **THEN**

Close the previous capacity level and open relevant capacity level

ELSE

Select the capacity levels which are big enough to satisfy the Shortage

Select the capacity level that has the minimum OC_{il} in this set

SET $\overline{OC}_1$ value as the relevant OC_{il} value

Calculate rOC_{il} values

WHILE Total Maximum Capacity < Total Demand

Select the capacity level that has the minimum rOC_{il} value

Close the previous capacity level and open the relevant capacity level

Calculate OC_{il} and rOC_{il} values for the next capacity level which is not in tabu list

Update $\overline{OC}_2 = \overline{OC}_2 + OC_{il}$

Update Total Maximum Capacity

Store recently opened capacity levels

END WHILE

SET Surplus = Total Maximum Capacity - Total Demand

WHILE One or more recently opened plants' capacity < Surplus

Calculate closing profit (CP_{il}) for relevant capacity levels

Select the capacity level that has the maximum CP_{il}

Close relevant capacity level and open previous one

Update the Surplus value

END WHILE

Reset the value of Total Maximum Capacity

WHILE Total Maximum Capacity < Total Demand

Select the capacity level that has the minimum OC_{il} value

Close the previous capacity level and open the relevant capacity level

Calculate OC_{il} value for next capacity level which is not in tabu list

Update $\overline{OC}_3 = \overline{OC}_3 + OC_{il}$

Update Total Maximum Capacity

Store recently opened capacity levels

END WHILE

WHILE One or more recently opened plants' capacity < Surplus

Calculate closing profit (CP_{il}) for relevant capacity levels

Select the capacity level that has the maximum CP_{il}

Close relevant capacity level and open previous one

Update the Surplus value

END WHILE

Compare the $\overline{OC}_1, \overline{OC}_2$ *and* $\overline{OC}_3$ values and select the minimum one

Open the new capacity levels, and close the older ones according to selected $\overline{OC}$

END IF

ELSE IF Total Minimum Capacity > Total Demand **THEN**

SET Surplus = Total Minimum Capacity - Total Demand

Calculate closing profit (CP_{il}) for the opened capacity levels that closing it does not disrupt the Case 1

IF Capacity level that has the maximum CP_{il} is big enough to satisfy the Surplus **THEN**

Close the relevant capacity level and open the previous capacity level

ELSE

Select the capacity levels which are big enough to satisfy the Surplus

Select the capacity level that has the minimum CP_{il} in the ordering

SET $\overline{CP}_1$ value as the relevant CP_{il} value

Calculate rCP_{il} values

WHILE Total Minimum Capacity > Total Demand

Select the capacity level that has the maximum rCP_{il} value

Close the relevant capacity level and open the previous capacity level

Calculate CP_{il} and rCP_{il} values for new capacity level if available

Update $\overline{CP}_2 = \overline{CP}_2 + CP_{il}$

Update Total Minimum Capacity

Store recently opened capacity levels

END WHILE

Reset the value of Total Minimum Capacity

WHILE Total Minimum Capacity > Total Demand

Select the capacity level that has the maximum CP_{il} value

Close the relevant capacity level and open the previous capacity level

Calculate OC_{il} value for new the capacity level if available

Update $\overline{CP}_3 = \overline{CP}_3 + CP_{il}$

Update Total Minimum Capacity

Store recently opened capacity levels

END WHILE

Compare the $\overline{CP}_1, \overline{CP}_2$ *and* $\overline{CP}_3$ values and select the maximum one

Close the older capacity levels, and open the new ones according to selected $\overline{CP}$

END IF

END IF

STOP

4.2.1.2 Warehouse Greedy Heuristic

As stated before, the constraint (21) in the second echelon is more successful than the constraint (20) which is in the first echelon. But still, the total capacity of the warehouses first obtained from the solution of the relaxed problem may be infeasible. We can come across with two cases:

(1) $\sum_{j \in J} w_j r_j \geq \sum_{k \in K} d_k$

(2) $\sum_{j \in J} w_j r_j < \sum_{k \in K} d_k$

In the first case, the total capacity of the warehouses is greater than or equal to the total demand. It means there is enough capacity to satisfy the demand of all customers. There is nothing we should do if the solution of the relaxed problem is feasible for the warehouses. The allocation heuristic can be executed directly for making feasible allocations for the original problem.

In the second case, one or more warehouses have to be opened in order to satisfy the total demand. In order to determine the warehouses that are going to be opened by the greedy heuristic, we use the fixed cost of opening the warehouse, g_j.

If the closed warehouse that has the minimum g_j value can also satisfy all the shortage by itself, then it is selected as the warehouse to be opened. Else $\overline{WOC_1}$ *and* $\overline{WOC_3}$ values are calculated in the same as the $\overline{OC_1}$, *and* $\overline{OC_3}$ values are calculated.

It can be easily predicted that warehouse greedy heuristic takes more computational time than the plant greedy heuristic, because there always exists more warehouses in the problem. So, in order to gain some computational time, we do not employ calculation of $\overline{WOC_2}$ value that uses the ratio of opening cost in the warehouse greedy heuristic. Because most of the time, opening at most two warehouses are sufficient due to the existence of the constraint (21) which is a strong valid inequality for the second echelon. In this situation, opening two warehouses that have least total costs is better than opening two warehouses that have least unit costs. The pseudo-code of the greedy heuristic is as follows:

```
START
Calculate Total Capacity
IF      Total Capacity < Total Demand THEN
        SET     Shortage = Total Demand - Total Capacity
        IF      The warehouse that has the minimum g_j is big enough to satisfy the
                Shortage THEN
                Open relevant warehouse
        ELSE
                Select the warehouses which are big enough to satisfy the Shortage
                Select the warehouse that has the minimum g_j in this set as WOC̄1
                WHILE Total Capacity < Total Demand
                        Select the warehouse that has the minimum g_j value
                        Open the relevant warehouse
                        Update WOC̄3 = WOC̄3 + WOC_il
                        Update Total Capacity
                        Store recently opened capacity levels
                END WHILE
                SET     Surplus = Total Capacity - Total Demand
                WHILE   A recently opened warehouse's capacity < Surplus
                        Calculate closing profit (WCP_j) for relevant warehouses
                        Select the warehouse that has the maximum WCP_j
                        Close relevant warehouse
                        Update the Surplus value
                END WHILE
                Compare the WOC̄1, and WOC̄3 values and select the minimum one
                Open the new warehouses according to selected OC̄
        END IF
END IF
STOP
```

4.2.2 Allocation Heuristics

4.2.2.1 Warehouse Allocation Heuristic

After global feasibility is obtained for both echelons using the developed greedy heuristics, we can proceed by assigning the customers to the opened warehouses.

Although the warehouses are capacitated, due to the allowance of multiple sourcing, the assignment is relatively easier. Every customer is to be assigned to the opened warehouse that minimizes the transportation cost of the customer. But it is not possible in all instances due to the capacities of the warehouses. So we have to specify a penalty value for each customer and assign these customers to the warehouses in an order that is related to their penalty values. We define the penalty value of a customer as the extra cost of assigning a customer to its second cheapest opened warehouse instead of the cheapest opened warehouse. Let WAC_{jk} be the warehouse assignment cost of the customer k for the warehouse *j*, then the values of WAC_{jk} and the penalty value, $PWAC_{jk}$ are calculated as follows:

$$WAC_k = c_{jk} d_k$$

$$PWAC_k = \underset{j' \in J'}{Min}\left\{WAC_{j'k}\right\} - 2^{nd}\underset{j' \in J''}{Min}\left\{WAC_{j''k}\right\} \qquad J' = \left\{ j' \in J \mid r_{j'} = 1 \right\}$$

Then the penalties of the customers are re-sorted in a non-increasing way. Let index *m* denote the position in the ordering:

$$PWAC_{j[1]} \geq PWAC_{j[2]} \geq \ldots \geq PWAC_{j[m]} \geq \ldots \geq PWAC_{j[|M|]}$$

Thus, starting from the customer that has the highest penalty value (in position 1), the customers are assigned to the cheapest warehouses. While assigning a customer to a warehouse that has still handling capacity left, one of the following two cases may occur:

1. The remaining capacity of the cheapest warehouse, $w_{j'}^{remain}$ is greater than the demand of the customer, $w_{j'}^{remain} \geq d_k$
2. The remaining capacity of the cheapest warehouse is less than or equal to the demand of the customer, $w_{j'}^{remain} < d_k$.

In the first case, the warehouse has still got enough capacity to serve the whole demand of the relevant customer. So the whole demand of this customer is assigned to the mentioned warehouse and the remaining capacity of the warehouse w_j^{remain} is updated. At this point, the assigned customer and its penalty are removed from the penalty set and the next customer that has the highest penalty value in the list is selected to be assigned to a warehouse.

$$z_{jk} = \begin{cases} d_k & \text{If } w_j^{remain} \geq d_k \\ w_j^{remain} & \text{Otherwise} \end{cases}$$

$$w_j^{remain} = w_j - \sum_{k \in K} z_{jk}$$

In the second case, the remaining capacity of the warehouse is not enough for satisfying the overall demand of the customer. Fortunately, multiple sourcing is allowed. Therefore, the maximum amount that the relevant warehouse can accept is assigned to the warehouse, and then the remaining demand of the customer is calculated as the total demand of the customer minus the assigned demand of this customer.

$$d_k^{remain} = d_k - \sum_{j \in J} z_{jk} \ .$$

At this stage, the associated warehouse has no remaining capacity. So it is closed for further demands and the penalty values are re-calculated by ignoring the fully loaded warehouses in the calculation of the WAC_{jk} values.

Then again the customer that has the maximum penalty value is selected, and it is assigned either fully or partially to a warehouse. This procedure is repeated until all customer demands are fully satisfied by the warehouses. The pseudo-code of the allocation heuristic is as follows:

START
SET Remaining warehouse capacity to w_j for opened warehouses
WHILE There exists one or more unassigned demand of customer
Calculate warehouse assignment costs (WAC_{jk}) for each unassigned customer – opened but not full warehouse pair
Calculate the penalty value ($PWAC_k$) for each unassigned customer
Select the customer that has the highest $PWAC_k$ value
Select the opened but not full warehouse that the selected customer has lowest WAC_{jk} value
IF Customer's unallocated demand < remaining warehouse capacity **THEN**
Assign all demand to relevant warehouse
Update the remaining warehouse capacity
Mark relevant customer demand as assigned
ELSE
Assign the maximum available demand to relevant warehouse
Update the unassigned demand amount
Mark relevant warehouse as full
END IF
END WHILE
IF No customer is assigned to an opened warehouse **THEN**
Close relevant warehouse
END IF
STOP

4.2.2.2 Plant Allocation Heuristic

After assigning all customers to the opened warehouses, the capacities of the warehouses, w_j^{actual}, are determined by adding all the demand of customers that are assigned to relevant warehouses:

$$w_j^{actual} = \sum_{k \in K} d_k z_{jk} \qquad \forall\ j \in J\ .$$

It should be noted that even if a warehouse is decided to be opened in the lowerbound solution, but its calculated capacity w_j^{actual} is equal to zero, then there is no need to keep this warehouse open in the upperbound, thus it is decided to be closed and the relevant r_j variable is set to zero:

$$r_j = \begin{cases} 1 & \text{If } w_j^{actual} > 0 \\ 0 & Otherwise \end{cases} \qquad \forall j \in J\ .$$

Assigning opened warehouses to the opened plants is much difficult than assigning customers to the opened warehouses due to the existence of the minimum supply requirement constraints of the capacity levels. As far as we know, there are no heuristic solutions for the plants which consider both the maximum and minimum supply limits of the plants. Even we develop a heuristic for this case; it may be far from the optimal solution. So we have decided to ignore the minimum flow constraints of the plants in the allocation heuristic and deal with this complexity in the next stage that is the local feasibility check stage. Without the minimum supply constraints, the allocation problem can be solved efficiently by using a heuristic that is similar to the warehouse allocation heuristic explained above.

First, for each opened plant and warehouse pair, the plant assignment cost PAC_{ijl}, which is the cost of assigning the whole demand of a warehouse to a specific plant, is calculated as:

$$PAC_{i'j'l'} = \left(c_{i'j'} + e_{i'l'}\right) w_{j'}^{actual} \qquad whre\ q_{i'l'} = 1\ and\ r_{j'} = 1$$

Then a penalty is calculated for each warehouse; that is the extra cost of assigning a warehouse to the second cheapest plant is as follows:

$$PPAC_j = \underset{i',l' \in IL'}{Min}\left\{PAC_{i'jl'}\right\} - 2^{nd}\underset{i',l' \in IL'}{Min}\left\{PAC_{i'jl'}\right\} \qquad IL' = \left\{i' \in I\ and\ l' \in L \mid q_{i'l'} = 1\right\}$$

Then the penalties of the opened warehouses are re-sorted in a non-increasing way. Let *m* be the index denoting the position in the ordering, and then:

$$PPAC_{j[1]} \geq PPAC_{j[2]} \geq \ldots \geq PPAC_{j[m]} \geq \ldots \geq PPAC_{j[|M|]} \qquad where\ |M| < |J|$$

Then, starting from the warehouse that has the highest penalty value, the warehouses are assigned to the cheapest plants. However, unlike the warehouses, the plants do have capacity limits. Thus while assigning a warehouse to a plant which still has production capacity left, one of the following two cases may occur:

1. The remaining capacity of the cheapest plant is greater than the warehouse demand w_j^{actual} .

2. The remaining capacity of the cheapest plant is less than or equal to the warehouse demand w_j^{actual} .

In the first case, the plant still has got enough capacity to serve the full demand of the relevant warehouse. So the whole demand of this warehouse is assigned to the mentioned plant and the remaining capacity of the plant is updated. Then the assigned warehouse and its penalty are removed from the list and the next warehouse that has the highest penalty value in the list is selected to be assigned to a plant.

In the second case, the remaining capacity of the plant is not enough for satisfying the overall demand of the relevant warehouse. Similar to the second echelon, multiple sourcing for a warehouse is allowed. For this reason, the maximum amount that the relevant plant can accept is assigned to the plant, and then the remaining (unmet) demand of the warehouse is calculated as “the overall demand of the warehouse” minus “the remaining capacity of the plant before assigning some of the warehouse’s demand”. At this stage, the associated plant has no more remaining capacity. So it is closed and the penalty values are re-calculated by ignoring the fully loaded plants.

Then again the warehouse demand that has the maximum penalty value is selected and this procedure is repeated until all demands of all warehouses are fully satisfied by the plants.

4.2.3 Local Feasibility Check

As we recall, for solving the plant allocation heuristic efficiently, we ignore the minimum supply constraints of the plants. Therefore some plants may supply less than their minimum limit. This local infeasibility, which is the infeasibility situation that is caused by one or more plants, makes the original problem infeasible, too. In the literature, in order to deal with this infeasibility, some interchange heuristics are used in some studies like Ayrım (2006) or in other studies upperbound is marked as infeasible for this iteration and it is proceeded with the next iteration as it is in the study of Correia et al. (2003). Interchange heuristics may be efficient for the capacitated facility location with minimum supply requirements problem class, but for the modular capacitated facility location problem, we have decided that the most efficient way to deal with this infeasibility is downgrading the capacity level of the infeasible plants: The lower capacity level always has lower cost compared to the higher capacity levels as shown in Figure 3.1. Therefore we have downgraded the capacity level of the plant until the solution becomes feasible. It should be noted that in the modular capacitated facility problem, there may be production intervals that are not covered by any of the capacity levels. For example in Figure 3.1 the interval between v_2^{max} *and* v_3^{min} is not covered by any capacity level. In other words, the plant in Figure 3.1 can not produce ψ amount of product if $v_2^{max} < \psi < v_3^{min}$. If the result of the lowerbound of a plant is in this uncovered interval, then the capacity level is decreased to the nearest lower capacity level, which is Capacity level 2 for our example.

After downgrading the infeasible plants, the problem may become globally infeasible, so we return to the greedy heuristic stage and re-execute the plant greedy heuristic and plant allocation heuristic with the downgraded capacity levels. But to prevent the downgraded facilities to be upgraded to the previous capacity level in greedy heuristic, we make a tabu list and do not calculate the opening cost for the capacity levels which are in the tabu list. Tabu list is valid only for an iteration and in each iteration tabu list is emptied after generating a feasible upperbound. If the local infeasibility still exists after a specified number of trials, we give up looking for a feasible upperbound in this iteration and assign positive infinite value to the upperbound solution of this iteration considering the computational burden of repeating the primal heuristic.

If all plants are feasible, then we can calculate the value of the primal heuristic at the current iteration, P^{UB}, by putting the values of the decision variables which are found in the primal heuristic, into the objective function of the original problem, *P*. Let us assume $\hat{x}_{ijl}$, $\hat{z}_{jk}$, $\hat{q}_{il}$ and $\hat{r}_j$ be the primal heuristic values of the decision variables, then the value of the primal heuristic at the current iteration is as follows:

$$P^{UB} = \sum_{i \in I} \sum_{j \in J} \sum_{l \in L} \left(b_{ij} + e_{il} \right) \hat{x}_{ijl} + \sum_{j \in J} \sum_{k \in K} c_{jk} \hat{z}_{jk} + \sum_{i \in I} \sum_{l \in L} f_{il} \hat{q}_{il} + \sum_{j \in J} g_j \hat{r}_j \ .$$

As mentioned before, primal heuristic is employed in our solution procedure for generating a feasible upperbound for the original problem, *P*. Since the original problem, *P* has minimization type objective function, the minimum of the primal heuristic found among certain iterations is selected as the best primal solution which is called as the upperbound value, UB.

4.3 Subgradient Optimization

In the section 4.1.1.1 we dealt with the issue of finding optimal Lagrange multipliers and chose to employ subgradient optimization in our heuristic to update the Lagrange multipliers. Beasley (1995) defines subgradient optimization as an iterative procedure which, from an initial set of Lagrange multipliers, generates further multipliers in a systematic fashion that attempts to maximize lowerbound value obtained from the Lagrangean dual problem. Subgradient optimization is developed and improved by Polyak (1967), and after the work of Held and Karp (1970, 1971) and Held, Wolfe and Crowder (1974), subgradient algorithm has been widely used in many different contexts for producing lowerbounds for large-scale linear programs.

Subgradients are partial differentials of the Lagrangean dual problem in a specific point and used instead of gradient of a function if the function is nondifferentiable. Unlike the gradient direction, the subgradient direction may not be an ascent/descent direction, and for this reason a line search can not be done in this method. Let us assume that $f(.)$ is a nondifferentiable function and $u^t \in \Re^n$, $\partial f(\overline{u})$ denotes the subdifferential, or the set of subgradients of $f(.)$ at u^t. Then

$$\partial f(\overline{u}) = \left\{ g \in \Re^n \mid f(u) \le f(\overline{u}) + g(u - \overline{u}) \; \forall u \in \Re^n \right\}.$$

For example, if we recall our illustration that has an objective function value $Z = Cx^1 + \lambda(b - Ax^1)$ at point x^1, the subgradient vector g^1 is calculated as below:

$$\partial Z(x^1, \lambda) = g^1 = b - Zx^1$$

Although subgradients are not an ascent/descent direction, Fumero (2001) states that the subgradient optimization guarantees that the new point is closer, in the Euclidean sense, to the optimal point u^t, since by definition, the subgradient vectors g^t and $(u^* - u^k)$ form an acute angle. Therefore, by selecting a sufficiently small step size, direction g^k allows to move towards the optimum, so that the Euclidean distance to the optimal solution is strictly decreasing. Thus, the choice of the step size is crucial for the convergence of the algorithm. The step size should decrease as the iterations increase, but not so quickly as to cause Lagrangean dual problem to converge to a non-optimal point. There are several popular choices for calculating the step size, δ^t for iteration t as cited below:

1. $\delta^t \to 0, \qquad \sum_{t=1}^{\infty} \delta^t = \infty.$
2. $\delta^t = C\rho^t, \qquad 0 < \rho < 1, \qquad C > 0.$

3. $\delta^t = \dfrac{\mu^t\left(f\left(u^*\right)-f\left(u^t\right)\right)}{\left\|g^t\right\|^2}, \qquad 0 < \mu^t \leq 2.$

where $f\left(u^*\right)$ denotes the optimal value of the function. First two alternatives are used in the literature, while the third formulation is the most common one. However, the optimal value of the problem is required to calculate the step size which makes the formulation impractical. In order to deal with this problem, either a fixed constant that is assumed to be slightly bigger than the optimal value but not as big, or a variable value that is calculated for each iteration is used instead of $f\left(u^*\right)$ in the formulation, if no primal heuristic is employed in the algorithm. On the other hand, if a primal heuristic is employed in the algorithm to find feasible solutions for the problem, then the best way to follow is using the best upperbound found so far instead of the optimal value.

The parameter μ^t used in formulation 3 above is a constant that lies between 0 and 2 in the heuristic. Often the sequence of μ^t is determined by setting an initial value that is less than or equal to 2 and halving μ^t whenever the lowerbound value fails to increase in some predetermined number of iterations.

After calculating the subgradients and the step size, the Lagrange multipliers, λ^t, are updated using the formulation below:

$$\lambda^t = \lambda^{t-1} + \delta^t d^t \ ,$$

where d^t denotes the selected direction in the t^{th} iteration. Crainic et al. (2001) denote that as for the early versions of the subgradient algorithm, only the subgradients are used to compute a direction d^t. However, it was quickly realized that taking into account the directions of the previous iterations could lead to performance improvements. Hence, for calculating the direction d^t at the t^{th}

iteration, the linear combination of all previous gradient directions is used rather than just the current subgradient. The general formula of the direction is as follows:

$$d^t = g^t + \theta^t d^{t-1} \ .$$

This formulation is the special case of the previous formulation where the constant θ^t is not equal to zero this time. There are many approaches for calculating the coefficient θ^t to obtain better directions. For example, Camerini–Fratta–Maffioli rule is developed in the study of Camerini et al. (1975) for determining θ^t values. According to this rule, the values of θ^t is calculated as below:

$$\theta^t = \begin{cases} {-\eta g^t d^{t-1}} \big/ {\left\| d^{t-1} \right\|^2} & \text{if } g^t d^{t-1} < 0 \\ 0 & \textit{Otherwise}, \end{cases}$$

where η is a constant whose optimal value could be determined through experiments. The rule of thumb indicates that using 1.5 for η is the good choice. On the other hand, the simplest and the most widely used alternative that is developed by Crowder (1976) is called the Crowder rule. According to the Crowder rule, the constant θ^t is fixed to a value that is less than 1. The rule of thumb for this alternative is fixing θ^t to 0.3. When the Crowder rule is applied, the direction becomes a power series as below:

$$d^t = \sum_{l=1}^{t-1} \left(\theta^{t-l-1} g_l \right) = g_{t-1} + \theta g_{t-2} + \theta^2 g_{t-3} + \ldots + \theta^t g_1 \, .$$

4.3.1 Subgradient Optimization for TSMCFLP

In our problem, we have decided to employ the direction that incorporates the previous directions as well and determine the value of θ^t constant using the Crowder rule with $\theta = 0.3$. We have two different Lagrange multiplier sets, thus different subgradients and directions are calculated for each Lagrange multiplier set, whereas only one step size is generated for all of the Lagrange multipliers. At iteration *t*, after finding the lower and upperbound values of the current iteration, first the subgradients ϕ_j^t *and* φ_k^t of the Lagrange multipliers α_j^t *and* β_k^t are calculated:

$$\phi_j^t = \sum_{k \in K} d_k z_{jk}^t - \sum_{i \in I} \sum_{l \in L} x_{ijl}^t$$
$$\varphi_k^t = 1 - \sum_{j \in J} z_{jk}^t$$

Secondly directions ϖ_j^t *and* ω_k^t are calculated as below:

$$\varpi_j^t = \phi_j^t + \theta \varpi_j^{t-1}$$
$$\omega_k^t = \varphi_k^t + \theta \omega_k^{t-1} .$$

Then the step size for the current iteration is calculated as follows:

$$\delta^t = \frac{\mu^t \left(1.05 * UB - P^{LR}\left(\alpha^t, \beta^t\right)\right)}{\sum_{j \in J} \phi_j^t \varpi_j^t + \sum_{k \in K} \varphi_k^t \omega_k^t} .$$

Finally, the Lagrange multipliers α_j^t *and* β_k^t are updated as follows:

$$\alpha_j^t = \alpha_j^{t-1} + \delta^t \varpi_j^t$$
$$\beta_k^t = \beta_k^{t-1} + \delta^t \omega_k^t .$$

4.4 Stopping Criteria

We have already mentioned that subgradient optimization is an iterative procedure. If some criteria are not defined for terminating the procedure, it would keep iterate forever. As Bahiense et al. (2002) state, the subgradient optimization has lack of well-defined stopping criteria. The theoretical stopping criterion of the subgradient optimization is that the norm of the projected subgradient is too small. But as Crainic et al. (2001) express unfortunately this stopping criterion almost never applies in practice, because it would require that the solution of the Lagrangean dual problem be feasible for the original problem. Therefore, alternative stopping criteria have to be determined. We employ two criteria for this purpose similar to the previous researches.

First criterion calculates a ratio of the gap, $Gap\%$, between the upperbound and the lowerbound value and stops the overall procedure if its value is small enough. Its value is calculated as follows:

$$Gap\% = \frac{(UB - LB)}{LB} \ .$$

We decide the threshold value as 0.001. It means that if the $Gap\%$ is less than or equal to 0.001, the developed Lagrangean heuristic is assumed to converge to the optimal point and the procedure stops.

The second criterion limits the number of iterations allowed to a certain number. If the heuristic does not converge to a point in the determined number of iterations, then this criterion steps in and stops the procedure. If this limit is set too small, the procedure is terminated without having an opportunity to converge. In this case, the computational time is too low, but the $Gap\%$ is too high. On the other hand, if the limit is set too high, the procedure converges to a near-optimal point during the procedure and keeps iterating without improving the solution of the procedure. Unfortunately, determining the value of the maximum number of iterations is empirical and problem specific. Previous researchers use various numbers between 100 and 50,000 according to the complexity and the size of their problem, and the heuristic. In the light of the previous researches, we have decided to set its value to 1000 for all problem instances.

After defining the stopping criteria, all components of the heuristic are conflated together. At each iteration first, the value of the relaxed problem is found and the best lowerbound is updated if available. Second, warehouse greedy heuristic and warehouse allocation heuristic are executed for finding a feasible solution for the second echelon. Third, plant greedy heuristic and plant allocation heuristic are executed alternately. Then local feasibility is checked for each opened plant. If there exists local infeasibility in one or more plants, the capacity levels of infeasible plants are adjusted until the infeasibility is eliminated and the plant greedy and allocations are re-executed for finding a feasible solution for the second echelon. If all plants are feasible, then stopping criteria are checked. As a final step subgradient optimization is employed and the Lagrangean multipliers are updated by this procedure. Figure 4.1 shows the overall solution procedure that is explained above verbally. The pseudo-code of the overall Lagrangean Heuristic is as follows:

START
INIT $LB = -\infty$, $UB = +\infty$, $\alpha_j = 0$, $\beta_k = 0$, no_improv = 0 and counter = 0
Define the values of iteration limit and μ
WHILE Counter < Iteration limit AND Gap > 0.001
  SET counter = counter + 1
  CALL Solver of first Main Subproblem
  CALL Solver of second Main Subproblem
  Calculate the solution value of the Lagrangean relaxation at this iteration ($P^{LR}\left(\alpha^t, \beta^t\right)$)
  IF $P^{LR}\left(\alpha^t, \beta^t\right)$ > LB **THEN**
    SET LB = $P^{LR}\left(\alpha^t, \beta^t\right)$
    SET no_improv = 0
  ELSE
    SET no_improv = no_improv +1
    IF no_improv equal to no_improv limit AND μ is greater than $\mu_{\min}$ **THEN**
      Update the value of μ
      SET no_improv = 0
    END IF
  END IF
  CALL Warehouse Greedy Heuristic
  CALL Warehouse Allocation Heuristic
  REPEAT
    CALL Plant Greedy Heuristic
    CALL Plant Allocation Heuristic
    FOR Each opened plant
      Check Local feasibility
    END FOR

UNTIL Local feasibility of each opened plant is OK

Calculate the value of the primal heuristic at this iteration (P^{UB})

IF P^{UB} > UB **THEN**

SET UB = P^{UB}

END IF

SET $Gap = {(UB - LB)}\big/{LB}$

IF Gap > 0.001 **THEN**

Calculate subgradients, directions and stepsize

Update Lagrange multipliers α_j *and* β_j

END IF

END WHILE

Write the performance measures

STOP

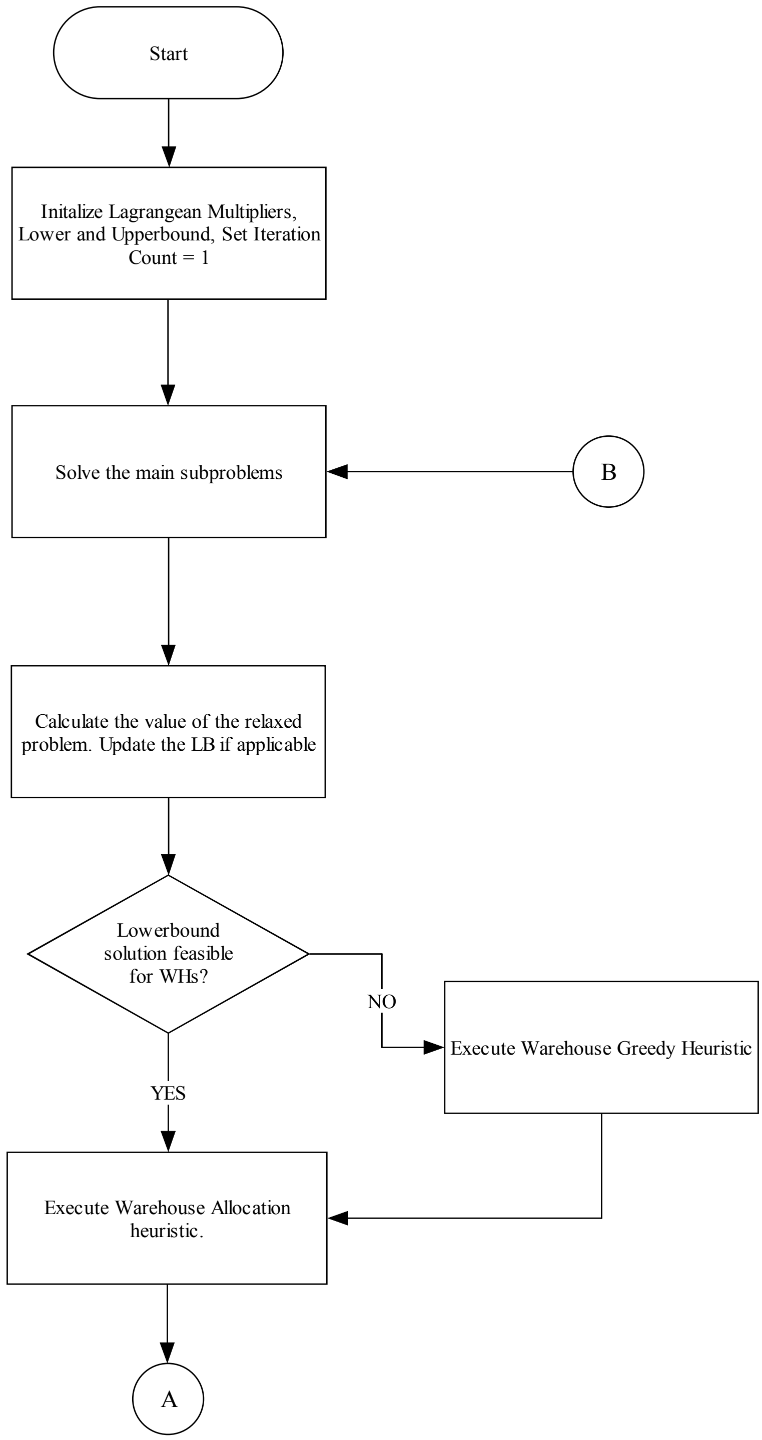

Figure 4.1: The Proposed Lagrangean Heuristic Approach

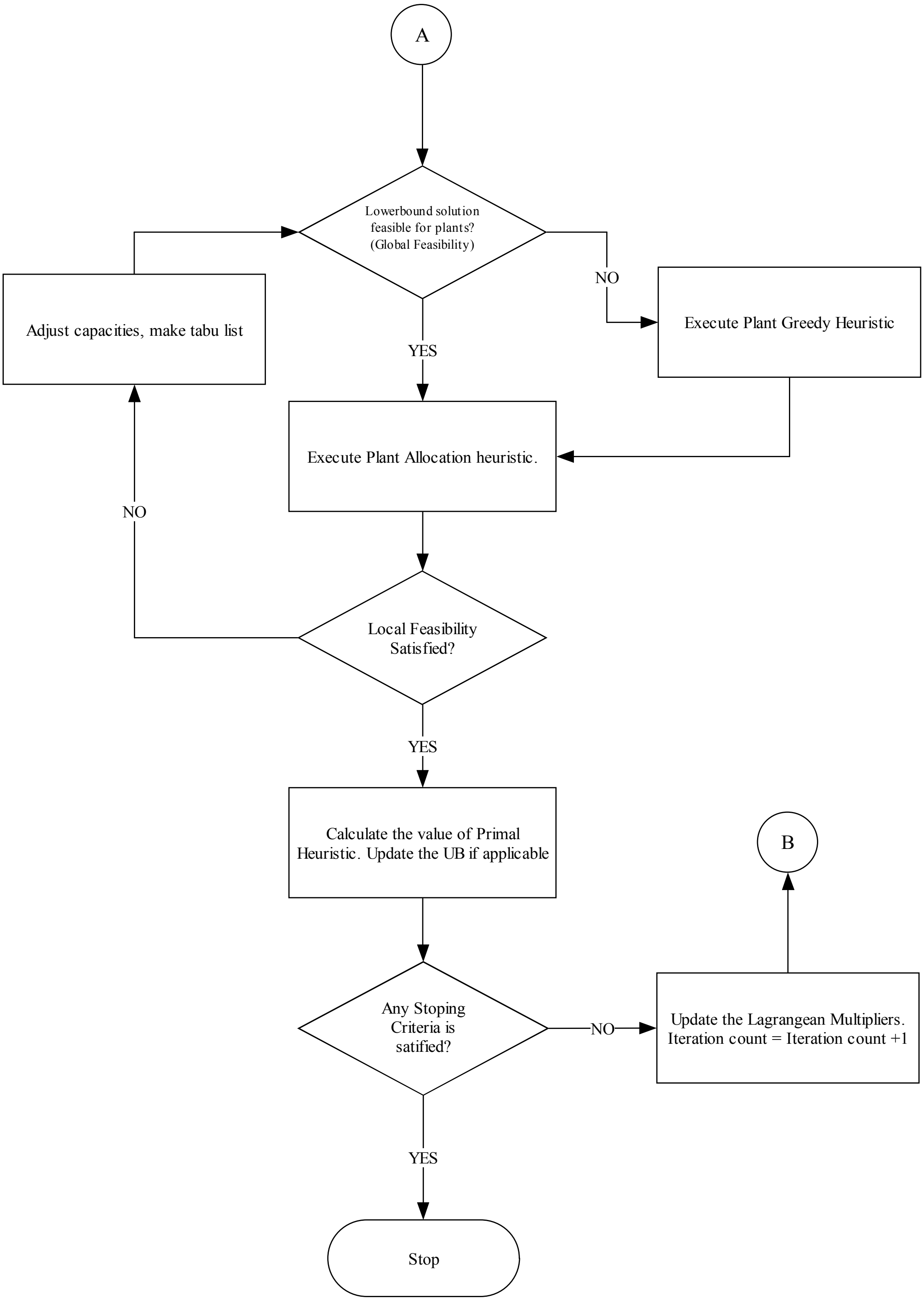

Figure 4.1 Cont’d

CHAPTER 5

COMPUTATIONAL STUDY

In this chapter, we discuss the results of the experiments designed to evaluate the performance of our solution approach. We first introduce the design of our experiments, i.e. the generation of the problem instances. Next we define the performance measures. In the last section, we report and discuss the results of the computational study.

5.1 Design of the Experiments

In order to test the performance of our solution procedure, a variety of problems are generated and solved for different sets of *I, J, K* and *L*. Both in the two-stage facility location and modular capacitated problems literature, all researchers generate their own data randomly for testing their solution procedure as there is no library that includes the data sets of the previous researches. All researchers use different parameters for generating data and problem instances. We have decided to implement a similar procedure to the procedure that was used in the study of Harkness and ReVelle (2002) in constructing the data for the test problems, because among the similar studies, this study has the most comprehensive data generation scheme. We have extended their data generation procedure for our TSMCFLP as follows:

The annual demands of the customers are drawn from a uniform distribution between 10 and 100.

The unit transportation costs of distributing the goods from the plants to the warehouses and from the warehouses to the customers are generated from a uniform distribution between 1 and 10.

The expected number of the maximum-capacitated facilities is computed by multiplying the two problem parameters for the number of candidate facility sites by the proportion of the largest facilities required to serve the total demand. Dividing the total demand by this expected number of the maximum-capacitated facilities yields a maximum capacity average. In order to obtain the maximum capacity for each site, this average is multiplied by a uniformly distributed random number in [0.75, 1.25].

The maximum capacities for the facilities less than the highest capacity level are assigned according to the following formula:

$$v_{il}^{\max} = v_{il-1}^{\max} + z\left(v_{i|L|}^{\max} - v_{il-1}^{\max}\right) / \left(|L| - l\right)$$

where z is a uniformly distributed random number in [0.25, 0.75]. This formula insures a disproportionate increase in the size of the facilities with the increase of *l*. The minimum production capacity of the first capacity levels is set to 1. The other capacity levels are determined as follows:

$$v_{il'}^{\min} = \begin{cases} v_{il'-1}^{\max} + 1 & \textit{with probability}\ 0.8 \\ \left(v_{il'-1}^{\max} + 1\right) z & \textit{with probability}\ 0.2 \end{cases} \qquad \forall i \in I \textit{ and } l' \neq 1$$

where z is a uniformly distributed random number in [1, 1.25]. The capacities of the warehouses are calculated as follows:

$$w_j = z\zeta / |J| \ ,$$

where z is a uniformly distributed random number in [1, 1.5] and ζ is the total demand factor which is calculated by multiplying the total demand by a constant. We take the parameter ζ constant as “3” in our study which means that on the average opening 1/3 of the warehouses is enough for satisfying the overall demand.

The annual fixed cost of establishing the highest capacity level of a plant is determined according to the following formula:

$$f_{i|L|} = z\xi v_{i|L|}^{\max} \ ,$$

where z is a uniformly distributed random number in [0.75, 1.25] and ξ is an average total cost factor. This factor is selected in order to determine the importance of the fixed costs compared to the transportation costs. We take its value as 50, reflecting the plant locations having “higher importance” than the transportation costs. The annual fixed costs of the other capacity levels are assigned via the following formula:

$$f_{il} = f_{il-1} + z\left(f_{i|L|} - f_{il-1}\right) / \left(|L| - l\right) \ ,$$

where z is a uniformly distributed random number in [0.25, 0.5]. The variable cost of production is assigned by the following formula:

$$e_{il} = z\frac{2}{l}\frac{\left(f_{il} - f_{il-1}\right)}{\left(v_{il}^{\max} - v_{il-1}^{\max}\right)}$$

where z is a uniformly distributed random number between 0.25 and 0.5 and $f_{i0} = v_{i0}^{\max} = 0$. The scope of the cost function and the 2/*l* factor insures that on the average, the variable cost is declining for higher capacity levels, reflecting the economies of scale.

The annual fixed cost of opening a warehouse is determined as follows:

$$g_j = z\hat{f}_{i1} \ ,$$

where z is a uniformly distributed random number in [0.375, 0.75] and $\hat{f}_{i1}$ is the average cost of installing the first level capacity for plants.

Our test problem instances are generated according to the defined procedure. In some cases, generated values may not be suitable for our problem because of the existence of random variables in the procedure. Therefore, at each stage the procedure checks the values of the parameters for conformity with the assumptions stated while defining the problem environment and adjust the values of the parameters if necessary. For example, all fractional values of the parameters are rounded to the nearest integer. Besides these randomly generated parameters, there are some parameters of the model that have to be determined by us. These parameters are the number of the maximum allowed warehouse, $R_{\max}$; the initial values of Lagrange multipliers, α^0 *and* β^0; the stepsize multiplier that is used for calculating the stepsize in the subgradient optimization procedure, μ; the allowed number of unsuccessful iterations before updating the value of μ. Unfortunately, all these parameters are problem specific and an efficient procedure for estimating their best value for the problem instance does not exist. Hence, we have decided to determine their values by examining the previous researches.

The maximum number of the warehouses allowed can not be a fixed value for each problem size and it has to be determined according to the number of possible warehouse locations in the model. It must be higher than the minimum number of warehouses that have to be opened in order to satisfy the overall demand, $R_{\min}$. We have decided its value as $R_{\max} = \frac{1}{2}|J|$. It means that at most half of the possible warehouse locations can be opened at any instance.

The most important issues of the subgradient optimization are determining the initial value of the stepsize multiplier, μ, and the procedure of updating the value of μ. Crainic et al. (2001) express that the computational experiments show that the setting of the initial value of μ usually makes up a significant portion of the difference between obtaining good performances and having the method diverge, never being able to improve on the initial estimate. Unfortunately, in the literature we do not find a procedure which seems to provide at least reasonable performances on all problem classes. However Held et al. (1974) states that taking μ between 0 and 2 assures the geometric convergence to the optimal point. This is the reason why μ is given values ranging from 0 to 2 in the previous researches. In our study, we have decided to take the initial value of μ as 2, considering the results of preliminary testing.

In the literature, there are a few criteria and procedures for updating the value of μ during the heuristic. All of them multiply the value of μ with a constant number between 0 and 1 after a number of consequent unsuccessful iterations in the lowerbound procedure up to a certain point. Unsuccessful iteration is an iteration that does not change the value of the UB. After reaching a certain point, few researchers terminate the overall heuristic procedure, few researchers reset the value of μ to its initial value, and the rest of them just stops multiplying μ with a constant number. If the number of consequent unsuccessful iterations is set too small or no certain point is set for the procedure, then the Lagrangean relaxation procedure may converge to a far-optimal point because of the very small values of μ.

In our subgradient procedure, we have decided to multiply μ with 0.5 after 50 consequent unsuccessful iterations until the value of μ has become less than or equal to 0.1 ($\mu_{\min} = 0.1$). After this point, we keep iterating the heuristic without changing the value of μ.

Initializing the values of Lagrange multipliers is a less important subject in subgradient optimization; that is why many of the researchers such as Tragantalerngsak et al. (1997, 2000), Pirkul and Jayaraman (1998, 2001), Mazzola and Neebe (1999) and Broek et al. (2006) have not mentioned their own procedures of generating the initial Lagrange multipliers generation.

A small part of the researchers such as Gavish (1978), Beasley (1993), Marin and Pelegrin (1999), Marin (2007), Bektas and Bulgak (2008) have tried to estimate the values of Lagrange multipliers in different ways and started the subgradient procedure with these estimated initial Lagrange multipliers. For example, Gavish (1978) solved the LP relaxation of the problem, and set the initial Lagrange multipliers as the dual variables of the LP relaxation.

The rest of the researchers including Fisher (1981), Jang et al. (2002), Correia and Captivo (2003, 2006) and Amiri (2006) preferred to set the initial values of Lagrange multipliers to zero, because it is assumed that in the first 50-100 iterations, the values of Lagrange multipliers of all the alternatives converge to very close values. For this reason, starting with good Lagrange multipliers can be seen as warm start of the procedure, but after some iterations usually no difference remains between a warm start and a cold start. In our study, we have decided to make a cold start and set the values of α^0 *and* β^0 to zero.

After defining the parameters used in the heuristic procedure, we may test our solution procedure in several problem sizes to show its efficiency and robustness. The parameters defining the problem size are the number of potential plant locations, $|I|$, the number of potential warehouse locations, $|J|$, the number of customers, $|K|$, and the number of available capacity levels for a plant location , $|L|$. In our experiments, we prefer to fix the number of available capacity levels to a constant value that both preserves the nature of the modular capacitated structure and takes less time in computing the optimal solution of the original problem, *P*. Hence, we have decided to make our elaborated runs with three levels of capacity. But in order to show the performance of the heuristic, we also make runs with 5 and 10 capacity levels for three problem sizes only.

Table 5.1. Problem Sizes in the Experiments

# of Plant Locations $\lvert I \rvert$	# of Warehouse Locations $\lvert J \rvert$	# of Customers $\lvert K \rvert$	# of Capacity Levels $\lvert L \rvert$
5	10	20	3
5	10	20	5
5	10	20	10
5	10	30	3
5	10	40	3
5	10	50	3
5	15	40	3
5	15	50	3
10	25	50	3
10	25	75	3
10	25	100	3
10	25	200	3
10	40	100	3
10	40	200	3
20	30	100	3
20	30	100	5
20	30	100	10
20	30	200	3
20	40	100	3
20	40	200	3
20	50	100	3
20	50	200	3
20	50	500	3
30	50	100	3
30	50	200	3
30	50	500	3
30	50	500	5
30	50	500	10

Other three parameters $\lvert I \rvert$, $\lvert J \rvert$ and $\lvert K \rvert$ are formed in such a way that a wide range of problem sizes is covered. Starting with the small-size problems of 5 potential locations, 10 warehouse locations and 20 customers, the larger-scale problems up to 30 plant locations, 50 warehouse locations and 500 customers are considered. The problem sizes that are employed during the experiments can be seen in Table 5.1. For each problem size, we have solved randomly but systemically generated 10 problem instances; as a total we have got 22 problem sizes with 3 capacity levels and 3 problem sizes with 5 and 10 capacity levels. By solving 10 problem instances for each set, we have carried out the experiments with 280 problem instances.

5.2 Performance Measures

In evaluating the performance of the developed solution procedure, we use the following main performance measures:

1. The following gap ratios are the most important measures for investigating the performance of the heuristic:

   a. The total gap ratio, $Gap\%$: It is the gap ratio between the upper and lower bounds with respect to the lowerbound. $Gap\%$ reveals the quality of the overall heuristic procedure. It is calculated as follows:

   $$Gap\% = (UB - LB)/LB$$

   b. The upperbound gap ratio, $Gap_{UB}\%$: It is the gap ratio between the upperbound and the optimal solution of the problem with respect to the optimal solution. As mentioned before, in the TSMCFLP, the lowerbound heuristic may be generating results far from feasibility. Therefore, much falls onto the greedy heuristic developed by us. By this ratio, we may examine the quality of the primal heuristic procedure. Besides, if we decide to implement this solution in real life, then our actual distance from the optimal solution, in other words the loss, will be determined by this measure. Therefore it is an important performance measure for us. If P^* represents the optimal solution of the original problem, P, then the upperbound gap ratio is calculated as follows:

$$Gap_{UB}\% = \left(UB - P^*\right) \Big/ P^*$$

c. The lowerbound gap ratio, $Gap_{LB}\%$: It is the gap ratio between the lowerbound and the optimal solution of the problem with respect to the lowerbound of the heuristic. This ratio shows the performances of Lagrangean relaxation and subgradient optimization techniques in our problem. It is calculated as:

$$Gap_{LB}\% = \left(P^* - LB\right) \Big/ LB$$

2. The solution time in central processing unit (CPU) seconds: We have solved each problem instance both with the developed heuristic and GAMS with CPLEX 10.0.1 and compared the solution times of both. This comparison shows the performance of the developed heuristic against the best known optimization package, CPLEX.

In addition to these main performance measures, we have collected some supportive performance measures, for investigating the performance of the developed upperbound heuristic procedure. These measures count how many times the warehouse greedy heuristic procedure and the plant greedy heuristic procedures are needed for. Some other measures count how many times each alternative solution of the greedy heuristic gives the best solution. Also there is another measure that counts how many times the local feasibility fails in one or more plants. In order to interpret their results, we show them in percentages in the average results table.

5.3 Experimental Results

The heuristic solution procedure is coded in GAMS environment. The optimal solutions of the problem instances are solved in GAMS with CPLEX 10.0.1. All computational studies are conducted on an Intel® Pentium® M 1.7 GHz processor with 512 MB RAM under Windows XP operating system.

The detailed results of the problem instances are given in Appendix 1 while a summary of those results is given in Table 5.2 and Table 5.3 below, where the average values of the main performance measures and the average values of the supportive performance measures for each problem size are presented, respectively.

In Table 5.2 and 5.3, each row shows the average values of the main and supportive performance measures for each problem size. The results are rounded to the nearest representable value.

CPLEX 10.0.1 is unable to solve 26 problem instances due to the memory limitations. In Table 5.2, the problem sizes that include the problem instances which cannot be solved by CPLEX are marked with an asterisk sign (*). In problem size 30x50x500x10, CLPEX can not solve a problem instance in 10 hours. This problem size marked with a dagger sign (†) in Table 5.2 and in Appendix 1 includes the problem instance mentioned. In problem size 30x50x500x5 CPLEX is unable to solve any of the ten instances thus the average $Gap_{LB}\%$ and $Gap_{UB}\%$ values of this problem size can not be calculated.

In Table 5.3, the "Avg. WHGH" column represents the average of how many times the warehouse greedy heuristic procedure is needed for in our heuristic procedure as a percentage of the number of the total iterations of the heuristic procedure. For example in the first row, for the problem size 5x10x20x3, the value of "Avg. WHGH" is 0.25%. It means that the warehouse greedy heuristic is executed for a problem in 0.25% of the iterations on the average. We have already mentioned that we have solved 10 problem instances for a problem size, and we have made 1000 iterations in one problem instance, if the first stopping criterion has not been satisfied earlier. Therefore, we need to call the warehouse greedy heuristic procedure 1000*0.25% = 2.5 times on the average for a problem instance and 10*1000*0.25% = 25 times in total for the relevant problem size.

The "Avg. POGH" and "Avg. PCGH" columns also show similar performance measures. They present the percentage of how many times the plant greedy heuristic procedure is needed for on the average in our heuristic procedure for opening more plants and for closing some of the opened ones, respectively. Their values are calculated as is the value of “Avg. WHGH”.

Table 5.2. Results of the Main Performance Measures in Averages

Plant $\lvert I \rvert$	Warehouse $\lvert J \rvert$	Customer $\lvert K \rvert$	Capacity Level $\lvert L \rvert$	Average $Gap_{LB}\%$	Average $Gap_{UB}\%$	Average $Gap\%$	Average Heuristic CPU (s)	Average CPLEX CPU (s)
5	10	20	3	1.558	0.340	1.904	3.552	1.985
5	10	20	5	2.197	0.431	2.635	4.479	1.619
5	10	20	10	1.975	0.602	2.590	6.524	3.346
5	10	30	3	3.278	0.226	3.514	4.967	2.482
5	10	40	3	1.905	0.359	2.271	6.822	2.853
5	10	50	3	1.664	0.518	2.190	8.920	5.870
5	15	40	3	1.919	0.388	2.314	8.843	9.246
5	15	50	3	2.557	0.219	2.781	11.157	11.435
10	25	50	3	1.575	0.485	2.068	18.021	29.839
10	25	75	3	2.346	0.593	2.819	27.685	422.153
10	25	100	3	1.736	0.667	2.413	39.035	166.176
10	25	200	3	2.036	0.478	2.523	107.965	187.337
10	40	100	3	1.890	0.564	2.465	57.065	309.656
10	40	200	3	1.934	0.540	2.483	141.009	376.617
20	30	100	3	1.033	0.907	1.949	49.249	916.166
20	30	100	5	0.905	0.916	1.816	56.432	1948.958 *
20	30	100	10	0.678	0.806	1.490	75.357	3935.654
20	30	200	3	1.188	0.693	2.053	121.138	1820.632 *
20	40	100	3	1.022	1.013	2.045	62.855	1510.097
20	40	200	3	1.083	0.970	2.063	146.647	3690.557
20	50	100	3	1.163	0.829	1.951	79.778	2011.426 *
20	50	200	3	1.035	1.016	2.189	177.632	1803.916 *
20	50	500	3	1.685	0.553	2.247	759.923	4311.758
30	50	100	3	0.615	0.876	1.457	88.456	2598.602 *
30	50	200	3	0.720	0.838	1.554	185.776	5890.367 *
30	50	500	3	1.298	0.934	2.253	834.614	6544.300 *
30	50	500	5	N/A	N/A	1.9036	840.605	21920.083 *
30	50	500	10	0.674	0.976	1.6047	937.701	22628.724 *†
Average				**1.543**	**0.657**	**2.198**	**172.037**	**2923.544**

Table 5.3. Results of the Supportive Performance Measures in Averages

Plant $\lvert I \rvert$	**WH** $\lvert J \rvert$	**Customer** $\lvert K \rvert$	**Cap. Level** $\lvert L \rvert$	**Avg. WHGH (%)**	**Avg. WHGH 1 (%)**	**Avg. WHGH 2 (%)**	**Avg. WHGH 3 (%)**	**Avg. POGH (%)**	**Avg. POGH 1 (%)**	**Avg. POGH 2 (%)**	**Avg. POGH 3 (%)**	**Avg. POGH 4 (%)**	**Avg. PCGH (%)**	**Avg. PCGH 1 (%)**	**Avg. PCGH 2 (%)**	**Avg. PCGH 3 (%)**	**Avg. PCGH 4 (%)**	**Avg. LFC (%)**
5	10	20	3	0.25	100	0	0	26.80	15.261	5.858	47.649	31.231	1.76	18.182	0	52.841	28.977	4.1
5	10	20	5	3.74	100	0	0	51.47	34.719	0.039	28.521	36.720	8.05	60.621	0	30.683	8.696	12.3
5	10	20	10	3.23	100	0	0	47.73	0.859	0.482	48.607	50.010	9.25	33.946	0	64.216	1.838	20.6
5	10	30	3	4.19	100	0	0	42.90	9.930	0.000	60.816	29.254	3.25	12.308	0	54.462	33.231	12.8
5	10	40	3	4.50	100	0	0	26.37	21.540	0.076	35.647	66.932	3.49	14.900	0	48.138	36.963	2.3
5	10	50	3	8.77	100	0	0	32.21	1.149	14.002	61.999	22.850	3.56	24.719	0	25.562	49.719	4.1
5	15	40	3	8.32	100	0	0	40.21	4.054	7.038	84.208	4.700	1.73	15.607	0	44.509	39.884	3.8
5	15	50	3	44.26	100	0	0	36.15	28.769	3.679	48.271	19.281	1.48	11.486	3.378	14.865	70.270	2.5
10	25	50	3	54.53	99.890	0.110	0	52.25	40.785	31.828	26.278	1.110	1.30	11.538	0	46.923	41.538	4.5
10	25	75	3	79.78	99.887	0.113	0	59.09	14.791	35.505	48.976	0.728	1.66	10.241	0	48.795	40.964	6.6
10	25	100	3	68.15	99.985	0.015	0	54.57	31.977	17.537	50.229	0.257	1.78	7.865	0	28.652	63.483	4.9
10	25	200	3	37.51	99.920	0.080	0	41.61	18.890	15.741	42.322	23.047	2.04	3.431	0	23.529	73.039	14.0
10	40	100	3	96.82	80.314	17.713	1.973	59.67	22.138	11.279	54.651	11.932	1.46	12.329	0	41.096	46.575	9.8
10	40	200	3	93.75	91.531	8.032	0.437	61.72	44.151	11.633	18.876	25.340	1.78	7.865	0	22.472	69.663	16.1
20	30	100	3	70.48	90.011	7.250	2.738	66.90	24.036	26.308	18.087	31.570	1.74	8.621	0	23.563	67.816	12.8

Table 5.3. Cont'd

Plant $\|I\|$	WH $\|J\|$	Customer $\|K\|$	Cap. Level $\|L\|$	Avg. WHGH (%)	Avg. WHGH 1 (%)	Avg. WHGH 2 (%)	Avg. WHGH 3 (%)	Avg. POGH (%)	Avg. POGH 1 (%)	Avg. POGH 2 (%)	Avg. POGH 3 (%)	Avg. POGH 4 (%)	Avg. PCGH (%)	Avg. PCGH 1 (%)	Avg. PCGH 2 (%)	Avg. PCGH 3 (%)	Avg. PCGH 4 (%)	Avg. LFC (%)
20	30	100	5	68.91	92.918	3.584	3.497	49.20	9.715	15.996	30.976	43.313	2.78	6.475	0	68.705	24.820	20.6
20	30	100	10	66.48	88.779	6.829	4.392	49.25	3.898	1.482	40.792	53.827	5.08	12.402	0	74.016	13.583	37.52
20	30	200	3	65.13	91.571	7.017	1.413	49.37	19.567	31.011	35.062	14.361	2.44	13.934	0	31.967	54.098	18.1
20	40	100	3	92.34	98.700	1.083	0.217	64.32	18.750	34.297	36.210	10.743	1.15	13.913	0	26.087	60.000	7.8
20	40	200	3	78.37	92.880	6.380	0.740	55.58	7.431	6.297	56.639	29.993	1.75	10.857	0	35.429	53.714	11.2
20	50	100	3	85.72	64.116	17.522	18.362	57.20	26.101	20.402	39.755	13.741	1.52	16.447	0	40.132	43.421	10.1
20	50	200	3	86.31	77.164	10.462	12.374	59.32	0.388	39.211	46.561	12.121	1.49	15.436	0	24.832	59.732	16.9
20	50	500	3	61.45	75.232	16.469	8.299	45.46	25.033	28.663	26.507	19.798	2.36	13.983	0	39.831	46.186	30.8
30	50	100	3	75.78	61.032	20.335	18.633	58.79	18.575	21.024	55.128	5.273	1.43	12.587	0	27.972	59.441	12.1
30	50	200	3	85.07	54.120	29.188	16.692	61.74	7.467	21.753	58.017	12.763	1.89	8.995	0	35.450	55.556	16.5
30	50	500	3	46.04	72.133	15.595	12.272	28.81	12.496	21.798	53.350	12.357	2.01	5.970	0	38.806	55.224	43.5
30	50	500	5	24.29	73.775	13.709	12.515	14.39	2.502	20.917	59.833	16.748	6.02	13.787	0	40.199	46.013	93.42
30	50	500	10	13.59	64.312	13.539	22.149	12.64	23.576	34.335	24.763	17.326	59.11	15.124	0	33.412	49.941	145.11
Average				**53.457**	**89.622**	**6.453**	**3.925**	**49.18**	**17.787**	**16.267**	**44.39**	**22.433**	**2.624**	**14.795**	**0.13**	**38.982**	**46.094**	**13.698**

In Section 4.2.1, we discuss the warehouse and plant greedy heuristics thoroughly. As readers may recall, we introduce three different solution alternatives for selecting the opened warehouses. First we find the warehouse that is decided to "remain closed" in the lowerbound solution and has the lowest opening cost. If its capacity is big enough for satisfying the demand shortage, then without evaluating the other alternatives, we select this solution as the best alternative; otherwise we select the warehouses to be opened in two different ways: Selecting the minimum-cost warehouse that is big enough to satisfy the shortage or collecting a set of warehouses that has the lowest opening cost until the shortage is satisfied.

The columns "Avg. WHGH 1", "Avg. WHGH 2" and "Avg. WHGH 3" in Table 5.3 represent how many times these alternatives are selected as the best alternative on the average as a percentage if the warehouse greedy heuristic is required to be executed. For example, for the problem size 20x30x100x3, the values of "Avg. WHGH 1", "Avg. WHGH 2" and "Avg. WHGH 3" are 90.011%, 7.250% and 2.738% respectively. It means that if we execute the warehouse greedy heuristic 100 times, then the first solution alternative is selected as the best alternative 90.011 times, the second solution alternative is selected as the best alternative 7.25 times and so on. The sum of these three values is approximately 100% as expected because there is no other alternative solution if the warehouse greedy heuristic is executed.

Using these three columns and the "Avg. WHGH" column, it is very simple to find how many times these alternatives are selected as the best alternative for a specific problem size. For example, the percentage of selecting the first alternative as the best alternative solution with respect to the total number of iterations of Lagrangean heuristic for the problem size 20x30x100x3 is $70.480\%*90.011\% \cong 63.44\%$. We know that we make 10*1000 = 10,000 iterations for each problem size. Accordingly, for the problem size 20x30x100x3, we select the first solution alternative as the best solution for $63.44\%*10,000 = 6344$ times.

The “Avg. POGH 1”, “Avg. POGH 2”, “Avg. POGH 3”, "Avg. POGH 4” and “Avg. PCGH 1”, “Avg. PCGH 2”, “Avg. PCGH 3”, “Avg. PCGH 4” columns represent similar measures that are presented in “Avg. WHGH 1”, “Avg. WHGH 2” and “Avg. WHGH 3” columns. As explained in Section 4.2.1.2, we have got four different solution alternatives both for opening the required capacity levels and for closing the excess capacity levels. The “Avg. POGH 1”, “Avg. POGH 2”, “Avg. POGH 3” and "Avg. POGH 4” columns show the average of the selection in percentages of the four solution alternatives while opening more plants, also “Avg. POGH 1”, “Avg. POGH 2”, “Avg. POGH 3” and "Avg. POGH 4” columns show the average of the selection in percentages of the four solution alternatives while closing some of the opened plants.

The last column, “Avg. LFC”, shows on the average the percentage of how many times the plant allocation heuristic yields results in which the production volumes of one or more plants are infeasible. For example, for the problem size 5x10x20x3 the value of “Avg. LFC” is 4.1%. It means that for the problem size 5x10x20x3, if we execute Lagrangean heuristic for 100 iterations, then the results of plant allocation heuristic is marked as the “local infeasible” in 4.1 iterations on average and some adjustments are made in the capacity levels of the infeasible plants in order to generate a feasible solution. It can be seen that for some problem sizes, the value of the “Avg. LFC” is greater than 100%. This means that in some iterations, the local infeasibility continues after adjusting the capacity levels and re-allocating the warehouses. Therefore, the local feasibility check stage has to run again and again for some iterations. For example for the problem size 30x50x500x10, the value of “Avg. LFC” is 145.11%. This means that the local feasibility check procedure has to adjust the capacity levels 1.45 times on the average for an iteration.

It can be seen in Table 5.2 that the solution procedure produces high quality results in short times, which makes our proposed heuristic a reasonably well solution alternative for the TSMCFLP. The gap percentage between the best feasible solution (upperbound) and the Lagrangean dual problem (lowerbound) is employed to be able to judge the quality of the solution. The $Gap\%$ values range between 0.438% and

11.760%, with an overall average of 2.198% confirming the high quality of the developed heuristic.

$Gap_{UB}\%$ can be treated as the actual gap as it represents the difference between the optimal solution and the best feasible solution found, which is the only alternative solution that could be implemented in real-life. The $Gap_{UB}\%$ values which are ranging between 0% and 1.880% with an overall average of 0.657% show that the employed allocation heuristic integrated with the developed greedy heuristic and the local feasibility check procedure yields very good upperbound solutions. Even in the problem instance that has the highest $Gap\%$ among the experiments, the value of $Gap_{UB}\%$ is 0.540%. This shows even in the problem instances in which subgradient optimization cannot converge rapidly, the upperbound procedure is effective and efficient.

$Gap_{LB}\%$ can be employed for measuring the quality of Lagrangean relaxation and subgradient optimization. In our study, $Gap_{LB}\%$ is ranging between 0.251% and 11.178% with an overall average of 1.543% which shows that even with the existence of some outliers, the lowerbound procedure is efficient on the average.

As it can be seen in Table 5.3, in the first several problem sizes, the average solution duration of CPLEX is less than the solution duration of the developed Lagrangean heuristic. This is predictable; although the TSMCLP is NP-Hard, the problem structure is very small and CPLEX is the most powerful solver in the market, especially for the small-sized problems. Besides, the maximum difference for a problem size between the average solution duration of CPLEX and the developed heuristic is 3.969 seconds, which is negligible. On the other hand, for most of the problem sizes, the developed heuristic outperforms CPLEX in terms of the average solution duration. On the overall average, the developed heuristic is approximately 17 times faster than CPLEX.

Following is an interesting observation about the solution duration of CPLEX and the developed heuristic: the solution durations of the problem instances within a specific problem size in CPLEX have major differences. On the other hand, the solution durations of the developed heuristic are close to each other within a specific problem size. Table 5.4 shows the standard deviation of the solution durations of the CPLEX and the developed heuristic. The standard deviation of Lagrangean heuristic varies between 0.139 and 12.844 with an average of 2.612 seconds, and the standard deviation of CPLEX solution varies between 0.623 and 10294.882 with an average of 2319.563 seconds. The reason of this high variance of CPLEX may occur due to many different factors that depend on the characteristics of the problem instances. The most possible factors according to us are as follows:

- Due to the unique structure of each problem, at some instances CPLEX is able to generate good initial solutions; however at some instances CPLEX cannot generate good initial solutions.

- CPLEX uses many different heuristics such as relaxation induced neighborhood search, feasibility pump heuristic, node heuristic, and apply many different cuts such as Gomory fractional cuts, clique cuts, mixed integer rounding cuts and so on. For some problem instances, CPLEX is able to generate efficient cuts and heuristics for the branch-and-cut tree, but not for other problem instances.

High variance in CPLEX solution durations makes the estimation of the optimal solution duration impossible, even if many problems having the same problem size have been solved previously. On the other hand, the developed heuristic has got low variance, which makes the estimation of the heuristic solution duration possible.

Robust design is defined as designing a product so that its functionality varies minimally, despite the disturbing factor influences. Our heuristic can solve each problem instance that has the same size but distinctive characteristics in close solution durations, which shows us the robustness of the developed heuristic.

Table 5.4. The Standard Deviations of the Solution Durations

Plant $\|I\|$	Warehouse $\|J\|$	Customer $\|K\|$	Capacity Level $\|L\|$	Standard Deviation (σ) Heuristic	CPLEX
5	10	20	3	0.161	1.187
5	10	20	5	0.222	0.623
5	10	20	10	0.305	1.811
5	10	30	3	0.202	1.192
5	10	40	3	0.139	1.071
5	10	50	3	0.333	1.463
5	15	40	3	0.413	5.415
5	15	50	3	0.377	5.047
10	25	50	3	0.476	41.963
10	25	75	3	0.638	1211.986
10	25	100	3	0.989	390.181
10	25	200	3	2.089	414.696
10	40	100	3	1.225	345.844
10	40	200	3	2.831	460.087
20	30	100	3	0.897	1909.731
20	30	100	5	1.607	2030.548
20	30	100	10	2.629	4468.668
20	30	200	3	2.470	2718.835
20	40	100	3	1.496	2038.426
20	40	200	3	1.542	5852.541
20	50	100	3	1.136	3670.370
20	50	200	3	4.030	3314.790
20	50	500	3	11.024	2487.193
30	50	100	3	2.588	3618.867
30	50	200	3	4.215	8015.969
30	50	500	3	8.922	5599.963
30	50	500	5	12.844	5768.042
30	50	500	10	7.324	10294.882
Average				**2.612**	**2319.563**

In the upperbound procedure, the warehouse greedy heuristic procedure is required to be executed varying between 0.25% and 96.82% with an average of 53.457% iterations of all the iterations. Its average is higher than the plant heuristic requires, which may cause doubts about the strength of the constraint (20). However, if we look at the values of “Avg. WHGH 1”, “Avg. WHGH 2” and “Avg. WHGH 3”, we can see that if the warehouse greedy heuristic is executed, the first solution alternative is selected as the 89.622 % on the average. The first solution alternative of all greedy heuristics can be selected if and only if the results of the lowerbound procedure are very close to the feasible region. In the execution of the plant opening heuristic and plant closing heuristic, the first alternative is selected as the best alternative only 17.787% and 14.795% on the average, respectively. These results show that the result of the lowerbound solution procedure of the first stage is not close to the feasible region due to the lack of a strong valid inequality.

In the first and the second solution alternatives of all the three greedy heuristic procedures, only one facility is opened whereas in the third and fourth solution alternatives, at least two or more facilities are opened for obtaining an initial feasible solution for the allocation heuristic. Table 5.3 shows that opening only one additional warehouse is selected as the best alternative in 96.075% of the warehouse allocation heuristic executions due to the reason explained in the previous paragraph. But in the plant opening and closing greedy heuristics; opening or closing more than one capacity level is selected as the best solution alternative compared to opening or closing only one capacity level.

In the following section we elaborate on the performances of the developed heuristic and analyze the effects of the number of possible plant locations and warehouses, the number of customers and the number of available capacity levels on the performance measures.

5.4 Analysis for Performance Measures

In this section, the effect of the problem size on the main and supportive performance measures is studied. The size of the problem is determined by the number of the potential plant and warehouse locations, the number of customers and the capacity levels. In the following sections, the effects of these problem parameters are analyzed separately. But it is impractical to compare each parameter to all of the performance measures, because even when the comparisons show a trend or a relationship between a parameter and a performance measure, this relationship may not be a causal relationship. Therefore, in order to distinguish the causal effects from the chance effects, for each parameter we select the performance measures which might be causally affected by the relevant parameter.

For example, if we compare the situations where all the problem parameters are the same except for a specific parameter and if in all the comparisons, the value of the $Gap\%$ seems to increase as the number of the specific parameter increases, we may conclude that there is a relationship between the specific parameter and the $Gap\%$ value. But this conclusion is not true. Previous researches show that the relationship between the problem size and the value of $Gap\%$ cannot be generalized as a causal effect. For some of the problem instances, $Gap\%$ value may/may not increase or decrease with the increase of the parameter. This can be interpreted in accordance with the characteristics of the problem instances where input parameters result in better approximations in the subgradient optimization that leads us to better gaps. Even if there appears to be a trend in all situations, it is a chance effect. The problem parameters and their effects on the selected performance measures are discussed in the following section.

5.4.1 Effects of the Number of Potential Plant Locations on the Performance Measures

In this section, we concentrate on the effect of the plant sites on the performance measures. We have already mentioned that $Gap\%$, $Gap_{LB}\%$ and $Gap_{UB}\%$ are not causally affected by any of these parameters. The “Avg. WHGH” cannot be affected either, because this measure is a procedure about warehouses. Hence, we select the solution duration, the “Avg. POGH”, “Avg. PCGH” and “Avg. LFC” performance measures as the relevant measures, i.e. the measures that may be affected by the increase in the number of plant locations.

Among the available problem sizes, we have considered the sizes where the only change is caused by the number of potential plant locations, and determined five such instances and presented them with the results of the relevant performance measures in the comparison tables in Table 5.5.

Table 5.5. Effects of the Number of Potential Plant Locations on the Performance Measures

I)							
Warehouse $\lvert J \rvert$	**Customer** $\lvert K \rvert$	**Capacity Level** $\lvert L \rvert$	**Plant** $\lvert I \rvert$	**Average POGH (%)**	**Average PCGH (%)**	**Average LFC (%)**	**Average Heuristic CPU (s)**
40	100	3	**10**	59.67	1.460	9.8	57.065
			20	64.32	1.150	7.8	62.855
Increase (%)				7.793	-21.233	-20.306	10.146
II)							
Warehouse $\lvert J \rvert$	**Customer** $\lvert K \rvert$	**Capacity Level** $\lvert L \rvert$	**Plant** $\lvert I \rvert$	**Average POGH (%)**	**Average PCGH (%)**	**Average LFC (%)**	**Average Heuristic CPU (s)**
40	200	3	**10**	61.72	1.780	16.1	61.72
			20	55.58	1.750	11.2	55.58
Increase (%)				-9.948	-1.685	-30.031	3.998
III)							
Warehouse $\lvert J \rvert$	**Customer** $\lvert K \rvert$	**Capacity Level** $\lvert L \rvert$	**Plant** $\lvert I \rvert$	**Average POGH (%)**	**Average PCGH (%)**	**Average LFC (%)**	**Average Heuristic CPU (s)**
50	100	3	**20**	57.20	1.520	10.1	79.778
			30	58.79	1.430	12.1	88.456
Increase (%)				2.780	-5.921	20.437	10.878

Table 5.5. Cont'd

IV)							
Warehouse $\|J\|$	**Customer** $\|K\|$	**Capacity Level** $\|L\|$	**Plant** $\|I\|$	**Average POGH (%)**	**Average PCGH (%)**	**Average LFC (%)**	**Average Heuristic CPU (s)**
50	200	3	**20**	59.32	1.490	16.9	177.632
			30	61.74	1.890	16.5	185.776
Increase (%)				4.080	26.846	-2.363	4.585
V)							
Warehouse $\|J\|$	**Customer** $\|K\|$	**Capacity Level** $\|L\|$	**Plant** $\|I\|$	**Average POGH (%)**	**Average PCGH (%)**	**Average LFC (%)**	**Average Heuristic CPU (s)**
50	500	3	**20**	45.46	2.360	30.8	759.923
			30	28.81	2.010	43.5	789.451
Increase (%)				-36.626	-14.831	41.423	3.886

It could easily be observed that the increase in the number of potential plant locations also increases the size of the problem and the average solution durations for all five comparison tables. The % of increases in the comparison table I and III, where the number of customers is equal to 100, are very close to each other and approximately 10%. In the tables II, IV and V the % of increase is approximately 4%. These tables show that the effect of the number of plant locations is very low in terms of solution duration and it is getting lower in percentage when the number of customers increases. This proves a negative interaction effect between the number of plant locations and the number of customers, which decreases the solution duration increase in percentage. Also the percentage remains almost unchanged in the tables I and III and the tables II and IV, which shows that there is no strong interaction effect between the number of plant locations and the number of warehouse locations.

With the increase of the number of possible plant locations, the "Avg. POGH", "Avg. PCGH" and "Avg. LFC" values increase in some tables and decrease in the others. Hence, we can state that the number of possible plant locations has no effect on these measures.

5.4.2 Effects of the Number of Potential Warehouse Locations on the Performance Measures

In this section, we consider the situations of problem instances, where the only change is in the number of potential warehouse locations, and all other parameters that determine the problem size remain the same. We select the solution duration, "Avg. WHGH" and "Avg. LFC" as the performance factors that may affected by the change in the number of potential warehouse locations. We have found out ten such situations presented in six comparison tables below in Table 5.6.

Table 5.6. Effects of the Number of Potential Warehouse Locations on the Performance Measures

I)						
Plant $\lvert I \rvert$	**Customers** $\lvert K \rvert$	**Capacity Level** $\lvert L \rvert$	**Warehouse** $\lvert J \rvert$	**Average WHGH (%)**	**Average LFC (%)**	**Average Heuristic CPU (s)**
5	40	3	**10**	4.50	2.3	6.822
			15	8.32	3.8	8.843
Increase (%)				84.889	63.090	29.622
II)						
Plant $\lvert I \rvert$	**Customers** $\lvert K \rvert$	**Capacity Level** $\lvert L \rvert$	**Warehouse** $\lvert J \rvert$	**Average WHGH (%)**	**Average LFC (%)**	**Average Heuristic CPU (s)**
5	50	3	**10**	8.77	4.1	8.920
			15	44.26	2.5	11.157
Increase (%)				404.675	-39.709	25.087

Table 5.6. Cont'd

III)						
Plant $\|I\|$	**Customers** $\|K\|$	**Capacity Level** $\|L\|$	**Warehouse** $\|J\|$	**Average WHGH (%)**	**Average LFC (%)**	**Average Heuristic CPU (s)**
10	100	3	**25**	68.15	4.9	39.035
			40	96.82	9.8	57.065
Increase (%)				42.069	98.381	46.190

IV)						
Plant $\|I\|$	**Customers** $\|K\|$	**Capacity Level** $\|L\|$	**Warehouse** $\|J\|$	**Average WHGH (%)**	**Average LFC (%)**	**Average Heuristic CPU (s)**
10	200	3	**25**	37.51	14.0	107.965
			40	93.75	16.1	141.009
Increase (%)				149.933	15.054	30.606

V)						
Plant $\|I\|$	**Customers** $\|K\|$	**Capacity Level** $\|L\|$	**Warehouse** $\|J\|$	**Average WHGH (%)**	**Average LFC (%)**	**Average Heuristic CPU (s)**
20	100	3	**30**	70.480	12.8	49.249
			40	92.340	7.8	62.855
			50	85.720	10.1	79.778
1st Increase (%)				31.016	-38.745	27.626
2nd Increase (%)				-7.169	29.065	26.924

VI)						
Plant $\|I\|$	**Customers** $\|K\|$	**Capacity Level** $\|L\|$	**Warehouse** $\|J\|$	**Average WHGH (%)**	**Average LFC (%)**	**Average Heuristic CPU (s)**
20	200	3	**30**	65.130	18.1	121.138
			40	78.370	11.2	146.647
			50	86.310	16.9	177.632
1st Increase (%)				20.329	-37.818	21.057
2nd Increase (%)				10.131	50.757	21.129

In all tables it can be seen that the solution duration of the developed heuristic increases in all instances drastically when the number of potential warehouse locations increases. The increment percentage varies between 21.057% and 46.190%, which shows that the effect of the number of possible warehouse locations is very strong in solution duration even when the accrual in the number of warehouse locations is so small. Table V and VI show that the increases in percentages are almost the same for the same increment in the number of potential warehouse locations. Depending on this observation, we can state that the effect of the number of potential warehouse locations may be a linear function, not nonlinear. If we compare the tables V and VI, we can see that the escalation in the percentage of the overall solution duration decreases when the number of customers increases. This might be a sign of a negative interaction effect of the number of potential warehouse locations and the number of customers on the solution duration.

The value of "Avg. WHGH" increases in nine of the ten instances, however in one instance in the comparison table V, its value decreases by 7.169% when the number of warehouse locations increases from 40 to 50. This might be an outlier, but due to this instance we cannot indicate that there is a relationship between the number of potential warehouse locations and the value of "Avg. WHGH". When the number of potential warehouse locations increases, the value of "Avg. LFC" increases in seven instances and decreases in three instances. Due to these observations, we can state that there is no effect of the number of potential warehouse locations on the value of "Avg. LFC".

5.4.3 Effects of the Number of Customers on the Performance Measures

In this section, the effect of the change in the number of customers on the selected performance measures is considered. We have considered all the performance measures and decided to seek a relationship between the number of customers and the solution duration. Because, bearing in mind our solution procedure, only the solution duration may be affected by the number of customers. Among the available problem instances, fourteen situations, in which the difference is only in the number of customers and where all the other parameters are the same, are found and presented in eight comparison tables in Table 5.7

Table 5.7. Effects of the Number of Customers on the Performance Measures

I)				
Plant $\lvert I \rvert$	**Warehouse** $\lvert J \rvert$	**Capacity Level** $\lvert L \rvert$	**Customer** $\lvert K \rvert$	**Average Heuristic CPU (s)**
5	10	3	**20**	3.552
			30	4.967
			40	6.822
			50	8.920
1st Increase (%)				39.832
2nd Increase (%)				37.344
3rd Increase (%)				30.746
II)				
Plant $\lvert I \rvert$	**Warehouse** $\lvert J \rvert$	**Capacity Level** $\lvert L \rvert$	**Customer** $\lvert K \rvert$	**Average Heuristic CPU (s)**
5	15	3	**40**	8.843
			50	11.157
Increase (%)				26.172
III)				
Plant $\lvert I \rvert$	**Warehouse** $\lvert J \rvert$	**Capacity Level** $\lvert L \rvert$	**Customer** $\lvert K \rvert$	**Average Heuristic CPU**
10	25	3	**50**	18.021
			75	27.685
			100	39.035
			200	107.965
1st Increase (%)				53.625
2nd Increase (%)				40.998
3rd Increase (%)				176.586

Table 5.7. Cont'd

IV)				
Plant $\lvert I \rvert$	**Warehouse** $\lvert J \rvert$	**Capacity Level** $\lvert L \rvert$	**Customer** $\lvert K \rvert$	**Average Heuristic CPU (s)**
10	40	3	**100**	57.065
			200	141.009
Increase (%)				147.101
V)				
Plant $\lvert I \rvert$	**Warehouse** $\lvert J \rvert$	**Capacity Level** $\lvert L \rvert$	**Customer** $\lvert K \rvert$	**Average Heuristic CPU (s)**
20	30	3	**100**	49.249
			200	121.138
Increase (%)				145.972
VI)				
Plant $\lvert I \rvert$	**Warehouse** $\lvert J \rvert$	**Capacity Level** $\lvert L \rvert$	**Customer** $\lvert K \rvert$	**Average Heuristic CPU (s)**
20	40	3	**100**	62.855
			200	146.647
Increase (%)				133.311
VII)				
Plant $\lvert I \rvert$	**Warehouse** $\lvert J \rvert$	**Capacity Level** $\lvert L \rvert$	**Customer** $\lvert K \rvert$	**Average Heuristic CPU (s)**
20	50	3	**100**	79.778
			200	177.632
			500	759.923
1st Increase (%)				122.659
2nd Increase (%)				327.806
VIII)				
Plant $\lvert I \rvert$	**Warehouse** $\lvert J \rvert$	**Capacity Level** $\lvert L \rvert$	**Customer** $\lvert K \rvert$	**Average Heuristic CPU (s)**
30	50	3	**100**	88.456
			200	185.776
			500	789.451
1st Increase (%)				110.021
2nd Increase (%)				324.947

It can be observed that in all of the situations the average solution duration increases as the number of customers increases. In some situations such as the one in the comparison tables I and II, the increase in the number of customers is small, so is the increase in the average solution duration. In the instances such as the ones in the comparison tables IV, V, VI, VII and VIII, the average solution duration increases significantly due to the high increment in the number of customers. The increase ratios in Table 5.7 show that the number of customers has a significant effect on the solution duration.

If we compare the comparison tables I and II, the tables III and IV and the tables V, VI and VII, we can see that the increment in the number of customers and the other fixed parameters are the same except for the number of potential warehouse locations. In all of these comparisons tables, the increase percentage of the average solution duration decreases. For example in the comparison table I, the increase percentage, when the number of customers increases from 40 to 50, is 30.746%. However, in the comparison table II, where the increment amount in the number of customers and all parameters have values similar to the values in the comparison table I, except for the number of possible warehouse locations, the increase percentage decreases compared to the increase percentage in the comparison table I and becomes 26.172%. The decrement in the increase percentage may be a sign of the existence of a negative interaction effect between the number of customers and the number of potential warehouse locations. Similarly, if we compare the increment percentages of the comparison tables IV and VI and the tables VII and VIII separately, we can observe that when the number of possible plant locations increases, the increment percentages decrease. But this time this decrement is not as significant as it is in the previous case. Still there might be a very weak interaction effect between the number of customers and the number of possible plant locations in the increase percentage of the solution duration.

5.4.4 Effects of the Number of Available Capacity Levels on Performance Measures

In this section, we consider the effect of increase in the number of available capacity levels on the relevant performance measures. As it is in the effect of potential plant locations section, we consider the solution duration, “Avg. POGH”, “Avg. PCGH” and “Avg. LFC” as the relevant measures that might be affected by the increase of the number of plant locations.

We find six situations where the only difference in the parameters is the number of available capacity levels, which are presented in three comparison tables in Table 5.8 below.

Table 5.8. Effects of the Number of Available Capacity Levels on Performance Measures

I)							
Plant $\lvert I \rvert$	**Warehouse** $\lvert J \rvert$	**Customer** $\lvert K \rvert$	**Capacity Level** $\lvert L \rvert$	**Average POGH (%)**	**Average PCGH (%)**	**Average LFC (%)**	**Average LH CPU (s)**
5	10	20	**3**	26.80	1.76	4.10	3.552
			5	51.47	8.05	12.30	4.479
			10	47.73	9.25	20.55	6.524
1st Increase (%)				92.052	357.386	200.000	26.099
2nd Increase (%)				-7.266	14.907	67.073	45.657
II)							
Plant $\lvert I \rvert$	**Warehouse** $\lvert J \rvert$	**Customer** $\lvert K \rvert$	**Capacity Level** $\lvert L \rvert$	**Average POGH (%)**	**Average PCGH (%)**	**Average LFC (%)**	**Average LH CPU (s)**
20	30	100	**3**	66.90	1.74	12.75	49.249
			5	49.20	2.78	20.58	56.432
			10	49.25	5.08	37.52	75.357
1st Increase (%)				-26.457	59.770	61.412	14.586
2nd Increase (%)				0.102	82.734	82.313	33.536

Table 5.8. Cont'd

III)							
Plant $\|I\|$	**Warehouse** $\|J\|$	**Customer** $\|K\|$	**Capacity Level** $\|L\|$	**Average POGH (%)**	**Average PCGH (%)**	**Average LFC (%)**	**Average LH CPU (s)**
30	50	500	**3**	28.81	2.01	43.53	789.451
			5	14.39	6.02	93.42	840.605
			10	12.64	59.11	145.11	937.701
1st Increase (%)				-50.052	199.502	114.611	6.480
2nd Increase (%)				-12.161	881.894	55.331	11.551

In all six situations in three comparison tables, it can be observed that the solution duration increases when the number of available capacity level increases. Even in the small amount of increases in the number of available capacity levels, the increase in the solution duration is quite visible. This means that the number of available capacity levels might have a causal effect on the solution duration of the developed heuristic.

The increase percentages of the "Avg. POGH" values are negative in four of the six situations; however in the other two situations the increase percentage is positive, which shows us that the number of available capacity levels has no effect on the "Avg. POGH". On the other hand, the value of "Avg. PCGH" increases on the increase of the number of available capacity levels in all situations. This may be a chance effect because we cannot find any relationship between the number of available capacity levels and the value of "Avg. POGH" which is a similar measure to "Avg. PCGH". However, the number of available capacity levels might have a causal effect on "Avg. PCGH", the underlying reason of which cannot be interpreted at first glance.

In all situations, the value of the performance measure "Avg. LFC" increases on the increment of the number of available capacity levels. This trend shows that there might be a cause and effect relationship between the number of available capacity levels and the performance measure “Avg. LFC”. This relationship can be explained in this manner: The increment in the number of available capacity levels decreases the production volume of each capacity level. Tight minimum and maximum production capacities increase the possibility of generating infeasible plants.

CHAPTER 6

CONCLUSION AND FURTHER RESEARCH

6.1 Conclusion

In this study, we have dealt with the two-stage, modular capacitated facility location problem. Our aim is to determine the locations and the capacities of plants and warehouses, and the pattern of the distribution network from the plants to customers via the warehouses, with the least total cost of opening and operating the logistics network. We model the problem as a mixed integer linear program and propose a heuristic solution based on the Lagrangean relaxation. We use the subgradient optimization algorithm to update the multipliers in the search of better solutions to obtain strong lowerbounds. In order to find feasible solutions, we have employed an allocation heuristic which uses the solutions of the lowerbound. However, lowerbound solutions do not always have to be feasible for the original problem. Therefore, we have developed a greedy heuristic in order to transform the infeasible lowerbound solutions into initial feasible solutions for the allocation heuristic.

The results of our computational study have revealed that the overall heuristic solution procedure yields reasonably good solutions. The employed subgradient optimization method is successful to find good multipliers, which takes us to tight-gap solutions. By using the Lagrangean relaxation method integrated with the subgradient optimization, we could succeed to solve a NP-hard mixed integer problem complicated with many binary variables. By exploiting the problem structure after relaxing a set of constraints, we are able solve the mentioned NP-hard problem without executing any commercial solver or external heuristic procedure. We just break the relaxed problem into two main subproblems, and then each main subproblem into many easily solvable subproblems. The solution of these subproblems can almost be obtained by inspection, which reduces the solution time drastically. By this way, the developed heuristic could solve the problems approximately 17 times faster on the average compared to CPLEX. The contributions of this study to the logistics literature are as follows:

- We have extended the modular capacitated facility location problem into two-stage by adding capacitated warehouses into the problem. To the best of our knowledge, this logistic system environment has never been studied before. For this not yet studied problem, we have developed an efficient Lagrangean relaxation based heuristic approach.

- We have coded the developed heuristic procedure using GAMS environment. Previous researchers have used GAMS frequently for solving the optimal problem. However as far as we know, GAMS has never been used for solving such complex procedure. Even many researchers are not aware of the fact that GAMS is an alternative for developing a heuristic. By using GAMS in our thesis, we also show that GAMS is an effective programming language as well.

- We have developed a greedy heuristic in order to generate initial solutions for the allocation heuristic. In the uncapacitated facility location problem, the result of the lowerbound is an initial solution for the upperbound by nature. No additional effort is needed in this problem type. In the capacitated facility location problem, strong valid inequalities can be attached to the model easily. Most of the time, these valid inequalities remove the necessity of a good initialization procedure. That is why the procedure of generating an initial solution for the upperbound heuristic seemed unworthy to be mentioned in the research. But in the modular capacitated facility location problem, the initialization procedure is more important, because there are no strong valid inequalities that can be used for this purpose. Therefore, in our study, we have given more emphasis to the initialization procedure for the first time and explained the developed procedure in details.

- We have introduced a local feasibility check stage after the allocation heuristic that tests the feasibility of each plant separately, and if necessary, re-solves the problem after adjusting the capacity levels. Previous researches have either marked the infeasible iteration and proceeded to the next iteration or developed an interchange heuristic which tries to generate feasible results without adjusting the capacity levels. Our experiments show that the allocation heuristic generates infeasible solutions ranging from 2.5% to 43.5% with the average of 12.746%. Therefore, the developed procedure is useful for generating more feasible solutions and strengthening the upperbound solution.

6.2 Further Research

Our study can be extended in two main directions: The structural extension and the conceptual extension. As for the structural extension, some of the structural properties of the problem may be changed without changing the assumptions of the presented model. In this direction, the problem addressed remains the same, because no assumptions are changed. As for the conceptual extension, some of the assumptions of the problem may be re-defined in order to develop a new problem type. After making conceptual extensions, also a structural extension may be required.

6.2.1 Structural Extensions

The possible structural extension of the problem is as follows:

- In order to strengthen the lowerbound, a different and stronger set of valid inequalities may be attached each of the main subproblems instead of the constraints (20) and (21). These new set of valid inequalities guarantees that the total maximum capacity of the opened facilities/levels is greater than the overall demand and the total minimum capacity of the opened facilities/levels is less than the overall demand. However, as Cornuejols et al. (1991) state, adding these valid inequalities makes the relaxed problem strongly NP-Hard even for the capacitated case. Therefore, in order to solve our two strongly NP-Hard main subproblems, a new solution procedure has to be employed or developed.

- In the allocation heuristic, we have to omit the minimum supply constraints of the facilities, because in the literature as far as we know, there has not been any heuristic which takes the minimum supply constraints into account. An efficient heuristic that also uses the minimum flow constraints in the allocation heuristic might be proposed.

- Our allocation heuristic consists of two stages: the warehouse allocation heuristic and the plant allocation heuristic. Solving these two stages separately worsens the solution of the upperbound, since this heuristic is not able to take into account the interactions between the stages. Therefore, a new allocation heuristic that solves the two separate stages simultaneously might be developed.

- We employ an allocation heuristic integrated with a greedy heuristic. In order to solve this allocation problem more efficiently, a meta-heuristic method might be developed. The efficiency of the meta-heuristics like the tabu search, genetic algorithms and hybrid algorithms are well proven in the literature. The developed meta-heuristic may have two stages: the initialization stage and the allocation stage as it is in our primal heuristic, or may have only one stage that makes the input feasible and allocates the customers to plants and warehouses simultaneously.

- We formulate the problem with a mixed integer linear programming model which is proved by Correia and Captivo (2003) as the best model with respect to $Gap\%$ and solution time. Other models can be extended to the two-stage environment in order to observe their performances. Also a four-indexed model can be developed to decrease the size of the mixed integer model.

- We find reasonably well upper and lowerbounds. The incorporation of those bounds to an exact solution procedure like the branch-and-bound can be an interesting research extension.

- In the second stage of our study, we relax the demand satisfaction constraint set. Other constraints may be relaxed or decomposed. Even though relaxing different constraints may yield better $Gap\%$ values, the computational burden will increase drastically as they are NP-Hard.

6.2.2 Conceptual Extensions

There are many conceptual extensions of the problem, because there are various assumptions in the model each of which can be changed in many different ways. The best possible directions are as follows:

- In our study, the warehouses are capacitated. They can be extended to a modular capacitated structure. In this case, changing the proposed solution methodology is not required, but the size of the problem will be increases polynomially with a degree of the number of the capacity levels of the warehouses. Most probably, this will increase the solution time and worsen the value of $Gap\%$.

- In our study, there is only one product which can also be assumed as a somehow aggregated product of two or more different products. Our problem can easily be extended to a multi-product case. But in this case, the specially structured knapsack problems, which are mentioned in Section 4.1.2.1 and

4.1.2.2, will be NP-Hard. In order to solve these NP-Hard problems, an efficient external knapsack heuristic as presented in the study of Martello and Toth (1990) may be employed in the heuristic procedure.

- In our study, multiple sourcing is allowed in each stage. Single-sourcing constraints may be added into the first or the second stage or both. In this case, the transportation variables of the single-sourced stage have to be replaced with some binary variables. This also makes the knapsack problems that are mentioned in the previous paragraph harder to solve. The solution methodology explained above can also be used in this case.

- In our study, all of the facilities are established simultaneously. Instead of that, the problem can be formulated as a multi-period problem. In this case, also the opening dates of each facility could be decided on in a given time period.

- We have assumed that all of the input parameters like demands and costs are deterministic. We may define some of these input parameters as uncertain parameters with a given probability density function and formulate the problem as a two-stage stochastic programming model with recourse.

REFERENCES

Agar, M.C., Salhi S., Lagrangian Heuristics Applied to a Variety of Large Capacitated Plant Location Problems, Journal of Operational Research Society, 49, 1998, 1072-1084.

Aikens, C. H., Facility Location Models for Distribution Planning, European Journal of Operational Research, 22, 1985, 263-279.

Amiri, A., Designing a Distribution Network in a Supply Chain System: Formulation and Efficient Solution Procedure, European Journal of Operational Research, 171, 2006, 567-576.

Avella, P., Benati, L., Martinez C., Dalby K., Di Girolamo, D., Dimitrijevic, B., Ghiani G., Giannikos, I., Guttmann, N., Hultberg, T. H., Fliege, J., Marin, A., Munoz Marquez, M., Ndiaye, M. M., Nickel, S., Peeters P., Perez Brito, D., Policastro, S., Saldanha de Gama, F.A., Zidda, P., Some Personal Views on the Current State and the Future of Location Analysis, European Journal of Operational Research, 104, 1998, 269-287.

Ayrım, Y. Z., A Lagrangian Heuristic for a Variant of Capacitated Facility Location with Single Source Constraints, Bilkent University, 2006, MSc Thesis.

Babaioff, M., Immorlica, N., Kempe, D., Kleinberg, R., A Knapsack Secretary Problem with Applications, In “Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques”, eds., Charikar, M., Jansen, K., Reingold, O., Rolim, J.D.P., Springer-Verlag Berlin Heidelberg, Germany, 2007.

Bahiense, L., Maculan, N.,Sagastizabal, C., The Volume Algorithm Revisited: Relation with Bundle Methods, Mathematical Programming, A94, 2002, 41-69.

Barahona, F., Chudak, F.A, Near-Optimal Solutions to Large-Scale Facility Location Problems, Discrete Optimization, 2, 2005, 35-50.

Barbarosoğlu, G., Özgür, D., Hierarchical Design of an Integrated Production and 2-Echelon Distribution System, European Journal of Operational Research, 118, 1999, 464-484.

Barcelo, J., Casanovas, A., A heuristic Lagrangean Algorithm for the Capacitated Plant Location Problem, European Journal of Operational Research, 15, 1984, 212-226.

Barcelo, J., Fernandez, E., Jornsten, K.O., Computational Results from a New Lagrangean Relaxation Algorithm for the Capacitated Plant Location Problem European Journal of Operational Research, 53, 1991, 38-45.

Beasley, J.E., An Algorithm for Solving Large Capacitated Warehouse Location Problems, European Journal of Operational Research, 33, 1988, 314-325.

Beasley, J.E., Lagrangean Heuristic for Location Problems, European Journal of Operational Research, 65, 1993, 383-399.

Beasley, J.E., Lagrangean Relaxation, In "Modern Heuristic Techniques for Combinatorial Problems", ed., C.R. Reeves, Blackwell Scientific Publications, Oxford, 1993.

Bektaş, T., Bulgak, A.A., Lagrangean-based Solution Approaches for the Generalized Problem of Locating Capacitated Warehouses, International Transactions in Operational Research, 15, 2008, 67–85.

Beltran, C., Tadonki, C., Vial, J.-Ph., Solving the p-Median Problem with a Semi-Lagrangian Relaxation, Computational Optimization and Applications, 35, 2006, 239-260.

Beltran-Royo, Vial, J.-Ph., Alonso-Ayuso, A., Solving the Uncapacitated Facility Location Problem with Semi-Lagrangian Relaxation, Optimization Online, 2007, http://www.optimization-online.org/DB_HTML/2007/02/1597.html.

Bertsekas, D.P., Tseng, P., RELAX-IV: A faster Version of the RELAX Code for Solving Minimum Cost Flow Problem, Massachusetts Institute of Technology, Laboratory for Information and Decision Systems, 1994, http://hdl.handle.net/1721.1/3392.

Bornstein, C.T., Campêlo, M., An ADD/DROP Procedure for the Capacitated Plant Location Problem, Pesquisa Operacional, 24, 2004, 151-162.

Brandeau, M.L., Chiu, S.S., An Overview of Representative Problems in Location Research, Management Science, 35, 1989, 645-674.

Broek, J., Schütz., P., Stougie, L., Tomasgard, A., Location of Slaughterhouses Under Economies of Scale, European Journal of Operational Research, 175, 2006, 740-750.

Camerini, P.M., Fratta, L., Maffioli, F., On Improving Relaxation Methods by Modified Gradient Techniques, Mathematical Programming Study, 3, 1975, 26-34.

Chen, C.-H., Ting C.-T., Combining Lagrangian Heuristic and Ant Colony System to Solve the Single Source Capacitated Facility Location Problem, Transportation Research Part E: Logistics and Transportation Review, 2007, Article in Press.

Christofides, N., Beasley, J., Extensions to a Lagrangean Relaxation Approach for the Capacitated Warehouse Location Problem, European Journal of Operational Research, 12, 1983, 19-28.

Cornuejols, G., Fisher, M.L, Nemhauser, G.L, Location of Bank Accounts to Optimize Float: An Analytic Study of Exact and Approximate Algorithms, Management Science, 23, 1977, 789-810.

Cornuejols, G., Sridharan, R., Thizy, J.M., A Comparison of Heuristics and Relaxations for the Capacitated Plant Location Problem, European Journal of Operational Research, 50, 1991, 280-297.

Correia, I., Captivo, M.E, A Lagrangean Heuristic for a Modular Capacitated Location Problem, Annals of Operations Research, 122, 2003, 141-161.

Correia, I., Captivo, M.E, Bounds for the Single Source Modular Capacitated Plant Location Problem, Computers & Operations Research, 33, 2006, 2991-3003.

Cortinhal, M. J., Captivo, M.E., Upper and Lower Bounds for the Single Source Capacitated Location Problem, European Journal of Operational Research, 151, 2003, 333-351

Crainic T.G., Frangioni A., Gendron B., Bundle-Based Relaxation Methods for Multicommodity Capacitated Fixed Charge Network Design, Discrete Applied Mathematics, 112, 2001, 73-99

Crowder, H., Computational Improvements for Subgradient Optimization, Symposia Mathematica, XIX, 1976, 357-372.

Dias, J., Captivo, M.E., Climaco, J., Efficient Primal-Dual Heuristics for a Dynamic Location Problem, Computers & Operations Research, 34, 2007, 1800-1823

Domschke, W., Drexl, A., ADD-heuristics' Starting Procedures for Capacitated Plant Location Models, European Journal of Operational Research, 21, 1985, 47-53.

Efroymson, M.A., Ray, T.L, A Branch-Bound Algorithm for Plant Location, Operations Research, 14, 1966, 361-368.

Elhedli, S., Goffin, J.L., Efficient Production-Distribution System Design, Management Science, 51, 2005, 1151-1164.

Erlenkotter, D., A Dual-Based Procedure for Uncapacitated Facility Location, Operation Research, 26, 1978, 992-1009.

Fisher, M. L., The Lagrangian Relaxation Method for Solving Integer Programming Models, Management Science, 27, 1981, 1-18.

Fumero, F., A Modified Subgradient Algorithm for Lagrangean Relaxation, Computers & Operations Research, 28, 2001, 33-52.

Gavish, B., On Obtaining the 'Best' Multipliers for a Lagrangean Relaxation for Integer Programming, Computers & Operations Research, 5, 1978, 55-71.

Geoffrion, A. M., Mc Bridge, R., Lagrangean Relaxation Applied to Capacitated Facility Location Problems, IIE Transactions, 10, 1978, 40-47

Geoffrion, A.M., Graves, G.W., Multicommodity Distribution System Design by Benders Decomposition, Management Science, 20, 1974, 329-347.

Ghosh, D., Neighborhood Search Heuristics for the Uncapacitated Facility Location Problem, European Journal of Operational Research, 150, 2003, 150-162.

Gourdin, E., Labbe, M., Laporte, G., The Uncapacitated Facility Location Problem with Client Matching, Operations Research, 48, 2000, 671-685.

Guignard, M., A Lagrangean Dual Ascent Algorithm for Simple Plant Location Problems, 35, 1988, 193-200.

Harkness, J., ReVelle, C., Facility Location with Increasing Production Costs, European Journal of Operational Research, 145, 2003, 1-13

Held, M., Karp, R.M., The Traveling Salesman Problem and Minimum Spanning Trees, Operations Research, 18, 1970, 1138-1162.

Held, M., Karp, R.M., The Traveling Salesman Problem and Minimum Spanning Trees: Part II, Mathematical Programming, 1, 1971, 6-25.

Held, M., Wolfe, P., Crowder, H.P., Validation of Subgradient Optimization, Mathematical Programming, 6, 1974, 62-88.

Hinojosa, Y., Puerto J., Fernandez, F.R., A Multiperiod Two-Echelon Multicommodity Capacitated Plant Location Problem, European Journal of Operational Research, 123, 2000, 271-291.

Holmberg, K., Ling, J., A Lagrangean Heuristic for the Facility Location Problem with Staircase Costs, European Journal of Operational Research, 97, 1997, 63-74.

Holmberg, K., Exact Solution Methods for Uncapacitated Location Problems with Convex Transportation Costs, European Journal of Operational Research, 114, 1999, 127-140.

Jang, Y.L., Jang, S.Y., Chang, B.M., Park, J., A Combined Model of Network Design and Production/Distribution Planning for a Supply Network, Computers & Industrial Engineering, 43, 2002, 263-281.

Jacobsen, S.K., Heuristics for the Capacitated Plant Location Model, European Journal of Operational Research 12, 1983, 253-261.

Jayaraman, V., An Efficient Heuristic Procedure For Practical-Sized Capacitated Warehouse Design Management, Decision Sciences, 29, 1998, 729-745.

Jayaraman, V., Pirkul H., Planning and Coordination of Production and Distribution Facilities for Multiple Commodities, European Journal of Operational Research, 133, 2001, 394-408.

Khumawala, B.M., An Efficient Branch and Bound Algorithm for the Warehouse Location Problem, Management Science, 18, 1972, B718-B731.

Klein Haneveld, W.K., van der Vlerk, M.H., Optimizing Electricity Distribution Using Integer Recourse Models, Stochastic Optimization: Algorithms and Applications, Kluwer Academic Publishers, 2000, 137-154.

Klincewicz, J.G, Luss, H., Rosenberg, E., Optimal and Heuristic Algorithms for Multiproduct Uncapacitated Facility Location, European Journal of Operational Research, 26, 1986, 251-258.

Klincewicz, J.G, Luss, H., A Dual Based Algorithm for Multiproduct Uncapacitated Facility Location, Transportation Science, 21, 1987, 198-206.

Klose A., A Branch and Bound Algorithm for an Uncapacitated Facility Location Problem with a Side Constraint, International Transactions in Operational Research, 5, 1998, 157-173.

Klose, A., An LP-Based Heuristic for Two-Stage Capacitated Facility Location Problems, The Journal of the Operational Research Society, 50, 1999, 157-166.

Klose, A., A Lagrangean Relax-and-Cut Approach for the Two-Stage Capacitated Facility Location Problem, European Journal of Operational Research, 126, 2000, 408-421

Klose, A., Görtz, S., A Branch-and-Price Algorithm for the Capacitated Facility Location Problem, European Journal of Operational Research, 179, 2007, 1109-1125

Korkel, M., On the Exact Solution of Large Scale Simple Plant Location Problems, European Journal of Operations Research, 39, 1989, 157-173.

Kuehn, A.A., Hamburger, M.J, A Heuristic Program for Locating Warehouses, Management Science, 9, 1963, 643-666.

Lee, C. Y., A Cross Decomposition Algorithm for a Multiproduct-Multitype Facility Location Problem, Computers & Operations Research, 20, 1993, 527-540.

Magee, T. M., Glover, F., Integer Programming, in "Mathematical Programming for Industrial Engineers", eds., M. Avriel and B. Golany, Marcel Dekker, Inc., New York, 1996.

Marin, A., Pelegrin, B., Applying Lagrangian Relaxation to the Resolution of Two-Stage Location Problems, Annals of Operations Research, 86, 1999, 179-198.

Marin, A., Lower bounds for the Two-Stage Uncapacitated Facility Location Problem, European Journal of Operational Research, 179, 2007, 1126-1142.

Martello, S., Toth, P., Knapsack Problems: Algorithms and Computer Implementations, John Wiley & Sons Inc., New York, 1990.

Mazzola, J.B, Neebe, A.W., Lagrangean-Relaxation-Based Solution Procedures for a Multiproduct Capacitated Facility Location Problem with Choice of Facility Type, European Journal of Operational Research, 115, 1999, 285-299.

Mirchandani, P., Francis R., Discrete Location Theory, John Wiley and Sons, 1990.

Narula, S.C., Ogbu, U.I., Lagrangean Relaxation and Decomposition in an Uncapacitated 2-Hierarchical Location-Allocation Problem, Computers & Operations Research, 12, 1985, 169-180.

Pirkul H., Efficient Algorithms for the Capacitated Concentrator Location Problem, Computers & Operations Research, 14, 1987, 197-208.

Pirkul, H., Jayaraman, V., A Multi-Commodity, Multi-Plant, Capacitated Facility Location Problem: Formulation and Efficient Heuristic Solution, Computers & Operations Research, 25, 1998, 869-878.

Poljak, B. T., Minimization of Unsmooth Functionals, USSR Computational Mathematics and Mathematical Physics, 9, 1969, 14-29.

Resende, M. G. G., Werneck, R.F., A Hybrid Multistart Heuristic for the Uncapacitated Facility Location Problem, European Journal of Operational Research, 174, 2006, 54-68.

Ro, H-B, Tcha, D-W., A Branch and Bound Algorithm for the Two-Level Uncapacitated Facility Location Problem with Some Side Constraints, European Journal of Operational Research, 18, 1984, 349-358.

Sankaran, J. K., On Solving Large Instances of the Capacitated Facility Location Problem, European Journal of Operational Research, 178, 2007, 663-676.

Simchi-Levi, D., Kaminsky, P., Simchi-Levi, E, Designing and Managing the Supply Chain: Concepts, Strategies, and Case Studies, McGraw-Hill/Irwin, 1999.

Sridharan, R., A Lagrangean Heuristic for the Capacitated Plant Location Problem with Single Source Constraints, European Journal of Operations Research, 66, 1993, 305-312.

Tragantalerngsak, S., Holt, J., Rönnqvist, M., Lagrangian Heuristics for the Two-Echelon, Single-Source, Capacitated Facility Location Problem, European Journal of Operational Research, 102, 1997, 611-625.

Tragantalerngsak, S., Holt, J., Rönnqvist, M., An Exact Method for the Two-Echelon, Single Source, Capacitated Facility Location Problem, European Journal of Operational Research, 123, 2000, 473-489.

Van Roy, T. J., Erlenkotter, D., A Dual-Based Procedure for Dynamic Facility Location, Management Science, 28, 1982, 1091-1105.

Vidal, C.J., Goetschalckx, M., Strategic Production-Distribution Models: A critical Review with Emphasis on Global Supply Chain Models, European Journal of Operational Research, 98, 1997, 1-18.

Zhou, Y., Improved Multi-Unit Auction Clearing Algorithms with Interval (Multiple-Choice) Knapsack Problems, In “Algorithms and Computation”, ed., Asano, T., Springer-Verlag Berlin Heidelberg, Germany, 2006.

APPENDIX 1

Table 1. The Detailed Results of the Problem Instances

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
5x10x20	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	73411	74262	74189	1.059	0.098	1.159	3.666	1.582	2	2	0	0	193	0	35	3	155	14	4	0	5	5	54
2	73582	75051	74833	1.700	0.291	1.996	3.755	2.614	1	1	0	0	619	52	85	334	148	15	1	0	10	4	133
3	74275	76278	75378	1.485	1.194	2.697	3.656	1.161	0	0	0	0	422	14	0	408	0	7	4	0	3	0	72
4	69138	71272	71272	3.087	0	3.087	3.445	1.342	3	3	0	0	213	12	0	1	200	10	3	0	7	0	37
5	84758	85336	85198	0.519	0.162	0.682	3.685	5.020	0	0	0	0	397	331	0	66	0	19	1	0	18	0	25
6	70600	71805	71686	1.538	0.166	1.707	3.515	2.565	3	3	0	0	186	0	1	185	0	28	4	0	12	12	12
7	88388	89750	89654	1.432	0.107	1.541	3.495	1.183	7	7	0	0	257	0	0	257	0	33	4	0	29	0	6
8	50483	51696	51477	1.969	0.425	2.403	3.365	1.553	0	0	0	0	103	0	1	2	100	14	4	0	4	6	11
9	65875	67087	66886	1.535	0.301	1.840	3.264	1.432	7	7	0	0	97	0	0	18	79	22	3	0	0	19	6
10	80711	82263	81728	1.260	0.655	1.923	3.676	1.393	2	2	0	0	193	0	35	3	155	14	4	0	5	5	54
			Avg	**1.558**	**0.340**	**1.904**	**3.552**	**1.985**	0.250	100.00	0.000	0.000	26.800	15.261	5.858	47.649	31.231	1.760	18.182	0.000	52.841	28.977	4.1

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
5x10x20x5	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	73041	75353	74217	1.610	1.531	3.166	4.617	2.964	19	19	0	0	567	0	1	22	544	20	1	0	18	1	188
2	65679	67846	67659	3.014	0.276	3.299	4.206	1.713	0	0	0	0	202	68	0	119	15	32	2	0	29	1	30
3	69643	71920	70871	1.763	1.480	3.269	4.857	1.743	0	0	0	0	941	901	1	29	10	47	23	0	20	4	53
4	59119	61379	61274	3.645	0.171	3.823	4.486	0.881	0	0	0	0	671	0	0	671	0	312	297	0	14	1	207
5	70820	71755	71507	0.969	0.347	1.320	4.236	1.242	31	31	0	0	203	1	0	0	202	42	0	0	32	10	63
6	66763	69366	69366	3.899	0.000	3.899	4.686	1.793	2	2	0	0	834	794	0	36	4	128	109	0	17	2	146
7	50443	52033	52029	3.144	0.008	3.152	4.496	1.462	0	0	0	0	547	9	0	17	521	18	4	0	12	2	22
8	60265	61438	61438	1.947	0.000	1.947	4.647	2.193	308	308	0	0	339	0	0	332	7	22	0	0	22	0	25
9	67803	68822	68704	1.329	0.172	1.503	4.276	0.942	14	14	0	0	582	0	0	10	572	42	1	0	32	9	248
10	51286	51785	51620	0.651	0.320	0.972	4.286	1.261	0	0	0	0	261	14	0	232	15	142	51	0	51	40	248
			Avg.	**2.197**	**0.431**	**2.635**	**4.479**	**1.619**	3.740	100.00	0.000	0.000	51.470	34.719	0.039	28.521	36.720	8.050	60.621	0.000	30.683	8.696	12.3

Table 1. Cont’d

	Value of						Solution Duration			WOH		WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
5x10x20x10	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	54166	55733	55248	1.997	0.878	2.893	6.379	4.136	14	14	0	0	586	1	1	10	574	41	4	0	35	2	45
2	52103	53259	53142	1.994	0.220	2.219	6.589	4.637	4	4	0	0	504	18	14	111	361	83	0	0	82	1	135
3	50288	51976	51069	1.552	1.776	3.356	6.740	2.504	0	0	0	0	663	0	0	661	0	36	0	0	36	0	496
4	67969	68952	68949	1.441	0.004	1.446	5.838	1.121	0	0	0	0	91	0	0	23	68	39	0	0	38	1	46
5	55350	56277	55996	1.167	0.502	1.674	6.589	1.753	1	1	0	0	573	5	3	443	122	193	0	0	192	1	134
6	59904	62913	62055	3.591	1.383	5.023	6.780	4.377	0	0	0	0	568	1	0	114	453	343	297	0	45	1	391
7	61070	61905	61854	1.283	0.082	1.367	6.740	1.723	1	1	0	0	570	16	0	500	54	49	0	0	48	1	264
8	62288	63376	63109	1.318	0.423	1.746	6.750	1.592	5	5	0	0	560	0	0	12	548	21	1	0	20	0	323
9	58018	60310	60252	3.850	0.096	3.950	6.189	6.078	3	3	0	0	252	0	0	45	207	82	12	0	66	4	173
10	65284	66735	66299	1.555	0.658	2.222	6.650	5.538	295	295	0	0	406	0	5	401	0	38	0	0	32	6	48
			Avg.	**1.975**	**0.602**	**2.590**	**6.524**	**3.346**	3.230	100.00	0.000	0.000	47.730	0.859	0.482	48.607	50.010	9.250	33.946	0.000	64.216	1.838	20.6

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
5x10x30	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	115692	119863	119395	3.201	0.392	3.605	4.897	5.340	1	1	0	0	605	279	0	326	0	114	0	0	65	49	169
2	112486	125715	125060	11.178	0.524	11.760	5.248	2.685	0	0	0	0	656	0	0	656	0	17	4	0	13	0	224
3	126313	127040	127039	0.575	0.001	0.576	4.787	2.124	4	4	0	0	340	39	0	301	0	35	8	0	27	0	244
4	120320	124119	124017	3.072	0.082	3.157	5.147	3.256	4	4	0	0	642	105	0	21	516	20	1	0	16	3	200
5	104235	108207	108063	3.673	0.133	3.811	4.827	2.614	0	0	0	0	592	0	0	592	0	23	5	0	0	18	0
6	100381	103808	103163	2.772	0.625	3.414	5.267	2.464	364	364	0	0	481	1	0	3	477	12	2	0	4	6	157
7	101705	103855	103600	1.863	0.246	2.114	4.978	1.052	46	46	0	0	242	1	0	234	7	32	5	0	27	0	8
8	98771	100695	100695	1.948	0	1.948	4.897	1.663	0	0	0	0	116	1	0	18	97	12	7	0	4	1	10
9	132932	135826	135599	2.006	0.167	2.177	4.977	2.144	0	0	0	0	458	0	0	458	0	26	2	0	0	24	261
10	119523	122606	122497	2.488	0.089	2.579	4.646	1.482	0	0	0	0	158	0	0	0	158	34	6	0	21	7	3
			Avg.	**3.278**	**0.226**	**3.514**	**4.967**	**2.482**	4.190	100.00	0.000	0.000	42.900	9.930	0.000	60.816	29.254	3.250	12.308	0.000	54.462	33.231	12.8

Table 1. Cont'd

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
5x10x40	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	153043	153713	153597	0.362	0.076	0.438	7.041	4.688	57	57	0	0	135	0	2	20	113	21	1	0	1	19	35
2	132648	137111	137063	3.328	0.035	3.364	6.830	2.534	314	314	0	0	176	0	0	176	0	24	1	0	22	1	5
3	128495	129722	129037	0.422	0.531	0.955	6.750	1.592	0	0	0	0	105	0	0	102	3	20	7	0	12	1	9
4	144477	148130	147436	2.048	0.471	2.529	6.780	1.453	59	59	0	0	181	10	0	17	154	25	3	0	7	15	13
5	165940	170491	170011	2.453	0.282	2.743	6.660	3.205	3	3	0	0	429	229	0	11	189	39	8	0	9	22	2
6	127928	130348	130048	1.657	0.231	1.892	6.780	1.843	3	3	0	0	0	319	0	0	319	13	10	0	0	3	38
7	159088	162934	161961	1.806	0.601	2.417	6.709	2.384	11	11	0	0	537	0	0	1	536	34	6	0	0	28	49
8	174005	181402	180180	3.549	0.678	4.251	6.740	3.706	0	0	0	0	445	0	0	25	420	47	7	0	23	17	33
9	216122	220070	219286	1.464	0.358	1.827	6.840	3.235	1	1	0	0	228	8	0	189	31	101	7	0	94	0	29
10	179537	183656	183056	1.960	0.328	2.294	7.090	3.886	2	2	0	0	401	2	0	399	0	25	2	0	0	23	20
			Avg.	**1.905**	**0.359**	**2.271**	**6.822**	**2.853**	4.500	100.00	0.000	0.000	26.370	21.540	0.076	35.647	66.932	3.490	14.900	0.000	48.138	36.963	2.3

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
5x10x50	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	224986	230717	229906	2.187	0.353	2.547	9.053	7.891	0	0	0	0	276	6	0	39	231	17	2	0	14	1	30
2	177841	181423	181355	1.976	0.037	2.014	8.973	7.030	14	14	0	0	516	0	0	516	0	37	6	0	0	31	219
3	168891	171377	170866	1.169	0.299	1.472	8.882	7.641	365	365	0	0	272	1	0	0	271	36	1	0	0	35	0
4	175536	177438	176086	0.313	0.768	1.084	8.672	4.617	28	28	0	0	123	0	0	0	123	6	1	0	5	0	11
5	195010	196801	196137	0.578	0.339	0.918	8.732	5.187	0	0	0	0	288	2	0	262	24	43	0	0	0	43	27
6	189660	191895	190887	0.647	0.528	1.178	8.732	3.605	79	79	0	0	87	0	0	0	87	74	19	0	46	9	33
7	160334	166812	166706	3.974	0.064	4.040	8.352	5.648	7	7	0	0	372	0	21	351	0	36	6	0	0	30	2
8	226986	234948	232475	2.418	1.064	3.508	9.343	6.940	16	16	0	0	616	3	430	183	0	68	49	0	19	0	49
9	175777	182184	180049	2.430	1.186	3.645	9.503	4.346	367	367	0	0	557	0	0	557	0	10	3	0	7	0	21
10	202232	205260	204158	0.952	0.540	1.497	8.953	5.799	1	1	0	0	114	25	0	89	0	29	1	0	0	28	21
			Avg.	**1.664**	**0.518**	**2.190**	**8.920**	**5.870**	8.770	100.00	0.000	0.000	32.210	1.149	14.002	61.999	22.850	3.560	24.719	0.000	25.562	49.719	4.1

Table 1. Cont'd

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
5x15x40	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	150873	152127	152125	0.830	0.001	0.831	8.923	10.786	0	0	0	0	121	0	0	121	0	21	0	0	7	14	49
2	185950	188821	187788	0.988	0.550	1.544	8.883	5.448	6	6	0	0	96	0	1	95	0	15	7	0	8	0	19
3	165304	167644	166586	0.776	0.635	1.416	7.912	4.316	0	0	0	0	465	160	280	3	22	17	4	0	13	0	9
4	145203	149458	147858	1.828	1.082	2.930	8.622	5.959	0	0	0	0	698	0	0	698	0	18	0	0	7	11	7
5	153141	157142	156713	2.332	0.274	2.612	8.773	13.060	175	175	0	0	771	0	0	771	0	18	0	0	14	4	209
6	197035	204245	203567	3.315	0.333	3.659	9.193	7.251	5	5	0	0	433	0	0	420	13	17	0	0	0	17	29
7	192726	200526	199986	3.767	0.270	4.047	9.443	2.284	637	637	0	0	615	2	0	579	34	17	1	0	16	0	13
8	157454	159418	159166	1.088	0.158	1.248	9.133	19.730	9	9	0	0	193	1	2	190	0	15	1	0	3	11	5
9	154892	157402	156964	1.338	0.279	1.621	8.643	15.123	0	0	0	0	105	0	0	0	105	14	13	0	1	0	26
10	131958	136220	135822	2.928	0.293	3.230	8.903	8.502	0	0	0	0	524	0	0	509	15	21	1	0	8	12	14
			Avg.	**1.919**	**0.388**	**2.314**	**8.843**	**9.246**	8.320	100.00	0.000	0.000	40.210	4.054	7.038	84.208	4.700	1.730	15.607	0.000	44.509	39.884	3.8

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
5x15x50	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	223614	225270	225020	0.629	0.111	0.740	11.166	5.958	304	304	0	0	303	275	0	0	28	16	0	0	0	16	22
2	198122	203903	203471	2.700	0.212	2.918	11.406	12.928	535	535	0	0	441	2	0	439	0	8	2	0	6	0	11
3	207502	213826	213351	2.819	0.223	3.048	11.176	14.260	939	939	0	0	362	0	31	331	0	9	2	0	0	7	48
4	218324	226644	226367	3.684	0.122	3.811	11.316	3.625	558	558	0	0	364	0	0	364	0	18	3	0	0	15	22
5	183547	188729	188680	2.797	0.026	2.823	10.375	8.192	2	2	0	0	242	0	0	0	242	8	3	0	0	5	6
6	158451	163526	163173	2.980	0.216	3.203	11.106	5.979	943	943	0	0	269	0	2	267	0	30	4	0	16	10	22
7	192330	199763	199164	3.553	0.301	3.865	11.507	17.816	912	912	0	0	255	0	0	253	2	17	1	0	0	16	3
8	186564	190295	188901	1.252	0.738	2.000	10.646	14.070	37	37	0	0	879	761	100	0	18	17	0	0	0	17	81
9	199322	202810	202446	1.567	0.180	1.750	11.577	16.904	196	196	0	0	93	0	0	91	2	10	0	5	0	5	12
10	209450	217101	216971	3.591	0.060	3.653	11.296	14.621	0	0	0	0	407	2	0	0	405	15	2	0	0	13	22
			Avg.	**2.557**	**0.219**	**2.781**	**11.157**	**11.435**	44.260	100.00	0.000	0.000	36.150	28.769	3.679	48.271	19.281	1.480	11.486	3.378	14.865	70.270	2.5

Table 1. Cont'd

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
10x25x50	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	174718	179646	177800	1.764	1.038	2.821	18.307	8.442	434	434	0	0	940	762	35	143	0	12	0	0	11	1	31
2	198267	204384	203839	2.810	0.267	3.085	17.395	9.123	948	948	0	0	269	0	0	269	0	17	1	0	3	13	19
3	189384	194017	193270	2.052	0.387	2.446	18.367	11.577	962	956	6	0	936	854	0	60	22	8	0	0	0	8	59
4	172362	174068	173176	0.472	0.515	0.990	18.176	10.805	690	690	0	0	67	1	0	66	0	13	4	0	9	0	20
5	196109	198753	198287	1.110	0.235	1.348	17.455	7.210	647	647	0	0	114	0	0	114	0	11	2	0	2	7	28
6	157194	159149	158522	0.845	0.396	1.244	17	105.071	28	28	0	0	470	18	323	129	0	12	0	0	12	0	54
7	181832	186768	185659	2.105	0.597	2.714	18.487	15.452	923	923	0	0	852	0	779	73	0	16	0	0	5	11	73
8	170586	175168	174619	2.364	0.314	2.686	18.306	113.423	16	16	0	0	570	0	526	20	24	16	3	0	10	3	48
9	197594	198742	198258	0.336	0.244	0.581	17.675	7.831	307	307	0	0	338	0	0	336	2	15	5	0	9	1	26
10	161140	165590	164182	1.888	0.858	2.762	18.607	9.454	498	498	0	0	669	496	0	163	10	10	0	0	0	10	89
			Avg.	**1.575**	**0.485**	**2.068**	**18.021**	**29.839**	54.530	99.890	0.110	0.000	52.250	40.785	31.828	26.278	1.110	1.300	11.538	0.000	46.923	41.538	4.5

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
10x25x75	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	259891	264281	263236	1.287	0.397	1.689	28.782	12.067	743	743	0	0	722	0	671	50	1	28	4	0	14	10	47
2	290036	298023	297250	2.487	0.260	2.754	27.409	33.438	962	960	2	0	875	706	0	167	2	17	2	0	15	0	91
3	297027	314457	308654	2.963	1.880	5.868	28.280	3870.355	600	600	0	0	579	0	0	579	0	16	1	0	0	15	25
4	280560	286382	285053	3.915	0.466	2.075	26.388	10.616	908	908	0	0	344	0	0	344	0	20	2	0	1	17	26
5	281503	288139	286286	1.699	0.647	2.357	27.680	11.566	957	957	0	0	447	0	0	445	2	19	5	0	13	1	48
6	276340	286177	283769	2.688	0.849	3.560	27.499	85.303	967	960	7	0	462	159	0	287	16	5	0	0	1	4	99
7	286573	293082	291957	1.879	0.385	2.271	27.279	72.264	607	607	0	0	504	0	187	317	0	15	0	0	0	15	189
8	296464	302842	302644	2.084	0.065	2.151	27.750	17.495	538	538	0	0	541	0	83	458	0	22	1	0	17	4	57
9	250860	258380	257129	2.499	0.487	2.998	28.060	89.448	964	964	0	0	868	7	794	45	22	13	0	0	11	2	32
10	255024	261304	260014	1.957	0.496	2.463	27.720	18.977	732	732	0	0	567	2	363	202	0	11	2	0	9	0	41
			Avg.	**2.346**	**0.593**	**2.819**	**27.685**	**422.153**	79.780	99.887	0.113	0.000	59.090	14.791	35.505	48.976	0.728	1.660	10.241	0.000	48.795	40.964	6.6

Table 1. Cont'd

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
10x25x100	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	380305	389300	387488	1.889	0.468	2.365	38.755	26.818	954	954	0	0	705	0	0	703	2	15	2	0	0	13	75
2	368711	379380	378842	2.748	0.142	2.894	39.547	46.127	897	897	0	0	960	817	0	143	0	13	0	0	11	2	25
3	386396	393904	389758	0.870	1.064	1.943	40.809	26.048	947	947	0	0	215	0	0	215	0	11	0	0	11	0	28
4	371882	385734	381520	2.592	1.105	3.725	39.627	61.579	170	170	0	0	503	0	458	45	0	17	2	0	0	15	44
5	381466	389098	386653	1.360	0.632	2.001	38.946	14.762	384	384	0	0	694	0	0	692	2	19	0	0	0	19	24
6	366448	375461	373477	1.918	0.531	2.460	38.505	1273.871	965	965	0	0	431	155	0	270	6	20	0	0	0	20	39
7	342190	352354	346716	1.323	1.626	2.970	37.914	101.516	940	939	1	0	475	0	0	475	0	39	8	0	20	11	152
8	322213	328658	326357	1.286	0.705	2.000	37.544	17.194	232	232	0	0	575	0	499	76	0	8	1	0	2	5	26
9	348809	359683	359171	2.971	0.143	3.117	40.058	24.285	864	864	0	0	786	765	0	19	2	21	1	0	0	20	40
10	347611	349891	348999	0.399	0.256	0.656	38.645	69.560	462	462	0	0	113	8	0	103	2	15	0	0	7	8	41
			Avg.	**1.736**	**0.667**	**2.413**	**39.035**	**166.176**	68.150	99.985	0.015	0.000	54.570	31.977	17.537	50.229	0.257	1.780	7.865	0.000	28.652	63.483	4.9

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
10x25x200	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	791828	815196	811097	2.433	0.505	2.951	103.809	54.318	882	879	3	0	463	134	0	150	179	26	3	0	18	5	76
2	717469	727464	725940	1.181	0.210	1.393	108.807	51.724	238	238	0	0	452	0	0	452	0	23	0	0	0	23	395
3	702898	719697	719209	2.320	0.068	2.390	106.724	62.530	413	413	0	0	674	640	0	2	32	13	1	0	4	8	84
4	768139	786213	782430	1.860	0.483	2.353	108.766	57.293	143	143	0	0	679	5	641	8	25	24	0	0	1	23	110
5	761958	785487	780282	2.405	0.667	3.088	109.588	1366.175	1	1	0	0	120	1	1	0	118	18	2	0	2	14	43
6	647738	659288	657919	1.572	0.208	1.783	108.556	61.478	86	86	0	0	491	0	0	23	468	24	0	0	0	24	300
7	786202	807036	802457	2.068	0.571	2.650	105.512	19.378	230	230	0	0	507	6	0	501	0	19	1	0	17	1	29
8	668396	688901	688296	2.977	0.088	3.068	107.605	32.126	720	720	0	0	138	0	0	4	134	15	0	0	3	12	51
9	741304	766618	753068	1.587	1.799	3.415	110.639	91.051	862	862	0	0	278	0	13	265	0	24	0	0	3	21	195
10	710435	725640	724305	1.952	0.184	2.140	109.647	77.301	176	176	0	0	359	0	0	356	3	18	0	0	0	18	112
			Avg.	**2.036**	**0.478**	**2.523**	**107.965**	**187.337**	37.510	99.920	0.080	0.000	41.610	18.890	15.741	42.322	23.047	2.040	3.431	0.000	23.529	73.039	14.0

Table 1. Cont'd

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
10x40x100	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	399502	408282	404318	1.205	0.980	2.198	57.743	293.703	970	918	49	3	646	340	3	292	11	13	1	0	12	0	28
2	416976	428368	426372	2.253	0.468	2.732	55.189	283.778	969	946	16	7	874	793	0	81	0	14	0	0	0	14	70
3	415648	424774	422213	1.579	0.607	2.196	56.892	440.454	966	804	158	4	639	0	0	639	0	13	1	0	9	3	92
4	402031	408681	405921	0.968	0.680	1.654	56.311	73.476	966	910	42	14	224	1	0	223	0	15	0	0	15	0	24
5	358671	371775	368847	2.837	0.794	3.653	55.310	1231.421	971	971	0	0	845	0	0	843	2	14	2	0	0	12	149
6	361094	371109	368550	2.065	0.694	2.773	58.644	300.733	967	638	297	32	483	0	0	483	0	27	8	0	14	5	259
7	415721	430550	427915	2.933	0.616	3.567	57.012	164.898	968	945	21	2	399	0	0	399	0	8	1	0	1	6	30
8	414019	423566	421709	1.857	0.440	2.306	58.835	58.915	969	153	789	27	638	187	0	222	229	18	3	0	9	6	231
9	413227	420573	419773	1.584	0.191	1.778	57.503	141.924	969	666	248	55	506	0	13	23	470	8	1	0	0	7	58
10	454548	462686	461900	1.617	0.170	1.790	57.212	107.255	967	825	95	47	713	0	657	56	0	16	1	0	0	15	39
			Avg.	**1.890**	**0.564**	**2.465**	**57.065**	**309.656**	96.820	80.314	17.713	1.973	59.670	22.138	11.279	54.651	11.932	1.460	12.329	0.000	41.096	46.575	9.8

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
10x40x200	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	807619	829087	825274	2.186	0.462	2.658	137.057	494.942	938	932	4	2	465	0	452	13	0	14	1	0	10	3	30
2	900715	913965	909830	1.012	0.454	1.471	139.961	25.396	869	869	0	0	665	1	0	662	2	23	9	0	0	14	292
3	817012	825806	819063	0.251	0.823	1.076	144.047	14.581	906	906	0	0	368	40	0	328	0	21	0	0	0	21	480
4	837232	867336	861262	2.870	0.705	3.596	139.460	1547.826	967	656	307	4	855	646	6	1	202	23	2	0	3	18	105
5	828699	850521	840475	1.421	1.195	2.633	141.553	604.550	946	935	7	4	640	0	1	63	576	29	2	0	18	9	336
6	740558	751357	748316	1.048	0.406	1.458	139.190	57.573	958	958	0	0	274	8	259	2	5	15	0	0	0	15	18
7	849740	868693	867771	2.122	0.106	2.231	140.653	108.856	936	665	255	16	917	695	0	2	220	7	0	0	0	7	44
8	881603	912935	903799	2.518	1.011	3.554	146.591	418.121	966	961	5	0	960	901	0	59	0	20	0	0	3	17	19
9	798941	824704	823819	3.114	0.107	3.225	138.689	200.889	970	780	175	15	700	434	0	3	263	14	0	0	0	14	244
10	834928	859394	858293	2.798	0.128	2.930	142.886	293.432	919	919	0	0	328	0	0	32	296	12	0	0	6	6	37
			Avg.	**1.934**	**0.540**	**2.483**	**141.009**	**376.617**	93.750	91.531	8.032	0.437	61.720	44.151	11.633	18.876	25.340	1.780	7.865	0.000	22.472	69.663	16.1

Table 1. Cont'd

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
20x30x100	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	291322	299090	293883	0.879	1.772	2.666	48.270	443.047	676	453	153	70	476	33	7	40	396	24	5	0	0	19	71
2	333330	340308	336693	1.009	1.074	2.093	49.862	124.710	727	720	7	0	741	0	0	24	717	14	1	0	2	11	52
3	344352	349087	347739	0.984	0.388	1.375	49.241	102.057	801	801	0	0	740	565	0	20	155	18	1	0	0	17	204
4	375627	379520	377773	0.571	0.462	1.037	50.984	458.600	638	638	0	0	459	11	190	250	8	16	2	0	0	14	236
5	321000	328635	323830	0.882	1.484	2.379	50.212	131.139	933	923	10	0	864	180	343	166	175	17	0	0	1	16	208
6	353272	359165	356765	0.989	0.673	1.668	48.890	1173.207	781	781	0	0	663	3	563	97	0	15	2	0	12	1	244
7	324129	329998	328405	1.319	0.485	1.811	48.169	6269.906	796	787	8	1	726	6	653	46	21	13	0	0	7	6	39
8	349250	357175	354456	1.491	0.767	2.269	48.510	149.916	622	241	266	115	599	0	1	65	533	15	1	0	1	13	74
9	322586	331183	327119	1.405	1.242	2.665	49.411	87.116	665	591	67	7	856	810	3	41	2	17	2	0	5	10	45
10	305340	310007	307790	0.802	0.720	1.528	48.940	221.959	409	409	0	0	566	0	0	461	105	25	1	0	13	11	102
			Avg.	**1.033**	**0.907**	**1.949**	**49.249**	**916.166**	70.480	90.011	7.250	2.738	66.900	24.036	26.308	18.087	31.570	1.740	8.621	0.000	23.563	67.816	12.8

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
20x30x100x5	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	290296	298030	292808	0.865	1.783	2.664	59.095	5146.781	908	906	2	0	775	4	747	3	21	30	3	0	22	5	154
2	290520	294751	293012	0.858	0.593	1.456	55.459	1148.711	882	854	24	4	497	0	19	239	239	24	4	0	16	4	210
3	284237	288435	287081	1.001	0.472	1.477	58.224	1845.434	656	643	13	0	708	0	0	97	611	28	1	0	23	4	143
4	285595	292131	287794	0.770	1.507	2.289	57.853	139.421	632	332	78	222	576	0	1	78	497	27	1	0	24	2	410
5	287494	292382	*	*	*	1.700	56.892	5449.185	597	597	0	0	637	469	0	72	96	30	3	0	14	13	347
6	305281	310818	308002	0.891	0.914	1.814	55.570	598.390	479	479	0	0	467	2	0	465	0	27	0	0	24	3	192
7	301660	307597	304578	0.967	0.991	1.968	55.951	866.426	610	610	0	0	498	0	19	85	394	32	0	0	20	12	307
8	268411	273342	271536	1.164	0.665	1.837	55.360	106.323	565	420	130	15	237	1	0	236	0	23	3	0	16	4	74
9	303765	309367	306729	0.976	0.860	1.844	56.241	3555.853	787	787	0	0	317	1	1	226	89	27	1	0	19	7	140
10	283333	286484	285184	0.653	0.456	1.112	53.677	633.051	775	775	0	0	208	1	0	23	184	30	2	0	13	15	81
			Avg.	**0.905**	**0.916**	**1.816**	**56.432**	**1948.958**	68.910	92.918	3.584	3.497	49.200	9.715	15.996	30.976	43.313	2.780	6.475	0.000	68.705	24.820	20.6

Table 1. Cont'd

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
20x30x100x10	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	237363	239110	238722	0.572	0.163	0.736	71.914	90.970	728	715	6	7	175	10	1	30	134	84	8	0	63	13	159
2	260558	264965	261997	0.552	1.133	1.691	79.084	3891.606	742	676	46	20	669	8	0	201	460	31	3	0	26	2	591
3	281978	283787	282790	0.288	0.353	0.642	74.577	70.992	716	694	18	4	482	2	2	478	0	52	0	0	38	14	185
4	248411	253173	250361	0.785	1.123	1.917	76.540	6495.481	586	212	208	166	417	0	5	4	408	63	27	0	25	11	499
5	278471	283028	280633	0.776	0.853	1.636	74.487	767.613	592	514	23	55	367	5	31	328	3	27	2	0	24	1	557
6	264868	268236	266624	0.663	0.605	1.272	74.287	3363.497	667	667	0	0	639	154	11	122	352	37	2	0	32	3	448
7	256882	258242	258050	0.455	0.074	0.529	70.772	54.348	542	349	153	40	75	1	0	73	1	33	7	0	25	1	87
8	241798	247126	244443	1.094	1.098	2.203	77.001	8924.053	763	763	0	0	831	0	20	77	734	46	6	0	39	1	242
9	268896	274211	271380	0.924	1.043	1.977	77.371	13439.525	754	754	0	0	641	2	0	634	5	52	6	0	31	15	502
10	245812	251451	247454	0.668	1.615	2.294	77.541	2258.457	558	558	0	0	629	10	3	62	554	83	2	0	73	8	482
			Avg.	**0.678**	**0.806**	**1.490**	**75.357**	**3935.654**	66.480	88.779	6.829	4.392	49.250	3.898	1.482	40.792	53.827	5.080	12.402	0.000	74.016	13.583	37.5

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
20x30x200	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	660922	670860	668257	1.110	0.390	1.504	123.338	561.989	799	376	342	81	772	134	281	144	213	14	0	0	2	12	95
2	656705	668838	666089	1.429	0.413	1.848	124.159	429.197	530	530	0	0	381	8	0	42	331	22	2	0	3	17	205
3	661423	672869	670376	1.354	0.372	1.731	121.665	1088.606	845	833	11	1	388	5	215	69	99	31	4	0	25	2	243
4	735587	748374	743795	1.116	0.616	1.738	120.223	229.711	390	390	0	0	546	4	508	34	0	20	0	0	1	19	195
5	647427	658766	655026	1.174	0.571	1.751	118.941	362.041	644	644	0	0	530	0	507	23	0	30	5	0	24	1	380
6	684895	701599	694401	1.388	1.037	2.439	125.000	830.605	360	360	0	0	775	739	2	3	31	20	2	0	0	18	87
7	644684	660878	*	*	*	2.512	117.159	7307.127	853	852	1	0	477	56	3	418	0	26	6	0	0	20	138
8	665648	678792	671854	0.932	1.033	1.975	119.532	270.659	725	689	31	5	471	12	15	413	31	30	6	0	4	20	199
9	685794	705718	*	*	*	2.905	121.484	6584.508	725	725	0	0	220	3	0	217	0	26	3	0	2	21	132
10	677972	692372	684747	0.999	1.114	2.124	119.883	541.879	642	565	72	5	377	5	0	368	4	25	6	0	17	2	132
			Avg.	**1.188**	**0.693**	**2.053**	**121.138**	**1820.632**	65.130	91.571	7.017	1.413	49.370	19.567	31.011	35.062	14.361	2.440	13.934	0.000	31.967	54.098	18.1

Table 1. Cont'd

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
20x40x100	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	338289	342968	340838	0.753	0.625	1.383	62.270	38.987	809	809	0	0	306	0	3	300	3	11	1	0	0	10	40
2	335658	346166	339794	1.232	1.875	3.131	62.961	1172.706	962	957	5	0	631	15	292	323	1	10	2	0	4	4	135
3	335288	341403	337985	0.804	1.011	1.824	65.554	57.452	958	958	0	0	784	21	615	144	4	8	1	0	0	7	39
4	334004	342888	338864	1.455	1.187	2.660	62.941	3262.762	962	944	13	5	582	0	1	47	534	14	3	0	0	11	80
5	329551	335119	332574	0.917	0.765	1.689	59.826	200.799	888	887	1	0	814	684	0	129	1	8	3	0	2	3	51
6	351428	359470	354061	0.749	1.528	2.288	63.822	946.581	889	889	0	0	641	14	501	3	123	12	1	0	6	5	112
7	331741	340626	337801	1.827	0.836	2.678	63.952	2435.592	942	889	48	5	924	0	783	141	0	12	2	0	1	9	50
8	340289	345315	341667	0.405	1.068	1.477	62.861	574.106	958	942	14	2	552	45	3	500	4	9	1	0	6	2	89
9	363316	368565	367467	1.143	0.299	1.445	62.550	10.675	960	960	0	0	433	0	0	431	2	7	0	0	0	7	36
10	380722	387858	384282	0.935	0.931	1.874	61.809	6401.305	906	879	19	8	765	427	8	311	19	24	2	0	11	11	149
			Avg.	**1.022**	**1.013**	**2.045**	**62.855**	**1510.097**	92.340	98.700	1.083	0.217	64.320	18.750	34.297	36.210	10.743	1.150	13.913	0.000	26.087	60.000	7.8

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
20x40x200	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	759964	773466	763497	0.465	1.306	1.777	148.213	132.320	639	240	353	46	442	0	0	208	234	22	0	0	0	22	112
2	635340	649545	642963	1.200	1.024	2.236	147.262	18192.492	907	899	7	1	719	3	0	593	123	8	2	0	5	1	88
3	692283	703239	696920	0.670	0.907	1.583	144.598	337.655	173	173	0	0	650	407	7	25	211	21	2	0	10	9	205
4	726552	742197	735416	1.220	0.922	2.153	145.099	5340.687	909	909	0	0	531	0	335	216	0	24	3	0	13	8	70
5	656596	671889	665497	1.356	0.960	2.329	145.789	934.284	906	890	13	3	448	0	6	442	0	27	6	0	11	10	111
6	677225	685053	682011	0.707	0.446	1.156	148.884	1737.608	890	884	4	2	607	0	1	45	561	17	4	0	1	12	131
7	668793	683548	677452	1.295	0.900	2.206	145.179	630.306	750	634	112	4	736	0	0	734	2	8	0	0	0	8	102
8	706356	721972	714610	1.168	1.030	2.211	145.990	8961.146	876	864	10	2	595	0	0	98	497	11	0	0	2	9	70
9	705850	722311	714438	1.217	1.102	2.332	146.922	153.401	880	879	1	0	493	0	1	492	0	17	2	0	5	10	157
10	770820	791241	782646	1.534	1.098	2.649	148.534	485.668	907	907	0	0	337	3	0	295	39	20	0	0	15	5	77
			Avg.	**1.083**	**0.970**	**2.063**	**146.647**	**3690.557**	78.370	92.880	6.380	0.740	55.580	7.431	6.297	56.639	29.993	1.750	10.857	0.000	35.429	53.714	11.2

Table 1. Cont'd

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
20x50x100	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	368296	373780	*	*	*	1.489	79.415	4768.236	966	510	390	66	553	118	0	411	24	17	7	0	6	4	180
2	374837	382204	379159	1.153	0.803	1.965	82.198	203.000	680	203	241	236	539	0	418	98	23	14	6	0	6	2	105
3	355273	364078	359705	1.247	1.216	2.478	79.904	367.989	968	750	118	100	897	486	2	20	389	16	0	0	14	2	75
4	339502	343319	342115	0.770	0.352	1.124	79.775	105.561	820	145	260	415	315	0	1	12	302	14	6	0	1	7	32
5	415028	424669	421521	1.565	0.747	2.323	79.765	232.404	960	929	25	6	593	0	0	578	15	17	0	0	14	3	59
6	326645	329184	328430	0.547	0.230	0.777	78.683	109.627	661	177	206	278	201	7	1	189	4	7	0	0	1	6	63
7	384399	393126	390571	1.606	0.654	2.270	79.654	1624.976	647	229	28	390	960	876	0	84	0	6	0	0	3	3	43
8	388162	399476	393254	1.312	1.582	2.915	80.716	11632.327	957	936	15	6	697	0	661	14	22	18	2	0	2	14	127
9	366701	373660	371119	1.205	0.685	1.898	79.765	842.322	966	679	217	70	499	2	1	496	0	19	3	0	0	16	188
10	319599	326849	323005	1.066	1.190	2.269	77.902	227.818	947	938	2	7	466	4	83	372	7	24	1	0	14	9	136
			Avg.	**1.163**	**0.829**	**1.951**	**79.778**	**2011.426**	85.720	64.116	17.522	18.362	57.200	26.101	20.402	39.755	13.741	1.520	16.447	0.000	40.132	43.421	10.1

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
20x50x200	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	691861	712255	702324	1.512	1.414	2.948	176.824	548.969	924	921	3	0	684	0	567	11	4	15	5	0	7	3	211
2	723650	735246	728549	0.677	0.919	1.602	180.029	1198.954	621	258	14	349	170	0	2	166	2	16	1	0	8	7	80
3	723661	740535	731204	1.042	1.276	2.332	175.432	1236.778	937	925	9	3	659	0	1	138	520	10	0	0	0	10	70
4	679230	688667	685319	0.896	0.489	1.389	179.158	997.214	958	698	248	12	780	2	588	166	24	15	3	0	12	0	108
5	732135	752029	739630	1.024	1.676	2.717	182.192	237.311	935	935	0	0	546	1	423	120	2	22	1	0	0	21	396
6	716439	728329	723370	0.967	0.686	1.660	180.950	450.458	669	240	101	328	443	0	0	443	0	22	8	0	4	10	306
7	707426	721625	714123	0.947	1.051	2.007	174.651	176.474	908	894	12	2	594	8	187	399	0	7	1	0	1	5	200
8	683897	701055	694506	1.551	0.943	2.509	170.315	2000.357	943	744	175	24	600	0	0	600	0	14	0	0	1	13	66
9	746722	757147	751932	0.698	0.694	1.396	182.702	108.185	804	127	329	348	733	12	4	717	0	16	0	0	0	16	50
10	730976	755346	*	*	*	3.334	174.071	11084.458	932	918	12	2	723	0	554	2	167	12	4	0	4	4	206
			Avg.	**1.035**	**1.016**	**2.189**	**177.632**	**1803.916**	86.310	77.164	10.462	12.374	59.320	0.388	39.211	46.561	12.121	1.490	15.436	0.000	24.832	59.732	16.9

Table 1. Cont'd

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
20x50x500	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	1881593	1917997	1910101	1.515	0.413	1.935	754.606	956.205	609	242	320	47	225	9	0	216	0	26	6	0	13	7	155
2	1789634	1838151	1816795	1.518	1.175	2.711	746.473	5977.575	790	790	0	0	397	34	45	94	224	33	9	0	5	19	246
3	1651412	1696516	1683181	1.924	0.792	2.731	764.189	4975.828	794	692	91	11	519	0	13	421	85	13	3	0	7	3	248
4	1896879	1923861	1918510	1.140	0.279	1.422	779.491	1657.463	127	122	4	1	574	557	1	16	0	25	3	0	13	9	872
5	1730832	1763404	1756706	1.495	0.381	1.882	763.127	4499.019	782	82	336	364	581	33	0	251	297	25	1	0	13	11	317
6	1675910	1721443	1708003	1.915	0.787	2.717	762.006	6693.314	614	609	4	1	256	0	254	2	0	18	4	0	13	1	100
7	1701046	1738148	1736033	2.057	0.122	2.181	753.434	1790.365	621	348	196	77	828	4	784	9	31	23	2	0	0	21	190
8	1913583	1961301	1951089	1.960	0.523	2.494	742.958	8689.776	795	769	20	6	775	501	11	2	261	22	0	0	16	6	289
9	1648627	1681128	1674340	1.560	0.405	1.971	771.630	5089.018	343	337	5	1	2	0	0	2	0	31	4	0	4	23	207
10	1836113	1880703	1868573	1.768	0.649	2.428	761.315	2789.021	670	632	36	2	389	0	195	192	2	20	1	0	10	9	454
			Avg.	**1.685**	**0.553**	**2.247**	**759.923**	**4311.758**	61.450	75.232	16.469	8.299	45.460	25.033	28.663	26.507	19.798	2.360	13.983	0.000	39.831	46.186	30.8

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
30x50x100	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	352072	357078	355106	0.862	0.555	1.422	85.393	248.160	683	509	133	41	934	666	2	261	5	13	0	0	9	4	43
2	349580	353657	*	*	*	1.166	87.496	8396.643	783	233	312	238	414	12	1	393	8	17	1	0	2	14	97
3	331969	336752	*	*	*	1.441	93.034	10067.095	729	223	114	392	653	67	203	376	7	13	1	0	3	9	157
4	389315	394663	390546	0.316	1.054	1.374	83.710	638.187	734	645	72	17	283	5	1	252	25	16	3	0	4	9	70
5	345095	350011	346916	0.528	0.892	1.425	90.580	2462.802	954	928	26	0	696	135	73	472	16	17	2	0	4	11	89
6	328734	336042	331321	0.787	1.425	2.223	89.328	434.389	638	244	206	188	254	0	2	240	12	11	4	0	0	7	109
7	323673	328803	326421	0.849	0.730	1.585	89.269	498.256	665	209	153	303	579	0	0	579	0	19	4	0	9	6	136
8	351281	355631	352971	0.481	0.754	1.238	89.118	2511.021	962	442	375	145	742	159	243	322	18	18	1	0	3	14	166
9	338479	342483	340650	0.641	0.538	1.183	88.107	381.629	467	467	0	0	518	7	408	51	52	8	0	0	3	5	297
10	317591	322393	319025	0.452	1.056	1.512	88.527	347.84	963	725	150	88	806	41	303	295	167	11	2	0	3	6	50
			Avg.	**0.615**	**0.876**	**1.457**	**88.456**	**2598.602**	75.780	61.032	20.335	18.633	58.790	18.575	21.024	55.128	5.273	1.430	12.587	0.000	27.972	59.441	12.1

Table 1. Cont'd

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
30x50x200	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	701799	711323	705834	0.575	0.778	1.357	183.635	328.132	877	225	536	116	543	0	7	535	1	21	0	0	11	10	83
2	718772	732357	723272	0.626	1.256	1.890	182.673	930.929	867	859	7	1	802	9	703	78	12	23	0	0	13	10	101
3	651420	663453	658387	1.069	0.769	1.847	187.109	422.397	779	215	314	250	559	2	0	553	4	16	2	0	8	6	427
4	692463	706495	697763	0.765	1.251	2.026	183.604	3357.948	896	881	8	7	647	293	2	344	8	28	0	0	2	26	142
5	610314	620369	614762	0.729	0.912	1.647	184.936	525.625	922	179	534	209	773	61	119	592	1	15	2	0	6	7	113
6	666090	676862	672004	0.888	0.723	1.617	187.379	24292.291	884	172	588	124	764	18	1	745	0	14	5	0	6	3	61
7	664588	670766	668503	0.589	0.339	0.930	184.976	4496.256	889	775	95	19	635	1	415	197	22	15	3	0	4	8	213
8	679714	689653	*	*	*	1.462	193.318	8971.541	713	221	175	317	483	42	2	66	373	21	1	0	3	17	252
9	692794	701846	696495	0.534	0.768	1.307	191.285	704.242	752	159	220	373	483	33	79	28	343	19	3	0	6	10	167
10	671744	681530	676481	0.705	0.746	1.457	178.847	14874.308	928	918	6	4	485	2	15	444	24	17	1	0	8	8	94
			Avg.	**0.720**	**0.838**	**1.554**	**185.776**	**5890.367**	85.070	54.120	29.188	16.692	61.740	7.467	21.753	58.017	12.763	1.890	8.995	0.000	35.450	55.556	16.5

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
30x50x500	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	1646248	1690139	1676521	1.839	0.812	2.666	785.620	3871.337	290	290	0	0	117	1	1	3	112	21	1	0	11	9	342
2	1665775	1695004	1686297	1.232	0.516	1.755	798.157	3583.393	425	419	6	0	147	1	0	2	144	17	1	0	10	6	321
3	1582597	1610083	1600013	1.100	0.629	1.737	779.351	4641.781	483	476	6	1	164	1	6	157	0	22	0	0	16	6	348
4	1642884	1681677	1659890	1.035	1.313	2.361	773.692	1259.581	761	60	242	459	541	2	3	478	58	19	2	0	2	15	231
5	1626013	1661750	1642141	0.992	1.194	2.198	790.968	3639.573	463	460	3	0	348	75	0	271	2	21	1	0	1	19	526
6	1673438	1721414	1698162	1.477	1.369	2.867	792.269	9872.837	411	250	126	35	321	0	221	67	33	27	3	0	15	9	606
7	1575753	1611070	1599390	1.500	0.730	2.241	797.347	1560.444	234	89	109	36	206	0	204	0	2	14	0	0	0	14	480
8	1676694	1715771	*	*	*	2.331	781.764	14479.180	485	479	5	1	408	280	0	128	0	26	1	0	5	20	549
9	1621733	1658057	1639772	1.112	1.115	2.240	795.574	4831.487	527	521	4	2	312	0	178	134	0	20	2	0	12	6	474
10	1613353	1647814	1635846	1.394	0.732	2.136	799.770	17703.387	525	277	217	31	317	0	15	297	5	14	1	0	6	7	476
			Avg.	**1.298**	**0.934**	**2.253**	**789.451**	**6544.300**	46.040	72.133	15.595	12.272	28.810	12.496	21.798	53.350	12.357	2.010	5.970	0.000	38.806	55.224	43.5

Table 1. Cont'd

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
30x50x500x5	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	1386955	1424904	*	*	*	2.736	835.791	17501.065	316	96	103	117	228	18	1	206	3	59	6	0	27	26	830
2	1600981	1634265	*	*	*	2.079	831.146	31399.460	125	124	1	0	3	0	0	3	0	40	5	0	24	11	913
3	1334347	1352931	*	*	*	1.393	825.367	23870.283	279	160	101	18	16	0	14	2	0	34	1	0	19	14	704
4	1391358	1405869	*	*	*	1.043	845.697	14583.740	272	242	22	8	97	4	1	55	37	43	2	0	19	22	971
5	1366139	1399835	*	*	*	2.467	823.674	14583.740	125	120	1	4	88	3	64	21	0	36	0	0	28	8	1294
6	1361711	1384250	*	*	*	1.655	842.872	17924.714	298	277	21	0	242	3	219	19	1	57	5	0	29	23	885
7	1479648	1504417	*	*	*	1.674	863.552	22550,196	181	71	24	86	197	5	0	4	188	35	6	0	17	12	1029
8	1443036	1471008	*	*	*	1.938	852.656	17057,927	211	187	18	6	191	0	1	190	0	100	7	0	28	65	1093
9	1347818	1371996	*	*	*	1.794	834.220	24601,265	445	441	3	1	357	1	0	355	1	68	16	0	23	29	728
10	1489409	1523025	*	*	*	2.257	851.073	30686,686	177	74	39	64	20	2	1	6	11	130	35	0	28	67	895
			Avg.	**N/A**	**N/A**	**1.904**	**840.605**	**21920,083**	24.290	73.775	13.709	12.515	14.390	2.502	20.917	59.833	16.748	6.020	13.787	0.000	40.199	46.013	93.4

	Value of						Solution Duration			WOH	WOH	WOH		POH	POH	POH	POH		PCH	PCH	PCH	PCH	
30x50x500x10	**LB**	**UB**	**CPLEX**	**LB Gap %**	**UB Gap %**	**Gap %**	**LR (s)**	**CPLEX (s)**	**WOH**	**1**	**2**	**3**	**POH**	**1**	**2**	**3**	**4**	**PCH**	**1**	**2**	**3**	**4**	**LFC**
1	1339993	1358114	*	*	*	1.352	929.096	17669.948	163	48	14	101	105	12	19	5	69	541	53	0	102	296	1361
2	1279156	1262724	*	*	*	1.301	933.532	12190.890	27	20	1	6	79	28	32	5	14	778	151	0	178	449	1672
3	1353831	1374393	*	*	*	1.519	944.538	17857.318	72	52	15	5	44	12	0	27	5	683	227	0	202	254	1472
4	1279623	1290728	†	†	†	0.868	925.751	36000.000	214	189	20	5	281	8	218	8	47	602	29	0	47	526	1299
5	1300891	1316591	*	*	*	1.207	946.622	27480.722	109	54	23	32	133	12	117	4	0	660	23	0	232	405	1541
6	1242211	1270748	*	*	*	2.297	944.078	19185.657	121	119	1	1	64	18	9	7	30	587	13	0	328	246	1398
7	1232683	1251601	1240310	0.619	0.910	1.535	944.498	32535.482	227	88	42	97	166	7	0	158	1	396	156	0	86	154	1418
8	1292099	1308883	*	*	*	1.299	939.732	10753.543	158	157	1	0	134	129	1	4	0	575	168	0	86	321	1445
9	1318521	1341964	1328136	0.729	1.041	1.778	936.998	13714.211	189	75	62	52	140	11	9	76	44	592	30	0	503	59	1349
10	1211329	1232049	*	*	*	1.711	932.161	38899.472	79	72	5	2	118	61	29	19	9	497	44	0	211	242	1556
			Avg.	**0.674**	**0.976**	**1.487**	**937.701**	**22628.724**	13.590	64.312	13.539	22.149	12.640	23.576	34.335	24.763	17.326	59.110	15.124	0.000	33.412	49.941	145.1