A Lagrangian Heuristic for a Variant of Capacitated Facility Location with Single Source Constraints

A THESIS
SUBMITTED TO THE DEPARTMENT OF INDUSTRIAL ENGINEERING
AND THE INSTITUTE OF ENGINEERING AND SCIENCES
OF BILKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

by
Yusuf Ziya Ayrım
September 2006

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

Assoc. Prof. Osman Oğuz (Principal Advisor)

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

Prof. Mustafa Ç. Pınar

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

Asst. Prof. Bahar Y. Kara

Approved for the Institute of Engineering and Sciences:

Prof. Mehmet Baray

Director of Institute of Engineering and Sciences

ABSTRACT

A Lagrangian Heuristic for a Variant of Capacitated Facility Location with Single Source Constraints

Yusuf Ziya Ayrım
M.S. in Industrial Engineering
Supervisor: Assoc. Prof. Osman Oğuz
September 2006

Facility location problems (FLP) are extensively studied in the literature in the context of supply chain management. Wide variety of real life situations are analyzed and modeled using techniques developed for FLP. In this thesis we take a comparably new model, Capacitated Facility Location with Single Source constraints (CFLPSS) from the literature and add an additional feature of Minimum Supply (MM) requirements (CFLPSSMM). Then we devise a Lagrangian Heuristic, which is highly efficient for CFLPSS models and for this new variant of CFLPSS. This heuristic, which is modified from the heuristics devised for CFLPSS, is then tested both on data from the literature and on new data set. Results indicate that it can be a resourceful alternative; especially the lower bounds provided by the heuristic are quite effective both for CFLPSS and CFLPSSMM.

Keywords: CFLP, Lagrangian Heuristic

ÖZET

Tek Kaynak Kısıtlı Kapasiteli Bina Yeri Seçimi Varyasyonu için bir Lagranj Sezgisel Programlaması

Yusuf Ziya Ayrım
M.S. Endüstri Mühendisliği
Süpervizor: Assoc. Prof. Osman Oğuz
Eylül 2006

Bina yer seçimi problemleri literatürde sıkça tedarik zinciri işletimi bağlamında işlenmiştir. Gerçek yaşamda karşılaşılan pek çok durum BYSP için geliştirilmiş tekniklerle analiz edilip, modellenmektedir. Bu tezde göreceli olarak yeni olan bir modeli, Tek Kaynak Sınırlı Kapasiteli Bina Yer Seçimi Problemini (TKKBYSP) ele alarak, bu modele yeni bir özellik olan En Az Tedarik (ET) kısıtını eklemekteyiz (TKKBYSFET). Daha sonra TKKBYSP ve bu varyasyonu için yüksek etkinlikte bir Lagranj Sezgisel Programı geliştirilmiştir. TKKBYSP için geliştirilmiş olan sezgisel programlardan uyarladığımız bu sezgisel programı, hem literatürdeki bir bilgi kümesi üzerinde hem de yeni bir bilgi kümesi üzerinde denemiş bulunmaktayız. Sonuçlar bu programın uygun bir alternatif olabileceğini işaret etmekte, özellikle TKKBYSF ve TKKBYSFET için çok etkin alt sınırlar verebileceğini göstermektedir.

Anahtar Kelimeler: KBYSP, Lagranj Sezgisel Programlama

To my family and my brother

That which does not kill us makes us stronger

ACKOWLEDGEMENT

I would like to express my sincere gratitude to Assoc. Prof. Osman Oğuz for all his attention, and support during my graduate study and, of course, for his valuable guidance and most importantly for his patience and trust.

I am indebted to members of my dissertation committee: Prof. Mustafa Ç. Pınar, Asst. Prof. Bahar Y. Kara for showing the kindness to accept to read and review this thesis. I am grateful to them for their effort, sparing their valuable time for me and for their support.

I would like also to thank my friends Muzo and Ayşegül for their guidance, Safa and Sibel for providing me the thesis format, Utku for his support in C coding, Paç and N. Barış for the morale boost and assistance. Most important of all I am grateful to all of my friends for the support and friendship they showed to me.

I am also indebted to Prof. Nesim Erkip, for his valuable comments on the subject and all the other IE faculty members.

Finally, i would like to express my deepest gratitude on my parents Serhat and Alev Ayrım and my brother Sami Murat Ayrım. Without their love and support I would never have finished the thesis. I feel very lucky for having such a wonderful brother and family. Special credits go to my grandmother for catering sponsorship…

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

Efficiency is an important concept for doing business in the competitive markets of modern times, and its importance increases with the globalization of these markets. The key to the success is making correct choices and companies strive to choose the best option that maximize their profits or minimize their costs. As establishing facilities is a major cost component and companies incur billions of dollars to establish new facilities each year, it is very crucial for companies to make good choices in this area. Facility location decisions are probably one of the most important determinants for success or failure of the related business in the long run.

Facility location decision is an issue companies face in many areas. As its name implies the most common usage is for plant, warehouse or distribution channel. However it is not limited to these aspects only; it is a problem that may be encountered in most sectors including telecommunications, location of emergency services etc. Facility in its broadest sense may be any thing that must be built or established to supply the markets or to serve a need.

Given a set of facility locations and a set of customers who are supposed to be served by these facilities; the general facility location problem is to determine which facilities should be open and which customers should be served from which facilities so as to minimize total cost for the company

(Selçuk[23]). By opening facilities we incur some fixed costs to open/operate these facilities. Only after opening facilities, we can assign the retailers to the opened facilities to serve the demands. A second cost is incurred when we assign these retailers/demand points to facilities. This is the cost to satisfy the demand of customers from the facilities or assignment costs; which may cover production and transportation costs.

Facility location problem (FLP), in its most simple form, is a balancing decision between these fixed costs and assignment costs. By increasing the number of facilities you will decrease the assignment costs as the more facilities opened, the distance between facility and demand points will decrease and so are the assignment costs. However, opening more facilities will increase the fixed costs. So there is a tradeoff between fixed costs and transportation costs. In more complex models there are also other costs included beside these two main cost components, such as the inventory costs.

The basic FLP is known as Uncapacitated Facility Location Problem (UFLP) or Simple Plant Location Problem. It only consists of fixed and transportation costs. Facilities in this model have no capacity limits, as its name implies. Although, it is an NP-hard problem as it will be discussed later, it is a relatively easy problem to solve due to its tight LP relaxations.

Capacitated Facility Location problem (CFLP) is the problem obtained when we add capacity constraints to UFLP. In CFLP, facilities can supply only a limited amount of demand. Although it seems like a simple modification, it makes the problem much more difficult to solve. This is due to the fact that in CFLP the demand of a single retailer can be divided across multiple facilities and this destroys the tightness of LP relaxations of the problem. As it will be discussed later, there are numerous research results in this area.

A less investigated area in FLP is Capacitated Facility Location Problems with Single Source constraints (CFLPSS). The only difference between its predecessor (CFLP), is the single source constraints, meaning demands of customers can not be divided across facilities and each must be served by exactly one facility. It is an extension of CFLP and as CFLP, it is NP-Hard. Unlike previous two problems (UFLP, CFLP), there is not much work out in literature about CFLPSS. It is also the main problem we based our research upon. Another feature of CFLPSS is that its branch and bound tree gets too large too quickly compared to UFLP and CFLP. It is a complete integer problem that requires too much memory and time to be solved by direct approaches like a straight forward branch and bound (B&B).

So as in other similar problems, it may be a good idea to try to achieve a meaningful solution in an acceptable time, rather then to strive for optimum at the cost of an excessive computational time. The main motivation behind our thesis is this main fact mentioned: Devising an efficient heuristic that would give a close to optimum solution. This solution can either be used as a decision tool to go for an optimum solution, or providing an applicable solution between acceptable limits.

Lagrangian Heuristics (LH) are favored by many researchers in the literature as a tool of obtaining effective solutions for facility location problems, especially for CFLPSS and CFLP. This is due to several facts:

- They work fairly quickly
- They provide generally better lower bounds than LP relaxations (will be discussed later in detail)
- They can be embedded into B&B methods to obtain optimal

solutions

These characteristics mentioned above make LH’s good candidates for facility location problems.

In this study, we consider a CFLPSS with an additional feature. This feature is based upon the assumption that some of the facilities in the CFLPSS model are not standard production facilities, instead they represent subcontractor firms. It is not uncommon for subcontracting firms imposing restrictions on the amount of minimum supply. This assumption results in another set of constraints, namely “minimum supply” constraints in addition to the basic CFLPSS constraints. These constraints will be same as capacity constraints, but rather being upper bounding, they will provide lower bounds on the amount of supply from facilities.

Then we devise a LH based upon prior work of Holmberg et al. [21] and Sridharan [26] on CFLPSS. Their heuristics are modified to be able to solve this new variant. Another approach to solve this problem would be by using a commercial software like Xpress-MP, GAMS, Lindo or CPLEX. We preferred state of the art software CPLEX, because of its robust and efficient framework. Also CPLEX includes a MIP (mixed integer programming) module including preprocessing and aggregation to decrease problem size. Moreover CPLEX uses a Branch and Cut (B&C) approach to aid solving MIP’s which can generate several general cut classes including cover, Gomory, clique, flow and etc. These specifications of CPLEX are the reasons of its being considered as one of the best of general case MIP solvers. Another reason for choice of CPLEX is its high compatibility with C/C++ programming language (as it’s also written in this language), in which our heuristic and the CPLEX caller is also coded.

We applied both approaches (LH and CPLEX) on 2 data sets. One of them is also the data set that Holmberg et al [21] used in his CFLPSS computational experiments. The other is generated according to Holmberg's [21] distributions for demand, capacity etc. Even though CPLEX gave considerably good results as expected, our LH is shown to be an efficient alternative to be considered solving this CFLPSS variant (CFLPSSMM, Capacitated Facility Location with Single Source and Minimum Supply Constraints) both with respect to solution quality and computational time. It is also shown that, the lower bounds of this heuristic are quite good, that may even give dual optimum for small cases and quite tight solutions for larger cases. It is noteworthy to mention that, in large cases the lower bounds of LH, are better than CPLEX, even with the large processing times of CPLEX.

The remainder of this thesis can be outlined as follows. In chapter 2, we review the prior work in the area of FLP, especially in the context of CFLP and CFLPSS. We also include a brief classification of FLPs and their solution approaches in this part. In the following chapter we outline our model. Chapter 4 is about the structure of our Lagrangian Heuristic. Then in chapter 5 we continue with computational results of our LH applied to CFLPSS and CFLPSSMM. Finally in 6th chapter conclusions and remarks are discussed.

Chapter 2

CFLP/CFLPSS STRUCTURE AND SOLUTION PROCEDURES

Supply chain management is "the management of the entire value-added chain, from the supplier to manufacturer right through to the retailer and the final customer." (x-solutions, [35]). As it constitute %10 of gross domestic product of USA (Daskin [14]), it is a major cost component and an area covering essential decisions for companies. These essential decisions are composed of wide range of interrelated areas; including supply contracts, information sharing and distribution networks design. Facility location, which is the main subject of this thesis, is a sub branch of supply chain management that covers the core topics of distribution system design.

Annually, 500 billion dollars are spent on establishing new facilities in USA (Selçuk [23]). Moreover, unlike most other supply chain decisions, it is quite hard if not impossible to reverse these decisions especially in short term. In medium horizon, it may be possible to change facility location decisions, but generally only at the expense of a vast amount of money. Only in strategic horizon (long term), it may be profitable to change the facility location (FL) decisions. These facts make FL decisions quite crucial for companies and extra precaution must be taken into consideration as it may be quite hard to change these later.

The term “facility” refers to an object that supplies goods or services to satisfy the demand. That is why we prefer the “facility location” name over other synonyms for this set of problems. “Plant location” imposes the production plants like manufacturing, assembly or energy. Another common usage is “warehouse location”, which focus on the warehouses and distribution centers (DCs) as facilities. On the contrary, “facility” is a very broad definition and it covers all the aspects of possible sectors, like telecommunication and emergency. In electronic networks, a router is the object you establish to satisfy the customer demand, which in this case is packets of information transmitted. In emergency sector facilities are the hospitals, fire stations and police stations. Obviously, these sets of problems are no different than “plant” or “warehouse” location in terms of model structure and solution procedures. Therefore, we will use the more general “facility location problem” or “FLP” from now on to address these set of problems.

Facility location mainly consists of two important strategic decisions for companies: locating facilities and allocation of goods or services to these facilities. Location part of the problem involves opening enough facilities to supply the demand. Allocation part is about assigning customers to the opened facilities in order to satisfy their demand. As these two main problems are interrelated, there must be a combined solution procedure that optimizes both sub problems together.

As FLP includes a wide range of problems, we start by giving some brief definition and classification of FLPs. Then we review some of the mainstream work in the literature on FLP, which will help the reader to understand the basics and motivation behind this thesis.

2.1 Definition and Classification of FLP's

Location theory is an extensively studied topic in the literature and it dates back to 1900's. It found its first formal introduction by Weber in 1909. Numerous applications of FLPs were already researched in these years, but it was not until mid of 19602s that these applications were tied together by a unified theory (Brandeu[6]).

Since mid 1960's, wide range of models and applications have been developed under context of location theory. Location theory is a vast field and our main focus will be on a much smaller branch of FLPs, family of CFLP/CFLPSS. To ease further reading and understanding we include some basic terminology and try to make a brief classification of FLP family.

Although there is no common classification of FLPs, it is easy to obtain one based on the type of objectives, cost terms and on the constraint sets.

There are several main objectives in FLPs in general case:

- Determining optimal number of facilities (A)
- Determining optimal location of facilities (B)
- Allocation of demands to these facilities optimally (C)
- Optimal inventory policies (where to stock, how many to stock) (D)
- Optimal vehicle routing considerations (E)
- Optimal network design (F)

(A) and (B) together constitute the location sub-problem of FLP. (C) is referred as allocation sub-problem. These first three parts are found in most of FLPs. On the other hand (D) and (E) are included only in some other more complex integrated models. Addition of (D) turns the standard FLP into a FLP with integrated inventory location and addition of (E) gives us a FLP with vehicle routing considerations. Finally addition of (F) brings out a FLP with network design. Obviously; (D),(E) and (F) are optional considerations and they increase the complexity of the problem.

Erlebacher [15] provides an integral approach to FLP. He illustrates the impact of FL decisions on inventory costs like holding costs and risk pooling effects. Selçuk [23] illustrates the cases where vehicle routing ought to be taken into consideration. He points out that, if LTL (less than truck load) systems are considered; integrating vehicle routing considerations are beneficial for creating a more realistic model. And Daskin et al.([12] and [13]) marks the benefits of considering a FL/network design integrated approach. They point out that, for LTL distribution systems, pipeline systems, telecommunications systems etc "it may be more economical to change the configuration of the underlying network instead of locating new facilities"(pg 481, Daskin [13]). They formulate the FLP in a network design model to incorporate this structure to their model.

Objectives of the models are interrelated with the cost structure of FLPs in general. There are several main cost structures in FLP models:

- Fixed costs (i)
- Production, transportation, assignment etc costs (ii)
- Inventory costs (iii)

- Arc (link) building costs (iv)

First of all, cost terms (i) are the fixed costs of opening and operating facilities. It is the cost of opening facility, and it is incurred in full, even if facility is open just to satisfy a single demand. Secondly, we see cost terms (ii), which are a main cost component of general case FLPs. Term (ii) includes costs of assigning/transporting a demand point (retailer etc) to/from a facility. Cost term (iii) is taken into account in joint inventory/location models as in Erlebacher's[15]. Lastly, cost term (iv) is found in network design-FL integrated approaches.

Other than these basic classifications, more extensive classifications can be found in Francis et al [18] where they classify FL under facility location and layout problems according to a 6 element criteria (pg 20). A vast amount of classifications exist in literature besides those mentioned above. One of the newest and a quite extensive one is those of Klose et al.'s ([30], pg. 5).

Klose [30] classifies FLP according to 9 aspects of the problem and model structure (examples are provided to briefly illustrate):

- Shape or topography of the facility, demand sets (Network, Planar, Discrete)
- Objectives (minisum, minmax)
- Capacity restrictions (Uncapacitated, Capacitated)
- Number of stages (Single, Multi)
- Number of commodities/products (Single, Multi)

- Demand relation with other decisions (Spatial, Correlated)
- Static versus Dynamic models (single period plan horizon, multi period plan horizon)
- Demand certainty (Deterministic, Stochastic)
- Quality of demand allocation and aggregation of demand

Obviously, the combination of these aspects results in many different models. These models generally try to capture different aspects of particular real life situations and complexities. However, in general such complexities may not exist or may be negligible due to the structure of problem. For instance in the models which demands are always multiples of TL (truck loads), we can neglect vehicle routing considerations as they may be only significant for LTL systems. Also, if demands are deterministic inventory costs, will not have a significant impact on facility location decisions, so they can be dropped out of the model. Moreover, for a simple distribution network design; using network topology may result in extravagant use of computational time.

The simple approaches can still be useful even if some complexities exist for a particular situation. They can still be used as decisions tools or approximations for complex integrated approaches. It is a well known fact that as the complexity increases, the probability of errors and miscalculations increases. Therefore, in most cases using simple approaches that capture essential structure of the problem is better than to devise a complex error prone model.

So we restrict our attention to a basic approach in FLPs, without optional objective and cost structures. We will only consider the general case of including (A),(B),(C) as its objectives and (i),(ii) as its cost structure. In other words, the objective is to find a tradeoff between fixed and assignment costs that will minimize the total cost. Solution of this problem will result in an optimal location of facilities and allocation of the customers to these facilities. This problem is referred as General Warehouse Location Problem (GWLP) by Beasley [5] and it is the predecessor of 4 well known FLP problems: UFLP (uncapacitated), CFLP (capacitated), CFLPSS (capacitated with single sourcing constraints) and p-median).

These are 4 sets of problems that can be derived from GWLP. As CFLP/CFLPSS of these 4 are the ancestors of our problem, review of the literature in this context may clarify our model and solution approach.

For further information on location theory reader is referred to Klose [30] and Brandeu[6] in which about 50 different problems are classified in location theory. There also can be found a selective bibliography of location theory between 60s and 70s in Francis[17].

2.2 About the Structure of CFLP/CFLPSS

Structure of CFLP/CFLPSS will be further discussed in the next chapters, however to be able to clarify things better we include some basic structure and notation concepts of CFLP/CFLPSS. As we repeat this name a lot in successive chapters, from now on we will refer to these family of problems as “problems” as a short hand notation, unless noted otherwise.

There are several important features of these problems. First of all these problems are based upon a “minisum” objective function, which is composed of a summation of fixed costs and transportation costs. There are 2 decision variables, opened facilities (y_j) and assignment of demands (x_{ij}), in these problems. The basic form of problems includes only 2 constraint sets (except integrality and binary requirements). First one is demand constraints that ensures all the demand is served ($\Sigma_i x_{ij} = 1$). Second one is capacity constraints that ensure capacity of the facilities do not get exceeded ($\Sigma_i d_i x_{ij} \leq s_j y_j$).

In addition to these basic constraint sets, 2 supplementary constraint sets are also included in some of the related literature. "Surrogate constraint" is an optional constraint that ensures total capacity of opened plants exceeds the total demand ($\Sigma_i d_i < \Sigma_j s_j$). This constraint primarily used for improving lower bounds obtained from lagrangian relaxations. Other optional constraint set is $x_{ij} \leq y_j$. Problems are referred as in "strong form", when $x_{ij} \leq y_j$ constraints are included and "weak form" otherwise. This constraint set strengthens especially the LP relaxation of the problems.

As there are 2 complicating sets of constraints, either demand constraints or capacity constraints are relaxed in lagrangian solution procedures in the literature (there are other relaxations too but not widely used, for detailed info please refer to Cornuejols [8]). We will adopt Cornuejols [8] et al's notation and use subscript in the place of a lagrangian relaxation and superscript in the place of a complete relaxation. Meaning Z_D and Z_C stand for demand constraint and capacity constraint lagrangian relaxation. Z^T stands for problem without surrogate constraint (total demand constraint).

The structure of problems is an interesting one that is interrelated with many discrete optimization problems. Z_C relaxation will result in a UFLP sub problem that can easily be solved by methods like Erlenkotter [16]. Z_D relaxation will reduce into knapsack problems for CFLP and a trivial problem. For CFLPSS same relaxation will result in 0-1 knapsacks as sub problems and a continuous knapsack. Moreover, these problems are closely related with problems such as transportation, general assignment (GAP) or matching. These structures are highly exploited in a lot of work in the literature (like Baker[4] exploiting TP and Holmberg [21] using rapid matching).

2.3 Solution Procedures

Facility location problems are defined and briefly classified in previous part of the section. As mentioned previously our work is built upon a small branch of FLPs, which are CFLP and CFLPSS. Moreover, our solution procedure is an extension of those in CFLP and CFLPSS. So we review the work in this area in this section.

Uncapacitated Facility Location Problem (UFLP) or "simple plant location problem" is a basic problem, which represents the foundation on which other FLP are based. It was formally formulated by Balinski in 1965 (see Harkness[19]). In this problem there is a set of candidate sites for facility locations and demand points. Unlike facility locations, demand points are not decision variables and they are fixed. Moreover, there exist two kinds of costs in the formulation: one fixed and other transportation costs. The objective is minimizing the sum of all costs (minsum objective), while serving all demand points. To serve demand points one must first incur fixed costs to open facilities to serve the demand, then by incurring transportation costs he must transport the products from facilities to demand points. Facilities in this problem have no capacity limits, under assumption that they can supply any amount of demand. Although this problem is NP-Hard, as its structure is "integer friendly", one can easily solve by LP relaxations with little or no resort to branch and bound (B&B). In the literature there are several efficient models to solve UFLPs. One of them is the well known Erlenkotter's [16] dual-based procedure.

2.3.1 Solution Procedures of CFLP

CFLP (capacitated facility location problem) is a problem derived from the basic UFLP. In this case, facilities can supply no more than a defined capacity limit. Unlike its predecessor, it is a much harder problem to solve. One of the earliest researches is work of Sa [22], in which he investigates B&B and approximate solutions to CFLP.

One of the mile stone papers in this area is those of Akinc&Khumawala [2]. They devised an efficient algorithm for CFLP based on a B&B algorithm, employing powerful penalty tests, that are used to fix facilities open (inclusion test) or closed. By this way problem size and complexity is decreased. These penalty tests also inspired many other researchers in the area (as they are cheap in computational time to employ). They also devised good UB and LB algorithms. Finally by establishing good node and branch selection criteria they formed an efficient B&B algorithm. They also tested these problems on Kuehn& Hamburger's [28] data set. This data set is also widely used in FLP literature and it is ranging from 20v10 (10 facility, 20 demand point) to 50v25 in its largest case.

Nauss [33] in 1978 proposed an "improved algorithm" for CFLP. He used the lagrangian relaxation, which is devised by Geoffrion to compute lower bounds. This relaxation in fact is the relaxation of demand constraints in a lagrangian fashion. He observed that lagrangian relaxation may give stronger lower bounds than LP relaxation by adding a constraint set, which is also known as “surrogate constraints” in the literature. (These constraints improve the quality of lower bounds and are used in many such relaxations. They are also used in the relaxation part of our model.) As a result, based on his computational results he claimed that his algorithm is at least as effective as Akinc & Khumawala’s [2] in worst case and can be up to 3 times effective in others.

Another well known algorithm is Van Roy’s [35] cross decomposition algorithm. In his paper he unified “Bender’s decomposition and lagrangian relaxation into a single framework” (pg. 145). He preferred the strong formulation of the problem. His work is based on the observation that for a fixed set of facilities, location problem turns into a transportation problem. Then he fixes facility locations to turn the problem into a transportation problem (TP). Subsequently, solution of TP is used to generate lagrangian multipliers. Then lagrangian relaxation is used to get the next fixed facility locations. This primal structure is then embedded into a decomposition scheme to ensure progress towards optimum. He also tested his procedure in Kuehn & Hamburger data set. His results are found out to be 10 times faster than other existing algorithms. His algorithm works very fast for small duality gap problems.

Sridharan’s[27] work is a good review of solution methodology in CFLP literature, in which he also contributed with works like [25] and [26]. First of all, he formulates the CFLP. In addition to standard formulation with demand and capacity constraints; he also included “surrogate constraints” in his formulation and preferred strong formulation. As he refers to Geoffrion & McBrides work in 1978([25], pg. 307), the lagrangian relaxation of CFLP without surrogate constraints is only as strong as LP relaxation with strong formulation. Then he reviews the common solution methodology for the problem: greedy heuristics (ADD&DROP), interchange heuristics, lagrangian heuristics, lagrangian or LP relaxations embedded in B&B, dual ascent method (extended from Erlenkotter's [16]), Benders decomposition and Cross Decomposition (as in Van Roy [36]).

Cornuejols [8] et al provided another useful review. Unlike Sridharan[27], they did not review the solutions but reviewed the bounds. They mainly compared strengths of different relaxations used throughout the literature. They proved many relations between qualities of the bounds. Most importantly, they proved that lagrangian relaxation of capacity constraints yield at least as tight as lagrangian relaxation of demand constraints ($Z_C \geq Z_D$). Moreover, they found out that "variable splitting does not yield stronger bounds than best lagrangian relaxation" (pg. 282). Then they computed the quality of several main relaxations on a problem set. It is found at that Z_C and Z_D with surrogate constraints provide tight (at most %1 -%3 respectively) lower bounds. Z_C is better as expected but at the cost of the computational complexity.

There are also several important works that is worth to mention in the area. Baker [3] provides a generalized constraint that can be used to create efficient valid inequalities. Unfortunately, the strength of the cut generated by this inequality largely depends on the parameters of the inequality and they could not provide a way to obtaining good parameters. Mateus [32] investigates the relation between fixed and assignment costs in a weak formulation of CFLP. This observation leads to a solution procedure involving exact tests and greedy heuristics (ADD&DROP heuristic). In addition to these many different methods are devised to solve CFLPs, like branch&price algorithm of Klose et al [31] and partial dual algorithm of Baker[4].

Some of the work in the field, is based on simple variations of CFLP. Shulman [24] investigates the CFLP with dynamic expansion sizes and describes a solution procedure based on a lagrangian technique. Sridharan [25] investigates the CFLP with side constraints. He bounds the number of open facilities from above and below. He then employs a lagrangian relaxation based heuristic to solve this variation of CFLP.

2.3.2 Solution Procedures of CFLPSS

Standard CFLP is same as CFLPSS, except for its single source constraints. These constraints force x_{ij} values to take binary variables (0-1). Obviously problem structure remains unaltered, however the number of integer variables increases greatly. For example in a 200v30 CFLP there exist only 30 integer variables (y_j), an easy target for direct brute force approaches (even a simple B&B will suffice most of the time). However with the addition of SS, number of integer variables will become 6030! (30 y_j + 6000 x_{ij}) This will greatly increase the problem size, memory requirements and computational complexity. As a result, heuristics are favored instead of exact procedures most of the time. Lagrangian heuristics are found out to be efficient for this case of problems so most of the related literature embodies lagrangian approaches. Moreover, lagrangian heuristics provide a readily available lower bound at each step as Klincewicz remarked [29].

Klincewicz et al's [29] work is one of the earliest works in CFLPSS. They relaxed capacity constraints in a lagrangian fashion and did not take "surrogate constraints" into account (Z_C^T in short notation). The resulting sub-problems are solved by Erlenkotter's [16] dual based method. Then they used ADD heuristic (a greedy approach in FLP literature) to create initial solutions. Later, they tested the problem with K&H data set [27], which is at most 50v25-

50v26. Darby-Dowman and Lewis[11] investigated this same case based upon the observation that for some set of problems Z_C^T is infeasible and ADD heuristic does not guarantee to yield feasible solutions. They found that for some set of problems Z_C^T is of no value. They conclude that, although it "is tempting to relax 'hard' constraints" (they refer to capacity constraints) in lagrangian fashion, "it may be worthwhile carrying out a preliminary analysis" (pg. 1039). In other words, relaxing capacity constraints may fail in some cases.

A different approach came from Sridharan [26]. Sridharan pointed that by relaxing capacity constraints (Z_C^T), resulting sub-problem will be UFLP, which is also known to be NP-Hard. Therefore, unlike Klincewicz et al [29], he preferred relaxing demand constraints. He also incorporated "surrogate constraints" to achieve better lower bounds (Z_D). He extended the Nauss's algorithm to solve CFLPSS. He devised a lagrangian heuristic that ping pongs between a single source transportation problem and Z_D. Unfortunately, data set he tested his algorithm is quite inadequate and small (35*20 at maximum).

Beasley in his work [5] also employed a lagrangian heuristic. He formulated GWLP(general warehouse location), and upon this formulation he built a general case lagrangian heuristic that both relaxes demand and capacity constraints (Z_{CD}). Despite being an insightful work and inspiring some further work (check Agar[1]), because it is a general case framework it is inefficient for CFLPSS in particular. This algorithm is a multi purpose algorithm that can be also used to solve CFLP, UFLP and p-median. The maximum size problem of data set is also 50v25-50v26 for CFLPSS.

Hindi and Pienkosz [20] follow the same approach as Sridharan. They also used Z_D as the lagrangian relaxation. However, their algorithm differs in terms of finding feasible solutions. They combine a greedy constructive search with restricted neighbor search to find better feasible solutions. In addition, to data from the literature they tested their algorithm on larger scale problems. Corthinhal et al.[9] used the same popular approach for lower bounds: Z_D. Upper bounds are created by using search methods and tabu metaheuristic.

Correia and Captivo [7] investigate a variant of CFLPSS where multiple possible capacities exist in a discrete space. They named this problem as CFLPSS modular. They provided lower and upper bounds by the use of demand constraint relaxation in lagrangian fashion. Then they enhanced this procedure by tabu and local search. In joint work of Cortinhal and Captivo [9], upper and lower bounds are provided by lagrangian relaxation and tabu metaheuristics. . Agar and Salhi [1] proposed a lagrangian heuristic build upon framework of Beasley [5], which can be used to solve large instances of several CFLP variants including CFLPSS. C

Holmberg et al. [21] proposed an ingenious algorithm based upon lagrangian relaxation. As the most of other researchers in this field, they relaxed demand constraints (Z_D) in a lagrangian fashion as a lower bound procedure. They created feasible solutions from the output of lagrangian relaxation and then empower these results with dual-based penalty tests (like A&K[2]) and a rapid matching algorithm. They provided a comparably large sized problem set, except for those of Agar&Salhi[1] and Hindi et al [20]. The largest problem size in this set is as large as 200v30, which is considerably good for testing purposes. Moreover, they embedded their heuristic in a B&B framework; so that it can be used as an exact procedure.

2.4 Remarks

Obviously, relaxing demand constraints in a lagrangian fashion is quite popular in CFLPSS. This is mainly because its sub-problems are relatively easy to solve and lower bounds obtained by lagrangian relaxations are better than those of LP relaxations. As you may recall from CFLP solution procedures, Z_D^T (without surrogate constraints) is no better than LP relaxation. However, this is not the case for CFLPSS, it is at least as tight as LP relaxation (Sridharan [26] pg.307) even without surrogate constraints (only for a limited set of extremely restrictive problems it can be equal to LP relaxation). Moreover, the sub -problems Z_D are knapsacks which can efficiently solved by dynamic programming approach. On the contrary sub-problem Z_C is computationally hard to solve.

Note that solution procedures for CFLPSS are extension of those of UFLP and CFLP. However, unlike CFLP and UFLP there are not much different solution approaches in the field. The main approach of solution is heuristics. This is quite intuitive as CFLPSS is IP, therefore its memory and computational requirements are very high. Two main types of heuristics are employed: exchange heuristics and lagrangian heuristics.

As our problem is an extension of CFLPSS and its structure is not altered much, obviously algorithms that work well with CFLPSS will work well with this new variant. Therefore we mainly based our algorithm upon the existing literature of CFLPSS. Our algorithm is an adaptation of Holmberg et al's [21] and Sridharan's [26] brilliant works. So we preferred a lagrangian heuristic based on Z_D, to solve this new variant.

Chapter 3

MODEL

In this chapter we will introduce a problem that stems from CFLP/CFLPSS problems. In fact, this problem can be thought as a general case for CFLPSS.

This chapter is organized as follows. In first section, we define our problem and discuss the main motivation behind it. Then in section 3.2., we will formulate our model as a mathematical model and state its parameters, variables and set notations.

3.1 Problem Definition

Our model is an extension of well known CFLP with single source constraints. It has the same basic objective and cost structure, where the only difference is the addition of minimum supply requirements to its constraint set. As other FLPs, CFLPSS has a large application area and its results may be interrelated to the other supply chain decisions.

In traditional FL approach, all demands must be supplied. We build our model upon the observation that the model can also be used to cover lost sales and sub-contract cases to the model without altering the structure of the model. This will allow seeing opportunities more clearly in the supply chain. For lost sales cases we assign the state of “lost sale” to a dummy facility. Then we can assign penalty costs in same manner as transportation costs. Then even the if the total demand exceeds capacity, model will work fine reflecting which demand should not be supplied (lost sales) in its optimal solution. The same things apply for a subcontractor situation. We could replace ordering costs with fixed costs, and their fees of production and transportation as assignment costs.

In the worst case of such a situation with N*M size (N retailers, M facilities) with addition of such R subcontractors, the problem complexity will not be more than N*(M+R) (N retailers, M+R facilities).

However, it is not uncommon for subcontractors to employ minimum supply constraints. One of the reasons of such a thing is that demands could be LTL for subcontractors. Secondly, the order amount could not be less than break-even point of subcontractor. Then they will not want to sell or produce less than some amount, as they can not profit from such an agreement.

Another such situation may arouse for real facilities too. Generally the fixed costs of facilities are calculated based on some expectations and assumptions. However, under some production volume it may be inefficient to use that facility or below some volume fixed and assignment costs related to facility may increase beyond the expected rates.

Cases mentioned above can easily be integrated into CFLPSS with a single addition of minimum supply constraints (MM). This constraint is very similar to capacity constraints, except for the direction of inequality. For capacity constraints the total demand assigned from (or produced at) that facility could not be larger than capacity ($\Sigma_j\ d_i\ x_{ij} \leq K_j\ y_j$). However, for MM the total demand supplied from that facility should be larger than some predetermined amount ($\Sigma_j\ d_i\ x_{ij} \geq L_j\ y_j$) if the production of facility is positive.

Although one can model sub-contractors case with assigning a different set of decision variables, we preferred not to change the normal formulation. This is due to the fact that, by this way we can take advantage of the highly efficient algorithms devised for CFLPSS and vast information in the literature regarding CFLP and CFLPSS.

Besides these cases mentioned above, such problems (CFLPSSMM) may exist in many real life circumstances. For instance in community service sector like school, hospital, police station and fire station location problems. In these sectors serving community is the real objective. So it may decrease the quality of service if the facilities serve less than some plausible amount. For instance, it may be less costly to open several hospitals rather than few central hospitals; however this may decrease the patient satisfaction as it is more probable that equipment in these hospitals are scarce or worse. Another possible sector could be waste incineration.

3.2 Model Formulation

The mathematical model and notations that are formulated below represent the capacitated facility location with single sourcing constraints and minimum supply requirements (CFLPSSMM):

Sets

N number of retailer/demand-point locations

M number of facility/supply-point locations

Subscripts

i subscript for retailer location, I Є [1….N]

j subscript for facility location, J Є [1….M]

Decision Variables

$$x_{ij} = \begin{cases} 1 & \text{if demand i is satisfied from facility j} \\ 0 & \text{o.w.} \end{cases}$$

$$y_j = \begin{cases} 1 & \text{if facility j is opened} \\ 0 & \text{o.w.} \end{cases}$$

Parameters

c_{ij} costs of producing and transporting all the demand i from facility j.

F_j fixed cost of opening facility j

d_i demand of retailer i

K_j capacity of facility j

L_j minimum supply limit on facility j

Assumptions

- Facility locations are a discrete finite set (topology)
- There is only one commodity(or service) to be provided in a single planning horizon
- It is a single stage distribution system with single facility types
- All demands must be satisfied
- Demands are deterministic
- Each demand can only be satisfied by a single facility (no splitting allowed)
- Facilities can not supply beyond their capacities and capacities are constant
- If a facility supplies at least one demand, then it must supply more than MM
- Facilities are either opened or closed (no partially opened facilities)
- Objective is to minimize total cost

Mathematical Model

In the light of the previous notations and considering the assumptions, our model can be formulated as follows:

$$\min \sum_{i=1}^{N} \sum_{j=1}^{M} c_{ij} x_{ij} + \sum_{j=1}^{M} F_j y_j \tag{1}$$

s.t.

$$\sum_{j=1}^{M} x_{ij} = 1 \qquad \forall i, \tag{2}$$

$$\sum_{i=1}^{N} d_i x_{ij} \le K_j y_j \qquad \forall j, \tag{3}$$

$$\sum_{i=1}^{N} d_i x_{ij} \ge L_j y_j \qquad \forall j, \tag{4}$$

$$x_{ij} \le y_j \qquad \forall i, j, \tag{5}$$

$$y_j \in \{0,1\} \qquad \forall j, \tag{6}$$

$$x_{ij} \in \{0,1\} \qquad \forall i, j, \tag{7}$$

Explanations of the Objective Function and Constraints

As discussed previously, objective function of FLP (1) is composed of 2 cost terms. First part of the objective, with the double summation, represents the assignment costs of allocating customers to facilities. On the other hand, second part with the single summation is the fixed costs of opening facilities.

The objective function minimizes total costs, which include these assignment and fixed costs.

Constraint set (2) is known as "demand constraints". It ensures the demands of all customers are satisfied.

"Capacity constraints" or constraint set (3) limits the amount of goods or services, which can be supplied by facilities. "K_j" is the capacity limits of the respective facility "j". No facility can produce than its capacity limit K.

Constraint set (4) is minimum supply requirements. They work much similar like capacity constraints, except for the fact that they impose limits on the amount supplied by facilities in a reverse fashion. Capacity constraints limit the total amount supplied by a facility from above. On the contrary, minimum supply requirements (MM) limit the minimum amount of supply for an open facility. In other words, capacity constraints act as an upper bound and MM act as a lower bound on the amount of supply of open facilities. If this constraint is omitted resulting problem will be same as CFLPSS.

Constraint set (5) is a supplementary set of constraints that is also known as "strong formulation" constraints. They originate from UFLP. It is possible to deduce them from other constraint sets, so removal of these constraints will not affect the main aspects of the model. The main reason of their inclusion into the model is due to their beneficial effect on the LP relaxation of the model. Also from experimentation it is observed that they reduce the memory requirements in direct approach (in CPLEX solutions we have used) by preventing branch and bound tree getting larger. This fact reduces the possibility of a memory overflow and getting an abrupt termination.

Constraint set (6) ensures, the facilities are either opened or closed. Finally, the constraint set (7) is single source constraints (SS). This set forces each demand to be served by exactly one facility. Without (4) and replacing (6) with a continuous range from 0 to 1 will result in CFLP.

Chapter 4

HEURISTIC

In previous chapter we discussed briefly about a new problem, CFLPSSMM, derived from CFLPSS. We also provided a mathematical model for this problem. In this chapter we discuss our proposed solution approach and subsequently, our algorithm will be provided.

4.1. Lagrangian Heuristic (LH)

In CFLPSS all variables are integer valued. An important consequence of this fact is a large branch and bound tree. As the number of facilities and retailers increases in polynomial fashion, the solution space grows in a exponential fashion. Therefore, heuristic approaches are common in context of CFLPSS. As reader may recall from section 2.3.2 (solution procedures for CFLPSS), nearly all of the work in this area are based on lagrangian procedures.

Agar & Salhi in their work ([1]) remarks that: "LR (lagrangian relaxation) was inspired from an important observation that the formulation of many hard combinatorial problems consists of an easy problem made difficult by the addition of a set of constraints." (pg. 1074). Transfer of this constraint set into objective function with penalty coefficients is the main idea behind lagrangian relaxation. Then the value obtained from the objective function of this relaxed problem will provide us a lower bound of the main problem. Moreover, this solution can be adjusted to provide an upper bound (feasible solution). Subsequently, the method which is known as subgradient search can be employed to obtain a new set of penalty coefficients (lagrangian multipliers) and hence a new lower bound. Finally, this framework can be repeated until an acceptable solution is found or time/iteration limit is reached. Three main components of a LH is lagrangian relaxation, primal heuristic and subgradient search as mentioned above.

The initial step of devising a LH is to decide, which constraint shall be relaxed. In the lagrangian heuristics in the literature of CFLPSS, demand constraints (Sridharan [26] , Holmberg [21]), capacity constraints (Hindi [20]) or both of these constraints at the same time (Agar & Salhi [1]) are relaxed. Relaxing both constraints at the same time will provide worst lower bounds but it also results in the easiest sub-problems to solve. A&S ([1]) and Beasley ([5]) used this method, in the purpose of devising general heuristics that can be applied to a wide variety of FLPs. Nevertheless, our intentions are to devise a heuristic for a particular set of problems, CFLPSS and CFLPSSMM (Note that CFLPSSMM is a general case of CFLPSS so it is quite intuitive that any algorithm that can solve CFLPSSMM, will also solve CFLPSS). Therefore, it is obvious that this is not a perfect choice for a specific heuristic.

Relaxing capacity constraint in lagrangian fashion is proved to give better lower bounds than relaxing demand constraints (Cornuejols [8]). Nonetheless, as acknowledged by many researchers, the excessive computational effort required to solve its sub-problem is not worth the minimal increase in the lower bounds. As a matter of fact its sub-problem is NP-Hard too, so in worst case a heuristic for this kind may be very unproductive.

The research in this field has shown that the most plausible choice would be relaxing demand constraints, as it results in quite good lower bounds in reasonable amounts of time. Holmberg et al. [21] and Sridharan [26] used this method and achieved comparably good results.

Our LH is also based up on relaxing demand constraints. We preferred the same methodical line as in Holmberg et al. and Sridharan. By relaxing demand constraints the lagrangian sub-problem reduces to 0-1 Knapsacks for CFLPSS and interval Knapsacks for CFLPSSMM. Although, these problems are also NP-Hard there exist quite efficient algorithms in the field to solve these. In addition to these sub-problems are much smaller than the main problem in terms of variable and constraint numbers, so B&C approaches seem to efficiently solve these small instance problems.

Our LH consists of 4 main parts:

- Lagrangian Relaxation (Z_D)
- Primal Heuristic
- Subgradient Search
- An improvement procedure (SSTP)

In the successive parts of this chapter, Sridharan's notation from his work [26] is used for mathematical models and subgradient search. The reader can refer to his work for more extensive information.

4.1.1 Lagrangian Relaxation (Z_D)

As reader may recall from previous section, the main idea behind the lagrangian relaxation is transferring a complicating set of constraints to the objective function by assigning penalty coefficients. The problem which is relaxed in this manner is also known as the "lagrangian relaxation" of the problem. Obviously, the solution space of lagrangian relaxation is at least as large as main problem as it has a set of relaxed constraints.

Our main problem, CFLPSSMM, is (1) subject to the constraint sets (2) – (7). From now on, we will refer the main problem as "Z". In this thesis we apply lagrangian relaxation to the demand constraints (2) in Z. The resulting objective function is:

$$\min \sum_i \sum_j c_{ij} x_{ij} + \sum_j F_j y_j + \sum_i u_i (1 - \sum_j x_{ij}) \tag{8}$$

where u_i are respective lagrangian multipliers for each relaxed demand constraints in the form $(1-\sum_j x_{ij})$. This new cost term, the third term, stems from the lagrangian relaxation of (2). This problem is known as the lagrangian relaxation of Z with lagrangian multipliers "u". With the proper adjustments this problem can be formulated as follows:

(Z[LR(u)]): "Z" s "L"agrangian "R"elaxation with lgr. multipliers "u"

$$\min \sum_i \sum_j (c_{ij} - u_i) x_{ij} + \sum_j F_j y_j + \sum_i u_i \tag{9}$$

s.t.

(3), (4), (5), (6) and (7)

As demand constraints are relaxed in this formulation, there is no constraint left to relate x_{ij} variables of different facilities. Therefore, Z[LR(u)] can now be divided into "M" independent sub-problems Z_j[LR(u)], one sub-problem for each facility. It is the same relaxation of Sridharan [26], except for the addition of minimum supply constraints (12) in Z(LR(u)) and addition of lower bound L_j in (14) of sub-problems Z_j(LR(u)):

$$Z(LR(u)) = \min \sum_j Z_j[LR(u)] + \sum_i u_i \ , \tag{10}$$

s.t.

$$\sum_j K_j y_j \geq \sum_i d_i \tag{11}$$

$$\sum_j L_j y_j \leq \sum_i d_i \tag{12}$$

where ,

$$Z_j(LR(u)) = \min \sum_i (c_{ij} - u_i) x_{ij} + F_j y_j \tag{13}$$

s.t.

$$L_j \le \sum_i d_i x_{ij} \le K_j \tag{14}$$

$$x_{ij} \le y_j \quad \text{for all i,j} \tag{15}$$

(6) and (7).

Note that it is trivial for solution of Z_j[LR(u)] that:

- It is equal to "0" if respective y_j value is "0" as all x_{ij} will also take value "0" to satisfy constraint set (15)

- It is equal to solution of $Z_j(LR(u)) = Z_j(LR(u))^*$

So the Z[LR(u)] and Z_j(LR(u)) can be reformulated as following according to this relation with corresponding y_j values:

$$Z(LR(u)) = \min \sum_j Z_j[LR(u)] y_j + \sum_i u_i \tag{16}$$

s.t.

$$\sum_j K_j y_j \ge \sum_i d_i \tag{17}$$

$$\sum_j L_j y_j \le \sum_i d_i \tag{18}$$

$$y_j \in \{0,1\} \qquad \forall j, \tag{19}$$

and,

$$Z_j[LR(u)] = \min \sum_i (c_{ij} - u_i)x_{ij} + f_j \tag{20}$$

s.t.

$$L_j \le \sum_i d_i x_{ij} \le K_j \tag{21}$$

$$x_{ij} \in \{0,1\} \qquad \forall i, j\,, \tag{22}$$

Z_j (LR(u)) are solved by CPLEX's MIP solver module. These will give coefficients of the y_j variables in Z(LR(u)). Then Z(LR(u)) is solved by CPLEX's same module to obtain the solution of Z(LR(u)).

Note that the constraint (15) is discarded from the Z_j (LR(u)). So x_{ij} values of closed facilities can be "1" in the resulting solution. Therefore, after obtaining a solution from the Z(LR(u)), we set x_{ij} values of closed facilities as "0":

$$x_{ij} = \begin{cases} x_{ij}^* \text{ if } y_j^* = 1 \text{ (which is obtained from solution of Z(LR(u)))} \\ 0 \text{ otherwise} \end{cases}$$

This modification will not alter the objective value of Z(LR(u)) as none of closed facilities' costs are added in Z(LR(u)), it just ensures feasibility. As a result the resulting solution is feasible for Z(LR(u)) and optimal for its relaxed counterpart (the one without the constraint set (15)). This same application is used in Sridharan's [26], reader can check Theorem 1 in his work to check the same principals.

(Z_D)

$$Z_D = \max_U Z[LR(u)] \tag{23}$$

Z_D is the lagrangian dual problem, with demand constraints relaxed in a lagrangian fashion. It is the maximum value that can be attained by sub-problems we formulated previously as $Z[LR(u)]$. As Z[LR(u)] are relaxations, their optimal solution can be at most equal to the main problems optimal (Z^*). So the maximum of these problems, namely Z_D, theoretically provide the same solution with main problem Z. The complication is that we do not know for which values of "u", Z_D is maximized. Consequently, we solve $Z[LR(u)]$ as the lagrangian sub-problem to achieve tighter lower bounds. Selection of lagrangian multipliers is a different issue and will be discussed later in "subgradient search" section.

Our lagrangian sub-problem is $Z[LR(u)]$, which is a knapsack problem. Its sub-problems Z_j[LR(u)] are interval knapsack problems.

We used the general B&C procedure of CPLEX to solve interval knapsacks and the main knapsack problem. Even though it is a general case algorithm, it has impressive results in solving knapsacks. More extensive information on lagrangian relaxation can be found in chapter 10 of Wolsey [37] and for application of lagrangian relaxation to CFLPSS problems with relaxing demand constraints, reader can check A & S [1] , Holmberg [21] and Sridharan [26] .

4.1.2 Primal Heuristic

After creating lower bounds in previous step, we aim to obtain good feasible solutions in next step. We employ a very basic greedy heuristic that is modified from Holmberg et al's [21] to create a feasible solution from the result of lagrangian relaxation. The solution from the lagrangian sub-problem is known to satisfy capacity constraints, single source constraints and minimum supply requirements. It satisfies all the necessary requirements of a feasible solution, except for the demand constraints. Therefore, the main point is, if we modify the solution to satisfy demand constraints without violating feasibility.

Three possible scenarios can take place for a demand “i” assignment in the solution of lagrangian sub-problem. First of all, the demand “i” can be assigned exactly to one facility ($\sum_j x_{ij}=1$), in which case there is no violation of demand constraint for this particular demand “i”. This demand is “normally supplied”. Secondly, it may be supplied from more than one facility at the same time ($\sum_j x_{ij} > 1$), causing an “over supply”. Finally, the demand may be not satisfied at all ($\sum_j x_{ij} =0$), “no supply”.

The set of demands that are assigned to single facilities, form the initial basis of our primal heuristic. Then the set of "over supplied" demands are investigated and the extra supplies are removed until only one facility supplies these demands. These reductions are based on a greedy approach that removes the one with the highest assignment cost value first. After this iteration, the remaining demands are either supplied by a single facility or by no facility at all. Then, we start to assign unsupplied demands. The algorithm prefers opened facilities over to the non opened ones for assignment of demands. If there is enough capacity in opened plants to satisfy a non supplied demand, the algorithm chooses the one with the lowest assignment costs. Otherwise, algorithm opens a plant and assigns the demand based on the lowest cost alternative, where the total cost is fixed cost of opening facility plus the assignment cost to that facility.

Note that this algorithm does not guarantee a feasible solution. Based on experimentation the most common violation, that prevents primal heuristic from producing feasible solutions is minimum supply requirements. Such an infeasibility is expected to occur especially when the last few demands are assigned to a newly opened facility. So we apply a procedure that increases the chances of getting a feasible solution, by reassigning some of the demands to ensure minimum supply requirements met.

Besides this case, it is also possible that some demands could not be assigned. However, this is a possibility for cases where capacity is very tight compared to demands. In general case, as the total capacity is quite larger than total demand, so it is an uncommon thing to encounter. In none of our experiments such a case is encountered. To sum, the primal heuristic in combination with the feasibility improving procedure does not guarantee to yield feasible results. Although this fact, this is a very rare occurrence and may occur only for cases where capacity constraints are very tight. At the end of the chapter, the exact algorithm which is coded in C for primal heuristic and its feasibility improver is given.

4.1.3 Subgradient Search

The subgradient search is similar to that of Holmberg. By experimentation it is observed that Holmberg's subgradient search can effectively be implemented to give tight lower bounds. Lagrangian multipliers are calculated according to the following procedure, where "k" stands for the iteration number, "i" for the customers and "j" for facilities.

$$NU(i) = 1 - \sum_{j \in J^*} x_{ij}^* \tag{24}$$

NU are the subgradients, to the concave function Z(LR(u)), this same subgradient is used both in Holmberg [21] and Sridharan [26] in CFLPSS. As the objective function Z(LR(u)) are same for both CFLPSS and CFLPSSMM (only constraint sets differ by extra MM constraints in CFLPSSMM), it is trivial that it can be used as a plausible subgradient in CFLPSSMM.

The lagrangian multipliers for iteration k+1 can be calculated by the following formula.

$$u_i^{k+1} = u_i^k + t_k NU(i) \tag{25}$$

where, u_i^k are lagrangian multipliers found in k th iteration and u_i^1 (initial lagrangean multipliers) are selected as:

$$u_i^1 = \min_j (c_{ij}) \tag{26}$$

This particular initial multiplier values are selected according to Sridharan's [26] method.

t_k is the step length calculated according to the following formula.

$$t_k = \frac{\lambda_k (Z^{UB} - Z(LR(u)))}{Norm^2} \tag{27}$$

Z^{UB} is the best feasible solution found so far and Z(LR(u)) is the most recent objective value obtained from lagrangian sub-problem (see lagrangian relaxations section). "Norm" value is the Euclidean norm of subgradients NU(i) , can be formulated as:

$$Norm = \left\| NU^k \right\| = \sqrt{NU(1)^2 + NU(2)^2 + ... + NU(N)^2} \tag{28}$$

All the respective NU(i) are calculated based on the formula given in the beginning of this section and on the most recent lagrangian relaxation solution (Z(LR(u))).

$$0 \leq \lambda_k \leq 2 \tag{29}$$

λ_k is taken as 1, and it is halved each time the lower bound does not improve for 5 iterations. Norm is taken as the Euclidean norm as in Sridharan's work [26]. If norm drops to "0" at any iteration we stop the LH, as it means an optimal solution is found. Note that, sub-problem Z(LR(u)) satisfies all the conditions of CFLPSSMM except the demand constraint.

Norm can only be "0" if and only if exactly one x_{ij} per each "i" is 1. In other words, Norm takes the value "0" only when all the demands are exactly satisfied. As the resulting solution satisfies all the constraints now and as we know that value of Z_D can not exceed the value of original problem (comes from the definition of relaxation), the resulting solution must be optimal solution.

Subgradient search is done by a code of the author that is written in C.

4.1.4 SSTP

Fixing subset of y_j open (for $j \in J^*$) in a CFLPSS results in a single source transportation problem (SSTP). This is the main primal method Sridharan [26] used to obtain feasible solutions for his lagrangian heuristic. There are several weaknesses of this method. The first of all it is an IP, so it may require large computational times to calculate at each step. Additionally, it does not guarantee to yield a feasible solution.

A modification of SSTP that is derived from CFLPSSMM is described as follows:

(SSTP)

$$Z(TP) = \min \sum_i \sum_j c_{ij} x_{ij} \tag{30}$$

s.t.

$$\sum_i x_{ij} = 1 \qquad \forall\ i\ , \tag{31}$$

$$\sum_i d_i x_{ij} \le K_j \qquad \forall j : j \in J^* \tag{32}$$

$$\sum_i d_i x_{ij} \ge L_j \qquad \forall j : j \in J^* \tag{33}$$

$$x_{ij} \in \{0,1\} \qquad \forall\ i, \forall j : j \in J^* \tag{34}$$

where $J^* = \{j : y_j^* = 1\}$

Set J^* represents the set of open facilities. This approach divides main problem into two sub-problems. The first part is decision of the set J^* and the second part is SSTP. The key remark is that if the optimal y_j values are fixed open, the problem will provide the optimum solution. Unfortunately, finding optimal y_j itself is a difficult problem on its own.

It is observed that, SSTP could be an inefficient algorithm to use as a primal heuristic. On the contrary, it is a very good aid as a primal heuristic supporter. Sridharan used SSTP directly after lagrangian sub-problem, but this method does not guarantee yielding feasible results. We approach the subject from a different perspective.

Primal Heuristic produces feasible results based on a greedy approach. However, such approaches are known to give different solutions than optimal solutions most of the time. So, they are used commonly in conjunction of methods as exchange heuristics to improve the solutions of greedy approaches.

We noticed that primal heuristic forms a perfect starting basis for SSTP. The y_j values found by primal heuristic guarantees that at least one feasible solution exists in the solution space of SSTP. By fixing this set of y_j 's from primal heuristic, we ensure the respective SSTP formed will not turn out to be infeasible. The solution from SSTP will provide a upper bound at least as tight as the primal heuristic, which it is created from. Due to it's computational complexity, it may not be a good idea to apply this improvement procedure SSTP every step the primal heuristic finds a feasible solution. So this method is applied at most a given number of times. By experimentation, applying SSTP 4 times during the heuristic and 1 time in the end is found out to be quite effective. The final SSTP is formed on the y_j values of best feasible solution found. We used CPLEX to solve SSTP's.

4.1.5 Remarks

Our heuristic is composed of these 4 routines that were outlined in previous chapters. The lagrangian relaxations are used to find lower bounds and primal heuristics in conjunction with SSTP provides feasible solutions. Then subgradient search is employed to calculate new set of lagrange multipliers so a new lower bound.

This is an iterative procedure and unless optimality conditions are satisfied, 400 iterations are done before termination. There are 2 optimality conditions for our heuristic. First one is the norm value taking 0 as mentioned in subgradient search. Second one is getting a lower bound (Z_{LB}), upper bound (Z^{UB}) that satisfy the following constraint:

$Z^{UB} < Z_{LB} + 1$,

This result is derived from the observation that; as all decision variables and costs are integer, the resulting feasible solution will be integer as well. As the Z_{LB} exceeds the next smallest integer value of Z^{UB}, Z^{UB} must be optimum.

Reader can check figures 4.1 and 4.2, to obtain brief information about heuristic. Figure 4.1 provides a flowchart of the heuristic and figure 4.2 is the pseudo-code of the heuristic. The LH is a C code that solves knapsack problems in the lagrangian relaxations and SSTP problem by using callable libraries of CPLEX. Main frame of our algorithm, Primal Heuristic and Subgradient Search are employed by C coding.

Lagrangian relaxations are based upon observations of Hindi[20] ,Holmberg [21] and Sridharan [26]. Primal heuristic is modified from Holmberg et al's [21] simple primal heuristic, modifications are done to tackle this general case of CFLPSS (CFLPSSMM). To improve feasibility a small greedy heuristic that exchanges demands across facilities to decrease MM violations is included. Subgradient search is modified from Sridharan [26] and Holmberg [21]. Initial lagrangian multiplier selection is based on Sridharan [26] and step size is calculated according to Holmberg [21]. SSTP is also used in Sridharan [26] but he used it as a primal heuristic. We modified it and it is used to improve the existing feasible solutions obtained from primal heuristic. By this way we decreased the amount of time spend to create feasible solutions and guarantee to get feasible solutions from SSTP.

4.2 Direct B&C approach by CPLEX

Rather than employing a heuristic, a second approach that could be plausible to solve such a problem is a Branch and Cut (B&C) approach. A B&C algorithm is a B&B algorithm with cutting planes generated throughout its B&B tree. (Wolsey [37]).

CPLEX, the state of art software, employs efficient methods for general case MIPs. CPLEX uses an efficient B&C procedure where several sets of cutting planes are generated as clique, Gomory and cover (refer to CPLEX user guide [10]). Therefore, it provides a good basis to compare our results from the heuristic. Callable library functions of CPLEX are used to operate CPLEX.

STEP1: Take the x_{ij} values and y_j values from the last lagrangian iteration.

STEP2: Group customers into 3 sets according to their demand constraint violations.

- Assign to set1 if for that customer i, $\sum_j x_{ij} = 1$, "normal supply"

- Assign to set2 if for that customer i, $\sum_j x_{ij} > 1$, "over supply"

- Assign to set3 if for that customer i, $\sum_j x_{ij} = 0$, "over supply"

STEP3: Select a customer from set2 , randomly. Select among the x_{ij} (assignment variables) of this customer that is "1" with highest c_{ij}.

-Remove it if its removal does not violate minimum supply requirements. If its removal closes a facility, set corresponding facility closed.

-If its removal violates MM constraints, pass to the next possible assignment variable.

STEP3: Repeat this procedure until the demand of customer i is assigned by only 1 facility. If its not possible to remove one or more of these extra supply assignments without violating MM, remove the one (s) with highest c_{ij} values and increase the MM violation count by 1 for each removal that violates a new MM requirement.

STEP4: Return to STEP2 as long as set2 has elements.

STEP5: Select a customer from set3, randomly. Assign it to one of the open facilities with enough demand. If there is a MM violation from previous steps, assign it to the facility with MM violation if there is enough capacity to assign it. Otherwise, select the one with lowest c_{ij}. If there is no enough capacity in open facilities pass to STEP6.

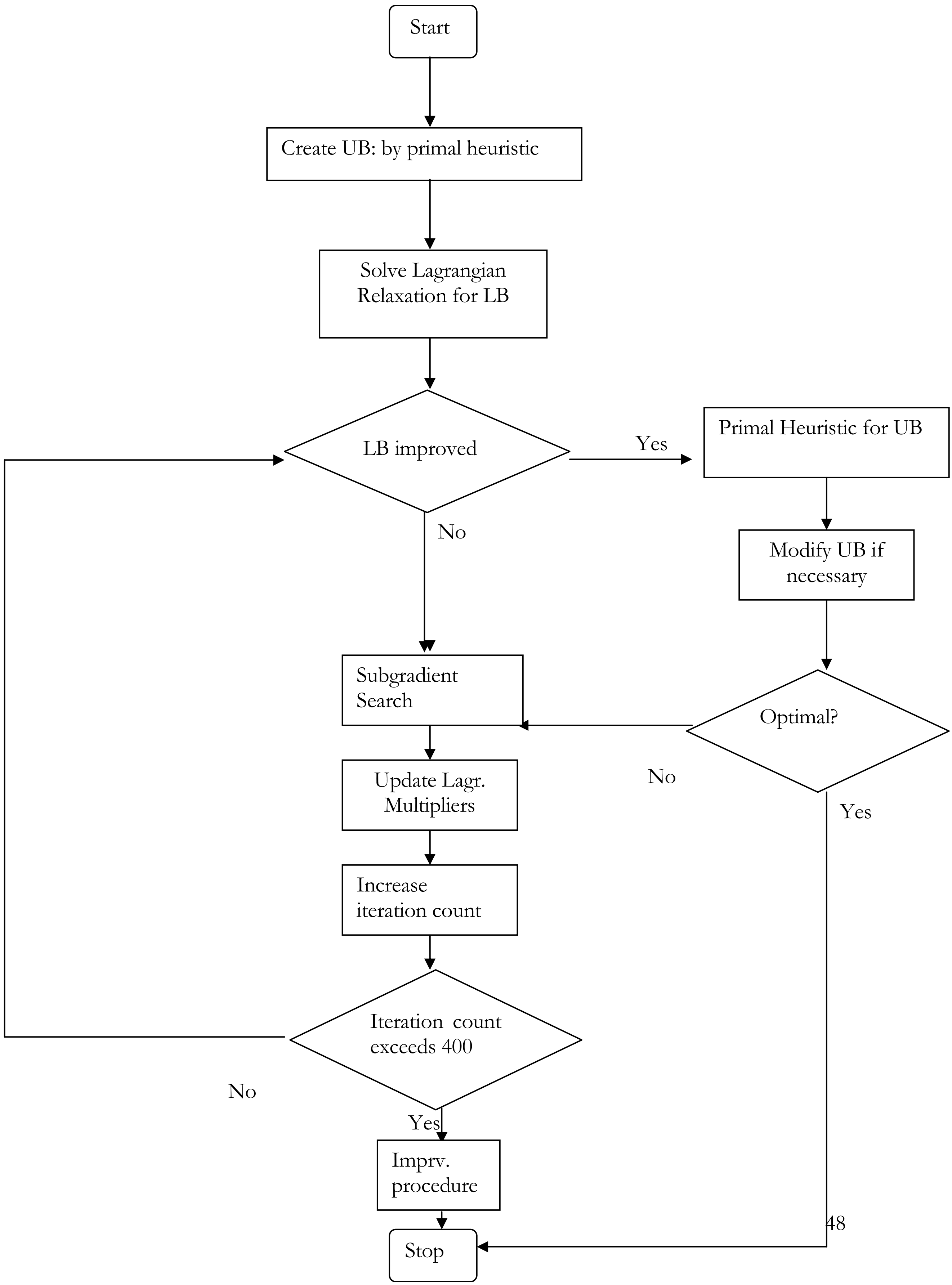
STEP6: If the customer can not be assigned to one of opened facilities, open the facility among the facilities that have enough capacity to supply demand with lowest ($c_{ij} + F_j$) assignment plus fixed cost.

STEP7: Return to STEP5 until all the elements of set3 are assigned. If there is no enough capacity to assign any one of set3 members, terminate the algorithm with infeasibility.

STEP8: If there is MM violations, select a random facility with MM violation. Assign remove a demand among from another facility that will not violate an MM violation itself and close the facility if it was its last demand supplied. The selection of demand and facility that will be reassigned is based on the minimum increase or maximum decrease (whichever is the case) in the cost that will be caused by this move. Repeat until all MM violations are restored. If no such reassignment can take place without violating a MM for the facility that the demand will be removed, terminate the program with infeasibility.

TERMINATE: If there is no capacity or MM violation , solution is primal feasible and return the solution to main program. Otherwise return infeasibility to main program, which will result discarding of this solution of Primal Heuristic.

Figure 4-1: Primal Heuristic algorithm



48

FIGURE 4.2: Flow chart of the heuristic

STEP1: Initialize $u^0[i] = \min_j (c_{ij})$, $ZUB = \sum_j f_j + \sum_i \max_j c_{ij}$, $ZLB = -\infty$,try to use primal to obtain a better ZUB;

STEP2: Solve Knapsack sub-problems for a LB (CPLEX is used), if LB > ZLB, modify ZLB = LB;

STEP3: If lower bound improved in previous step or per each 10 iteration apply primal heuristic (coded in C) else goto STEP5;

STEP4: Set UB=primal heuristic solution, if UB < ZUB modify upper bound ZUB=UB. Store y_j of best incumbent solution;

STEP5: If ZUB < ZLB + 1 terminate algorithm, found optimum else go to STEP6

STEP6: Apply subgradient search (coded in C), update $u^k[i]$

STEP7: If Norm = 0, terminate found optimum (ZLB is optimum in this case)

STEP8: Increase iteration count by 1, if iteration count > 400, terminate. Best feasible solution found is ZUB. Else go to step2

<u>Terminate</u>: Use CPLEX to solve SSTP for best incumbent solution's y_j values. Return ZUB, ZLB, Time

FIGURE 4.3: Pseudo-code of the Heuristic

Chapter 5

COMPUTATIONAL EXPERIMENTS

In this chapter, we will discuss our computational experiments. We start the chapter by introducing our experimental setting, and then we will continue with the results of the computational experiments. Finally we will analyze the results.

5.1 Settings

The algorithm we devised is intended to work on CFLPSSMM. As this problem is a general case of CFLPSS from the literature, it is also expected to give good solutions for CFLPSS. Therefore, we tested our algorithm both on CFLPSS and CFLPSSMM.

We used 2 different data sets to test our heuristics. The first data set is from the literature. It is the data set that Holmberg et al used to test their algorithms in their work [20]. The structure of the problems showed that they are not reasonable alternatives to test CFLPSSMM. This is due to the fact that in optimal solutions, nearly all of the capacities of open facilities are used. Therefore, unless added minimum supply requirements are very tight it will not change the optimality of the existing solutions. Consequently, we only tested CFLPSS with this data set. This data set, which is available at OR Library ([34]), consists of 4 smaller data sets. They range from 50v10 (N=50, M=10) to 200v30. Their sizes can be checked from the Appendix -1.

70 instances exist in this data set. Note that in original data set there are 71 instances. The missing instance is the problem 67, whose data was corrupted.

The second data set we used is a generated data set. Its generation procedure is very similar to Holmberg's [21] except for assignment costs. The capacities (K_j) and fixed costs (F_j) are in the range of 500 to 800 and 500-1500, respectively. The demands are uniformly distributed across the range 30 to 80. The assignment costs range from 1 to 4 per unit of demand. The respective distributions of parameters are tabulated in Appendix-2.

For 2nd data set, we tested 4 different scenarios based on the stringency of minimum supply requirements. The stringency of minimum supply requirements range from %0 to %40. The stringency value (S_j) shows the ratio of minimum supply (L_j) to capacity (K_j). A problem instance with %40 stringency must supply at least %40 of its capacity, if it is open. Or we can say that $L_j = K_j * S_j$. The scenario for %0 stringency is the standard CFLPSS. There are 4 sizes of problem instances, that range from 40v10 to 160v40 (40 and 160 customers respectively; 10 and 40 possible facility sites respectively).

A group of problems exist for each combination of 4 scenarios and problem sizes, with each group having 10 instances. It makes a total of 160 instances. In addition to our LH, a direct CPLEX approach is also tested on this 160 instances. CPLEX version 8.1 is used to test these data and this is done via a C code interacting with callable libraries of this commercial software. As the problem in hand is IP, the memory requirements and computational time is an important issue. Therefore, a time limit is imposed on the CPLEX. This time limit is 1000 seconds for small, medium and large sizes. For the problem instances of X-large size (160v40), this time limit is increased to 2000 seconds.

Both our LH and the CPLEX approach is tested on the "Pascal" server of Bilkent University. It is a UNIX server with 2.6 Ghz dual AMD CPU and 2GB RAM . During calculations only 1 CPU is assigned to CPLEX and LH.

5.2 Computational Results

The computational results of the experiments and interpretation of tables will be discussed in this part. As the computational experiments are performed on two separate data sets, we will discuss their results under two separate headings.

5.2.1 Computational Results of Data Set 1 (Literature Data)

The first data set we performed experiments was taken from Holmberg et al. There are 4 subsets in this data set. The first subset instances range from p1 to p24. They can be considered small sized problems. They have 50 customers to be satisfied and 10 possible locations of facilities. In table 5-1, we present the computational results for problem instances p1 to p12 from data set 1. Based on this table, we provide some brief explanations about data set 1 tables from Appendix 1.

Problem	Optimum	LH			Gap	
	Zopt	ZUB	ZLB	Time	Opt	LH
p1	8848	8848	8848	0,16	0,0000	0,0000
p2	7913	7913	7913	1,75	0,0000	0,0000
p3	9314	9314	9313,7	0,11	0,0000	0,0000
p4	10714	10714	10695,9	2,74	0,0000	0,0017
p5	8838	8838	8837,1	0,31	0,0000	0,0000
p6	7777	7777	7776,42	0,13	0,0000	0,0000
p7	9488	9488	9479,39	2,58	0,0000	0,0009
p8	11088	11088	11079,44	2,54	0,0000	0,0008
p9	8462	8477	8453	0,93	0,0018	0,0028
p10	7617	7617	7610	0,78	0,0000	0,0009
p11	8932	8932	8932	0,15	0,0000	0,0000
p12	10132	10132	10114	1,32	0,0000	0,0018
			Average=	1,13	0,0001	0,0007
			Total=	13,50	0,0018	0,0089

Table 5-1: Computational Results for Data Set 1 instances

Under the heading of "Optimum" there is Zopt. These values are the optimum values of corresponding problem instances, which are taken from literature. The next heading from the table is "LH" with 3 sub-headings beneath: "ZUB", "ZLB" and "Time". These are three main results of our LH. Under ZUB and ZLB the best feasible solution and the best lower bound values found from our LH is stored respectively. "Time" is the total run time for our heuristic for that particular instance in CPU seconds. In next column of the table, the gaps are stored. The first gap value gives : $\frac{Zopt - Z^{UB}}{Zopt}$ under the heading "Opt". The second one gives the duality gap between ZUB and ZLB ($\frac{Z^{UB} - Z^{LB}}{Z^{LB}}$). Other tables of the data set 1 can be found in Appendix 1.

We observed that our algorithm is quite effective in CFLPSS. As a matter of fact, it is an expected result, because our algorithm is an extension of CFLPSS heuristics. In a significant number of cases our heuristic found the optimal value, even verified the optimality of the solution in considerable amount of these. And for %79 per cent of the time the solution of LH returned quite impressive solutions in terms of duality gap (<%1). Only %11 per cent of the solutions have more gap than %3 of optimal solution. For the rest of the solutions the worst solution of LH did not exceed %5.5 of the lower bound found (in p31), and even for this case the real duality gap was less then %4 per cent. Distributions of solutions are illustrated in the figure below. "Optimum" is the number of optimum solutions found. "Verified" stands for the LH solutions that are optimal and LH was able to verify the optimality by the lower bounds obtained.

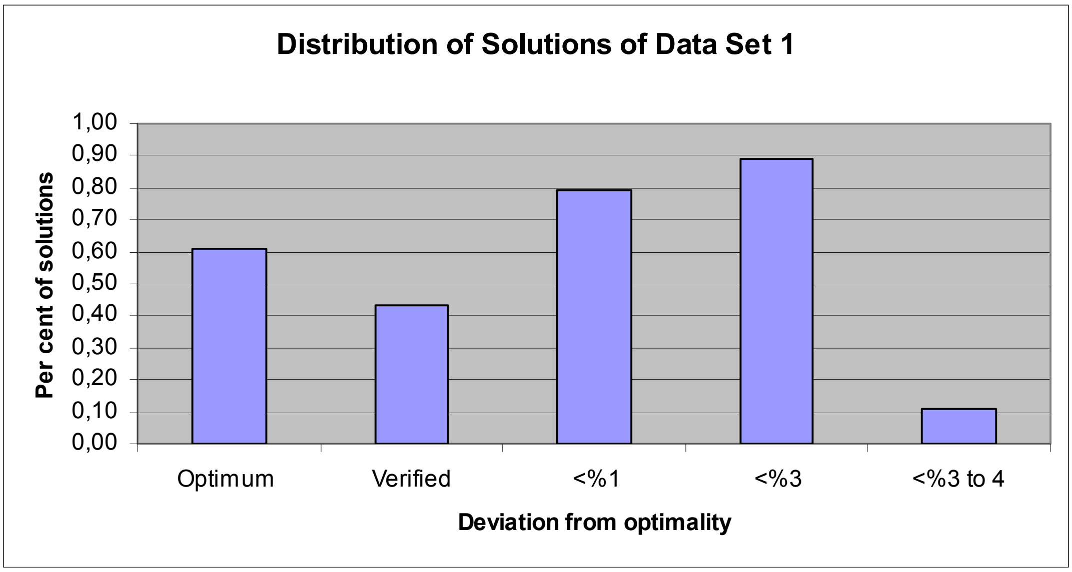


Figure 5-1

The computational times of our LH in data set 1, range from 0.11 CPU second in p3 to 15.42 CPU seconds in p70. The total and average run times for data set 1 are tabulated below in Table 5-2.

Problems	Set1(p1-p24)	Set2(p25-p40)	Set3(p41-p55)	Set(p56-p71)
Total CPU Time	29,86	72,18	40,29	98,24
Average CPU time	1,24	4,51	2,69	6,56

Table 5-2: CPU times of Data Set 1 (in CPU seconds)

We observe that average run times are pretty good considering the sizes of the problems. LH solves small cases with 50v10 in an average time of 1,24 and it can find a solution for a large problem with 200v30 in an average time of 6,56. Despite the LH seeming a bit slower than its counterparts (A&S[1]), note that it is general case solver that is initially intended to solve CFLPSSMM. Therefore, it has some extra procedures and more complex sub-problems than CFLPSS. Regarding this factor, it is obvious that it can be a resourceful CFLPSS solver.

5.2.2 Computational Results of Data Set 2 (Generated Data)

The CFLPSS with MM is the general case of CFLPSS. To the best of the author's knowledge, there is no prior work in this context. In addition, the data available in the literature is inadequate to test this new feature of CFLPSSMM. Therefore, a new set of data is generated according to Holmberg [21] et al.'s distribution of parameters.

This data consists 4 different sizes of problems, varying from 40v10 to 160v40. Each set is tested with 4 different minimum supply scenarios: %20,%30,%40 and %0 minimum supply . For instance in %20 minimum supply case, the minimum supply value of a facility is one fifth of its capacity value. In this case, any open facility must supply at least one fifth of its capacity. We apply our LH and CPLEX approach on a total of 160 instances from data set 2. The results are tabulated in tables in Appendix -2.

In the table 5-3 below we will discuss the results of a case with problem size "small" and minimum supply stringency "low". Each table is consisted of 10 problem instances with the same characteristics and distribution parameters that can be found in A2-ii. Problem size small corresponds to 40v10 in our notation. MM stringency "low" in this example corresponds to %20. These notation can be found in Table A2-i. Shortly, this table corresponds to a case with 40 demands and 10 possible locations. Moreover, each open facility must supply at least %20 of its capacity value.

Problem	CPLEX		LH			Gap		
	Z*	Time	UB	LB	Time	Z*/UB	UB/LB	Time
P1	6681	6.87	6681	6635.9	3.12	0.0000	0.0068	1.20
P2	6051	0.39	6051	6051.0	0.15	0.0000	0.0000	1.60
P3	6993	7.94	6993	6932.4	0.39	0.0000	0.0087	19.36
P4	6446	1.67	6446	6445.2	0.08	0.0000	0.0000	19.88
P5	5890	0.53	5890	5889.5	0.36	0.0000	0.0000	0.47
P6	6256	0.52	6256	6256.0	0.12	0.0000	0.0000	3.33
P7*	5695	1000.00	5695	5624.2	3.42	0.0000	0.0126	291.40
P8	6362	7.33	6382	6382.0	2.78	0.0031	0.0000	1.64
P9	6082	0.30	6090	6077.3	3.10	0.0013	0.0021	-0.90
P10	6332	2.04	6332	6332.0	0.20	0.0000	0.0000	9.20
Total	62788	1027.59	62816	62625.5	13.72	0.0045	0.0302	347.17
Average	6278.8	102.76	6281.6	6262.5	1.37	0.0004	0.0030	34.72

Table 5-3: Computational Results for Data Set 2 instances
Problem size: Small
MM: Low

The table can be partitioned into three main columns. In the first one CPLEX results can be found. In the second one best UB,LB and time values of LH can be found. Finally, in the last column gap values are notated. Notice that Z^*/UB and Time gaps can be turn out to be negative like Time gap turning out to be negative in p9 of this table. Negative Time gap imply CPLEX performed faster than our algorithm. In this case CPLEX performed 10 times faster than our algorithm, it may seem a bad result but it is not as we will discuss later.

Unlike data set 1, optimal values of problem instances are not known beforehand. Therefore, problems are first approached by CPLEX. CPLEX solutions under heading Z^* are optimal values, unless that particular CPLEX run exceeds the time limit without a verified optimal solution. Such cases are marked with an asterisk ("*") next to the problem instances. For example, problem 7 could not been solved in the time limit of 1000 CPU seconds so the solution provided may not be the optimum solution. On the other hand our algorithm may beat the CPLEX in terms of quality of the solution especially in large instances where CPLEX fails to yield optimal results in plausible times. For such cases Z*/UB gap will turn out to be negative indicating that solution of LH is better. The solution value of CPLEX can be found under Z^* and the computation time of CPLEX run is found under "time" heading under CPLEX.

Our heuristic provides 3 main values. First of all it provides an upper bound or a best feasible solution, that is store below heading "UB". Then it provides a lower bound, which is achieved via lagrangian relaxations. The respective values of this lower bounds can be found under "LB". Finally run times of our LH is stored under "time".

Gap column provides the relative differences. They are calculated according to following formulas:

$$Z^* / UB = \frac{UB - Z^*}{Z^*},$$

$$UB / LB = \frac{UB - LB}{LB},$$

$$Time = \frac{Time_{CPLEX} - Time_{LH}}{Time_{LH}},$$

where $Time_{CPLEX}$ is the run time of CPLEX and $Time_{LH}$ is the run time of our heuristic. From this table we can see that 8 out of 10 solutions of our heuristic is same as CPLEX, which 7 of them are proved to be optimal by CPLEX. In addition, our algorithm performed 34 times faster on average than CPLEX. Even with p7 removed, LH performs approximately 6 times faster. By checking Z*/UB column, it is found that the average percentage error is %0.04 for this case. This value is the amount we deviated from CPLEX solutions (which are 9 optimal out of 10). In other words, to reach solutions 34 times faster, we sacrificed approximately (as the solution to p7 could not be proved to be optimal by CPLEX) %0.04 deviation of optimality. That can be considered a good trade-off. The rest of the computational results of data set 2 can be found in Appendix 2.

	Relative Error w.rt. CPLEX solutions	Relative Error w.r.t Lower Bounds	Cmp. Time Relation btw. LH and CPLEX
40v10	0,0006	0,0032	18,43
60v15	0,0037	0,0058	5,91
80v20	0,0007	0,0028	128,11
160v40	0,0081	0,0191	43,78
Avg	0,0033	0,0077	49,06

Table 5-4: Relative Differences of Data Set 2 According to Problem Sizes

In table 5-4, the average gaps Z*/UB , Z*/LB and Time and are given. They are average values of the gap values formulated below the table 5-3. Tabulation in table 5-4 is done according to the problem size of the sets. Each of the problem sets is composed of 40 instances with same problem size. We observe that as the problem size increases, the average error margin of LH also increases. Although, it does not exceed %0.4 of the CPLEX solutions and %0.8 of the lower bounds provided by lagrangian relaxation. Time statistics show that LH performs between 2499 times faster than CPLEX in best case (p2 of table A2-3d) and 15 times worse than CPLEX in worst case (p1 of table A2-1d). In the worst case, CPLEX solved at a time of 0.04 seconds, where it took LH to solve 0.63 seconds. So "15 times" worse does not indicate significant amount of time; as such cases mostly exist in small sized problems, the time difference is not noticeable at all. On the other side, for large problems the LH performs seemingly 40 times better than CPLEX. However, this 50 times corresponds to 50 CPU seconds of LH run and 2000 CPU seconds of CPLEX run. This makes a difference of 1950 seconds, which would be probably much more if the time limit of 2000 seconds had not existed. As a last remark for this table, while analyzing time statistics we must take the time limits into consideration. From experimentation, it is observed that for 160v40 instances, CPLEX runs may exceed 25000 seconds, without verifying optimality. However as the time limit is just 2000 seconds, it may seem the efficiency of LH decreases as the problem size increases.

	Relative Error w.rt. CPLEX solutions	Relative Error w.r.t Lower Bounds	Cmp. Time Relation btw LH and CPLEX
20%	0,0033	0,0080	60,15
30%	0,0047	0,0090	15,99
40%	0,0027	0,0066	21,86
0%	0,0025	0,0075	98,22
Avg	0,0033	0,0077	49,06

Table 5-5: Relative Differences of Data Set 2 According to MM Requirements

Table 5-5, demonstrates the results gathered from data set 2 from a different perspective. In this table, results are classified based on their minimum supply requirements. As in table 5-4, all the problem sets in table 5-5 consists of 40 problem instances. Table demonstrates that in none of respective sets the duality gap exceeds %0.9. Another interesting result is about time statistics. For "%0" case, problem turns out to be a CFLPSS and LH performed approximately 100 times faster than CPLEX for this case. In addition, in set "%20" LH performs 60 times faster. On the other side, for "%30" and "%40" efficiency of LH seems to decrease. This is quite natural, as for %0 LH do not have to employ extra procedures to ensure to avoid MM violations and its sub-problems are easier in lagrangian relaxation. Also, "%20" minimum supply requirement is not a very tight one. Therefore, structure of the problem does not change much. As for other cases the MM are tight and LH has to spend extra time on solving its harder sub-problems and keep solutions without MM violations.

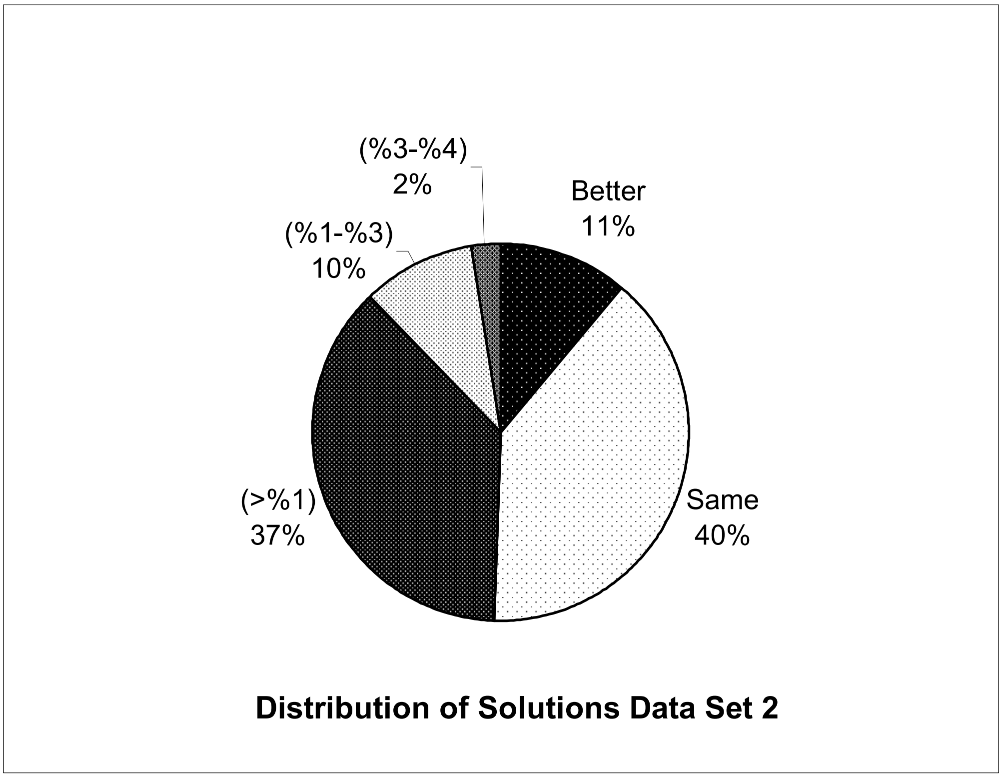


Figure 5-2

In figure 5-2, we present the overall results of LH solutions in terms of CPLEX results. Out of 160 instances in total, our LH performed better than CPLEX runs in 18 instances or %11 of total instances. In an additional 65 of the instances both approaches resulted in same solutions, in %40 of total instances. Moreover, difference between CPLEX and LH results less than only %1 per cent for a further %37 of the solutions 61 instances total. Only for %13 of the results the relative error is worse, and just %2 of these exceed %3. According to these computational experiments, LH on average provides %98 of its results with a small error margin with CPLEX results or better than

CPLEX results. Only 4 out of 160 instances the relative error is more than %3 and in none of the instances this margin exceeds %4.

For small instances CPLEX may prove optimal results within a very short time. However, even for small instances there is a possibility of very high computational times compared to LH. For instance as in p7 of Table A2-1a, CPLEX failed to prove optimality in 1000 seconds, whereas LH found the same solution with CPLEX, in less than 5 seconds. As for large cases, CPLEX has a significant probability of exceeding 8 or more hours (25000 seconds) of computational time, based on our observations. In other words, it is very hard to guess if CPLEX will return the optimal solution in an acceptable amount of time. As for x-large setting (160v40), CPLEX returned verified optimal solutions in less than 2000 seconds for only 2 out of 40 problems. Moreover, with this setting 12 solutions of LH out of 40 was better than CPLEX and LH was able to verify an optimal solution in 35,73 seconds, which the CPLEX can not (see p9 in A2-4c).

Another important aspect of LH is its computational time. It grows more like a polynomial fashion, where as CPLEX solution times increase out of bounds.

Sets	**Avg. Times**
40v10	1,16
60v15	5,55
80v20	9,59
160v40	49,30

Table 5-6: Avg. Times for Data Set 2 (in CPU seconds)

It can be seen from Table 5-6, that run times of LH is quite modest compared to CPLEX. CPLEX exceeded 1000 second time limit in 10 instances out of 40 in large setting (80v20) and exceeded 2000 second time limit in 38 of 40 instances in x-large setting (160v40) (see Appendix 2 Tables 3a to 4d).

CPLEX has better run times in 34 instances, in which only 7 of them in large problem size setting and "none" in x-large problem size setting. The worst case LH solution in terms of difference between run times is instance p9 from Table A2-2a, in which LH solved the problem in 9,62 seconds and CPLEX in 1,60 seconds, resulting in a time difference of approximately 8 seconds.

	Time Gained	Time Spent in CPLEX	% gain
Small	1216	1263	0,96
Medium	1144	1365	0,84
Large	12138	12521	0,97
X-large	76340	78312	0,97
total	90837	93461	0,97

Table 5-7: Computational Times for Data Set 2 (in CPU seconds)

Table 5-7 provides the total time run in CPLEX and the time gains of using LH instead of CPLEX. As it can be seen CPLEX operations in a total of 160 instances took a time of 93461 CPU seconds. By using LH we spend only %3 of this time to solve same set of problems. Obviously, CPLEX run time increases too fast when compared to the increase in problem size.

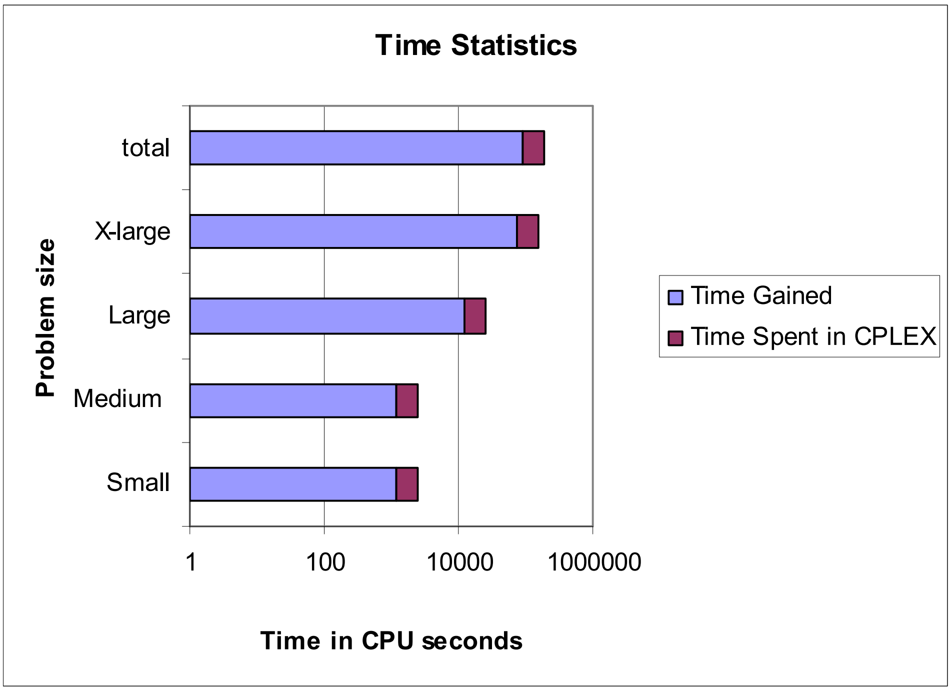


Figure 5-3

Figure 5-3, also shows the comparison of time statistics in a visual manner.

Chapter 6

CONCLUSIONS

In this study, we introduce a new facility location problem that is an extension of capacitated facility location problem with single source constraints (CFLPSS). This problem is applicable to several situations including sub-contractors case and to cases where facilities have very large setup costs. In our problem all variables are binary, so solving it to optimality will be computationally inefficient. Therefore, we aim to devise a heuristic that would solve the problem with a small error margin and with fast computational times.

First, we described our problem and formulated it mathematically. Then, as the problem is the general case of CFLPSS, we focused our research on previous work on CFLPSS. Based on lagrangian heuristics of Holmberg [21] and Sridharan [25], we introduced an improvement of their heuristics that can cope with this general case CFLPSS. In this heuristic, we relaxed demand constraints of the problem in a lagrangian fashion and solved the resulting sub-problem (Z_D) with a general case B&C algorithm, CPLEX. We also devised a primal heuristic to obtain feasible solutions with short duality gaps. The lower bounds obtained from lagrangian relaxation and upper bounds obtained from primal heuristics then embedded into an iterative procedure of a subgradient search. Finally, best feasible solutions were improved through solving SSTP problems.

We tested our model and heuristics on 2 different data sets. The first data set was taken from literature and it was consisted of 70. We tested our heuristics efficiency as CFLPSS solver for these instances. It was found out that for %89 of the cases the solution had less duality gap then %3, %78 of which had less then %1 duality gap. It was shown also that our LH is quite efficient as it has an average run time of 6,56 seconds at most.

The first data set was not suitable to test our LH in terms of our main problem, CFLPSSMM. Therefore, we used a second data set that consisted of 160 instances. These were generated according to a similar procedure as Holmberg et al's [21]. Then we applied both our LH and a direct approach using CPLEX with time limits. It had parallel results with the previous data set, %77 of the solutions had less than %1 duality gap or were better than CPLEX results. Moreover, only %2 of the solutions were more than %3 of CPLEX results and none of the solutions exceeded %4. The average error margin at most was in 160v40 setting and it was %2 and %0,8 when compared to lower bounds of lagrangian relaxation and compared to CPLEX results respectively. On the average duality gap between lower bounds and feasible solutions was %0,8.

In addition of LH being an efficient heuristic in terms of solution quality, we observed that it had also very competitive run times. It was shown that even for large cases as 160v40, run time of heuristics did not exceed 50 seconds on average. It is shown that a significant time gain can be achieved by using LH instead of the general B&C approach of CPLEX. LH solved 160 instances in a time of approximately 3000 seconds, where as the same problems were solved in a time more than 93000 seconds by CPLEX. The time gain was a considerable amount : %97.

Our LH is shown to be an effective tool for solving both CFLPSS and CFLPSSMM problems. By using a knapsack specific B&B code in lagrangian sub-problem, the computational times may be further reduced.

Bibliography

1. Agar, M.C., Salhi S. (1998), "Lagrangian heuristics applied to a variety of large capacitated plant location problems", *Journal of Operational Research Society* **49:** 1072-1084.

2. Akinc, U., Khumawala, B.M. (1977), "An efficient branch and bound algorithm for the capacitated warehouse location problem", *Management Science* **23** (6)**:** 585-594.

3. Baker, B.M. (1982), "Linear relaxations of the capacitated warehouse location problem", *Journal of Operational Research Society* **33:** 475-479.

4. Baker, B.M. (1986), "A partial dual algorithm for the capacitated warehouse location problem", *European Journal of Operational Research,* **23:** 48-56.

5. Beasley J.E. (1993),"Lagrangian heuristic for location problems", *European Journal of Operational Research,* **65:** 383-399.

6. Brandeu, M.L., Chiu, S.S. (1989), "An overview of representative problems in location research", *Management Science* **35** (6)**:** 645-674.

7. Correia, I., Captivo, M.E. (2006), "Bounds for the single source modular capacitated plant location problem", *Computers & Operations Research* **33:** 2991-3003.

8. Cornuejols, G., Sridharan, R., Thizy, J.M. (1991), "A comparison of heuristics and relaxations for the capacitated plant location problem", *European Journal of Operational Research,* **50:** 280-297.

9. Cortinhal, M.J., Captivo, M.E. (2003), "Upper and lower bounds for the single source capacitated location problem", *European Journal of Operational Research,* **151:** 333-351.

10. CPLEX 8.1, User's Manual & Reference Manual

11. Darby-Dowman, K., Lewis, H.S. (1988), "Lagrangian relaxation and the single-source capacitated facility-location problem", *Journal of Operational Research Society* **39** (11)**:** 1035-1040.

12. Daskin. M.S., Melkote, S. (2001), "An integrated model of facility location and transportation network design", *Transportation Research Part A* **35:** 515-538.

13. Daskin. M.S., Melkote, S. (2001), "Capacitated facility location/network design problems", *European Journal of Operational Research,* **1295:** 481-495.

14. Daskin, M., Snyder, L.V, Berger, R.T. (2003) "Facility location in supply chain design", *Working paper*
www.iems.northwestern.edu/images/PDF/WP_03_010.pdf

15. Erlebacher, S.J., Meller, R.D. (1999), "The interaction of location and inventory in designing distribution systems", *IIE Transactions* **32:** 155-166.

16. Erlenkotter, D. (1978), "A dual-based procedure for uncapacitated facility location", *Operations Research* **26** (6)**:** 992-1009.

17. Francis, R.L., Goldstein, J.M. (1973), "Location Theory: A Selective Bibliography". Technical Notes, The University of Florida.

18. Francis, R.L, McGinnis, L.F., White, J.A. (1992), "Facility layout and location: an analytical approach", 2nd Edition. *Prentice Hall* , USA.

19. Harkness, J., ReVelle, C. (2003), "Facility location with increasing production costs", *European Journal of Operational Research,* **145:** 1-13.

20. Hindi, K.S., Pienkosz, K. (1999), "Efficient solution of large scale, single-source, capacitated plant location", *Journal of Operational Research Society* **50** (7)**:** 268-274.

21. Holmberg, K., Rönnqvist, M., Yuan, D. (1997), "An exact algorithm for the capacitated facility location problems with single sourcing", *European Journal of Operational Research,* **113:** 544-559.

22. Sa, G. (1969), "Branch and bound approximate solutions to the capacitated plant location problem", *Operations Research* **17** (6)**:** 1005-1016.

23. Selçuk, B. (2002), "Facility location decisions under vehicle routing considerations", M.S. Thesis, Bilkent University, Ankara, Turkey.

24. Shulman, A. (1991), "An algorithm for solving dynamic capacitated plant location problems with discrete expansion sizes", *Operations Research* **39** (3)**:** 423-436.

25. Sridharan, R. (1991), "A lagrangian heuristic for the capacitated plant location problem with side constraints", *Journal of Operational Research Society* **42** (7)**:** 579-585.

26. Sridharan, R. (1993), "A Lagrangian heuristic for the capacitated plant location problem with single source constraints", *European Journal of Operational Research* **66:** 305-312.

27. Sridharan, R. (1995). "The capacitated plant location problem", *European Journal of Operational Research* **87:** 203-213.

28. Kuehn, A.A., Hamburger, M.J. (1963), "A heuristic program for locating warehouses", *Management Science* **9:** 643-666.

29. Klincewicz, J.G., Luss, H. (1986), "A lagrangian relaxation heuristic for capacitated facility location with single-source constraints", *Journal of Operational Research Society* **37** (5)**:** 495-500.

30. Klose, A., Drexj, A. (2004), "Facility location models for distribution system design", *European Journal of Operational Research* **162:** 4-29.

31. Klose A., Görtz, S. (2006), "A branch and price algorithm for the capacitated facility location problem", *European Journal of Operational Research: In Press.(www.sciencedirect.com)*

32. Mateus, G.R., Bornstein, C.T. (1991), "Dominance criteria for the capacitated warehouse location problem", *Journal of Operational Research Society* **42** (2)**:** 145-149.

33. Nauss, R.M. (1978), "An improved algorithm for the capacitated facility location problem", *Journal of Operational Research Society* **29** (12)**:** 1195-1201.

34. OR Library, http://people.brunel.ac.uk/~mastjjb/jeb/info.html, CFLPSS data of Holmberg [21]

35. Supply Chain Definition, http://www.x-solutions.poet.com/eu/newsevents/glossar

36. Van Roy, T.J. (1986), "A cross decomposition algorithm for capacitated facility location", *Operations Research* **34** (1)**:** 145-162.

37. Wolsey, L.A. (1998), "Integer Programming", *Wiley-Interscience Series in Discrete Mathematics and Optimization,* New York, United States of America.

Appendix

Appendix - 1

Tables of Computational Experiments with Data Set 1

Set	Problems	Demands (N)	Facilities (M)	T.Capacity/T.Demand
1	p1-p12	50	10	1.37-2.06
1	p13-p24	50	10	2.77-3.50
2	p25-p40	150	30	3.03-6.06
3	p41-p55	70-100	10-30	1.52-8.28
4	p56-p71	200	30	1.97-3.95

Table A1-I

Problem	Optimum	LH			Gap	
	Zopt	ZUB	ZLB	Time	Opt	LH
p1	8848	8848	8848	0,16	0,0000	0,0000
p2	7913	7913	7913	1,75	0,0000	0,0000
p3	9314	9314	9313,7	0,11	0,0000	0,0000
p4	10714	10714	10695,9	2,74	0,0000	0,0017
p5	8838	8838	8837,1	0,31	0,0000	0,0000
p6	7777	7777	7776,42	0,13	0,0000	0,0000
p7	9488	9488	9479,39	2,58	0,0000	0,0009
p8	11088	11088	11079,44	2,54	0,0000	0,0008
p9	8462	8477	8453	0,93	0,0018	0,0028
p10	7617	7617	7610	0,78	0,0000	0,0009
p11	8932	8932	8932	0,15	0,0000	0,0000
p12	10132	10132	10114	1,32	0,0000	0,0018
			Average=	1,13	0,0001	0,0007
			Total=	13,50	0,0018	0,0089

Table A1-1a

Problem	Optimum	LH			Gap	
	Zopt	ZUB	ZLB	Time	Opt	LH
p13	8252	8252	8251,82	1,46	0,0000	0,0000
p14	7137	7137	7137	0,44	0,0000	0,0000
p15	8808	8808	8808	0,52	0,0000	0,0000
p16	10408	10435	10382,63	3,75	0,0026	0,0050
p17	8227	8227	8225,93	1,98	0,0000	0,0000
p18	7125	7125	7125	0,3	0,0000	0,0000
p19	8886	8907	8849,14	2,27	0,0024	0,0065
p20	10486	10486	10467,19	2,78	0,0000	0,0018
p21	8068	8068	8067,1	0,28	0,0000	0,0000
p22	7092	7092	7092	0,22	0,0000	0,0000
p23	8746	8746	8740,42	1,33	0,0000	0,0006
p24	10273	10273	10202,98	1,03	0,0000	0,0069
			Average=	1,36	0,0004	0,0017
			Total=	16,36	0,0050	0,0209

Table A1-1b

Problem	Optimum	LH			Gap	
	Zopt	ZUB	ZLB	Time	Opt	LH
p25	11630	11632	11560,25	6,1	0,0002	0,0062
p26	10771	10791	10720,14	5,39	0,0019	0,0066
p27	12322	12373	12189,47	6,32	0,0041	0,0151
p28	13722	13730	13589,68	7	0,0006	0,0103
p29	12371	12391	12313,13	11,29	0,0016	0,0063
p30	11331	11604	11107,99	6,86	0,0241	0,0447
p31	13331	13834	13113,51	7,66	0,0377	0,0549
p32	15331	15351	15109,18	8,58	0,0013	0,0160
p33	11629	11632	11614,8	5,86	0,0003	0,0015
p34	10632	10632	10631,67	1,69	0,0000	0,0000
p35	12232	12232	12231,1	0,65	0,0000	0,0000
p36	13832	13832	13831,54	0,63	0,0000	0,0000
p37	11258	11258	11257,04	2,1	0,0000	0,0000
p38	10551	10551	10550,54	1,22	0,0000	0,0000
p39	11824	11824	11823,27	0,32	0,0000	0,0000
p40	13024	13024	13023,4	0,51	0,0000	0,0000
			Average=	4,51	0,0045	0,0101
			Total=	72,18	0,0718	0,1616

Table A1-2

Problem	Optimum	LH			Gap	
	Zopt	ZUB	ZLB	Time	Opt	LH
p41	6589	6590	6577,88	3,25	0,0002	0,0018
p42	5663	5666	5624,32	3,03	0,0005	0,0074
p43	5214	5214	5213,3	2,12	0,0000	0,0000
p44	7028	7028	7026,11	2,74	0,0000	0,0003
p45	6251	6251	6250	1,48	0,0000	0,0002
p46	5651	5803	5636,3	4,57	0,0269	0,0296
p47	6228	6228	6227,48	0,19	0,0000	0,0000
p48	5596	5596	5583,96	2,48	0,0000	0,0022
p49	5302	5364	5301,98	2,89	0,0117	0,0117
p50	8741	8756	8659,73	4,04	0,0017	0,0111
p51	7414	7481	7265,3	4,82	0,0090	0,0297
p52	9178	9178	9174,48	4,68	0,0000	0,0004
p53	8531	8531	8530,28	0,27	0,0000	0,0000
p54	8777	8777	8776,15	0,32	0,0000	0,0000
p55	7654	7685	7616,23	3,41	0,0041	0,0090
			Average=	2,69	0,0036	0,0069
			Total=	40,29	0,0541	0,1033

Table A1-3

Problem	Optimum	LH			Gap	
	Zopt	ZUB	ZLB	Time	Opt	LH
p56	21103	21331	20950,89	8,39	0,0108	0,0181
p57	26039	26214	25832,31	11,22	0,0067	0,0148
p58	37239	37414	37035,26	11,64	0,0047	0,0102
p59	27282	27556	27113,73	11,22	0,0100	0,0163
p60	20534	20534	20533,24	0,81	0,0000	0,0000
p61	24454	24454	24453,58	2,14	0,0000	0,0000
p62	32643	32919	32385,12	8,63	0,0085	0,0165
p63	25105	25105	25083,89	6,96	0,0000	0,0008
p64	20530	20530	20529,75	0,68	0,0000	0,0000
p65	24445	24445	24445	1,22	0,0000	0,0000
p66	31415	31642	31175,59	7,81	0,0072	0,0150
p67	-	-	-	0	0,0000	0,0000
p68	20538	20538	20537,6	0,59	0,0000	0,0000
p69	24532	24532	24531,3	2,77	0,0000	0,0000
p70	32321	32403	32227,6	15,42	0,0025	0,0054
p71	25540	25540	25539,56	8,84	0,0000	0,0000
			Average=	6,56	0,0034	0,0065
			Total=	98,34	0,0505	0,0972

Table A1-4

Appendix - 2

Tables of Computational Experiments with Data Set 2

Problem Size		
Small	N=40	M=10
Medium	N=65	M=15
Large	N=80	M=20
X-Large	N=160	M=40

Minimum Supply Requirements (in % capacity)	
Low	20%
Medium	30%
High	40%
No	0%

Table A2-i

Distribution of Parameters	
Kj	U(500,800)
Fj	U(500,1500)
Di	U(30,80)
Cij	U(1,4)

*Cij is per 1 unit of demand

Distribution of Parameters

Table A2-ii

Problem	CPLEX		LH			Gap		
	Z*	Time	UB	LB	Time	Z*/UB	UB/LB	Time
p1	6681	6,87	6681	6635,9	3,12	0,0000	0,0068	1,20
p2	6051	0,39	6051	6051,0	0,15	0,0000	0,0000	1,60
p3	6993	7,94	6993	6932,4	0,39	0,0000	0,0087	19,36
p4	6446	1,67	6446	6445,2	0,08	0,0000	0,0000	19,88
p5	5890	0,53	5890	5889,5	0,36	0,0000	0,0000	0,47
p6	6256	0,52	6256	6256,0	0,12	0,0000	0,0000	3,33
p7*	5695	1000,00	5695	5624,2	3,42	0,0000	0,0126	291,40
p8	6362	7,33	6382	6382,0	2,78	0,0031	0,0000	1,64
p9	6082	0,30	6090	6077,3	3,10	0,0013	0,0021	-0,90
p10	6332	2,04	6332	6332,0	0,20	0,0000	0,0000	9,20
Total	62788	1027,59	62816	62625,5	13,72	0,0045	0,0302	347,17
Average	6278,8	102,76	6281,6	6262,5	1,37	0,0004	0,0030	34,72

Table A2-1a Problem size: Small
MM: Low

Problem	CPLEX		LH			Gap		
	Z*	Time	UB	LB	Time	Z*/UB	UB/LB	Time
p1	6575	1,23	6575	6574,4	0,21	0,0000	0,0000	4,86
p2	5907	0,53	5907	5906,2	0,05	0,0000	0,0000	9,60
p3	6174	0,84	6174	6173,8	0,11	0,0000	0,0000	6,64
p4	6578	0,77	6578	6578,0	0,10	0,0000	0,0000	6,70
p5	5418	0,31	5418	5417,3	0,42	0,0000	0,0000	-0,26
p6	5807	0,15	5807	5806,9	0,24	0,0000	0,0000	-0,38
p7	5953	1,13	5953	5952,6	0,07	0,0000	0,0000	15,14
p8	6170	1,11	6170	6169,3	0,10	0,0000	0,0000	10,10
p9	6920	136,33	6920	6831,1	3,24	0,0000	0,0130	41,08
p10	5583	0,70	5585	5568,0	3,68	0,0004	0,0031	-0,81
Total	61085	143,10	61087	60977,6	8,22	0,0004	0,0161	92,67
Average	6108,5	14,31	6108,7	6097,8	0,82	0,0000	0,0016	9,27

Table A2-1b Problem size: Small
MM: Medium

Problem	CPLEX		LH			Gap		
	Z*	Time	UB	LB	Time	Z*/UB	UB/LB	Time
p1	5875	0,21	5875	5874,9	0,22	0,0000	0,0000	-0,05
p2	6590	0,37	6590	6590,0	0,93	0,0000	0,0000	-0,60
p3	6038	21,57	6038	5994,5	2,97	0,0000	0,0073	6,26
p4	6013	1,10	6091	5939,4	4,25	0,0130	0,0255	-0,74
p5	5957	0,49	5957	5956,0	0,11	0,0000	0,0000	3,45
p6	6234	0,42	6234	6233,1	0,75	0,0000	0,0000	-0,44
p7	6433	8,25	6433	6432,0	0,12	0,0000	0,0000	67,75
p8	6814	5,63	6814	6804,7	2,91	0,0000	0,0014	0,93
p9	6468	5,77	6468	6467,6	0,36	0,0000	0,0000	15,03
p10	6657	20,54	6657	6657,0	0,11	0,0000	0,0000	185,73
Total	63079	64,35	63157	62949,1	12,73	0,0130	0,0342	277,33
Average	6307,9	6,44	6315,7	6294,9	1,27	0,0013	0,0034	27,73

Table A2-1c Problem size: Small
MM: High

Problem	CPLEX		LH			Gap		
	Z*	Time	UB	LB	Time	Z*/UB	UB/LB	Time
p1	6249	0,04	6249	6248,5	0,63	0,0000	0,0000	-0,94
p2	6145	0,38	6145	6145,0	0,69	0,0000	0,0000	-0,45
p3	6344	0,27	6344	6344,0	0,10	0,0000	0,0000	1,70
p4	6330	11,33	6351	6177,6	3,69	0,0033	0,0281	2,07
p5	7122	1,67	7122	7122,0	0,26	0,0000	0,0000	5,42
p6	5202	0,03	5202	5201,7	0,06	0,0000	0,0000	-0,50
p7	6844	1,46	6844	6843,3	0,69	0,0000	0,0000	1,12
p8	6345	11,36	6357	6229,9	2,39	0,0019	0,0204	3,75
p9	5619	0,57	5619	5618,0	0,06	0,0000	0,0000	8,50
p10	6358	0,84	6361	6354,4	3,34	0,0005	0,0010	-0,75
Total	62558	27,95	62594	62284,5	11,91	0,0057	0,0495	19,93
Average	6255,8	2,80	6259,4	6228,5	1,19	0,0006	0,0049	1,99

Table A2-1d Problem size: Small
MM: No

Problem	CPLEX		LH			Gap		
	Z*	Time	UB	LB	Time	Z*/UB	UB/LB	Time
p1	8789	1,29	8789	8788,3	2,87	0,0000	0,0000	-0,55
p2	8029	6,35	8038	8013,6	5,53	0,0011	0,0030	0,15
p3	7818	16,23	7818	7718,1	7,77	0,0000	0,0129	1,09
p4	8680	2,76	8711	8667,8	6,88	0,0036	0,0050	-0,60
p5	8363	1,29	8373	8361,2	6,89	0,0012	0,0014	-0,81
p6	8344	0,86	8346	8343,5	6,32	0,0002	0,0003	-0,86
p7	8946	5,12	8946	8945,6	0,80	0,0000	0,0000	5,40
p8	9210	5,18	9210	9209,4	1,70	0,0000	0,0000	2,05
p9	8659	1,60	8683	8647,2	9,62	0,0028	0,0041	-0,83
p10	8369	1,34	8369	8368,9	0,57	0,0000	0,0000	1,35
Total	85207	42,02	85283	85063,6	48,95	0,0089	0,0268	6,38
Average	8520,7	4,20	8528,3	8506,4	4,90	0,0009	0,0027	0,64

Table A2-2a Problem size: Medium
MM: Low

Problem	CPLEX		LH			Gap		
	Z*	Time	UB	LB	Time	Z*/UB	UB/LB	Time
p1	8720	17,73	8720	8719,7	0,60	0,0000	0,0000	28,55
p2	9124	294,57	9190	9100,2	9,97	0,0072	0,0099	28,55
p3	9339	148,46	9513	9284,3	9,35	0,0186	0,0246	14,88
p4	8023	10,73	8028	7980,6	5,12	0,0006	0,0059	1,10
p5	7681	26,44	7684	7679,5	7,55	0,0004	0,0006	2,50
p6	7942	1,90	7942	7941,6	2,55	0,0000	0,0000	-0,25
p7	8985	23,10	9003	8965,3	8,63	0,0020	0,0042	1,68
p8	8546	105,32	8558	8470,0	10,42	0,0014	0,0104	9,11
p9	8443	77,86	8541	8421,7	7,74	0,0116	0,0142	9,06
p10	7901	3,60	7923	7897,9	9,47	0,0028	0,0032	-0,62
Total	84704	709,71	85102	84460,7	71,40	0,0447	0,0730	94,54
Average	8470,4	70,97	8510,2	8446,1	7,14	0,0045	0,0073	9,45

Table A2-2b Problem size: Medium
MM: Medium

Problem	CPLEX		LH			Gap		
	Z*	Time	UB	LB	Time	Z*/UB	UB/LB	Time
p1	8497	3,48	8497	8496,8	4,18	0,0000	0,0000	-0,17
p2	8708	13,10	8860	8696,9	8,08	0,0175	0,0188	0,62
p3	8403	0,86	8403	8402,4	0,89	0,0000	0,0000	-0,03
p4	8651	1,17	8651	8651,0	1,83	0,0000	0,0000	-0,36
p5	8333	4,57	8333	8333,0	2,92	0,0000	0,0000	0,57
p6	9380	60,21	9382	9354,7	11,40	0,0002	0,0029	4,28
p7	8579	14,83	8580	8565,3	9,82	0,0001	0,0017	0,51
p8	8716	10,24	8719	8666,2	9,67	0,0003	0,0061	0,06
p9	8409	0,92	8409	8408,0	0,57	0,0000	0,0000	0,61
p10	7868	2,20	7868	7867,7	5,23	0,0000	0,0000	-0,58
Total	85544	111,58	85702	85442,0	54,59	0,0181	0,0295	5,51
Average	8554,4	11,16	8570,2	8544,2	5,46	0,0018	0,0029	0,55

Table A2-2c Problem size: Medium
MM: High

Problem	CPLEX		LH			Gap		
	Z*	Time	UB	LB	Time	Z*/UB	UB/LB	Time
p1	8559	31,04	8559	8558,1	0,67	0,0000	0,0000	45,33
p2	8638	10,25	8704	8597,3	7,10	0,0076	0,0124	0,44
p3	7690	4,60	7690	7689,6	6,68	0,0000	0,0000	-0,31
p4	8729	271,92	8774	8696,5	6,18	0,0052	0,0089	43,00
p5	8707	1,71	8728	8689,8	7,13	0,0024	0,0044	-0,76
p6	8623	2,51	8752	8611,1	6,35	0,0150	0,0164	-0,60
p7	8105	0,75	8105	8104,3	0,09	0,0000	0,0000	7,33
p8	7766	0,10	7766	7765,1	0,55	0,0000	0,0000	-0,82
p9	8925	22,52	9296	8820,2	7,69	0,0416	0,0539	1,93
p10	8699	156,64	8747	8676,2	4,42	0,0055	0,0082	34,44
Total	84441	502,04	85121	84208,0	46,86	0,0773	0,1042	129,98
Average	8444,1	50,20	8512,1	8420,8	4,69	0,0077	0,0104	13,00

Table A2-2d Problem size: Medium
MM: No

Problem	CPLEX		LH			Gap		
	Z*	Time	UB	LB	Time	Z*/UB	UB/LB	Time
p1	10420	5,69	10420	10420,0	3,37	0,0000	0,0000	0,69
p2	11025	4,75	11035	11023,0	12,63	0,0009	0,0011	-0,62
p3*	11106	1000,00	11150	11043,7	13,66	0,0040	0,0096	72,21
p4*	11139	1000,00	11116	11094,1	14,93	-0,0021	0,0020	65,98
p5	11918	615,40	11957	11879,9	16,53	0,0033	0,0065	36,23
p6	10189	20,91	10235	10138,4	10,91	0,0045	0,0095	0,92
p7	10657	7,90	10657	10656,4	11,65	0,0000	0,0000	-0,32
p8*	10308	1000,00	10310	10230,3	11,65	0,0002	0,0078	84,84
p9*	11222	1000,00	11222	11221,2	0,69	0,0000	0,0000	1448,28
p10	11796	52,98	11804	11794,7	16,11	0,0007	0,0008	2,29
Total	109780	4707,63	109906	109501,5	112,13	0,0115	0,0373	1710,48
Average	10978	470,76	10990,6	10950,2	11,21	0,0011	0,0037	171,05

Table A2-3a Problem size: Large
MM: Low

Problem	CPLEX		LH			Gap		
	Z*	Time	UB	LB	Time	Z*/UB	UB/LB	Time
p1*	10356	1000,00	10356	10306,1	13,69	0,0000	0,0048	72,05
p2	10243	76,52	10251	10188,5	12,13	0,0008	0,0061	5,31
p3	9960	11,30	10040	9940,9	15,24	0,0080	0,0100	-0,26
p4	11370	8,90	11370	11370,0	10,06	0,0000	0,0000	-0,12
p5	10553	147,05	10572	10539,7	12,09	0,0018	0,0031	11,16
p6	11743	6,34	11752	11742,1	11,32	0,0008	0,0008	-0,44
p7	9974	27,04	9974	9974,0	9,65	0,0000	0,0000	1,80
p8	10144	111,01	10150	10131,1	11,34	0,0006	0,0019	8,79
p9	10840	105,48	10850	10763,0	14,36	0,0009	0,0081	6,35
p10	10820	21,16	10820	10820,0	4,94	0,0000	0,0000	3,28
Total	106003	1514,80	106135	105775,3	114,82	0,0129	0,0348	107,92
Average	10600,3	151,48	10613,5	10577,5	11,48	0,0013	0,0035	10,79

Table A2-3b Problem size: Large
MM: Medium

Problem	CPLEX		LH			Gap		
	Z*	Time	UB	LB	Time	Z*/UB	UB/LB	Time
p1	10842	623,74	10868	10768,5	13,88	0,0024	0,0092	43,94
p2	10587	39,61	10587	10586,8	2,62	0,0000	0,0000	14,12
p3	10465	7,08	10465	10464,3	5,56	0,0000	0,0000	0,27
p4*	10969	1000,00	10977	10965,2	10,29	0,0007	0,0011	96,18
p5	10331	20,39	10332	10330,9	13,10	0,0001	0,0001	0,56
p6*	11098	1000,00	11097	11062,9	13,25	-0,0001	0,0031	74,47
p7	11288	8,63	11293	11285,1	9,74	0,0004	0,0007	-0,11
p8	10726	57,47	10726	10725,8	2,64	0,0000	0,0000	20,77
p9	10319	40,28	10367	10299,1	13,92	0,0047	0,0066	1,89
p10	10192	22,94	10195	10189,0	12,21	0,0003	0,0006	0,88
Total	106817	2820,14	106907	106677,6	97,21	0,0085	0,0214	252,97
Average	10681,7	282,01	10690,7	10667,8	9,72	0,0009	0,0021	25,30

Table A2-3c Problem size: Large
MM: High

Problem	CPLEX		LH			Gap		
	Z*	Time	UB	LB	Time	Z*/UB	UB/LB	Time
p1	10951	326,32	10955	10950,8	8,23	0,0004	0,0004	38,65
p2*	11353	1000,00	11338	11337,4	0,40	-0,0013	0,0000	2499,00
p3	10260	6,58	10260	10259,1	0,82	0,0000	0,0000	7,02
p4*	11863	1000,00	11812	11692,0	7,17	-0,0043	0,0103	138,47
p5*	10817	1000,00	10814	10813,8	2,78	-0,0003	0,0000	358,71
p6	10440	18,93	10440	10439,1	10,29	0,0000	0,0000	0,84
p7	9449	25,42	9454	9442,3	8,24	0,0005	0,0012	2,08
p8	10012	93,03	10033	10002,3	10,49	0,0021	0,0031	7,87
p9	10177	6,07	10177	10177,0	2,73	0,0000	0,0000	1,22
p10	11011	2,18	11011	10990,0	8,19	0,0000	0,0019	-0,73
Total	106333	3478,53	106294	106103,8	59,34	-0,0029	0,0169	3053,14
Average	10633,3	347,85	10629,4	10610,4	5,93	-0,0003	0,0017	305,31

Table A2-3d Problem size: Large
MM: No

Problem	CPLEX		LH			Gap		
	Z*	Time	UB	LB	Time	Z*/UB	UB/LB	Time
p1*	20437	2000,00	20923	20270,1	56,56	0,0238	0,0322	34,36
p2*	20261	2000,00	20208	20140,2	50,80	-0,0026	0,0034	38,37
p3*	19747	2000,00	19650	19124,3	55,30	-0,0049	0,0275	35,17
p4	19206	2000,00	19703	19167,6	53,78	0,0259	0,0279	36,19
p5*	18725	1262,57	19272	18722,1	51,65	0,0292	0,0294	23,44
p6*	19602	2000,00	19645	19593,2	50,54	0,0022	0,0026	38,57
p7*	20731	2000,00	21081	20403,7	54,89	0,0169	0,0332	35,44
p8*	20790	2000,00	20759	20144,4	56,81	-0,0015	0,0305	34,21
p9*	19256	2000,00	19694	19117,0	56,06	0,0227	0,0302	34,68
p10*	19050	2000,00	18987	18856,0	61,19	-0,0033	0,0070	31,69
Total	197805	19262,57	199922	195538,4	547,58	0,1084	0,2239	342,11
Average	19780,5	1926,26	19992,2	19553,8	54,76	0,0108	0,0224	34,21

Table A2-4a Problem size: X-large
MM: Low

Problem	CPLEX		LH			Gap		
	Z*	Time	UB	LB	Time	Z*/UB	UB/LB	Time
p1*	21035	2000,00	21022	20904,9	68,62	-0,0006	0,0056	28,15
p2*	19417	2000,00	20062	19330,9	54,27	0,0332	0,0378	35,85
p3*	19755	2000,00	20429	19661,5	53,39	0,0341	0,0390	36,46
p4*	20034	2000,00	20108	19640,5	51,71	0,0037	0,0238	37,68
p5*	19387	2000,00	19650	19085,0	54,08	0,0136	0,0296	35,98
p6*	19179	2000,00	19276	18813,3	65,73	0,0051	0,0246	29,43
p7*	18744	2000,00	19027	18482,2	55,74	0,0151	0,0295	34,88
p8*	20028	2000,00	20614	19927,0	54,85	0,0293	0,0345	35,46
p9*	20086	2000,00	20024	19823,8	54,17	-0,0031	0,0101	35,92
p10*	18553	2000,00	18540	18535,5	55,96	-0,0007	0,0002	34,74
Total	196218	20000,00	198752	194204,6	568,52	0,1296	0,2348	344,55
Average	19621,8	2000,00	19875,2	19420,5	56,85	0,0130	0,0235	34,46

Table A2-4b Problem size: X-large
MM: Medium

Problem	CPLEX		LH			Gap		
	Z*	Time	UB	LB	Time	Z*/UB	UB/LB	Time
p1*	20066	2000,00	20177	19760,9	70,18	0,0055	0,0211	27,50
p2*	18470	2000,00	18494	18450,7	55,99	0,0013	0,0023	34,72
p3*	19126	2000,00	19138	19068,0	67,03	0,0006	0,0037	28,84
p4*	19362	2000,00	19304	19040,8	63,99	-0,0030	0,0138	30,25
p5*	19606	2000,00	20032	19447,2	55,11	0,0217	0,0301	35,29
p6*	20242	2000,00	20583	20033,5	53,91	0,0168	0,0274	36,10
p7*	19118	2000,00	19129	18987,6	64,62	0,0006	0,0074	29,95
p8*	20176	2000,00	20034	19421,7	71,97	-0,0070	0,0315	26,79
p9*	18370	2000,00	18370	18369,6	35,73	0,0000	0,0000	54,98
p10*	21867	2000,00	22560	21694,0	56,98	0,0317	0,0399	34,10
Total	196403	20000,00	197821	194274,0	595,51	0,0683	0,1773	338,52
Average	19640,3	2000,00	19782,1	19427,4	59,55	0,0068	0,0177	33,85

Table A2-4c Problem size: X-large
MM: High

Problem	CPLEX		LH			Gap		
	Z*	Time	UB	LB	Time	Z*/UB	UB/LB	Time
p1*	20630	2000,00	20621	19964,6	29,07	-0,0004	0,0329	67,80
p2*	19229	2000,00	19314	18854,8	28,27	0,0044	0,0244	69,75
p3*	19072	2000,00	18989	18808,1	27,07	-0,0044	0,0096	72,88
p4*	18907	2000,00	18819	18681,5	26,53	-0,0047	0,0074	74,39
p5	18050	1049,20	18109	18043,0	23,60	0,0033	0,0037	43,46
p6*	19577	2000,00	19920	19343,2	27,31	0,0175	0,0298	72,23
p7*	20200	2000,00	20187	20006,7	29,60	-0,0006	0,0090	66,57
p8*	18623	2000,00	18634	18527,9	22,90	0,0006	0,0057	86,34
p9*	20031	2000,00	20098	20021,4	21,36	0,0033	0,0038	92,63
p10*	18760	2000,00	18749	18711,9	24,73	-0,0006	0,0020	79,87
Total	193079	19049,20	193440	190962,9	260,44	0,0185	0,1282	725,92
Average	19307,9	1904,92	19344	19096,3	26,04	0,0018	0,0128	72,59

Table A2-4d Problem size: X-large
MM: No

Appendix – 3

Abbreviations for Common Terminology

B&B: Branch and Bound

B&C: Branch and Cut

CFLP: Capacitated Facility Location Problems

CFLPSS: Capacitated Facility Location Problem with Single Source constraints

CFLPSSMM: Facility Location Problem with Single Source constraints and Minimum Supply Requirements

FLP: Facility Location Problems

IP: Integer Programming

MIP: Mixed Integer Programming

Surrogate constraint: A constraint that imposes total capacity of open plants must exceed the total demand. Supplementary, used as a valid inequality to increase lower bounds

Strong Formulation: Indicates that the corresponding FLP includes $x_{ij} \leq y_j$ type of constraints. Supplementary, used as a valid inequality to increase lower bounds

UFLP: Uncapacitated Facility Location Problems

Z_C: The lagrangian sub-problem where the “Capacity” constraint is relaxed in lagrangian fashion

Z_D: The lagrangian sub-problem where the “Demand” constraint is relaxed in lagrangian fashion