VENDOR LOCATION PROBLEM

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

By
Yüce Çınar
July, 2009

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. Hande Yaman(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. Ayşegül Altın

Approved for the Institute of Engineering and Science:

Prof. Mehmet B. Baray
Director of the Institute

ABSTRACT

VENDOR LOCATION PROBLEM

Yüce Çınar
M.S. in Industrial Engineering
Supervisor: Assoc. Prof. Hande Yaman
July, 2009

In this study, we aim to design a distribution system with the following components: the location of vendors, the number of vendors, the service region of the vendors, the number of vehicles and workers, and the assignment of demand points to these vendors and vehicles. We define our problem as a two-level capacitated discrete facility location problem with minimum profit constraints and call it *Vendor Location Problem.* In order to formulate the problem, two different objective functions are used: vendors's profit maximization and maximization of the demand covered. Integer linear programs for these two versions of the problem are formulated. Valid inequalities are used to strengthen the upper bounds. Finally, the performance of these models with different parameters are compared in terms of linear programming relaxation gap, optimality gap, CPU time, and the number of opened nodes for four different types of instances: instances with demand and profit which are independent of distance; profit function of distance; demand function of distance; demand and profit function of distance.

Keywords: Vendor location, two-level capacitated facility location problem, distribution system, minimum profit constraint, valid inequalities.

ÖZET

BAYİ YER SEÇİMİ PROBLEMİ

Yüce Çınar
Endüstri Mühendisliği, Yüksek Lisans
Tez Yöneticisi: Doç. Dr. Hande Yaman
Temmuz, 2009

Bu tez çalışması; bir firmanın bayileri için yer seçimi, bayi sayısı, bayi çalışan ve araç sayıları ile müşterilerin bayilere ve araçlara atanması kararlarını içeren bir dağıtım sistemi tasarımını amaçlamıştır. Problem literatürdeki iki aşamalı ve kapasiteli kesikli tesis yerleşim problemi olarak tanımlanmış ve *Bayi Yer Seçimi Problemi* olarak adlandırılmıştır. Bayi karını ve servis edilen talebi enbüyültmek olmak üzere iki farklı amaç fonksiyonu tanımlanmış ve bu iki problem için doğrusal tamsayı programları sunulmuştur. Geçerli eşitsizlikler eklenerek problemlerin üst limitleri düşürülmüş ve problemler çözümlenmiştir. Ayrıca, sayısal deneyler için dört farklı örnek grubu oluşturulmuştur: uzaklıktan bağımsız kar ve talep fonksiyonlarını; uzaklığa bağlı kar fonksiyonunu; uzaklığa bağlı talep fonksiyonunu; uzaklığa bağlı kar ve talep fonksiyonlarını içeren örnekler. Modeller oluşturulan örnek gruplarında parametreleri değiştirilerek doğrusal gevşetme farkı, eniyilik farkı, CPU süresi ve açılan düğüm sayısı bakımından karsılaştırılmıştır.

Anahtar sözcükler: Bayi yer seçimi, iki aşamalı kapasiteli tesis yerleşim problemi, dağıtım sistemi, minimum kar kısıtı, geçerli eşitsizlikler.

Acknowledgement

I would like to express my deepest and most sincere gratitude to my advisor and mentor, Assoc. Prof. Hande Yaman for her invaluable trust, guidance, encouragement and motivation during my graduate study. She has been supervising me with everlasting patience and interest from the beginning to the end.

I would like to express my special thanks to Prof. Mustafa Ç. Pınar and Asst. Prof. Ayşegül Altın for accepting to read and review this thesis and for their suggestions. Their remarks and recommendations have been very helpful.

I would like to thank Prof. İhsan Sabuncuoğlu and all other faculty members of the Department of Industrial Engineering at Bilkent University for their intellectual contributions and encouragement during both my undergraduate and graduate academic life.

I would like to thank to my friends Tuğçe Akbaş and Didem Batur for all wonderful time we cherished over the last three years. Without their academic and most importantly morale support things would have been much more difficult and boring. I wish to thank Gökay Eron and Sıtkı Gülten for their camaraderie and helpfulness.

I would like to thank Barış Noyan and Bige Ayan Ulusoy for their understanding and valuable comments on my career.

I want to express my heartily thanks to Cemil Baysal who have always supported and encouraged me and sincerely gave all the help and everlasting love.

Last but not least, I would like to express my gratitute to my mother Hümeyra Çınar, my sister Özge Çınar and my nephew Çınar Demir for their trust and motivation during my study. My father Atila Çınar deserves special mention for his everlasting support and love. I owe them a lot for every success I have in my life.

Contents

List of Figures

List of Tables

Chapter 1

INTRODUCTION

Recently, the importance of customer satisfaction has increased dramatically for the firms in the service sector. One of the key factor that has a big impact on customer satisfaction is the service time. Therefore, these firms pay more attention to production, logistics, and distribution management to create a competitive advantage over their competitors. Hence, vendor location decisions become an important component as a management concept that affect the service time excessively. Moreover, the design of distribution system is typically a costly and time-sensitive project. The main factors to be determined before locating facilities are the area of the location, the number of facilities, and capacity specifications.

In this thesis, we aim to design the distribution system for firms, which sell their products through vendors. This system design problem includes the decision on the location of their vendors, the service region of each vendor, the number of vehicles and workers for each vendor, and the assignments of customers to these vendors and vehicles. Customer/ demand point and facility/ vendor are used interchangeably hereafter.

1.1 Motivation

We consider a discrete facility location problem encountered by one of the major demijohn water sellers in Ankara. A few years ago, the company decided to introduce its own brand and wanted to locate a number of vendors and to determine disjoint sales regions for its vendors in a way that each vendor can achieve at least a minimum level of profit.

The sales of demijohn water works on a kind of membership of customers. Every brand has its own bottles. A customer who wants to buy the product of a given brand is charged for the first bottle. After the first purchase, the empty bottle is changed with a full bottle, and the customer is only charged as much as the price of the water. This discourages customers from switching frequently from a brand to another.

Before the company introduced its product, a detailed market analysis has been conducted to determine the criteria that the customers use in deciding which brand to buy and to forecast the demand for the new product. It has been observed that the customers valued the most, the quality of the water (taste, hygiene, chemical composition etc.) and the quality of the service. The quality of the service was strongly related to service times and the satisfaction was affected by the presence of competitors in the same region who could provide shorter service times.

It was concluded that the number of customers that the company could attract from a given region depended highly on the distance between the region and the vendor to which this region would be assigned to and the distances between the region and the vendors of competitor brands. Hence opening vendors at many locations could increase the market share of the company. However this could result in vendors with insufficient sales to achieve at least a minimum level of profit.

The distribution system of the company is depicted in Figure 1.1.

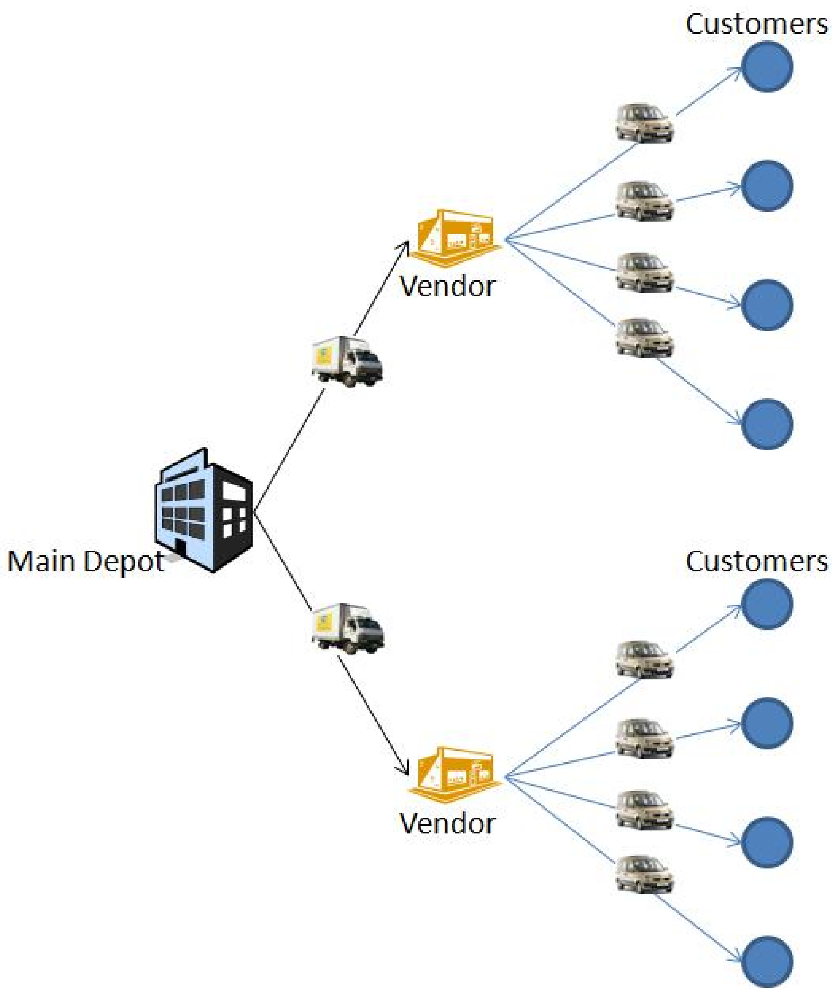

Figure 1.1: The Distribution System

1.2 Problem Definition

We aim to establish a distribution system for companies by deciding where to locate their vendors, the number of vehicles for each vendor as well as the assignment of customers to these vendors and vehicles.

We define the *Vendor Location Problem (VLP)* as follows. We are given a set of demand points which correspond to population zones and a set of possible locations for vendors. For each vendor, there is a maximum number of vehicles that this vendor can use. We are given the fixed cost of operating a vendor office (rent, insurance, salaries of employees at office etc.) at a given location and the cost (including the salary of the driver) and capacity of a vehicle.

For a given demand point, there is a set of eligible vendors that can serve this demand point. Demands of demand points change according to the proximity to vendors and also the proximity of other brands' vendors in the region. Hence, the demand and the profit (sales revenue minus the transportation cost) a demand point generates depends on the vendor that serves it.

Now, the *VLP* is to locate a given number of vendors and to assign each demand point to at most one vehicle of an eligible vendor such that capacities of vehicles are not exceeded and each vendor achieves a minimum level of profit. We consider two objective functions. In *ProfitVLP*, the aim is to maximize the total profit and in *CoverageVLP*, the aim is to maximize the coverage, i.e., the total demand served.

1.3 Contribution

In this study, we introduce two new two-level facility location problems, namely *ProfitVLP* and *CoverageVLP*, that are motivated by a real life problem. Different from the classical facility location problems, here we have minimum profit constraints for open facilities and capacity constraints for their vehicles. We investigate the computational complexity of these problems and prove that they are strongly NP-hard. We propose integer programming formulations, valid inequalities and extra constraints to be able to use the cutting planes of off-the-shelf integer programming solvers. We report the outcomes of a computational study where we used four types of instances which differ in their demand and profit functions. We investigate the effect of valid inequalities on linear programming relaxation bounds and solution times for these different types of instances. Finally, we analyze the optimal solutions of *ProfitVLP* and *CoverageVLP* and report how the differences in demand and profit functions effect the locations of facilities and their service regions for an example problem.

1.4 Contents

The remainder of the thesis is organized as follows:

In Chapter 2, we give information on two companies that use vendors for sales.

In Chapter 3, we provide a review of the literature in facility location problems by comparing our problem with these problems.

In Chapter 4, we formally define our problem and then propose an integer linear program to solve the problem exactly.

In Chapter 5, we derive some valid inequalities to strengthen the models presented in Chapter 4.

In Chapter 6, we will present four types of instances. Experimental results related with valid inequalities are given and discussed by comparing the models with each other.

In Chapter 7, we conclude the thesis by giving an overall summary of our contribution to the existing literature and list some possible future research directions.

Chapter 2

VENDOR SYSTEM IN TURKEY

We have interviewed two different firms which sell their products through vendors. The first company is in LPG (Liquefied Petroleum Gas) cylinder market in Turkey. The other one is a beverage company, who produces 19 L HOD-Demijohn water. Since we are not allowed to use their brand names, we call them as Company X and Y, respectively. In the following two sections, we give some general information about these companies and their vendor systems.

2.1 LPG Company

LPG cylinder companies sell their products through vendors. Distribution network is the most important issue to attract customers and meet customer satisfaction for this kind of companies.

The distribution network is composed of filling facilities and vendors. Vendors prefer managing the logistics by their own, although Company X provides free logistics. The reason behind it is that vendors want to order the products in the specified amount and time based on their needs rather than the fixed amount and time the distribution centers determine. The vendors close to the filling facilities do not tend to keep inventory.

The distribution of the products to the customers depends on the type of customer. Customers of Company X can be categorized into three segments as individual subscribers, corporate customers, and the customers of secondary vendors. Individual and corporate customers make a phone call to make an order and the vendors bring their products via their own vehicles to replace the empty LPG cylinders with full ones. However, secondary vendors meet the demand of people in villages or suburbs without taking any order. Besides, in suburbs, the vehicles go around with the company's jingle to capture the consumer.

To locate the vendors, Company X does not take into account the distance between vendors. Therefore, customers sometimes complain about the imbalance of delivery time of each vendor. Since they have no region division for the vendors, customers are free to choose their own vendors. Sometimes, this leads to long delivery times and high transportation costs.

The most important factor affecting the delivery time is the number of vehicles. The number of vehicles each vendor has is not determined by company X and it changes from region to region. As vendors patronize more customers, the need for an extra vehicle occurs. Vendors assign one more vehicle as a result of increase in the complaints from customers about the delivery time. The aim of Company X is to minimize the number of vehicles without decreasing the service quality.

The customers mainly make their decisions for which brand to choose according to three criteria: reliability that comes from the name of the brand, price, and delivery time. The aim of the company is to keep the lead time of delivery of products below half an hour, but it changes between 20 and 60 minutes due to the reasons mentioned above.

If a vendor cannot compensate its costs and make enough profit due to decrease in demand, then it has to be merged with one of the neighboring vendors.

It may happen that Company X locates its vendors in the same area and this may create serious risk in terms of vendor's profitability and customer satisfaction. This can cause a vendor to be out of business since it is not possible to cover the costs. In 2008, seven vendors closed their businesses, and three vendors were merged to compensate the costs.

2.2 Demijohn Water Company

One of the most proper example for selling products through vendors is demijohn water companies. In Turkey, there are more than 400 brands in the demijohn sector. To understand the vendor system, we interviewed one of these brands, which we call Company Y.

The distribution network of Company Y has distribution centers in cities who supply demijohns for vendors. There are two distribution centers in Ankara: the east region distribution center and the west region distribution center. Each of these centers provides supply to 17 vendors. The separation is based on the amount of demand and region.

These distribution centers order demijohn waters to the company once a day. The delivery time for demijohns for reaching the distribution centers from the factory is 24 hours. Vendors give their orders to the distribution centers at the end of the working day to have the products at the beginning of the next day. Generally, the distribution center is responsible for logistics of the demijohns to the vendors, unless the vendor is willing to receive from the stock warehouse thanks to the proximity. The west distribution center has 4 trucks with capacities 100, 300, 450, and 700 demijohns. Most of the time, one trip per truck per day is enough, if necessary they assign the trucks for a second trip through a day, starting with the smallest truck. Distribution centers have the safety stock which is enough for a daily demand. The sale of a distribution center is 2300- 2500 demijohns per day on the average. It can change in the range of 10% and 15%.

Vendors are responsible to deliver products to customers via their own vehicles. The process of receiving an order from a customer starts when the customer calls the responsible vendor to leave a message of his/her request to the vendor's computer. Then, the request is directed to the deliveryman who is responsible for that region. The deliveryman brings the full bottle to the customer's adress and takes the empty bottle to take it back to the vendor.

The service regions of the vendors are strictly separated from each other and the vendors are prohibited to serve in an other vendor's district. Company decides about the vendor locations with respect to the demands of the regions. First, they choose elite districts to serve and then they decide about the remaining regions.

Moreover, the region of the each vendor's vehicles is also determined and even if the delivery takes less time, customers of a deliveryman cannot be served by another deliveryman working for the same vendor. To shorten the delivery time and to minimize the transportation cost, vehicles assigned to each part of the vendor's region have specific points to wait in their service area. They are not allowed to return back to their vendors during the day before finishing all demijohns loaded on their vehicles.

Another important issue is to decide about the number of vehicles since it affects the delivery time. In the demijohn water sector, each vendor has a different number of vehicles. Regarding that shipping 50-60 demijohns in a day is ideal and more than 80-90 can be shipped by a vehicle, the vendor starts with a vehicle at first. As the vendor patronizes more customers, they augment the number of vehicles in order to have acceptable delivery time.

One of the distribution centers of Company Y has vendors with 1 to 6 vehicles that can be loaded 3 times and accomplish at most 3 tours in a day. The average number of vendor's vehicles is 3. If a vendor cannot afford to buy more vehicles and demand cannot be met within reasonable service times, the need for an extra vendor in that region occurs. This is called "vendor split". Each vendor has to compensate its cost and to continue his business. If a customer is far from the other customers of the vendor, advertisements or promotions are applied to capture more customers at that region to cover the costs. The vendor does not accept the demand if there is still not enough demand in that direction to compensate its own cost.

Although the number of vehicles and delivery times are crucial for vendors to attract the customers, Company Y does not control vendors in terms of delivery time, unless customers deliver a complaint to the company. However, the sales of vendors are inspected and necessary precautions are taken if any loss occurs. In addition to this, Company Y follows the shifts to other brands and tries to get information which other brands customers prefer to Company Y.

Apart from the delivery time, the reasons customers choose the specific brand of demijohn water are reliability of the brand, price, taste, and some emotional factors. For example, the deliveryman has an affect on the process of deciding the demijohn water to buy. They are strictly prohibited to enter customer's house and must be neat. Moreover, customers in Turkey are not tightly coupled with a brand. The regions with well-educated customers have steady sales. However, the demand of less-educated customers is more sensitive due to promotions, free-of-charge exchange demijohns or campaigns. The importance given to price differs according to region.

Chapter 3

LITERATURE SURVEY

There is a wide variety of location problems in the literature. Generally, distribution, transportation and telecommunication are the most important areas for facility location problems. The need for locating facilities arises both in private and public sector. In private sector, industrial firms, retail facilities or banks have to locate their facilities and for the latter one, government agencies decide on the location of schools, hospitals, fire stations, and ambulances.

Location Problems were first introduced in 1909 by Alfred Weber [19] who studied the problem of locating a warehouse in the plane on which the customers are spatially distributed with the objective of minimizing the total walking distance of customers to the facility. More realistic models and algorithms were introduced in the mid-1960s.

Facility Location Problems can be classified regarding various criteria. Klose and Drexl [14] classified facility location problems using the following criteria:

1. The space of location designs

   Location problems are divided into two groups according to the space of d-dimensional real space and network location space. Both two groups are subdivided into continuous and discrete location problems. In location problems with continuous space, facilities can be located anywhere in the space or on the network. However, facilities are sited at the points in a finite set in discrete location problems.

ReVelle and Eiselt [15] stated in their survey that distances in real spaces are often calculated using metric Minkowski distances. The most focused distance types are Rectilinear distances, Euclidean metric and Chebyshev metric.

In network location problems, shorthest path is the method for calculating the distance between nodes which presented by Dijkstra [9].

Our problem is a discrete location problem in 2-dimensional real space and we use Euclidean metric to compute the distances between facilities and customers.

2. Classes of location objectives

   Eiselt and Laporte [10] examined different objective functions for location models. They categorized objective functions into three groups: pull, push and balance objectives. First, if the aim is to locate facilites close to the customers, it is a pull objective. Minisum, maximum capture, minimax and covering problems are the major classes of problems of pull objectives. When the aim is to maximize sales, revenue and customers captured, maximum capture objectives arise. If the issue is to minimize the maximum distance, minimax objectives are used. Max cover and min (cost) cover problems are the two version of covering objectives. In max cover problems, the idea is that facilities are located to maximize the demand captured with a fixed number of facilities. Min (cost) cover problems aim to cover the whole demand with a variable number of facilites by satisfying the distance constraint.

   Push objectives aims to locate undesirable facilities. Finally, balance objectives try to balance distances between facilites and customers, i.e., minimize the variability of the distribution of distances.

   Generally, public sector aims to increase the accessibility of facilities e.g., minimizing the maximum distance between facilities and customers. However, private sector chooses to maximize profits or minimize cost. We define two objective functions including both two groups for our problem: maximizing profit and maximizing coverage.

3. Product type

   Problems can include single-product or multi-product. In single-product models, there is a homogeneous product which does not differ in terms of cost, quality, capacity or demand attributes. All products are homogeneous in our problem.

4. Demand type

   In location problems, demand can be classified as elastic and inelastic demand. In inelastic demand, spatial decision does not influence demand. On the contrary, if demand is elastic, it may change according to proximity. Elastic demand is usually a component of competitive facility location problems.

   In Competitive location, private sectors' organizations struggle to be close to the customers in order to attract them to their retailers. Characteristics of this problem are various and one of the important component is objective function which is generally based on the utility function. Aboolian et al. [2] suggested a spatial interaction model with multiple facilities, elastic concave demand and multiple design characteristics for competitive location problem. They solved the model for a real life example to locate a set of retail facilities in Toronto, Canada. They used Tangent- Line Approximation (TLA) by adopting the piecewise linear function to linearize the nonlinear concave model for medium-size instances. They also developed an ascent heuristic for larger problems.

   Berman and Drezner [4] studied the multiple facility location problem on a network. They define stochastic demand function as distance dependent, that is to say, it decreases as distance increases. Their objective function is to maximize the demand. They developed heuristic algorithms that are ascent algorithm, tabu search and simulated annealing and concluded that the best approach is simulated annealing.

We model both the demand and the profit as a function of distance. In contrast to these studies mentioned above, we model demand using a piecewise linear function.

5. Planning period and data type

   In facility location problems, static and dynamic models are studied in the literature. Static models optimize the system for a time period. In dynamic models, multiple periods are considered, data varies over time and it is possible to relocate the system components in the given planning horizon. Our problem is a static location problem.

   In terms of certainty, static models can be divided into deterministic and probabilistic models. Deterministic models ignore the uncertainty. However, probabilistic models' data is not known with certainty.

   Brotcorne et al. [5] worked on ambulance location and relocation models. After they mentioned how the emergency services operate, they presented static and dynamic models for ambulance location problems. They concluded that fast heuristics and sufficient computing power make the dynamic models useful in real life. The Location Set Covering Model (LSCM) of Toregas et al. [18] aimed to minimize the number of ambulances so as to cover all demand points. The objective of Maximal covering location problem (MCLP) studied by Church and ReVelle [7] is to maximize coverage with limited resource available.

6. Routing

   ReVelle and Laporte [16] presented Location Routing Problems considered as plant location problems with spatial interaction. These models simultaneously locate facilities and construct routes of delivery and/or collection. There are the Median Tour (MTP) and Covering Tour Problems (CTP), the Newspaper Delivery Problem (NDP) and Multiple Tour Plant Location Problems (MTLP) in this category. The Median Tour Problem introduced by Current and Schilling [8] is the extension of the Generalized Traveling Salesman Problem. Its objective is to minimize both the length of the Hamiltonian tour among facilities and the sum of radial distances between remaining sites and the closest facility. Current and Schilling [8] works on Covering Tour Problem which is a version of MTP. Mailbox location is the application of MTP and CTP.

In the Newspaper Delivery Problem, Jacobsen and Madsen [13] aimed to minimize the total length of all tours. Primary tours through a subset of sites and secondary tours associating sites to the primary tours are computed.

Finally, Multiple Tour Plant Location Problem corresponds the NDP, but there are no tours between facilities and there is a fixed charge for opening facilities.

Although our problem includes delivery/collection process, we cannot construct a route for vehicles as in the case of MTP, CTP, NDP and MTLP, since customer's order triggers the delivery/collection process.

7. Capacity constraints

   If the model has no capacity constraints, it is called uncapacitated facility location problem (UFLP). If facilities have capacity constraints, then the problem is called capacitated facility location problem (CFLP).

   Models with capacity constraints are separated into two groups: single-source and multiple-source. In capacitated facility location problems with single-source, each customer has to be served by only one facility.

   In the literature, one of the common way of solving CFLP is to use Lagrangian Heuristics. Holmberg et al. [12] suggested a Lagrangian Heuristic including a strong primal heuristic and a branch-and-bound for CFLP with single sourcing (SSCFLP). They use subgradient optimization in Lagrangian Heuristic and repeated matching in primal heuristic. To relax the set of constraints, they chose assignment constraints, so they worked on knapsack problems. They concluded that Primal heuristic with Lagrangian relaxation is a very efficient method since Lagrangian relaxation provides strong lower bounds and primal heuristic finds optimal or near optimal solutions quickly.

In our problem, vehicles of vendors have capacity limits, so our problem is capacitated.

Albareda et al. [3] introduced a new problem called Capacity and Distance Constrained Plant Location Problem (CDCPLP) which is an extension of discrete capacitated plant location problem. They propose mathematical formulations and a solution technique for this problem. Their problem has the following properties:

- After customers are assigned to facilities, each customer is also assigned to a vehicle.
- There are plant capacities and upper bounds on the total distance traveled by each vehicle.
- Demands are not divisible.

They proposed alternative mathematical models that minimize the total cost: the fixed cost for opening plants, the vehicle utilization cost and the assignment cost. They add some inequalities to their first alternative to avoid symmetries that arise since vehicles are identical.

In the first model, they suggested a bilevel model that minimizes required the vehicles to satisfy the assigned customer demands by separating the problem to Bin Packing Problems.

Second, they proposed a relaxed model for CDCPLP to generate good lower bounds by changing the capacity constraints with surrogate aggregated capacity constraints and adding valid inequalities. They improve tabu search based heuristic with three levels of search: plant-level, assignment level and packing-level. The results show that tabu search heuristic provides optimal or near optimal solutions within reasonable computational times.

Our problem is related with CDCPLP in many points. VLP has similar properties mentioned above except the second item. In our problem, not facilities but vehicles have capacity limits and we do not set upper bounds on total distances traveled. Besides, we have an additional constraint which provides the minimum profit of each vendor. Although we have no constraint on total distance traveled, demands are decreasing as distance between vendors and demand points is increasing in VLP. Furthermore, we set a distance restriction for each vendor that is allowed to serve a customer.

8. Hierarchical stages

   Hierarchical location problems occur in many areas such as health care, industrial and telecommunication network contexts. In industrial context, goods start to move from manufacturing plants to warehouses and from warehouses to demand points. Although supply chain concept comes up at this point, the difference between supply chain and location problems is that primary consideration is to focus on the design and secondary issue is operation in location problems like as in hierarchical location problems.

   In hierarchical facility location problems, there are k levels representing the different type of facilites having interaction. Şahin and Süral [17] reviewed the hierarchical facility location problems. First, they focused on four attributes of this type of problems: flow-pattern, service varieties, spatial configuration and objective. According to these attributes, they mention the real life applications of hierarchical facility location problems such as healthcare system, solid waste management system, production-distribution system, education system, emergency medical service (EMS) system and telecommunication networks. They give mathematical formulations of these problems based on above attributes and solution methods.

   Our problem can be seen as a hierarchial facility location problem where customers are level 0, vehicles are level 1 and vendors are level 2. According to the attributes they defined, our problem is single-flow, since a customer to be served by the highest-level facility goes to a level 1 first and then passes through level 2 which is a vendor. Moreover, our problem is non-nested in terms of service varieties, since facilities at each level offer different services. According to spatial configuration, our system is coherent, because each vehicle belongs to a vendor.

   Another study about hierarchical facility location problem was conducted by Aardal et al. [1]. They studied the two-level uncapacitated facility location (TUFL) problem. After they improved for multi-commodity flow formulations of TUFL, they compared them with one-level uncapacitated facility location problem. They presented new families of facets and valid inequalities for TUFL. They discussed useful inequalities for computational purposes for alternative models they developed.

Moreover, Chardaire et al. [6] also studied hierarchical facility location problem. They presented upper and lower bounds for the two-level simple plant location problem. They characterized their problem for two-level concentrator access network in telecommunications industry. First, they introduce a simplified version of the two-level simple plant location problem. They assumed that there is no capacity constraint and all concentrators are of the same type. They developed an effective simulated annealing algorithm for this model to improve some of the upper bounds of Lagrangian relaxation algorithm. Then, they presented improved model formulation for the two-level simple plant location problem. Although both formulations' linear programming relaxation have the same optimal value, improved formulation was tightened using a family of polyhedral cuts that define facets of the convex hull of integer solutions.

Our formulations are an extension of the formulations presented in the two papers mentioned above. We have additional constraints which are capacity and minimum profit.

Chapter 4

MODEL DEVELOPMENT AND COMPLEXITY

In this chapter, we introduce the notation and then present formulations for our problem. Then we prove that both problems are strongly NP-hard.

4.1 Notation and Parameters

Let I be the set of demand points and J be the set of possible locations for vendors. For a demand point $i \in I$, J_i is the set of vendors that can serve i. In our problem, we define J_i to be the set of vendors whose travel time i does not exceed a given bound. We also define $I_j = \{i \in I : j \in J_i\}$ for $j \in J$.

We denote with f_j the fixed cost of the vendor office and with v_j the fixed cost of a vehicle for a vendor located at $j \in J$. We define ρ_{min} to be the minimum profit a vendor should achieve.

We denote with p the number of vendors to be located. The vendor at location $j \in J$ may have up to k_j^{max} vehicles. Let $K_j = \{1, \ldots, k_j^{max}\}$ for $j \in J$. The capacity of a vehicle is equal to γ.

Demand point $i \in I$ has demand q_{ij} and generates profit ρ_{ij} if it is served by the vendor at location $j \in J_i$.

4.2 Decision Variables

After defining the parameters, we introduce the decision variables used to formulate the problem. We define y variables to open facilities, z variables to indicate a purchase of a vehicle for facilities, and finally x variables to assign customers to facilities and vehicles.

$$x_{ijk} := \begin{cases} 1, & \text{if demand point } i \text{ is assigned to vehicle } k \text{ of vendor } j \\ 0, & o.w \end{cases}$$

$\forall i \in I, j \in J_i, k \in K_j$

$$z_{jk} := \begin{cases} 1, & \text{if vendor } j \text{ uses vehicle } k \\ 0, & o.w \end{cases}$$

$\forall j \in J, k \in K_j$

$$y_j := \begin{cases} 1, & \text{if a vendor is located at location } j \\ 0, & o.w \end{cases}$$

$\forall j \in J.$

4.3 Objective Functions

We have two different objective functions. First one is:

$$\max \sum_{i \in I} \sum_{j \in J_i} \sum_{k \in K_j} \rho_{ij} x_{ijk} - \sum_{j \in J} \sum_{k \in K_j} v_j z_{jk} - \sum_{j \in J} f_j y_j.$$

First objective function aims to maximize profit, which consists of three terms: the revenue of facilites after deducting the transportation cost between demand points and facilites, the fixed vehicle cost, and the fixed facility cost.

Our second objective function which is to maximize the coverage of demand is the following:

$$\max \textstyle\sum_{i\in I} \sum_{j\in J_i} \sum_{k\in K_j} q_{ij}x_{ijk}.$$

4.4 Constraints

Constraints of our model are as follows.

$$\sum_{j\in J_i}\sum_{k\in K_j} x_{ijk} \leq 1 \qquad \forall i \in I \tag{4.1}$$

$$\sum_{j\in J} y_j = p \tag{4.2}$$

$$\sum_{k\in K_j} x_{ijk} \leq y_j \qquad \forall i \in I, j \in J_i \tag{4.3}$$

$$\sum_{i\in I_j} q_{ij}x_{ijk} \leq \gamma z_{jk} \qquad \forall j \in J, k \in K_j \tag{4.4}$$

$$\sum_{i\in I_j} \rho_{ij} \sum_{k\in K_j} x_{ijk} \geq \sum_{k\in K_j} v_j z_{jk} + (\rho_{min} + f_j) y_j \qquad \forall j \in J \tag{4.5}$$

$$x_{ijk} \in \{0,1\} \qquad \forall i \in I, j \in J_i, k \in K_j \tag{4.6}$$

$$z_{jk} \in \{0,1\} \qquad \forall j \in J, k \in K_j \tag{4.7}$$

$$y_j \in \{0,1\} \qquad \forall j \in J. \tag{4.8}$$

Constraints (4.1) ensure that a demand point is assigned to at most one vehicle of one eligible vendor. Constraint (4.2) states that the number of vendors to be located is p. If a vendor is not located at a given location, then a demand point cannot be served by any of its vehicles due to constraints (4.3). Constraints (4.4) are capacity constraints for vehicles. At the same time, they ensure that demand points are not assigned to vehicles which are not in use. Constraints (4.5) ensure that each vendor makes a profit of at least ρ_{min} units.

Constraint (4.6), Constraint (4.7) and Constraint (4.8) are binary constraints for the variables x, y, z.

Moreover, there is the additional restriction that if a vendor is located at a given demand point, then the demand of this point should be served by itself. To handle this, we added the constraint

$$\sum_{k \in K_j} x_{jjk} = y_j \qquad \forall j \in J. \tag{4.9}$$

Since the vehicles are identical, there is symmetry in the set of feasible solutions and multiple representations for the same solution. To reduce this symmetry, following constraints can be added:

$$z_j = x_{jj1} \qquad \forall j \in J_i \tag{4.10}$$

$$x_{jj1} = y_{j1} \qquad \forall j \in J_i. \tag{4.11}$$

As a result, we have the following integer linear programming formulations:

ProfitVLP

$$\max \sum_{i \in I} \sum_{j \in J_i} \sum_{k \in K_j} \rho_{ij} x_{ijk} - \sum_{j \in J} \sum_{k \in K_j} v_j z_{jk} - \sum_{j \in J} f_j y_j$$

s.t.

$$\text{s.t.} \quad (4.1)\text{-}(4.11).$$

CoverageVLP

$$\max \sum_{i \in I} \sum_{j \in J_i} \sum_{k \in K_j} q_{ij} x_{ijk}$$

s.t.

$$\text{s.t.} \quad (4.1)\text{-}(4.11).$$

4.5 Complexity

Now, we investigate the complexity of the problems.

Theorem 1. *ProfitVLP and CoverageVLP are strongly NP-hard.*

Proof. We prove that decision versions of *ProfitVLP* and *CoverageVLP* are NP-complete in the strong sense by a reduction from the decision version of the bin packing problem.

Given a finite set of items U, a size $s_i \in \mathbb{Z}_+$ for each $i \in U$, a positive integer bin capacity B and a positive integer κ, the decision version of the bin packing problem is defined as follows. Is there a partition of set U into $U_1, \ldots, U_\kappa$ such that $\sum_{i \in U_u} s_i \leq B$ for all $u = 1, \ldots, \kappa$? This problem is NP-complete in the strong sense (see problem [SR1] in Garey and Johnson [11]).

First remark that when $v_j = f_j = 0$ for all $j \in J$ and $\rho_{ij} = q_{ij}$ for all $i \in I$ and $j \in J_i$, problems *ProfitVLP* and *CoverageVLP* become the same problem. Hence in the remaining part of the proof, we only consider *CoverageVLP*.

We define the decision version of *CoverageVLP* as follows. Given the parameters of the problem and a positive constant Φ, does there exist a feasible solution with coverage at least Φ? This problem is in NP.

Given an instance of the bin packing problem, let J be a singleton, $I = I_1 = U$, $p = 1$, $v_1 = 0$, $f_1 = 0$, $\rho_{min} = 0$, $k_1^{max} = \kappa$, $\rho_{i1} = q_{i1} = s_i$ for $i \in I$, $\gamma = B$, $\Phi = \sum_{i \in I} q_{i1}$. Now there exists a solution to the decision version of the bin packing problem if and only if there exists a solution to the decision version of *CoverageVLP*. Hence, the decision version of *CoverageVLP* is NP-complete in the strong sense. □

Chapter 5

VALID INEQUALITIES

In this chapter, we propose some valid inequalities for both versions of the *VLP*. Let F be the set of solutions that satisfy constraints (4.1)-(4.11). We use some substructures in the formulation to derive our valid inequalities.

5.1 Lower bounds on the number of vehicles

Albareda-Sambola et al.[3] propose the optimality cuts $\sum_{k \in K_j} z_{jk} \geq y_j$ for $j \in J$. These inequalities imply that if a vendor is located then it should use at least one vehicle. In our problem, since we have minimum profit constraints, in some cases we can obtain tighter bounds on the number of vehicles to be used by a vendor. Besides the resulting inequalities are valid inequalities.

For $j \in J$ and a positive integer m, consider the following problem:

$$\begin{aligned}
\delta_j(m) = \max \sum_{i \in I_j} \sum_{k=1}^{m} \rho_{ij}\alpha_{ik} - \sum_{k=1}^{m} \upsilon_j \beta_k - f_j \\
\text{s.t. } \sum_{k=1}^{m} \alpha_{ik} \leq 1 \qquad \forall i \in I_j \\
\sum_{i \in I_j} q_{ij}\alpha_{ik} \leq \gamma\beta_k \qquad \forall k = 1, \ldots, m \\
\alpha_{ik} \in \{0,1\} \qquad \forall i \in I_j, k = 1, \ldots, m \\
\beta_k \in \{0,1\} \qquad \forall k = 1, \ldots, m.
\end{aligned}$$

This problem maximizes the total profit for vendor j if vendor j can use at most m vehicles. Let m_j be the smallest integer with $\delta_j(m_j) \geq \rho_{min}$. Then for vendor j to achieve a minimum level of profit of ρ_{min} units, it should have at least m_j vehicles. If m_j is a positive integer which is less than or equal to k_j^{max}, then the inequality $\sum_{k \in K_j} z_{jk} \geq m_j y_j$ is a valid inequality. If m_j does not exist or if $m_j > k_j^{max}$, then vendor j cannot be profitable. Hence we can set $y_j = 0$.

The above problem is a single sourcing capacitated facility location problem which is an NP-hard problem. As a result, computing the $\delta_j(m)$ values may be quite time consuming. Hence we propose a way of computing lower bounds on m_j values.

Proposition 1. *Let $j \in J$ and $\sigma_j = \max_{i \in I_j} \frac{\rho_{ij}}{q_{ij}}$. The inequality*

$$\sum_{k \in K_j} z_{jk} \geq \left\lceil \frac{\rho_{min} + f_j}{\sigma_j\gamma - \upsilon_j} \right\rceil y_j \tag{5.1}$$

is valid.

Proof. For $j \in J$, $\sigma_j q_{ij} \geq \rho_{ij}$ for all $i \in I_j$. Multiplying constraints (4.4) with σ_j and summing over $k \in K_j$ yields $\sum_{i \in I_j} \sigma_j q_{ij} \sum_{k \in K_j} x_{ijk} \leq \sigma_j \gamma \sum_{k \in K_j} z_{jk}$. Since $\sigma_j q_{ij} \geq \rho_{ij}$ for all $i \in I_j$, this implies $\sum_{i \in I_j} \rho_{ij} \sum_{k \in K_j} x_{ijk} \leq \sigma_j \gamma \sum_{k \in K_j} z_{jk}$. Now combining this with constraint (4.5), we obtain

$$\sigma_j \gamma \sum_{k \in K_j} z_{jk} \geq \sum_{i \in I_j} \rho_{ij} \sum_{k \in K_j} x_{ijk} \geq \sum_{k \in K_j} \upsilon_j z_{jk} + (\rho_{min} + f_j) y_j$$

which gives

$$(\sigma_j \gamma - v_j) \sum_{k \in K_j} z_{jk} \geq (\rho_{min} + f_j) y_j.$$

This implies that if $y_j = 1$, i.e., if a vendor is located at location j, then $\sum_{k \in K_j} z_{jk} \geq \frac{\rho_{min} + f_j}{\sigma_j \gamma - v_j}$. Since the left hand side is integer in a feasible solution, we can round up the right hand side. □

For $j \in J$, σ_j can be computed in $O(|I_j|)$ time.

5.2 Cover inequalities for vehicle capacity constraints

For $i \in I$, $j \in J_i$, and $k \in K_j$, inequality

$$x_{ijk} \leq z_{jk} \tag{5.2}$$

is a valid inequality. These inequalities are often dominated by cover inequalities that may be generated using the knapsack structure of the capacity constraints (4.4) for the vehicles. Cover inequalities that are valid for each of these knapsack constraints are also valid for F. Let $j \in J$, $k \in K_j$, and $C \subseteq I_j$ be such that $\sum_{i \in C} q_{ij} > \gamma$. Then the cover inequality $\sum_{i \in C} x_{ijk} \leq (|C| - 1) z_{jk}$ is a valid inequality.

Most of the integer programming solvers recognize knapsack constraints and use cover inequalities as cutting planes. So here we limit our attention to some lifted cover inequalities that are not many in number so that they can be added to the formulation before giving it to the solver.

For a given location $j \in J$, we first consider all demand points with demand larger than half of the capacity of a vehicle. Then we know that at most one of these points may be assigned to a given vehicle of vendor j. This leads to the following set of inequalities.

Proposition 2. *For $j \in J$ and $k \in K_j$, the lifted cover inequality*

$$\sum_{i \in I_j : q_{ij} > \frac{\gamma}{2}} x_{ijk} \leq z_{jk} \tag{5.3}$$

is valid for F.

Proof. Easy. □

Next, we generate lifted cover inequalities for each demand point $i \in I_j$ with demand not more than half the capacity.

Proposition 3. *Let $i \in I_j$ be such that $q_{ij} \leq \frac{\gamma}{2}$. Define $C_{ij} = \{l \in I_j : q_{ij} + q_{lj} > \gamma\}$. Then the lifted cover inequality*

$$x_{ijk} + \sum_{l \in C_{ij}} x_{ljk} \leq z_{jk} \tag{5.4}$$

is valid for F.

Proof. If $x_{ijk} = 1$, then as $q_{ij} + q_{lj} > \gamma$ for each $l \in C_{ij}$, none of these demand points can be served by the same vehicle. If $x_{ijk} = 0$, then as $q_{lj} + q_{mj} > \gamma$ for l and m in C_{ij}, we know that $\sum_{l \in C_{ij}} x_{ljk} \leq z_{jk}$. □

Notice that if C_{ij} is empty, then inequality (5.4) reduces to (5.2).

5.3 Cover inequalities for the minimum profit constraints

Finally, we propose cover inequalities for the minimum profit constraints. This is done by complementing sums of assignment variables and rewriting the minimum profit constraints as 0-1 knapsack constraints.

Proposition 4. *Let $j \in J$, $S_1 \subseteq I_j$, and $S_2 \subseteq K_j$ with $|S_2|v_j + (\rho_{min} + f_j) > \sum_{i \in I_j \setminus S_1} \rho_{ij}$. The inequality*

$$\sum_{k \in S_2} z_{jk} \leq \sum_{i \in S_1} \sum_{k \in K_j} x_{ijk} + (|S_2| - 1)y_j \tag{5.5}$$

is valid.

Proof. Let $j \in J$. For $i \in I_j$, define the variable $\overline{x}_{ij} = 1 - \sum_{k \in K_j} x_{ijk}$. Notice that $\overline{x}_{ij}$ is a 0-1 variable. Now the minimum profit constraint (4.5) can be rewritten as

$$\sum_{i \in I_j} \rho_{ij} \geq \sum_{i \in I_j} \rho_{ij} \overline{x}_{ij} + \sum_{k \in K_j} v_j z_{jk} + (\rho_{min} + f_j) y_j$$

which is a knapsack inequality. Suppose that $y_j = 1$. Let $S_1 \subseteq I_j$ and $S_2 \subseteq K_j$. If $\sum_{i \in S_1} \rho_{ij} + |S_2|v_j + (\rho_{min} + f_j) > \sum_{i \in I_j} \rho_{ij}$, then the cover inequality $\sum_{i \in S_1} \overline{x}_{ij} + \sum_{k \in S_2} z_{jk} \leq |S_1| + |S_2| - 1$ is valid.

We can rewrite this inequality as $\sum_{i \in S_1}(1 - \sum_{k \in K_j} x_{ijk}) + \sum_{k \in S_2} z_{jk} \leq |S_1| + |S_2| - 1$ which simplifies to $\sum_{k \in S_2} z_{jk} \leq \sum_{i \in S_1} \sum_{k \in K_j} x_{ijk} + |S_2| - 1$. If $y_j = 0$, then $x_{ijk} = 0$ for all $i \in I_j$ and $k \in K_j$ and $z_{jk} = 0$ for all $k \in K_j$. Hence inequality (5.5) is valid. □

Chapter 6

COMPUTATIONAL RESULTS

In this chapter, we describe the test data and report the outcomes of two experiments. In the first experiment, we investigate for which sizes we can solve the formulations to optimality in reasonable times and the effect of valid inequalities on the quality of LP relaxation upper bound and the solution times. In the second experiment, we compare the solutions for the two versions of the problem for different parameters.

6.1 Models

Let *ProfitM0* and *CoverageM0* be the models presented in Chapter 4 as *ProfitVLP* and *CoverageVLP* respectively.

Let *ProfitM1* and *CoverageM1* be the models *ProfitM0* and *CoverageM0* strengthened with valid inequalities (5.1).

The fact that if a vendor is located at a demand point, then the point should use its first vehicle can further be used to obtain stronger lifted cover inequalities for the first vehicles:

$$\sum_{i \in I_j \setminus \{j\}: q_{ij}+q_{jj}>\gamma} x_{ij1} = 0 \qquad \forall j \in J \tag{6.1}$$

$$\sum_{i \in I_j \setminus \{j\}: q_{ij}>\frac{\gamma-q_{jj}}{2}} x_{ij1} \leq z_{j1} \qquad \forall j \in J \tag{6.2}$$

$$x_{ij1} + \sum_{l \in I_j \setminus \{j\}: q_{ij}+q_{lj}>\gamma-q_{jj}} x_{lj1} \leq z_{j1} \qquad \forall j \in J, i \in I_j \setminus \{j\} : q_{ij} \leq \frac{\gamma - q_{jj}}{2} \tag{6.3}$$

We add the above cover inequalities for the first vehicles and inequalities (5.3) and (5.4) for the remaining vehicles to models *ProfitM1* and *CoverageM1* and call the resulting models *ProfitM2* and *CoverageM2*, respectively.

Moreover, we remove constraints (4.5) from models *ProfitM2* and *CoverageM2* and add the following variables and constraints to obtain models *ProfitM3* and *CoverageM3*:

$$\overline{x}_{ij} = 1 - \sum_{k \in K_j} x_{ijk} \qquad \forall i \in I, j \in J \tag{6.4}$$

$$\sum_{i \in I_j} \rho_{ij}\overline{x}_{ij} + \sum_{k \in K_j} v_j z_{jk} + (\rho_{min} + f_j)y_j \leq \sum_{i \in I_j} \rho_{ij} \qquad \forall j \in J \tag{6.5}$$

$$\overline{x}_{ij} \in \{0, 1\} \qquad \forall i \in I, j \in J \tag{6.6}$$

The aim is to enable the solver to see the knapsack structure in the minimum profit constraints so that it can generate cover inequalities as presented in Chapter 5.

Next, we obtain models *ProfitM4* and *CoverageM4* by adding inequalities (6.7) to models *ProfitM3* and *CoverageM3*, respectively.

$$z_{jk} \leq y_j \qquad \forall j \in J, k \in K_j \tag{6.7}$$

After we analyzed the results, we tested one more formulation for each problem type. We see that the CPU time usually increases in *ProfitVLP*, when we add cover inequalities for vehicle capacity constraints (Constraints (5.3) - (5.4) and (6.1) - (6.3)). We generate model *ProfitM5* by removing these constraints from model *ProfitM4*.

Observing the best results for CoverageVLP in terms of CPU time, we see that the most beneficial one is Constraints (6.7). So, we add only this constraints to model *CoverageM0* to derive model *CoverageM5*.

To sum up, the constraints of models *ProfitVLP* and *CoverageVLP* (*CoverVLP*) are listed in Tables 6.1 and 6.2 respectively.

ProfitM0	*ProfitM1*	*ProfitM2*	*ProfitM3*	*ProfitM4*	*ProfitM5*
(4.1)-(4.11)	(4.1)-(4.11)	(4.1)-(4.11)	(4.1)-(4.4)	(4.1)-(4.4)	(4.1)-(4.4)
			(4.6)-(4.11)	(4.6)-(4.11)	(4.6)-(4.11)
	(5.1)	(5.1)-(5.3)	(5.1)-(5.3)	(5.1)-(5.3)	(5.1)
		(6.1)-(6.3)	(6.1)-(6.6)	(6.1)-(6.7)	(6.4)-(6.7)

Table 6.1: Constraints for model *ProfitVLP*

CoverM0	*CoverM1*	*CoverM2*	*CoverM3*	*CoverM4*	*CoverM5*
(4.1)-(4.11)	(4.1)-(4.11)	(4.1)-(4.11)	(4.1)-(4.4)	(4.1)-(4.4)	(4.1)-(4.11)
			(4.6)-(4.11)	(4.6)-(4.11)	
	(5.1)	(5.1)-(5.3)	(5.1)-(5.3)	(5.1)-(5.3)	
		(6.1)-(6.3)	(6.1)-(6.6)	(6.1)-(6.7)	(6.7)

Table 6.2: Constraints for model *CoverageVLP*

For each value of p, k^{max}, and ρ_{min}, we have four different types of instances with different demand and profit structures. In **A** type problems, we take $q_{ij} = q_i$ and $\rho_{ij} = \rho_i$ for all $j \in J_i$ and $i \in I$. So in **A** type instances, the demand and profit are independent of the distance between the demand point and its vendor.

In **B** type problems, we take $q_{ij} = q_i$ and $\rho_{ij} = c_{ij} q_i$ for all $j \in J_i$ and $i \in I$ where c_{ij} is the unit profit that vendor j gains if it serves demand point i and is a function of the distance between i and j.

In **C** type problems, we take q_{ij} to be a function of the distance between i and j and $\rho_{ij} = cq_{ij}$ for all $j \in J_i$ and $i \in I$ where c is the unit profit and does not depend on distances. In this case, we let $q_{ij} = q_i$ for vendors j that are within a short traveling time of i and then let q_{ij} decrease with the distance between i and j for other eligible vendors.

Finally in **D** type instances, we take both the demands and the profits as functions of the distances.

6.2 Input Data and Parameter Selection

We are required to design the distribution system for HOD-Demijohn Water brand that will enter the market. We use the data from this demijohn water company. The data includes 84 demand points, their estimated demands, the distances, and cost parameters. Demand points are the customers buying HOD-Demijohn water and facilites are the vendors of this new brand. The set of possible locations for the vendors is the same as the set of demand points.

We define that the distance between the vendor and the customer, d_{ij}, is allowed to be at most 10 km. Therefore, we construct the set J_i as following:

$$J_i := \{j : d_{ij} \leq 10\} \quad \forall i \in I.$$

We are given the daily demand of points, q_i by the HOD-Demijohn water company. Since the waiting time is crucial for customers, we define the demand, q_{ij} demonstrated in Figure 6.1, as a function of the distance d_{ij} between demand points and vendors as follows:

$$q_{ij} := \begin{cases} q_i, & \text{if } d_{ij} \leq 5 \\ q_i(1.5 - 0.1d_{ij}), & \text{if } 5 < d_{ij} \leq 10 \end{cases}$$

$\forall i \in I, j \in J_i.$

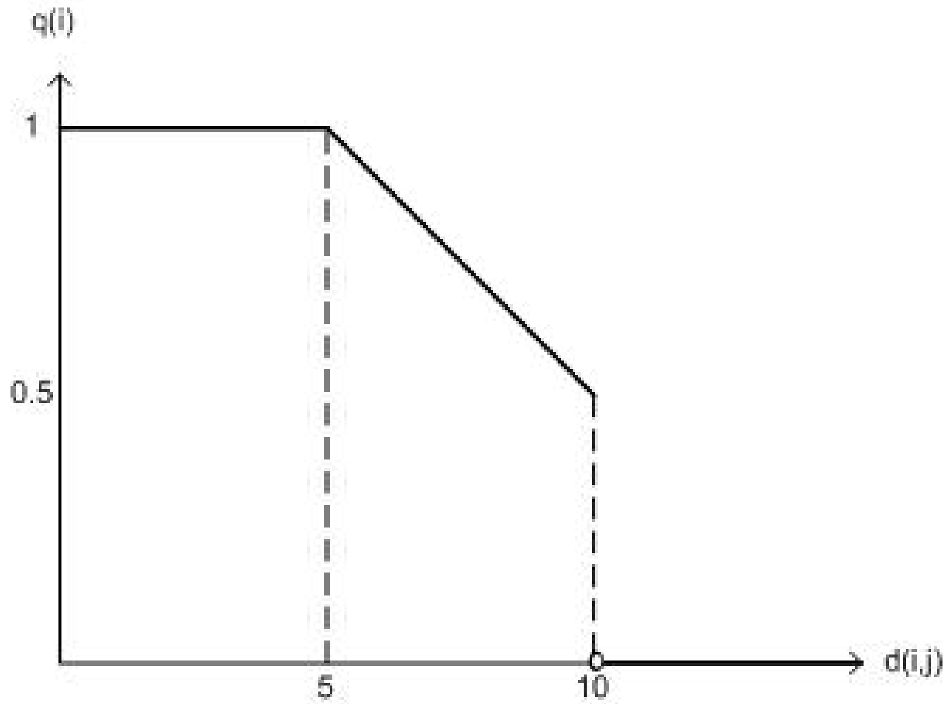

Figure 6.1: q_{ij} function in terms of distance

Company determines the fixed facility cost f_j; the vehicle cost v_j including the 5 years forward purchase cost, depreciation, tax, maintenance and the personnel salary; and the profit of each product, m, which is independent of the distance between the demand point and its vendor.

The unit profit that vendor j gains if it serves demand point i is equal to c_{ij} and is defined as follows:

$$c_{ij} := (m - ud_{ij}) \quad \forall i \in I, j \in J_i.$$

Transportation cost for each product, u, is obtained by dividing the fuel cost of 1 km by the vehicle's capacity, γ.

Finally, we set the daily capacity of each vehicle γ to 60, since every vehicle has 20 bottles capacity and can be reloaded at most 3 times in a day.

Using the above parameters, we change the number of opened facilities $p =$

$\{4, 6, 8\}$, the maximum number of vehicles $k_j^{max} = \{6, 8, 10\}$, and the minimum profit value $\rho_{min} = \{50, 100, 150\}$ to see the effects of changes in parameters p, k_j^{max}, and ρ_{min}.

6.3 Comparison of Models

In this section, we will give the comparison among our models in terms of LP relaxation gaps, CPU times or final IP gaps for unsolvable instances, and the number of branch and cut nodes. Also, the effects of valid inequalities are analyzed. We try to analyze which formulation is better in which cases.

All models are solved using GAMS 22.5 and CPLEX 11.0.0 on an AMD Opteron 252 processor (2.6 GHz) with 2 GB of RAM operating under the system CentOS (Linux version 2.6.9-42.0.3.ELsmp). We have a time limit of one hour.

In Tables A.1-A.4 in the Appendix, we report the results for *ProfitVLP* and the four types of instances, **A**, **B**, **C**, and **D**, respectively. Tables A.5-A.8 in the Appendix are the results for *CoverageVLP* and the four types of instances, **A**, **B**, **C**, and **D**, respectively. For each instance and model, we report the percentage gap between the upper bound obtained by solving the linear programming relaxation of the corresponding model and the best lower bound for the integer problem in the column *LP Gap*. Then we report the cpu times in seconds. If the problem is not solved to optimality in one hour, then we report the remaining percentage gap in parenthesis. Finally, we report the number of nodes in the branch-and-cut tree for each model and instance. The best results are marked bold.

Each table has a summary, where we can see the averages of linear programming relaxation gaps, final optimality gaps, cpu times, number of nodes, the number of instances solved to optimality with each model, and the number of times each model was among the best for the considered criterion.

In the rest of the thesis, *ProfitM0-ProfitM5* and *CoverageM0-CoverageM5* are abbreviated as *PM0-PM5* and *CM0-CM5*, respectively.

Comparing models *ProfitVLP* and *CoverageVLP* in general, it is clear that for *CoverageVLP* we can reach optimality more quickly than for *ProfitVLP*. LP relaxation gap, which is measured as $(100*(LPoptimal - IPoptimal)/IPoptimal)$, is smaller for *CoverageVLP* than the one for *ProfitVLP* with valid inequalities.

Both problems *ProfitVLP* and *CoverageVLP* were infeasible for $\rho = 150$, $p = 8$ and all 4 demand and profit structures. These instances are removed from the results.

For the LP relaxation gap of model *ProfitVLP* for type **A** problems, *PM4* gives the best results in all the instances. On the average, *PM4* reduces the LP relaxation gap from 55.14% to 4.45%. The CPU times and the number of opened nodes are also less in *PM4* than in other models on average. However, *PM3* has the highest number of best solutions over 24 instances in terms of CPU times and the number of opened nodes. *PM3*, *PM4*, and *PM5* solve all problems whereas *PM0*, *PM1*, and *PM2* cannot. It is clear that *PM3* has better CPU times for $\rho_{min} = 150$. Results of the model *ProfitVLP* for type **A** problems are shown in Table A.1.

Table A.2 gives the LP relaxation gaps, CPU times, and the number of opened nodes of *ProfitVLP* for type **B** problems. *PM4* improves LP relaxation gaps for these instances as well. *PM3* has better average CPU times. *PM2* being the third on average in terms of CPU times has the largest number of results with the smallest CPU times, since problems with $\rho_{min} = 50$ are solved most quickly with this formulation. However, it cannot reach optimality in one instance with $\rho_{min} = 100$. Formulations which can reach optimality in all instances are only *PM3*, *PM4* and *PM5* like in *ProfitVLP* for type **A** problems. *PM4* has the best average performance in terms of the number of opened nodes.

When we generate the demand function in terms of distance between demand points and facilities as in models *ProfitVLP* for type **C** and **D** problems, it gets harder to reach optimality. For the LP relaxation gap of type **C**, *PM4* has the best results except for three instances where it could not reach optimality. According to CPU times, *PM0* has the smallest value with a slight diffence from *PM5*. The smallest final IP gap which is calculated by $(100 * (Bestpossible - IPsolution)/Bestpossible)$ is given by *PM5* that is 0.78% wheras it is 0.83% for *PM0*. However, the formulation reaching the optimality most frequently is *PM0*. *PM0* can solve 7 instances to optimality, and others can solve 3,4,5,6, and 5 instances, respectively. On the other hand, *PM5* gives the smallest CPU times in 11 out of 24 instances, which is the best result among all of the six model types. *PM4* gives the smallest number of opened nodes on the average and the largest number of best solution in terms of opened nodes over 24 instances. Table A.3 includes the results of model *ProfitVLP* for type **C** problems.

The last type of instances are type **D** for *ProfitVLP* problems. From Table A.4, the smallest average LP relaxation gap is obtained using *PM4*, which has the best results in 18 out of 24 instances. Analyzing the CPU times and final IP gaps for unsolvable instances, we see that *PM*5 has the smallest CPU time, 2794.21 sec. 9 instances can be solved by *PM5*, while others can solve fewer number of instances. *PM5* giving the best solutions in 8 instances also has the largest number of best solution. According to final IP gaps, it is in the second place with 0.93%. However, the difference between the first one, *PM1*, is 0.01%. *PM3* has the best results in 10 instances on average in terms of the number of opened nodes.

We can conclude that *PM5* generally leads shorter CPU times not for all types of models, but for *ProfitVLP* type **C** and **D** problems.

For *CoverageVLP*, the LP relaxation gap is about 1% and the best model for it is *CM4* in all instances and all model types. Although it seems that the decrease in LP gap relaxation is provided due to Constraint (6.7), the same results cannot be obtained when only Constraint (6.7) is added. So, we can conclude that although the most helpful one is Constraint (6.7), other constraints also help to decrease the LP relaxation gap.

When we compare the CPU times of model *CoverageVLP* for type **A** problems, *CM4* decreases the CPU time from 521.64 sec. to 114.03 sec. This formulation

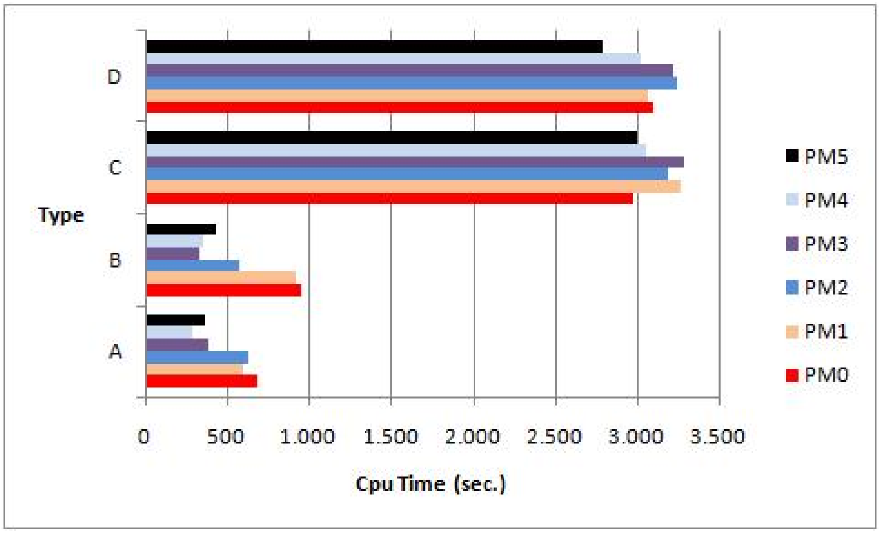

Figure 6.2: Comparison of average cpu times for *ProfitVLP*

can solve all instances, whereas others cannot. However, the best solutions are obtained in 7 instances with *CM4*, as *CM5* can solve 8 instances within the smallest CPU time. *CM4* also opens the least number of nodes with a huge difference from other formulations. The results of model *CoverageVLP* for type **A** problems are seen in Table A.5.

Results of problem *CoverageVLP* for type **B** instances indicate that *CM4* has the smallest average CPU time with 190.60 sec. This formulation not only solves all of the 24 instances within 1 hr and gives the quickest results for 9 instances. *CM5* gives the best results in terms of CPU times on 8 instances, but its average CPU time is 420.15 sec., which is about two times of *CM4*. In addition to its superiority in CPU times, *CM4* also opens the least number of nodes on the average. This outcome does not change in 16 instances. The results of *CoverageVLP* for type **B** problems are presented in Table A.6.

CoverageVLP for type **C** and **D** problems can reach optimality within 1 hr in most of the models unlike *ProfitVLP*. On the average, *CoverageVLP* has the smallest CPU time in *CM5* with 68.41 sec. for type **C** problems which are solved to optimality by all of the formulations. *CM4*, which gives the best results for *CoverageVLP* for type **A** and **B** problems, is in the second place with 96.23. 8 out of 24 instances get the smallest CPU time with *CM5*. However, *CM4* has the least number of nodes in 20 instances over 24 and provides opening fewer number of nodes on the average. The results are shown in Table A.7.

Finally, we analyze the results of *CoverageVLP* for type **D** problems in Table A.8. None of the models except *CM3* and *CM4* can reach optimality within 1 hr in 2 instances. *CM4* has the smallest average CPU time with 129.62 sec., but only 3 instances prove the optimality most quickly by *CM4*. *CM3* is the best one in this criterion with 6 instances. However, there is a slight difference between *CM3* and *CM4*'s CPU time in 6 instances, which get the best results with *CM3*. Like the CPU time, average number of opened nodes is the least in *CM4* and it has the best results in 15 instances.

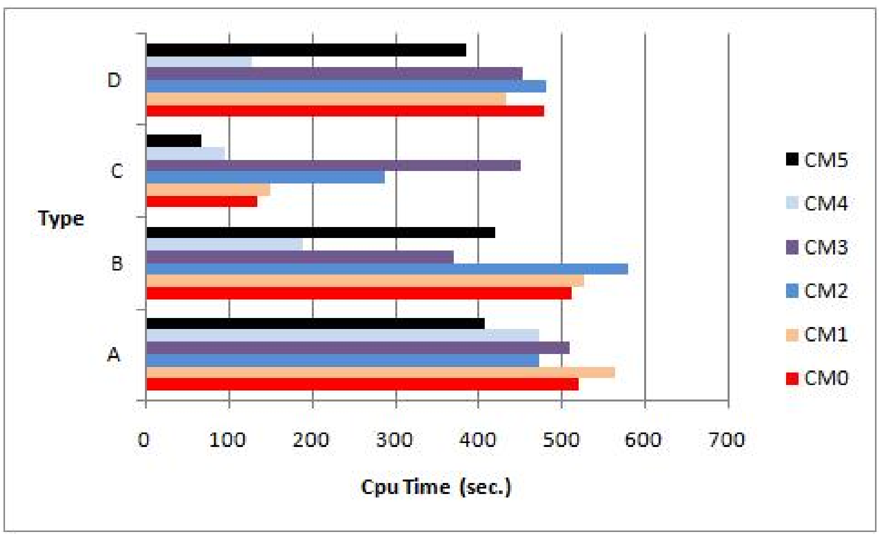

Figure 6.3: Comparison of average cpu times for *CoverageLP*

6.4 Solution Analysis of an Instance

We select an instance to analyze the solution for type **A** and type **D** problems. The instance has ρ_{min}=100, k_j^{max}=8 and p=6. The best results in terms of CPU time are obtained in *PM2*, *PM3*, *CM4* and *CM2* for the problems of *ProfitVLP* for type **A**, *ProfitVLP* for type **D**, *CoverageVLP* for **A** and *CoverageVLP* for **D** respectively. The solutions of *ProfitVLP* for type **A**, *ProfitVLP* for type **D**, *CoverageVLP* for type **A** and *CoverageVLP* for type **D** problems are shown on a map in Figure 6.4, 6.5, 6.6 and 6.7 respectively.

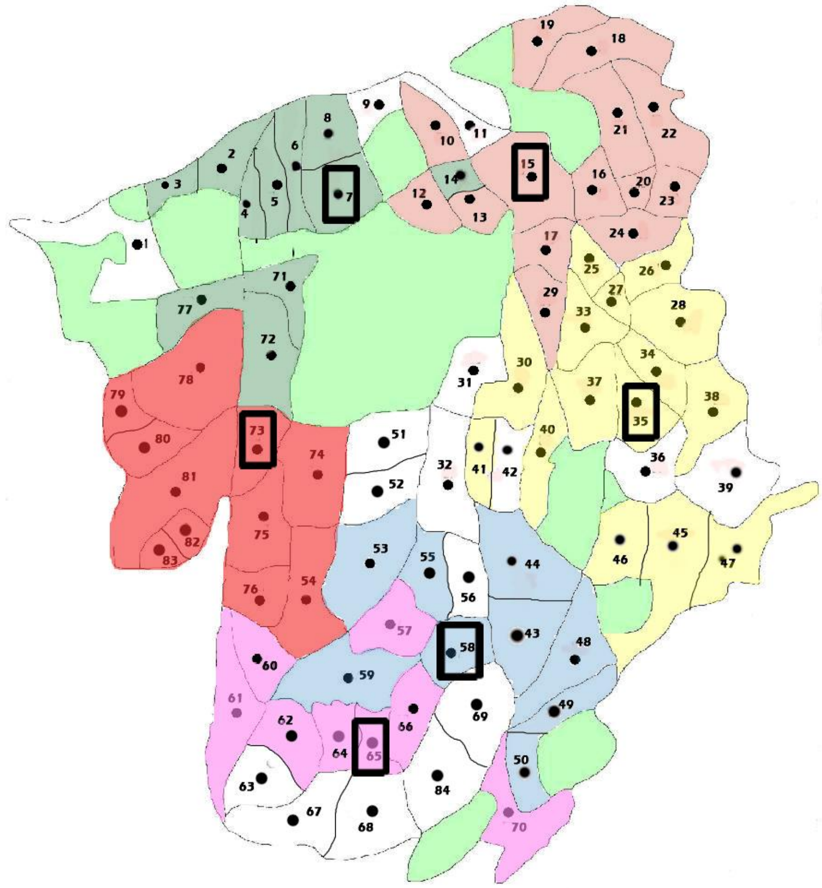

Figure 6.4: The solution of *PM2* for *ProfitVLP* type **A** problem

Comparing the solutions of problems, we see that demand points assigned to the same vendor lie close to each other in both *ProfitVLP* and *CoverageVLP* type **D** problems, wheras some demand points serviced from same vendor are seperated from the group in *ProfitVLP* and *CoverageVLP* for type **A** problems.

Moreover, uncovered demands which are white regions in the figures are less in type **D** problems than type **A**. The amounts of uncovered demand are 526, 113, 328, and 98 in *ProfitVLP* for type **A**, *ProfitVLP* for type **D**, *CoverageVLP* for type **A** and *CoverageVLP* for type **D** problems respectively.

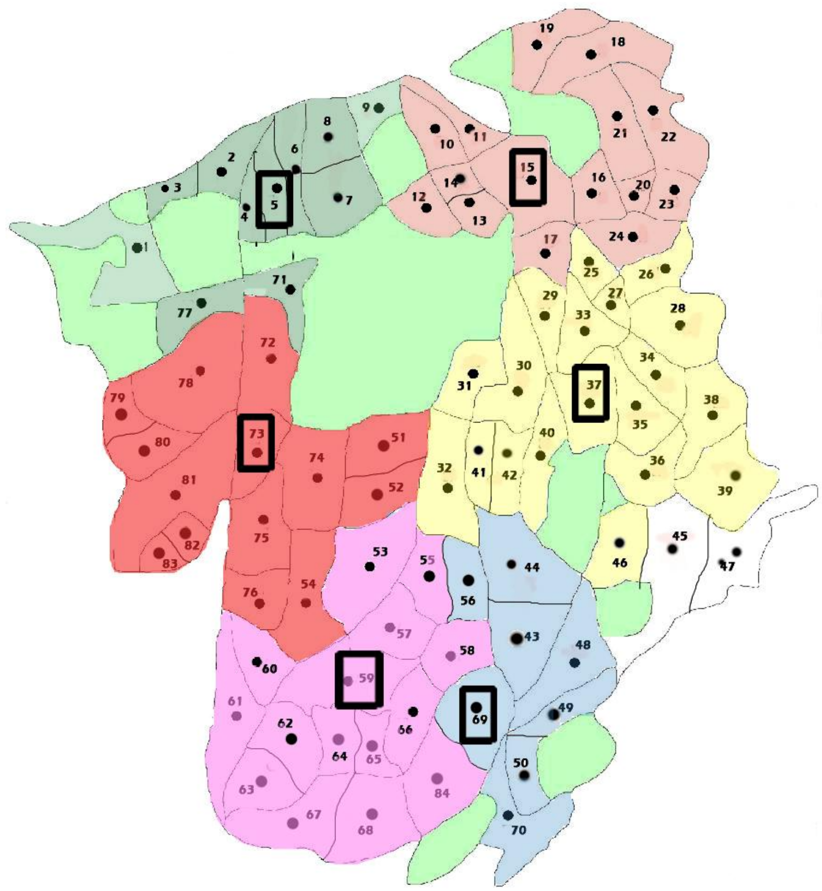

Figure 6.5: The solution of *PM3* for *ProfitVLP* type **D** problem

CoverageVLP problem types provide service to more demand points according to corresponding *ProfitVLP*. Besides, 1152.70, 1214.32, 1032.20 and 992.36 are total profits of *ProfitVLP* for type **A**, *ProfitVLP* for type **D**, *CoverageVLP* for type **A** and *CoverageVLP* for type **D** problems.

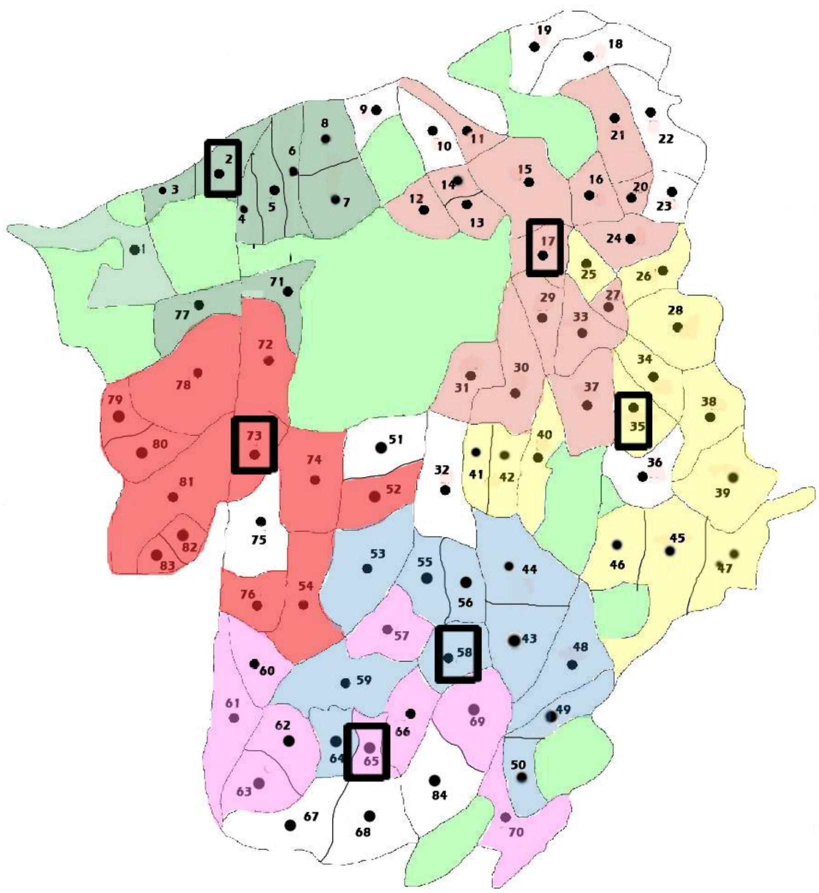

Figure 6.6: The solution of *CM4* for *CoverageVLP* type **A** problem

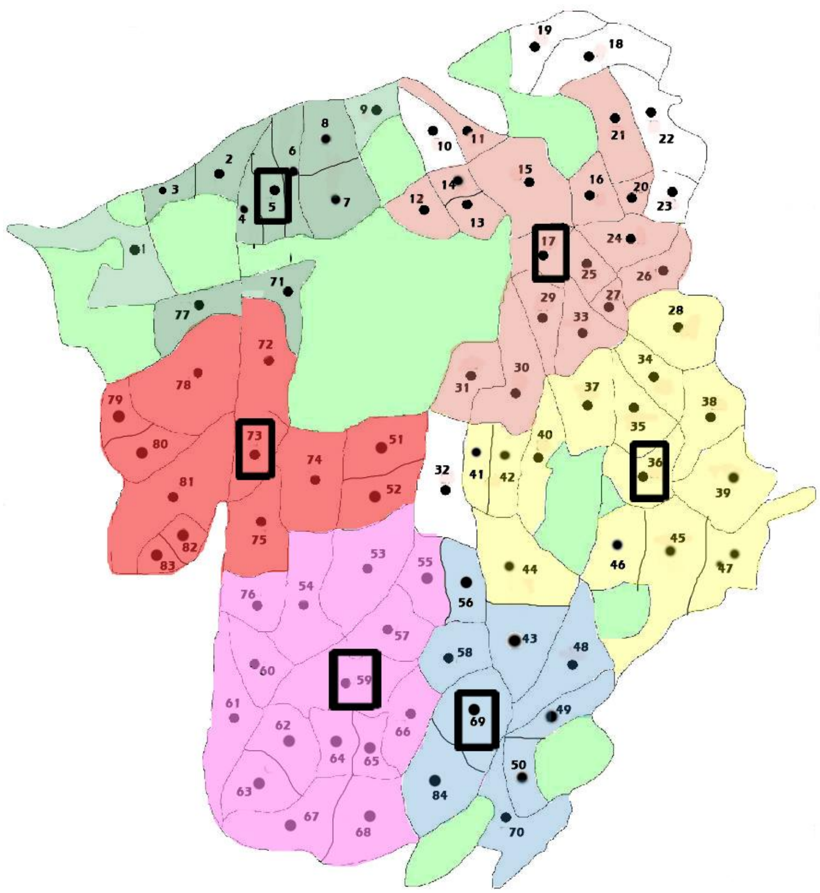

Figure 6.7: The solution of *CM2* for *CoverageVLP* type **D** problem

Chapter 7

CONCLUSION

In this thesis, we considered the *Vendor Location Problem (VLP)*. The problem is to decide where to locate vendors, the number of vehicles each vendor should have, and the assignments of customers to these vendors and vehicles.

Firstly, we developed an linear integer program for VLP with profit (*ProfitVLP*) and coverage (*CoverageVLP*) maximization objectives. We construct four different types of problems. In **A** type problems, demand and profit are independent of distance between vendors and demand points. In **B** type problems, profit is a function in terms of distance. In **C** type problems, demand is a function of the distance. In this case, demand decreases as the distance between customers and facilities increases. In **D** type problems, both the demand and the profit are functions of the distance. All problems are extensions of two-level facility location problem with capacity and minimum profit constraints.

Since both problems are NP-Hard, we added valid inequalities to the models in order to get optimal solutions at faster times and reduce the linear programming relaxation gap. We have four groups of valid inequalities: lower bounds on the number of vehicles, cover inequalities for vehicle capacity constraints, cover inequalities for the minimum profit, and vehicle-vendor inequalities.

We added above inequalities to all our problems one by one to see the effects of inequalities. The models with valid inequalities are tested by changing p, k_j^{max} and ρ_{min} parameters. All problems are solved within a time limit of 1 hour. Although valid inequalities reduce the linear programming relaxation gap, the effect of valid inequalities differ in each instance in terms of the CPU time. However, we can conclude that *CoverageVLP* is easier to solve than *ProfitVLP*. Moreover, *ProfitVLP* type **C** and type **D** problems, which include the demand as a function of the distance between demand points and vendors make the problem harder. As a result, optimality is attained for all instances except for some of *ProfitVLP* type **C** and type **D** problems within the given 1 hour time limit.

A future research direction may be to investigate different demand and profit functions.

Another future research may be to develop a heuristic for *ProfitVLP* type **C** and **D** problems, since we cannot reach optimality for some instances within 1 hour.

Finally, another future research may be an extension of Vendor Location Problem in competitive location context where other brands also have vendors. In our study, we construct profit and coverage profit maximization objectives. VLP in competitive location context can have multi-objective functions.

Bibliography

[1] Aardal K., Labbé M., Leung J., Queyranne M., On the two-level uncapacitated facility location problem. *INFORMS J. Comput.*, 8, pages 289-301, 1996.

[2] Aboolian R., Berman O., Krass D., Competitive facility location and design problem. *European Journal of Operational Research*, 182, pages 40-62, 2007.

[3] Albareda-Sambola M., Fernández E., Laporte G., The capacity and distance constrained plant location problem. *Computers & Operations Research*, 36, pages 597-611, February 2009.

[4] Berman O., Drezner Z., Location of congested capacitated facilites with distance-sensitive demand. *IIE Transaction*, 38, pages 213-221, 2006.

[5] Brotcorne L., Laporte G., Semet F., Ambulance location and relocation models. *European Journal of Operational Research*, 147, pages 451-463, 2003.

[6] Chardaire P., Lutton J.-L., Sutter A., Upper and lower bounds for the two-level simple plant location problem. *Annals of Operations Research*, 86, pages 117-140, 1999.

[7] Church R.L., Revelle C.S., The maximal covering location problem. *Papers of the Regional Science Association*, 32, pages 101-118, 1974.

[8] Current J.R., Schilling D.A., The Median Tour and Maximal Covering Tour Problems. *European Journal of Operational Research*, 73, pages 114-126, 1994.

[9] Dijkstra E.W., A note on two problems in connection with graphs. *Numerische Mathematik 1*, pages 269-271, 1959.

[10] Eiselt H.A., Laporte G., Objectives in location problems. In:Drezner, Z., Facility Location: A Survey of Applications and Methods. *Springer-Verlag*, pages 151-180, 1995.

[11] Garey M.R., Johnson D.S., Computers and Intractability, A Guide to the Theory of NP-Completeness, *Freeman*, 1979.

[12] Holmberg M., Rönnqvist M., Yuan D., An exact algorithm for the capacitated facility location problems with single sourcing. *European Journal of Operational Research* , 113, pages 544-559, 1999.

[13] Jacobsen S.K., Madsen O.B.G., A Comparative Study of Heuristics for a Two-level Location-Routing Problem. *European Journal of Operational Research* , 5, pages 378-387, 1980.

[14] Klose A., Drexl A., Facility location models for distribution system design. *European Journal of Operational Research*, 162, pages 4-29, 2005.

[15] ReVelle C.S., Eiselt H.A., Location analysis: A synthesis and survey. *European Journal of Operational Research*, 165, pages 1-19, 2005.

[16] Revelle C.S., Laporte G., The Plant Location Problem: New Models and Research Prospects. *Operation Research*, vol. 44, no. 6, pages 864-874, December 1996.

[17] Şahin G., Süral H., A review of hierarchial location models. *Computers & Operations Research*, 165, pages 1-19, 2005.

[18] Toregas C.R., Swain R., Revelle C.S., Bergman L., The location of emergency service facilities. *Operations Research*, 19, pages 1363-1373, 1971.

[19] Weber A., Uber den Standort der Industrien, 1909; translated as Alfred Weber's Theory of the Location of Industries, University of Chicago, 1929.

Appendix A

Tables of Computational Results

Parameters			LP Gap (%)						CPU Time (sec.)/Optimality Gap(%)						Number of Nodes					
ρ_{min}	k^{max}	p	*PM0*	*PM1*	*PM2*	*PM3*	*PM4*	*PM5*	*PM0*	*PM1*	*PM2*	*PM3*	*PM4*	*PM5*	*PM0*	*PM1*	*PM2*	*PM3*	*PM4*	*PM5*
50	6	4	76.77	76.77	29.21	29.21	**6.48**	**6.48**	397.61	203.00	284.25	454.59	261.57	**177.23**	29387	13208	14983	28200	13612	**8503**
50	8	4	48.05	48.05	8.56	8.56	**5.89**	25.82	411.81	**127.10**	562.07	2874.28	866.48	1002.29	20792	**7760**	26693	103257	23868	43420
50	10	4	42.51	42.51	**3.92**	**3.92**	**3.92**	42.31	244.58	153.81	32.81	12.25	**9.46**	43.52	27254	17360	3119	248	**85**	780
50	6	6	58.36	58.36	9.26	9.26	**5.04**	21.57	116.99	**110.11**	123.83	196.02	166.54	110.79	3847	**2575**	4269	4660	3846	2894
50	8	6	50.91	50.91	4.80	4.80	**4.15**	50.91	1180.66	568.44	**60.53**	112.02	91.72	653.58	49637	20099	**929**	1621	1683	28500
50	10	6	49.93	49.93	**4.12**	**4.12**	**4.12**	49.93	287.11	772.12	149.12	**84.97**	215.28	407.10	14542	25647	1890	**1496**	2853	10654
50	6	8	55.24	55.24	2.01	2.01	**1.91**	55.24	128.86	143.02	**105.59**	161.81	130.32	285.60	2767	2380	**1437**	2094	1617	3476
50	8	8	55.09	55.09	**1.93**	**1.93**	**1.93**	55.09	262.26	280.51	286.32	**218.60**	318.31	471.21	5330	3551	4569	**2248**	3228	6962
50	10	8	55.09	55.09	**1.93**	**1.93**	**1.93**	55.09	2062.02	996.55	1013.52	871.82	**838.39**	1068.35	57867	20087	19222	17513	**8925**	21266
100	6	4	76.77	76.77	29.21	29.21	**6.48**	**6.48**	**99.38**	224.41	214.90	226.33	174.20	161.32	**6639**	13122	10132	12692	9235	7816
100	8	4	48.05	48.05	8.56	8.56	**5.89**	25.82	907.42	**330.63**	1712.14	718.57	630.69	859.88	58284	**13090**	96939	39225	26042	39542
100	10	4	42.51	42.51	**3.92**	**3.92**	**3.92**	42.31	93.30	67.80	20.82	10.61	**7.48**	30.33	7063	3413	504	274	**64**	533
100	6	6	58.36	58.36	9.26	9.26	**5.04**	21.57	**85.56**	168.24	108.66	113.17	248.37	121.41	**1947**	7306	3870	2791	5043	2483
100	8	6	50.91	50.91	4.77	4.77	**4.12**	50.91	565.66	593.62	**73.19**	80.21	117.88	312.07	18574	24299	1034	**1032**	1598	12511
100	10	6	49.93	49.93	**4.09**	**4.09**	**4.09**	49.93	326.92	292.73	176.99	**100.61**	139.74	477.56	9777	11121	4441	**1315**	1655	14364
100	6	8	55.24	55.24	1.84	1.84	**1.82**	55.24	270.22	**159.86**	330.99	310.32	408.94	285.72	4837	**2495**	4595	3669	4189	2963
100	8	8	55.24	55.24	**1.84**	**1.84**	**1.84**	55.24	3259.19	1315.94	608.58	**460.38**	467.88	472.78	66262	30370	10986	**3431**	4171	4406
100	10	8	55.24	55.24	**1.84**	**1.84**	**1.84**	55.24	(0.05)	(0.05)	(0.05)	**741.72**	897.38	742.00	54635	50175	47481	**4803**	5475	5494
150	6	4	76.77	76.77	29.21	29.21	**6.48**	**6.48**	427.28	205.37	340.14	**142.27**	166.03	132.58	34745	10641	23289	6411	7191	**5580**
150	8	4	48.05	48.05	8.56	8.56	**5.89**	25.82	282.47	883.12	976.47	961.01	**164.50**	398.14	20508	46832	54511	38617	**5468**	22293
150	10	4	42.51	42.51	**3.92**	**3.92**	**3.92**	42.31	27.77	25.76	19.65	**9.84**	36.53	52.64	738	690	638	**160**	555	661
150	6	6	61.23	61.23	11.02	11.02	**6.85**	23.77	242.70	256.54	141.90	**97.67**	125.62	120.08	11248	6592	4629	**1580**	1757	3557
150	8	6	55.77	55.77	7.06	7.06	**6.76**	55.77	366.69	249.83	619.59	**52.91**	139.67	108.36	10633	11690	23780	652	666	**198**
150	10	6	54.73	54.73	**6.35**	**6.35**	**6.35**	54.73	979.08	2620.63	(0.04)	**202.62**	279.85	163.29	35910	64156	115684	**877**	928	981
Average			55.14	55.14	8.22	8.22	**4.45**	38.92	691.24	597.88	631.76	383.94	**287.62**	360.74	23051	17027	19984	11604	**5573**	10411
Avg. Opt. Gap(%)									**(0.00)**	**(0.00)**	**(0.00)**	**(0.00)**	**(0.00)**	**(0.00)**						
# of Solved Ins.(/24)									23	23	22	**24**	**24**	**24**						
# of Best Solutions(/24)					10	10	**24**	3	2	4	3	**10**	4	1	2	4	2	**9**	4	3

Table A.1: LP gap, CPU time/ optimality gap and number of nodes of *ProfitVLP* in **A** type instances

Parameters			LP Gap (%)						CPU Time (sec.)/Optimality Gap(%)						Number of Nodes					
ρ_{min}	k^{max}	p	*PM0*	*PM1*	*PM2*	*PM3*	*PM4*	*PM5*	*PM0*	*PM1*	*PM2*	*PM3*	*PM4*	*PM5*	*PM0*	*PM1*	*PM2*	*PM3*	*PM4*	*PM5*
50	6	4	77.61	77.61	29.38	29.38	**6.93**	7.78	**118.79**	243.27	403.13	435.34	394.87	157.66	**6510**	14967	26960	26043	23355	6967
50	8	4	48.76	48.76	8.72	8.72	**6.04**	26.82	1004.05	2726.00	858.60	664.81	**140.27**	483.59	73179	103608	66030	33633	**3897**	16495
50	10	4	43.51	43.51	**4.32**	**4.32**	**4.32**	43.31	80.74	185.76	**10.28**	19.22	20.70	54.16	4358	10526	**278**	396	368	690
50	6	6	59.28	59.28	9.42	9.42	**5.33**	22.89	378.75	173.11	**108.48**	302.37	206.38	204.76	17782	8094	**4562**	7770	5924	10605
50	8	6	52.04	52.04	5.12	5.12	**4.55**	52.02	368.67	119.70	149.56	**71.49**	147.88	165.26	14233	4161	5163	**1678**	2169	5203
50	10	6	51.17	51.17	**4.53**	**4.53**	**4.53**	51.17	1969.78	350.27	**118.53**	144.66	203.60	1920.13	82051	13989	**2111**	2356	4122	51737
50	6	8	57.31	57.31	2.38	2.38	**2.21**	57.23	277.26	308.33	**101.84**	172.87	178.26	213.94	7455	5374	**1530**	2400	1923	2837
50	8	8	56.75	56.75	2.03	2.03	**2.02**	56.74	245.46	285.58	**147.91**	314.23	417.90	306.81	4151	4682	**2970**	4010	4293	4268
50	10	8	56.75	56.75	**2.04**	**2.04**	**2.04**	56.75	765.83	680.23	**428.27**	563.48	1128.88	847.33	16686	13179	**5266**	6304	9672	9794
100	6	4	77.61	77.61	29.38	29.38	**6.93**	7.78	625.21	392.02	**168.70**	187.98	204.20	698.42	49140	29850	10168	8666	14426	**6520**
100	8	4	48.76	48.76	8.72	8.72	**6.04**	26.82	371.98	836.81	356.90	782.17	568.03	**340.19**	25341	37881	18964	34894	22507	**14522**
100	10	4	43.51	43.51	**4.32**	**4.32**	**4.32**	43.31	148.09	44.02	**8.64**	17.50	39.84	68.94	10875	1288	**307**	348	576	1074
100	6	6	59.28	59.28	9.41	9.41	**5.33**	22.89	201.71	256.55	218.15	239.38	428.11	**180.65**	10180	10857	8823	9104	18007	**6085**
100	8	6	52.10	52.10	5.11	5.11	**4.53**	52.08	626.96	380.98	398.19	**75.75**	167.60	246.32	49338	28355	40180	**1286**	3042	10095
100	10	6	51.23	51.23	**4.52**	**4.52**	**4.52**	51.23	539.51	338.08	222.25	**148.77**	189.52	755.32	28173	22321	9924	**2428**	3587	25125
100	6	8	58.33	58.32	2.81	2.81	**2.70**	58.25	861.99	1562.33	1284.50	661.40	506.51	**447.86**	30967	35979	24283	6785	**4620**	6645
100	8	8	57.53	57.53	2.29	2.29	**2.27**	57.53	(0.21)	(0.03)	2493.71	**743.06**	775.63	1250.77	80039	112763	56433	**7183**	7549	14744
100	10	8	57.31	57.31	**2.15**	**2.15**	**2.15**	57.31	(0.06)	(0.01)	(0.05)	1384.88	1296.78	**1069.77**	70320	87082	101959	9137	**9011**	11935
150	6	4	77.61	77.61	29.38	29.38	**6.93**	7.78	1263.19	**107.18**	154.69	217.05	214.49	156.04	76293	6924	7789	11128	12527	**6577**
150	8	4	48.76	48.76	8.72	8.72	**6.04**	26.82	1382.57	2057.22	869.85	508.54	795.61	**407.60**	86130	120951	42405	31613	32705	**19499**
150	10	4	43.51	43.51	**4.32**	**4.32**	**4.32**	43.31	147.35	32.47	22.35	**11.43**	18.68	77.87	13428	550	685	**214**	339	1622
150	6	6	61.61	61.61	10.30	10.30	**6.71**	24.69	544.47	132.41	**117.53**	120.22	167.07	94.94	19639	4502	3275	**2975**	4778	3757
150	8	6	56.82	56.82	6.78	6.78	**6.50**	56.80	(0.01)	125.11	1006.17	**88.80**	124.49	152.24	113245	901	53107	664	**660**	1605
150	10	6	55.87	55.87	**6.15**	**6.15**	**6.15**	55.87	111.98	(0.00)	167.07	138.64	**53.78**	137.25	2345	89894	3173	743	**573**	764
Average			56.38	56.38	8.43	8.43	**4.73**	40.30	951.44	922.41	558.98	**333.91**	349.55	434.91	37161	32028	20681	8823	**7943**	9965
Avg. Opt. Gap(%)									(0.01)	**(0.00)**	**(0.00)**	**(0.00)**	**(0.00)**	**(0.00)**						
# of Solved Ins.(/24)									21	21	23	**24**	**24**	**24**						
# of Best Solutions(/24)					8	8	**24**		1	1	**9**	6	2	5	1		**7**	6	5	5

Table A.2: LP gap, CPU time/ optimality gap and number of nodes of *ProfitVLP* in **B** type instances

Parameters			LP Gap (%)						CPU Time (sec.)/Optimality Gap(%)						Number of Nodes					
ρ_{min}	k^{max}	p	*PM0*	*PM1*	*PM2*	*PM3*	*PM4*	*PM5*	*PM0*	*PM1*	*PM2*	*PM3*	*PM4*	*PM5*	*PM0*	*PM1*	*PM2*	*PM3*	*PM4*	*PM5*
50	6	4	36.40	36.40	34.53	34.53	**3.99**	**3.99**	415.66	545.82	908.56	808.90	235.64	**137.35**	21173	29775	25878	16029	**4582**	5389
50	8	4	13.23	13.23	11.75	11.75	**11.27**	12.69	(0.20)	(0.77)	(1.46)	**1823.53**	(1.39)	(0.39)	263317	210493	**33642**	55561	113821	195620
50	10	4	13.27	13.27	**11.80**	**11.80**	**11.80**	13.27	(1.31)	(2.43)	(2.19)	(3.42)	(1.38)	**(1.16)**	153821	125472	**46926**	64855	58345	115305
50	6	6	21.78	21.78	17.42	17.42	**14.10**	14.80	(0.06)	(0.02)	(1.82)	(0.23)	**2466.03**	(0.35)	89914	81063	**22019**	25754	25452	66734
50	8	6	12.88	12.88	**9.04**	**9.04**	**9.04**	12.88	**1029.34**	(0.86)	(1.15)	(0.16)	(0.02)	(0.68)	33616	52149	**25086**	65312	41496	57322
50	10	6	12.88	12.88	**9.04**	**9.04**	**9.04**	12.88	(0.88)	(0.81)	(0.71)	(0.86)	(0.81)	**(0.58)**	60059	55828	24680	31578	**22146**	67819
50	6	8	18.37	18.37	11.74	11.74	**10.90**	17.56	(1.72)	(1.47)	(1.30)	(4.04)	(5.18)	**(0.68)**	29311	25123	18448	**17679**	24301	57322
50	8	8	16.49	16.49	**9.96**	**9.96**	**9.96**	16.49	(0.63)	(0.74)	(1.19)	(1.10)	(1.01)	**(0.55)**	22195	21204	15233	14515	**14275**	21764
50	10	8	16.53	16.53	**10.01**	**10.01**	**10.01**	16.53	**(1.05)**	(1.26)	(1.74)	(1.96)	(1.76)	(1.18)	21175	21012	12847	**11431**	13802	18843
100	6	4	36.40	36.40	34.53	34.53	**3.99**	**3.99**	901.64	1212.46	690.56	1312.17	227.35	**157.72**	44662	52312	21072	23443	**3208**	5496
100	8	4	13.23	13.23	11.75	11.75	**11.27**	12.69	**3191.66**	(0.81)	(0.67)	(0.75)	(0.58)	(0.39)	119696	182370	183491	72687	**67202**	217651
100	10	4	13.27	13.27	**11.80**	**11.80**	**11.80**	13.27	(3.44)	(2.10)	(2.07)	(2.41)	(2.58)	**(1.37)**	94767	153907	61780	52502	**42564**	116855
100	6	6	21.78	21.78	17.41	17.41	**14.10**	14.80	(0.58)	(0.02)	(0.06)	(0.41)	2271.14	**1773.18**	111839	75878	32600	28226	27477	**24158**
100	8	6	12.88	12.88	**8.95**	**8.95**	**8.95**	12.88	(0.74)	(0.47)	(0.55)	**(0.06)**	(0.52)	(0.48)	69973	73212	43094	68349	**41428**	85315
100	10	6	12.88	12.88	**8.95**	**8.95**	**8.95**	12.88	(0.66)	**(0.59)**	(0.96)	(0.76)	(0.78)	(0.81)	87953	57383	**25728**	26137	46347	49519
100	6	8	20.07	20.07	13.44	13.44	**12.61**	19.38	(1.68)	(1.32)	**(0.64)**	(1.49)	(2.55)	1.97	29552	23039	17985	**14123**	18713	22312
100	8	8	16.49	16.49	**9.95**	**9.95**	**9.95**	16.49	(0.53)	**(0.30)**	(2.43)	(0.81)	(1.58)	(0.91)	23977	27750	13911	**11026**	11332	15121
100	10	8	16.53	16.53	**10.00**	**10.00**	**10.00**	18.40	(2.09)	(1.31)	(1.82)	(2.36)	**(1.17)**	(2.99)	20528	18319	9930	11658	**9690**	17731
150	6	4	36.40	36.40	34.53	34.53	**3.99**	**3.99**	350.66	1091.24	783.26	3397.84	213.28	**126.14**	17131	61934	24382	106412	**3816**	4747
150	8	4	13.23	13.23	11.75	11.75	**11.27**	12.69	**2749.68**	(0.73)	(0.95)	3295.62	3170.46	(0.57)	245267	114423	**85248**	91799	118125	189367
150	10	4	13.27	13.27	**11.80**	**11.80**	**11.80**	13.27	**(1.16)**	(2.20)	(2.20)	(2.19)	(1.25)	**(1.16)**	149194	132028	52135	82438	**50298**	153399
150	6	6	21.78	21.69	17.20	17.20	**14.10**	14.80	1725.05	(0.02)	2271.70	(0.39)	(0.52)	**1430.80**	46143	67638	**31197**	26376	32231	17171
150	8	6	14.34	14.34	**10.18**	**10.18**	**10.18**	14.34	(1.34)	(1.34)	(1.18)	**(0.99)**	(1.55)	(1.14)	80373	82554	52963	44136	**42435**	77732
150	10	6	14.34	14.34	**10.18**	**10.18**	**10.18**	14.34	(1.86)	(1.77)	(1.77)	(1.77)	**(1.41)**	(1.49)	48787	58336	**30143**	34224	53502	67901
Average			18.28	18.28	14.49	14.49	**10.14**	13.24	**2982.99**	3268.79	3194.00	3293.33	3057.75	3001.11	78518	75133	37934	41510	**36941**	69608
Avg. Opt. Gap(%)									(0.83)	(0.89)	(1.49)	(1.09)	(1.09)	**(0.78)**						
# of Solved Ins.(/24)									**7**	3	4	5	6	5						
# of Best Solutions(/24)					13	13	**24**	3	5	2	1	3	3	**11**			8	4	**11**	1

Table A.3: LP gap, CPU time/ optimality gap and number of nodes of *ProfitVLP* in **C** type instances

Parameters			LP Gap (%)						CPU Time (sec.)/Optimality Gap(%)						Number of Nodes					
ρ_{min}	k^{max}	p	*PM0*	*PM1*	*PM2*	*PM3*	*PM4*	*PM5*	*PM0*	*PM1*	*PM2*	*PM3*	*PM4*	*PM5*	*PM0*	*PM1*	*PM2*	*PM3*	*PM4*	*PM5*
50	6	4	36.72	36.72	34.79	34.79	**4.31**	4.77	561.12	488.34	633.79	1956.61	232.02	**160.37**	28765	25766	15534	34652	**5139**	7266
50	8	4	13.44	13.44	11.93	11.93	**11.47**	12.93	1164.72	**455.10**	(0.28)	(1.16)	(1.09)	2809.47	67315	**28114**	94733	100649	56289	170833
50	10	4	13.48	13.48	**11.98**	**11.98**	**11.98**	13.48	(3.12)	(2.15)	(2.67)	(2.30)	**(1.64)**	(2.26)	100368	110347	68070	**28127**	66041	111921
50	6	6	21.92	21.92	17.43	17.43	**13.98**	14.99	(0.01)	(0.01)	(0.60)	(0.70)	**1757.17**	2004.23	84878	98031	62948	22232	**14977**	26490
50	8	6	13.27	13.27	**9.32**	**9.32**	**9.32**	13.27	(1.58)	(0.74)	(0.73)	(0.54)	(0.71)	**(0.09)**	81713	56679	**29972**	30236	45316	72111
50	10	6	13.27	13.27	**9.32**	**9.32**	**9.32**	13.27	(1.04)	(0.75)	(0.86)	(0.79)	**(0.62)**	(0.82)	52052	47068	19260	**18872**	40327	57108
50	6	8	20.76	20.76	13.77	13.77	**12.87**	19.97	(3.12)	**(2.14)**	(2.17)	(2.63)	(4.98)	(2.56)	32632	29856	21022	**16450**	26431	29340
50	8	8	17.01	17.01	**10.25**	**10.25**	**10.25**	17.01	(1.10)	**(0.90)**	(1.36)	(1.71)	(1.25)	(1.53)	29339	18631	13130	15488	**12540**	22387
50	10	8	17.03	17.03	**10.27**	**10.27**	**10.27**	17.01	**(1.33)**	(1.62)	(1.69)	(1.61)	(2.39)	(3.12)	23767	20408	**11139**	11722	13368	20784
100	6	4	36.72	36.72	34.79	34.79	**4.31**	4.77	3386.70	507.64	1106.28	1203.65	698.44	**164.79**	121859	20229	30277	20402	13817	**7294**
100	8	4	13.44	13.44	11.93	11.93	**11.47**	12.93	**1712.80**	(1.62)	3393.36	(0.21)	3577.48	2669.71	143447	187751	115416	**91689**	133343	154873
100	10	4	13.48	13.48	**11.98**	**11.98**	**11.98**	13.48	(2.29)	(2.14)	(3.88)	(1.46)	(2.42)	**(0.82)**	150628	127016	61970	70603	**47248**	164907
100	6	6	21.92	21.92	17.42	17.42	**13.98**	14.99	(0.58)	(0.01)	(0.01)	(0.58)	1718.09	**1618.76**	90554	77487	56953	24518	**14288**	17998
100	8	6	13.27	13.27	**9.21**	**9.21**	**9.21**	13.27	(0.71)	(0.09)	(0.32)	**3097.56**	(0.91)	(0.64)	72710	67244	**35361**	40838	80211	71418
100	10	6	13.27	13.27	**9.21**	**9.21**	**9.21**	13.27	(0.93)	(1.19)	(0.81)	**(0.73)**	(0.81)	**(0.73)**	48044	44638	31362	31477	**21707**	52541
100	6	8	20.60	20.60	13.74	13.74	**12.86**	19.86	(1.25)	(0.47)	**(0.04)**	(2.09)	(2.89)	(1.35)	28266	29342	28226	**15084**	18162	19170
100	8	8	17.01	17.01	**10.24**	**10.24**	**10.24**	17.01	**(0.60)**	(0.78)	(0.76)	(2.54)	(1.35)	(1.71)	25717	19918	13032	**10990**	14204	20374
100	10	8	17.01	17.01	**10.24**	**10.24**	**10.24**	17.01	**(1.10)**	(1.22)	(1.29)	(2.77)	(3.31)	(1.22)	22651	17197	11800	**10205**	11563	18160
150	6	4	36.72	36.72	34.79	34.79	**4.31**	4.77	419.46	290.98	801.11	1717.41	224.96	**178.18**	23992	14481	38333	27987	**4510**	7150
150	8	4	13.44	13.44	11.93	11.93	**11.47**	12.93	2384.95	(0.62)	(1.68)	**932.03**	(0.53)	1703.53	118142	116417	99826	**32216**	110576	113171
150	10	4	13.48	13.48	**11.98**	**11.98**	**11.98**	13.48	(2.10)	(2.74)	(2.99)	**(1.81)**	(2.42)	(2.40)	118407	194520	**49374**	75712	64753	86667
150	6	6	22.45	22.45	17.82	17.82	**14.60**	15.60	(0.72)	(0.01)	(0.00)	(2.17)	3056.64	**1750.43**	59335	83396	46790	**19144**	25628	22813
150	8	6	14.78	14.78	**10.49**	**10.49**	**10.49**	14.78	(1.40)	(1.33)	(1.48)	(1.35)	**(0.83)**	(1.41)	80001	83067	53054	**39108**	43934	53446
150	10	6	14.78	14.78	**10.49**	**10.49**	**10.49**	14.78	(1.86)	(1.61)	(1.45)	**(1.44)**	(1.53)	(1.71)	48006	61615	**24584**	26854	36213	61914
Average			18.72	18.72	14.81	14.81	**10.44**	13.73	3101.29	3072.64	3247.35	3221.22	3019.44	**2794.21**	68858	65801	43007	**33965**	38358	57922
Avg. Opt. Gap(%)									(1.04)	**(0.92)**	(1.05)	(1.19)	(1.24)	(0.93)						
# of Solved Ins.(/24)									6	4	4	5	7	**9**						
# of Best Solutions(/24)					13	13	**24**		4	3	1	5	4	**8**		1	5	**10**	7	1

Table A.4: LP gap, CPU time/ optimality gap and number of nodes of *ProfitVLP* in **D** type instances

Parameters			LP Gap (%)						CPU Time (sec.)/Optimality Gap(%)						Number of Nodes					
ρ_{min}	k^{max}	p	*CM0*	*CM1*	*CM2*	*CM3*	*CM4*	*CM5*	*CM0*	*CM1*	*CM2*	*CM3*	*CM4*	*CM5*	*CM0*	*CM1*	*CM2*	*CM3*	*CM4*	*CM5*
50	6	4	54.24	53.24	53.83	53.83	**1.91**	**1.91**	261.90	175.16	374.14	246.76	146.60	**113.41**	19701	13054	25542	11277	8610	**6198**
50	8	4	23.05	23.05	22.71	22.71	**1.58**	7.26	255.51	811.10	801.58	260.04	**51.14**	203.96	17862	53197	71304	15729	**2597**	8852
50	10	4	5.77	5.77	5.66	5.66	**0.62**	5.64	17.32	10.26	14.79	24.50	15.63	**8.96**	863	811	745	654	715	**629**
50	6	6	30.68	30.68	30.34	30.34	**1.31**	5.78	85.34	61.67	89.38	88.92	79.11	**53.68**	4634	2807	3692	2097	1910	**1710**
50	8	6	9.27	9.27	9.27	9.27	**1.15**	9.27	71.74	49.29	**39.93**	99.65	68.32	42.57	1907	843	685	1333	**570**	752
50	10	6	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	6.21	3.54	1.74	2.69	1.71	**0.61**	211	20	**0**	**0**	**0**	**0**
50	6	8	9.21	9.21	9.21	9.21	**0.44**	9.21	1329.72	1996.49	(0.29)	(0.30)	**106.77**	148.13	233085	32144	40052	31589	**625**	3298
50	8	8	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**3.79**	16.66	4.00	13.34	12.14	33.74	**0**	250	**0**	8	10	380
50	10	8	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	5.16	13.68	5.78	6.94	5.64	**3.02**	80	150	**0**	**0**	**0**	134
100	6	4	54.24	54.24	53.70	53.70	**1.91**	**1.91**	219.06	209.80	183.54	339.86	131.53	**88.21**	16737	15631	9254	13453	6647	**2866**
100	8	4	23.05	23.05	22.69	22.69	**1.58**	7.26	571.61	696.36	1687.35	543.48	**78.93**	888.37	32944	29950	66906	35824	**6046**	49649
100	10	4	5.77	5.77	5.66	5.66	**0.62**	5.64	22.05	19.96	7.91	15.94	**7.65**	16.25	1326	945	**257**	690	544	1141
100	6	6	30.68	30.68	30.34	30.34	**1.31**	5.78	**44.16**	63.87	75.78	112.94	97.16	52.00	1473	2951	2647	2710	2328	**1069**
100	8	6	9.44	9.44	9.44	9.44	**1.32**	9.44	58.18	67.67	87.82	214.85	**32.86**	50.73	1426	2435	830	4594	**425**	1163
100	10	6	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	8.88	4.77	2.19	3.38	1.66	**1.28**	381	67	**0**	**0**	**0**	**0**
100	6	8	9.78	9.78	9.78	9.78	**0.96**	9.78	3412.02	(0.63)	(0.57)	(0.57)	**952.09**	3014.50	29703	30739	30454	27678	**8222**	35885
100	8	8	0.88	0.88	**0.00**	**0.00**	**0.00**	0.33	(0.88)	220.31	**25.54**	142.29	199.87	(0.33)	12777	1615	**40**	557	800	14182
100	10	8	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	131.76	625.47	73.83	**20.53**	22.89	202.88	1516	7030	519	103	**30**	2267
150	6	4	54.24	54.24	53.62	53.62	**1.91**	**1.91**	105.72	189.55	262.52	242.02	137.16	**92.46**	8900	15466	18429	8928	6324	**5142**
150	8	4	23.05	23.05	22.69	22.69	**1.58**	7.26	947.77	**88.44**	238.58	1361.99	113.91	235.52	44811	**5313**	11125	40893	6276	9365
150	10	4	5.77	5.77	5.66	5.66	**0.62**	5.64	19.76	23.46	20.99	**8.33**	13.13	13.19	1085	1275	968	**576**	611	800
150	6	6	31.32	31.32	29.10	29.10	**1.78**	6.30	387.94	96.44	108.43	**87.50**	130.49	146.91	21164	2363	3592	1955	2730	**1854**
150	8	6	12.38	12.38	10.35	10.35	**3.69**	12.38	765.14	(0.34)	759.93	198.72	**171.76**	505.64	24278	78776	18036	963	**809**	18144
150	10	6	6.99	6.99	**5.08**	**5.08**	**5.08**	6.99	188.60	916.46	189.82	**135.33**	158.56	278.16	3086	18242	642	**480**	491	4892
Average			16.66	16.66	16.21	16.21	**1.22**	4.99	521.64	565.15	510.66	473.76	**114.03**	408.10	11248	13170	12738	8420	**2388**	7099
Avg. Opt. Gap(%)									(0.04)	(0.04)	(0.04)	(0.04)	**(0.00)**	(0.01)						
# of Solved Ins.(/24)									23	22	22	22	**24**	23						
# of Best Solutions(/24)			5	5	7	7	**24**	8	2	1	2	4	7	**8**	1	1	6	5	**10**	9

Table A.5: LP gap, CPU time/ optimality gap and number of nodes of *CoverageVLP* in **A** type instances

Parameters			LP Gap (%)						CPU Time (sec.)/Optimality Gap(%)						Number of Nodes					
ρ_{min}	k^{max}	p	*CM0*	*CM1*	*CM2*	*CM3*	*CM4*	*CM5*	*CM0*	*CM1*	*CM2*	*CM3*	*CM4*	*CM5*	*CM0*	*CM1*	*CM2*	*CM3*	*CM4*	*CM5*
50	6	4	54.24	54.24	53.75	53.75	**1.91**	**1.91**	193.65	249.32	443.58	229.15	182.90	**93.13**	16136	19605	26851	9866	5938	**5315**
50	8	4	23.05	23.05	22.69	22.69	**1.58**	7.26	217.40	250.88	1371.16	326.72	**100.75**	653.08	15880	24977	58418	19454	**6358**	28674
50	10	4	5.77	5.77	5.66	5.66	**0.62**	5.64	18.51	16.73	14.12	8.75	19.83	**5.49**	1125	898	658	622	968	**498**
50	6	6	30.68	30.68	30.34	30.34	**1.31**	5.78	85.60	**70.00**	108.38	91.01	85.15	92.55	5087	3068	4460	2661	**2254**	3648
50	8	6	9.27	9.27	9.27	9.27	**1.15**	9.27	46.20	58.52	70.34	126.16	**35.54**	43.66	1618	962	666	920	**410**	733
50	10	6	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	2.78	3.43	1.96	1.58	**1.49**	1.62	20	49	**0**	**0**	**0**	**0**
50	6	8	9.21	9.21	9.21	9.21	**0.44**	9.21	993.46	2832.36	(0.31)	997.20	106.86	**85.75**	13939	34035	38492	10581	974	**874**
50	8	8	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	71.84	73.22	**9.42**	26.15	17.15	23.76	845	786	20	50	**10**	400
50	10	8	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	10.22	6.66	**4.26**	5.42	5.18	6.82	220	59	**0**	**0**	**0**	190
100	6	4	54.24	54.24	53.65	53.65	**1.91**	**1.91**	193.65	351.52	270.66	303.63	138.82	**58.47**	16136	24076	15971	10857	7955	**2824**
100	8	4	23.05	23.05	22.69	22.69	**1.58**	7.26	217.40	2141.64	300.77	1272.10	**77.90**	446.48	15880	129486	23257	73610	**5558**	24236
100	10	4	5.77	5.77	5.66	5.66	**0.62**	5.64	17.30	22.74	11.46	20.61	15.67	**11.45**	978	1065	714	**653**	968	902
100	6	6	30.68	30.68	30.28	30.28	**1.31**	5.78	67.19	**63.94**	113.23	106.41	76.37	65.47	3114	2682	6621	2508	**1761**	2360
100	8	6	9.44	9.44	9.44	9.44	**1.32**	9.44	108.21	79.64	145.11	173.98	**52.91**	100.91	3842	2313	1113	1836	**521**	4664
100	10	6	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	11.27	14.70	2.25	2.51	**1.48**	5.31	418	492	**0**	**0**	**0**	89
100	6	8	9.78	9.78	9.78	9.78	**0.96**	9.78	(0.44)	(0.46)	(0.47)	(0.36)	2115.30	**2009.85**	29528	29173	31932	21391	**15394**	18291
100	8	8	0.88	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	(0.88)	1016.32	2016.47	686.79	**481.46**	1789.27	20365	11097	6476	1574	**1477**	10274
100	10	8	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	1.14	725.10	343.28	42.00	**11.03**	58.59	(1.14)	7686	3190	100	20	**19**	15575
150	6	4	54.24	54.24	53.62	53.62	**1.91**	**1.91**	113.95	233.72	227.44	259.32	95.73	**73.11**	7836	18333	12214	10286	**3105**	4149
150	8	4	23.05	23.05	22.69	22.69	**1.58**	7.26	1028.98	598.60	160.18	232.00	71.69	**18.17**	68400	32616	8133	13838	**4084**	5525
150	10	4	5.77	5.77	5.66	5.66	**0.62**	5.64	23.13	23.08	9.60	10.04	**8.53**	18.17	1200	1301	491	**466**	645	776
150	6	6	31.32	31.32	28.23	28.23	**1.76**	6.30	100.34	**52.67**	162.01	62.76	64.42	96.12	7090	1542	7745	1524	**1489**	2055
150	8	6	13.14	13.14	10.29	10.29	**4.12**	13.14	310.53	312.13	879.47	195.80	**107.40**	450.92	9465	3899	14650	903	**457**	4974
150	10	6	8.34	8.34	5.63	5.63	**5.55**	8.34	545.08	269.56	348.89	**157.98**	653.42	208.04	5125	2907	773	**438**	1353	2306
Average			16.71	16.71	16.19	16.19	**1.26**	5.11	512.58	528.53	579.71	371.14	**190.60**	420.15	10497	14525	10823	7669	**2571**	5806
Avg. Opt. Gap(%)									(0.06)	(0.02)	(0.03)	(0.02)	**(0.00)**	(0.05)						
# of Solved Ins.(/24)									22	23	22	23	**24**	23						
# of Best Solutions(/24)			5	6	6	6	**24**	8		3	2	2	**9**	8			3	6	**16**	4

Table A.6: LP gap, CPU time/ optimality gap and number of nodes of *CoverageVLP* in **B** type instances

Parameters			LP Gap (%)						CPU Time (sec.)/Optimality Gap(%)						Number of Nodes					
ρ_{min}	k^{max}	p	*CM0*	*CM1*	*CM2*	*CM3*	*CM4*	*CM5*	*CM0*	*CM1*	*CM2*	*CM3*	*CM4*	*CM5*	*CM0*	*CM1*	*CM2*	*CM3*	*CM4*	*CM5*
50	6	4	25.73	25.73	25.71	25.71	**1.19**	**1.19**	432.22	900.92	3165.87	1843.91	**150.42**	158.94	25535	48016	165456	43576	**5481**	7935
50	8	4	2.12	2.12	2.12	2.12	**1.73**	**1.73**	17.82	32.34	34.28	45.52	**22.19**	43.85	3541	8041	5304	5359	**3325**	7035
50	10	4	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.45**	0.50	0.81	0.94	0.91	0.46	**0**	**0**	**0**	**0**	**0**	**0**
50	6	6	9.33	9.33	9.32	9.32	**4.20**	4.40	786.90	485.58	725.56	1299.77	533.40	**261.94**	43195	15868	14914	19261	5390	**4445**
50	8	6	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	0.48	0.56	0.93	0.88	0.96	**0.46**	**0**	**0**	**0**	**0**	**0**	**0**
50	10	6	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.53**	0.67	0.90	0.96	0.73	0.67	**0**	**0**	**0**	**0**	**0**	**0**
50	6	8	3.07	3.07	3.01	3.01	**1.34**	2.54	**56.66**	62.70	96.86	144.78	82.68	105.36	3140	3299	2386	2468	**1164**	5045
50	8	8	0.04	0.04	**0.02**	**0.02**	**0.02**	0.04	2.06	1.45	**0.64**	0.79	0.82	1.38	163	40	**0**	**0**	**0**	20
50	10	8	0.04	0.04	**0.02**	**0.02**	**0.02**	0.04	5.09	4.43	**0.78**	1.02	1.00	6.75	456	418	**0**	**0**	**0**	486
100	6	4	25.73	25.73	25.71	25.71	**1.19**	**1.19**	342.98	379.13	477.49	1277.67	**91.70**	140.39	20271	19062	17330	30092	**2723**	7229
100	8	4	2.12	2.12	2.12	2.12	**1.73**	**1.73**	85.01	25.16	**14.24**	29.73	15.64	92.02	11960	3464	**1240**	4053	1927	27160
100	10	4	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	1.19	0.92	0.87	0.87	1.22	**0.56**	**0**	9	**0**	**0**	**0**	**0**
100	6	6	9.33	9.33	9.32	9.32	**4.20**	4.40	284.11	541.96	388.66	1703.63	475.63	**237.37**	10200	13656	10011	23266	**4724**	4902
100	8	6	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.42**	1.56	1.02	1.02	0.81	0.53	**0**	57	**0**	**0**	**0**	**0**
100	10	6	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	4.08	**0.60**	1.26	1.08	1.21	2.44	330	**0**	**0**	**0**	**0**	39
100	6	8	3.56	3.56	3.13	3.13	**1.76**	3.09	196.36	273.46	191.69	180.66	192.90	**170.01**	8225	17602	4506	2313	**1987**	8702
100	8	8	0.88	0.88	0.40	0.40	**0.38**	0.88	11.23	11.54	10.08	6.99	**1.41**	12.65	482	581	43	28	**0**	474
100	10	8	0.88	0.88	**0.40**	**0.40**	**0.40**	0.88	13.58	18.82	25.41	17.09	**4.97**	14.22	533	709	415	170	**12**	525
150	6	4	25.73	25.73	25.71	25.71	**1.19**	**1.19**	367.43	383.30	1142.58	2294.78	172.68	**91.51**	17763	18110	61517	64122	5660	**4664**
150	8	4	2.12	2.12	2.12	2.12	**1.73**	**1.73**	73.07	38.82	12.76	61.09	79.50	**11.40**	7779	4756	1314	6003	9639	**1095**
150	10	4	**0.06**	**0.06**	**0.06**	**0.06**	**0.06**	**0.06**	**1.48**	5.31	3.39	4.54	2.66	7.63	528	667	972	558	**230**	1002
150	6	6	9.33	9.33	8.99	8.99	**4.20**	4.40	572.66	417.78	585.98	1927.39	473.40	**267.09**	15987	10033	10880	19465	**4232**	6283
150	8	6	0.13	0.13	**0.06**	**0.06**	**0.06**	0.13	4.86	2.34	3.61	1.16	**1.06**	8.42	283	64	20	**0**	**0**	474
150	10	6	0.13	0.13	**0.06**	**0.06**	**0.06**	0.13	5.03	6.50	**1.32**	1.36	1.56	5.78	457	402	**0**	**0**	**0**	332
Average			5.02	5.02	4.93	4.93	**1.06**	1.24	135.70	149.85	286.96	451.98	96.23	**68.41**	7118	6869	12346	9197	**1937**	3660
Avg. Opt. Gap(%)									**(0.00)**	**(0.00)**	**(0.00)**	**(0.00)**	**(0.00)**	**(0.00)**						
# of Solved Ins.(/24)									**24**	**24**	**24**	**24**	**24**	**24**						
# of Best Solutions(/24)			7	7	12	12	**24**	13	5	1	4		6	**8**	5	3	9	10	**20**	**8**

Table A.7: LP gap, CPU time/ optimality gap and number of nodes of *CoverageVLP* in **C** type instances

Parameters			LP Gap (%)						CPU Time (sec.)/Optimality Gap(%)						Number of Nodes					
ρ_{min}	k^{max}	p	*CM0*	*CM1*	*CM2*	*CM3*	*CM4*	*CM5*	*CM0*	*CM1*	*CM2*	*CM3*	*CM4*	*CM5*	*CM0*	*CM1*	*CM2*	*CM3*	*CM4*	*CM5*
50	6	4	25.73	25.73	25.71	25.71	**1.19**	**1.19**	353.54	609.85	598.64	758.12	259.94	**92.84**	17552	34889	17870	20915	7610	**4460**
50	8	4	2.12	2.12	2.12	2.12	**1.73**	**1.73**	47.49	19.55	35.59	**11.86**	15.30	47.08	7082	2734	4079	1497	**1449**	5086
50	10	4	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.41**	0.42	0.97	0.92	0.99	0.52	**0**	**0**	**0**	**0**	**0**	**0**
50	6	6	9.33	9.33	9.32	9.32	**4.20**	4.40	565.19	**349.39**	510.66	2235.29	490.20	669.14	15258	10412	18273	36960	**5097**	8462
50	8	6	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	1.16	**0.51**	0.90	1.09	0.91	1.30	70	**0**	**0**	**0**	**0**	**0**
50	10	6	10.91	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	2.04	1.88	**0.98**	1.11	1.20	1.25	69	51	**0**	**0**	**0**	39
50	6	8	3.20	3.20	3.12	3.12	**1.46**	2.67	48.02	**36.32**	113.23	113.57	104.06	119.09	3000	1902	3122	1808	**1366**	6319
50	8	8	0.04	0.04	**0.02**	**0.02**	**0.02**	0.04	2.39	2.92	1.06	0.86	**0.70**	3.69	156	270	**0**	**0**	**0**	342
50	10	8	0.04	0.04	**0.02**	**0.02**	**0.02**	0.04	4.06	3.19	0.86	0.86	**0.80**	5.31	363	255	**0**	**0**	**0**	400
100	6	4	25.73	25.73	25.71	25.71	**1.19**	**1.19**	334.94	430.85	779.60	2111.85	243.26	**110.99**	15638	23082	27070	47622	7012	**4881**
100	8	4	2.12	2.12	2.12	2.12	**1.73**	**1.73**	22.64	36.73	**17.40**	38.84	107.88	45.60	6179	5577	**1087**	4738	11227	6980
100	10	4	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	0.56	**0.48**	0.87	0.86	0.88	0.54	**0**	**0**	**0**	**0**	**0**	**0**
100	6	6	9.33	9.33	9.32	9.32	**4.20**	4.40	269.34	637.17	633.62	1841.72	468.53	**186.58**	9873	14758	10579	26504	5138	**3494**
100	8	6	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	3.58	1.58	**1.17**	1.60	1.19	2.98	365	46	**0**	**0**	**0**	474
100	10	6	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	5.05	5.59	5.35	**1.24**	1.32	2.94	486	500	474	**0**	**0**	61
100	6	8	3.68	3.64	2.99	2.99	**1.82**	**1.82**	420.18	321.45	**206.31**	415.81	311.55	395.33	35763	16979	8808	4794	**3219**	18690
100	8	8	1.21	1.21	**0.64**	**0.64**	**0.64**	0.68	(0.23)	(0.05)	(0.28)	**41.16**	86.22	(0.31)	104629	69792	94567	**515**	952	97678
100	10	8	1.21	1.21	**0.64**	**0.64**	**0.64**	1.21	(0.32)	(0.31)	(0.29)	123.07	**85.88**	(0.32)	117575	65763	60112	1633	**1493**	118411
150	6	4	25.73	25.73	25.71	25.71	**1.19**	**1.19**	1801.76	345.84	447.40	1900.28	187.28	**105.06**	94078	17714	18728	54512	**4178**	5010
150	8	4	2.12	2.12	2.12	2.12	**1.73**	**1.73**	14.85	60.46	40.80	**13.44**	39.14	70.16	**1580**	5394	6144	1782	4801	8268
150	10	4	**0.06**	**0.06**	**0.06**	**0.06**	**0.06**	**0.06**	4.72	6.56	5.40	**2.53**	3.87	6.35	827	559	569	**161**	254	760
150	6	6	9.33	9.27	8.98	8.98	**4.20**	4.40	431.83	363.12	985.60	1289.40	697.38	**187.72**	10803	8963	23610	14116	5708	**3302**
150	8	6	0.13	0.13	**0.04**	**0.04**	**0.04**	0.13	2.34	1.72	2.53	**0.92**	1.04	7.82	141	50	20	**0**	**0**	240
150	10	6	0.13	0.13	**0.04**	**0.04**	**0.04**	0.13	**0.72**	1.60	1.34	1.46	1.37	0.81	**0**	23	**0**	**0**	**0**	**0**
Average			5.49	5.03	4.94	4.94	**1.09**	1.20	480.71	434.89	482.94	454.49	**129.62**	385.98	18395	11655	12296	9065	**2479**	12223
Avg. Opt. Gap(%)									(0.02)	(0.01)	(0.02)	**(0.00)**	**(0.00)**	**(0.03)**						
# of Solved Ins.(/24)									22	22	22	**24**	**24**	22						
# of Best Solutions(/24)			6	7	13	13	**24**	14	2	4	4	**6**	3	5	4	3	8	12	**15**	7

Table A.8: LP gap, CPU time/ optimality gap and number of nodes of *CoverageVLP* in **D** type instances