DYNAMIC AND ROBUST CAPACITATED FACILITY LOCATION IN TIME VARYING DEMAND ENVIRONMENTS

A Dissertation

by

JOAQUIN EMMANUEL TORRES SOTO

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

May 2009

Major Subject: Industrial Engineering

DYNAMIC AND ROBUST CAPACITATED FACILITY LOCATION IN TIME VARYING DEMAND ENVIRONMENTS

A Dissertation

by

JOAQUIN EMMANUEL TORRES SOTO

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Chair of Committee,	Halit Üster
Committee Members,	Guy L. Curry
	Benito E. Flores
	Eylem Tekin
Head of Department,	Brett A. Peters

May 2009

Major Subject: Industrial and Systems Engineering

ABSTRACT

Dynamic and Robust Capacitated Facility Location
in Time Varying Demand Environments. (May 2009)

Joaquin Emmanuel Torres Soto, B.S., Tecnológico de Monterrey, Chihuahua, México
M.E., Tecnológico de Monterrey, Chihuahua, México
Chair of Advisory Committee: Dr. Halit Üster

This dissertation studies models for locating facilities in time varying demand environments. We describe the characteristics of the time varying demand that motivate the analysis of our location models in terms of total demand and the change in value and location of the demand of each customer. The first part of the dissertation is devoted to the *dynamic* location model, which determines the optimal time and location for establishing capacitated facilities when demand and cost parameters are time varying. This model minimizes the total cost over a discrete and finite time horizon for establishing, operating, and closing facilities, including the transportation costs for shipping demand from facilities to customers. The model is solved using Lagrangian relaxation and Benders' decomposition. Computational results from different time varying total demand structures demonstrate, empirically, the performance of these solution methods.

The second part of the dissertation studies two location models where relocation of facilities is not allowed and the objective is to determine the optimal location of capacitated facilities that will have a good performance when demand and cost parameters are time varying. The first model minimizes the total cost for opening and operating facilities and the associated transportation costs when demand and cost parameters are time varying. The model is solved using Benders' decomposition.

We show that in the presence of high relocation costs of facilities (opening and closing costs), this model can be solved as a special case by the dynamic location model. The second model minimizes the maximum *regret* or opportunity loss between a *robust* configuration of facilities and the optimal configuration for each time period. We implement local search and simulated annealing metaheuristics to efficiently obtain near optimal solutions for this model.

To the memory of Father Enrique Lopez del Rio.

ACKNOWLEDGMENTS

I would like to thank my adviser Dr. Halit Üster, whose guidance made this dissertation possible. I would like to thank my committee members, Dr. Guy L. Curry, Dr. Eylem Tekin, and Dr. Benito E. Flores, for their comments and observations in reviewing the manuscript. I am thankful to the Industrial and Systems Engineering Department for giving me support during my doctoral studies. I am also thankful to the Office for Latin American Programs of Texas A&M University for giving me the opportunity of working as a graduate assistant. I am in debt with CONACYT (Consejo Nacional de Ciencia y Tecnología) and SEP (Secretaría de Educación Pública) for supporting my doctoral studies.

Finally, I would like to thank my family for their unconditional support since the very first day I decided to start this project. I would like to thank the love of my life, Rayna L. McKinley, for standing by me and giving me support during the difficult times.

TABLE OF CONTENTS

CHAPTER Page

LIST OF TABLES

TABLE Page

LIST OF FIGURES

CHAPTER I

INTRODUCTION

In general, facility location problems deal with the decisions of *where* to optimally locate facilities (factories, distribution centers, warehouses, schools, hospitals, etc.) and *how* to allocate customers to facilities such that the demand for some service or product is satisfied. Usually, these decisions are made by considering the associated costs (or profits) of satisfying the demand and the costs related to establishing (or operating) the facilities.

The conventional facility location models available in the literature assume that demand and costs are known and do not change by time. Once the facilities have been optimally located, they are assumed to remain sited regardless of how demand and costs may change in future periods. For this reason, the conventional location models are also called *single-period* or *static* location models. In practice, however, demand is unknown and is time varying. Also, the transportation and operation costs may increase or decrease from one period to another.

If the total demand for a given product or service is time varying, it might be necessary to relocate the facilities to meet the upcoming changes. An increase in the total demand for a given period may require opening new facilities, increasing the total capacity available to meet the additional demand and to reduce the associated transportation costs at the expense of opening and operation costs. Similarly, a decrease in the total demand in any given period may lead to the closure of some existing facilities to obtain savings on facility operation costs at the expense of the associated closing costs.

This dissertation follows the style and format of *Operations Research.*

The static location models can not provide an optimal configuration of facilities when demand is time varying. Observe that relocating the facilities in response to changes in the total demand can lead to a reduction in transportation costs. Several models have been developed in the literature to overcome this limitation; they are known as *dynamic* location models. Dynamic location models assume that demand and cost parameters are time varying. Over a given time horizon, they determine the optimal time and location of facilities in order to minimize the total costs (or maximize the total profits) for serving demand and for operating and relocating the facilities.

In some practical situations, the relocation of facilities may not be possible due to budget constraints. Such situations may arise in the public or private sectors where facilities like power plants, hospitals, schools, etc. are expected to be operating for a long period of time. Assuming that the total demand is time varying, a possible strategy is to determine a fixed configuration of facilities which will remain operational over the entire time horizon. This configuration should meet the time varying demand while minimizing the total transportation and operation costs over the entire time horizon.

When relocation is not allowed, another possible approach is to determine a *robust* configuration of facilities such that no matter the value of the parameters in future periods, the total cost will remain as low as possible. Observe that, in the absence of relocation costs, the best approach to follow is to use the static (single-period) location models and optimally determine the locations of facilities for each period on the time horizon. However, if relocation is not allowed, we can determine a robust configuration of facilities by minimizing the maximum difference (or deviation) in total cost between the robust configuration and the optimal configuration for each time period.

In this dissertation, we present mathematical models for dynamic and robust capacitated facility location problems in time varying demand environments. The dynamic location model considers the problem of finding the locations of facilities with limited capacity to satisfy the demand from a set of customers over a discrete and finite time horizon. The total demand for a single product is assumed to be time varying in a known way. There are fixed charges (or costs) associated with establishing (or opening) new facilities, operating existing facilities, and for closing existing facilities. Also, there is a transportation cost for serving the demand of customers from open facilities. The main objective of the model is to determine an optimal sequence for locating facilities to satisfy the time varying demand while observing the capacity restrictions over the time horizon. We denote this problem as Dynamic Capacitated Fixed Charge Location Problem (DCFLP).

We also present two location models considering a similar problem setting as in the DCFLP but without relocation of facilities. The first model determines a fixed configuration of facilities that minimizes the total costs for opening and operating facilities and for shipping demand from facilities to customers over the entire time horizon. We denote this problem as Dynamic Demand Capacitated Fixed Charge Location Problem without relocation (DDCFLP). In Chapter IV, we show that when relocation costs are considerably large, the DDCFLP can be solved by the DCFLP model as a special case.

The second model considers the problem of finding a configuration of facilities that minimizes the maximum regret or difference in total cost with respect to the optimal solution for each time period. We denote this problem as Robust Capacitated Fixed Charge Location Problem (RCFLP).

I.1. Research Contributions

The conventional facility location models ignore the time varying behavior of demand and cost parameters, regardless of how long the facilities are expected to remain operational. The dynamic and robust location models in this dissertation address this limitation by incorporating strategic decisions for locating facilities throughout the time horizon. In particular, we contribute to the literature in facility location as follows.

1. We develop a mathematical model for the DCFLP to determine the optimal time and location for establishing capacitated facilities (as well as the allocation of customers to facilities) in order to minimize the total cost, when demand and cost parameters are time varying. We present a Lagrangian relaxation-based algorithm as well as a Benders' decomposition framework to solve the model. The efficiency of the solution methods depends on the structure of the problem and characteristics of the input data. The Lagrangian relaxation algorithm shows to be efficient in solving problems where the expected number of open facilities is small, and the total fixed operation cost accounts for more than half the objective function value. The Benders' decomposition algorithm demonstrates to be efficient for problems with a large number of expected open facilities, and significantly more efficient when the total fixed operation cost represents the major portion of the objective function value.

2. We develop a mathematical model for the DDCFLP. The model determines a fixed configuration of facilities that minimizes the total cost when demand and cost parameters are time varying. We present a Benders' decomposition algorithm to solve this model.

3. We develop a mathematical model for the RCFLP. The model determines a robust configuration of facilities that minimizes the maximum difference in terms of total cost with respect to the optimal solution for each time period. We implement Local Search and Simulated Annealing metaheuristics to solve this model.

4. We conduct an empirical analysis that gives strategic insights for decision makers when dealing with location problems when the total demand is time varying, in a known way, and following an increasing, decreasing, or steady pattern.

I.2. Organization of the Dissertation

The remainder of this dissertation is organized as follows. In Chapter II, we present a review of the literature on dynamic and robust facility location. In Chapter III, we describe the characteristics of the time varying demand structures and the generation of random data for experimentation. In Chapter IV, we formulate and solve the DCFLP model and present an empirical analysis of the solution methods. In Chapter V, we present the formulation of the DDCFLP model and the solution approach. We show that when relocation costs are considerably large, this problem can be solved as a special case by the DCFLP model. In Chapter VI, we formulate the RCFLP model, discuss the solution methodology, and present computational results. Finally, in Chapter VII, we present conclusions and discuss areas of future research.

CHAPTER II

LITERATURE REVIEW

II.1. Introduction

In general, facility location problems can be classified into two groups: *continuous* and *discrete* location problems. Continuous (or planar) location problems consider the location of demand or customers and new facilities to be at any point on the plane. Distances between points are generally represented by norms. The ℓ_p norm is a commonly used norm for distance representation (Love et al., 1988); its special forms include $p = 2$ (Euclidean distance) and $p = 1$ (rectangular distance). On the other hand, discrete location problems consider the location of demand and facilities on the nodes and links of a graph or network (usually only at the nodes); the travel distances between demand points and facilities are represented by the arcs of the network. Complete surveys of facility location problems are provided by Brandeau and Chiu (1989), Francis and Mirchandani (1990), Drezner (1995b), Owen and Daskin (1998), and Drezner and Hamacher (2002).

The minisum and minimax location problems are classic location problems that have been formulated as continuous or discrete location models. The minisum problem has the objective of finding the location of a single or multiple facilities in such a way that the weighted Euclidean distances from a fixed number of points to the facilities are minimum. In the minimax problem, the objective is to determine the location of facilities such that the maximum distance from a set of points to the new facilities is minimum.

The classification of location problems can be further extended to consider certainty or uncertainty in the parameters. In certainty situations, the value of the parameters is assumed to be known. In uncertainty situations the value of the parameters in the future is unknown and several possible realizations or scenarios need to be considered. Location problems in uncertainty can be stochastic or robust. Stochastic location problems consider a probability distribution associated with each possible realization or scenario. In robust location problems, no probability distribution information is available and a set of possible scenarios needs to be considered. The main objective of robust location problems is to find the location of facilities that will have a good performance (cost wise) under future changes in the value of uncertain parameters. Common robustness measures used in the literature consider minimizing the maximum cost, minimizing the worst-case cost (or regret), and minimizing the maximum relative regret (or relative deviation) (Kouvelis and Yu, 1997).

In this dissertation, our main focus is on dynamic and robust location models. We assume discrete locations for facilities and customers. The demand and cost parameters are assumed to be changing by time in a known way. Thus, our location models are discrete and deterministic.

This chapter is divided into two main sections. In Section II.2, we review the literature in dynamic facility location. In Section II.2.2, we review the literature in robust facility location with special focus on solution strategies applicable to our robust location model. In Section II.4, we describe the position of this dissertation in the current literature. Finally, in Section II.4, we present a summary of the chapter.

Throughout this chapter we assume the reader is familiar with continuous and discrete location models. The interested reader is referred to Love et al. (1988), Drezner (1995b) or Daskin (1995) for an introduction to location theory.

II.2. Dynamic Facility Location

Dynamic facility location models can be considered to be extensions of the conventional (single-period or static) models as they include time varying demand. Most of the models developed in the literature for dynamic location problems assume that facilities can be relocated between periods in response to changes in demand. There are associated relocation costs for changing the location of facilities between periods, which can represent the initial investment for establishing new facilities and the cost (or savings) for the closure of existing facilities.

Most of the continuous and discrete static location problems are known to be NP-hard. Dynamic location problems are at least as difficult to solve as the static problems due to the additional consideration of time. However, the algorithms and solution approaches developed for static location problems can be adapted to solve the dynamic problems. Throughout this section we review the literature in dynamic facility location. We separate the dynamic models into continuous and discrete location models.

II.2.1. Continuous Location Models

Perhaps the first model that considers time varying demand and relocation of facilities is given by Ballou (1968). The dynamic location/relocation model considers a single warehouse where the objective is to maximize the total net profit along a finite and discrete planning horizon. The model is solved using the recursion formula of discrete dynamic programming (Bellman, 1966). The set of candidate locations for facilities is constructed from the optimal solutions of the static warehouse location problem for each period. This is a restriction on the dynamic programming procedure to work only with a fixed state space of alternative locations for each period. A drawback of this solution approach is that the set of candidate locations may exclude potential sites that can increase the maximum profit for the overall problem. Therefore, this approach can be considered to be a heuristic solution method.

A different type of problem is given by Scott (1971), introducing two different models for the single facility dynamic location-allocation problem. In this problem a single facility is to be located at the beginning of each period of a finite and discrete time horizon. It is assumed that the number of customers and facility locations are stationary and the transportation cost remains constant in subsequent periods after the end of the time horizon. The first model considers a myopic optimization process which does not anticipate the future. The minimum cost location of a single facility is determined only for the current time period, considering the existing (sited) facilities at that time, and continues in this fashion until the last facility is located. The second model uses dynamic programming to determine the over-all optimum taking into account future events. The recursion formula of discrete dynamic programming is used to determine the complete sequencing of the facility construction plan that minimizes the cumulative cost.

Wesolowsky (1973) presents a general multi-period formulation of the Weber problem (Weber and Friedrich, 1929). This dynamic formulation allows changes in the location of a single facility along a finite planning horizon. The demand, number of destinations (demand points), and the associated costs for serving demand and relocating the facility are assumed to be time varying. A dynamic programming algorithm is used to optimize the sequence of locations in order to meet changes in costs, volumes, and locations of destinations. The procedure is represented by a decision tree where each node represents a sequence of configurations for each time period, allowing the elimination of duplicated solutions. This *incomplete* dynamic programming algorithm reduces the number of static problems to be computed more than using complete enumeration. This algorithm is an optimal solution method for the problem presented by Ballou (1968).

In a subsequent paper, Wesolowsky and Truscott (1976) propose a dynamic multi-facility minisum problem. The model can be considered to be a reformulation of the previous model introduced by Wesolowsky (1973), by allowing the establishment of new facilities and the removal of both existing and new facilities within the planning horizon. In this formulation, the locations for a fixed number of facilities are assumed to be at any place on the plane. Changes in location, in response to changes in demands and costs, are permitted with an associated relocation cost. A *segmented bounded* algorithm is developed to solve the problem. The algorithm solves a series of static minisum problems, corresponding to segments of a binary matrix, and selects the least cost solution. This binary matrix specifies the movements of facilities along the planning horizon. Using this approach the total number of static problems to be solved is reduced considerably compared to using a complete enumeration approach.

Megiddo (1986) considers two types of dynamic 1-center problems: *global optimization* and *steady-state*. Demand points are assumed to be moving (or changing location) according to a linear function over the time horizon. The global optimization problem looks for a point in time when the solution to the static 1-center problem yields the best solution for all time periods. In the steady-state problem the objective is to determine the steady-state behavior of the system, i.e., a point in time when the location of the facility or center will remain unchanged in the following periods. The dynamic 1-center problem in the plane is used to solve both problems. Solution algorithms for the static 1-center problem are adapted to solve these dynamic location problems.

Drezner and Wesolowsky (1991) study the problem of locating a facility among a given set of demand points when the weights associated with each demand point change in time in a known way, and the location of the facility is allowed to change one or more times during the time horizon. The weight associated with each demand point is assumed to be a continuous function of time. The problem is to find the time breaks when the location of the facility must be changed as well as its location during the time span between breaks. Both minisum and minimax formulations are considered as dynamic location problems. The solution algorithm for the minisum problem considers rectangular distances; it scans all possible break points that can be optimal and selects the best. To solve the minimax problem, two algorithms are given using bisection search and considering Euclidean distances.

Drezner (1995a) presents a formulation of the dynamic p-median problem when demand is changing over time and new facilities are built at given times. This problem is called *progressive* p-median problem, since new facilities or medians are established sequentially in each period. The solution approach for the problem is derived using a numerical example. In this example, two new facilities are to be located considering Euclidean distances. This problem is solved by a special algorithm, similar to the 2-median problem given by Drezner (1984). The general problem is solved using standard nonlinear mathematical programming code. This type of p-median problem arises in situations where demand is increasing over time in a known way, such that new facilities need to be established sequentially at given time periods.

II.2.2. Discrete Location Models

The literature available in discrete dynamic location problems is considerably richer than continuous models. Usually, discrete location problems are formulated as integer or mixed integer programs and solved using the optimization methods developed for this type of mathematical models.

The restriction of facility locations to the nodes of a network is considered by

Erlenkotter (1974), introducing a network-based model for the single facility location problem. Each discrete time period in the planning horizon is represented by a node. The arcs of the network denote the relocation and operation costs of the facility between time periods. The optimal solution is obtained recursively by evaluating the minimum location policy cost discounted to time period over the time span between the periods at which the facility is relocated. This network solution approach is equivalent to the incomplete dynamic programming algorithm presented by Wesolowsky (1973) with discrete locations.

Kolesar and Walker (1974) present an application of the dynamic set covering location problem to the relocation of fire companies in New York City. The problem is divided into several stages and solved sequentially to determine the relocation plan that gives a coverage with minimum response time. This procedure considers the solution of a series of linear integer programs. A heuristic algorithm and a computer-based program are proposed to determine the best relocation plan.

A special type of p-median problem is presented by Wesolowsky and Truscott (1975). The model considers the multi-period discrete space location-allocation problem. The purpose of this model is to devise a plan of optimal locations and relocations in response to predicted changes in the demand volume originating at demand points over a finite planning horizon. The model is solved using a dynamic programming algorithm with backward recursion for small size problems.

Roodman and Schwarz (1975) give a dynamic model for the uncapacitated facility location problem. This formulation determines the time at which a set of initially open facilities are to be closed. Once a facility is closed it can not be reopened. This situation arises when demand is decreasing over the time horizon and facilities are supposed to be closed sequentially in each time period. The problem is solved by exploiting the special economic structure of the problem. The algorithm consists of *partial assignments* of customers to facilities and a modified branch and bound procedure, similar to the branching rules method given by Efroymson and Ray (1966) and Khumawala (1972). Also, a heuristic procedure is used together with the branching rules to obtain approximate optimal solutions.

Revelle et al. (1976) study a multi-period extension of the set covering problem. In this formulation it is assumed that the set of customers at each time period is a subset of the next period. The set of potential locations for facilities remains the same in each period of the planning horizon. The location pattern over time is obtained by solving the static set covering problem. Facilities are located only when they are required. A disadvantage of this model is that after the solution is obtained, a secondary procedure is required to find the time-phasing of facilities.

Sweeney and Tatham (1976) propose an improved model for solving the multi-period multiple warehouse location problem with opening and closing of capacitated facilities. This type of location problem was first discussed by Ballou (1968). The improved model integrates the mixed integer program formulation of the single warehouse location problem with a dynamic programming procedure for finding the optimal sequence of configurations over multiple periods. It is shown that only the best ranked-order solutions (ranked in nondecreasing order of the objective function value) in any single period need to be considered as candidates in the optimal multi-period solution. This consideration reduces the state space of the dynamic programming algorithm in a similar way to the solution approach given by Wesolowsky (1973).

Schilling (1980) presents an application of the dynamic maximum covering location problem for establishing emergency services. The mixed integer program formulation is an extension of the static maximum covering problem. This formulation requires that a certain number of facilities must be open at each period. In this model, the objective function is represented as a vector with the multiple objectives of maximizing the coverage in each period. The model is solved using a heuristic myopic procedure that evaluates alternative configurations between successive periods as long as the maximum coverage is improved. This heuristic is combined with a weighting method to evaluate dominating solutions for the problem.

Erlenkotter (1981) presents a comparison of seven approximate methods for a dynamic model of the CFLP considering both discrete-time and continuous-time frameworks. The general problem is to locate new capacity over time to minimize the total discounted costs of meeting growing demand at several locations. Due to the level of complexity of the problem, the use of mixed integer programming optimization methods does not guarantee that the optimal solution for practical size problems is obtained. Two industrial problems given by Manne (1967) are used to test the performance of these seven heuristic methods. The comparative results show that the type of formulation using discrete or continuous time affects the form of the solution. For discrete-time formulations the solutions tend to force the capacity expansions into multiples of individual period demand increments. In continuous-time formulations there is more flexibility to choose the size and restrictions of the capacity expansion. Concluding remarks point out that improved results can be obtained using a combination of the heuristic solution methods.

Chrissis et al. (1982) present a dynamic version of the set covering problem that considers the phase-in/phase-out cost of facilities. The model considers the facility operation and relocation costs. To determine the penalties or cost due to relocation, a logic constraint is added to the model. The objective of this model is to minimize the total number of facilities required to cover all the demand points over all time periods. The model is solved using an approximation algorithm.

Gunawardane (1982) introduces dynamic formulations of the set covering and maximum covering problems considering the phase-in/phase-out of facilities. The dynamic set covering formulation is based on the model discussed by Revelle et al. (1976). Since the dynamic model has the same number of constraints but more variables than the static set covering model, the solution procedure considers the linear programming relaxation. The optimal solutions obtained for a set of test problems display the integrality property. The dynamic formulation is extended to consider the individual cases of phase-in and phase-out of facilities. A shortcoming of the proposed dynamic formulations is that for practical size problems the integral solution of the linear programming relaxation is not guaranteed.

Van Roy and Erlenkotter (1982) study a dynamic location model similar to the model introduced by Roodman and Schwarz (1977). This model prevents the relocation of facilities, that is, opening a new facility at the most once and closing an initially existing facility at the most once. The solution method, denoted as DYNALOC, is a dual-based algorithm combined with a primal-dual adjustment procedure and a branch and bound algorithm. This solution approach is a modified version of the DUALOC procedure proposed by Erlenkotter (1978) for the static uncapacitated facility location problem.

An application of the solution approach proposed by Sweeney and Tatham (1976) is given by Kilmer et al. (1983). The purpose of this study is to determine the adjustments over time required in number, size, and location of citrus packing-houses in east Florida. It is assumed that the volume and location of production is changing by time. The mixed integer programming formulation does not consider opening/closing costs for facilities. Each single period problem is solved using a search code procedure. A dynamic programming algorithm is used to find the path adjustments of packing-houses over the planning horizon and to obtain the optimal configuration.

A different approach for the dynamic location problem is proposed by Kelly and Marucheck (1984). The problem considers that the optimal decision to open or close a facility at a given point in time would become suboptimal when the planning horizon is extended or when the problem parameters change in subsequent periods. The mixed integer programming model determines the set of warehouse locations for each time period of a finite time horizon. This model incorporates the facility operation, opening, and closing costs. The solution methodology consists of obtaining a partial optimal solution by a bounding procedure similar to the delta and omega tests proposed by Efroymson and Ray (1966) and Khumawala (1972). The reduced model is then solved using Benders' decomposition. An optimal solution is later examined to determine if post horizon conditions could affect the location decisions. The purpose of this model is to determine the optimum relocation plan of facilities considering post-horizon conditions.

Chand (1988) considers the single facility location/relocation problem in an infinite time horizon. The location/relocation decisions are defined under the concepts of decision horizons/forecast horizons (DH/FH), i.e., the length of time where initial location decisions are to be taken (DH), and the number of time periods of forecasted data (FH) needed to make such decisions. The main objective is to determine the minimum number of periods needed to optimally define the DH as well as the FH for an infinite time horizon. A forward dynamic programming algorithm is developed to determine the optimal initial decisions within a finite horizon to determine the optimal location of a single facility.

Frantzeskakis and Watson-Gandy (1989) consider the problem of finding a location plan over a planning horizon, which selects the location of facilities in each period in such a way that the total costs of transportation, operation, and relocation are minimized. The problem is formulated as a dynamic program, restricting the number of open facilities in each period. The problem is solved using dynamic programming and a branch and bound procedure with state space relaxation.

Hakimi et al. (1999) study the 1-median and k-median problems on a time varying or dynamic network. Time is considered a discrete variable and the parameters of the network (demands at the vertices and lengths of the arcs) are known functions of time. The location of the facility during each time period can be a point along some edge on the network; this choice may or may not change in the next period. The 1-median problem is to find the locations of the facility along the time horizon that minimizes the cost of servicing demand and relocating the facility. The dynamic k-median problem is defined in a similar way but to find the locations of k facilities on the dynamic network. The dynamic 1-median problem can be solved using an augmented graph, computing the shortest path between locations. The k-median problem can be solved in a similar way by successively solving the problem for each time period and finding the k locations that minimize the total cost.

The problem of capacity expansion and dynamic plant location is presented by Shulman (1991). This class of dynamic capacitated location problem considers different types of facilities with finite capacities. The objective is to find the optimal facility expansions (or mix of facilities) at each location when more than one facility can be placed at a given location. The problem is formulated as a mixed integer program. This formulation is solved using Lagrangian relaxation of the capacity constraints. This type of relaxation simplifies the problem into small optimization subproblems, one for each candidate facility location. These subproblems are solved for fixed values of Lagrangian multipliers using dynamic programming. Two solution algorithms are designed. The first solves the general dynamic problem considering different types of facilities. The complexity of this algorithm is exponential in the number of facilities and may be used only for small problems. The second algorithm considers the case where different types of facilities can not be located at the same location. This algorithm has a polynomial complexity and can be used for large problems.

Bastian and Volkmer (1992) study a similar problem considered by Chand (1988). A perfect forward procedure is developed to determine the optimal initial decision by using only information from the smallest forecast horizon. Given an infinite planning horizon problem, it may be possible to find the optimal initial decision by using only information from a finite number of periods. The fixed cost of relocating the facility may depend on the period as well as the locations. The solution algorithm uses the data structure of a policy tree which is adapted from the lot tree approach for solving dynamic lot size problems.

Most of the dynamic location models consider the planning horizon as an exogenous input. Daskin et al. (1992) consider a dynamic location model in which the objective is to find a planning horizon (optimal forecast horizon) and a first period decision (optimal initial decision) such that the conditions after the planning horizon do not influence the choice of the optimal initial decision. This approach suggests that the planning horizon for a dynamic location problem should be determined endogenously. Using an empirical approach it can be determined whether or not forecast horizons are likely to exist. The concepts of ϵ-optimal forecast horizon and the ϵ-optimal initial decision are introduced. For given empirical tests, it is shown that good initial decisions and empirical ϵ-optimal forecast horizons could be found for small size problems.

Andreatta and Mason (1994) present a note regarding the work of Bastian and Volkmer (1992). This note refers to the previous work of Chand (1988) about the perfect forward algorithm for the solution of the single facility dynamic location/relocation problem. A numerical example is solved to demonstrate that this problem does not always have a finite forecast horizon. The perfect algorithm that is presented differs from the policy trees proposed by Bastian and Volkmer (1992) and the regeneration sets used by Chand (1988). Instead, this algorithm uses all possible ending states. This approach can be viewed as a modified version of the Dijkstra's algorithm.

Chardaire et al. (1996) give a quadratic programming formulation for the dynamic uncapacitated facility location problem. The model is solved using Lagrangian relaxation and Simulated Annealing. The Lagrangian subproblem is solved using dynamic programming to optimality. The set of open facilities is given as an input to Simulated Annealing to find a good feasible solution. It is shown that the bound obtained by the Lagrangian dual is equal to the bound obtained from the linear programming relaxation of the linearization of the quadratic model.

Location problems can be extended to the case where facilities can be established in different geographic regions and operate under different economic environments. Canel and Khumawala (1996) present mixed integer programming formulations for the capacitated and uncapacitated multi-period international facility location problem (IFLP). This class of location problem is similar in purpose to the location model studied by Ríos-Ramírez (2003). In addition, the IFLP incorporates the quantitative characteristics of locating facilities in foreign countries and the economic implications. These economic considerations include factors such as international customers and competition, market access and proximity, lower labor costs, economies of scale, taxes, incentives, inflation rates, and so on. The IFLP arises in situations where companies respond to the external environment and seek advantage available at international locations. The objective of the multi-period IFLP is to determine in which countries to locate facilities, the timing for the location decisions, and the quantities to be produced and shipped to the customers such that either total costs are minimized or total after-tax profits are maximized. The structure of the model considers the existence of a domestic plant and facilities that can be located in foreign countries to supply the demand of customers in a global market. Both mixed integer programming formulations are compared with an actual company case given in the literature. The models are solved using standard optimization software. Also, a sensitivity analysis is conducted to evaluate alternative plans for the problem.

Hormozi and Khumawala (1996) give an exact algorithm for the multi-period facility location problem. The mixed integer programming formulation corresponds to the multi-period, multi-stage facility location problem, incorporating opening and closing costs for facilities. The model considers a set of plants with limited capacity that can serve customers and facilities. The solution algorithm is based on the method presented by Sweeney and Tatham (1976) and provides an improvement over this procedure by reducing the computational requirements. Two simplification procedures are introduced to reduce the size of the general facility location problem (improved lower bound and delta/omega augmentation). This algorithm considers a rank-ordered number of solutions to static problems for each period of the planning horizon. Dynamic programming is used to obtain the optimal sequence of facility configurations that minimizes the total cost. The reduction techniques reduce the number of single period problems that need to be considered by the dynamic programming part. The proposed improved algorithm required fewer single period problems and took less computational time when tested and compared to the procedure of Sweeney and Tatham (1976).

Canel and Khumawala (1997) propose a branch and bound algorithm to solve the multi-period IFLP. The mixed integer programming formulation incorporates quantitative factors such as demand, investment cost, manufacturing and labor costs, transportation and transfer costs, taxes and tariffs, exchange rates, plant equipment and fixed costs. The proper calculation of the relevant costs impact the efficacy of the model. The branch and bound algorithm uses the simplifications and branching rules given by Khumawala (1972) and Hormozi and Khumawala (1996). Using the data from the previous case of study (Canel and Khumawala, 1996), the formulation is solved for uncapacitated and capacitated problems using the branch and bound algorithm.

Saldanha da Gama and Captivo (1998) give a discrete dynamic formulation for the uncapacitated fixed charge location problem (UFLP) that considers fixed costs for operating, opening, and closing facilities. Opening/closing of facilities is limited to take place at most once for each period except for the last period of the time horizon. The model is solved using a two-phase heuristic. The first phase consists of a modification of the drop procedure introduced by Kuehn and Hamburger (1963). The procedure begins with all facilities open for all periods and iteratively takes out periods in the operation of some facility until no further elimination is possible without losing feasibility. In the second phase, local search is applied using a radius-k neighborhood with $k = 1$. The neighborhood of a feasible solution is defined as the set of different feasible solutions with the addition or removal of no more than k operation periods in some facility. The purpose of local search is to adjust the initial feasible solution obtained in the drop phase. To test and compare the performance of the two-phase heuristic a computational experiment is presented. A comparison between the heuristic method and the solution approach proposed by Van Roy and Erlenkotter (1982) showed that the heuristic obtained good results in computing time and solution quality.

Canel and Das (1999) consider the multi-period facility location problem with profit maximization. The objective function considers the fixed and investment costs for locating facilities, transfer and manufacturing costs for transportation charges, and the revenue for sent quantities from facilities to customers. The mixed integer programming formulation is solved using an implementation of the branch and bound algorithm and simplification rules given by Efroymson and Ray (1966) and

Khumawala (1972). For a series of test problems, the results obtained showed that the proposed algorithm is efficient in obtaining optimal solutions in short time when compared with standard optimization software.

In a research report given by Dias et al. (2001a), three types of dynamic capacitated location problems with opening, closure, and reopening of facilities are presented. The first type of problem considers facilities with a maximum capacity at the opening period. This maximum capacity remains constant during the operating time of the facility. The second problem considers a maximum and minimum capacity for facilities. The third problem considers facilities with maximum decreasing capacity at the opening period or a maximum expansion at the reopening period. It is assumed that capacity decreases as customers are assigned to the facility. For this third problem, the possibility for a facility to be closed even if its available capacity has not been depleted is considered. If this facility is reopened in a subsequent period it will, in addition, have its remaining capacity from when it was closed. For each type of problem a mixed integer program and its associated dual formulation are given. Primal-dual heuristics are used to solve each type of problem. These heuristics are based on the work of Erlenkotter (1978) and Guignard-Spielberg and Spielberg (1977). The procedure uses a dual ascend, dual adjustment, a primal procedure, and dual descent procedure for each dual variable. A numerical example is solved for each problem to illustrate the performance of the heuristics.

Dias et al. (2001b) present a hybrid heuristic algorithm to solve capacitated and uncapacitated dynamic location problems. This research report considers the model previously discussed by Dias et al. (2001a). The formulation of a general mixed integer programming model is extended to consider opening, closing, and reopening of four types of facilities: uncapacitated facilities, facilities with maximum and/or minimum capacity, facilities with maximum decreasing capacity, and facilities composed of one or more elements of different dimensions (capacities). The general formulation is also extended to consider additional restrictions and multi-objectives. The heuristic solution method integrates genetic algorithm with local search. The genetic algorithm phase works with the generation, diversification, and evolution of solutions (individuals). A binary matrix is used to represent the opening, closing, and reopening of facilities in each time period (chromosomes). The genetic operators used are selection (binary tournament), crossover (adaptation of one-point crossover), and mutation (probabilistic). The local search phase works with one solution at a time to improve its fitness (objective function value) by searching k-neighborhoods. A k-neighborhood consists of different solutions obtained by inserting or extracting k operating periods to a facility. The genetic algorithm considers a random initial configuration of open facilities (population) and generational replacement with elitism. This hybrid algorithm is extended to include additional restrictions or multi-objectives in the formulation.

Dias et al. (2004a) develop a model for dynamic location problems with discrete expansion and reduction sizes of capacity. This model is similar to the model given by Shulman (1991), since facilities of equal or different capacities can be established at the same location. In addition, this model considers opening, closing, and reopening of facilities more than once along the planning horizon. The mixed integer programming formulation is similar to the model presented by Dias et al. (2001a) (maximum capacity case). However, the model is adapted to consider facilities with different discrete capacities. The primal-dual heuristic is very similar to the solution approach previously discussed by Dias et al. (2001a). The concluding remarks mention that the results obtained using this approach can be improved by incorporating local search for improving the primal solution.

Dias et al. (2004b) extend the application of the primal-dual heuristic proposed by Dias et al. (2001a) to dynamic multi-level capacitated and uncapacitated location problems. This research report presents mixed integer programming formulations for several dynamic uncapacitated and capacitated multi-level location problems. These models consider the possibility of a facility being opened, closed, and reopened more than once during the planning horizon. Two types of capacity restrictions are considered, maximum capacity and maximum and minimum capacity but without flow conservation at the intermediate facilities. For each dynamic multi-level problem a dual formulation and the complementary conditions are derived to define the primal-dual procedure (Erlenkotter, 1978).

Balakrishnan (2004) extends the work of Hormozi and Khumawala (1996) by proposing a pruning rule for the multi-period facility location algorithm. The use of this rule can reduce the number of single period configurations to be considered by the dynamic programming algorithm. The same example given by Hormozi and Khumawala (1996) is solved to illustrate the effectiveness of the pruning rule. Also, an experiment is conducted to test its general effectiveness. The additional computational effort to implement it is minimal. This type of reduction is possible by considering each period independently. Separating the material flow cost and the location configuration rearrangement cost, some configurations with low material flow cost within a period can be ignored from consideration in the dynamic programming algorithm. This occurs if these configurations have a rearrangement cost not greater than their higher material flow cost.

II.3. Robust Facility Location

As we mentioned before, robust location problems provide solutions with acceptable results when the future value of parameters is uncertain. Robust location problems can be classified according to characteristics of uncertain parameters, for instance, discrete scenarios are used when no probability distribution is known. The robustness of a solution represents a measure of a decision under uncertainty. Typical measures of robustness discussed in the literature include the minimization of the maximum cost (or minimax), minimum worst-case deviation cost (or minimax regret), and minimization of the maximum relative regret (or minimax relative deviation).

To illustrate each one of these robustness measures consider the following notation. Let $x_\ell, \ell = 1, \ldots, k$ be the decision variables, X the set of feasible solutions, $s \in S$ the possible scenarios, $w_{\ell s}$ the cost for decision variable ℓ in scenario s, and Z_s^* the optimal total cost for each scenario.

In the minimax approach, the objective is to find a solution for which the maximum cost over all possible scenarios is minimum:

$$\min_{x \in X} \max_{s \in S} \sum_{\ell=1}^{k} w_{\ell s} x_\ell \tag{2.1}$$

In the minimax regret approach, on the other hand, a solution is defined as robust if it minimizes the maximum difference or deviation over all scenarios with respect to the optimal solution for each scenario:

$$\min_{x \in X} \max_{s \in S} \left\{ \sum_{\ell=1}^{k} w_{\ell s} x_\ell - Z_s^* \right\} \tag{2.2}$$

Finally, the minimax relative regret approach considers the case when the difference between the robust configuration and the optimal solution for each scenario varies considerably. Thus, the ratio between the difference or deviation and the optimal solution for each scenario is used instead:

$$\min_{x \in X} \max_{s \in S} \left\{ \left(\sum_{\ell=1}^{k} w_{\ell s} x_\ell - Z_s^* \right) / Z_s^* \right\} = \min_{x \in X} \max_{s \in S} \left\{ \left(\sum_{\ell=1}^{k} w_{\ell s} x_\ell / Z_s^* \right) - 1 \right\} \tag{2.3}$$

Robust location problems with a minimax objective function are more difficult to solve than problems with minimization (or maximization) objective function and require higher computational effort. For this reason, most of the models developed in the literature consider small size problems (usually 1-center or 1-median on a tree).

The literature available in robust facility location is quite recent and scarce, compared to the literature developed in dynamic facility location. The interested reader in robust optimization is refereed to the book of Kouvelis and Yu (1997).

Averbakh and Berman (1997) consider a minimax regret p-center problem on a general network. The weights or demands at the nodes of the network are assumed to be uncertain. The value of the demands is estimated using an interval. To solve the problem, a polynomial time algorithm is developed. This algorithm solves n static p-center problems, one for each node on the original network, and one in an auxiliary network. For the 1-center problem on a general network and the p-center problem on a tree, the solution time of the algorithm is shown to be polynomial.

Daskin et al. (1997) study a variant of the minimax regret p-median problem on a network. In this problem a probability is assigned to each scenario and only a subset of the scenarios is selected such that the total probability is at least a predefined value, α. The model minimizes the maximum expected regret over the selected scenarios. This approach is denoted as α-reliable since the regret of the selected scenarios will be bounded by the solution obtained from the model. A computational experiment is performed using commercial optimization software to test the model for different values of α.

Current et al. (1998) introduce two approaches for the dynamic p-median problem when the total number of facilities to be located is uncertain (NOFUN). The problem is analyzed by two different criteria: the minimization of expected opportunity loss and the minimization of maximum regret. In general, these criteria assume that there are a finite number of options and a finite number of possible states of nature. For each scenario there is a possible initial configuration of open facilities, each with a given probability of resulting in the final state configuration. The optimal solution for the problem may consider the restriction that the initial configuration is a subset of the final configuration. A solution with minimum expected opportunity loss can be obtained by solving a binary integer formulation. The solution of NOFUN problems by the minimax regret criterion does not consider the probabilities for the various states of nature. The optimal set of open facilities for the initial configuration is obtained by minimizing the maximum difference between the optimal solution of the p-median (without the restriction of having the initial configuration in the final state) for each possible state of nature, and the optimal solution of the p-median (with the restriction) for each potential state of nature and for each of the potential initial siting configurations.

Serra and Marianov (1998) present minimax and minimax-regret discrete location models for the p-median problem when demand and travel times or distance are uncertain. The application of the models is to locate fire stations in the city of Barcelona, Spain. Both models consider several possible scenarios to select the set of locations that will perform well over all future scenarios. The initial solution for both models is obtained by constructing a matrix with the optimal solutions of the static p-median for each scenario. The heuristic algorithm proposed for both models considers an exchange heuristic to improve the initial solution.

Vairaktarakis and Kouvelis (1999) study several formulations for the 1-median problem on a tree. These formulations consider dynamic change and uncertainty in the demand and transportation costs over a discrete and finite time horizon. Dynamic demand at the nodes and transportation costs in the arcs' length are represented by linear functions. The uncertainty in demand and transportation cost are represented by scenarios. The robustness measures considered are minimax regret and relative regret. For all the models a polynomial algorithm is developed.

Averbakh and Berman (2000) consider the 1-center problem on a tree where the demand or weights at the nodes and the arcs' lengths are modeled as uncertain parameters. The value of the uncertain parameters is assumed to be random, with unknown probability distribution, and is estimated within a given interval. The objective is to find the location of the center that minimizes the maximum regret over all possible scenarios. A polynomial time algorithm is developed. For the special case where the weights are certain and equal for all points, the complexity of the solution algorithm reduces significantly.

Averbakh (2000) study a group of combinatorial optimization problems with minimax objective function and uncertain parameters. The methodology to find minimax regret solutions consists of reducing the problems with uncertainty into a series of deterministic problems. The solution algorithms for the deterministic problems are then used to obtain efficient algorithms for the uncertain problems. The optimization problems solved consider minimax regret bottleneck combinatorial optimization problems, minimax multi-facility location problems, and maximum weighted tardiness scheduling problems.

Carrizosa and Nickel (2003) introduce the concept of p-robust location for the single facility minisum problem. In this problem demand is assumed to be uncertain and only an estimate value is known. The total transportation cost should never exceed a predefined value, in which case it will become inadmissable. The robust solution must find a location with the largest minimum difference, between the value of demand and its estimate, such that the total transportation cost becomes inadmissable. An iterative algorithm is developed for the general formulation and a search procedure for the case of rectangular distances.

Averbakh (2005) study the 1-median problem on a network with uncertain demand or weights of nodes. The uncertainty of the weights is estimated using an interval. The location of the center is to minimize the maximum relative regret over all possible scenarios. The solution obtained using the relative regret is significantly different from the absolute regret, thus it requires a special solution algorithm. For a general network, a polynomial time algorithm is developed through the structural properties of the problem. For the 1-median on a tree and a path, polynomial time algorithms are also given.

Snyder (2006) provides a survey of the literature in stochastic and robust facility location models and their applications. Robust location problems with special structure, such as the 1-median and 1-center problem, have been studied rigourously since the development of algorithms is computationally feasible. General location problems, such as the p-center and p-median problem, are more difficult to solve and only heuristic algorithms have been developed in the literature. The main contribution of this review is the analysis of different robustness measures and their applications.

Snyder and Daskin (2006) introduce stochastic robust location models for the UFLP and k-median problem. Demand and transportation costs are assumed to be uncertain. A probability distribution is associated to discrete scenarios for the uncertain parameters. The models consider the minimization of the total expected cost; a new robustness measure is introduced, denoted as p-robustness, which restricts the relative regret for each scenario to be within a given value. The main issue with this approach is that feasible solutions may be difficult to find when the value of p is small. A variable splitting (or Lagrangian decomposition) algorithm together with a branch and bound procedure is proposed to solve the stochastic and robust location models. For the stochastic p-robust-k-median problem, the split is performed on the demand variables, and in the p-robust-UFLP on both location and demand variables. A heuristic procedure is developed to solve the modified formulations of the stochastic models to consider a minimax-regret objective function.

II.4. Positioning in the Current Literature

This dissertation can be positioned in the current literature in dynamic and robust facility location with the following contributions:

- Our mathematical model for the DCFLP is novel in considering the opening and closing of facilities with associated fixed costs for opening, operating, and closing facilities. Most of the models developed in the literature consider only a single cost for relocation. In practical cases, there is a cost associated with establishing new facilities, a cost for operating existing facilities, and a cost (or saving) for the closure of existing facilities.

- Our mathematical model for the DDCFLP is novel in considering time varying demand and cost parameters to determine the optimal location of facilities when relocation is not allowed. The model includes the fixed costs for opening and operating facilities. In the literature, location problems without relocation are considered as static location models and ignore the time varying characteristics of demand and cost parameters.

- Our model for the RCFLP is novel in considering a general problem where the number of facilities is not fixed or given. The models developed in the literature consider special cases involving the location of a single facility on a tree or network.

- We consider different demand structures with attributes described by the behavior of the total demand and the change in value and location of the costumers' demand, based on a region or geographical location. These types of demand structures are the motivation for the analysis and development of our dynamic and robust location models. Most of the models studied in the literature consider a single demand structure.

II.5. Summary

In this chapter, we reviewed the literature in dynamic and robust facility location. The models developed for dynamic location problems determine the optimal location plan when demand and cost parameters are time varying. The application of dynamic location problems consider a wide variety of optimization problems in both the public and private sectors. The solution methods for dynamic problems rely on the methods derived for the static location problems. We found a richer variety of dynamic location problems studied in the literature compared to robust problems. The literature in robust location problems is quite recent and studies robust models using different robustness measures. The application of robust location models considers decisions under uncertainty, where the decision maker needs to evaluate several possible scenarios. Most of the solution methods developed for robust models consider small size problems, for which efficient algorithms are available, and make use of heuristic solution methods for practical size problems. For both dynamic and robust location problems, the time varying characteristics of demand and cost parameters are important and they give a motivation to the development and study of these type of location problems.

CHAPTER III

TIME VARYING DEMAND AND COST PARAMETERS

In this chapter, we describe the demand structures and cost parameters considered in the analysis of our facility location models. The characteristics of the demand, described in terms of the structure of the total demand and the change in value and location of each customer's demand, motivate the development of our location models, as well as the methodology used to generate random data to test the performance of our solution algorithms. This chapter is organized as follows. In Section III.1, we give a description of each demand pattern and the method used to randomly generate the demand for each customer location. In Section III.2, we describe the method to generate the capacity for facility locations. In Section III.3, we give a description of the method to generate the random cost parameters. Finally, in Section III.4, we present a summary of the chapter.

III.1. Total Demand Structures

The dynamic and robust location problems considered in this dissertation assume that demand and cost parameters are changing by time, in a known way, over a discrete and finite time horizon. The total demand is associated with a group of customers that have a known requirement for a single product along the time horizon. Assuming that all demands need to be satisfied in each period, facilities need to be established accordingly. Shipping demand from facilities to customers incurs a transportation cost proportional to quantity and distance. Also, there are fixed costs for establishing, operating, and closing the facilities. The possible locations for establishing the facilities and the available capacity at each location are assumed to be known for each period.

Observe that, if the total demand in any given period surpasses the total capacity available, then in order to satisfy the demand new facilities must be established. The decision to establish new facilities needs to consider the trade-off between paying the additional fixed costs associated with operating existing facilities, opening new facilities, and the possible decrease or savings in total transportation cost. On the other hand, if the total capacity in any given period surpasses the total demand, then to decrease the associated operation costs, some of the existing facilities can be closed. The decision to close facilities needs to consider the trade-off between the fixed costs associated with closing existing facilities and the possible savings in total operation cost. Finally, if the total demand in any given period is stable or has a minimum level of variation from the previous period, then the existing facilities can remain operational, incurring only the associated variable transportation and fixed operation costs for that period.

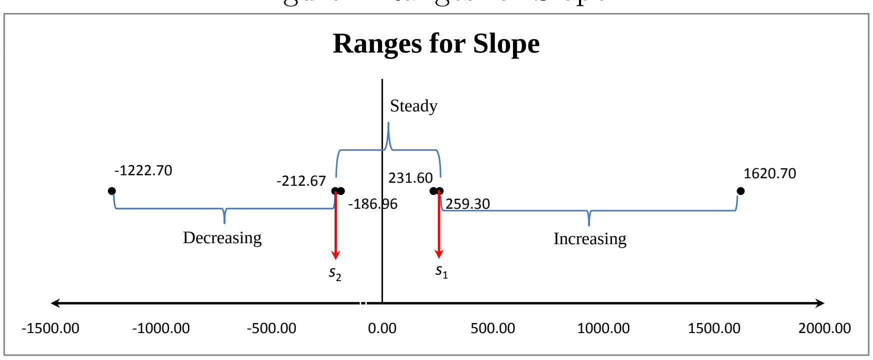
We note that these location decisions are driven by fluctuations or changes in the total demand. In particular, we identify three possible patterns in the behavior of the total demand: increasing, decreasing, and steady.

Let I denote the set of customer locations, indexed by $i = 1, \ldots, n$, let T denote the set of periods in the time horizon, indexed by $t = 1, \ldots, \tau$, and let J denote the set of possible facility locations, indexed by $j = 1, \ldots, m$. We assume that $m = n$, i.e., each customer location is a candidate facility location. Let w_{it} denote the demand of customer location i in period t, and let $D_t = \sum_{i \in I} w_{it}, t \in T$, be the total demand in period t.

We describe the three total demand structures by the value of the slope or rate of change of the total demand between periods using linear regression. The slope, σ, of the linear regression equation is computed using the ordinary least squares method.

We define scalars $s_1 > 0$ and $s_2 < 0$, such that $\sigma \geq s_1$, when the total demand is increasing; $\sigma \leq s_2$, when it is decreasing; and $s_2 < \sigma < s_1$, when it is steady. The value of σ is obtained using the linear regression model $G_t = \sigma D_t + h$, where G_t is the dependent variable and h its intercept, for each total demand data set. The values of s_1 and s_2 are obtained as follows. For each total demand structure, we find the range $[\ell, \upsilon]$, where $\ell = \min \sigma_k$ and $\upsilon = \max \sigma_k$ for each instance k considered in the experiments. The values of s_1 and s_2 are the ranges or break points that separate each demand pattern as shown in Figure 1. Figures 2, 3, and 4 show the slope values for a total of 40 instances for each total demand structure. These 40 instances belong to four classes of problems with $n = 50$ and 100 locations and $\tau = 5$ and 10 periods, considering 10 instances per class.

Figure 1 Ranges for Slope



In addition to the behavior of the total demand, we consider the way in which the demand of each customer might change. In practical situations, the resources available at a particular geographical region may be unevenly distributed. The population in a city can be distributed in such a way that regions with a larger population can have a higher demand for services and goods. Also, the demand for a certain product (or service) can change due to the introduction of a new product, marketing campaigns, or and increase or decrease in the price of the product. To service the high

Figure 2 Slope Value of Instances with Total Increasing Demand

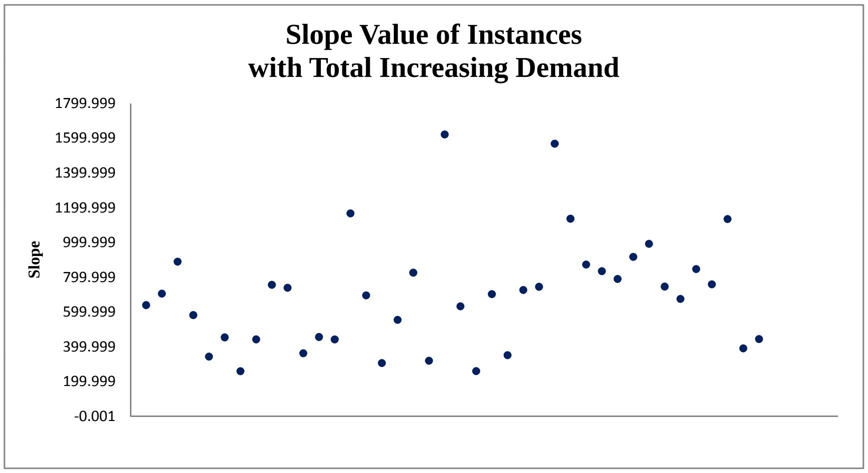
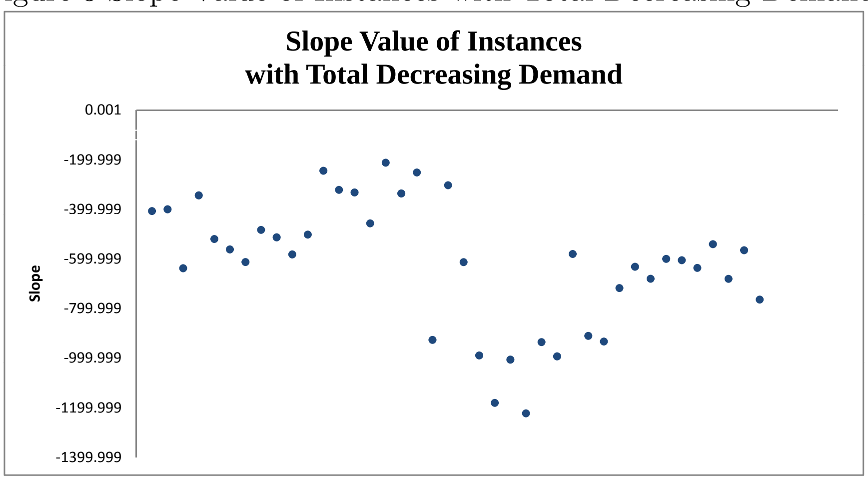


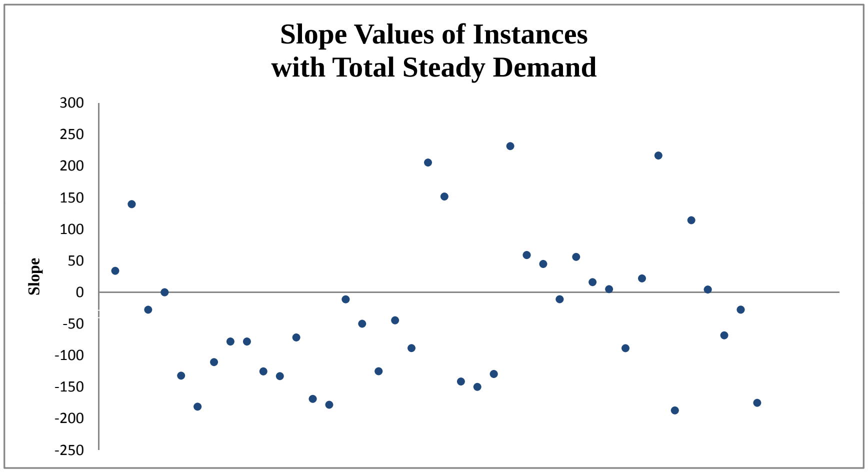
Figure 3 Slope Value of Instances with Total Decreasing Demand



level demand in this region, new facilities such as power plants, convenience stores, schools, hospital, fire stations, etc. need to be established. In regions with lower levels of demand for services and goods, it would be possible to observe the closure or relocation of facilities since they are more needed in regions with higher demand. This observation leads us to consider a possible shift in the demand of the customers from a particular geographical region to another along the time horizon.

Thus, in addition to these three possible total demand structures, we consider

Figure 4 Slope Value of Instances with Total Steady Demand



a shift in the value and location of the demand of each customer. We describe this shift according to the geographical region in which each customer is located. This composite structure of demand not only requires the relocation of facilities due to fluctuations along the planning horizon, but also it will determine the establishment of new facilities closer to regions where the concentration of demand is higher, and the closure of facilities where demand is lower.

To generate the demand, we first generate the customer locations. For each customer location i, we randomly generate an integer valued pair (x_i, y_i) of coordinates; each coordinate is uniform distributed, $x_i \in U[0, 150]$, $y_i \in U[0, 100]$, thus the coordinates of the customer locations are restricted to a rectangular area (150×100). Figure 5 shows an example of the customer locations for $n = 50$. For computational purposes, we do not allow two customer locations to have the same coordinates.

Once we have the customer locations, we evenly divide the x-axis into three regions, say A, B, and C. Each customer location is assigned to a region based on its x-coordinate as follows, region A if $x_i \in [0, 50)$, region B if $x_i \in [50, 100)$, and region C if $x_i \in [100, 150]$. Figure 6 shows an example for $n = 50$ locations.

Figure 5 Customer Locations

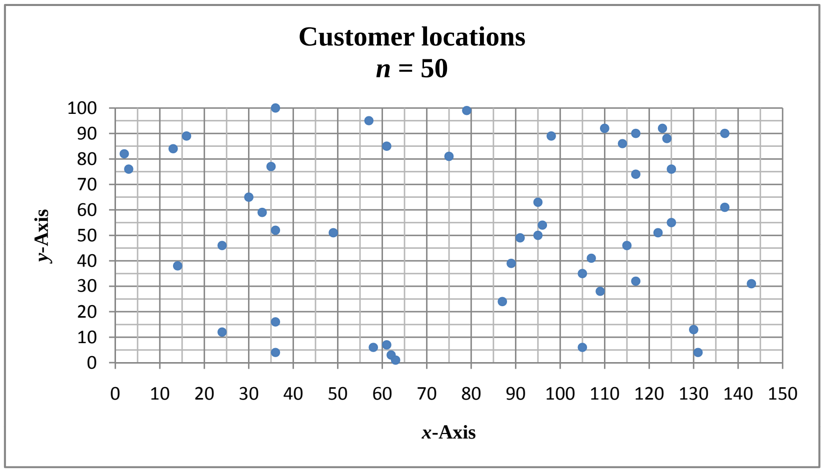
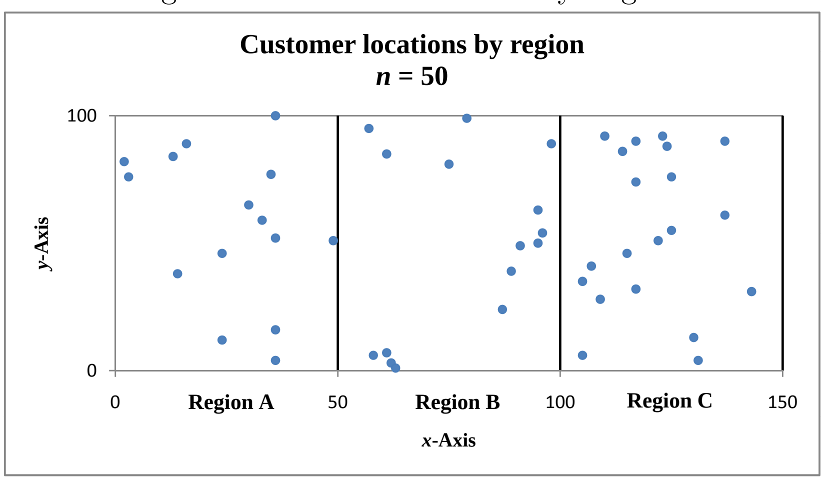
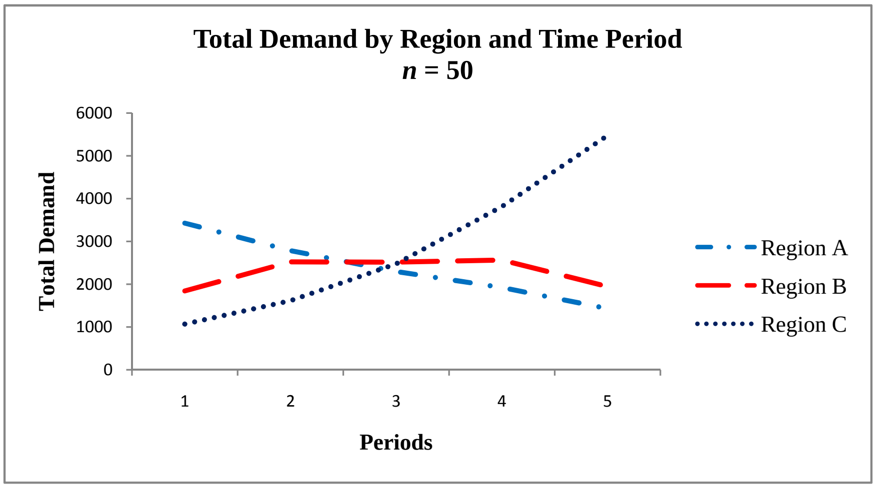


Figure 6 Customer Locations by Region



The demand for each customer in each period is randomly generated from a discrete uniform distribution depending on the region its x-coordinate belongs to. For all demand structures, the total demand of customers in region A is decreasing in value over the entire time horizon. In region B, the total demand increases during the first third of the time horizon, then it remains stable up to the second third, and finally decreases during the last third of the time horizon. Finally in region C, the total demand is increasing along the time horizon. Figure 7 shows the total demand by region for 50 locations and 5 periods.

Figure 7 Total Demand by Region and Time Period

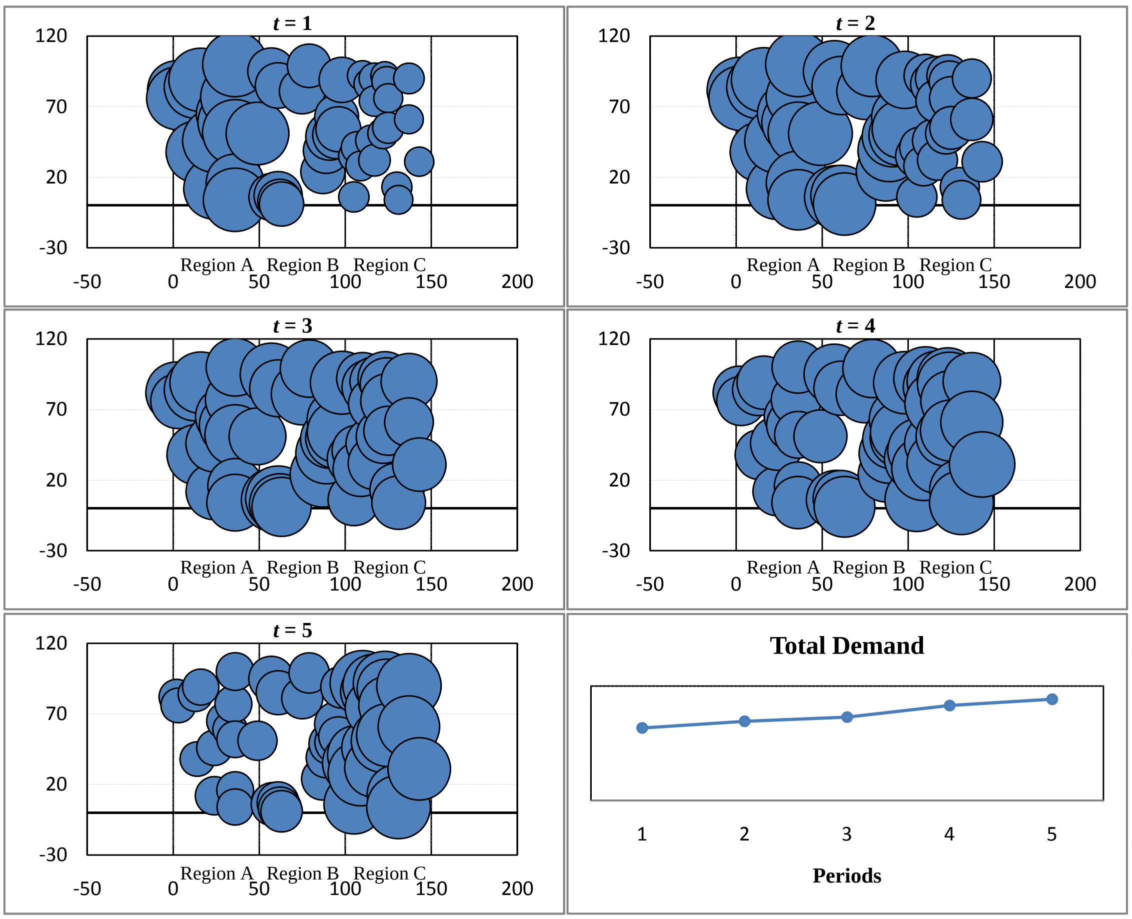


To generate each demand structure, first we define a base value for each region, $d_r \geq 0, r \in \{A, B, C\}$. We define $k_{1t} = 300d_r$ and $k_{2t} = 400d_r$, $t \in T$. The demand in the first period for each customer location is randomly generated from a discrete uniform distribution $U[\lfloor k_{11} \rfloor, \lfloor k_{21} \rfloor]$. For $r = A$ and C, we define non-negative scalars δ_1 and δ_2 to generate a random number, $\hat{s}$, from a uniform distribution $U[\delta_1, \delta_2]$; then for $2 \leq t \leq \tau + 1$, the demand for each customer location is randomly generated from a discrete uniform distribution $U[\lfloor \hat{s} \cdot k_{1t-1} \rfloor, \lfloor \hat{s} \cdot k_{2t-1} \rfloor]$.

For $r = B$, we evenly divide the length of the time horizon in three intervals, say $t \leq \tau/3$, $\tau/3 < t \leq 2\tau/3$, and $2\tau/3 < t \leq \tau$. We define the values of δ_1 and δ_2 to generate the random number $\hat{s}$ in each interval. For $t \leq \tau/3$, the demand for each customer location is randomly generated from a discrete uniform distribution $U[\lfloor \hat{s} \cdot k_{1t-1} \rfloor, \lfloor \hat{s} \cdot k_{2t-1} \rfloor]$; for $\tau/3 < t \leq 2\tau/3$, from a discrete uniform distribution $U[\lfloor k_{1t-1} \rfloor, \lfloor k_{2t-1} \rfloor]$; and for $2\tau/3 < t \leq \tau$, from a discrete uniform distribution $U[\lfloor \hat{s} \cdot k_{1t-1} \rfloor, \lfloor \hat{s} \cdot k_{2t-1} \rfloor]$. In Table 1 we give the values of the parameters used to generate increasing, decreasing, and steady total demand.

Figures 8, 9, and 10, respectively, give an example of the shift in customer's demand for increasing, decreasing, and steady total demand considering 50 locations and 5 periods. The center of each circle corresponds to the customer's location, and its radius to the value of the customer's demand in that period.

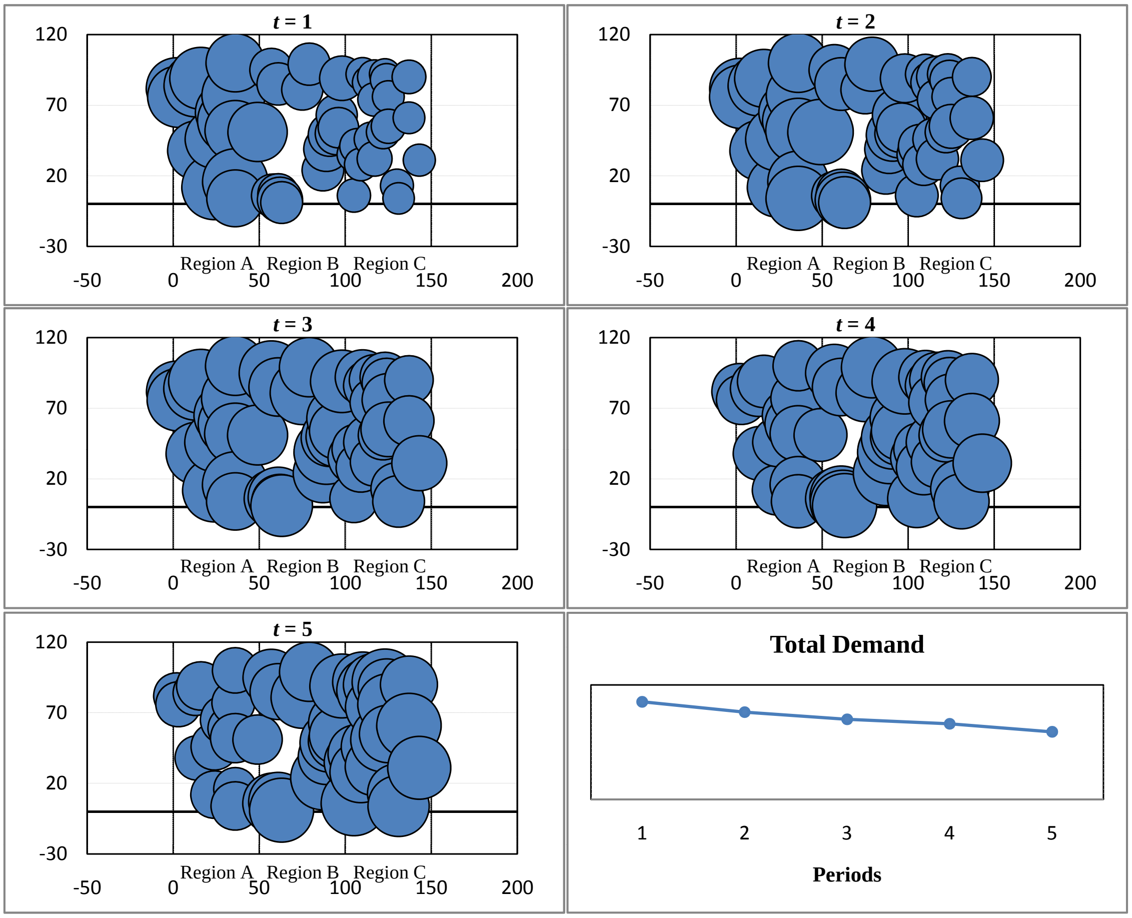
Figure 8 Customer Demands by Time and Region Total Increasing Demand



III.2. Capacity of Facilities

The dynamic and robust location models considered in this dissertation assume that facilities have a finite capacity in the amount of demand that they can supply. The

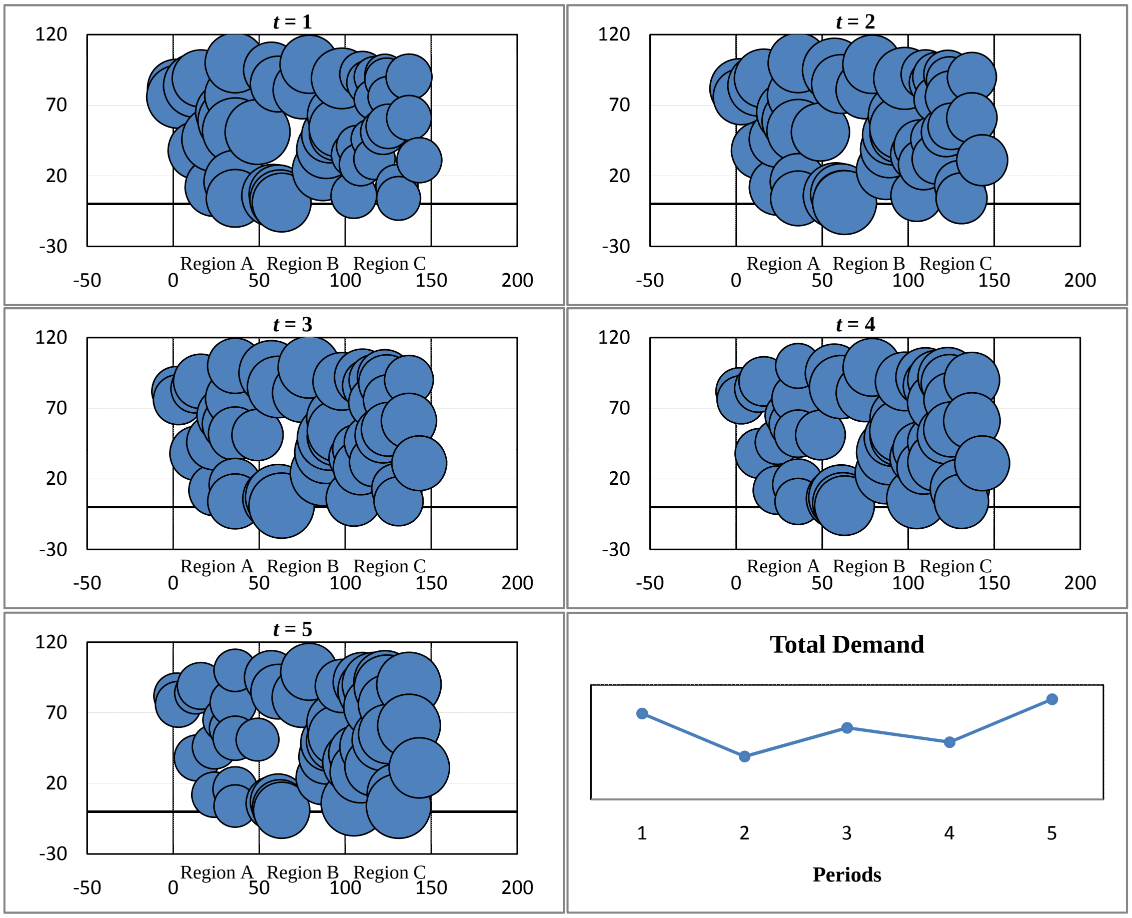
Figure 9 Customer Demands by Time and Region Total Decreasing Demand



capacity at each facility location is assumed to be known. We assume that $m \leq n$, otherwise if $m > n$ the problem becomes trivial since an optimal solution will have a facility established at each customer location, provided that the total capacity available is at least the total demand for each period. Thus, in our models the number of facilities to be established is an unknown and is obtained as a byproduct from the solution to the model.

An instance for which the capacity at each facility location is never greater than or equal to the demand of each customer location, in any time period, is considered infeasible and no solution exists, unless $m > n$. Since the total demand is assumed

Figure 10 Customer Demands by Time and Region Total Steady Demand



to be time varying, possibly following an increasing, decreasing or steady pattern, we would like to consider the largest possible value of the total demand over the time horizon. The time period with the largest total demand will determine the appropriate amount of capacity for the facilities to ensure feasibility.

We randomly generate the capacity for each facility location considering an expected number of open facilities. We also assume that the capacity at each facility location does not change by time.

Let p be a scalar, $0 < p < 1$, denoting the percentage of expected open facilities from the total number of possible locations m. Let $D = \max_{t \in T} D_t$. The base capacity value, Q, for each facility location is determined by the quotient:

$$Q = \left\lfloor \frac{D}{pm} \right\rfloor \tag{3.1}$$

Once we obtain the base capacity value Q, we randomly generate the capacity $q_j, j \in J$, for each facility location from a uniform distribution $U[0.8Q, 1.2Q]$, truncating the value towards zero.

III.3. Cost Parameters

The location models studied in this dissertation consider that cost parameters are known or can be accurately predicted for each period of the time horizon. Making a decision of whether to open a new facility (or close an existing facility) now or in a future period requires the consideration of the time value of money. Usually, the analysis of a series of future costs or investments considers the present value of these costs. We assume that all cost parameters are computed in terms of their present value.

We consider that shipments of demand from facilities to customers incur a transportation cost proportional to quantity and distance. Let $\alpha > 0$ denote the per unit distance per unit demand cost. The distance, d_{ij}, between locations $i \in I$ and $j \in J$ is computed using the Euclidean or straight-line metric:

$$d_{ij} = \left[(x_i - x_j)^2 + (y_i - y_j)^2\right]^{\frac{1}{2}}, \quad i \in I, j \in J \tag{3.2}$$

The transportation cost, c_{ijt}, for shipping demand of customer location i from a facility at location j in period t is computed as follows:

$$c_{ijt} = \alpha w_{it} d_{ij}, \quad i \in I, j \in J, t \in T \tag{3.3}$$

For all the experiments we set $\alpha = 1$, and truncated the value towards zero to obtain integer values.

The fixed operation cost is randomly generated from a discrete uniform distribution $U[\ell, \upsilon]$, $0 < \ell < \upsilon$. Let $\theta = (\ell + \upsilon)/2$. The fixed opening cost is randomly generated from a uniform distribution $U[0.75\theta, 0.85\theta]$, and the fixed closing cost from a uniform distribution $U[0.10\theta, 0.15\theta]$, truncating the value of each fixed cost towards zero to obtain integer values.

The solution algorithms developed to solve our dynamic and robust location models were coded in standard C++ code using ILOG CPLEX 9.0 and ILOG Concert Technology (trademarks of ILOG, Inc.). All the experiments were performed on a Dell OptiPlex 755 desktop computer with 3.16 GHz Dual Core 2 processor and 4.0 GB of memory.

III.4. Summary

In this chapter we described the time varying behavior and structure of the total demand. We consider three total demand structures, increasing, decreasing, and steady. We also consider a shift in the value and location of the demand for each customer. Each customer location is assigned to a region according to its x-coordinate. Each region defines a particular type of shifting in the value of the customer's demand. We also described the methods to generate the capacity of facility locations to ensure feasible solutions for all instances, the computation of the variable transportation cost, and the fixed costs for opening, operating, and closing facilities.

Table 1 Parameters Used to Randomly Generate Total Demand Structures

Total Demand Structure	*Region*	*Time Period*	*Parameters* d_r	δ_1	δ_2	k_{1t}	k_{2t}
$\tau = 5$ Periods							
Increasing	A	$2 \le t \le \tau$	$d_A = 0.80$	0.75	0.85	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
	B	$t \le \tau/3$	$d_B = 0.35$	1.35	1.40	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
	B	$\tau/3 < t \le 2\tau/3$				k_{1t-1}	k_{2t-1}
	B	$2\tau < t \le \tau$		0.75	0.85	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
	C	$2 \le t \le \tau$	$d_C = 0.15$	1.50	1.55	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
Decreasing	A	$2 \le t \le \tau$	$d_A = 0.65$	0.65	0.70	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
	B	$t \le \tau/3$	$d_B = 0.25$	1.05	1.15	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
	B	$\tau/3 < t \le 2\tau/3$				k_{1t-1}	k_{2t-1}
	B	$2\tau < t \le \tau$		0.60	0.65	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
	C	$2 \le t \le \tau$	$d_C = 0.15$	1.10	1.15	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
Steady	A	$2 \le t \le \tau$	$d_A = 0.65$	0.80	0.85	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
	B	$t \le \tau/3$	$d_B = 0.65$	1.06	1.09	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
	B	$\tau/3 < t \le 2\tau/3$				k_{1t-1}	k_{2t-1}
	B	$2\tau < t \le \tau$		1.06	1.09	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
	C	$2 \le t \le \tau$	$d_C = 0.65$	1.10	1.15	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
$\tau = 10$ Periods							
Total Demand Structure	*Region*	*Time Period*	*Parameters* d_r	δ_1	δ_2	k_{1t}	k_{2t}
Increasing	A	$2 \le t \le \tau$	$d_A = 0.95$	0.85	0.90	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
	B	$t \le \tau/3$	$d_B = 0.35$	1.25	1.30	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
	B	$\tau/3 < t \le 2\tau/3$				k_{1t-1}	k_{2t-1}
	B	$2\tau < t \le \tau$		0.80	0.85	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
	C	$2 \le t \le \tau$	$d_C = 0.15$	1.25	1.30	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
Decreasing	A	$2 \le t \le \tau$	$d_A = 0.80$	0.75	0.80	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
	B	$t \le \tau/3$	$d_B = 0.25$	1.05	1.15	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
	B	$\tau/3 < t \le 2\tau/3$				k_{1t-1}	k_{2t-1}
	B	$2\tau < t \le \tau$		0.75	0.80	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
	C	$2 \le t \le \tau$	$d_C = 0.15$	1.05	1.10	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
Steady	A	$2 \le t \le \tau$	$d_A = 0.65$	0.90	0.95	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
	B	$t \le \tau/3$	$d_B = 0.65$	1.00	1.05	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
	B	$\tau/3 < t \le 2\tau/3$				k_{1t-1}	k_{2t-1}
	B	$2\tau < t \le \tau$		0.85	0.90	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$
	C	$2 \le t \le \tau$	$d_C = 0.65$	1.00	1.05	$s \cdot k_{1t-1}$	$s \cdot k_{2t-1}$

CHAPTER IV

DYNAMIC CAPACITATED FIXED CHARGE LOCATION PROBLEM (DCFLP)

In this chapter, we investigate the problem of finding the locations of facilities with limited capacity to satisfy the demand of customers over a discrete and finite time horizon. The total demand of customers is assumed to be changing by time in a known way, and can be split or served by one or more facilities. There are fixed costs associated with establishing or opening new facilities, operating the facilities, and for closing existing facilities. Also, there is a variable transportation cost for serving the customers' demand. The main objective is to find an optimal sequence for locating facilities to satisfy a time varying demand while observing the capacity restrictions over the time horizon.

The chapter is organized as follows. In Section IV.1, we give the problem statement. In Section IV.2, we present the mixed integer programming formulation and notation for the DCFLP. In Section IV.3, we develop a Lagrangian relaxation and Benders' decomposition algorithms to solve the DCFLP. In Section IV.4, we present an empirical analysis using different total demand patterns to test the performance of the solution algorithms. Finally, in Section IV.5, we summarize the results and give concluding remarks.

IV.1. Problem Statement

Specifically, the DCFLP can be stated as follows. Consider a given group of customers on a geographical region. Each customer has a given demand for a certain product. Along a discrete and finite time horizon, the total demand of the customers is changing in a known way. This situation can be related to changes in population, changes in the shopping habits of customers, new trends or fashions that increase the total demand for new products or decrease the total demand for obsolete products. An increase or decrease of the total demand for services and goods, in a particular geographical region, may soon require the construction of facilities or services such as power substations, convenience stores, hospitals, schools, fire stations, etc.

The establishment of facilities is required to supply the customers' demand over the entire time horizon. Each facility location has a known limit or capacity in the amount of demand that can be supplied. Each customer can be served from one or more facilities. The shipments of demand between facilities and customers incur a variable transportation cost proportional to quantity and distance. Further, the establishment of a new facility incurs a fixed opening cost, which can represent the initial investment for construction, equipment, and resources needed to start operations. An additional fixed operating cost is incurred in each period the facility remains operational, this can be thought of as the per period cost associated with the initial investment or the total expenses for services and labor. Finally, if the facility is not needed in any given period it can be closed incurring a fixed closing cost, which represents the expenses for shutting down production or decreasing the labor force. Establishment and closure of facilities are immediate and take place at the beginning of each period.

The main decisions are determining the number of facilities required to supply the demand in each period, selecting the locations to establish the facilities, and allocating demand to facilities in such a way that the total fixed and variable costs are minimum without exceeding the capacity of facilities over the entire time horizon.

IV.2. Model and Notation

In this section, we provide a mixed integer programming formulation of the DCFLP. We use the following notation.

Parameters

I set of customer locations, $i = 1, \ldots, n$

J set of facility locations, $j = 1, \ldots, m$

T set of periods, $t = 1, \ldots, \tau$

f_{jt} fixed cost for having a facility open (operating) in location j during period t

a_{jt} fixed cost for opening a new facility (not existing in the previous period) in location j at the beginning of period t

b_{jt} fixed cost for closing an existing facility (already open in the previous period) in location j at the beginning of period t

w_{it} demand of customer in location i during period t

q_j capacity available if a facility is opened at location j

d_{ij} distance between facility at location j to customer i

α per unit distance per unit demand cost

c_{ijt} transportation cost for shipping demand of customer location i from facility at location j in period t, $c_{ijt} = \alpha w_{it} d_{ij}$

Decision Variables

x_{ijt} fraction of demand of customer location i shipped from facility at location j in period t

y_{jt} 1 if a facility is open in location j at the beginning of period t, 0 otherwise

u_{jt} 1 if a new facility is opened in location j at the beginning of period t, 0 otherwise

v_{jt} 1 if an existing facility is closed in location j at the beginning of period t, 0 otherwise

$$\text{(DCFLP)}\quad \min \sum_{i\in I}\sum_{j\in J}\sum_{t\in T} c_{ijt}x_{ijt} + \sum_{j\in J}\sum_{t\in T}\left(f_{jt}y_{jt} + a_{jt}u_{jt} + b_{jt}v_{jt}\right) \tag{4.1}$$

subject to

$$\sum_{j\in J} x_{ijt} = 1 \qquad i\in I,\ t\in T \tag{4.2}$$

$$x_{ijt} \le y_{jt} \qquad i\in I, j\in J, t\in T \tag{4.3}$$

$$\sum_{i\in I} w_{it}x_{ijt} \le q_j y_{jt} \qquad j\in J,\ t\in T \tag{4.4}$$

$$v_{jt} - u_{jt} + y_{jt} - y_{jt-1} = 0 \qquad j\in J,\ t\in T \tag{4.5}$$

$$x_{ijt} \ge 0, y_{jt}, u_{jt}, v_{jt} \in \{0,1\} \qquad i\in I,\ j\in J,\ t\in T \tag{4.6}$$

The objective function (4.1) includes the total cost over the time horizon; it has two main components. The first component represents the total transportation cost between facilities and customer locations. The second component represents the total fixed cost for operating open facilities, opening new facilities, and closing existing facilities. The constraints (4.2) are the demand constraints (for each customer, all the demand must be met), (4.3) ensure that demand is allocated to open facilities, (4.4) are the capacity constraints (no facility can supply more than its capacity), (4.5) are the logic constraints for relocation, and (4.6) are the nonnegativity and integrality constraints.

Note that decision variables x_{ijt} being continuous allows the demand of each customer location to be split between open facilities, this is called multi-sourcing. If these decision variables are restricted to be binary integers, then the demand of each customer must be served by only one of the open facilities, this is called single-sourcing.

The logic constraints (4.5) state that a facility can be opened (closed) only if it was closed (opened) in the previous period. If there are not existing facilities at the beginning of the time horizon, then we can set $y_{j0} = 0, \forall j \in J$. This set of constraints also helps to incorporate the fixed costs for opening and closing the facilities. Table 2 shows the values of the binary integer decision variables for all possible combinations.

Table 2 Possible Values for Location Decision Variables

y_{jt-1}	y_{jt}	u_{jt}	v_{jt}	*Implies*
0	0	0	0	No facility in location j
1	0	0	1	Existing facility is closed
1	1	0	0	Facility remains open
0	1	1	0	New facility is opened

An optimal solution to the DCFLP model returns the values of the decision variables indicating for each time period the location of open facilities, the amount of demand from each customer allocated to each open facility, and the period and location where facilities are to be opened or closed.

The Capacitated Fixed Charge Location Problem (CFLP) is known to be NP-hard (Cornuejols et al., 1991). The DCFLP, which has the additional dimension of time in the number of decision variables and constraints, is also NP-hard since it contains the CFLP as a special case. For practical size problems, solution methods developed for mixed integer programs may be inefficient in trying to solve the entire model at once (such as branch and bound). In the next section we develop two efficient solution algorithms for the DCFLP that exploit the special structure of the model.

IV.3. Solution Procedure

In this section, we develop a Lagrangian relaxation and Benders' decomposition algorithms to solve the DCFLP.

IV.3.1. Lagrangian Relaxation

The Lagrangian relaxation approach considers the relaxation of a set of constraints by incorporating it into the objective function using a set of Lagrange multipliers. The set of Lagrange multipliers is a penalty imposed to solutions that violate the relaxed set of constraints. The purpose of this type of relaxation is to obtain a Lagrangian problem which is easier to solve than the original problem. The Lagrangian relaxation approach was introduced by Held and Karp (1970, 1971) to solve the traveling salesman problem.

Lagrangian relaxation is considered to be an efficient solution method for the CFLP. Applications of the Lagrangian relaxation for the CFLP consider the relaxation of different sets of constraints. Results and analysis of different implementations to solve the CFLP can be found in Geoffrion (1974), Geoffrion and McBride

(1978), Nauss (1978), Christofides and Beasley (1983), Nemhauser and Wolsey (1988), Beasley (1988), Beasley (1993), and Baker and Sheasby (1999) for the multi-source case; Barcelo and Casanovas (1984), Klincewicz and Luss (1986), Sridharan (1993), Holmberg et al. (1999), and Hindi and Pieńkosz (1999) for the single-source case.

Consider relaxing constraints (4.4) and incorporating them into the objective function with associated non-negative Lagrange multipliers λ_{jt}. We obtain the following Lagrangian subproblem after rearranging terms:

$$LR(\lambda) = \min_{x,y} \sum_{i\in I}\sum_{j\in J}\sum_{t\in T} (c_{ijt} + w_{it}\lambda_{jt})\, x_{ijt} + \sum_{j\in J}\sum_{t\in T} (f_{jt} - q_j\lambda_{jt})\, y_{jt} + \sum_{j\in J}\sum_{t\in T} (a_{jt}u_{jt} + b_{jt}v_{jt}) \tag{4.7}$$

subject to

$$\sum_{j\in J} x_{ijt} = 1 \qquad i \in I,\ t \in T \tag{4.8}$$

$$x_{ijt} \le y_{jt} \qquad i \in I,\ j \in J,\ t \in T \tag{4.9}$$

$$\sum_{i\in I} w_{it} \le \sum_{j\in J} q_j y_{jt} \qquad t \in T \tag{4.10}$$

$$v_{jt} - u_{jt} + y_{jt} - y_{jt-1} = 0 \qquad j \in J,\ t \in T \tag{4.11}$$

$$x_{ijt} \ge 0, y_{jt}, u_{jt}, v_{jt} \in \{0,1\} \qquad i \in I,\ j \in J,\ t \in T \tag{4.12}$$

Here we have added the surrogate constraints (4.10) to the relaxed problem. This set of surrogate constraints follows from constraint set (4.2) and (4.3), summing over $j \in J$. These constraints are useful in obtaining feasible solutions since we get a set of open facilities with enough capacity to satisfy the demand in each period. This particular type of relaxation has been proven to give a stronger bound over other possible relaxations for the CFLP (Cornuejols et al., 1977).

Note that we have added a non-positive term into the objective function, thus $LR(\lambda) = \underline{Z}$ is a lower bound for the objective function value, Z, of the DCFLP. For given values of the decision variables, $\hat{y}_{jt}$, $\hat{u}_{jt}$, and $\hat{v}_{jt}$ an upper bound can be obtained by solving the following transportation problem:

$$TP(x|\hat{y}) = \min \sum_{i\in I}\sum_{j\in J}\sum_{t\in T} c_{ijt}x_{ijt} + \sum_{j\in J}\sum_{t\in T}(f_{jt}\hat{y}_{jt} + a_{jt}\hat{u}_{jt} + b_{jt}\hat{v}_{jt}) \tag{4.13}$$

subject to

$$\sum_{j\in J} x_{ijt} = 1 \qquad i \in I,\, t \in T \tag{4.14}$$

$$\sum_{i\in I} w_{it}x_{ijt} \leq q_j\hat{y}_{jt} \qquad j \in J,\, t \in T \tag{4.15}$$

$$x_{ijt} \geq 0 \qquad i \in I,\, j \in J,\, t \in T \tag{4.16}$$

Problem $TP(x|\hat{y})$ separates into $|T|$ independent transportation problems (one for each period $t \in T$). The objective function value, $TP(x|\hat{y}) = \bar{Z}$, is an upper bound for the objective function value, Z^*, of any optimal solution to the DCFLP, i.e., $LR(\lambda) = \underline{Z} \leq Z^* \leq \bar{Z} = TP(x|\hat{y})$.

Thus, an optimal solution can be obtained by closing the optimality gap between the lower and upper bound. Since the inequality $LR(\lambda) \leq Z$ holds for all λ, we need to find a vector λ of Lagrange multipliers that gives the largest lower bound. In other words, we need to solve the Lagrangian dual:

$$Z_{LD}(\lambda) = \max_{\lambda} LR(\lambda) \tag{4.17}$$

The function $Z_{LD}(\lambda)$ is a piece-wise linear concave function of λ, non-differentiable at the maximum point (Fisher, 2004). The maximum point can be obtained using the subgradient optimization procedure (Held et al., 1974).

At each iteration of the subgradient procedure, a set of Lagrange multipliers is given as input to the Lagrangian subproblem $LR(\lambda)$. A new feasible solution and upper bound are obtained solving $TP(x|\hat{y})$. The values of the Lagrange multipliers are updated and the process is repeated.

Since the convergence of the subgradient procedure is not guaranteed, we need to keep track of the values of the best lower and upper bound. We can stop the subgradient method when the difference between the best upper and lower bound are within a predefined threshold or after a given number of iterations.

At each iteration k in the subgradient method, the value of the step size, π^k, is updated as follows:

$$\pi^k = \frac{\delta^k \left(Z_{UB} - Z_{lb}^k\right)}{\sum_{j\in J}\sum_{t\in T}\left(\sum_{i\in I} w_{it}x_{ijt}^k - q_j y_{jt}^k\right)^2} \tag{4.18}$$

where Z_{UB} is the best upper bound, Z_{lb}^k the optimal objective function value of Lagrangian subproblem for given Lagrange multipliers λ_{jt}^k, δ^k a scalar, $0 < \delta^k \leq 2$, x_{ijt}^k and y_{jt}^k the optimal decision variables for Lagrangian subproblem.

For $Z_{LD}(\lambda)$ to give a lower bound, it is necessary that at each iteration k of the subgradient optimization procedure we adjust the value of the non-negative Lagrange multipliers λ_{jt} as follows:

$$\lambda_{jt}^{k+1} = \max\left\{0, \lambda_{jt}^k + \pi^k\left(\sum_{i\in I} w_{it}x_{ijt}^k - q_j y_{jt}^k\right)\right\} \quad j \in J,\, t \in T \tag{4.19}$$

Observe that violations to constraints (4.4) are penalized by increasing the value of the associated Lagrange multipliers; thus the Lagrangian subproblem will try to find solutions with the least level of violation to the capacity constraints.

Display 1 presents the pseudo-code of the subgradient optimization algorithm to solve the DCFLP. We define the following notation used in the pseudo-code of the solution algorithms:

ε	non-negative threshold, $0 \leq \varepsilon < 1$
ϵ	positive scalar, $0 < \epsilon < 1$
β	positive scalar, $0 < \beta < 1$
∞	a very large number
S	best feasible solution
Z_{lb}^k	trial lower bound
Z_{LB}	best lower bound
Z_{ub}^k	trial upper bound
Z_{UB}	best upper bound
M	maximum number of iterations
N	predefined number of iterations without improvement in the lower bound value

Display 1 Pseudo-code subgradient optimization algorithm

Initialize $k \leftarrow 0, \ell \leftarrow 0, Z_{LB} \leftarrow -\infty, Z_{UB} \leftarrow \infty, \delta^k, \lambda_{jt}^k$
while $k \leq$ M **do**
  Solve $LR(\lambda)$
  $Z_{lb}^k \leftarrow LR(\lambda)$
  if $Z_{lb}^k > Z_{LB}$ **then**
    $Z_{LB} \leftarrow Z_{lb}^k$
    $\ell \leftarrow 0$
  else
    $\ell \leftarrow \ell + 1$
    if $\ell = N$ **then**
      $\delta^k \leftarrow \beta\delta^k$
    end if
  end if
  Solve $TP(x|\hat{y})$
  $Z_{ub}^k \leftarrow TP(x|\hat{y})$
  if $Z_{ub}^k < Z_{UB}$ **then**
    $Z_{UB} \leftarrow Z_{ub}^k$
    Record S
  end if
  if $(Z_{UB} - Z_{LB})/Z_{UB} \leq \varepsilon$ or $\delta^k \leq \epsilon$ **then**
    Stop
  else
    Update $\pi^k, \lambda_{jt}^{k+1}$
  end if
  $k \leftarrow k + 1$
end while
Return S, Z_{UB}

The optimality gap between the best lower and upper bound, $(Z_{UB} - Z_{LB})/Z_{UB}$, is computed in the same way CPLEX computes the optimality gap. The initial values of the Lagrange multipliers, λ_{jt}^0, can be set to zero or to a predefined value. The positive scalar β is used to decrease the value of δ^k, when the best lower bound fails to improve after N consecutive iterations of the subgradient optimization procedure. When the value of the best upper bound is updated, we record the current feasible solution S (set of open facilities and allocation of customers to facilities), which at termination is returned as the best feasible solution together with the best upper bound value.

The main advantage of using Lagrangian relaxation to solve the DCFLP is in the problem structure. By relaxing the capacity constraints (4.4) we obtain a Lagrangian subproblem that is at most as difficult to solve as a dynamic Uncapacitated Facility Location Problem (UFLP), which does not have the integrality property. Furthermore, to obtain a feasible solution and upper bound, we solve a multi-period transportation problem (linear program), which can be efficiently solved. In Section IV.4, we conduct an empirical analysis to test the performance of the Lagrangian relaxation algorithm in solving the DCFLP.

IV.3.2. Benders' Decomposition

The DCFLP model has a special structure. For fixed values of the location variables the resulting problem is a multi-period transportation problem (a linear program). Thus, we can *decompose* the problem into two subproblems, one which with the binary integer variables gives a solution with a set of open facilities, and a problem with continuous variables that allocates the demand of customers to facilities. This special structure can be exploited using Benders' decomposition (Benders, 1962).

The main idea behind Benders' decomposition is to separate or decompose a linear mixed integer program into two smaller problems (or subproblems) by separating the continuous variables from the integer variables. One subproblem is constructed with only continuous variables, called the Benders' subproblem. The integer variables, or complicating variables, are part of the second subproblem called the Benders' master problem which has only one additional continuous variable.

Applications of Benders' decomposition to solve the CFLP can be found in Geoffrion and Graves (1974) for the multi-commodity distribution system design (a generalization of the CFLP), McDaniel and Devine (1977) for the linear programming relaxation approach to solve the master problem during early iterations of the decomposition algorithm, Van Roy (1986) for the cross decomposition approach that combines Lagrangian relaxation and Benders' decomposition, and Wentges (1996) in the development of efficient algorithms to accelerate the convergence of Benders' decomposition.

Depending on the structure of the problem, an optimal solution to the original problem is obtained by successively and iteratively solving each subproblem. The solution to the master problem is given as an input to the subproblem, which returns a dual constraint or cut to the master problem restricting its feasible region. The process is repeated until the optimality gap between the subproblems is closed.

The special primal structure of the DCFLP makes it a good candidate for Benders' decomposition. For given values of the location variables, $\hat{y}_{jt}$, we obtain the following Benders' subproblem (a transportation problem):

$$(SP_y) \quad \min \sum_{i \in I} \sum_{j \in J} \sum_{t \in T} c_{ijt} x_{ijt} \tag{4.20}$$

subject to

$$\sum_{j \in J} x_{ijt} = 1 \qquad i \in I,\ t \in T \tag{4.21}$$

$$x_{ijt} \leq \hat{y}_{jt} \qquad i \in I,\ j \in J,\ t \in T \tag{4.22}$$

$$\sum_{i \in I} w_{it} x_{ijt} \leq q_j \hat{y}_{jt} \qquad j \in J,\ t \in T \tag{4.23}$$

$$x_{ijt} \geq 0 \qquad i \in I,\ j \in J,\ t \in T \tag{4.24}$$

The associated dual of the subproblem:

$$(DSP_y) \quad \max \sum_{i\in I}\sum_{t\in T}\lambda_{it} - \sum_{i\in I}\sum_{j\in J}\sum_{t\in T}\hat{y}_{jt}\mu_{ijt} - \sum_{j\in J}\sum_{t\in T}q_j\hat{y}_{jt}\gamma_{jt} \tag{4.25}$$

subject to

$$\lambda_{it} - \mu_{ijt} - w_{it}\gamma_{jt} \leq c_{ijt} \qquad i \in I,\ j \in J,\ t \in T \tag{4.26}$$

$$\lambda_{it} \text{ unrestricted}, \mu_{ijt} \geq 0, \gamma_{jt} \geq 0 \qquad i \in I,\ j \in J,\ t \in T \tag{4.27}$$

which can be further decomposed into $|T|$ independent dual subproblems (one for each period $t \in T$). Since the feasible region of the primal subproblem is non-empty and bounded, we do not need to consider the extreme rays of the feasible region of the dual subproblem.

Let $\left\{(\lambda^k, \mu^k, \gamma^k) : k \in P\right\}$ denote all the extreme points of (DSP_y). Let ρ^k denote the objective function value corresponding to the kth extreme point, that is:

$$\rho^k = \sum_{i\in I}\sum_{t\in T}\lambda^k_{it} - \sum_{i\in I}\sum_{j\in J}\sum_{t\in T}\hat{y}_{jt}\mu^k_{ijt} - \sum_{j\in J}\sum_{t\in T}q_j\hat{y}_{jt}\gamma^k_{jt}, \quad k \in K \subseteq P \tag{4.28}$$

where K is an appropriate index set. Since at least one optimal solution for a linear program occurs at an extreme point of its feasible region, the optimal solution to the dual subproblem, ρ^*, is at least as large as any objective function value ρ^k for all extreme points $k \in P$ (since this is a maximization problem). Thus, we have the following Benders' master problem:

$$(MP_K) \quad \min \quad \rho + \sum_{j \in J} \sum_{t \in T} (f_{jt} y_{jt} + a_{jt} u_{jt} + b_{jt} v_{jt}) \tag{4.29}$$

subject to

$$\rho \geq \sum_{i \in I} \sum_{t \in T} \lambda_{it}^k - \sum_{i \in I} \sum_{j \in J} \sum_{t \in T} \mu_{ijt}^k y_{jt} - \sum_{j \in J} \sum_{t \in T} q_j \gamma_{jt}^k y_{jt} \qquad k \in K \subseteq P \tag{4.30}$$

$$\sum_{i \in I} w_{it} \leq \sum_{j \in J} q_j y_{jt} \qquad t \in T \tag{4.31}$$

$$v_{jt} - u_{jt} + y_{jt} - y_{jt-1} = 0 \qquad j \in J,\, t \in T \tag{4.32}$$

$$\rho \geq 0, y_{jt}, u_{jt}, v_{jt} \in \{0, 1\} \qquad j \in J,\, t \in T \tag{4.33}$$

where ρ represents the objective function value of the dual subproblem. We can restrict $\rho \geq 0$ as long as $c_{ijt} \geq 0$. We have added the surrogate constraints (4.31) to the master problem to obtain a feasible solution for (SP_y).

Note that the number of extreme points of the dual subproblem may be very large. We do not need to enumerate all the constraints (4.30) explicitly since at an optimal solution of the master problem only a subset of constraints (4.30) is expected to be binding. If we consider only a subset of constraints (4.30), then we will obtain a lower bound on the optimal objective function value of the DCFLP.

An upper bound can be obtained for fixed values of the location variables, solving the associated transportation problems and then adding the corresponding fixed cost for operation and relocation of facilities. Each time we solve the primal and dual subproblems we obtain another constraint of the form (4.30), thus tightening the lower bound obtained from the master problem.

IV.3.2.1. Generation of Strong Cuts

The special structure of the DCFLP provides some level of computational simplification for implementing a decomposition algorithm. However, it is well known that the subproblem (transportation problem) has a high level of degeneracy, thus the dual subproblem can have alternative optimal solutions. Since the improvement in the value of the lower bound (obtained from the solution to the relaxed master problem) is tightened by the Benders' cuts obtained from the dual subproblem, it is important that at each iteration of the decomposition procedure we select the best possible cut.

To strengthen the Benders' cuts obtained from the dual subproblem we implement the algorithm proposed by Van Roy (1986). The values of the dual variables μ_{ijt} and γ_{jt} can be improved without affecting the objective function value of the dual subproblem for the closed facilities. Let $C_t = \{j \in J : y_{jt} = 0\}, t \in T$ denote the set of closed facilities in period t, and $O_t = \{j \in J : y_{jt} = 1\}, t \in T$ the set of open facilities in period t. Also let $j_{(i)}t$ denote the allocation of customer location i to candidate location j in period t, obtained from an optimal solution to (SP_y). Let $(\hat{\lambda}_{it}, \hat{\mu}_{ijt}, \hat{\gamma}_{jt})$ denote the value of the optimal dual variables obtained from the dual subproblem (DSP_y). We set $y_{jt} = 1, j \in C_t, t \in T$, then solve the following linear program:

$$(SC_y) \quad \max -\sum_{i \in I} \sum_{j \in J} \sum_{t \in T} \bar{\mu}_{ijt} y_{jt} - \sum_{j \in J} \sum_{t \in T} q_j \bar{\gamma}_{jt} y_{jt} \tag{4.34}$$

subject to

$$\hat{\lambda}_{it} - \bar{\mu}_{ijt} - w_{it} \bar{\gamma}_{jt} \leq c_{ijt} \qquad i \in I,\ j \in J,\ t \in T \tag{4.35}$$

$$\bar{\mu}_{ijt} \geq 0, \bar{\gamma}_{jt} \geq 0 \qquad i \in I,\ j \in J,\ t \in T \tag{4.36}$$

From the optimal solution to (SC_y), we set $\hat{\mu}_{ijt} = \bar{\mu}_{ijt}, \hat{\gamma}_{jt} = \bar{\gamma}_{jt}, j \in C_t, t \in T$, and leave the previous values of the dual variables, $\hat{\mu}_{ijt}, \hat{\gamma}_{jt}, j \in O_t, t \in T$, unchanged. Note that, constraints (4.35) guarantee that $(\hat{\lambda}_{jt}, \bar{\mu}_{ijt}, \bar{\gamma}_{jt})$ is a feasible solution to the dual subproblem.

IV.3.2.2. Generation of Pareto-Optimal Cuts

The generation of strong cuts produces significant savings in computation time for the decomposition procedure. However, we can further improve the Benders' cuts by considering the closed and open facilities. This procedure relies on the concept of pareto-optimal cuts introduced by Magnanti and Wong (1981). The main idea is to generate a cut that dominates any other cut, that is, a constraint which is tighter than any other. This is called a pareto-optimal cut.

Wentges (1996) developed an algorithm to generate pareto-optimal cuts for the CFLP by considering the open and closed facilities. Observe that, from the relationship between the primal and dual subproblems, the value of the dual variables $\hat{\lambda}_{it}$ represent the cost for serving demand of customer i in period t, and the value of the dual variables $\hat{\mu}_{ijt}$ the cost for allocating costumer i to facility j in period t. The fair cost that this customer should pay for being served by facility $j_{(i)}$, which is closer and more convenient, can be thought of as the additional cost that this customer would have to pay for being served by the second nearest facility. Thus, for the open facilities we can increase the value of $\hat{\lambda}_{it}$ and $\hat{\gamma}_{jt}$, and decrease the cost (or give a reward) of $\hat{\mu}_{ij_{(i)}t}$. In doing so, the objective function value of the dual subproblem remains unchanged and constraints (4.35) are satisfied. However, the improvement on the value of the dual variables $\hat{\lambda}_{it}$ could be too high since the closed facilities are not considered. It is possible that customer i could be better served by one of the closed facilities. Thus, in addition to the open facilities we can improve the value of the dual variables by considering the closed facilities.

The algorithm to develop pareto-optimal cuts improves the values of the dual variables by considering both the open and closed facilities. The additional service cost for the open facilities is determined between the first and second smallest elements in the set $\{c_{ijt}+\hat{\gamma}_{jt} : j \in O_t\}$. Note that if x_{ijt} happens to be in the basis of the primal subproblem, then $\hat{\lambda}_{it} = c_{ij_{(i)}t} + \hat{\gamma}_{j_{(i)}t}$, for some $j_{(i)} \in O_t$ (complementary slackness). In selecting the additional service cost for the closed facilities we selected the third smallest value in the set $\{c_{ijt} + \bar{\gamma}_{jt} : j \in C_t\}$, as it gave the best improvement in the efficiency of the Benders' decomposition algorithm to solve the DCFLP.

This algorithm showed to be crucial in improving the performance of the Benders' decomposition algorithm since the number of open and closed facilities, for each period, gives a considerable opportunity to strengthen the Benders' cuts. The pseudo-code of the algorithm is given in Display 2. The pseudo-code of the Benders' decomposition algorithm is given in Display 3.

Display 2 Pseudo-code pareto-optimal cuts for open and closed facilities

Solve SP_y, DSP_y, and SC_y
for $i = 1$ to n **do**
  for $t = 1$ to τ **do**
    Determine smallest ψ_{it}, second smallest ϕ_{it} from: $\{c_{ijt} + \hat{\gamma}_{jt} : j \in O_t\}$
    Determine third smallest ℓ_{it} from: $\{c_{ijt} + \hat{\gamma}_{jt} : j \in C_t\}$
    Calculate $\theta_{it} \leftarrow \max\{0, \min\{\phi_{it} - \psi_{it}, \ell_{it} - \psi_{it}\}\}$
    Set $\mu^*_{ijt} \leftarrow 0 \; j \in O_t, j \neq j_{(i)}t$
    if $\theta_{it} > 0$ **then**
      Set $\bar{\lambda}_{it} \leftarrow \hat{\lambda}_{it} + \theta_{it}$, $\mu^*_{ij_{(i)}t} \leftarrow \theta_{it}$
    else
      Set $\bar{\lambda}_{it} \leftarrow \hat{\lambda}_{it}$, $\mu^*_{ij_{(i)}t} \leftarrow 0$
    end if
    Solve SC_y again to calculate $\mu^*_{ijt}, \gamma^*_{jt}, j \in C_t$
  end for
end for
Return $(\bar{\lambda}_{it}, \mu^*_{ijt}, \gamma^*_{jt})$

Display 3 Pseudo-code Benders' decomposition algorithm

Initialize $k \leftarrow 0, Z_{LB} \leftarrow -\infty, Z_{UB} \leftarrow \infty$
Solve MP_K
$Z_{lb}^k \leftarrow MP_K$
if $Z_{lb}^k > Z_{LB}$ **then**
    $Z_{LB} \leftarrow Z_{lb}^k$
end if
while $k \leq$ M **do**
    Solve DSP_y
    $Z_{ub}^k \leftarrow DSP_y +$ fixed costs
    if $Z_{ub}^k < Z_{UB}$ **then**
        $Z_{UB} \leftarrow Z_{ub}^k$
        Record S
    end if
    if $(Z_{UB} - Z_{LB})/Z_{UB} \leq \varepsilon$ **then**
        Stop
    else
        Obtain pareto-optimal cut $(\bar{\lambda}_{it}, \mu_{ijt}^*, \gamma_{jt}^*)$
        Solve MP_K with $(\bar{\lambda}_{it}, \mu_{ijt}^*, \gamma_{jt}^*)$
        $Z_{lb}^k \leftarrow MP_K$
        if $Z_{lb}^k > Z_{LB}$ **then**
            $Z_{LB} \leftarrow Z_{lb}^k$
        end if
        if $(Z_{UB} - Z_{LB})/Z_{UB} \leq \varepsilon$ **then**
            Stop
        end if
    end if
    $k \leftarrow k + 1$
end while
Return S, Z_{UB}

During the decomposition procedure, it is possible that after several iterations the addition of Benders' cuts to the master problem may increase its size and the computational effort to solve it. Geoffrion and Graves (1974) introduced a variant to the Benders' decomposition approach known as *feasibility seeking* or *ε-optimal*. The idea behind this variant is that initially, instead of solving the master problem to optimality, we stop whenever we find the first feasible solution within some tolerance value of the best upper bound, $Z_{UB} - \varepsilon$. Since the solution to the master problem no longer provides a valid lower bound for the original problem, the decomposition procedure stops whenever the master problem is unable to find a feasible solution with a value lower than $Z_{UB} - \varepsilon$.

To obtain an ε-optimal solution for the DCFLP, the following constraint is added to the Benders' master problem:

$$\rho + \sum_{j \in J} \sum_{t \in T} (f_{jt} y_{jt} + a_{jt} u_{jt} + b_{jt} v_{jt}) \quad \leq \quad Z_{UB} - \varepsilon \tag{4.37}$$

In Section IV.4, we conduct an empirical analysis to test the performance of Benders' decomposition and ε-optimal algorithms.

IV.4. Numerical Results

In this section, we conduct an empirical analysis to test the performance of the solution algorithms developed for the DCFLP. The empirical analysis was designed considering the three total demand structures, increasing, decreasing, and steady (described in Chapter III); two values for n = 50 and 100 locations; two values for τ = 5 and 10 periods; three values for the percentage of expected number of open facilities p = 0.05, 0.10, and 0.15; and three discrete uniform distributions to randomly generate the fixed operation cost, $U[100000, 150000]$, $U[200000, 250000]$, and $U[300000, 350000]$. Thus, the total number of different classes of problems is 108. For each class, we randomly generated 10 instances.

For comparison purposes, we arranged the classes for each total demand structure into 12 clusters, each cluster containing three classes. Table 3 shows the 12 clusters of classes. For all the experiments, we assumed that $y_{j0} = 0, j \in J$, i.e., no existing facilities at the beginning of the first period.

Table 3 DCFLP Classes of Problems Arranged in Clusters

	Parameters					*Parameters*			
Cluster	n	τ	p %	f	Cluster	n	τ	p %	f
1	50	5	5	$U[100000, 150000]$	7	100	5	5	$U[100000, 150000]$
	50	5	10	$U[100000, 150000]$		100	5	10	$U[100000, 150000]$
	50	5	15	$U[100000, 150000]$		100	5	15	$U[100000, 150000]$
2	50	5	5	$U[200000, 250000]$	8	100	5	5	$U[200000, 250000]$
	50	5	10	$U[200000, 250000]$		100	5	10	$U[200000, 250000]$
	50	5	15	$U[200000, 250000]$		100	5	15	$U[200000, 250000]$
3	50	5	5	$U[300000, 350000]$	9	100	5	5	$U[300000, 350000]$
	50	5	10	$U[300000, 350000]$		100	5	10	$U[300000, 350000]$
	50	5	15	$U[300000, 350000]$		100	5	15	$U[300000, 350000]$
4	50	10	5	$U[100000, 150000]$	10	100	10	5	$U[100000, 150000]$
	50	10	10	$U[100000, 150000]$		100	10	10	$U[100000, 150000]$
	50	10	15	$U[100000, 150000]$		100	10	15	$U[100000, 150000]$
5	50	10	5	$U[200000, 250000]$	11	100	10	5	$U[200000, 250000]$
	50	10	10	$U[200000, 250000]$		100	10	10	$U[200000, 250000]$
	50	10	15	$U[200000, 250000]$		100	10	15	$U[200000, 250000]$
6	50	10	5	$U[300000, 350000]$	12	100	10	5	$U[300000, 350000]$
	50	10	10	$U[300000, 350000]$		100	10	10	$U[300000, 350000]$
	50	10	15	$U[300000, 350000]$		100	10	15	$U[300000, 350000]$

The performance of the Lagrangian relaxation algorithm relies on the value of the parameters used by the subgradient optimization procedure. To determine the most appropriate values for these parameters, we solved two instances per class for each total demand structure. For the initial value of the Lagrange multipliers, we considered the values of 0 and 1.0; for the initial value of δ^0, 2.0, 1.8, and 1.5; for β, 0.50, 0.60, and 0.80; and for N, 5, 10, and 20 iterations. We selected the combination of parameters with the lowest average optimality gap and lowest average solution time in seconds over all the classes. We set the initial values of the Lagrange multipliers $\lambda_{jt}^0 = 0, j \in J, t \in T$, $\delta^0 = 1.8$, $M = 200$, $N = 5$, and $\beta = 0.5$, since this combination gave the best over all performance for each total demand structure.

At each iteration of the subgradient algorithm, we solved the Lagrangian subproblem using CPLEX with early stopping at 1.0% optimality gap. We took the lower bound value from the solution given by CPLEX at early stopping and set it as the trial lower bound. Note that this approach gives a valid lower bound. The set of open facilities obtained from the Lagrangian subproblem was given as an input to CPLEX to solve $|T|$ transportation problems to optimality and obtain the value of the trial upper bound. The stopping criteria for the subgradient algorithm was set to the first occurrence of three conditions: 1.5% optimality gap, $\delta^k \leq 0.001$, and $M = 200$ iterations.

For Benders' decomposition and ε-optimal algorithms, we solved the master problem initially with a large optimality gap and gradually reduced it as the algorithm progressed. We followed this approach since in the initial iterations the master problem does not have enough information from the dual subproblem until several Benders' cuts are added. During the first 10 iterations of both decomposition algorithms, we solved the master problem using CPLEX with early stopping considering an optimality gap of 5%. This optimality gap was reduced every 10 iterations to 3.5, 2.5, 1.5, and 1.0%. To determine this sequence of values for early stopping, we solved two instances for each class and for each total demand structure. We defined three stages for the optimality gap of the master problem: initial, intermediate, and final. For the initial stage, we considered 5 and 10% optimality gap; for the intermediate stage, we considered three percentage values: 3.5, 2.5, and 1.5% for initial stage gap of 5%, and 8.5, 4.5, and 1.5% for initial stage gap of 10%. For the final stage, we considered a percentage gap of 1.0%. For the number of iterations, we considered 5, 10, and 15 iterations. We selected the combination that gave the lower average optimality gap and lower average solution time over all the classes.

At each iteration of the decomposition algorithm, we solved the master problem using CPLEX, with early stopping as described above. We took the lower bound value from the solution given by CPLEX and set it as the trial lower bound. To compute the upper bound, the set of open facilities obtained from the master problem was given as an input to CPLEX to solve $|T|$ dual problems to optimality. The value of the trial upper bound at iteration k was computed by $Z_{ub}^k = DSP_y$ plus the associated fixed opening, operation, and closing costs. The primal subproblem and the strong-cuts algorithm were also solved using CPLEX. The stopping criteria for the Benders' decomposition algorithms was set to 1.5% optimality gap and $M = 200$ iterations.

The benchmark solutions were obtained solving the DCFLP model with CPLEX, which uses a branch and cut algorithm, using default settings. We used early stopping with an optimality gap of 1.5%, and recorded the lower and upper bound values.

For all the experiments, we limited the running time for each instance with $n = 50$ to 3000 seconds, and $n = 100$ to 4000 seconds. For each class, we reported the average (Avg.) and maximum (Max.) value of the optimality gap and the average and maximum solution time. Tables 4, 5, and 6 on pages 68, 69, and 70 respectively, describe the performance of the solution algorithms for each total demand structure. We denote by NS the benchmark classes Not Solved by CPLEX within the maximum solution time (in most cases, the solution to the root node exceeded the maximum running time).

As part of the empirical analysis, we considered the analysis of the cost split, in percentage value, of the total cost corresponding to variable transportation cost and fixed opening, operation, and closing costs. For each class, we selected the solution with minimum average optimality gap to compute the average percentage for each type of cost. In Table 7 on page 71, we report the average cost split per class for each total demand structure. The analysis of the cost split is important, we can identify a relation between the efficiency of the solution algorithms and the structure of the cost split for each class of problems.

Table 4 DCFLP Computational Results Increasing Demand

Cluster	Branch & Cut Gap (%) Avg.	Branch & Cut Gap (%) Max.	Branch & Cut Time (sec.) Avg.	Branch & Cut Time (sec.) Max.	Lagrangian Relaxation Gap (%) Avg.	Lagrangian Relaxation Gap (%) Max.	Lagrangian Relaxation Time (sec.) Avg.	Lagrangian Relaxation Time (sec.) Max.	Benders' Decomposition Gap (%) Avg.	Benders' Decomposition Gap (%) Max.	Benders' Decomposition Time (sec.) Avg.	Benders' Decomposition Time (sec.) Max.	ε-Optimal Time (sec.) Avg.	ε-Optimal Time (sec.) Max.
	2.58	7.70	930.06	3000.18	1.13	1.48	10.05	33.09	1.48	1.50	45.74	68.59	9.16	22.91
1	3.91	5.90	2671.92	3000.34	1.32	1.60	172.62	586.10	1.45	1.50	56.04	200.00	11.80	31.00
	3.23	4.76	2562.44	3000.27	1.35	1.53	625.47	2215.69	1.31	1.50	27.99	57.38	5.93	13.22
	4.22	13.70	2432.35	3000.31	1.20	1.49	5.62	26.25	1.41	1.49	20.61	37.42	3.52	7.27
2	5.78	8.87	2933.35	3000.27	1.15	1.49	10.37	57.07	1.41	1.50	14.99	22.64	3.26	6.17
	4.06	6.67	2929.43	3000.63	1.35	1.50	131.48	549.10	1.33	1.50	7.98	17.17	1.88	5.75
	6.45	15.47	2769.05	3000.23	0.99	1.48	15.95	119.83	1.39	1.49	17.27	27.05	2.43	4.14
3	6.50	9.72	3000.14	3000.29	1.18	1.49	7.06	46.36	1.30	1.49	10.02	25.08	1.98	4.52
	4.35	6.37	3000.17	3000.33	1.16	1.43	47.82	446.57	1.19	1.49	3.81	12.00	1.24	3.33
	1.86	4.26	1060.86	3000.16	1.12	1.49	231.29	1122.73	2.83	5.19	2754.52	3000.42	2273.30	3000.83
4	5.01	7.92	3000.36	3000.74	1.93	3.49	1794.73	3000.11	2.03	3.03	2245.70	3000.31	1570.16	3000.78
	3.79	4.87	3000.13	3000.19	2.21	3.44	2755.58	3000.10	1.73	2.25	2492.25	3000.93	1092.41	3000.26
	5.27	8.93	3000.18	3000.35	1.15	1.45	305.23	1416.47	1.41	1.75	732.04	3000.33	171.51	769.00
5	6.84	10.39	3000.18	3000.33	1.52	2.24	1269.91	3000.10	1.35	1.49	335.68	1527.04	61.15	149.64
	5.09	6.34	3000.14	3000.28	1.40	1.53	1558.33	3000.10	1.39	1.49	127.56	313.33	36.60	74.03
	8.24	13.64	3000.25	3000.49	1.00	1.48	209.89	1103.33	1.43	1.50	183.76	638.90	39.08	128.20
6	7.52	10.78	3000.15	3000.31	1.32	1.53	258.93	2043.78	1.42	1.49	64.74	165.73	16.26	45.25
	5.55	6.89	3000.20	3000.49	1.42	1.80	643.77	3000.09	1.38	1.50	30.13	78.00	9.99	19.30
	2.35	4.06	3642.66	4000.44	1.60	2.12	1274.96	3047.06	2.39	4.42	3106.21	4000.28	2155.16	4000.74
7	2.42	3.55	3648.03	4000.39	2.71	4.33	3542.36	4000.25	1.69	2.34	2764.87	4000.25	1853.41	4000.61
	2.49	3.59	3543.14	4000.52	2.90	4.75	3587.30	4000.25	1.49	1.91	701.07	3161.74	221.36	697.79
	5.99	9.91	4000.42	4000.73	1.30	1.75	650.54	3116.78	1.47	1.97	972.86	4000.21	283.62	1630.43
8	3.70	4.72	3930.49	4000.56	1.54	1.78	1961.96	4000.23	1.43	1.49	121.64	183.33	32.79	85.28
	2.45	4.37	3310.76	4000.41	1.49	1.69	3048.84	4000.20	1.38	1.50	20.14	68.20	9.43	22.50
	7.15	10.83	4000.31	4000.54	1.15	1.47	122.54	539.54	1.41	1.50	270.72	927.81	45.14	175.24
9	4.17	5.03	4000.28	4000.38	1.44	1.84	641.60	4000.15	1.45	1.50	54.08	130.56	16.16	25.91
	2.62	4.17	3255.19	4000.82	1.44	1.73	798.78	4000.17	1.17	1.46	22.44	72.99	5.85	9.33
	11.30	92.41	2960.44	4000.78	2.35	4.77	3232.84	4000.94	6.97	9.00	4000.74	4000.90	4000.52	4000.82
10	5.59	6.96	4000.40	4000.80	5.59	6.99	4000.52	4000.70	4.03	5.13	4000.53	4000.88	4000.35	4000.65
	3.52	4.64	4000.69	4000.94	5.63	6.59	4000.40	4000.47	3.04	4.10	4000.22	4000.89	4000.63	4000.91
	9.98	23.02	4000.87	4000.93	2.09	4.03	3466.38	4000.45	3.66	5.48	4000.80	4000.87	3858.22	4000.61
11	23.23	91.75	4000.49	4000.95	3.57	4.67	4000.43	4000.52	2.18	2.69	4000.41	4000.90	4000.25	4000.84
	20.46	88.27	4000.22	4000.49	1.73	3.89	2403.36	4000.20	1.69	1.98	3245.65	4000.90	2593.36	4000.64
	NS	NS	NS	NS	1.75	2.78	2623.05	4000.44	2.15	3.35	3411.07	4000.90	2520.12	4000.14
12	22.42	92.25	4000.51	4000.81	2.57	4.16	4000.40	4000.50	1.51	1.58	2722.55	4000.40	1321.27	3715.67
	NS	NS	NS	NS	3.01	4.86	3875.29	4000.76	1.42	1.48	766.44	2659.59	434.76	2186.92

Table 5 DCFLP Computational Results Decreasing Demand

	Branch & Cut				Lagrangian Relaxation				Benders' Decomposition				ε-Optimal	
	Gap (%)		Time (sec.)		Gap (%)		Time (sec.)		Gap (%)		Time (sec.)		Time (sec.)	
Cluster	Avg.	Max.	Avg.	Max.	Avg.	Max.	Avg.	Max.	Avg.	Max.	Avg.	Max.	Avg.	Max.
	4.30	7.76	2877.76	3000.23	0.73	1.19	1.71	2.08	1.25	1.49	23.55	43.17	3.88	7.36
1	5.20	8.10	3000.23	3000.55	1.10	1.49	13.77	55.13	1.42	1.50	20.92	35.39	4.51	8.66
	4.08	5.80	2966.01	3000.30	1.26	1.47	67.94	251.33	1.42	1.50	13.65	24.06	2.87	6.20
	10.23	12.62	3000.06	3000.13	0.56	0.91	1.48	2.14	1.46	1.50	14.46	21.19	2.15	4.02
2	7.89	10.88	3000.11	3000.27	0.89	1.47	19.18	178.47	1.36	1.49	10.31	23.31	2.12	3.64
	5.68	7.91	3000.19	3000.39	1.00	1.36	9.17	55.25	1.39	1.49	2.03	5.83	1.26	2.53
	12.65	14.92	3000.22	3000.51	0.37	0.68	1.40	1.83	1.36	1.50	12.17	19.03	1.52	3.17
3	8.80	11.11	3000.16	3000.34	0.84	1.29	3.06	15.98	1.38	1.50	3.36	18.64	1.30	2.09
	6.11	8.69	3000.18	3000.36	0.85	1.45	8.13	72.83	1.27	1.46	2.65	13.17	1.10	1.70
	8.24	11.61	3000.15	3000.34	1.01	1.42	48.09	327.79	1.48	1.50	229.89	329.06	40.78	64.08
4	7.89	10.93	3000.16	3000.30	1.36	1.94	745.26	3000.08	1.41	1.50	171.23	671.96	42.28	196.13
	6.35	8.22	3000.15	3000.28	1.45	2.04	1386.82	3000.10	1.42	1.50	86.69	212.28	22.15	50.03
	13.28	18.31	3000.16	3000.28	0.66	1.21	9.30	16.30	1.44	1.49	58.26	86.16	8.97	16.31
5	10.10	12.80	3000.13	3000.26	0.98	1.49	175.36	1148.02	1.44	1.50	37.31	125.72	7.50	21.53
	6.39	8.16	3000.11	3000.26	1.13	1.42	53.89	185.66	1.34	1.50	15.56	37.31	4.98	8.11
	15.80	20.55	3000.26	3000.53	0.42	0.88	7.18	15.38	1.42	1.50	34.38	55.08	4.31	9.16
6	11.25	13.13	3000.22	3000.81	1.06	1.49	83.76	505.16	1.41	1.49	12.72	34.55	4.01	7.67
	7.20	9.29	3000.24	3000.50	1.00	1.49	38.07	269.71	1.22	1.47	7.71	27.34	3.19	6.42
	7.52	13.85	4000.51	4000.94	1.25	2.10	395.88	3029.97	1.43	1.50	690.45	2741.11	161.69	424.77
7	4.11	5.58	4000.29	4000.36	1.45	1.92	1531.85	4000.25	1.45	1.49	288.73	994.61	61.67	240.55
	2.10	3.16	3612.82	4000.50	1.69	2.25	2041.14	4000.29	1.34	1.42	79.85	267.88	107.79	790.78
	9.34	11.82	4000.22	4000.43	0.94	1.43	83.87	573.92	1.42	1.49	190.76	324.24	32.24	58.38
8	5.17	6.72	4000.28	4000.41	1.29	1.46	337.72	1724.95	1.42	1.50	58.92	179.52	16.06	29.11
	2.54	3.84	4000.35	4000.66	1.23	1.47	231.88	1340.98	1.29	1.49	8.30	16.69	7.92	24.84
	10.70	13.03	4000.36	4000.53	0.65	1.23	28.11	49.19	1.43	1.48	107.85	194.52	21.17	44.44
9	5.93	8.07	4000.30	4000.42	1.12	1.41	96.16	721.89	1.34	1.50	21.71	78.59	8.04	10.69
	2.69	3.74	4000.33	4000.53	1.14	1.43	8.67	15.28	1.19	1.49	6.29	15.97	6.85	18.44
	59.75	94.95	4000.47	4000.58	1.82	2.74	2956.17	4000.41	3.22	4.67	4000.68	4000.98	4000.29	4000.55
10	23.49	92.22	4000.44	4000.63	2.45	3.15	4000.39	4000.50	1.85	2.43	3690.20	4000.45	3016.34	4000.57
	NS	NS	NS	NS	4.14	5.23	4000.38	4000.40	1.52	2.16	1283.12	3165.85	1476.95	4000.85
	NS	NS	NS	NS	1.66	2.75	1815.96	4000.76	1.47	1.86	1936.23	4000.41	741.09	1884.45
11	NS	NS	NS	NS	1.69	3.32	2251.07	4000.48	1.43	1.49	587.50	1307.02	132.29	257.42
	NS	NS	NS	NS	1.69	3.36	2382.17	4000.39	1.33	1.49	133.61	555.84	59.23	151.97
	NS	NS	NS	NS	1.39	2.87	1401.02	4000.61	1.37	1.48	605.46	1370.67	141.36	292.66
12	NS	NS	NS	NS	1.32	2.13	1597.51	4000.39	1.36	1.50	223.60	571.92	43.55	116.36
	NS	NS	NS	NS	1.14	1.81	615.66	4000.29	1.15	1.48	63.03	236.61	28.05	73.55

Table 6 DCFLP Computational Results Steady Demand

	Branch & Cut				Lagrangian Relaxation				Benders' Decomposition				ε-Optimal	
	Gap (%)		Time (sec.)		Gap (%)		Time (sec.)		Gap (%)		Time (sec.)		Time (sec.)	
Cluster	Avg.	Max.	Avg.	Max.	Avg.	Max.	Avg.	Max.	Avg.	Max.	Avg.	Max.	Avg.	Max.
	1.89	3.72	1082.90	3000.13	0.63	1.46	2.73	7.69	1.43	1.50	77.54	333.28	31.61	142.72
1	4.02	5.37	2906.78	3000.50	1.23	1.47	6.56	16.11	1.36	1.49	29.00	55.02	18.38	66.69
	2.80	3.89	2532.03	3000.47	1.34	1.49	32.54	185.95	1.42	1.50	23.60	39.02	12.06	47.13
	10.50	13.14	3000.18	3000.33	0.97	1.44	3.53	13.81	1.46	1.50	25.18	34.66	13.66	43.78
2	8.03	10.08	3000.34	3001.08	1.14	1.49	3.46	8.64	1.41	1.48	14.37	21.59	9.30	36.55
	4.79	6.27	3000.23	3000.42	1.25	1.46	5.78	15.91	1.41	1.50	6.49	13.91	6.83	27.03
	14.06	16.94	3000.12	3000.30	0.53	1.45	1.96	4.63	1.35	1.50	22.09	32.44	8.75	29.25
3	9.67	11.45	3000.21	3000.45	1.04	1.46	3.32	19.75	1.35	1.50	10.58	32.59	5.74	21.33
	5.95	7.56	3000.15	3000.20	1.12	1.41	5.73	20.92	1.38	1.50	3.94	10.91	4.59	18.75
	3.48	6.70	1948.88	3000.23	0.92	1.46	19.75	119.52	2.53	4.62	2117.48	3000.25	1390.26	3000.69
4	6.28	8.15	3000.16	3000.33	1.28	1.46	436.98	1261.06	1.43	1.55	1184.42	3000.28	339.79	871.75
	4.99	7.07	3000.20	3000.36	1.38	1.92	1098.22	3000.13	1.52	2.24	949.24	3000.38	363.00	2141.07
	9.96	14.97	3000.17	3000.32	0.72	1.23	29.85	100.25	1.49	1.69	644.01	3000.31	88.14	270.30
5	8.49	10.33	3000.44	3000.60	1.10	1.48	230.97	1008.91	1.35	1.50	135.82	244.09	44.26	75.84
	6.37	7.96	3000.18	3000.33	1.32	1.64	560.01	3000.06	1.39	1.49	75.95	171.03	26.86	95.88
	13.66	17.63	3000.20	3000.33	0.46	1.00	71.21	272.63	1.45	1.50	145.24	330.70	35.98	91.89
6	9.47	11.14	3000.41	3000.55	1.07	1.46	283.12	946.95	1.40	1.50	60.15	106.52	18.79	55.72
	7.03	8.42	3000.18	3000.33	1.18	1.46	219.61	977.23	1.35	1.49	34.82	60.83	13.25	56.92
	2.62	4.40	3313.10	4000.39	1.11	1.50	103.41	447.45	2.32	4.28	2906.04	4000.42	2048.47	4000.63
7	2.06	2.88	3502.42	4000.76	1.62	2.91	1140.63	4000.19	1.73	2.66	3142.59	4000.90	1871.93	4000.90
	1.84	2.44	2921.99	4000.41	2.94	3.70	4000.19	4000.25	1.61	2.35	1992.24	4000.16	1993.18	4000.79
	6.54	8.49	4000.32	4000.38	1.26	1.86	514.59	4000.22	1.46	1.72	1562.13	4000.27	311.82	823.94
8	3.62	4.32	4000.31	4000.48	1.34	1.62	462.34	1915.77	1.47	1.50	174.88	284.59	108.51	446.72
	1.59	1.91	2515.35	4000.37	1.86	4.22	1890.40	4000.27	1.45	1.49	74.73	298.89	67.48	395.17
	8.78	12.33	4000.34	4000.48	1.06	1.49	212.08	1580.67	1.47	1.50	291.83	745.99	73.42	148.41
9	4.36	5.17	4000.30	4000.36	1.31	1.49	310.68	1299.27	1.45	1.50	79.67	167.80	31.78	68.06
	1.79	2.22	3457.48	4000.77	1.40	1.64	807.51	4000.14	1.35	1.50	26.38	78.03	18.64	76.06
	2.18	4.72	3474.67	4000.90	1.13	1.79	1072.46	4000.31	4.59	6.08	4000.54	4000.92	4000.57	4000.98
10	2.85	3.79	4000.63	4000.88	1.90	3.96	3252.77	4000.69	2.91	4.54	4000.45	4000.98	4000.48	4000.86
	5.34	13.31	4000.49	4000.69	5.49	6.84	4000.44	4000.50	2.47	3.18	4000.27	4000.92	3897.80	4000.91
	6.52	8.54	4000.58	4000.79	1.09	2.04	1050.06	4000.30	2.26	3.54	2855.31	4000.89	2964.53	4000.98
11	4.09	4.74	4000.59	4000.97	2.26	7.96	1593.08	4000.38	1.64	2.45	3027.55	4000.34	2214.34	4000.96
	27.41	86.22	3864.43	4000.70	2.73	3.85	3885.88	4000.61	1.42	1.50	1206.68	3133.23	1025.62	2386.57
	17.70	94.27	4000.63	4000.91	1.02	1.79	822.20	4000.31	1.68	2.58	2209.85	4000.12	1017.06	3712.49
12	13.32	90.43	4000.50	4000.86	1.35	1.77	1328.75	4000.42	1.44	1.60	683.11	4000.44	221.36	777.68
	NS	NS	NS	NS	1.71	3.00	2273.73	4000.67	1.39	1.49	91.33	431.36	117.38	363.13

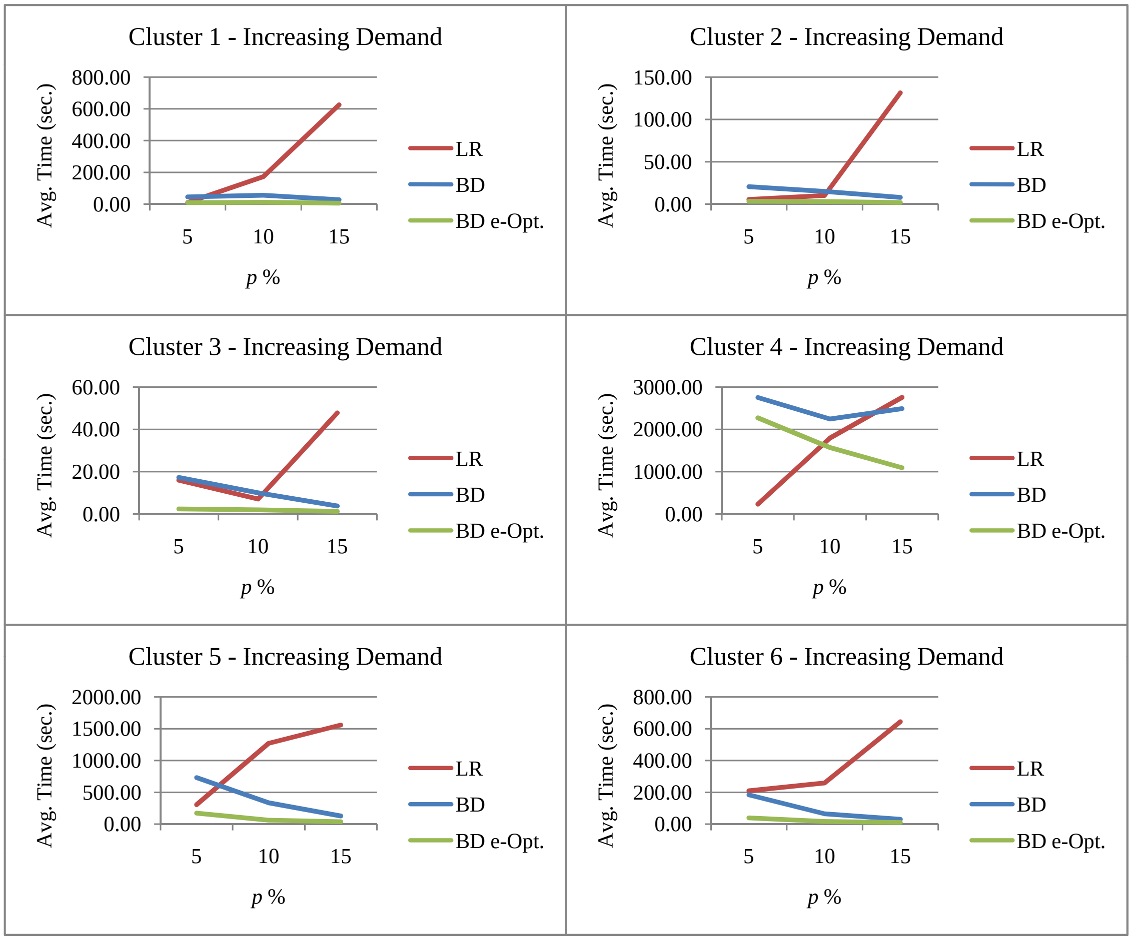
Table 7 Average Cost Split

	Increasing Total Demand Cost Split (%)				*Decreasing Total Demand Cost Split* (%)				*Steady Total Demand Cost Split* (%)			
Cluster	Tran.	Oper.	Open.	Clos.	Tran.	Oper.	Open.	Clos.	Tran.	Oper.	Open.	Clos.
	49.13	43.37	7.50	0.00	29.70	56.20	13.44	0.65	56.47	37.24	6.29	0.00
1	23.18	63.79	12.99	0.03	15.35	68.30	15.34	1.01	24.41	64.26	11.28	0.04
	16.34	79.14	4.52	0.00	9.98	73.04	16.01	0.97	16.46	71.00	12.50	0.03
	28.50	59.04	12.46	0.00	18.47	66.07	14.75	0.72	41.33	50.35	8.32	0.00
2	14.20	71.72	14.04	0.05	9.04	74.27	15.76	0.93	15.00	72.80	12.16	0.04
	9.83	85.54	4.63	0.00	5.68	77.24	16.10	0.98	9.83	77.24	12.90	0.03
	21.40	64.59	13.94	0.07	13.39	70.53	15.33	0.75	21.12	67.50	11.25	0.12
3	10.30	75.16	14.50	0.04	6.59	76.46	15.96	1.00	10.91	76.52	12.52	0.04
	6.82	78.25	14.87	0.06	4.14	78.86	16.07	0.93	6.97	79.89	13.11	0.03
	54.77	40.44	4.65	0.14	35.23	57.05	7.20	0.52	58.89	37.78	3.32	0.00
4	30.50	62.22	7.19	0.08	19.20	71.47	8.68	0.64	26.57	67.10	6.17	0.16
	20.31	71.20	8.33	0.15	12.57	77.53	9.20	0.70	17.51	75.45	6.86	0.18
	37.12	55.75	7.00	0.13	22.93	68.18	8.28	0.61	33.20	60.79	5.71	0.30
5	18.78	73.07	8.03	0.12	11.55	78.67	9.07	0.72	17.16	75.86	6.76	0.22
	12.64	81.03	6.33	0.00	7.40	82.55	9.33	0.71	11.11	81.53	7.15	0.21
	29.82	62.77	7.34	0.08	17.28	73.18	8.86	0.68	25.65	67.68	6.33	0.33
6	14.67	77.03	8.24	0.06	8.84	81.19	9.26	0.71	12.55	80.15	7.09	0.21
	9.00	83.67	7.33	0.00	5.30	84.54	9.41	0.75	8.13	84.30	7.34	0.23
	36.31	52.40	11.17	0.13	26.02	59.87	13.31	0.79	40.22	50.86	8.92	0.00
7	19.63	66.27	13.95	0.15	12.72	71.13	15.30	0.85	22.45	65.94	11.59	0.02
	12.93	72.49	14.51	0.07	8.17	74.91	16.00	0.91	14.46	72.51	12.97	0.06
	24.43	63.40	12.17	0.00	16.07	68.61	14.46	0.85	26.80	62.64	10.56	0.00
8	12.46	73.78	13.76	0.00	7.44	76.09	15.55	0.92	13.54	74.02	12.42	0.02
	8.30	77.58	14.13	0.00	4.75	78.42	15.92	0.91	8.95	78.02	13.02	0.02
	18.11	68.92	12.97	0.00	11.62	72.56	14.93	0.90	20.31	68.28	11.38	0.03
9	9.05	77.03	13.92	0.00	5.57	77.92	15.58	0.93	9.86	77.34	12.78	0.02
	5.93	79.78	14.29	0.00	3.36	79.79	15.92	0.93	6.49	80.30	13.19	0.02
	46.82	47.83	5.32	0.03	31.72	60.54	7.21	0.52	43.18	52.19	4.62	0.01
10	25.53	66.18	8.10	0.19	16.23	74.25	8.87	0.66	23.56	69.75	6.60	0.09
	16.61	74.10	9.11	0.18	10.10	79.56	9.58	0.76	15.31	77.28	7.31	0.10
	33.31	59.98	6.66	0.05	20.97	70.42	8.02	0.60	29.34	65.08	5.55	0.03
11	16.49	75.03	8.34	0.14	9.71	80.63	9.01	0.65	14.40	78.82	6.74	0.05
	10.74	80.56	8.61	0.10	6.19	83.84	9.28	0.68	9.34	83.51	7.11	0.05
	26.22	66.66	7.10	0.03	15.43	75.42	8.53	0.62	22.29	71.66	6.02	0.04
12	12.43	79.18	8.31	0.08	7.06	83.06	9.21	0.67	10.54	82.52	6.90	0.04
	8.10	83.30	8.53	0.07	4.50	85.42	9.39	0.69	6.95	85.84	7.18	0.03

From the results given in Tables 4, 5, and 6 on pages 68 to 70, we see that in general, the performance of Lagrangian relaxation and Benders' decomposition algorithms outperformed the branch and cut procedure of CPLEX. Even for the smaller classes, the average solution time using CPLEX was higher than our solution algorithms for all classes.

From the results given in Table 4 on page 68 for total increasing demand, we observe that Lagrangian relaxation performs better for small problems, contrary to Benders' decomposition which requires less computational time to solve large problems. Within a cluster of classes, we note that the performance of the solution algorithm also depends on the value of p, the expected number of open facilities. As the value of p increases from 5 to 15%, the average solution time for Lagrangian relaxation increases, and for Benders' decomposition tends to decrease. Figures 11 and 12 show the behavior of the average solution time for each solution algorithm considering different values of p. In each figure, we denote Lagrangian relaxation by LR; Benders' decomposition by BD; and ε-optimal by BD ε-Opt. This observation indicates that the Lagrangian subproblem becomes more difficult to solve as the number of expected open facilities increases. For Benders' decomposition, having a larger number of open facilities increases the amount of fixed costs in the objective function value of the master problem. Since the dual subproblem considers the total fixed costs and total variable transportation costs, an increased number of open facilities decreases the difference in value between the master problem and dual subproblem. From Table 7 on page 71, we see that clusters 3, 6, 9, and 12 with a cost split of $70 - 80\%$ for fixed operation cost can be solved faster by Benders' decomposition. Clusters with a cost split structure of fixed operation cost near $45 - 60\%$ are solved faster by Lagrangian relaxation. The ε-optimal algorithm showed the best performance for clusters 1, 2, and 3. For classes of larger size, the performance of the ε-optimal algorithm improved

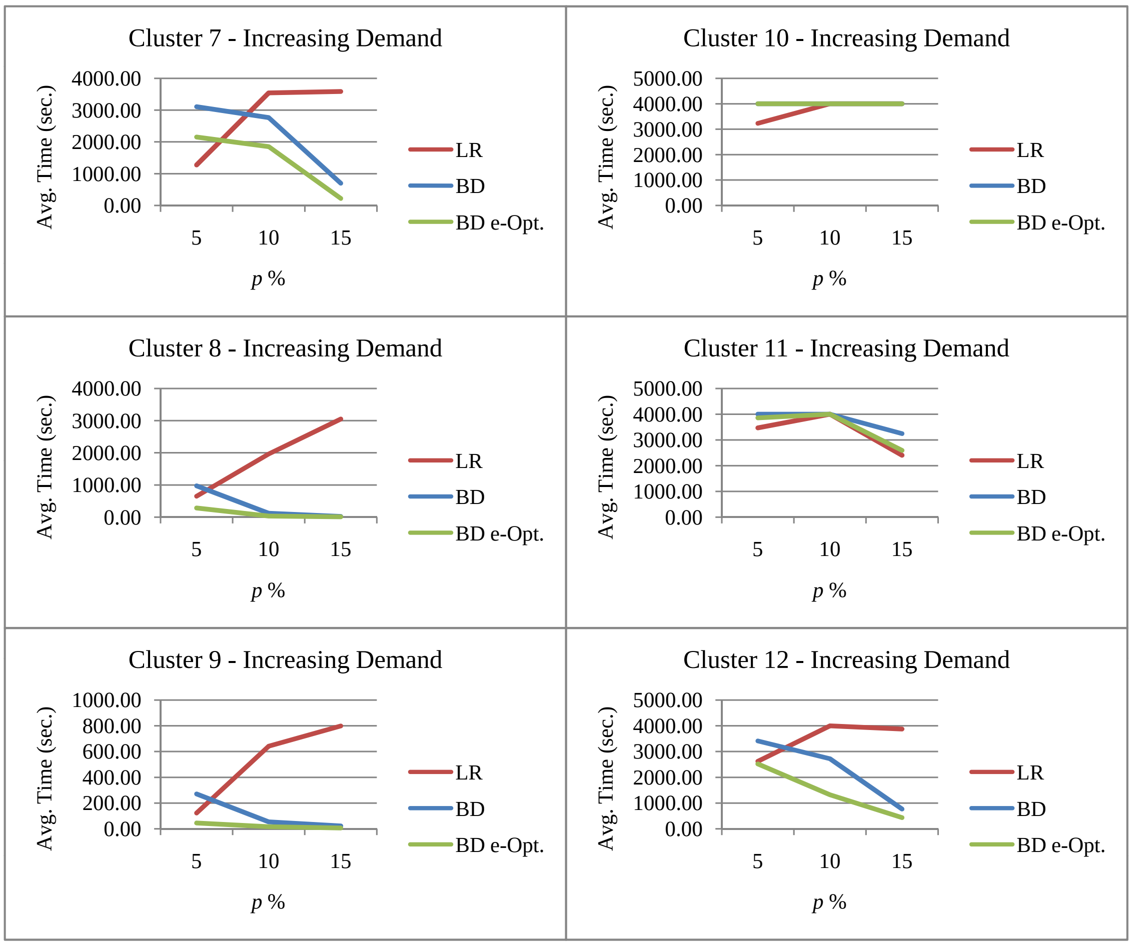
Figure 11 Average Solution Time Increasing Demand (Clusters 1 to 6)



when the fixed operation cost represented $70 - 80\%$ of the cost split. For the rest of the classes, we observed that the algorithm spent too much time looking for a feasible solution within the optimality criteria. Since we limited the solution time, only classes with an average and maximum solution time lower than the maximum time are considered to be 1.5% optimal.

From Table 5 on page 69, we see that classes with total decreasing demand were solved faster than increasing and steady total demand. Similar to classes with increasing demand, a pattern in the average solution time is present for each cluster of classes. Observe that, as the expected number of open facilities increases, the average

Figure 12 Average Solution Time Increasing Demand (Clusters 7 to 12)



solution time given by Lagrangian relaxation increases and for Benders' decomposition decreases. Again, increasing the value of the fixed operation cost reduces the average solution time for both decomposition algorithms. In this case, since demand is decreasing, the proportion of fixed costs in the objective function value becomes larger compared to the variable transportation cost. This cost structure still benefits the Benders' decomposition algorithm in terms of the objective function value of the master problem. From Table 7 on page 71, Benders' decomposition performs better for classes with a cost split of $70-85\%$ for fixed operation cost. For Lagrangian relaxation, having decreasing demand leads to a faster process with a reduced number of open facilities in the Lagrangian subproblem. Also, the adjustment of the Lagrange multipliers is faster in correcting the violations to the capacity constraints. Figures 13 and 14 show the behavior of the average solution time for each cluster of classes for different values of p. From Table 7 on page 71, observe that the cost split shows a higher average percentage value for fixed operation costs than for increasing demand; this is because the demand is decreasing and the total transportation cost represents a lower percentage of the objective function value. The ε-optimal algorithm attained its best performance for classes with decreasing demand. For almost all the clusters, it solved all the problems within 1.5% optimality with the lowest average solution time, outperforming Lagrangian relaxation and Benders' decomposition algorithms.

For steady total demand, we see a similar pattern in Table 6 on page 70 for the average solution time. For classes of problems with a cost split of $70-85\%$ for fixed operation cost, Benders' decomposition obtained lower average solution. For Lagrangian relaxation, we observe that average solution time increases as the fixed operation cost increases. Lagrangian relaxation performed better for classes where the cost split has an average fixed operation cost of less than 65%. Figures 15 and 16 show the behavior of the average solution time for each cluster of classes and for

Figure 13 Average Solution Time Decreasing Demand (Clusters 1 to 6)

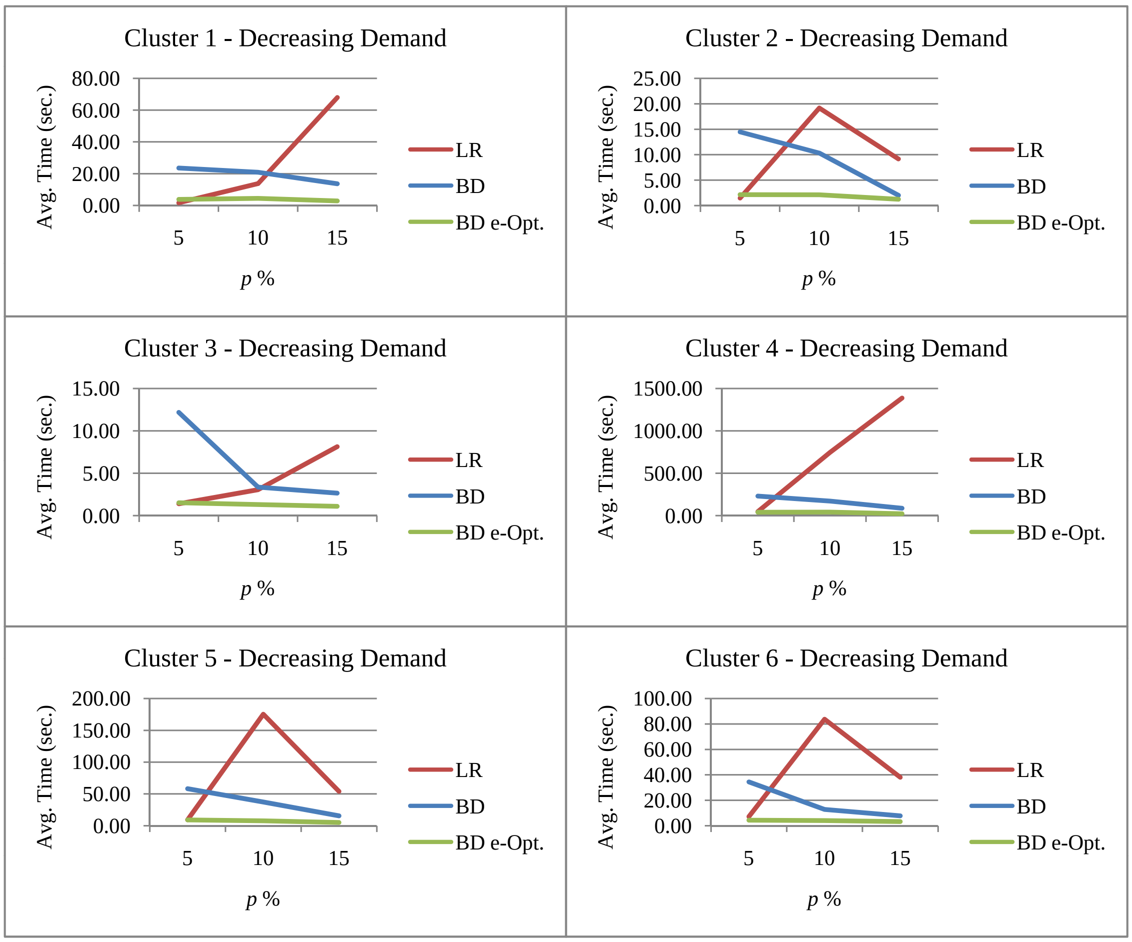


Figure 14 Average Solution Time Decreasing Demand (Clusters 7 to 12)

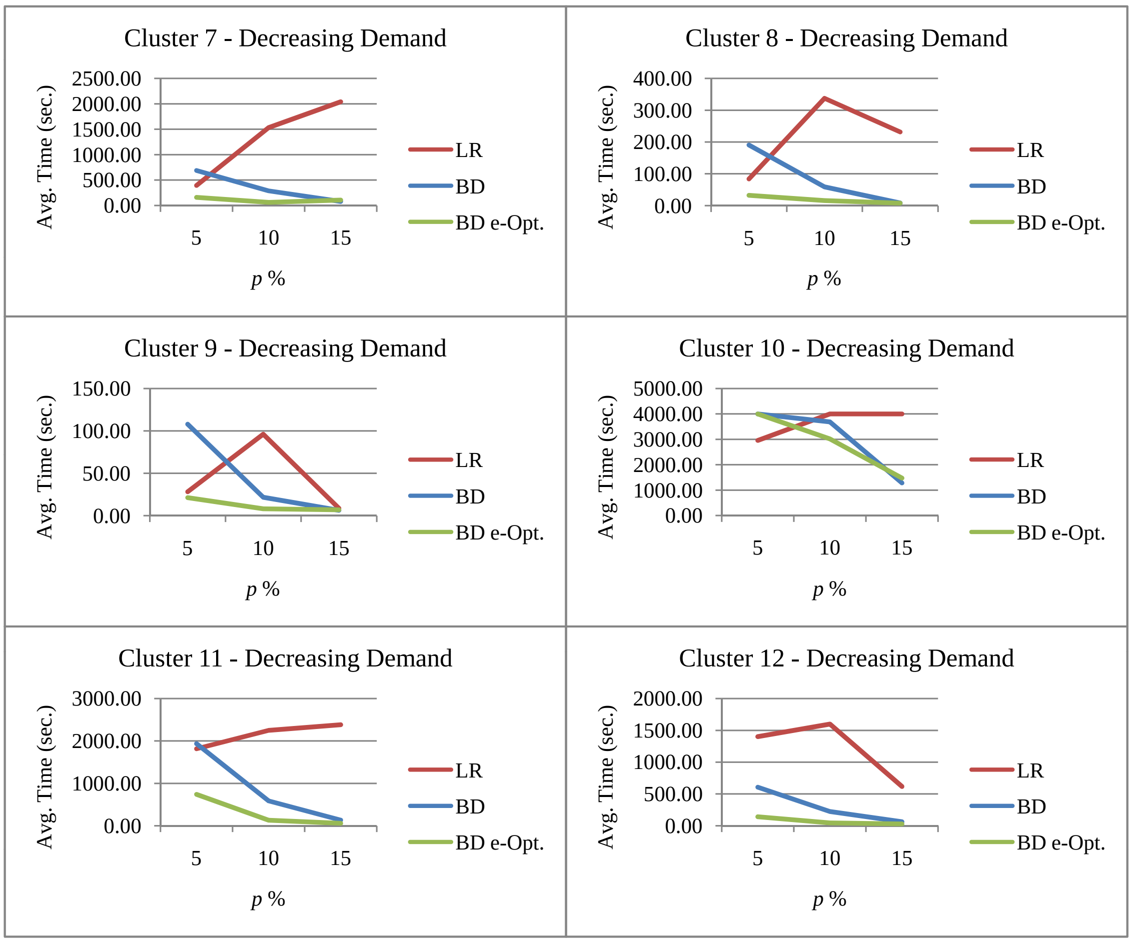
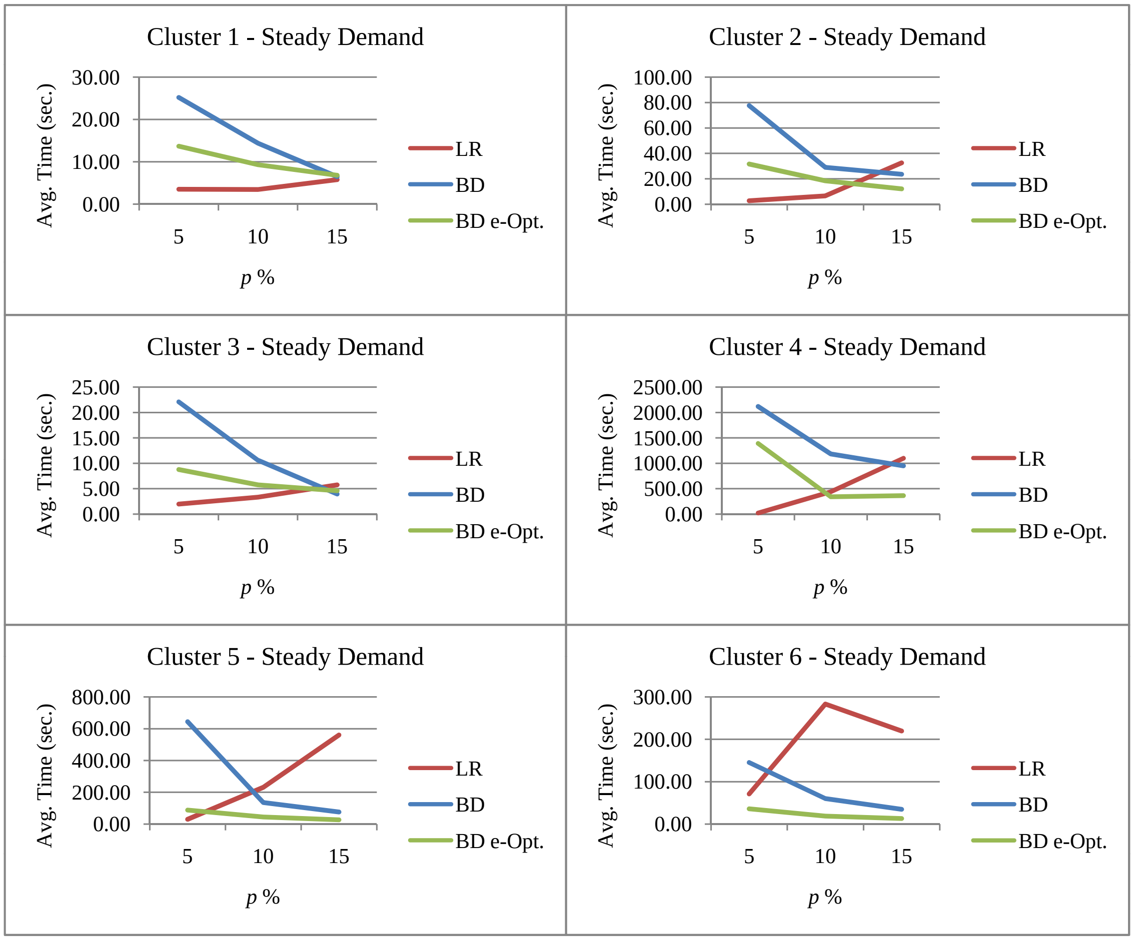
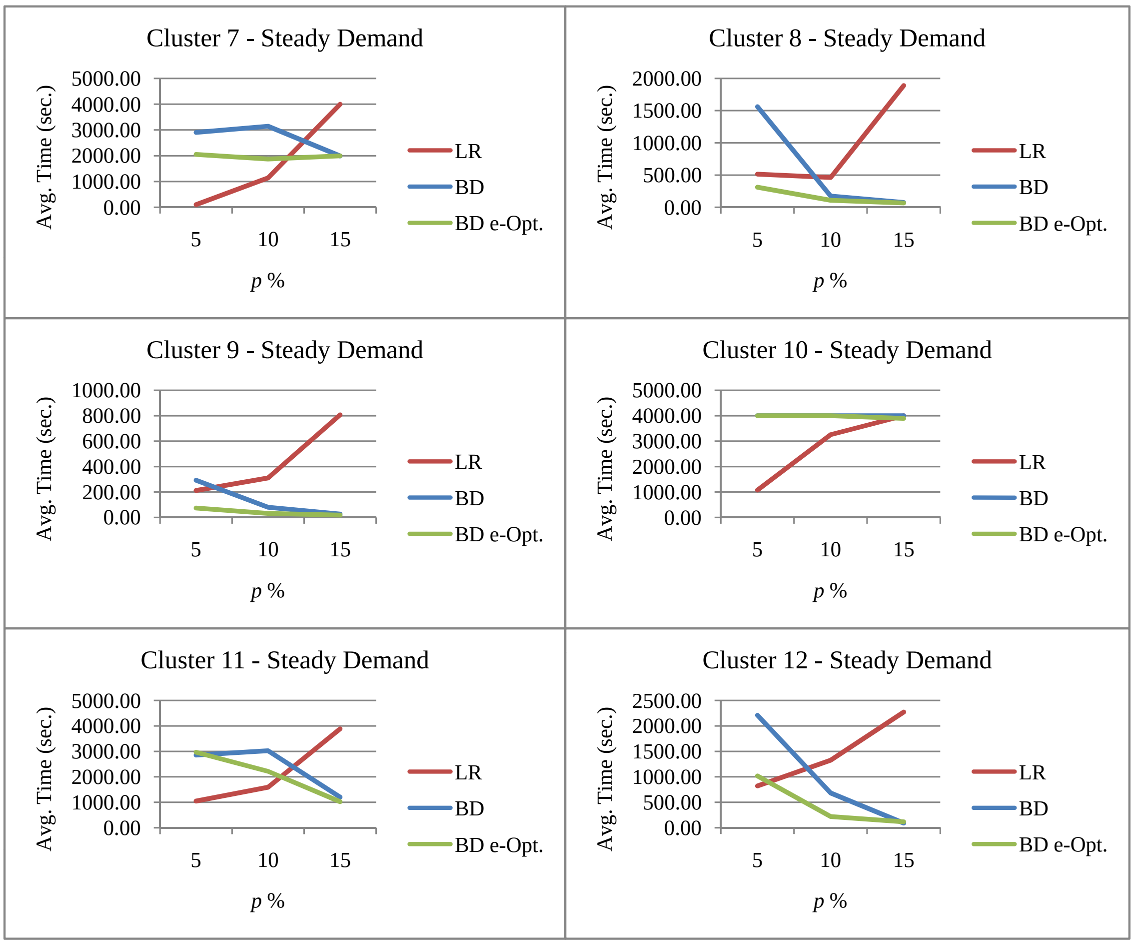


Figure 15 Average Solution Time Steady Demand (Clusters 1 to 6)



different values of p. The ε-optimal algorithm showed a good performance for classes with $\tau = 5$ periods and for classes with a cost split of $75 - 80\%$ of fixed operation cost.

Figure 16 Average Solution Time Steady Demand (Clusters 7 to 12)



IV.5. Summary and Conclusions

In this chapter we described the DCFLP and presented a mixed integer programming formulation. We developed a Lagrangian relaxation and Benders' decomposition algorithms to solve the model. Both algorithms showed to be more efficient compared with conventional branch and cut in solving classes of problems for each total demand structure. We observed that the efficiency of the solution algorithms depends on the cost structure and demand pattern considered. Lagrangian relaxation performed better for classes of problems with a smaller number of open facilities, and for classes where the fixed operation cost represents $50-65\%$ of the average total cost. Benders' decomposition performed better for classes of problems with a larger number of open facilities and for classes of problems where the cost split of the average total cost considered $70-85\%$ of fixed operation costs. The ε-optimal algorithm performed better for classes of problems with total decreasing demand, in particular for small size problems and classes with a cost split of $70-80\%$ for fixed operation cost.

CHAPTER V

DYNAMIC DEMAND CAPACITATED FIXED CHARGE LOCATION PROBLEM WITHOUT RELOCATION (DDCFLP)

In this chapter, we investigate the problem of finding the locations of facilities with limited capacity to satisfy the demand of a set of customers over a discrete and finite time horizon when relocation of facilities is not allowed. The demand of each customer is assumed to be time varying (in a known way) and can be split or served by one or more facilities. There are fixed costs associated with establishing or opening new facilities and for operating the facilities. Also, there is a variable transportation cost for serving the demand of customers. The main objective is to find an optimal set of locations for facilities to satisfy the time varying demand while observing the capacity restrictions over the time horizon.

The chapter is organized as follows. In Section V.1, we give the problem statement. Section V.2, presents the mixed integer programming formulation and notation for the DDCFLP. In Section V.3, we develop a Benders' decomposition algorithm to solve the DDCFLP. In Section V.4, we present numerical results using different total demand patterns to test the performance of the solution algorithm. In Section V.5, we show that when relocation costs (for opening and closing facilities) are considerably large, the DDCFLP can be solved as a special case using the DCFLP model. Finally, in Section V.6, we summarize the results and give concluding remarks.

V.1. Problem Statement

The DDCFLP statement is as follows. Consider a geographical region where a given group of customers are dispersed. Each customer has a given demand for a certain product. Along a discrete and finite time horizon, the total demand of the customers is time varying in a known way.

The establishment of facilities is required to supply the demand of customers over the entire time horizon. Each facility has a limit or capacity in the amount of demand that can be supplied to the customers. Each customer can be supplied by one or more facilities. The shipments of demand between facilities and customers incur a variable transportation cost proportional to quantity and distance. Further, the establishment of a new facility incurs a fixed opening cost, which can represent the initial investment for construction, equipment, and resources needed to start operations. An additional fixed operation cost is incurred in each period the facility remains operational; this can be thought of as the per period cost associated with the initial investment or the total expenses for services and labors. Establishment of facilities takes place at once in the beginning of the time horizon and can not be closed or relocated.

The main decisions are determining the number of facilities required to supply the demand, selecting the locations to establish the facilities at the beginning of the time horizon, and allocating demand to facilities in such a way that the total fixed and variable costs are minimal without exceeding the capacity of facilities over the entire time horizon.

V.2. Model and Notation

In this section, we provide a mixed integer programming formulation of the DDCFLP. We use the following notation.

Parameters

I set of demand locations, $i = 1, \ldots, n$

J set of facility locations, $j = 1, \ldots, m$

T set of periods, $t = 1, \ldots, \tau$

f_{jt} fixed cost for having a facility open (operating) in location j during period t

a_j fixed cost for opening a new facility in location j

w_{it} amount of demand in location i during period t

q_j capacity available if a facility is open at location j

d_{ij} distance between facility at location j to customer i

α per unit distance per unit demand cost

c_{ijt} transportation cost for shipping demand of location i from facility at location j in period t, $c_{ijt} = \alpha w_{it} d_{ij}$

Decision Variables

x_{ijt} fraction of demand of location i shipped from facility at location j in period t

y_j 1 if a facility is open in location j, 0 otherwise

$$\text{(DDCFLP)} \quad \min \sum_{i \in I} \sum_{j \in J} \sum_{t \in T} c_{ijt} x_{ijt} + \sum_{j \in J} \sum_{t \in T} f_{jt} y_j + \sum_{j \in J} a_j y_j \tag{5.1}$$

subject to

$$\sum_{j \in J} x_{ijt} = 1 \qquad i \in I,\ t \in T \tag{5.2}$$

$$x_{ijt} \leq y_j \qquad i \in I, j \in J, t \in T \tag{5.3}$$

$$\sum_{i \in I} w_{it} x_{ijt} \leq q_j y_j \qquad j \in J,\ t \in T \tag{5.4}$$

$$x_{ijt} \geq 0, y_j \in \{0, 1\} \qquad i \in I,\ j \in J,\ t \in T \tag{5.5}$$

The objective function (5.1) includes the total cost over the time horizon; it has three main components. The first component represents the total transportation cost between facilities and customers. The second component represents the total fixed cost for operating open facilities. Finally, the third component represents the total fixed cost for establishing facilities at the beginning of the time horizon. The constraints (5.2) are the demand constraints (for each customer, all the demand must be met), (5.3) ensure that demand is allocated to open facilities, (5.4) are the capacity constraints (no facility can supply more than its capacity), and (5.5) are the nonnegativity and integrality constraints.

V.3. Solution Procedure

In this section we develop a Benders' decomposition algorithm to solve the DDCFLP.

V.3.1. Benders' Decomposition

The special primal structure of the DDCFLP makes it a good candidate for Benders' decomposition. For fixed values of the location variables, $\hat{y}_j$, we obtain the following Benders' subproblem:

$$(SP_y) \quad \min \sum_{i \in I} \sum_{j \in J} \sum_{t \in T} c_{ijt} x_{ijt} \tag{5.6}$$

subject to

$$\sum_{j \in J} x_{ijt} = 1 \qquad i \in I,\, t \in T \tag{5.7}$$

$$x_{ijt} \leq \hat{y}_j \qquad i \in I,\, j \in J,\, t \in T \tag{5.8}$$

$$\sum_{i \in I} w_{it} x_{ijt} \leq q_j \hat{y}_j \qquad j \in J,\, t \in T \tag{5.9}$$

$$x_{ijt} \geq 0 \qquad i \in I,\, j \in J,\, t \in T \tag{5.10}$$

And the associated dual of the subproblem:

$$(DSP_y) \quad \max \sum_{i \in I} \sum_{t \in T} \lambda_{it} - \sum_{i \in I} \sum_{j \in J} \sum_{t \in T} \hat{y}_j \mu_{ijt} - \sum_{j \in J} \sum_{t \in T} q_j \hat{y}_j \gamma_{jt} \tag{5.11}$$

subject to

$$\lambda_{it} - \mu_{ijt} - w_{it}\gamma_{jt} \leq c_{ijt} \qquad i \in I,\ j \in J,\ t \in T \tag{5.12}$$

$$\lambda_{it} \text{ unrestricted}, \mu_{ijt} \geq 0, \gamma_{jt} \geq 0 \qquad i \in I,\ j \in J,\ t \in T \tag{5.13}$$

which can be further decomposed into $|T|$ independent dual subproblems (one for each period $t \in T$). Since the feasible region of the primal subproblem is non-empty and bounded, we do not need to consider the extreme rays of the feasible region of the dual subproblem.

We have the following Benders' master problem:

$$(MP_K) \quad \min \rho + \sum_{j \in J} \sum_{t \in T} f_{jt} y_j + \sum_{j \in J} a_j y_j \tag{5.14}$$

subject to

$$\rho \geq \sum_{i \in I} \sum_{t \in T} \lambda_{it}^k - \sum_{i \in I} \sum_{j \in J} \sum_{t \in T} \mu_{ijt}^k y_j - \sum_{j \in J} \sum_{t \in T} q_j \gamma_{jt}^k y_j \qquad k \in K \subseteq P \tag{5.15}$$

$$\sum_{i \in I} w_{it} \leq \sum_{j \in J} q_j y_j \qquad t \in T \tag{5.16}$$

$$\rho \geq 0, y_j \in \{0, 1\} \qquad j \in J \tag{5.17}$$

where $\left\{(\lambda^k, \mu^k, \gamma^k) : k \in P\right\}$ denote all the extreme points of (DSP_y), K is an appropriate index set, and ρ denotes the objective function value of the dual subproblem. We can let $\rho \geq 0$ provided that $c_{ijt} \geq 0$. Adding the surrogate constraints (5.16) to the master problem guarantees that any solution to the master problem is a feasible solution (with enough capacity) for the primal and dual subproblems.

The number of extreme points of the dual subproblem can be very large, thus increasing the size and computational effort to solve the master problem. Observe that, in an optimal solution to the master problem, only a small subset of constraints (5.15) will be binding. Thus, we can consider only a subset of these constraints. Clearly, this relaxed master problem gives a lower bound on the optimal objective function value of the DDCFLP.

An upper bound can be obtained for fixed values of the location variables, $\hat{y}_j$, obtained from the solution to the master problem, solving the following transportation problem:

$$(TP_y) \quad \min \sum_{i \in I} \sum_{j \in J} \sum_{t \in T} c_{ijt} x_{ijt} + \sum_{j \in J} \sum_{t \in T} f_{jt} \hat{y}_j + \sum_{j \in J} a_j \hat{y}_j \tag{5.18}$$

subject to

$$\sum_{j \in J} x_{ijt} = 1 \qquad i \in I,\, t \in T \tag{5.19}$$

$$\sum_{i \in I} w_{it} x_{ijt} \leq q_j \hat{y}_j \qquad j \in J,\, t \in T \tag{5.20}$$

$$x_{ijt} \geq 0 \qquad i \in I,\, j \in J,\, t \in T \tag{5.21}$$

V.3.1.1. Generation of Strong Cuts

It is known that the Benders' subproblem (transportation problem) has a high level of degeneracy, thus the dual subproblem can have alternative optimal solutions. Since the improvement in the value of the lower bound (obtained from the solution to the relaxed master problem) is tightened by the Benders' cuts obtained from the dual subproblem, it is important that at each iteration of the decomposition procedure we select the best possible cut.

To strengthen the Benders' cuts obtained from the dual subproblem, we implement the algorithm proposed by Van Roy (1986). The values of the dual variables μ_{ijt} and γ_{jt} can be improved without affecting the objective function value of the dual subproblem for the closed facilities. Let $C = \{j \in J : y_j = 0\}$ denote the set of closed facilities, and $O = \{j \in J : y_j = 1\}$ the set of open facilities. Also let $j_{(i)}t$ denote the allocation of customer location i to candidate location j in period t, obtained from an optimal solution to (SP_y). Let $(\hat{\lambda}_{it}, \hat{\mu}_{ijt}, \hat{\gamma}_{jt})$ denote the value of the optimal dual variables obtained from the dual subproblem (DSP_y). We set $y_j = 1, j \in C$, then solve the following linear program:

$$(SC_y) \quad \max - \sum_{i \in I} \sum_{j \in J} \sum_{t \in T} \bar{\mu}_{ijt} y_j - \sum_{j \in J} \sum_{t \in T} q_j \bar{\gamma}_{jt} y_j \tag{5.22}$$

subject to

$$\hat{\lambda}_{it} - \bar{\mu}_{ijt} - w_{it} \bar{\gamma}_{jt} \leq c_{ijt} \qquad i \in I,\, j \in J,\, t \in T \tag{5.23}$$

$$\bar{\mu}_{ijt} \geq 0, \bar{\gamma}_{jt} \geq 0 \qquad i \in I,\, j \in J,\, t \in T \tag{5.24}$$

From the optimal solution to (SC_y), we set $\hat{\mu}_{ijt} = \bar{\mu}_{ijt}, \hat{\gamma}_{jt} = \bar{\gamma}_{jt}, j \in C$, and leave the previous values of the dual variables, $\hat{\mu}_{ijt}, \hat{\gamma}_{jt}, j \in O$, unchanged. Note that, constraints (5.23) guarantee that $(\hat{\lambda}_{jt}, \bar{\mu}_{ijt}, \bar{\gamma}_{jt})$ is a feasible solution to the dual subproblem.

V.3.1.2. Generation of Pareto-Optimal Cuts

The generation of strong cuts produces significant savings in computation time for the decomposition procedure. However, we can further improve the Benders' cuts by considering the closed and open facilities. This procedure relies on the concept of pareto-optimal cuts introduced by Magnanti and Wong (1981). The main idea is to generate a cut that dominates any other cut, that is, a constraint which is tighter than any other. This is called a pareto-optimal cut.

Wentges (1996) developed an algorithm to generate pareto-optimal cuts for the CFLP by considering the open and closed facilities. Observe that, from the relationship between the primal and dual subproblems, the value of the dual variables $\hat{\lambda}_{it}$ represents the cost for serving demand of customer i in period t, and the value of the dual variables $\hat{\mu}_{ijt}$ the cost for allocating costumer i to facility j in period t. The fair cost that customer i should pay for being served by facility $j_{(i)}$, which is closer and more convenient, can be thought of as the additional cost for being served by the second nearest facility. Thus, for the open facilities we can increase the value of $\hat{\lambda}_{it}$ and $\hat{\gamma}_{jt}$, and decrease the cost (or give a reward) of $\hat{\mu}_{ij_{(i)}t}$. In doing so, the objective function value of the dual subproblem remains unchanged and also constraints (5.12) are satisfied. However, the improvement on the value of the dual variables $\hat{\lambda}_{it}$ could be too high since the closed facilities are not considered. It is possible that customer i could be better served by one of the closed facilities. Thus, in addition to the open facilities we can improve the value of the dual variables considering the closed facilities.

The algorithm to develop pareto-optimal cuts improves the values of the dual variables by considering both the open and closed facilities. The additional service cost for the open facilities is determined between the first and second smallest costs in the set $\{c_{ijt}+\hat{\gamma_{jt}} : j \in O\}$. Note that if x_{ijt} happens to be in the basis of the primal subproblem, then $\hat{\lambda}_{it} = c_{ij_{(i)}t} + \hat{\gamma}_{j_{(i)}t}$, for some $j_{(i)}t \in O$ (complementary slackness). In selecting the additional service cost for the closed facilities we selected the third smallest value in the set $\{c_{ijt} + \bar{\gamma_{jt}} : j \in C\}$ as it gave the best improvement in the efficiency of the Benders' decomposition algorithm.

The pseudo-code of the algorithm is given in Display 4. The pseudo-code of the Benders' decomposition algorithm is given in Display 5.

Display 4 Pseudo-code pareto-optimal cuts for open and closed facilities

Solve SP_y, DSP_y and SC_y
for $i = 1$ to n **do**
  for $t = 1$ to τ **do**
    Determine smallest ψ_{it}, second smallest ϕ_{it} from: $\{c_{ijt} + \hat{\gamma}_{jt} : j \in O\}$
    Determine third smallest ℓ_{it} from: $\{c_{ijt} + \hat{\gamma}_{jt} : j \in C\}$
    Calculate $\theta_{it} \leftarrow \max\{0, \min\{\phi_{it} - \psi_{it}, \ell_{it} - \psi_{it}\}\}$
    Set $\mu^*_{ijt} \leftarrow 0 \; j \in O, j \neq j_{(i)}t$
    if $\theta_{it} > 0$ **then**
      Set $\bar{\lambda}_{it} \leftarrow \hat{\lambda}_{it} + \theta_{it}$, $\mu^*_{ij_{(i)}} \leftarrow \theta_{it}$
    else
      Set $\bar{\lambda}_{it} \leftarrow \hat{\lambda}_{it}$, $\mu^*_{ij_{(i)}t} \leftarrow 0$
    end if
    Solve SC_y again to calculate $\mu^*_{ijt}, \gamma^*_{jt}, j \in C$
  end for
end for
Return $(\bar{\lambda}_{it}, \mu^*_{ijt}, \gamma^*_{jt})$

Display 5 Pseudo-code Benders' decomposition algorithm

Initialize: $k \leftarrow 0, Z_{LB} \leftarrow -\infty, Z_{UB} \leftarrow \infty$
Solve MP_K
$Z_{lb}^k \leftarrow MP_K$
if $Z_{lb}^k > Z_{LB}$ **then**
    $Z_{LB} \leftarrow Z_{lb}^k$
end if
while $k \leq$ M **do**
    Solve DSP_y
    Set: $Z_{ub}^k \leftarrow DSP_y +$ fixed costs
    if $Z_{ub}^k < Z_{UB}$ **then**
        $Z_{UB} \leftarrow Z_{ub}^k$
        Record S
    end if
    if $(Z_{UB} - Z_{LB})/Z_{UB} \leq \varepsilon$ **then**
        Stop
    else
        Obtain pareto-optimal cut: $(\bar{\lambda}_{it}, \mu_{ijt}^*, \gamma_{jt}^*)$
        Solve MP_K with $(\bar{\lambda}_{it}, \mu_{ijt}^*, \gamma_{jt}^*)$
        Set: $Z_{lb}^k \leftarrow MP_K$
        if $Z_{lb}^k > Z_{LB}$ **then**
            $Z_{LB} \leftarrow Z_{lb}^k$
        end if
        if $(Z_{UB} - Z_{LB})/Z_{UB} \leq \varepsilon$ **then**
            Stop
        end if
    end if
    $k \leftarrow k + 1$
end while
Return S, Z_{UB}

V.4. Numerical Results

In this section we conduct a numerical experiment to test the performance of the decomposition algorithm developed for the DDCFLP. We designed 72 classes of problems considering three total demand structures, increasing, decreasing, and steady;

four values of $n = 50$, 100, 150, and 200 locations; two values of $\tau = 5$, and 10 periods; one value for $p = 0.15$; and three discrete uniform distributions to randomly generate the fixed operation cost, $U[100000, 150000]$, $U[200000, 250000]$, and $U[300000, 350000]$. The fixed opening cost, a_j, was generated as described in Chapter III, considering only the costs for the first period, i.e., $a_j = a_{j1}, j \in J$. For each class, we randomly generated 10 instances. For comparison purposes, we arranged the classes for each total demand structure into 8 clusters, each cluster containing three classes. Table 8 shows the arrangement of classes into eight clusters.

Table 8 DDCFLP Classes of Problems Arranged in Clusters

	Parameters					*Parameters*			
Cluster	n	τ	p %	f	Cluster	n	τ	p %	f
	50	5	15	$U[100000, 150000]$		150	5	15	$U[100000, 150000]$
1	50	5	15	$U[200000, 250000]$	5	150	5	15	$U[200000, 250000]$
	50	5	15	$U[300000, 350000]$		150	5	15	$U[300000, 350000]$
	50	10	15	$U[100000, 150000]$		150	10	15	$U[100000, 150000]$
2	50	10	15	$U[200000, 250000]$	6	150	10	15	$U[200000, 250000]$
	50	10	15	$U[300000, 350000]$		150	10	15	$U[300000, 350000]$
	100	5	15	$U[100000, 150000]$		200	5	15	$U[100000, 150000]$
3	100	5	15	$U[200000, 250000]$	7	200	5	15	$U[200000, 250000]$
	100	5	15	$U[300000, 350000]$		200	5	15	$U[300000, 350000]$
	100	10	15	$U[100000, 150000]$		200	10	15	$U[100000, 150000]$
4	100	10	15	$U[200000, 250000]$	8	200	10	15	$U[200000, 250000]$
	100	10	15	$U[300000, 350000]$		200	10	15	$U[300000, 350000]$

We solved the master problem using CPLEX with early stopping at 4% optimality gap. This optimality gap was reduced every 10 iterations to 3, 2, 1.5, and 1.0%. To determine this sequence of values for early stopping, we solved two instances for each class and for each total demand structure. We defined three stages for the optimality gap of the master problem: initial, intermediate, and final. For the initial stage, we considered 4 and 8% optimality gap; for the intermediate stage, we considered three percentage values: 3, 2, and 1.5% for initial stage gap of 4%, and 6, 3, and 1.5% for initial stage gap of 8%. For the final stage, we considered a percentage gap of 1.0%. For the number of iterations, we considered 5, 10, and 15 iterations. We selected the combination that gave the lowest average optimality gap and lowest average solution time over all the classes.

At each iteration of the decomposition algorithm, we solved the master problem using CPLEX, with early stopping as described above. We took the lower bound value from the solution given by CPLEX and set it as the trial lower bound. To compute the upper bound, the set of open facilities obtained from the master problem was given as an input to CPLEX to solve $|T|$ dual problems to optimality. The value of the trial upper bound at iteration k was computed by $Z_{ub}^{k} = DSP_y$ plus the associated fixed costs. The primal subproblem and the strong-cuts algorithm were also solved using CPLEX. The stopping criteria for the Benders' decomposition algorithms was set to 1.5% optimality gap and $M = 200$ iterations.

The benchmark solutions were obtained solving the DDCFLP model with CPLEX, which uses a branch and cut algorithm, using default settings. We used early stopping with an optimality gap of 1.5%, and recorded the lower and upper bound values.

For all the experiments, we limited the running time for each instance with $n = 50$ to 3000 seconds, $n = 100$ to 4000 seconds, $n = 150$ to 5000 seconds, and $n = 200$ to 6000 seconds. For each class, we reported the average and maximum value of the optimality gap and the average and maximum solution time. Tables 9, 10, and 11 on pages 95, 96, and 97 respectively, describe the performance of the solution algorithms for each total demand structure. We denote by NS the benchmark classes Not Solved by CPLEX within the maximum solution time.

In the analysis, we considered the cost split, in percentage value, of the total cost corresponding to variable transportation cost and fixed opening, operation, and closing costs. For each class, we selected the solution with minimum average optimality gap to compute the average percentage for each type of cost. Tables 12, 13, and 14 on pages 98, 99, and 100 respectively, report the average cost split per class for each total demand structure, and the average number of open facilities. The analysis of the cost split is important to identify a possible relationship between the efficiency of the solution algorithm and the structure of the cost split for each class of problems.

From Tables 9, 10, and 11 on pages 95, 96, and 97 respectively, we see that Benders' decomposition is shown to be more efficient in solving the DDCFLP than the branch and cut procedure used by CPLEX. For the three demand structures, the decomposition algorithm required less solution time on average.

For classes with increasing and steady demand, we observe on pages 95 and 97 in Tables 9 and 11 respectively, that increasing the value of the fixed operation cost decreases the average solution time of Benders' decomposition algorithm. The reason for this can be explained by looking into the structure of the master problem. The three main components in the objective function of the master problem are the fixed operation cost, fixed opening cost, and the auxiliary variable associated with the dual subproblem that considers the total transportation cost. For increasing demand, a larger number of facilities is expected to be established in the first period and remain operational along the time horizon. Since the largest demand is expected to be in the last period, the set of open facilities carries over some extra capacity that is eventually used as demand increases. Thus, the main contribution to the objective function value comes from the facility fixed costs. Since the upper bound value incorporates these costs, plus the additional variable transportation costs which gradually become larger, the optimality gap between the relaxed master problem and primal subproblem diminishes quickly. From Table 12 on page 98 we observe that for increasing demand the average solution time of Benders' decomposition is smaller for classes with an average cost split of $80 - 85\%$ of fixed operation cost. The average number of facilities for each class is near the expected number of open facilities, pm. Note that, for increasing demand, as we increase the fixed operation cost the average number of open facilities remains the same. Also, the average number of facilities tends to be located in regions B and C, since demand is shifting towards these regions.

For classes with steady demand, the fixed costs represent the main portion of the objective function value, as we observe from Table 11 on page 97. In these classes, the solution to the master problem takes more time than classes with increasing demand. Since demand is fluctuating, the set of open facilities tends to be determined by the locations where the average total demand concentrates over the time horizon. We see from Table 14 on page 100, that the average number of open facilities is larger in regions B and C.

For the classes with decreasing demand, we observe from Table 10 on page 96 that in general the average solution time is smaller than the average time taken to solve problems with increasing and steady demand. Furthermore, we observe in this case that increasing the fixed operation cost does not have a significant impact on the average solution time. However, increasing the problem size, specially the number of periods, seems to increase the average solution time. Observe that, the main contribution to the objective function value comes from the fixed costs since the transportation cost is decreasing, thus the optimality gap is closed faster. From Table 13 on page 99, we see that Benders' decomposition takes less average solution time for classes with an average cost split with $80 - 90\%$ of fixed operation cost. Note that in this case, since demand is decreasing, the percentage in the cost split for fixed operation cost is higher than the split for increasing demand. Further, notice that the average number of open facilities is larger in regions A and B. Since demand in decreasing, the open facilities will have to serve regions with the higher levels of

Table 9 DDCFLP Computational Results Increasing Demand

	Branch & Cut				Benders' Decomposition			
	Gap (%)		Time (sec.)		Gap (%)		Time (sec.)	
Cluster	Avg.	Max.	Avg.	Max.	Avg.	Max.	Avg.	Max.
1	2.10	5.74	983.04	3000.28	1.34	1.48	6.00	12.41
	2.92	8.59	912.23	3000.58	1.22	1.39	5.83	23.45
	3.25	9.82	912.04	3000.43	1.19	1.49	6.74	22.97
2	1.53	6.39	934.08	3000.22	1.22	1.48	19.47	30.44
	0.56	1.25	131.91	760.16	1.22	1.45	6.87	30.95
	0.69	1.50	159.41	923.63	1.29	1.47	4.40	23.94
3	0.59	1.02	587.66	3550.08	1.46	1.49	19.00	53.33
	1.00	1.49	328.13	423.27	1.25	1.49	25.10	68.24
	1.13	1.50	715.77	1761.55	1.15	1.49	20.39	72.83
4	1.88	3.93	2314.47	4000.58	1.45	1.49	160.99	302.19
	1.14	5.10	1671.21	4000.44	1.36	1.49	43.50	137.79
	1.39	5.46	1921.63	4000.33	1.41	1.49	21.26	43.06
5	0.88	3.24	2041.50	5000.34	1.36	1.47	82.21	199.58
	0.67	1.33	1599.20	2219.89	1.43	1.50	23.66	35.41
	0.93	1.43	2607.74	3849.98	1.24	1.49	24.12	68.47
6	NS	NS	NS	NS	1.43	1.50	786.02	3280.98
	NS	NS	NS	NS	1.41	1.50	153.26	508.01
	NS	NS	NS	NS	1.22	1.48	113.73	301.22
7	0.60	1.90	3515.96	6000.52	1.36	1.47	490.50	3701.36
	NS	NS	NS	NS	1.28	1.50	136.21	364.38
	NS	NS	NS	NS	1.28	1.50	121.73	384.14
8	NS	NS	NS	NS	1.42	1.49	736.69	1350.70
	NS	NS	NS	NS	1.35	1.50	262.68	605.00
	NS	NS	NS	NS	1.09	1.48	591.32	1497.94

demand.

Table 10 DDCFLP Computational Results Decreasing Demand

	Branch & Cut				Benders' Decomposition			
	Gap (%)		Time (sec.)		Gap (%)		Time (sec.)	
Cluster	Avg.	Max.	Avg.	Max.	Avg.	Max.	Avg.	Max.
1	2.56	7.36	924.47	3000.36	1.30	1.44	2.73	7.86
	3.17	9.72	906.98	3000.34	0.99	1.38	3.69	13.24
	3.57	10.94	906.35	3000.27	0.94	1.49	5.97	25.84
2	1.26	1.38	214.19	3000.33	1.28	1.46	4.65	16.56
	0.74	1.38	66.21	83.69	1.23	1.49	1.71	3.69
	0.81	1.49	73.71	123.02	1.21	1.48	4.68	32.78
3	1.49	1.87	520.62	4000.80	1.16	1.49	11.70	55.73
	1.22	1.50	1338.19	3766.56	1.06	1.48	21.01	58.77
	1.36	1.79	1900.72	4000.47	1.00	1.46	24.31	70.39
4	1.86	5.18	1663.55	4000.53	1.28	1.50	22.76	74.33
	1.66	6.24	2440.01	4000.47	1.26	1.48	17.39	104.91
	1.09	1.91	2477.76	4000.56	1.26	1.50	19.22	131.10
5	1.31	3.93	3110.19	5000.56	1.29	1.48	21.09	34.09
	0.80	1.50	2028.26	3190.67	1.24	1.48	9.54	28.93
	1.04	1.68	2665.06	5000.50	1.21	1.49	10.00	22.92
6	NS	NS	NS	NS	1.35	1.49	91.79	443.52
	NS	NS	NS	NS	1.25	1.50	49.28	95.64
	NS	NS	NS	NS	1.09	1.45	90.99	351.17
7	NS	NS	NS	NS	1.32	1.46	81.67	316.72
	NS	NS	NS	NS	1.28	1.49	74.73	412.61
	NS	NS	NS	NS	1.27	1.50	67.66	359.41
8	NS	NS	NS	NS	1.40	1.50	171.02	396.34
	NS	NS	NS	NS	1.31	1.49	212.49	721.72
	NS	NS	NS	NS	1.05	1.49	315.17	974.33

Table 11 DDCFLP Computational Results Steady Demand

	Branch & Cut				Benders' Decomposition			
	Gap (%)		Time (sec.)		Gap (%)		Time (sec.)	
Cluster	Avg.	Max.	Avg.	Max.	Avg.	Max.	Avg.	Max.
1	1.99	4.50	955.71	3000.14	1.29	1.49	12.28	15.22
	2.74	7.67	914.00	3000.31	1.38	1.49	19.12	37.09
	3.12	9.57	913.11	3000.35	1.36	1.50	8.41	18.41
2	1.83	6.25	1138.73	3000.13	1.33	1.49	17.87	24.92
	0.81	1.98	503.15	3000.16	1.25	1.49	14.35	42.92
	0.58	1.38	181.62	1105.06	1.30	1.46	2.27	5.25
3	1.64	3.25	1419.20	4000.31	1.47	1.91	469.16	4000.83
	1.11	4.73	696.28	4000.35	1.38	1.50	26.46	68.78
	1.08	1.45	740.34	3117.80	1.30	1.45	21.41	59.81
4	2.01	4.12	2779.75	4000.63	1.37	1.49	798.10	4000.23
	1.46	4.74	1754.26	4000.53	1.46	1.53	900.75	4000.72
	1.25	5.01	1994.48	4000.46	1.32	1.49	64.59	227.98
5	1.41	3.22	2383.63	5000.56	1.42	1.47	305.05	514.04
	0.67	1.06	1724.57	2204.06	1.38	1.49	53.27	101.80
	0.85	1.24	1945.48	3441.57	1.38	1.45	35.81	59.56
6	NS	NS	NS	NS	1.55	2.34	2287.12	5000.55
	NS	NS	NS	NS	1.44	1.50	237.82	459.53
	NS	NS	NS	NS	1.25	1.44	169.77	485.59
7	1.62	1.39	3299.55	4811.56	1.48	1.65	1117.99	6000.26
	1.10	2.27	4634.15	6000.84	1.41	1.60	728.27	6000.14
	1.58	2.36	4017.03	6000.82	1.30	1.49	177.18	423.79
8	NS	NS	NS	NS	1.45	1.49	2875.95	5617.58
	NS	NS	NS	NS	1.34	1.46	571.57	739.77
	NS	NS	NS	NS	1.34	1.49	391.91	1235.16

Table 12 Average Cost Split and Open Facilities for Increasing Demand

	Cost Split (%)			Open Facilities			
Cluster	Transportation	Operation	Opening	Avg.	A	B	C
1	16.36	71.25	12.39	6.3	1.9	2.2	2.2
	9.49	77.67	12.84	6.3	1.9	2.2	2.2
	6.85	80.05	13.10	6.3	1.9	2.1	2.3
2	22.24	71.58	6.18	6.0	2.0	1.7	2.3
	12.89	80.47	6.64	6.0	2.1	1.6	2.3
	9.37	83.77	6.86	6.0	2.0	1.8	2.2
3	12.45	74.62	12.93	13.0	4.3	4.2	4.5
	7.34	79.49	13.17	13.0	4.3	4.1	4.6
	5.21	81.52	13.27	13.0	4.0	4.5	4.5
4	15.80	77.48	6.72	13.0	4.3	4.2	4.5
	9.48	83.66	6.87	13.0	3.9	4.2	4.9
	6.75	86.22	7.03	13.0	3.6	4.5	4.9
5	11.22	75.62	13.16	19.0	6.1	6.1	6.8
	6.40	80.30	13.30	19	6.1	6.3	6.6
	4.50	82.12	13.38	19.0	5.8	6.5	6.7
6	14.01	79.17	6.81	19.0	5.2	6.2	7.6
	8.34	84.66	7.00	19.0	5.2	6.6	7.2
	5.78	87.08	7.14	19.0	5.3	6.2	7.5
7	9.84	76.77	13.39	26.1	5.2	6.2	7.6
	5.70	80.93	13.36	26.0	8.7	8.4	8.9
	4.00	82.54	13.46	26.0	8.2	8.9	8.9
8	12.40	80.62	6.98	26.0	7.1	8.5	10.4
	7.38	85.52	7.10	26.0	6.7	9.1	10.2
	5.27	87.60	7.13	26.0	7.0	8.7	10.3

Table 13 Average Cost Split and Open Facilities for Decreasing Demand

	Cost Split (%)			Open Facilities			
Cluster	Transportation	Operation	Opening	Avg.	A	B	C
1	9.06	77.44	13.50	6.3	2.3	2.2	1.8
	5.56	80.96	13.48	6.3	2.1	2.6	1.6
	3.87	82.68	13.46	6.3	2.2	2.5	1.6
2	11.14	81.80	7.06	6.0	2.3	2.1	1.6
	6.25	86.61	7.14	6.0	2.4	1.9	1.7
	4.35	88.43	7.23	6.0	2.3	2.0	1.7
3	6.57	79.63	13.81	13.0	5.5	3.8	3.7
	3.61	82.67	13.72	13.0	4.9	4.9	3.2
	2.65	83.75	13.60	13.0	5.2	4.5	3.3
4	7.46	85.14	7.40	13.0	5.9	3.9	3.2
	4.14	88.57	7.29	13.0	5.6	4.3	3.1
	2.97	89.75	7.29	13.0	6.0	4.1	2.9
5	5.76	80.25	13.99	19.0	7.7	6.2	5.1
	3.28	82.97	13.76	19.0	7.4	6.2	5.4
	2.23	84.09	13.68	19.0	6.9	6.4	5.7
6	6.51	86.08	7.41	19.0	7.6	6.4	5.0
	3.76	88.89	7.35	19.0	7.5	6.6	4.9
	2.66	89.97	7.37	19.0	7.9	6.1	5.0
7	4.97	80.94	14.09	26.0	10.7	9.3	6.0
	2.84	83.37	13.79	26.0	10.2	9.1	6.7
	2.04	84.25	13.71	26.0	9.9	8.8	7.3
8	5.65	86.83	7.52	26.0	11.1	8.6	6.3
	3.24	89.36	7.40	26.0	10.6	9.0	6.4
	2.37	90.28	7.35	26.0	10.2	9.4	6.4

Table 14 Average Cost Split and Open Facilities for Steady Demand

	Cost Split (%)			Open Facilities			
Cluster	Transportation	Operation	Opening	Avg.	A	B	C
1	19.90	68.24	11.86	6.3	1.3	2.6	2.4
	12.07	75.37	12.56	6.3	1.5	2.5	2.3
	8.89	78.30	12.81	6.3	1.0	3.0	2.3
2	22.65	71.22	6.13	6.0	2.0	1.7	2.3
	13.42	79.99	6.58	6.0	2.0	1.7	2.3
	9.63	83.53	6.84	6.0	1.9	1.9	2.2
3	15.02	72.47	12.51	13.0	2.8	4.6	5.6
	8.93	78.09	12.98	13.0	3.1	4.3	5.6
	6.63	80.27	13.10	13.0	3.0	4.5	5.5
4	15.91	77.38	6.72	13.1	2.9	5.0	5.2
	9.89	83.26	6.85	13.0	3.1	4.6	5.3
	7.10	85.92	6.98	13.0	3.2	4.2	5.6
5	12.96	74.19	12.85	19.0	5.2	6.2	7.6
	7.59	79.32	13.09	19.0	4.8	6.4	7.8
	5.48	81.29	13.23	19.0	4.6	6.7	7.7
6	13.49	79.65	6.86	19.0	4.7	6.8	7.5
	7.81	85.15	7.04	19.0	5.2	6.3	7.5
	5.60	87.24	7.16	19.0	5.0	6.3	7.7
7	11.06	75.74	13.19	26.1	6.7	9.4	10.0
	6.68	80.08	13.24	26.0	6.9	9.3	9.8
	4.78	81.91	13.32	26.0	6.6	9.5	9.9
8	12.23	80.79	6.98	26.0	5.9	9.2	10.9
	7.19	85.72	7.08	26.0	6.1	8.8	11.1
	5.24	87.62	7.14	26.0	6.1	8.6	11.3

V.5. DDCFLP as a Special Case of DCFLP

In this section we present a computational analysis to show that in the presence of large relocation costs the DCFLP model provides a solution to the DDCFLP. To begin, consider the objective function of the DCFLP, and let Z denote the objective function value:

$$Z = \min \sum_{i\in I}\sum_{j\in J}\sum_{t\in T} c_{ijt}x_{ijt} + \sum_{j\in J}\sum_{t\in T} f_{jt}y_{jt} + \sum_{j\in J}\sum_{t\in T}(a_{jt}u_{jt} + b_{jt}v_{jt}) \quad (5.25)$$

If we let the fixed opening cost, a_{jt}, and fixed closing cost, b_{jt}, take a very large value, say $a_{jt} = b_{jt} = \infty, j \in J, t \in T$, then it is never beneficial (in terms of minimizing the total cost) to open a new facility or close an existing facility. Note that in any feasible solution, the fixed opening cost is incurred at least in the first period regardless of its value.

Let $O = \{j \in J : y_{j1} = 1\}$ denote the set of open facilities in the first period and let $y_{j0} = 0, j \in J$. With these large fixed costs, a feasible solution to the DCFLP will have $y_{jt} = y_{j1} = 1, j \in O, t \geq 2$, $y_{jt} = 0, j \notin O, t \in T$, $u_{j1} = y_{j1} = 1, j \in O$, and $u_{jt} = v_{jt} = 0, j \in J, t \geq 2$.

Since the fixed opening cost is incurred in the first time period, we can set $a_{jt} = \infty, j \in J, t \geq 2$, and is never beneficial to open a new facility in any period $t \geq 2$. In this case, the objective function value of the DCFLP is as follows:

$$Z = \min \sum_{i\in I}\sum_{j\in J}\sum_{t\in T} c_{ijt}x_{ijt} + \sum_{j\in J}\sum_{t\in T} f_{jt}y_{j1} + \sum_{j\in J} a_{j1}y_{j1} \quad (5.26)$$

Note that the objective function value of the DCFLP is equivalent to the objective function value of the DDCFLP:

$$Z = \min \sum_{i\in I}\sum_{j\in J}\sum_{t\in T} c_{ijt}x_{ijt} + \sum_{j\in J}\sum_{t\in T} f_{jt}y_{j} + \sum_{j\in J} a_{j}y_{j} \quad (5.27)$$

Clearly, with this setting for relocation costs a feasible solution to the DCFLP is also feasible for the DDCFLP. Thus, we should question the need for a mixed integer programming model to solve the DDCFLP if we can use the DCFLP model, with special input data modifications, instead. In other words, is the DDCFLP model necessary? The answer is, yes.

It is possible to develop different formulations for an optimization problem such as the DDCFLP. However, a particular formulation will be preferred among the others. This ideal formulation will be the one that provides the tightest formulation since this can affect the performance of the solution method. A *weak* formulation will have a very large feasible region, making it too time consuming to explore. On the other hand, a *strong* formulation will have a very tight feasible region, taking less computational effort to be explored.

We can think of the DCFLP as the weak formulation since it has a larger number of variables and constraints, most of them taking zero value, and the DDCFLP model as the strong formulation. Thus, we would expect that the performance of any solution algorithm will be more efficient in solving the DDCFLP model than solving the DCFLP with special input data.

To support this claim, we conducted a numerical experiment consisting of 18 classes of problems using three total demand structures described in Chapter III, increasing, decreasing, and steady; three values for $n = 50$, 100, and 150 locations; two values for $\tau = 5$ and 10 periods; $p = 0.15$; and a single discrete uniform distribution, $U[300000, 350000]$, to generate the fixed operation cost. For the DCFLP, we set $a_{jk} = b_{jk} = 700 \times 10^6$, $j \in J, t \in T, k \geq 2$. For the DDCFLP, fixed opening cost considered only the costs for the first period, i.e., $a_j = a_{j1}, j \in J$. Table 15 shows the arrangement of classes into six clusters.

Table 15 Classes of Problems Arranged in Clusters

	Parameters					*Parameters*			
Cluster	n	τ	p %	f	Cluster	n	τ	p %	f
1	50	5	15	$U[300000, 350000]$	4	50	10	15	$U[300000, 350000]$
2	100	5	15	$U[300000, 350000]$	5	100	10	15	$U[300000, 350000]$
3	150	5	15	$U[300000, 350000]$	6	150	10	15	$U[300000, 350000]$

We used the corresponding implementation of Benders' decomposition to solve each model. The stopping criteria for both implementations considered an optimality gap of 1.0%. We also limited the running time for each class to 3000, 4000, and 5000 seconds for $n = 50$, 100, and 150 locations, respectively. For each class, we solved 10 instances and reported the average and maximum optimality gap and solution time. Tables 16, 17, and 18 describe the performance of the Benders' decomposition algorithm for each model.

From the results in Table 16, for increasing demand, the average solution time of the DCFLP was shorter for the first three clusters, which are the smaller problems. For classes with $n = 150$ locations, the average solution time and average gap for the DDCFLP were smaller. From Table 17, for decreasing demand, the solution algorithm performed better for the DDCFLP in all but the fourth cluster. Finally, in Table 18, for steady demand, we observe that Benders' decomposition performed better for the DDCFLP model with the exception of the third and fourth cluster.

In general, we can say that the mixed integer programming formulation for the DDCFLP is efficient and necessary. The DCFLP model with special data modifications in fact provides a solution to the DDCFLP, thus can be considered as an alternative tool to solve this problem. Although, the DCFLP did not show to be the most efficient model for all classes of problems, it may be more efficient for small problems or for instances with special data structure.

Table 16 Computational Results Increasing Demand

	DCFLP				DDCFLP			
	Gap (%)		Time (sec.)		Gap (%)		Time (sec.)	
Cluster	Avg.	Max.	Avg.	Max.	Avg.	Max.	Avg.	Max.
1	0.85	0.99	12.36	35.98	0.86	0.98	14.43	49.47
2	0.85	1.00	32.59	104.30	0.82	1.00	33.40	98.08
3	0.91	0.98	45.27	186.11	0.68	0.91	90.66	190.13
4	0.87	0.99	297.51	650.80	0.86	1.00	179.25	441.03
5	0.88	0.94	148.12	447.58	0.84	1.00	78.38	234.83
6	0.92	1.00	538.42	1580.08	0.85	0.99	140.75	497.31

Table 17 Computational Results Decreasing Demand

	DCFLP				DDCFLP			
	Gap (%)		Time (sec.)		Gap (%)		Time (sec.)	
Cluster	Avg.	Max.	Avg.	Max.	Avg.	Max.	Avg.	Max.
1	0.75	0.99	9.01	29.88	0.73	0.99	5.00	16.52
2	0.77	0.98	47.07	134.03	0.81	0.99	26.26	103.24
3	0.84	0.97	97.77	208.77	0.46	0.99	95.55	194.92
4	0.85	0.99	190.37	472.81	0.64	0.99	191.05	462.73
5	0.81	1.00	203.63	757.86	0.68	0.98	53.85	185.52
6	0.89	1.00	383.89	1605.89	0.73	0.99	99.56	347.32

V.6. Summary and Conclusions

In this chapter we introduced the DDCFLP and presented a mixed integer programming formulation. We developed a Benders' decomposition algorithm to solve the model, which showed to be more efficient compared with branch and cut approach in solving classes of problems for each demand structure. We observed that the efficiency of the decomposition procedure is related to the problem structure and input parameters. We also presented a comparison between the DCFLP and DDCFLP models, showing that the later model can be used to solve the former by modifying

Table 18 Computational Results Steady Demand

	DCFLP				DDCFLP			
	Gap (%)		Time (sec.)		Gap (%)		Time (sec.)	
Cluster	Avg.	Max.	Avg.	Max.	Avg.	Max.	Avg.	Max.
1	0.88	1.00	34.29	60.25	0.89	1.00	21.88	38.61
2	0.91	1.00	55.40	133.11	0.88	1.00	34.10	68.47
3	0.94	1.00	78.26	132.30	0.94	0.99	123.81	208.24
4	0.89	1.00	414.50	1352.55	0.91	0.98	510.83	706.69
5	0.94	1.00	219.68	413.38	0.91	0.99	84.55	197.97
6	0.91	0.99	538.45	1047.75	0.93	0.99	248.98	665.84

the input data. In general, the implementation of Benders' decomposition showed to be more efficient in solving the DDCFLP. However, the DCFLP showed to be efficient in solving small instances for a particular data set. Thus, it can be considered as an alternative approach to solve this problem.

CHAPTER VI

ROBUST CAPACITATED FIXED CHARGE LOCATION PROBLEM (RCFLP)

In this chapter, we investigate the problem of finding the locations of facilities with limited capacity to satisfy the demand of a set of customers over a discrete and finite time horizon when relocation of facilities is not allowed. The demand of each customer is assumed to be changing by time in a known way, and can be split or served by one or more facilities. There are fixed costs for operating the facilities in each period and a variable transportation cost for serving the demand of customers. The objective is to minimize the worst-case cost or regret. The regret is the difference between the total cost incurred by the robust configuration of facilities, chosen at the beginning of the time horizon, and the total cost incurred in each period by the optimal configuration of facilities obtained by solving the associated CFLP in each time period.

The chapter is organized as follows. In Section VI.1, we give the problem statement. In Section VI.2, we present the mixed integer programming formulation and notation for the RCFLP. In Section VI.3, we implement two metaheuristics to solve the RCFLP. In Section VI.4, we present numerical results using different demand patterns to test the performance of the heuristics. Finally in Section VI.5, we summarize the results and give concluding remarks.

VI.1. Problem Statement

The RCFLP statement is as follows. Consider a geographical region where a given group of customers are dispersed. Each customer has a given demand for a certain product. Along a discrete and finite time horizon, the total demand of the customers is time varying in a known way.

The establishment of facilities is required to supply the customers' demand over the entire time horizon. Each facility has a finite capacity in the amount of demand that can be supplied to the customers. Each customer can be supplied by one or more facilities. The shipments of demand between facilities and customers incur a variable transportation cost proportional to quantity and distance. A fixed operation cost is incurred in each period the facility remains operational, this can be thought as the per period cost associated with the initial investment or the total expenses for services and labor. Establishment of facilities takes place at the beginning of the time horizon. Facilities can not be closed or relocated in subsequent periods.

In selecting the locations for facilities, the decision maker may consider the best approach possible for this problem. In the absence of relocation costs, the best approach would be to determine the optimal location of facilities for each period by solving the associated CFLP. If this configuration of facilities happens to be the same for each period then an optimal solution would be at hand. Otherwise, this solution will imply that at some period of time the facilities will have to be relocated, violating the assumption that relocation is not allowed.

The decision maker may want to consider a solution with minimum deviation from the best possible location plan for each period. Since the decision has to be made at the beginning of the time horizon and no changes in the location of facilities can me made afterwards, a possible approach is to consider the worst-case scenario, i.e., the maximum difference in cost that would have to be paid if a particular choice of a fixed configuration of facilities is made at the beginning of the first period instead of the optimal configuration for each period. We would like this worst-case cost or deviation to be as minimal as possible, thus in a sense the best fixed configuration of facilities will be *robust* such that the worst-case cost will be minimal regardless of future changes in demand and cost parameters.

In robust optimization problems, the value of parameters is uncertain and no probability information is available about the possible states of nature or scenarios. These types of problems consider a solution to be robust if it has the overall best performance across all possible scenarios, thus not necessarily optimal for each scenario. Usual measures of robustness consider the worst-case scenario, such as the minimization of the maximum regret or opportunity loss. This robustness measure minimizes the difference or deviation between a solution taken for a given scenario and the optimal solution for that scenario. The regret is the cost for having to make a decision before knowing which state of nature will happen to pass.

This approach is applicable to the incumbent problem of finding a fixed configuration of facilities with minimum deviation from the optimal solution for each time period. The main decisions are determining the number of facilities required to supply the demand, selecting the locations to establish the facilities at the beginning of the time horizon, and allocating demand to facilities without exceeding the capacity of the facilities. The main objective is to minimize the maximum regret or difference in total cost between the robust configuration of facilities and the optimal configuration for each time period.

VI.2. Model and Notation

In this section, we provide a mixed integer programming formulation of the RCFLP. We use the following notation.

Parameters

I set of demand locations, $i = 1, \ldots, n$

J set of facility locations, $j = 1, \ldots, m$

T set of periods, $t = 1, \ldots, \tau$

f_{jt} fixed cost for having a facility open (operating) in location j during period t

w_{it} amount of demand in location i during period t

q_j capacity available if a facility is open at location j

d_{ij} distance between facility at location j to customer i

α per unit distance per unit demand cost

c_{ijt} transportation cost for shipping demand of location i from facility at location j in period t, $c_{ijt} = \alpha w_{it} d_{ij}$

Z_t^* optimal objective function value in period t

Decision Variables

x_{ijt} fraction of demand of location i shipped from facility at location j in period t

y_j 1 if a facility is open in location j, 0 otherwise

$$\text{(RCFLP)} \quad \min \quad \max_{t \in T} \left\{ \sum_{i \in I} \sum_{j \in J} c_{ijt} x_{ijt} + \sum_{j \in J} f_{jt} y_j - Z_t^* \right\} \tag{6.1}$$

subject to

$$\sum_{j \in J} x_{ijt} = 1 \qquad i \in I,\, t \in T \tag{6.2}$$

$$x_{ijt} \le y_j \qquad i \in I,\, j \in J,\, t \in T \tag{6.3}$$

$$\sum_{i \in I} w_{it} x_{ijt} \le q_j y_j \qquad j \in J,\, t \in T \tag{6.4}$$

$$x_{ijt} \ge 0, y_j \in \{0,1\} \qquad i \in I,\, j \in J,\, t \in T \tag{6.5}$$

The objective function (6.1) minimizes the maximum deviation or regret in total cost between the robust configuration of facilities and the optimal configuration for each period. The value of Z_t^* corresponds to the optimal objective function value of the CFLP in period $t \in T$, which is given as an input to the model. The total cost considers the variable transportation cost for shipping demand from facilities to customers, and the fixed operation cost for open facilities. Constraints (6.2) are the demand constraints (for each customer, all the demand must be met), (6.3) ensure that demand is allocated to open facilities, (6.4) are the capacity constraints (no facility can supply more than its capacity), and (6.5) are the nonnegativity and integrality constraints.

Alternatively, we can use the following formulation:

$$\text{(RCFLP)} \quad \min \rho \tag{6.6}$$

subject to

$$\rho \geq \sum_{i \in I} \sum_{j \in J} c_{ijt} x_{ijt} + \sum_{j \in J} f_{jt} y_j - Z_t^* \qquad t \in T \tag{6.7}$$

$$\sum_{j \in J} x_{ijt} = 1 \qquad i \in I,\, t \in T \tag{6.8}$$

$$x_{ijt} \leq y_j \qquad i \in I,\, j \in J,\, t \in T \tag{6.9}$$

$$\sum_{i \in I} w_{it} x_{ijt} \leq q_j y_j \qquad j \in J,\, t \in T \tag{6.10}$$

$$\rho \geq 0, x_{ijt} \geq 0, y_j \in \{0, 1\} \qquad i \in I,\, j \in J,\, t \in T \tag{6.11}$$

which replaces the inner maximization part of the objective function by constraint set (6.7) using a continuous variable ρ. Observe that minimizing this variable is equivalent to having a minimax objective function. We can restrict $\rho \geq 0$ since the right hand side of constraint set (6.7) is non-negative.

VI.3. Solution Procedure

Solving the RCFLP model using conventional mixed integer programming methods is a difficult task. The structure of the problem involves the solution of two embedded optimization problems. The solution algorithms developed in the literature for problems with minimax or minimax regret objective functions consider special cases where the number of facilities to be established is small and given. For general problems, only heuristic solution algorithms seem to be computationally feasible.

Initially, we considered the implementation of Lagrangian relaxation to solve the RCFLP. The relaxation of constraint set (6.7) leads to an unbounded Lagrangian subproblem; the Lagrangian relaxation of constraints (6.10) provides better bounds, although the improvement in the lower bound value is considerably slow. This particular type of relaxation requires a branch and bound procedure to close the optimality gap once the subgradient algorithm stops.

In solving the RCFLP, we implement two heuristics, Local Search (LS) and Simulated Annealing (SA). The main difference between these two heuristics is that LS selects the best feasible solution from the neighborhood of the current incumbent solution and stops whenever the best feasible solution fails to improve, having the limitation of reaching a local optima. On the other hand, SA randomly selects a solution from the neighborhood of the current incumbent solution and can accept non-improving solutions with a certain probability. Thus, SA provides a mechanism to leave a local optimum by exploring different regions of the solution space.

For an introduction to SA, the interested reader is referred to Kirkpatrick et al. (1983), Mavridou and Pardalos (1997), and van Laarhoven and Aarts (1987). Applications of SA to solve facility location problems can be found in Kincaid (1992) for a comparison of SA with Tabu Search in locating noxious facilities, Drezner et al.

(2002) for an implementation of SA to solve a p-median model with competitive locations, Chardaire et al. (1996) for an implementation of SA to solve the dynamic uncapacitated location problem, and Arostegui et al. (2006) for an empirical analysis of Tabu Search, SA, and Genetic Algorithm to solve several types of facility location problems.

VI.3.1. Initial Feasible Solution

Consider the Lagrangian relaxation of constraints (6.10) using non-negative Lagrange multipliers λ_{jt}. We obtain the following Lagrangian subproblem:

$$LR(\lambda) = \min \rho + \sum_{i\in I}\sum_{j\in J}\sum_{t\in T} w_{it}\lambda_{jt}x_{ijt} - \sum_{j\in J}\sum_{t\in T} q_j\lambda_{jt}y_j \tag{6.12}$$

subject to

$$\rho \geq \sum_{i\in I}\sum_{j\in J} c_{ijt}x_{ijt} + \sum_{j\in J} f_{jt}y_j - Z_t^* \qquad t \in T \tag{6.13}$$

$$\sum_{j\in J} x_{ijt} = 1 \qquad i \in I,\, t \in T \tag{6.14}$$

$$\sum_{j\in J} x_{ijt} \leq m y_j \qquad i \in I,\, t \in T \tag{6.15}$$

$$\sum_{i\in I} w_{it} \leq \sum_{j\in J} q_j y_j \qquad j \in J,\, t \in T \tag{6.16}$$

$$\rho \geq 0, x_{ijt} \geq 0, y_j \in \{0,1\} \qquad i \in I,\, j \in J,\, t \in T \tag{6.17}$$

We have replaced constraint set (6.9) with constraint set (6.15), which is obtained by summing over $j \in J$. This set has a reduced number of constraints. Also, we have included the surrogate constraints (6.16) to obtain a set of open facilities with enough capacity to serve the demand in each period.

We set the value of the Lagrange multipliers $\lambda_{jt} = 0$ and solve the Lagrangian subproblem. For given values of the location variables, $\hat{y}_j$, we can obtain a feasible solution to the original problem by solving the following linear program:

$$(TP_y) \quad \min \rho \tag{6.18}$$

subject to

$$\rho \geq \sum_{i \in I} \sum_{j \in J} c_{ijt} x_{ijt} + \sum_{j \in J} f_{jt} \hat{y}_j - Z_t^* \qquad t \in T \tag{6.19}$$

$$\sum_{j \in J} x_{ijt} = 1 \qquad i \in I,\, t \in T \tag{6.20}$$

$$\sum_{i \in I} w_{it} x_{ijt} \leq q_j \hat{y}_j \qquad j \in J,\, t \in T \tag{6.21}$$

$$\rho \geq 0, x_{ijt} \geq 0 \qquad i \in I,\, j \in J,\, t \in T \tag{6.22}$$

This initial feasible solution is given as an input to the LS and SA algorithms to improve its objective function value.

VI.3.2. Neighborhood Function

We represent a solution by a binary vector, $\mathbf{y}$, which has a value of 1 in the jth entry if $y_j = 1, j \in J$, and a value of 0 otherwise. We define the *neighborhood* of a solution as the set of binary vectors with at least one different entry value. For each feasible solution, its neighborhood is obtained by three types of moves: *add*, *drop*, and *exchange*. The add move changes an entry of the binary vector $\mathbf{y}$ with value of 0 to 1. The drop move changes the value of an entry from 1 to 0, provided that the resulting set of open facilities after the move has a total capacity of at least the maximum total demand over all periods. This ensures that closing an open facility will lead to a feasible solution. The exchange move switches or flips the values between an entry with value of 1 and an entry with value of 0. This move is allowed as long as the resulting set of open facilities provides a feasible solution. Display 6 gives the pseudo-code for the neighborhood function. We define the following notation:

O	set of open facilities
O'	auxiliary set
I'	auxiliary set
TC	total cost
$\mathbf{z}$	binary array
q	total capacity
w	maximum total demand
$\mathbb{N}$	neighborhood

Display 6 Pseudo-code neighborhood function

```
Initialize: $\mathbf{y}$, $\mathbb{N} = \emptyset$, $O = \{j \in J : y_j = 1\}$, $q \leftarrow \sum_{j \in O} q_j$, $w \leftarrow \max_{t \in T} \left\{\sum_{i \in I} w_{it}\right\}$
for $j = 1$ to $m$ do
    if $\mathbf{y}[j] = 1$ then
        if $q - q_j \geq w$ then
            $\mathbf{z} = \mathbf{y}$
            $\mathbf{z}[j] \leftarrow 0$
            Add $\mathbf{z}$ to $\mathbb{N}$
            for $k = j + 1$ to $m$ do
                if $\mathbf{y}[k] = 0$ then
                    if $q - q_j + q_k \geq w$ then
                        $\mathbf{z} = \mathbf{y}$
                        $\mathbf{z}[j] \leftarrow 0$
                        $\mathbf{z}[k] \leftarrow 1$
                        Add $\mathbf{z}$ to $\mathbb{N}$
                    end if
                end if
            end for
        end if
    else
        $\mathbf{z} = \mathbf{y}$
        $\mathbf{z}[j] \leftarrow 1$
        Add $\mathbf{z}$ to $\mathbb{N}$
        for $k = j + 1$ to $m$ do
            if $\mathbf{y}[k] = 1$ then
                if $q + q_j - q_k \geq w$ then
                    $\mathbf{z} = \mathbf{y}$
                    $\mathbf{z}[k] \leftarrow 0$
                    $\mathbf{z}[j] \leftarrow 1$
                    Add $\mathbf{z}$ to $\mathbb{N}$
                end if
            end if
        end for
    end if
end for
Return $\mathbb{N}$
```

The binary vector $\mathbf{y}$ has $|J| = m$ entries. Let ℓ denote the number of entries of vector $\mathbf{y}$ with value of 1. The total number of add moves will be equal to $(m - \ell)$, the total number of drop moves will be ℓ, and the total number of exchange moves $\ell\,(m - \ell)$.

Thus, the size of the entire neighborhood will be $(m-\ell)+\ell+\ell\,(m-\ell)$.

The process of solving the linear program (TP_y) to evaluate the objective function value for each neighborhood solution can be very time consuming when solved to optimality. Instead, we use a heuristic algorithm to approximate the objective function value of (TP_y) for each candidate solution. For each period $t \in T$, the algorithm sorts the set I of customer locations in non-increasing order of demand w_{it}, then allocates the demand of each customer to the nearest open facility. If the capacity of the facility is depleted, then the remaining demand is allocated to the second nearest facility. The algorithm keeps track of the capacity for each open facility in each period. The algorithm stops when all demands are allocated, then adds the associated fixed operation costs for the open facilities.

Observe that this approach is a valid method to evaluate the objective function value of a candidate solution, since the value obtained by solving the linear program (TP_y) differs only by the constant term Z_t^* for each $t \in T$. Once the best feasible solution is selected, its objective function value can be computed by solving problem (TP_y). Display 7 gives the pseudo-code of the approximation algorithm.

Display 7 Pseudo-code approximation algorithm for objective function value

Initialize: $O \leftarrow \{j \in J : y_j = 1\}$, $TC \leftarrow 0$
for $t \leftarrow 1$ to τ **do**
    $I' \leftarrow I$
    for $j \in O$ **do**
        $k_j \leftarrow q_j$
    end for
    Sort I' in non-increasing order of w_{it}
    for each $i \in I'$ **do**
        $O' \leftarrow O$
        while $w_{it} > 0$ **do**
            $j^* \leftarrow \arg\min_{j \in O'} \{c_{ijt}\}$
            $O' \leftarrow O' \backslash \{j^*\}$
            if $k_{j^*} - w_{it} \geq 0$ **then**
                $k_{j^*} \leftarrow k_{j^*} - w_{it}$
                $w_{it} \leftarrow 0$
                $TC \leftarrow TC + c_{ij^*t}$
            else
                $TC \leftarrow TC + [(w_{it} - k_{j^*}) / w_{it}]\, c_{ijt}$
                $w_{it} \leftarrow w_{it} - k_{j^*}$
                $k_{j^*} \leftarrow 0$
            end if
        end while
    end for
    for each $j \in O$ **do**
        $TC \leftarrow TC + f_{jt}$
    end for
end for
Return TC

VI.3.3. Local Search

The LS algorithm takes as an input the set of open facilities obtained from the solution to the Lagrangian subproblem $LR(\lambda)$, with Lagrange multipliers $\lambda_{jt} = 0, j \in J, t \in T$. This initial solution is set as the best incumbent feasible solution. The objective function value of this incumbent solution is then computed by solving problem (TP_y), setting this value as the best objective function value. The neighborhood of the incumbent solution is generated using add, drop, and exchange moves. Each neighbor solution is evaluated using the approximation algorithm. The solution with minimum total cost over the entire neighborhood is selected. Then, problem (TP_y) is solved again to obtain the exact objective function value. If this value is less than the best objective function value, the new solution is taken as best incumbent feasible solution as well as its objective function value. The process is repeated until the first iteration when the best objective function value fails to improve. Display 8 gives the pseudo-code for the LS algorithm. We use the following notation:

S	set of open facilities
S^*	best set of open facilities
$f(TP_y(S))$	objective function value of problem TP_y given set S
Z	auxiliary variable
Z^*	best objective function value
ℓ	auxiliary variable
$\aleph\{\cdot\}$	neighborhood function
$\mathbb{N}$	neighborhood

Display 8 Pseudo-code LS heuristic

$S^* \leftarrow \{j \in J : y_j = 1\}$
Solve $TP_y(S^*)$
$Z^* \leftarrow f\,(TP_y(S^*))$
$\ell \leftarrow 1$
while $\ell > 0$ **do**
  $\mathbb{N} \leftarrow \aleph\{S^*\}$
  Use approximation algorithm to find $S \in \mathbb{N}$
  Solve $TP_y(S)$
  $Z \leftarrow f\,(TP_y(S))$
  if $Z < Z^*$ **then**
    $Z^* \leftarrow Z$
    $S^* = S$
  else
    $\ell \leftarrow 0$
  end if
end while
Return S^*, Z^*

VI.3.4. Simulated Annealing

The SA algorithm takes as an input the set of open facilities obtained from the solution to the Lagrangian subproblem $LR(\lambda)$, with Lagrange multipliers $\lambda_{jt} = 0, j \in J, t \in T$. This initial solution is set as the best and current incumbent feasible solution. The objective function value of this incumbent solution is then computed by solving problem (TP_y), setting this value as the best and current costs. The algorithm calls the *Metropolis* subroutine giving as an input the current solution, best solution, current cost, best cost, maximum number of iterations, and initial temperature.

In the Metropolis subroutine, the neighborhood of the incumbent solution is generated using add, drop, and exchange moves. A neighbor solution is randomly selected. Then, problem (TP_y) is solved again to obtain the exact objective function value of the neighbor solution. If this solution is better than the best solution, the new solution is taken as the current and best incumbent feasible solutions; the best and current costs are updated accordingly. Otherwise, the probability of acceptance is computed using the current temperature value. If the solution is accepted, the current solution and current cost are updated and the Metropolis subroutine is repeated until it reaches the maximum number of iterations.

Once the Metropolis subroutine stops, the value of the temperature is decreased and the maximum number of iterations for the Metropolis subroutine is increased. The SA algorithm is repeated for a fixed number of iterations. Display 9 gives the pseudo-code for the SA algorithm and Display 10 the pseudo-code for the Metropolis procedure. We use the following notation:

T_0	temperature
S_0	initial solution
S	current solution
S_1	new solution
S^*	best solution
N	maximum number of iterations for SA
M	number of iterations for Metropolis subroutine
$Cost(S)$	cost of solution S
Z	current cost
Z_1	new cost
Z^*	best cost
$\Delta Cost$	difference between $S_1 - S$
ζ	cooling rate
ξ	positive scalar
$\aleph(\cdot)$	neighborhood function
$RAND$	a uniform distributed random number

e mathematical constant, $e = 2.71828\ldots$

Display 9 Pseudo-code SA heuristic

Initialize: $k \leftarrow 0$, T_0, M, N, ζ, ξ
$S \leftarrow S_0$
$S^* \leftarrow S$
$Z \leftarrow Cost(S)$
$Z^* \leftarrow Cost(S^*)$
while $k < N$ **do**
    $Metropolis(S, Z, S^*, Z^*, T, M)$
    $k \leftarrow k + M$
    $T_0 \leftarrow \zeta T_0$
    $M \leftarrow \xi M$
end while
Return S^*, Z^*

Display 10 Pseudo-code Metropolis procedure

Input: S, Z, S^*, Z^*, T_0, M
while $M > 0$ **do**
    Randomly select $S_1 \in \aleph(S)$
    $Z_1 \leftarrow Cost(S_1)$
    $\Delta Cost \leftarrow (Cost(S_1) - Cost(S))$
    if $\Delta Cost < 0$ **then**
        $S \leftarrow S_1$
        if $Z_1 < Z^*$ **then**
            $Z^* \leftarrow Z_1$
        end if
    else
        if $RAND < e^{\Delta Cost / T_0}$ **then**
            $S \leftarrow S_1$
        end if
    end if
    $M \leftarrow M - 1$
end while
Return S^*, Z^*

VI.4. Numerical Results

In this section we conduct a numerical experiment to test the performance of the heuristic algorithms developed for the RCFLP.

We designed 12 classes of problems considering three total demand structures, increasing, decreasing, and steady; two values for $n = 50$ and 100 locations; two values for $\tau = 5$, and 10 periods; one value of $p = 0.15$; and one interval to randomly generate the fixed operation cost from a discrete uniform distribution $U[100000, 150000]$. The optimal objective function value Z_t^* for each period $t \in T$ was computed by solving the associated CFLP with Benders' decomposition. Table 19 shows four classes of problems considered in the experiments.

We solved the Lagrangian subproblem with CPLEX to optimality, setting the value of the Lagrange multipliers $\lambda_{jt} = 0, j \in J, t \in T$. Problem (TP_y) was solved using CPLEX. We stopped the LS algorithm at the first iteration when the value of the best feasible solution failed to improve.

For SA, we performed a test experiment to determine the values of the algorithm. We solved 5 instances per class for each demand structure. We considered three values for the initial temperature, $T_0 = 90000, 85000$, and 80000. For the cooling rate, we considered three values, $\zeta = 0.90, 0.88$, and 0.80; three values for $\xi = 1.1, 1.2$, and 1.3; three values for $N = 200, 300$, and 500; and three values for $M = 10, 15$, and 20. We selected the combination of parameter values that obtained the best minimum regret in less computational time.

We set the initial value of the temperature $T_0 = 85000$, the cooling rate $\zeta = 0.88$, scalar $\xi = 1.2$, the maximum number of iterations for SA $N = 300$, and the number of iterations for Metropolis subroutine $M = 15$. The benchmark solutions were obtained solving the RCFLP model with CPLEX, which uses a branch and cut algorithm, using default settings and reporting the upper (UB) and lower bound (LB) values within a maximum running time of 3600 seconds.

For each class, we solved 10 instances. For LS and SA, we reported the best objective function value (Obj. Value) and the solution time. We computed the percentage difference (Diff.) between CPLEX upper bound value and each heuristic best objective function value, Z^*, by $100(Z^* - UB)/UB$. Tables 20 to 31 present the computational results for each total demand structure and for each heuristic algorithm.

From Tables 20 and 21 for increasing demand, we observe that over all classes LS provided a lower objective function value in less computational time. In particular, for classes 3 and 4 (with larger problem sizes) LS obtained better results. Also, LS obtained an optimal solution for classes that were solved to optimality by CPLEX.

For decreasing demand, the results from Tables 22 and 23 show that LS obtained an optimal solution for almost all the classes or obtained a lower objective function value. In this case, the initial feasible solution provided a very good starting point for the algorithm since the solution time required for all classes was smaller compared to classes with increasing demand.

In the case of steady demand, we observe from Tables 24 and 25 that these instances are more difficult to solve. For almost all classes CPLEX reached the maximum running time with higher average optimality gaps. Although, the improvement in objective function value provided by LS was not as significant as in the case of increasing and decreasing demand, it provided a good improvement in less computational time.

In general, the performance of SA was not as good as LS, but it provided a lower objective function value for almost all the classes of problems. For increasing demand, the computational results from Tables 26 and 27 indicate that the overall performance of SA was better than CPLEX both in objective function value and solution time.

For decreasing demand, we observe from Tables 28 and 29 that SA obtained optimal solutions for the same classes that CPLEX solved up to optimality. For those instances not optimally solved, SA obtained a lower objective function value.

Table 19 RCFLP Classes of Problems

	Parameters			
Class	n	τ	p %	f
1	50	5	15	$U[100000, 150000]$
2	50	10	15	$U[100000, 150000]$
3	100	5	15	$U[100000, 150000]$
4	100	10	15	$U[100000, 150000]$

Finally, for problems with steady demand the overall performance of SA was good, especially for some classes 1 and 3. From Tables 30 and 31 we observe that SA provided improved objective function values compared to LS.

Over all the experiments, the performance of LS obtained better results than SA. Note that the candidate feasible solution in LS is selected from the entire neighborhood. If the initial solution reaches a region near the optimum, it is possible that this best neighbor solution will lead to the exploration of improving solutions. In general, the average solution time of LS is smaller since it stops whenever the best objective function value fails to improve. On the other hand, SA randomly selects a neighbor solution and accepts non-improving solutions with a certain probability. This procedure allows the algorithm to leave the local optima, which is the main limitation of LS. The average solution time for SA is larger since it runs for a fixed number of iterations. For some classes of problems, it provided improved solutions compared to LS.

Table 20 Computational Results LS Increasing Demand (Classes 1 and 2)

Class 1	CPLEX			LS	LS-CPLEX	CPLEX	LS
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	200874.00	175423.43	12.67	203430.00	1.27	3600.28	27.10
2	152561.00	152561.00	0.00	152561.00	0.00	32.64	8.26
3	141059.00	101982.10	27.70	133277.00	-5.52	3600.20	34.30
4	150659.00	150659.00	0.00	151536.00	0.58	73.83	8.91
5	173181.00	173181.00	0.00	179913.00	3.89	748.92	23.89
6	75125.00	75125.00	0.00	75125.00	0.00	46.94	9.14
7	161073.00	119591.05	25.75	150635.00	-6.48	3600.23	135.31
8	186338.00	153073.16	17.85	162386.00	-12.85	3600.24	154.59
9	85425.00	85425.00	0.00	85425.00	0.00	268.91	37.36
10	86480.00	86480.00	0.00	86992.00	0.59	204.70	46.31
		Avg.	8.40	Avg.	-1.85	1577.69	48.52

Class 2	CPLEX			LS	LS-CPLEX	CPLEX	LS
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	169845.00	169845.00	0.00	169845.00	0.00	101.50	18.12
2	253107.00	253086.64	0.01	253107.00	0.00	2786.90	285.23
3	267963.00	226789.42	15.37	266965.00	-0.37	3600.26	246.75
4	185320.00	154517.55	16.62	185320.00	0.00	3600.18	312.32
5	202101.00	165527.44	18.10	170043.00	-15.86	3600.26	268.65
6	302453.00	302446.28	0.00	323902.00	7.09	349.44	28.45
7	296851.00	257372.67	13.30	260350.00	-12.30	3600.23	291.55
8	164826.00	164826.00	0.00	164826.00	0.00	563.99	111.19
9	285119.00	285094.01	0.01	285119.00	0.00	209.13	32.80
10	179675.00	179662.48	0.01	179675.00	0.00	1639.95	290.47
		Avg.	6.34	Avg.	-2.14	2005.18	188.55

Table 21 Computational Results LS Increasing Demand (Classes 3 and 4)

Class 3	CPLEX			LS	LS-CPLEX	CPLEX	LS
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	270386.00	270374.41	0.00	287601.00	6.37	1228.41	382.32
2	215179.00	215177.40	0.00	215179.00	0.00	1582.38	92.91
3	248270.00	248248.85	0.01	263253.00	6.03	3388.78	447.26
4	514485.00	445945.02	13.32	452514.00	-12.05	3600.36	160.67
5	226587.00	165404.41	27.00	189561.00	-16.34	3600.33	92.77
6	407989.00	375231.63	8.03	400510.00	-1.83	3600.28	128.63
7	494646.00	462180.27	6.56	484512.00	-2.05	3600.42	292.47
8	252686.00	244113.80	3.39	271367.00	7.39	3600.50	466.30
9	332245.00	270257.16	18.66	295594.00	-11.03	3600.19	304.65
10	301935.00	264398.61	12.43	288633.00	-4.41	3600.55	487.66
		Avg.	8.94	Avg.	-2.79	3140.22	285.56

Class 4	CPLEX			LS	LS-CPLEX	CPLEX	LS
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	558079.00	480350.24	13.93	515376.00	-7.65	3600.45	863.15
2	494246.00	474283.28	4.04	520205.00	5.25	3600.58	816.90
3	577493.00	505582.79	12.45	538035.00	-6.83	3600.80	1000.91
4	466804.00	440616.56	5.61	478011.00	2.40	3600.42	271.59
5	525553.00	446861.13	14.97	486324.00	-7.46	3600.42	962.71
6	467743.00	467743.00	0.00	481577.00	2.96	1771.44	1094.85
7	522475.00	450064.00	13.86	505247.00	-3.30	3600.52	784.77
8	638842.00	563592.16	11.78	603267.00	-5.57	3600.47	563.56
9	403563.00	332035.99	17.72	399002.00	-1.13	3600.63	860.39
10	444049.00	380239.43	14.37	423335.00	-4.66	3600.39	445.90
		Avg.	10.87	Avg.	-2.60	3417.61	766.47

Table 22 Computational Results LS Decreasing Demand (Classes 1 and 2)

Class 5	CPLEX			LS	LS-CPLEX	CPLEX	LS
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	267263	267238	0.01	267263	0.00	1332.97	8.33
2	200544	200524	0.01	200544	0.00	526.09	6.02
3	313671	294553	6.10	311915	-0.56	3600.30	8.09
4	225080	225080	0.00	225687	0.27	19.69	2.32
5	221074	221067	0.00	221074	0.00	78.84	7.76
6	237742	237742	0.00	237742	0.00	24.30	11.34
7	299716	281456	6.09	298346	-0.46	3600.29	15.85
8	253168	253144	0.01	253168	0.00	148.31	6.91
9	243937	243913	0.01	243937	0.00	1709.89	40.65
10	243999	243975	0.01	243999	0.00	1029.02	22.22
		Avg.	1.22	Avg.	-0.07	1206.97	12.95

Class 6	CPLEX			LS	LS-CPLEX	CPLEX	LS
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	307865.00	307865.00	0.00	307865.00	0.00	47.17	14.41
2	340797.00	325991.33	4.34	343557.00	0.81	3600.36	237.63
3	249534.00	249531.07	0.00	249534.00	0.00	210.19	70.58
4	474207.00	440357.62	7.14	457962.00	-3.43	3600.27	138.64
5	369269.00	369233.39	0.01	369269.00	0.00	408.97	13.09
6	236873.00	236857.44	0.01	236873.00	0.00	480.28	96.03
7	279579.00	265943.64	4.88	275086.00	-1.61	3600.45	99.50
8	315755.00	315751.49	0.00	315755.00	0.00	422.59	91.67
9	236059.00	236043.48	0.01	236059.00	0.00	774.05	90.77
10	345205.00	345171.62	0.01	345205.00	0.00	2344.59	105.11
		Avg.	1.64	Avg.	-0.42	1548.89	95.74

VI.5. Summary and Conclusion

In this chapter we described the RCFLP and presented a mixed integer programming formulation. We implemented two metaheuristics to solve this model, local search (LS) and simulated annealing (SA). Both heuristics take as an initial feasible solution the set of open facilities obtained by solving the Lagrangian subproblem obtained from the Lagrangian relaxation of the capacity constraints. The neighborhood function for both heuristics consider add, drop, and exchange moves. In reducing the computational effort to evaluate the objective function value for each neighbor solu-

Table 23 Computational Results LS Decreasing Demand (Classes 3 and 4)

Class 7	CPLEX			LS	LS-CPLEX	CPLEX	LS
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	565828.00	565828.00	0.00	565828.00	0.00	100.09	31.08
2	627375.00	627316.54	0.01	627375.00	0.00	92.39	37.80
3	552293.00	544830.41	1.35	548868.00	-0.62	3600.78	286.46
4	414351.00	414323.63	0.01	414351.00	0.00	480.50	98.03
5	700215.00	668166.52	4.58	696280.00	-0.56	3600.67	376.95
6	565170.00	512693.61	9.29	548129.00	-3.02	3600.56	346.81
7	423403.00	372225.87	12.09	395033.00	-6.70	3600.33	225.02
8	521384.00	521377.07	0.00	521384.00	0.00	724.91	47.58
9	593353.00	531283.41	10.46	540575.00	-8.89	3600.32	225.21
10	566148.00	559344.86	1.20	566148.00	0.00	3600.28	224.88
		Avg.	3.90	Avg.	-1.98	2300.08	189.98

Class 8	CPLEX			LS	LS-CPLEX	CPLEX	LS
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	702783.00	647558.29	7.86	675775.00	-4.00	3600.56	355.21
2	715920.00	715862.35	0.01	716010.00	0.01	1094.09	85.80
3	741152.00	658947.02	11.09	670544.00	-10.53	3600.44	245.74
4	741904.00	741904.00	0.00	741904.00	0.00	322.75	103.83
5	746109.00	663828.11	11.03	677983.00	-10.05	3600.56	284.75
6	656385.00	656353.32	0.00	656385.00	0.00	1931.55	131.53
7	751999.00	679209.63	9.68	692109.00	-8.65	3600.38	252.11
8	699806.00	609001.81	12.98	637120.00	-9.84	3600.34	296.55
9	813327.00	767014.63	5.69	783272.00	-3.84	3600.52	282.71
10	752052.00	690376.36	8.20	706800.00	-6.40	3600.30	250.54
		Avg.	6.65	Avg.	-5.33	2855.15	228.88

tion, we develop an approximate algorithm. Once a candidate solution is selected by the heuristic, we compute the exact objective function value solving a linear program. Both, LS and SA algorithms showed to be efficient in minimizing the maximum regret compared with branch and cut approach to solve classes of problems for each demand structure. We observed that the initial feasible solution provides significant improvements in the performance of the heuristic algorithms. In general, the implementation of LS showed to be more efficient than SA in minimizing the maximum regret in less computational time.

Table 24 Computational Results LS Steady Demand (Classes 1 and 2)

Class 9	CPLEX			LS	LS-CPLEX	CPLEX	LS
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	61362.00	47294.20	22.93	66940.00	9.09	3600.19	100.86
2	83338.00	83336.32	0.00	93342.00	12.00	2290.95	62.18
3	107371.00	79307.31	26.14	116036.00	8.07	3600.24	74.43
4	49795.00	49793.80	0.00	49795.00	0.00	294.86	35.43
5	50318.00	50318.00	0.00	50318.00	0.00	446.36	37.40
6	62753.00	62753.00	0.00	62753.00	0.00	185.74	39.18
7	89816.00	56215.02	37.41	85687.00	-4.60	3600.22	102.76
8	29958.00	29958.00	0.00	29958.00	0.00	52.58	32.87
9	66910.00	66909.04	0.00	66910.00	0.00	1853.06	221.74
10	69737.00	69733.82	0.00	69737.00	0.00	1756.09	143.77
		Avg.	8.65	Avg.	2.46	1768.03	85.06

Class 10	CPLEX			LS	LS-CPLEX	CPLEX	LS
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	105897.00	105897.00	0.00	106032.00	0.13	576.24	71.46
2	140244.00	73939.02	47.28	121301.00	-13.51	3600.21	247.56
3	163021.00	96080.19	41.06	112741.00	-30.84	3600.21	352.32
4	193196.00	156247.14	19.13	193196.00	0.00	3600.26	102.17
5	201706.00	149748.82	25.76	159905.00	-20.72	3600.26	491.32
6	88416.00	77396.04	12.46	106592.00	20.56	3600.21	549.05
7	128912.00	71278.48	44.71	105097.00	-18.47	3600.21	360.43
8	131308.00	131295.68	0.01	131308.00	0.00	3378.05	81.38
9	132139.00	79046.94	40.18	94904.00	-28.18	3600.21	420.13
10	134748.00	128992.51	4.27	134748.00	0.00	3600.30	133.69
		Avg.	23.49	Avg.	-9.10	3275.62	280.95

Table 25 Computational Results LS Steady Demand (Classes 3 and 4)

Class 11	CPLEX			LS	LS-CPLEX	CPLEX	LS
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	118247.00	118247.00	0.00	118247.00	0.00	2706.85	113.57
2	120718.00	120718.00	0.00	120718.00	0.00	1729.18	312.32
3	113619.00	85369.13	24.86	120356.00	5.93	3600.23	788.50
4	94627.00	94624.20	0.00	94627.00	0.00	3179.52	112.12
5	176508.00	114857.37	34.93	175664.00	-0.48	3600.34	925.79
6	164798.00	120372.47	26.96	193083.00	17.16	3600.26	1212.76
7	133503.00	99524.66	25.45	132919.00	-0.44	3600.26	980.46
8	100485.00	93597.26	6.85	122139.00	21.55	3600.27	573.17
9	170397.00	99234.88	41.76	137030.00	-19.58	3600.26	729.50
10	156581.00	81108.27	48.20	130918.00	-16.39	3600.35	666.04
		Avg.	20.90	Avg.	0.78	3281.75	641.42

Class 12	CPLEX			LS	LS-CPLEX	CPLEX	LS
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	259174.00	203043.10	21.66	310377.00	16.50	3600.43	977.59
2	168000.00	149650.64	10.92	191866.00	12.44	3600.41	783.77
3	233450.00	163305.42	30.05	201862.00	-15.65	3600.46	1234.31
4	273129.00	175091.13	35.89	229240.00	-19.15	3600.73	863.18
5	247613.00	184005.98	25.69	218736.00	-13.20	3600.48	1044.71
6	297753.00	236711.97	20.50	263873.00	-12.84	3600.53	728.79
7	289097.00	206656.18	28.52	245165.00	-17.92	3600.45	1274.12
8	191863.00	126900.62	33.86	236312.00	18.81	3600.73	1236.38
9	303282.00	240716.07	20.63	262744.00	-15.43	3600.50	920.40
10	288895.00	218745.37	24.28	276762.00	-4.38	3600.46	1285.30
		Avg.	25.20	Avg.	-5.08	3600.52	1034.85

Table 26 Computational Results SA Increasing Demand (Classes 1 and 2)

Class 1	CPLEX			SA	SA-CPLEX	CPLEX	SA
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	200874.00	175423.43	12.67	200806.00	-0.03	3600.28	129.25
2	152561.00	152561.00	0.00	152561.00	0.00	32.64	127.86
3	141059.00	101982.10	27.70	136708.00	-3.08	3600.20	141.44
4	150659.00	150659.00	0.00	151536.00	0.58	73.83	128.31
5	173181.00	173181.00	0.00	180849.00	4.43	748.92	134.39
6	75125.00	75125.00	0.00	75125.00	0.00	46.94	50.66
7	161073.00	119591.05	25.75	155529.00	-3.44	3600.23	61.33
8	186338.00	153073.16	17.85	162386.00	-12.85	3600.24	54.06
9	85425.00	85425.00	0.00	85425.00	0.00	268.91	63.38
10	86480.00	86480.00	0.00	86992.00	0.59	204.70	59.60
		Avg.	8.40	Avg.	-1.38	1577.69	95.03

Class 2	CPLEX			SA	SA-CPLEX	CPLEX	SA
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	169845.00	169845.00	0.00	169845.00	0.00	101.50	64.42
2	253107.00	253086.64	0.01	253107.00	0.00	2786.90	59.11
3	267963.00	226789.42	15.37	270722.00	1.03	3600.26	103.92
4	185320.00	154517.55	16.62	185320.00	0.00	3600.18	80.38
5	202101.00	165527.44	18.10	170043.00	-15.86	3600.26	103.67
6	302453.00	302446.28	0.00	302453.00	0.00	349.44	153.28
7	296851.00	257372.67	13.30	303994.00	2.41	3600.23	182.09
8	164826.00	164826.00	0.00	164826.00	0.00	563.99	161.58
9	285119.00	285094.01	0.01	296871.00	4.12	209.13	145.19
10	179675.00	179662.48	0.01	179675.00	0.00	1639.95	179.73
		Avg.	6.34	Avg.	-0.83	2005.18	123.34

Table 27 Computational Results SA Increasing Demand (Classes 3 and 4)

Class 3	CPLEX			SA	SA-CPLEX	CPLEX	SA
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	270386.00	270374.41	0.00	270386.00	0.00	1228.41	285.77
2	215179.00	215177.40	0.00	215179.00	0.00	1582.38	234.24
3	248270.00	248248.85	0.01	248270.00	0.00	3388.78	259.77
4	514485.00	445945.02	13.32	452514.00	-12.05	3600.36	224.86
5	226587.00	165404.41	27.00	202495.00	-10.63	3600.33	762.86
6	407989.00	375231.63	8.03	407889.00	-0.02	3600.28	1075.30
7	494646.00	462180.27	6.56	494139.00	-0.10	3600.42	443.66
8	252686.00	244113.80	3.39	275147.00	8.89	3600.50	242.44
9	332245.00	270257.16	18.66	319078.00	-3.96	3600.19	408.39
10	301935.00	264398.61	12.43	291187.00	-3.56	3600.55	218.14
		Avg.	8.94	Avg.	-2.14	3140.22	415.54

Class 4	CPLEX			SA	SA-CPLEX	CPLEX	SA
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	558079.00	480350.24	13.93	516841.00	-7.39	3600.45	376.55
2	494246.00	474283.28	4.04	521168.00	5.45	3600.58	313.80
3	577493.00	505582.79	12.45	547825.00	-5.14	3600.80	1933.17
4	466804.00	440616.56	5.61	486745.00	4.27	3600.42	1275.50
5	525553.00	446861.13	14.97	510963.00	-2.78	3600.42	2229.25
6	467743.00	467743.00	0.00	467743.00	0.00	1771.44	297.05
7	522475.00	450064.00	13.86	509827.00	-2.42	3600.52	1108.20
8	638842.00	563592.16	11.78	638189.00	-0.10	3600.47	893.62
9	403563.00	332035.99	17.72	402420.00	-0.28	3600.63	1492.39
10	444049.00	380239.43	14.37	425921.00	-4.08	3600.39	174.52
		Avg.	10.87	Avg.	-1.25	3417.61	1009.41

Table 28 Computational Results SA Decreasing Demand (Classes 1 and 2)

Class 5	CPLEX			SA	SA-CPLEX	CPLEX	SA
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	267263.00	267238.23	0.01	267263.00	0.00	1332.97	206.75
2	200544.00	200524.10	0.01	200544.00	0.00	526.09	205.05
3	313671.00	294552.69	6.10	313031.00	-0.20	3600.30	210.86
4	225080.00	225080.00	0.00	225687.00	0.27	19.69	205.05
5	221074.00	221066.82	0.00	221074.00	0.00	78.84	198.75
6	237742.00	237742.00	0.00	237742.00	0.00	24.30	205.80
7	299716.00	281456.38	6.09	298346.00	-0.46	3600.29	205.00
8	253168.00	253144.00	0.01	253168.00	0.00	148.31	209.00
9	243937.00	243912.76	0.01	243937.00	0.00	1709.89	217.41
10	243999.00	243974.78	0.01	243999.00	0.00	1029.02	196.91
		Avg.	1.22	Avg.	-0.04	1206.97	206.06

Class 6	CPLEX			SA	SA-CPLEX	CPLEX	SA
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	307865.00	307865.00	0.00	307865.00	0.00	47.17	150.36
2	340797.00	325991.33	4.34	343557.00	0.81	3600.36	162.83
3	249534.00	249531.07	0.00	249534.00	0.00	210.19	154.95
4	474207.00	440357.62	7.14	457962.00	-3.43	3600.27	150.59
5	369269.00	369233.39	0.01	369269.00	0.00	408.97	161.92
6	236873.00	236857.44	0.01	236873.00	0.00	480.28	52.19
7	279579.00	265943.64	4.88	275086.00	-1.61	3600.45	57.86
8	315755.00	315751.49	0.00	315755.00	0.00	422.59	43.50
9	236059.00	236043.48	0.01	237190.00	0.48	774.05	47.55
10	345205.00	345171.62	0.01	345205.00	0.00	2344.59	60.69
		Avg.	1.64	Avg.	-0.37	1548.89	104.24

Table 29 Computational Results SA Decreasing Demand (Classes 3 and 4)

Class 7	CPLEX			SA	SA-CPLEX	CPLEX	SA
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	565828.00	565828.00	0.00	564632.00	-0.21	100.09	100.91
2	627375.00	627316.54	0.01	627375.00	0.00	92.39	73.67
3	552293.00	544830.41	1.35	547844.00	-0.81	3600.78	119.34
4	414351.00	414323.63	0.01	414351.00	0.00	480.50	74.94
5	700215.00	668166.52	4.58	696280.00	-0.56	3600.67	107.64
6	565170.00	512693.61	9.29	548129.00	-3.02	3600.56	96.95
7	423403.00	372225.87	12.09	395033.00	-6.70	3600.33	70.50
8	521384.00	521377.07	0.00	521384.00	0.00	724.91	63.27
9	593353.00	531283.41	10.46	540575.00	-8.89	3600.32	70.88
10	566148.00	559344.86	1.20	566148.00	0.00	3600.28	81.66
		Avg.	3.90	Avg.	-2.02	2300.08	85.97

Class 8	CPLEX			SA	SA-CPLEX	CPLEX	SA
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	702783.00	647558.29	7.86	675775.00	-4.00	3600.56	2542.26
2	715920.00	715862.35	0.01	716010.00	0.01	1094.09	298.14
3	741152.00	658947.02	11.09	670544.00	-10.53	3600.44	281.78
4	741904.00	741904.00	0.00	741904.00	0.00	322.75	177.01
5	746109.00	663828.11	11.03	677983.00	-10.05	3600.56	1599.67
6	656385.00	656353.32	0.00	656385.00	0.00	1931.55	160.80
7	751999.00	679209.63	9.68	692109.00	-8.65	3600.38	422.40
8	699806.00	609001.81	12.98	637120.00	-9.84	3600.34	1713.63
9	813327.00	767014.63	5.69	783272.00	-3.84	3600.52	1514.26
10	752052.00	690376.36	8.20	706800.00	-6.40	3600.30	438.39
		Avg.	6.65	Avg.	-5.33	2855.15	914.83

Table 30 Computational Results SA Steady Demand (Classes 1 and 2)

Class 9	CPLEX			SA	SA-CPLEX	CPLEX	SA
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	61362.00	47294.20	22.93	61362.00	0.00	3600.19	83.39
2	83338.00	83336.32	0.00	83338.00	0.00	2290.95	48.33
3	107371.00	79307.31	26.14	100983.00	-5.95	3600.24	40.61
4	49795.00	49793.80	0.00	49795.00	0.00	294.86	46.16
5	50318.00	50318.00	0.00	52091.00	3.52	446.36	49.72
6	62753.00	62753.00	0.00	62753.00	0.00	185.74	38.44
7	89816.00	56215.02	37.41	91518.00	1.89	3600.22	37.19
8	29958.00	29958.00	0.00	29958.00	0.00	52.58	34.88
9	66910.00	66909.04	0.00	66910.00	0.00	1853.06	69.66
10	69737.00	69733.82	0.00	69737.00	0.00	1756.09	53.94
		Avg.	8.65	Avg.	-0.05	1768.03	50.23

Class 10	CPLEX			SA	SA-CPLEX	CPLEX	SA
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	105897.00	105897.00	0.00	106032.00	0.13	576.24	58.25
2	140244.00	73939.02	47.28	121301.00	-13.51	3600.21	183.09
3	163021.00	96080.19	41.06	129168.00	-20.77	3600.21	125.98
4	193196.00	156247.14	19.13	193196.00	0.00	3600.26	86.98
5	201706.00	149748.82	25.76	159905.00	-20.72	3600.26	71.09
6	88416.00	77396.04	12.46	106032.00	19.92	3600.21	82.35
7	128912.00	71278.48	44.71	111228.00	-13.72	3600.21	129.72
8	131308.00	131295.68	0.01	131308.00	0.00	3378.05	90.63
9	132139.00	79046.94	40.18	94904.00	-28.18	3600.21	89.14
10	134748.00	128992.51	4.27	134748.00	0.00	3600.30	75.20
		Avg.	23.49	Avg.	-7.68	3275.62	99.24

Table 31 Computational Results SA Steady Demand (Classes 3 and 4)

Class 11	CPLEX			SA	SA-CPLEX	CPLEX	SA
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	118247.00	118247.00	0.00	118247.00	0.00	2706.85	198.92
2	120718.00	120718.00	0.00	120718.00	0.00	1729.18	108.75
3	113619.00	85369.13	24.86	122607.00	7.91	3600.23	864.68
4	94627.00	94624.20	0.00	94627.00	0.00	3179.52	273.25
5	176508.00	114857.37	34.93	178339.00	1.04	3600.34	665.85
6	164798.00	120372.47	26.96	185660.00	12.66	3600.26	143.06
7	133503.00	99524.66	25.45	129498.00	-3.00	3600.26	143.80
8	100485.00	93597.26	6.85	122139.00	21.55	3600.27	143.63
9	170397.00	99234.88	41.76	170171.00	-0.13	3600.26	140.42
10	156581.00	81108.27	48.20	133598.00	-14.68	3600.35	143.22
		Avg.	20.90	Avg.	2.53	3281.75	282.56

Class 12	CPLEX			SA	SA-CPLEX	CPLEX	SA
Instance	UB	LB	Gap (%)	Obj. Value	Diff. (%)	Time (sec.)	Time (sec.)
1	259174.00	203043.10	21.66	312598.00	17.09	3600.43	2285.75
2	168000.00	149650.64	10.92	193570.00	13.21	3600.41	540.27
3	233450.00	163305.42	30.05	207997.00	-12.24	3600.46	730.91
4	273129.00	175091.13	35.89	234088.00	-16.68	3600.73	735.99
5	247613.00	184005.98	25.69	229582.00	-7.85	3600.48	1360.50
6	297753.00	236711.97	20.50	264392.00	-12.62	3600.53	1228.81
7	289097.00	206656.18	28.52	265622.00	-8.84	3600.45	1651.03
8	191863.00	126900.62	33.86	274482.00	30.10	3600.73	1972.09
9	303282.00	240716.07	20.63	266197.00	-13.93	3600.50	2059.46
10	288895.00	218745.37	24.28	284409.00	-1.58	3600.46	2350.73
		Avg.	25.20	Avg.	-1.33	3600.52	1491.55

CHAPTER VII

CONCLUSIONS AND FUTURE DIRECTIONS

In this dissertation, we presented models for dynamic location, dynamic demand without relocation of facilities, and robust location problems. The dynamic model determines the optimal time and location for establishing capacitated facilities to supply the demand of customers over a discrete and finite time horizon. The model for dynamic demand without relocation finds a fixed configuration of capacitated facilities to serve the time varying demand. The robust model determines a fixed configuration of capacitated facilities with the objective of minimizing the worst-case cost or maximum regret. This measure of robustness is commonly used in the literature to evaluate decisions under uncertainty.

We also described three different structures for the total demand of customers and the behavior of the demand for each customer location. These demand structures motivate the analysis of the dynamic and robust models and provided a mean to test the performance of the solution methods developed for each model.

The Lagrangian relaxation and Benders' decomposition algorithms developed for the dynamic model studied in Chapter IV performed well providing good quality solutions in acceptable computational time, compared with conventional branch and cut. The structure of the associated subproblems for each algorithm played an important role in solving the dynamic model. We observed that for classes of problems where the split or portion of the total average cost considers a large portion of fixed cost, the solution algorithms have an improved performance.

The Benders' decomposition algorithm for dynamic demand without relocation presented in Chapter V also showed to be efficient. We showed that when relocation costs are considerably large, the dynamic location model can be used to solve the model for dynamic demand without relocation. The cost structure of the average total cost also determined the performance of the solution algorithm as in the dynamic model.

For the robust model studied in Chapter VI, we implemented two heuristic algorithms, both being efficient in providing near optimal solutions in acceptable computational time. We obtained improved solutions using a Lagrangian relaxation approach to obtain the initial feasible solution. The neighborhood function, which considers add, drop, and exchange moves also provided an efficient way of exploring the neighborhood of each candidate solution. Overall the classes of problems, the heuristic solution algorithms provided solutions with minimum worst-case regret in less computational time than conventional branch and cut.

Future research directions for the dynamic location model is the additional restriction in the allocation of customers to a single facility, or single sourcing. Another interesting area of research for the dynamic location model is to consider multi-stage location problems, such as distribution system design. The location of facilities and distribution centers, as well as the allocation of customers can be considered when demand and cost parameters are time varying. The problem structure offers a possibility to implement decomposition algorithms and the analysis of different demand structures.

For the problem of dynamic demand without relocation, we can consider a fixed number of open facilities in each period. That is, a p-median problem with dynamic demand and without relocation of facilities. The solution algorithms developed for our model can be applicable to the p-median version since the only difference is that the number of open facilities is given.

Future research for robust location model may include the consideration of different robustness measures. An interesting problem for the robust location model is to determine the location of emergency services when demand and cost parameters change by time and the objective is to minimize the maximum response time or minimizing the worst-case regret in response time. Alternative measures of robustness can lead to a different mixed integer programming formulation for which any of the solution algorithms developed in this dissertation may be applicable.

REFERENCES

Andreatta, G., F. M. Mason. 1994. A note on: A perfect forward procedure for a single facility dynamic location/relocation problem. *Operations Research Letters* **15**(2) 81–83.

Arostegui, M. A., S. N. Kadipasaoglu, B. M. Khumawala. 2006. An empirical comparison of tabu search, simulated annealing, and genetic algorithms for facilities location problems. *International Journal of Production Economics* **103**(2) 742–754.

Averbakh, I. 2000. Minmax regret solutions for minimax optimization problems with uncertainty. *Operations Research Letters* **27**(2) 57–65.

Averbakh, I. 2005. The minmax relative regret median problem on networks. *INFORMS Journal on Computing* **17**(4) 451.

Averbakh, I., O. Berman. 1997. Minimax regret p-center location on a network with demand uncertainty. *Location Science* **5**(4) 247–254.

Averbakh, I., O. Berman. 2000. Algorithms for the robust 1-center problem on a tree. *European Journal of Operational Research* **123**(2) 292–302.

Baker, B. M., J. Sheasby. 1999. Accelerating the convergence of subgradient optimisation. *European Journal of Operational Research* **117**(1) 136–144.

Balakrishnan, J. 2004. An improved algorithm for solving a multi-period facility location problem. *IIE Transactions* **36**(1) 19–22.

Ballou, R. H. 1968. Dynamic warehouse location analysis. *Journal of Marketing Research* **5**(3) 271–276.

Barcelo, J., J. Casanovas. 1984. A heuristic lagrangean algorithm for the capacitated plant location problem. *European Journal of Operational Research* **15**(2) 212–226.

Bastian, M., M. Volkmer. 1992. A perfect forward procedure for a single facility dynamic location/relocation problem. *Operations Research Letters* **12**(1) 11–16.

Beasley, J. E. 1988. An algorithm for solving large capacitated warehouse location problems. *European Journal of Operational Research* **33**(3) 314–325.

Beasley, J. E. 1993. Lagrangean heuristics for location problems. *European Journal of Operational Research* **65**(3) 383–399.

Bellman, R. 1966. Dynamic programming. *Science* **153**(3731) 34–37.

Benders, J. F. 1962. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik* **4**(1) 238–252.

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

Canel, C., S. R. Das. 1999. The uncapacitated multi-period facilities location problem with profit maximization. *International Journal of Physical Distribution & Logistics Management* **29**(6) 409–433.

Canel, C., B. M. Khumawala. 1996. A mixed-integer programming approach for the international facilities location problem. *International Journal of Operations & Production Management* **16**(4) 49–68.

Canel, C., B. M. Khumawala. 1997. Multi-period international facilities location: an algorithm and application. *International Journal of Production Research* **35**(7) 1891–1910.

Carrizosa, E., S. Nickel. 2003. Robust facility location. *Mathematical Methods of Operations Research (ZOR)* **58**(2) 331–349.

Chand, S. 1988. Decision/forecast horizon results for a single facility dynamic location/relocation problem. *Operations Research Letters* **7**(5) 247–251.

Chardaire, P., A. Sutter, M. C. Costa. 1996. Solving the dynamic facility location problem. *Networks* **28**(2) 117–124.

Chrissis, J. W., R. P. Davis, D. M. Miller. 1982. Dynamic set covering problem. *Applied Mathematical Modelling* **6**(1) 2–6.

Christofides, N., J. E. Beasley. 1983. Extensions to a lagrangean relaxation approach for the capacitated warehouse location problem. *European Journal of Operational Research* **12**(1) 19–28.

Cornuejols, G., M. L. Fisher, G. L. Nemhauser. 1977. Location of bank accounts to optimize float: An analytic study of exact and approximate algorithms. *Management Science* **23**(8) 789–810.

Cornuejols, G., R. Sridharan, J. M. Thizy. 1991. A comparison of heuristics and relaxations for the capacitated plant location problem. *European Journal of Operational Research* **50**(3) 280–297.

Current, J., S. Ratick, C. ReVelle. 1998. Dynamic facility location when the total number of facilities is uncertain: A decision analysis approach. *European Journal of Operational Research* **110**(3) 597–609.

Daskin, M. S. 1995. *Network and Discrete Location: Models Algorithms and Applications*. Wiley, New York, NY, USA.

Daskin, M. S., S. M. Hesse, C. S. Revelle. 1997. α-Reliable p-minimax regret: A new model for strategic facility location modeling. *Location Science* **5**(4) 227–246.

Daskin, M. S., W. J. Hopp, B. Medina. 1992. Forecast horizons and dynamic facility location planning. *Annals of Operations Research* **40**(1) 125–151.

Dias, J., M. E. Captivo, J. Clímaco. 2001a. Capacitated dynamic location problems with opening, closure and reopening of facilities. Research report, INESC-Coimbra, Lisboa, Portugal.

Dias, J., M. E. Captivo, J. Clímaco. 2001b. A hybrid algorithm for dynamic location problems. Research report, INESC-Coimbra, Lisboa, Portugal.

Dias, J., M. E. Captivo, J. Clímaco. 2004a. Dynamic location problems with discrete expansion and reduction sizes of available capacities. Research report, INESC-Coimbra, Lisboa, Portugal.

Dias, J., M. E. Captivo, J. Clímaco. 2004b. Dynamic multi-level capacitated and uncapacitated location problems: An approach using primal-dual heuristics. Research report, INESC-Coimbra, Lisboa, Portugal.

Drezner, T., Z. Drezner, S. Salhi. 2002. Solving the multiple competitive facilities location problem. *European Journal of Operational Research* **142**(1) 138–151.

Drezner, Z. 1984. The planar two-center and two-median problems. *Transportation Science* **18** 351–361.

Drezner, Z. 1995a. Dynamic facility location: The progressive p-median problem. *Location Science* **3**(1) 1–7.

Drezner, Z. 1995b. *Facility location: A Survey of Applications and Methods*. Springer-Verlag, New York, NY, USA.

Drezner, Z., H. Hamacher. 2002. *Facility Location: Applications and Theory*. Springer-Verlag, New York, NY, USA.

Drezner, Z., G. O. Wesolowsky. 1991. Facility location when demand is time dependent. *Naval Research Logistics* **38**(5) 763–777.

Efroymson, M. A., T. L. Ray. 1966. A branch-bound algorithm for plant location. *Operations Research* **14**(3) 361–368.

Erlenkotter, D. 1974. Dynamic facility location and simple network models. *Management Science* **20**(9) 1311–1311.

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

Erlenkotter, D. 1981. A comparative study of approaches to dynamic location problems. *European Journal of Operational Research* **6** 133–143.

Fisher, M. L. 2004. The lagrangian relaxation method for solving integer programming problems. *Management Science* **50**(13) 1861–1871.

Francis, R. L., P. B. Mirchandani. 1990. *Discrete location theory*. Wiley, New York, NY, USA.

Frantzeskakis, M., C. D. T. Watson-Gandy. 1989. The use of state space relaxation for the dynamic facility location problem. *Annals of Operations Research* **18**(1) 187–211.

Geoffrion, A. M. 1974. Lagrangean relaxation for integer programming. *Mathematical Programming Study* **2**(1974) 82–114.

Geoffrion, A. M., G. W. Graves. 1974. Multicommodity distribution system design by benders decomposition. *Management Science* **20**(5) 822–844.

Geoffrion, A. M., R. McBride. 1978. Lagrangean relaxation applied to capacitated facility location problems. *IIE Transactions* **10**(1) 40–47.

Guignard-Spielberg, M., K. Spielberg. 1977. *A dual method for the mixed plant location problem*. Department of Statistics, University of Pennsylvania, The Wharton School, Philadelphia, PA, USA.

Gunawardane, G. 1982. Dynamic versions of set covering type public facility location problems. *European Journal of Operational Research* **10**(2) 190–195.

Hakimi, S. L., M. Labbe, E. F. Schmeichel. 1999. Locations on time-varying networks. *Networks* **34**(4) 250–257.

Held, M., R. M. Karp. 1970. The traveling salesman problem and minimum spanning trees. *Operations Research* **18**(6) 1138–1162.

Held, M., R. M. Karp. 1971. The traveling-salesman problem and minimum spanning trees: Part ii. *Mathematical Programming* **1**(1) 6–25.

Held, M., P. Wolfe, H. P. Crowder. 1974. Validation of subgradient optimization. *Mathematical Programming* **6**(1) 62–88.

Hindi, K. S., K. Pieńkosz. 1999. Efficient solution of large scale, single-source, capacitated plant location problems. *Journal of the Operational Research Society* **50**(3) 268–274.

Holmberg, K., M. Rönnqvist, D. Yuan. 1999. An exact algorithm for the capacitated facility location problems with single sourcing. *European Journal of Operational Research* **113**(3) 544–559.

Hormozi, A. M., B. M. Khumawala. 1996. An improved algorithm for solving a multi-period facility location problem. *IIE Transactions* **28**(2) 105–114.

Kelly, D. L., A. J. Marucheck. 1984. Planning horizon results for the dynamic warehouse location problem. *Journal of Operations Management* **4** 279–94.

Khumawala, B. M. 1972. An efficient branch-bound algorithm for the warehouse location problem. *Management Science* **18**(12) 718–732.

Kilmer, R. L., T. Spreen, D. S. Tilley. 1983. A dynamic plant location model: The east florida fresh citrus packing industry. *American Journal of Agricultural Economics* **65**(4) 730–737.

Kincaid, R. K. 1992. Good solutions to discrete noxious location problems via metaheuristics. *Annals of Operations Research* **40**(1) 265–281.

Kirkpatrick, S., C. D. Gelatt, M. P. Vecchi. 1983. Optimization by simulated annealing. *Science* **220**(4598) 671–680.

Klincewicz, J. G., H. Luss. 1986. Lagrangian relaxation heuristic for capacitated facility location with single-source constraints. *Journal of the Operational Research Society* **37** 495–500.

Kolesar, P., W. E. Walker. 1974. An algorithm for the dynamic relocation of fire companies. *Operations Research* **22**(2) 249–274.

Kouvelis, P., G. Yu. 1997. *Robust Discrete Optimization and Its Applications*. Springer, Secaucus, NJ, USA.

Kuehn, A. A., M. J. Hamburger. 1963. A heuristic program for locating warehouses. *Management Science* **9**(4) 643–666.

Love, R. F., J. G. Morris, G. O. Wesolowsky. 1988. *Facilities Location: Models & Methods*. North-Holland, New York, NY, USA.

Magnanti, T. L., R. T. Wong. 1981. Accelerating Benders decomposition: Algorithmic enhancement and model selection criteria. *Operations Research* **29**(3) 464–484.

Manne, A. S. 1967. *Investments for Capacity Expansion: Size, Location and Time Phasing*. The MIT Press, Cambridge, MA, USA.

Mavridou, T. D., P. M. Pardalos. 1997. Simulated annealing and genetic algorithms for the facility layout problem: A survey. *Computational Optimization and Applications* **7**(1) 111–126.

McDaniel, D., M. Devine. 1977. A modified Benders partitioning algorithm for mixed integer programming. *Management Science* **24**(3) 312–319.

Megiddo, N. 1986. Dynamic location problems. *Annals of Operations Research* **6**(10) 311–319.

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

Nemhauser, G. L., L. A. Wolsey. 1988. *Integer and Combinational Optimization*. John Wiley & Sons, New York, NY, USA.

Owen, H. S., M. S. Daskin. 1998. Strategic facility location: A review. *European Journal of Operational Research* **111**(3) 423–447.

Revelle, C., C. Toregas, L. Falkson. 1976. Applications of the location set-covering problem. *Geographical Analysis* **8** 65–76.

Ríos-Ramírez, A. 2003. International assembly facility location problem. *Investigación Multidisciplinaria* **1**(1) 1–8.

Roodman, G. M., L. B. Schwarz. 1975. Optimal and heuristic facility phase-out strategies. *AIIE Transactions* **7**(2) 177–184.

Roodman, G. M., L. B. Schwarz. 1977. Extensions of the multi-period facility phase-out model: New procedures and applications to a phase-in/phase-out problem. *AIIE Transactions* **9**(1) 103–107.

Saldanha da Gama, F., M. E. Captivo. 1998. A heuristic approach for the discrete dynamic location problem. *Location Science* **6**(1) 211–223.

Schilling, D. 1980. Dynamic location modeling for public-sector facilities: A multi-criteria approach. *Decision Sciences* **11**(4) 714–724.

Scott, A. J. 1971. Dynamic location-allocation systems: Some basic planning strategies. *Environment and Planning A* **3**(1) 73–82.

Serra, D., V. Marianov. 1998. The p-median problem in a changing network: The case of Barcelona. *Location Science* **6**(1-4) 383–394.

Shulman, A. 1991. An algorithm for solving capacitated plant location problems with discrete expansion sizes. *Operations Research* **39**(3) 423–36.

Snyder, L., M. Daskin. 2006. Stochastic p-robust location problems. *IIE Transactions* **38**(11) 971–985.

Snyder, L. V. 2006. Facility location under uncertainty: A review. *IIE Transactions* **38**(7) 547–564.

Sridharan, R. 1993. A lagrangian heuristic for the capacitated plant location problem with single source constraints. *European Journal of Operational Research* **66**(3) 305–312.

Sweeney, D. J., R. L. Tatham. 1976. An improved long-run model for multiple warehouse location. *Management Science* **22**(7) 748–758.

Vairaktarakis, G. L., P. Kouvelis. 1999. Incorporation dynamic aspects and uncertainty in 1-median location problems. *Naval Research Logistics* **46**(2) 147–168.

van Laarhoven, P. J. M., E. H. L. Aarts. 1987. *Simulated Annealing: Theory and Applications*. Reidel, Dordrecht, The Netherlands.

Van Roy, T. J. 1986. A cross decomposition algorithm for capacitated facility location. *Operations Research* **34**(1) 145–163.

Van Roy, T. J., D. Erlenkotter. 1982. A dual based procedure for dynamic facility location. *Management Science* **28** 1091–1105.

Weber, A., C. J. Friedrich. 1929. *Alfred Weber's Theory of the Location of Industries*. The University of Chicago Press, Chicago, IL, USA.

Wentges, P. 1996. Accelerating Benders' decomposition for the capacitated facility location problem. *Mathematical Methods of Operations Research* **44**(2) 267–290.

Wesolowsky, G. O. 1973. Dynamic facility location. *Management Science* **19**(11) 1241–1248.

Wesolowsky, G. O., W. G. Truscott. 1975. The multiperiod location-allocation problem with relocation of facilities. *Management Science* **22**(1) 57–65.

Wesolowsky, G. O., W. G. Truscott. 1976. Dynamic location of multiple facilities for a class of continuous problems. *American Institute of Industrial Engineers Transactions* **8**(1) 76–83.

VITA

Joaquin Emmanuel Torres Soto received his Bachelor of Science degree in Industrial and Systems Engineering in 2001 and his Master of Engineering degree in Quality Systems and Productivity in 2003, both from Tecnológico de Monterrey Campus Chihuahua. He worked for three years as a research assistant, lecturer, and coordinator of the Research Department of the School of Business and Humanities at Tecnológico de Monterrey Campus Chihuahua. He received a scholarship from CONACYT (Consejo Nacional de Ciencia y Tecnología) in 2004 and joined the Industrial and Systems Engineering Department at Texas A&M University for doctoral studies. After receiving his Ph.D. he joined the faculty of Tecnológico de Monterrey. His research interest areas are discrete optimization, facility location, and logistics systems design. He is currently working on a joint research project with CONACYT and Tecnológico de Monterrey related to facility location in the public sector. Mr. Torres Soto may be reached at Tecnológico de Monterrey Campus Chihuahua, 4700 H. Colegio Militar, Col. Nombre de Dios, Chihuahua, Chih., México 31050. His email address is joaquin.torres@itesm.mx.