UNIVERSITY of SOUTHAMPTON

FACULTY of BUSINESS, LAW and ART

SOUTHAMPTON BUSINESS SCHOOL

General Modelling and Exact Solution Techniques for Restricted Facility Location Problems

by

Murat Oğuz

Thesis for the degree of Doctor of Philosophy in Management Science

January 2017

Declaration of Authorship

I, Murat Oğuz, declare that this thesis titled, 'General Modelling and Exact Solution Techniques for Restricted Facility Location Problems' and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: ____

Date: ____

UNIVERSITY of SOUTHAMPTON

Abstract

Faculty Name
FACULTY of BUSINESS, LAW and ART

Doctor of Philosophy

General Modelling and Exact Solution Techniques for Restricted Facility Location Problems

by Murat Oğuz

The aim of this thesis is to devise new optimisation techniques for restricted continuous facility location problems. In this work, forbidden regions and barriers are considered as the main restriction types for locations.

In the first research paper, a general modelling framework for restricted facility location problems is introduced with arbitrarily shaped forbidden regions or barriers. Phi-objects are canonically closed sets of points and efficient tools in mathematical modelling of 2D and 3D geometric optimisation problems. Point sets that contain isolated points, points that are removed from an object (deleted points) and objects with self-intersection of their frontiers are not classified as phi-objects. In the first research paper, the phi-objects are used to model real objects mathematically. It is shown that the proposed modelling framework can be applied to both median and center facility location problems, either with barriers or forbidden regions. The instances from the existing literature for this class of problems are solved to optimality using the new framework. Further, a realistic multi-facility problem instance derived from an archipelago vulnerable to earthquakes is also solved to optimality. This problem instance is significantly more complex than any other instance described in the literature.

In the second research paper, a new formulation based on multi-commodity flows with unknown destination is described and adapted to the problem type at hand. The proposed model is defined on a discretised space and this discretisation technique can also be applied to deterministic and stochastic continuous restricted location problems using any distance metric. As a solution method for the presented formulation, a solution algorithm, based on Benders Decomposition, is developed to take advantage of the discrete network structure. Extensive computational experiments, which are derived from

a well-known and complex instance from the literature are carried out, on both deterministic and stochastic multi-facility restricted location problems. Results are analyzed to evaluate the performance of the proposed solution technique.

In the third research paper, capacitated versions of the restricted facility location problems are studied. To the best of our knowledge, these have not been studied before. Similar to the second research paper, the continuous space is discretised and a model developed on a discrete space. Various acceleration techniques for Benders Decomposition are tested to improve the efficiency of solving the problem instances. A large number of deterministic instances is generated from three core instances. Two of these are well-known instances in the literature. The other instance is newly proposed in this paper. Numerical results are presented to determine the most efficient solution approach for the approximated continuous problem.

Acknowledgements

I would like to express my appreciation gratitude to my research advisory team, Prof Tolga Bektaş and Prof Julia Bennell for their valuable support, encouragement, indulgence and patience throughout my PhD studies. The most important factors in the accomplishment of this study were critical guidance and advice. I am also very grateful to Prof Jörg Fliege for his critical advice during my study. I also want to thank Dr Antonio Martinez-Sykora for his assistance with coding.

I especially thank Prof Tolga Bektaş for his time, effort and valuable suggestions that are the most significant factors to complete my thesis. During our meetings, he gave special attention to understand the underlying structure of the models and the solution algorithms presented in the thesis. I am also grateful for his advice on the formal writing and the structure of the thesis.

I would also like to specially thank Prof Julia Bennell for her kind help during my PhD journey, especially on the concepts of Phi-functions and Phi-objects, which I was not familiar with at first. Without her guidance, I could not have completed my first published paper.

I am also thankful to my fellow CORMSIS students and staff; I will never forget our discussions and the way we motivate each other. For their incredible encouragement, support and patience, I also want to thank all my friends.

Finally, I owe my deepest gratitude to my family. With their support and belief in me, a dream comes true.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

A vast number of organizations, regardless of their particular characteristics, have to make locational assessments, evaluations and decisions to satisfy organizational goals and objectives effectively. According to Handler and Mirchandani (1979), some of these decisions are locating fire stations, clinics, warehouses, aid stations, radar stations and computer centres. In deciding on a set of facility locations, it is important to define travel performance indices for measuring a system's performance such as average distance or response time. In this context, as a branch of operational research, facility location problems are concerned with the optimal positioning one or more facilities in order to optimise a general function that defines the performance of a system (Farahani et al. 2010).

To model facility location problems, there are five fundamental elements that have to be considered. These are the number of new facilities to be located, the number of demand points, the objective, the demand levels and space. If the aim of the problem is to locate one new facility, then the problem is named a single-facility location problem. If the aim of the problem is to locate several facilities, then this gives rise to the multi-facility location problem. In multi-facility location problems, in addition to location of the new facilities, decisions have to be made on the allocation of customers to new facilities. There can be various objectives in facility location problems such as minimising distance, travel time, and operating cost or maximising responsiveness. Among these objectives, the objective of the median problem and the objective of the centre problem are the most intensely studied objectives in the facility location literature. In median problems, the objective is to minimise the sum of weighted distances between demand points and facilities, whereas in the centre problem, the objective is to minimise the maximum weighted distance between demand points and the new facilities. Mostly in the literature, median problems are used in private sector applications, since the objective is to minimise total cost. On the other hand, centre problems are more suitable for public sector problems, since the objective is to minimise the maximum distance from demand points to new facilities. Consider the location of an emergency aid centre. The aim of the aid centre is to send relief items as soon as possible to all demand points without seeking a profit after a possible natural disaster (Toregas et al. 1971).

Considering the another fundamental element, space, Arabani and Farahani (2012) define three categories, which are discrete facility location problems, network facility location problems and continuous facility location problems. In discrete facility location problems, a discrete set of demand points and potential sites for new facilities are given. The aim is to find the locations of new facilities among the given set of potential sites in order to optimise the objective function of the problem. Network facility location problems are similar to discrete facility location problems, in the sense that the space in both problems is discretised. A network, which is a combination of arcs and nodes, has to be set up to define the space of the problem. Generally the demand arises on the nodes of the network, but in some cases, it may occur on the arcs as well. Besides, the new facilities can be located on anywhere along an arc or a node. In other words, the network location is not strictly discrete. As distinct from discrete and network facility location problems, in continuous facility location problems, the new facilities can be located anywhere within a defined area and demand is treated as continuous by using a density function to describe it. In this thesis, a continuous demand area is aggregated into a point, such as the central location of the demand area. Another important point that has to be considered in continuous facility location problems is the distance that defines relative positions of the new facilities to fixed points. Generally, in these problems l_p metrices and block distances are considered as distance measures.

There are various applications that can be modeled as facility location problems. Other than classical examples, such as location of stations, warehouses, public places, depots, emergency services and health services, there are also examples that are less obvious, but still have same logic in terms of finding the best locations for objects. Some of these less obvious examples are locations of satellite orbits (Drezner 1988), telecommunication switching centres (Hakimi 1965), rain gauges (Hogan 1990) and forest harvesting sites (Hodgson et al. 1987).

When dealing with real life location instances, it is possible to face some restrictions on the location of the new facilities or on the paths among the facilities. For example, consider a national park, where siting a new facility is prohibited. In such case, a new facility must be opened in some other place. In another instance, there can be a mountain in the study area, which may affect the travel path between demand points and a new facility. Other than these known obstacles for location and travelling, sometimes there can be unpredictable obstacles that arise due to a natural disaster or some other unpredicted phenomena. The problems involving prohibitions on the location of new facilities are generally named as *restricted facility location problems*. The existence of restricted regions on the space increases the complexity of such problems. There are three types of restricted regions: forbidden regions, barriers and congested regions, according to Canbolat and Wesolowsky (2010). The placement of new facilities within these restricted regions is not allowed, but traveling through them depends on the restriction type. In the case of barriers, travelling through is strictly restricted, whereas in congested regions travelling within is not strictly restricted, it is allowed in return of a penalty. In forbidden regions, traveling within the regions is allowed. In the literature, the considered shapes of restricted regions are mostly limited to circles and polygons. Aneja and Parlar (1994), Hamacher and Nickel (1994), Hamacher and Schöbel (1997), McGarvey and Cavalier (2003), and Bischoff and Klamroth (2007) are the main papers that consider polygons as the restricted regions, whereas Katz and Cooper (1981) considers a circle as the restricted region. A comprehensive literature review on these problems will be presented in the following chapter.

In this research, the main focus is on two restriction types: forbidden regions and barriers, but the presented models can easily be adapted to problems with congested regions by a modification in the objective functions. Protected areas, which have cultural value, are an example of forbidden regions. Lakes, parks, cemeteries and rivers are examples of barriers and a traffic jam in a metropolis is an example of congested region. The real life restriction type examples are illustrated in Figure 1.1. As well as these kinds of static restricted regions, there are probabilistic restricted regions in real life as well. Railroads, random accidents, construction and natural disasters may create temporary restricted regions, which is uncertain in length of time, in restricted region shape and in the total effect on the system.

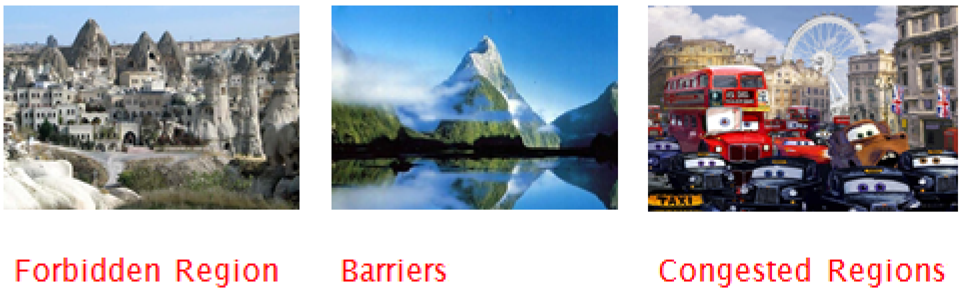


Figure 1.1: Examples of restriction types

As these problems are common in real life situations, it is necessary to develop further research aiming to find effective solutions. In order to meet this need, both the private and public sector must ensure that future facilities are located in feasible regions. Additionally, thorough consideration of potential restricted regions that could prevent transportation of goods from these future facilities to demand points must be done. Taking into account these two requirements, the locations of new facilities can be determined, whereby the solutions are optimal with respect to all features of possible application. For example, if the application is for the scope of a private organization, an optimal solution would be to build the facility in such a place that revenue could be maximised by transferring goods as efficiently as possible and cutting future expenses to meet customer demand. Similarly, these problems are also important for public sector applications. For instance, in the event of a natural disaster or accident, the facility locations must be able to meet demand for relief items as efficiently as possible, even when considering the most remote locations.

It must be noted that previous research in this subject focuses on achieving specific targets and not necessarily considering a perspective that could have universal application. Studies have attempted to provide solutions for this problem based on certain distance matrices, objectives, and restricted region shapes, for example see Butt and Cavalier (1996), Butt and Cavalier (1997), Klamroth (2004) and Bischoff et al. (2009). Based on this insight, our aim in this research is to put forth a general approach that could have a universal application with optimal or high quality approximated solutions given different distance metrics, restriction shapes and objective functions. This would enable organizational decision makers to devise solutions that are general and flexible in a wide possible range of restricted facility locational problems. As rapid urbanization continues and the uncertainty of natural disasters remains, the relevancy of the problem will continue to grow over time.

1.2 Research Aim and Objectives

The PhD thesis aims to solve uncapacitated and capacitated facility location problems to optimality that have, different characteristics such as distance, restricted region shape, number of new facilities and objective function. Both deterministic and stochastic versions are considered. The thesis aims to provide the groundwork for developing general modelling and solution techniques in relevant facility location problems.

The research objectives are:

- To introduce a general modelling framework for restricted continuous facility location problems by using the concept of phi-objects, which is mainly considered in Cutting and Packing field,
- To transform the continuous space of the problem into a discrete, multi-commodity flow network problem to obtain a linearised version of the introduced general modelling framework,
- To adapt stochastic programming methods for solving the restricted facility location problems with probabilistic barriers and solve them to optimality by describing a state-of-the-art exact algorithm based on Benders decomposition.
- To model and solve capacitated restricted facility location problems by using Benders decomposition and to describe and test various algorithmic enhancements for this algorithm.

1.3 Structure and Content of the Thesis

In the first research paper, a general modelling framework for restricted facility location problems is developed by using the concept of phi-objects from the cutting and packing literature (Bennell et al. 2010). Phi-objects are closed sets of points that are used to model real objects geometrically. Functions can be defined to describe an interaction between a point and phi-objects, which can be used to design a continuous feasible region for potential facility sites that does not have intersections with the restricted regions on the continuous plane. By introducing these functions, it is possible to formulate a general modelling framework for restricted facility location problems. These models can be in the form of mixed integer linear programming (MILP) or mixed integer non-linear programming (MINLP) formulations, depending on the considered distance functions and the functions used to model the restricted regions.

In the second research paper, the continuous space of the problem is discretised by designing a network based on the idea of multi-commodity network design problem. Multi-commodity network design problem searches for the right flows of commodities from their origin nodes to desired destination nodes with a minimum total cost of routing the flows on a set arcs. To adapt this idea to facility location problems, a formulation is developed for a problem with unknown destinations. Hence, the unknown destinations will be the locations of the new facility sites. Thanks to the integrality theorem, the newly introduced model is in the form of LP and the linear nature of the model provides an advantage in terms of performance to deal with large size stochastic multi-facility problems with restricted regions. An exact algorithm based on Benders decomposition is also proposed to solve the presented models to optimality in an efficient manner.

In a third research paper, a network is designed to discretise a continuous capacitated restricted facility location problem. What differs from the second research paper is that the integrality property does not hold because of the capacity constraint, which results in a MILP formulation. Modified versions of Benders decomposition algorithm are presented to solve the problem at hand. The main motivation of the third research paper is to find the highest performance algorithm based on Benders decomposition for the capacitated restricted facility location problem.

Table 1.1: Mathematical model structures

Chapters	LP	MILP	NLP	MINLP
Chapter 2		✓	✓	✓
Chapter 3	✓			
Chapter 4		✓		

Table 1.1 shows which mathematical model structures are used in Chapters 2, 3 and 4. As can be seen in Table 1.1, 4 different mathematical programming classes are dealt with in this thesis.

The remainder of the thesis is organised as follows. The first, second and third research papers are presented Chapter 2, 3 and 4, respectively, including relevant literature review. Conclusions are given in Chapter 5.

Chapter 2

A Modelling Framework for Solving Restricted Planar Location Problems Using Phi-Objects

Abstract

This paper presents a general modelling framework for restricted facility location problems with arbitrarily shaped forbidden regions or barriers, where regions are modeled using phi-objects. Phi-objects are an efficient tool in mathematical modelling of 2D and 3D geometric optimisation problems and are widely used in cutting and packing problems and covering problems. The paper shows that the proposed modelling framework can be applied to both median and center facility location problems, either with barriers or forbidden regions. The resulting models are either mixed-integer linear or nonlinear programming formulations, depending on the shape of the restricted region and the considered distance measure. Using the new framework, various instances from the existing literature for this class of problems are solved to optimality. The paper also introduces and optimally solves a realistic multi-facility problem instance derived from an archipelago vulnerable to earthquakes. This problem instance is significantly more complex than any other instance described in the literature.
Keywords: mathematical modelling, facility location, phi-objects

2.1 Introduction

Facility location problems are concerned with the optimal placement of facilities in order to minimise a general cost function including that of distance and demand from customer sites. If the underlying space for both the potential facility sites and the customer sites is continuous, such problems are termed continuous facility location problems. These problems are different from their discrete counterparts, where facilities can only be placed on a finite set of candidate sites. Dasci and Verter (2001) state that discrete location models may provide optimal solutions, but when a model becomes more realistic, data and the computational requirements increase significantly, which is likely to result in a decrease in model accuracy.

The most commonly studied continuous facility location problems are the minisum and the minimax problems (Drezner 1995). The former corresponds to minimising the total weighted distance from customer sites, while the latter is concerned with minimising the maximum weighted distance. Minisum problems with p facilities are often referred to as p-median problems; minimax problems with p facilities are called p-center problems. When real life instances are studied, however, these problems might fail to model the problem in a realistic manner, since there can be geographical restrictions on the location of the facilities or on the paths to the facilities. For instance, a huge mountain range can hinder facility location and transportation. These mountain ranges can be seen as a restriction on location and transportation between facilities.

The restriction on locations presents a challenge for these classes of problems. According to Canbolat and Wesolowsky (2010), it is possible to classify the restriction types into three categories, namely forbidden regions, barriers and congested regions. The placement of new facilities within a forbidden region is not allowed, but traveling through it is permitted. For barriers, neither placement of facilities within or travel through is permitted. The placement of new facilities is also not permitted in congested regions, but traveling within may be allowed in return for a penalty. In this paper, I will focus on two planar facility location problem restrictions: one with forbidden regions, another with barriers. There are various applications of such types of problems. Assembly of printed circuit boards (Foulds and Hamacher 1994), obnoxious facility planning (Carrizosa and Plastria 1993) and location of emergency facilities are some examples for problems with forbidden regions (Hamacher and Nickel 1995), whereas urban applications considering lakes, parks, cemeteries and rivers can be listed as examples for problems with barriers.

The problem at hand is to find the optimal locations of new facilities on a continuous plane, considering the restrictions mentioned above. These restricted regions on the plane make the feasible region non-convex and arbitrarily shaped. To define a feasible region for this problem type, I use phi-objects, which are an efficient tool in mathematical modelling for 2D and 3D geometric optimisation problems. The concept of phi-objects is mainly associated with cutting and packing problems and covering problems. It lends itself to the problems addressed in this paper as they have the capability to model any arbitrary shaped region, including regions that are bounded by arcs, and as a result, naturally form a mathematical model. While they have the advantage of efficiently modelling shapes to a high level of fidelity, they can also be used where a lower level of fidelity is acceptable and take advantage of the computational efficiency offered by approximation of the shape. A model that has general applicability at any chosen level of fidelity is new to the literature. On top of the capability of modelling any arbitrary shaped region, using phi-objects enables us to model different types of restrictions within a single modelling framework, which would allow formulating and solving real-life instances featuring various types of restrictions and distance metrics. In this paper, I will show that it is possible to adapt this concept to continuous facility location problems with restricted regions and model arbitrarily shaped regions by using phi-objects. The main idea is to use phi-objects to create a general model for this class of problems, which can then be used to solve them to optimality. In this modelling approach, phi-objects can represent either restricted or feasible regions on the plane. I show that the resulting formulations are either mixed-integer linear programs (MILP) or mixed-integer nonlinear programs (MINLP), depending on the shape of the restricted region and the considered distance measure.

In addition to the models, I also present an application of a cutting plane technique to solve models with a significant number of barriers. This technique is used to relax models by decreasing the number of constraints significantly.

The contributions of this paper are listed below:

- A new modelling approach for continuous facility location problems with arbitrarily shaped forbidden regions or barriers is presented by using the concept of phi-objects. This concept enables us to deal with arcs as well as planar edges defining the region. It is shown that problems with different restriction types can be solved within a single modelling framework. The flexibility provided by phi-objects increases the variety of geometric objects that can be considered as restricted regions. Thus, real geographical shapes found in nature can be modeled with negligible approximation error.
- It is shown that the proposed general modelling framework can be adapted to multi-facility location problems with forbidden regions and to single facility location problems with barriers with various objective functions.
- I present computational results showing that various instances for this class of problem previously described in the literature can be solved to optimality by using the proposed approach.

The remainder of this paper is organised as follows. In Section 2.2, I review the relevant literature on continuous location problems with a particular focus on forbidden regions and barriers. Section 2.3 will present background information on phi-objects and illustrate the application of this concept to restricted continuous facility location problems with forbidden regions or barriers. I introduce the general modelling framework in Section 2.4. In Section 2.5, I test our approach by solving instances from the literature, as well as a new instance derived from a real geographical setting in Section 2.6. Conclusions are given in Section 2.7.

2.2 Literature Review

Restricted planar location problems can be classified into three categories, namely those with (i) forbidden regions, (ii) barriers and (iii) congested regions. In this paper, I focus on the restricted planar problems with forbidden regions and barriers. For a discussion of the congested region problem see Butt and Cavalier (1997) and Sarkar et al. (2004).

2.2.1 Restricted Problems with Forbidden Regions

To our knowledge, the first study on this problems is by Aneja and Parlar (1994), who describe algorithms for the single facility location problem with forbidden regions and barriers separately, with a metric defined on $\mathbb{R}^n$ and with respect to a parameter $1 \leq p \leq \infty$, namely the l_p metric, which has a general form represented as $[|x - x_i|^p + |y - y_i|^p]^{1/p}$. For $p = 1$, the l_p metric is known as the rectilinear or Manhattan distance. Similarly, l_2 is known as the Euclidean distance and l_∞ is the Chebyshev distance. The algorithms proposed by Aneja and Parlar (1994) are applicable to cases where $1 < p \leq 2$. For the forbidden region problem, they present algorithms for polygonal convex and non-convex forbidden regions. The algorithms are based on the premise that if an optimal solution of a problem solved by ignoring the forbidden region is located outside the forbidden region, then this solution would also be the optimal solution for the original problem with the forbidden region. If not, then they use an algorithm that assumes that the optimal solution would be on the boundary of the forbidden region.

Hamacher and Nickel (1994) introduce several algorithms for the 1-median problem on a plane using distance metrics including Manhattan, squared Euclidean and Chebyshev. Their assumption is that the considered forbidden region is a union of pairwise disjoint convex sets. They present some basic examples to test their algorithms. Hamacher and Nickel (1995) extend their previous work to multi-facility median and center problems, where the forbidden region is again a union of pairwise disjoint convex sets. They present a heuristic algorithm which consists of a sequential solution of p single facility problems for the p-median problem with a forbidden region, and an efficient solution algorithm based on level sets and lines for the 1-center problem with Manhattan and Chebyshev distance metrics.

Hamacher and Schöbel (1997) study a center problem with a polyhedral convex forbidden region. A polynomial solution algorithm is proposed, using the Euclidean distance metric. With few modifications, the algorithm can be used for a problem with a polyhedral non-convex forbidden region. Later, Woeginger (1998) shows that this problem can be solved with a better time complexity by applying standard computational geometry techniques.

2.2.2 Restricted Problems with Barriers

Restricted planar location problems with barriers, to the best of our knowledge, are first presented by Katz and Cooper (1981), where the barrier is a single circular region and the distance metric is Euclidean. In this paper, an algorithm is described to compute the shortest distance following, which the modified objective function with the shortest distance is converted into a sequence of unconstrained minimisation problems. Klamroth (2004) presents new structural results for the problem described by Katz and Cooper (1981). Larson and Sadiq (1983) solve a p-median problem in the case of rectilinear distance. A cell formation technique is used to reduce the p-median problem to a discrete search problem. The authors state that any of the existing algorithms available for the p-median problem can be applied to solve their problem.

Aneja and Parlar (1994) use the concept of visibility and the Dijkstra algorithm to compute the shortest distance between customer sites and the location of the new facility, using polygons as barriers. Butt and Cavalier (1996) consider the restricted 1-median problem with convex polygonal forbidden regions. They use the Euclidean distance metric and describe an iterative solution procedure.

Hamacher and Klamroth (2000) consider the distances defined by polyhedral norms in the restricted single facility median problem, with the barriers being convex polyhedral subsets. They use a grid construction method to prove a discretisation result, which implies a polynomial algorithm to solve location problems with barriers and block norms. Klamroth (2001) presents an exact algorithm and a heuristic solution procedure for the single facility problem with polyhedral barriers based on reducing this non-convex optimisation problem to a finite set of convex subproblems. Dearing et al. (2002) present a single facility center problem with polygonal barriers and considered rectilinear distance as a distance metric. A polynomial algorithm is developed to solve the problem to optimality. The same authors later adapt this modification model to a problem considering block distance (Dearing et al. 2005).

Dearing and Segars Jr. (2002a) develop a modification technique for barriers and prove that the objective function values for both the original problem and the modified problem are the same. This modification technique is based on properties of rectilinear distances. In a companion paper, the authors present a solution algorithm based on partitioning the feasible region into convex subsets by using the mentioned modification technique (Dearing and Segars Jr. 2002b).

McGarvey and Cavalier (2003) develop an iterative solution procedure based on a modified "Big Square Small Square" branch-and-bound method to solve a 1-facility median problem with convex polygonal barriers, where the distance metric is Euclidean. Bischoff and Klamroth (2007) reduce the original problem to a finite series of convex subproblems to solve them by using Weiszfeld algorithm to find a heuristic solution in the case of Euclidean distances. The visibility arguments are also taken into consideration to decrease the number of convex subproblems.

Recent research on restricted facility location problems have considered stochastic features, for example Canbolat and Wesolowsky (2010), Amiri-Aref et al. (2011), Shiripour et al. (2012), Amiri-Aref et al. (2013) and Javadian et al. (2014), where the location of the barriers are uncertain.

2.2.3 Restricted Problems with Forbidden Regions and Barriers

To the best of our knowledge, the only paper that considers a facility planar location problem with both forbidden regions and barriers is that by Batta et al. (1989). It is a p-median problem in the presence of arbitrarily shaped barriers and convex forbidden regions. The distance metric is Manhattan (rectilinear). They use and extend a cell formation technique, introduced by Larson and Sadiq (1983), for solving the problem.

2.2.4 Discussion

As the preceding review shows, it is possible to categorise restricted location problems based on the types of restriction. While each problem has so far been individually addressed, there is a lack of a general approach that is able to collectively address all problems in each category above. Table 2.1 details the types of problems addressed in the literature and the applicability of the published approaches to these problems. With respect to the shapes of restricted regions, previous studies limit the shape of the restricted regions to polygonal. These shapes can substantially vary type in real world geographical problems, however. Finally, previous studies only focus on specific objectives (e.g. median problem, center problem). A general approach that can deal with various objectives is needed. To address the need for such an approach, I present a modelling framework to formulate problems with various restriction categories, objectives, distance metrics, and arbitrarily shaped restricted regions, such that the models can efficiently be solved to optimality using existing software. Before doing so, I first describe how regions are represented using phi-objects in the next section.

2.3 Phi-objects for Modelling Forbidden Regions

This section describes the concept of phi-objects. Phi-objects are most commonly found in the cutting and packing literature. They are used to model real objects mathematically. These geometric objects are canonically closed sets of points. Point sets that contain isolated points, points that are removed from an object (deleted points) and objects with self-intersection of their frontiers are not classified as phi-objects. Note that

Table 2.1: Restricted location problems literature overview

	Restriction Type		Distance Metric			Objective		Restricted Region Shape		
	Forbidden Region	Barrier	Rectilinear	Euclidean	Chebyshev	Median	Center	Polygon	Circular	Arbitrary
Katz and Cooper (1981)		✓		✓		✓			✓	
Larson and Sadiq (1983)		✓	✓			✓				✓
Aneja and Parlar (1994)	✓	✓		✓		✓		✓		
Hamacher and Nickel (1994)	✓		✓	✓	✓	✓		✓		
Hamacher and Nickel (1995)	✓		✓	✓	✓	✓	✓	✓		
Butt and Cavalier (1996)		✓		✓		✓		✓		
Hamacher and Schöbel (1997)	✓			✓			✓	✓		
Woeginger (1998)	✓			✓			✓	✓		
Klamroth (2001)		✓	✓	✓	✓	✓	✓	✓		
Dearing et al. (2002)		✓	✓				✓	✓		
Dearing and Segars Jr. (2002a)		✓	✓			✓	✓			✓
Dearing and Segars Jr. (2002b)		✓	✓			✓	✓			✓
McGarvey and Cavalier (2003)		✓		✓		✓		✓		
Klamroth (2004)		✓		✓		✓			✓	
Bischoff and Klamroth (2007)		✓		✓		✓		✓		
This paper	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

in the cutting and packing literature, the concept extends to generating the convolution of objects, while here I simply use the objects in their original form.

The example presented in Figure 2.1a is a phi-object, formed by a closed set of points. The example shown in Figure 2.1b is not a phi-object, since it self-intersects with its frontier. Similarly, the example in Figure 2.1c contains deleted points and is not classified as a phi-object. Readers are referred to Bennell et al. (2010) for further details on phi-objects. In this paper, phi-objects are used as geometrical tools to model complex geographical formations such as lakes, archipelagos and bays which often give rise to forbidden regions or barriers.

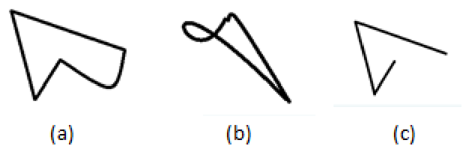


Figure 2.1: Examples of arbitrarily shaped 2D objects

To model a complex shaped geometric object, it is convenient to divide phi-objects into three categories, namely primitive, basic and composed objects (Bennell et al. 2015). A half-plane, a circle and the complement of a circle (circular hole) are types of primitive objects, as shown in Figure 2.2. The representation of basic objects is through intersections of a finite number of primitive objects, whereas a composed object is obtained by unions of a finite number of basic objects. Figure 2.3 shows four examples of basic objects.

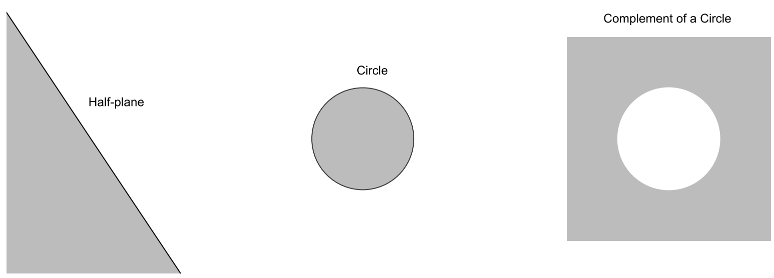


Figure 2.2: Primitive object types

In Figure 2.3a, the basic object is obtained by the intersection of four half-planes. The non-convex object shown in Figure 2.3b is the intersection of a circle, the complement of another circle and a half plane. In Figure 2.3c, the basic object is an intersection of two

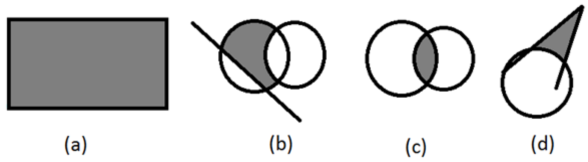


FIGURE 2.3: Examples of basic objects

circles. The basic object presented in Figure 2.3d is the intersection of two half-planes and the complement of a circle.

Figure 2.4 shows the union of the four basic objects that are presented in Figure 2.3. This arbitrarily shaped non-convex shape is a composed object.

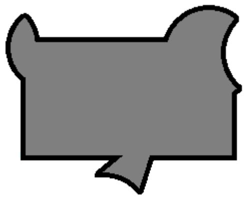


FIGURE 2.4: A composed object

More formally, let B be the index set of basic objects and P be the index set of primitive objects. Each basic object B_t, $t \in \mathsf{B}$, is defined with respect to an index set $\mathsf{P}_t \subseteq \mathsf{P}$ of primitive objects, where each primitive object is denoted by P_k. Thus, B_t can be written as

$$B_t = \bigcap_{k \in \mathsf{P}_t} P_k \quad \forall t \in \mathsf{B}. \tag{2.1}$$

Let R be the index set of composed objects. For each composed object R_i, $i \in \mathsf{R}$, let $\mathsf{B}_i \subseteq \mathsf{B}$ denote the index set of basic objects defining this region. A composed object R_i is then represented by

$$R_i = \bigcup_{k \in \mathsf{B}_i} B_k \quad \forall i \in \mathsf{R}. \tag{2.2}$$

Functions can be defined for phi-objects to describe the interaction between a point and phi-object. These functions allow the arrangement of points, (in our case facility locations), with respect to phi-objects, (in our case forbidden regions or barriers), in order to form an objective for a mathematical model. Each primitive object P_k is associated with a half-plane, circle or the complement of a circle. Let $f_k(X) \leq 0$, where X is a point with a coordinate (x, y) and $f_k(X)$ is a function defining the boundary of the primitive object in such a way that it is less than 0 if X is inside the phi-object. Since basic object B_t is defined by an intersection of primitive objects, the maximum of $f_k(X)$, $k \in P_t$ is taken. Therefore, the function $\gamma_t(X, B_t)$ representing the relationship between point X and the basic object B_t is

$$\gamma_t(X, B_t) = \max_{k \in \mathsf{P}_t} \{f_k(X)\}\,. \tag{2.3}$$

If $\gamma_t(X, B_t) = 0$, then the point X is on the boundary of B_t. Alternatively, if $\gamma_t(X, B_t) > 0$, then the point X is outside the basic object B_t.

Similarly, since a composed object R_i is a union of basic objects B_k, $k \in \mathbf{B}_i$, it is defined by the minimum of $\gamma_k(X, B_t)$. Therefore, the function $\Gamma_i(X, R_i)$ representing the relationship between point X and the composed object R_i is

$$\Gamma_i(X, R_i) = \min_{k \in \mathsf{B}_i} \{\gamma_k(X, B_t)\}\,. \tag{2.4}$$

If $\Gamma_i(X, R_i) = 0$, then the point X is on the boundary of R_i. Alternatively, if $\Gamma_i(X, R_i) > 0$, then the point X is outside the composed object R_i.

The above mathematical model is based on the set of primitive and basic phi-objects. These objects are known and described in Bennell et al. (2010) and Bennell et al. (2015). In order to model an arbitrary object, the object needs to be decomposed into basic and primitive objects. Chernov et al. (2012) provides a decomposition algorithm for this purpose. They define four types of basic objects. These are (i) a convex polygon (Figure 2.5a), formed by an intersection of three or more half-planes, (ii) a circular segment (Figure 2.5b), as an intersection of a circle and a half-plane, (iii) a hat (Figure 2.5c), formed by an intersection of a complement of a circle and two half-planes whose corresponding boundaries are tangent to the circle, and (iv) a horn (Figure 2.5d), an intersection of a circle, a complement of a circle and a half-plane. Chernov et al. (2012) showed that any arbitrarily shaped composed object bounded by circular arcs and line segments can automatically be decomposed into a set of defined basic objects by the given algorithm.

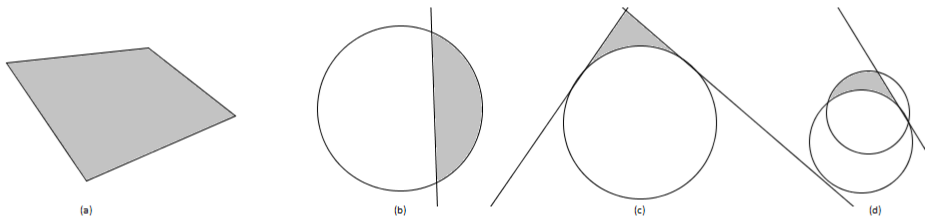


Figure 2.5: Four types of basic objects (Chernov et al. 2012)

These functions will be used in the proposed modelling framework to define the feasible regions for the possible locations of new facilities. The next section describes the framework in greater detail.

2.4 A General Modelling Framework

In this section, I present a general modelling framework to formulate continuous facility location problems with restricted regions where regions are modeled using the concept of phi-objects.

2.4.1 Restricted Problems with Forbidden Regions

By using the functions defined above, it is possible to develop a general model for problems with forbidden regions in which locating a facility is forbidden but traveling through is permitted. Let $\mathbf{M}$ be the index set of customer sites and $\mathbf{N}$ be the index set of new facilities, where $X_1, ..., X_n$ are the locations for new facilities with coordinates $(x_1, y_1), ..., (x_n, y_n)$ respectively. Similarly, $\bar{X}_1, ..., \bar{X}_m$ are locations for customer sites with coordinates $(\bar{x}_1, \bar{y}_1), ..., (\bar{x}_m, \bar{y}_m)$, respectively, and $d(X_n, \bar{X}_m)$ is the distance between nth new facility and mth customer site. Let $\Gamma_i(X_n, R_i)$ denote the function defining the relationship between a new facility X_n and a region R_i. A binary variable z_{mn} is defined, which is equal to 1 if point $\bar{X}_m$ is assigned to a new facility X_n, 0 otherwise, and $Z = \{z_{mn} \in \{0,1\}, m \in \mathbf{M}, n \in \mathbf{N}\}$. The general modelling framework of the continuous facility location problem with a forbidden region is shown below.

$$\underset{Z,X_n}{\text{minimise}}\, F\left(\left(z_{mn}, d(X_n, \bar{X}_m)\right)_{n\in\mathbf{N}, m\in\mathbf{M}}\right) \tag{2.5}$$

subject to

$$\Gamma_i(X_n, R_i) \geq 0 \qquad \forall i \in \mathbf{R}, n \in \mathbf{N} \tag{2.6}$$

$$\sum_{n\in\mathbf{N}} z_{mn} = 1 \qquad \forall m \in \mathbf{M} \tag{2.7}$$

$$z_{mn} \in \{0,1\} \qquad \forall m \in \mathbf{M}, n \in \mathbf{N}. \tag{2.8}$$

In constraints (2.6), " $\geq$ " is used because locating a new facility is not permitted in region R_i. If phi-objects represent the feasible region on a plane, then constraints (2.6) should read $\Gamma_i(X_n, R_i) \leq 0$.

There can be various objectives in facility location problems such as minimising distance, time, operating cost or maximising responsiveness. The objective of the proposed modelling framework assumes minimisation of a weighted distance and not the distance itself. A weighted distance is a general term that can represent any of the measures listed above. As an example, if one wishes to minimise time and time is proportional to distance, then the objective can be modified with suitable weights to reflect this relationship. The following are two of the most commonly studied global objective functions, both of which can easily be included in the proposed modelling framework:

1. F can be decomposed into a sum of one-dimensional functions, i.e. $F\left(\left(z_{mn}, d(X_n, \bar{X}_m)\right)_{n\in\mathbf{N}, m\in\mathbf{M}}\right) = \sum_{n\in\mathbf{N}}\sum_{m\in\mathbf{M}} z_{mn}w_m d(X_n, \bar{X}_m)$, as in the case of the standard median problem, where w_m are non-negative weights for each customer site $m \in \mathbf{M}$ (Akyüz et al. 2010). Median problems often arise in the private sector, since the objective is to minimise total cost. Several applications for this objective are in industrial transportation and telecommunications.

2. F is the maximum of one-dimensional functions, i.e. $F\left(\left(z_{mn}, d(X_n, \bar{X}_m)\right)_{n\in\mathbf{N}, m\in\mathbf{M}}\right) = \max_{\substack{m\in\mathbf{M}\\ n\in\mathbf{N}}} z_{mn}w_m d(X_n, \bar{X}_m)$, as is the objective function of the well-known center problem (Drezner 1984). Center problems are usually more suitable for modelling public sector problems, since the objective is to minimise the maximum weighted distance from demand points to new facilities. An example would be locating an emergency aid center from where relief items would be sent to all demand points as soon as possible following a possible natural disaster.

In our modelling framework, the function (2.4) that appears in the constraint set includes min and max operators, which introduces a non-linearity. The way to deal with this non-linearity is to represent the function by a series of individual constraints using an either-or representation. While this representation in turn may include linear or non-linear expressions, it is convenient to use this transformation when using off-the-shelf solvers that cannot deal with min and max operators. Let us first look at the case where phi-objects represent forbidden regions and where the new facilities can be located on the boundary of or outside phi-objects. In this case the constraint set (2.6) is transformed into (Schouwenaars et al. 2001):

$$f_k(X_n) \geq 0 - Mv_{knt} \qquad \forall k \in \mathbf{P}_t, n \in \mathbf{N}, t \in \mathbf{B} \tag{2.9}$$

$$\sum_{k=1}^{K_t} v_{knt} \leq K_t - 1 \qquad \forall n \in \mathbf{N}, t \in \mathbf{B}, \tag{2.10}$$

where K_t is the total number of elements in set $\mathbf{P}_t$. Here v_{knt} is a binary variable equal to 0 if the constraint associated with a $k \in \mathbf{P}_t, n \in \mathbf{N}$ and $t \in \mathbf{B}$ is binding, 1 otherwise. M is a sufficiently large positive number. In this manner at least one corresponding binary variable will take the value 0 for each new facility and basic object. In other words, through constraint (2.9), at least one of the functions forming the basic object B_t will be non-negative, which means the maximum of these functions will be non-negative.

I now look at the case where the phi-object represents a feasible region and outside of the region is forbidden. In this case, the new facilities can be located on the boundary or inside of the phi-object, and the constraint set

$$\Gamma_{ni}(X_n, R_i) \leq 0 \quad \forall i \in \mathbf{R}, n \in \mathbf{N}, \tag{2.11}$$

can be converted into

$$f_k(X_n) \leq 0 + M v_{nt} \qquad \forall k \in \mathbf{P}_t, n \in \mathbf{N}, t \in \mathbf{B} \tag{2.12}$$

$$\sum_{t=1}^{T} v_{nt} \leq T - 1 \qquad \forall n \in \mathbf{N}, \tag{2.13}$$

where T is the total number of elements in set $\mathbf{B}$. Here v_{nt} is a binary variable defined for a basic object B_t and a new facility X_n, and is equal to 1 if the new facility is not contained within B_t, 0 otherwise. Constraints (2.12) ensure that at least each member of a group of functions forming one of the basic objects B_t will be nonpositive. Some of the other members of other function groups that are forming other basic objects may be positive. Therefore, at least one of the maximums will be non-positive, which implies that the minimum of the maximums will be non-positive.

As for M, it is important to define a lower bound for this number to be able to decrease the value as much as possible, since large values for M can have a negative effect on algorithm performance. In the following two cases, a lower bound value of M can be determined. In the first case where phi-objects represent forbidden regions, I define the smallest rectangle Q that contains all forbidden regions and customer sites within it or on its boundary. Note that the locations of the new facilities must also be inside or on the boundary of Q. In this case,

$$M \geq \max_{X_n \in Q, \forall n \in \mathbf{N}, k \in \mathbf{P}_t} \{|f_k(X_n)|\}. \tag{2.14}$$

In the second case where phi-objects represent feasible regions, a rectangle is not necessary as in the previous case since the locations of the new facilities will be inside or on the boundary of the phi-objects $R_i, \forall i$. Therefore, a lower bound of M can be defined as,

$$M \geq \max_{X_n \in \bigcup\limits_{i \in \mathbf{R}} R_i, \forall n \in \mathbf{N}, k \in \mathbf{P}_t} \{|f_k(X_n)|\}. \tag{2.15}$$

It is also possible to tailor the M values by indexing them. Inequalities (2.16) and

(2.17) present the lower bounds for $M_{nk}, \forall n \in \mathbf{N}, k \in \mathbf{P}_t$ for the cases where phi-objects represent forbidden regions and feasible regions, respectively.

$$M_{nk} \geq |f_k(X_n)| \quad X_n \in Q, \forall n \in \mathbf{N}, k \in \mathbf{P}_t. \tag{2.16}$$

$$M_{nk} \geq |f_k(X_n)| \quad X_n \in \bigcup_{i \in \mathsf{R}} R_i, \forall n \in \mathbf{N}, k \in \mathbf{P}_t. \tag{2.17}$$

After these modifications in the constraint set, it is possible to produce models for both multi-facility median and center problems with an arbitrarily shaped forbidden region and the relevant distance metric. These models can be in the form of mixed integer linear programming (MILP) or mixed integer non-linear programming (MINLP) formulations, depending on the objective function and the functions used to model the forbidden region. For instance, if the distance metric is rectilinear and the forbidden region is a convex polygon, then the resulting model is a MILP, whereas if the distance metric is Euclidean, then the model will be in the form of a MINLP due to the non-linearity in the objective function.

2.4.2 Restricted Single Facility Problems with Barriers

The modelling framework for single facility location problems with barriers is very similar to that of forbidden regions. The main challenge in modelling these problems is the requirement that the paths between customer sites and the new facility must lie outside the barriers. To overcome this difficulty, a large number of points have to be defined for each path connecting each customer site with the new facility. These must all lie outside the barrier, to ensure that the paths themselves will be outside the barriers. To illustrate, I provide an example in Figure 2.6, which shows the shortest path between points A and B given the existence of a rectangular barrier. Here, the path between A and B is formed by 100 points, which are all restricted to be outside of the rectangle. This idea can clearly be adapted to a facility location problem with barriers as explained below.

Let $\mathbf{J} = \{1, 2, ..., J_{max}\}$ be the index set of the points generating the paths, and $\bar{X}_{mj}$ be the location of the jth point on the path starting from the mth customer site and ending at the location of the new facility. The general modelling framework of the single facility median problem with barriers is shown below.

$$\underset{X, \bar{X}_{mj}}{\text{minimise}}\ F\left(\left(d(\bar{X}_{mj-1}, \bar{X}_{mj})\right)_{m \in \mathbf{M}, j \in \mathbf{J} \setminus \{0\}}\right) \tag{2.18}$$

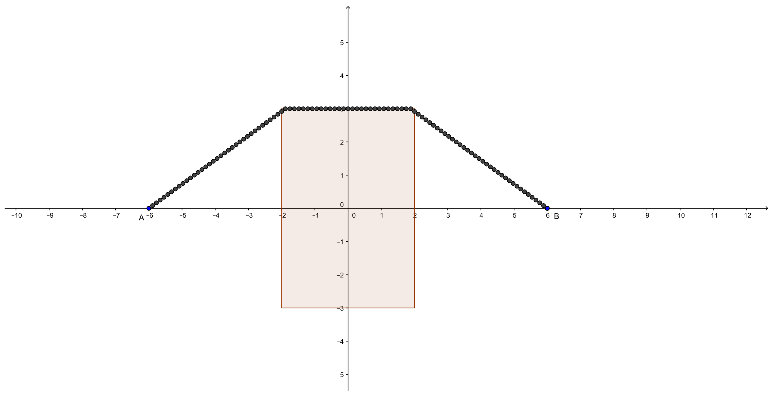


FIGURE 2.6: Shortest path between points A and B

subject to

$$d(\bar{X}_{mj-1}, \bar{X}_{mj}) \leq V t_{max} \quad \forall m \in \mathbf{M}, j \in \mathbf{J} \setminus \{0\} \tag{2.19}$$

$$\Gamma_{mji}(\bar{X}_{mj}, R_i) \geq 0 \quad \forall i \in \mathbf{R}, m \in \mathbf{M}, j \in \mathbf{J} \tag{2.20}$$

$$X = \bar{X}_{mj} \quad \forall m \in \mathbf{M}, j = J_{max}. \tag{2.21}$$

The objective function (2.18) minimises a function of distances between each point on each path until the last point, which is the location of the new facility. $F\left(\left(d(\bar{X}_{mj-1}, \bar{X}_{mj})\right)_{m \in \mathbf{M}, j \in \mathbf{J} \setminus \{0\}}\right)$ will be equal to $\sum_{m \in \mathbf{M}} \sum_{j \in \mathbf{J} \setminus \{0\}} w_m d(\bar{X}_{mj-1}, \bar{X}_{mj})$ for single facility median problems and equal to $\max_{m \in \mathbf{M}} \sum_{j \in \mathbf{J} \setminus \{0\}} w_m d(\bar{X}_{mj-1}, \bar{X}_{mj})$ for single facility center problems. Constraints (2.19) guarantee that the travel time from a point j on a path to point $j+1$ with a constant velocity V cannot be more time than a specified limit t_{max}. This constraint therefore imposes an upper bound on the distance between any two consecutive points on a path. Setting an upper bound for each distance reduces the chance that the segment between any 2 points will pass through the barrier. Constraints (2.20) state that each point on each path between the customer sites and the new facility must not lie in an object that defines a barrier. Constraints (2.21) require that the last point of each path must be the same as the location of the new facility. In our notation, I denote by $\bar{X}_{mj}$, $j = 0$ and $\forall m \in \mathbf{M}$, as the location of each customer site.

2.5 Results for Instances from the Literature

In this section, instances of restricted continuous facility location problems previously described in the literature are modeled with the proposed general modelling framework. Table 2.2 shows the properties of these instances and where they were published. All instances are solved on a PC with an Intel(R) Core(TM) 2.60 GHz processor and 4.00

GB of RAM using CPLEX 12.5.1.0 for MILP, BONMIN 1.7 for MINLP, and IPOPT 3.11 for NLP. For the instances with barriers, the optimisation package, IPOPT 3.11, that can automatically handle the chain of max/min operators in the constraints without the need of using big Ms, is used to solve them to optimality.

TABLE 2.2: Instances

Source	Problem Type	Restriction Type	Restricted Region Type	Distance Measure
Katz and Cooper (1981)	1-median	Barrier	Circle	Euclidean
Katz and Cooper (1981)	1-median	Barrier	Circle	Euclidean
Aneja and Parlar (1994)	1-median	Forbidden Region	Polygonal	Euclidean
Aneja and Parlar (1994)	1-median	Barrier	Polygonal	Euclidean
Hamacher and Nickel (1994)	1-median	Forbidden Region	Polygonal	Euclidean
Hamacher and Nickel (1994)	1-median	Forbidden Region	Polygonal	Rectilinear
Hamacher and Nickel (1994)	1-median	Forbidden Region	Polygonal	Chebyshev
Archipelago Instance	1-center	Forbidden Region	Arbitrary	Euclidean
Archipelago Instance	2-center	Forbidden Region	Arbitrary	Euclidean

2.5.1 Forbidden Region Instances

There exists four instances with forbidden regions in the literature, one described by Aneja and Parlar (1994), and three by Hamacher and Nickel (1994) which I model using the proposed framework and subsequently solve to optimality. In the remaining sections, I introduce these instances and discuss in detail how they were solved.

2.5.1.1 The Aneja and Parlar (1994) Instance

The instance introduced by Aneja and Parlar (1994) is shown in Figure 2.7. This is a non-convex polygonal forbidden region for a single facility median problem. First, I will show that the non-convex forbidden region can be modeled as a phi-object. Then, I will construct a model by using the function defining the relationship between specified forbidden region and the new facility and solve the model to obtain an optimal solution to the problem.

Originally, this problem is posed as a 1-median problem with a non-convex forbidden region R shown in Figure 2.7. The coordinates of the customer sites are $e_1 = (0, -10)$, $e_2 = (11, -10)$, $e_3 = (0, 11.6)$ and $e_4 = (11, 11.6)$, each of which has a weight equal to 1. The distance metric is Euclidean.

The non-convex composed object in Figure 2.7 is the union of nine basic objects, as shown in Figure 2.8, each of which is a convex polygon. Each basic object in turn is formed by intersecting primitive objects, which in this case are half planes.

For example, the basic object B_1 in Figure 2.8a is the intersection $B_1 = \bigcap_{k \in \mathsf{P}_1} P_k$ where $\mathsf{P}_1 = \{l_1, l_2, l_3\}$ and $l_1 - l_3$ denote the three half-planes, whose bounds are defined by

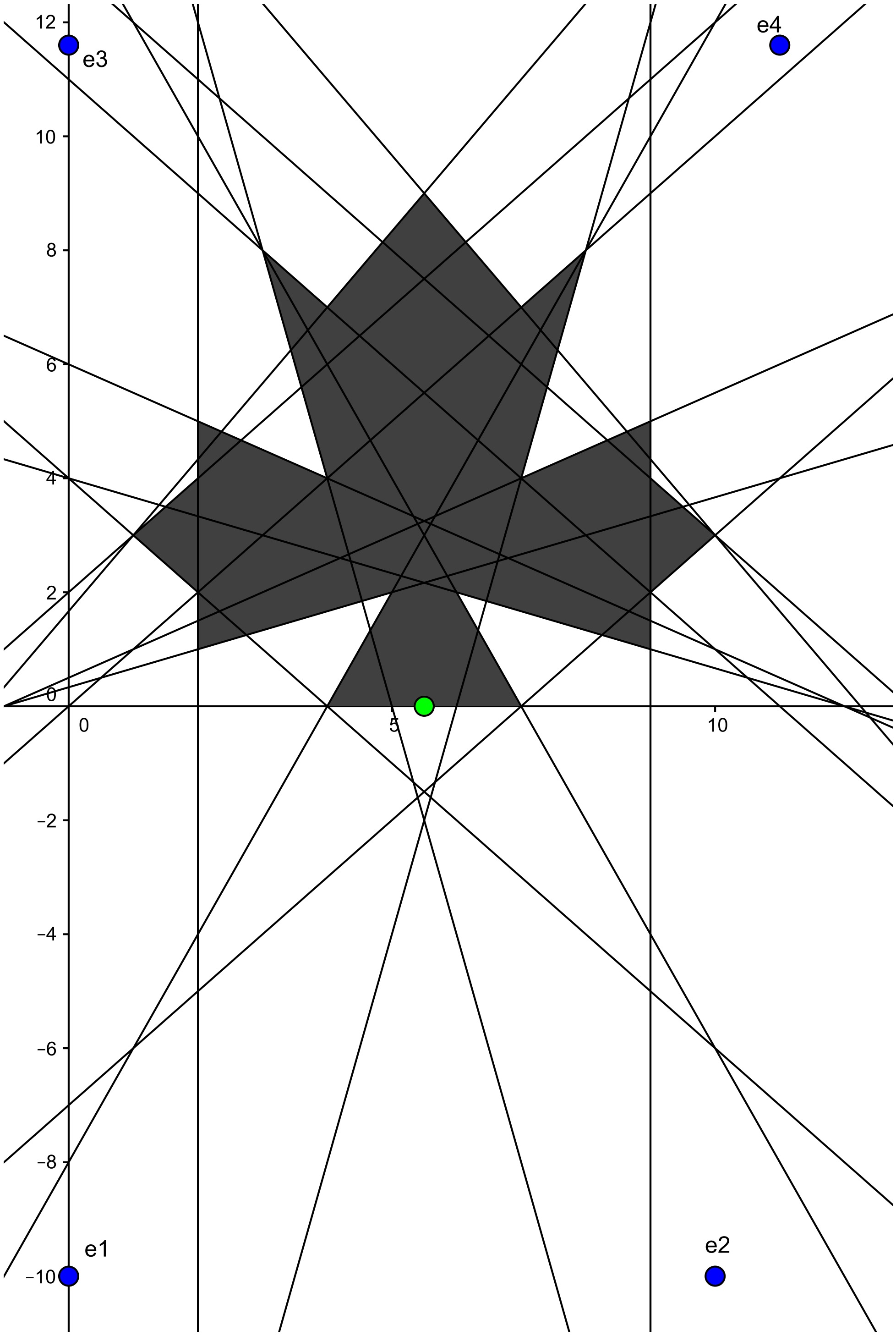


FIGURE 2.7: The numerical example of Aneja and Parlar (1994)

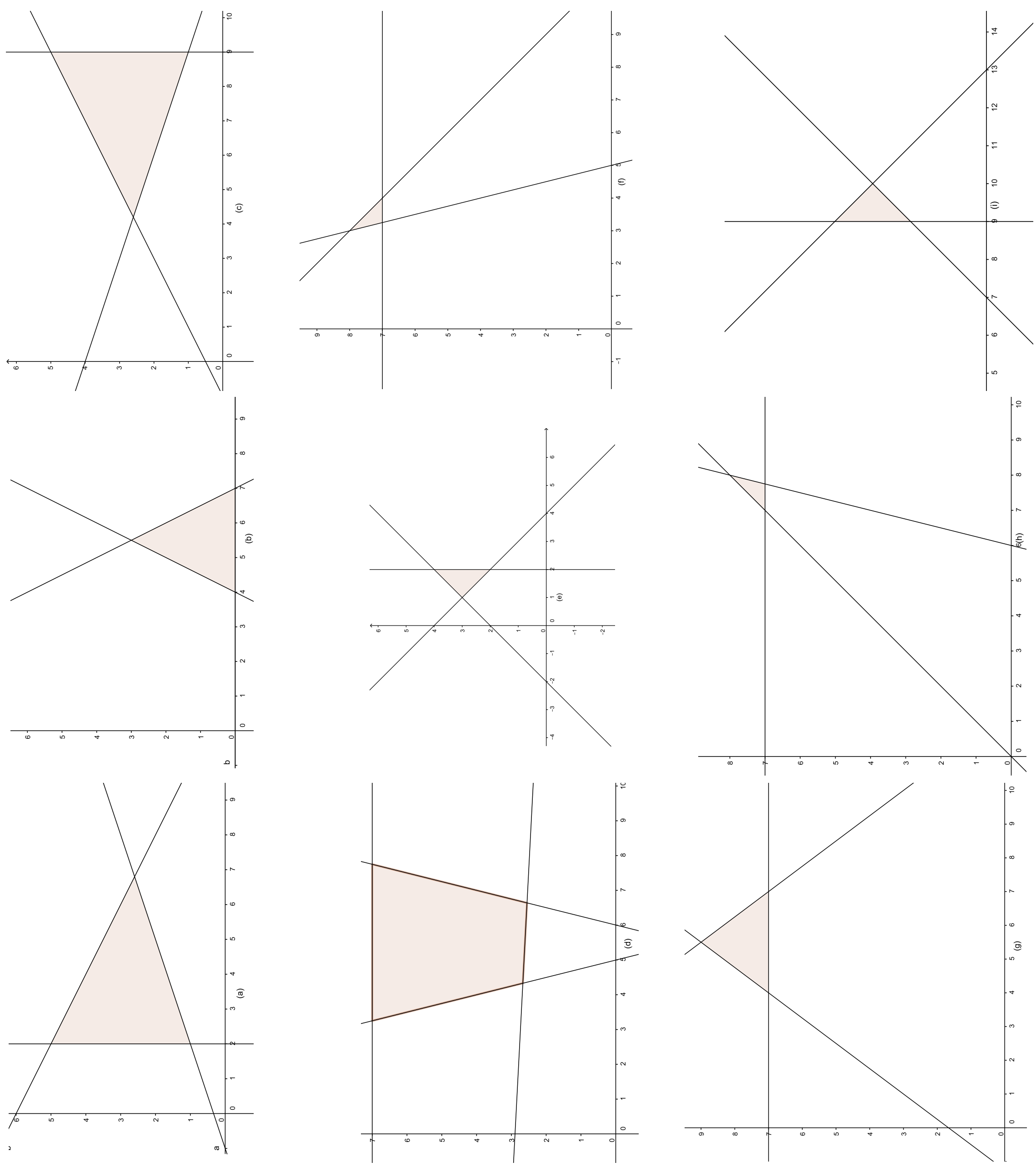


FIGURE 2.8: Nine basic objects of the numerical example

the following three functions $f_1 : -x + 2 = 0$, $f_2 : -y + 0.333x + 0.333 = 0$ and $f_3 : y + 0.5x - 6 = 0$, respectively. The rest of the basic objects $B_2 - B_9$ shown in Fig8b – Fig8i respectively can be defined in a similar way. The forbidden region R in Figure 2.7 is then modeled as $\bigcup\limits_{k \in \mathsf{B}} B_k$.

Since this instance is of a 1-median problem, the function of point X representing the relationship between location of the new facility and the forbidden region R is shown below, and since R is a forbidden region, the function, $\Gamma(X, R)$, must be greater or equal to 0 in the model.

$$\begin{aligned} \Gamma(X,R) = \min\{ &\max_{k=1,2,3}\{f_k\}, \max_{k=4,5,6}\{f_k\}, \max_{k=7,8,9}\{f_k\}, \max_{k=10,11,12,13}\{f_k\}, \\ &\max_{k=14,15,16}\{f_k\}, \max_{k=17,18,19}\{f_k\}, \max_{k=20,21,22}\{f_k\}, \max_{k=23,24,25}\{f_k\}, \max_{k=26,27,28}\{f_k\}\}, \end{aligned} \tag{2.22}$$

where f_k, $k = 1, ..., 28$, is a linear function associated with each primitive object.

As mentioned in Section 2.4, I use either-or constraints to linearise the terms max and min appearing in the constraints. For instance, $\max\limits_{k=1,2,3}\{f_k\}$ in (2.22) can be converted into the constraints,

$$-x + 2 \geq 0 - Mv_{111} \tag{2.23}$$

$$-y + 0.333x + 0.333 \geq 0 - Mv_{211} \tag{2.24}$$

$$y + 0.5x - 6 \geq 0 - Mv_{311} \tag{2.25}$$

$$v_{111} + v_{211} + v_{311} \leq 2 \tag{2.26}$$

$$v_{111}, v_{211}, v_{311} \in \{0, 1\}. \tag{2.27}$$

The resulting model is a MINLP. BONMIN 1.7 is used to solve the formulation to optimality, yielding the optimal solution $d = (5.5, 0)$, which is same with the solution given in Aneja and Parlar (1994). The execution time is 0.008 seconds.

2.5.1.2 The Hamacher and Nickel (1994) Instances

Hamacher and Nickel (1994) introduce some single facility median instances with a convex forbidden region represented by a rectangle $R = [3, 11] \times [9, 15]$, which is a basic object in our framework. The forbidden region and the customer sites are shown in Figure 2.9. The coordinates of the customer sites are $e_1 = (5, 13)$, $e_2 = (7, 11)$, and $e_3 = (5, 11)$, all with unit weights. This instance has been solved with three different distance metrics, namely rectilinear, squared Euclidean and Chebyshev.

Since the forbidden region in this problem is a basic object, the function defining the relation between the point X and basic object B is shown as follows;

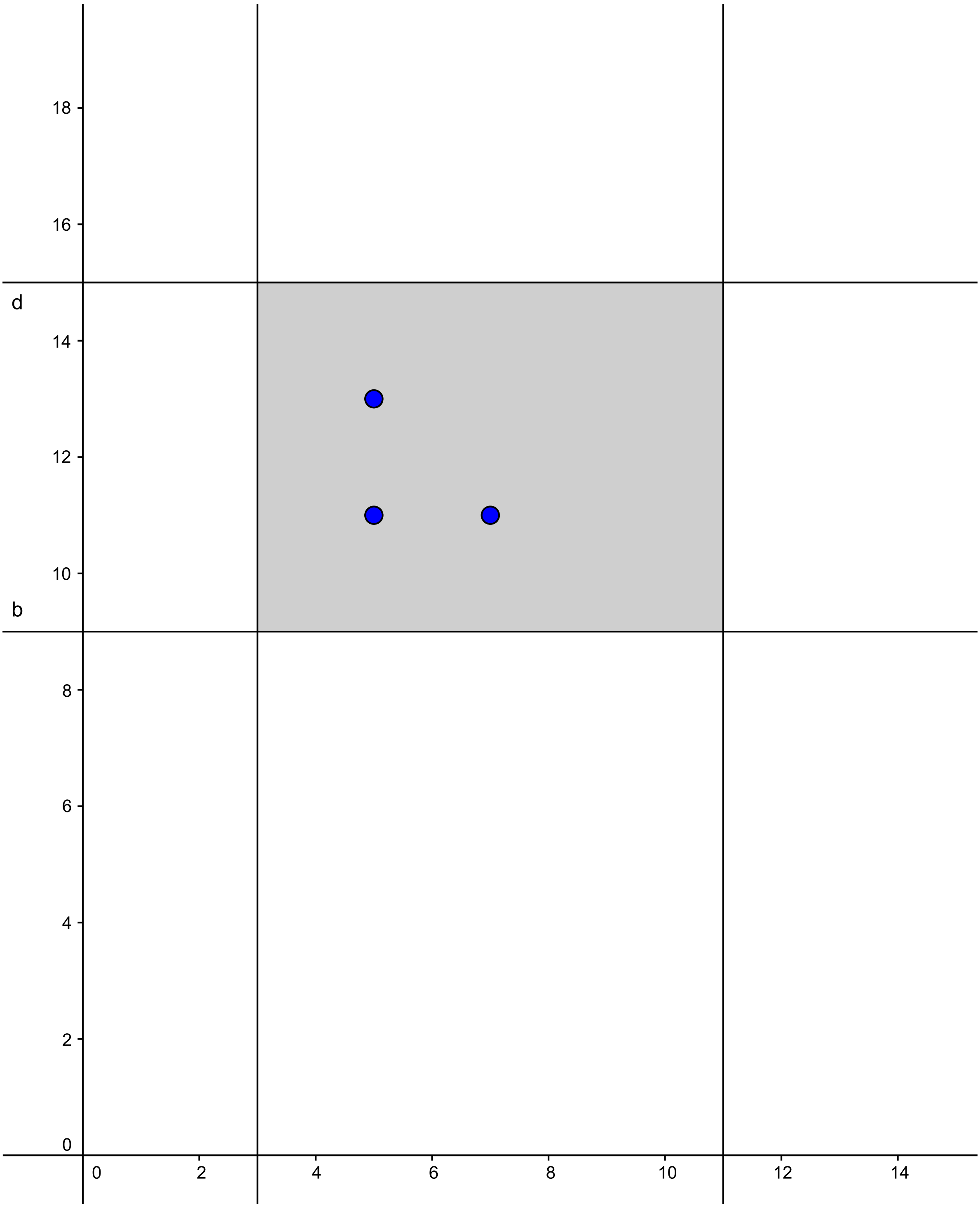


FIGURE 2.9: Example in Hamacher and Nickel (1994)

$$\gamma(X, B) = \max_k (f_k), \tag{2.28}$$

where f_k, $k = 1, 2, 3, 4$, is a linear function associated with the half planes that bound each edge of the rectangle. Our proposed solution technique applies to all three distance functions mentioned above by only modifying the objective function of the model but keeping the constraint set the same.

In these models, either-or constraints are used to define the maximum in (2.26). In the second and third models, a linearisation method is applied, which has resulted in a mixed integer programming formulation, using the linearisation technique for rectilinear and Chebyshev distance functions presented in Hamacher and Nickel (1995). The first model remains a mixed integer non-linear program, since the distance metric is squared Euclidean. In the third model, the Chebyshev distance function is transformed to rectilinear distance function (Hamacher and Nickel 1995).

The first model is solved by BONMIN 1.7 solver. The optimal location for the new facility is $(5.667, 9)$ with an objective value 26.667. The execution time is 0.004 seconds. The second and third models are solved by CPLEX 12.5.1.0. The optimal location of the new facility for the second model is $(3, 11)$ with an objective value 10. For the third model, $(3, 11)$ is the optimal location with an objective value, 8. All of these results are also reported in Hamacher and Nickel (1994).

On a related note, I have observed that these instances exhibit alternative optima. In order to find these solutions, the CPLEX solution pool feature can be used to prune the optimal solutions found. Using this feature, I were able to obtain $(3, 11.667)$, $(5, 9)$ and $(5, 9)$ as alternative optima for the first, second and third model respectively, which coincide with the solutions reported in Hamacher and Nickel (1994).

2.5.2 Barrier Instances

There are three barrier instances that have been described in the existing literature, all of which are modeled through the proposed framework. The first two instances were first presented in Katz and Cooper (1981), for which the parameters use the following values: $J_{max} = 100$, $V = 0.01$ m/s and $t_{max} = 0.2$ seconds. The third instance was first presented in Aneja and Parlar (1994), for which the parameters are $J_{max} = 100$, $V = 0.01$ m/s and $t_{max} = 0.2$ seconds. These values were decided by considering the performance of the solvers at hand and the instances being solved. I now present the solutions of these instances in more detail.

2.5.2.1 Katz and Cooper (1981) Instances

In the first instance, the barrier is a circle with a radius of two, with its center located at $(0, 0)$. There are five customer sites, with coordinates $(-8, -6)$, $(-7, 13)$, $(-1, -5)$,

$(6.6, -0.5)$ and $(4.4, 10)$, each with a weight equal to one. The distance metric is Euclidean. This instance is also studied in various papers including Butt and Cavalier (1996), Klamroth (2001), Bischoff and Klamroth (2007) and Klamroth (2004).

Since the barrier in this problem is a circle, the corresponding phi-object is simply a primitive object, which makes the model a NLP. IPOPT 3.11 is used to solve the formulation to optimality, yielding the solution $(-1.186, 2.060)$ with an objective value of 48.257 as shown in Figure 2.10. This solution is better than the one found in Katz and Cooper (1981) and same as given in Butt and Cavalier (1996), Klamroth (2001), Bischoff and Klamroth (2007) and Klamroth (2004). The execution time is 0.234 seconds.

The second instance in Katz and Cooper (1981) is very similar to the first one with the exception that the radius is now three. The number of customer sites is increased to 10, which are located on $(8, 8)$, $(5, 7)$, $(6, 4)$, $(-3, 5)$, $(-6, 6)$, $(-3, -4)$, $(-5, -6)$, $(-8, -8)$, $(5, -5)$ and $(8, -8)$. Bischoff and Klamroth (2007) also applied their solution algorithm to this problem.

The model for this problem is also a NLP which is solved by IPOPT 3.11. The solver states that the location of the new facility is $(3.306, -0.068)$ with an objective value of 88.326 as an optimal solution, which is similar to the results given in Bischoff and Klamroth (2007), but different from the result in Katz and Cooper (1981). Katz and Cooper (1981) claim that there are 8 local minima ranging from 79.225 to 76.558. Due to Bischoff and Klamroth (2007) and the solver used in this thesis, the result claimed by Katz and Cooper (1981) is infeasible. The execution time is 0.452 seconds. The solution is shown in Figure 2.11.

2.5.2.2 Aneja and Parlar (1994) Barrier Instance

In this instance there are 10 convex and two non-convex polygon barriers. There are 18 facilities, each with a unit weight, placed on points $(1, 2)$, $(6, 1)$, $(9, 1)$, $(14, 2)$, $(5, 5)$, $(7, 4)$, $(9, 5)$, $(14, 4)$, $(17, 4)$, $(2, 8)$, $(8, 8)$, $(16, 8)$, $(3, 12)$, $(6, 11)$, $(9, 10)$, $(17, 10)$, $(10, 12)$ and $(19, 13)$. This is a 1-median instance where the aim is to find the location of the new facility that minimises the total weighted distance.

Since there are numerous barriers in this problem, there will be a large number of constraints in the model, which can be detrimental to the performance of a solver. To overcome this drawback, the model is solved using a cutting plane technique, whereby a relaxed version of the model is solved and constraints are added in an iterative manner. As a first step, barriers are treated as forbidden regions. Figure 2.12 shows the solution of the relaxed problem, where the new facility is placed on the coordinate $(8.913, 6.355)$ and is shown by a red dot.

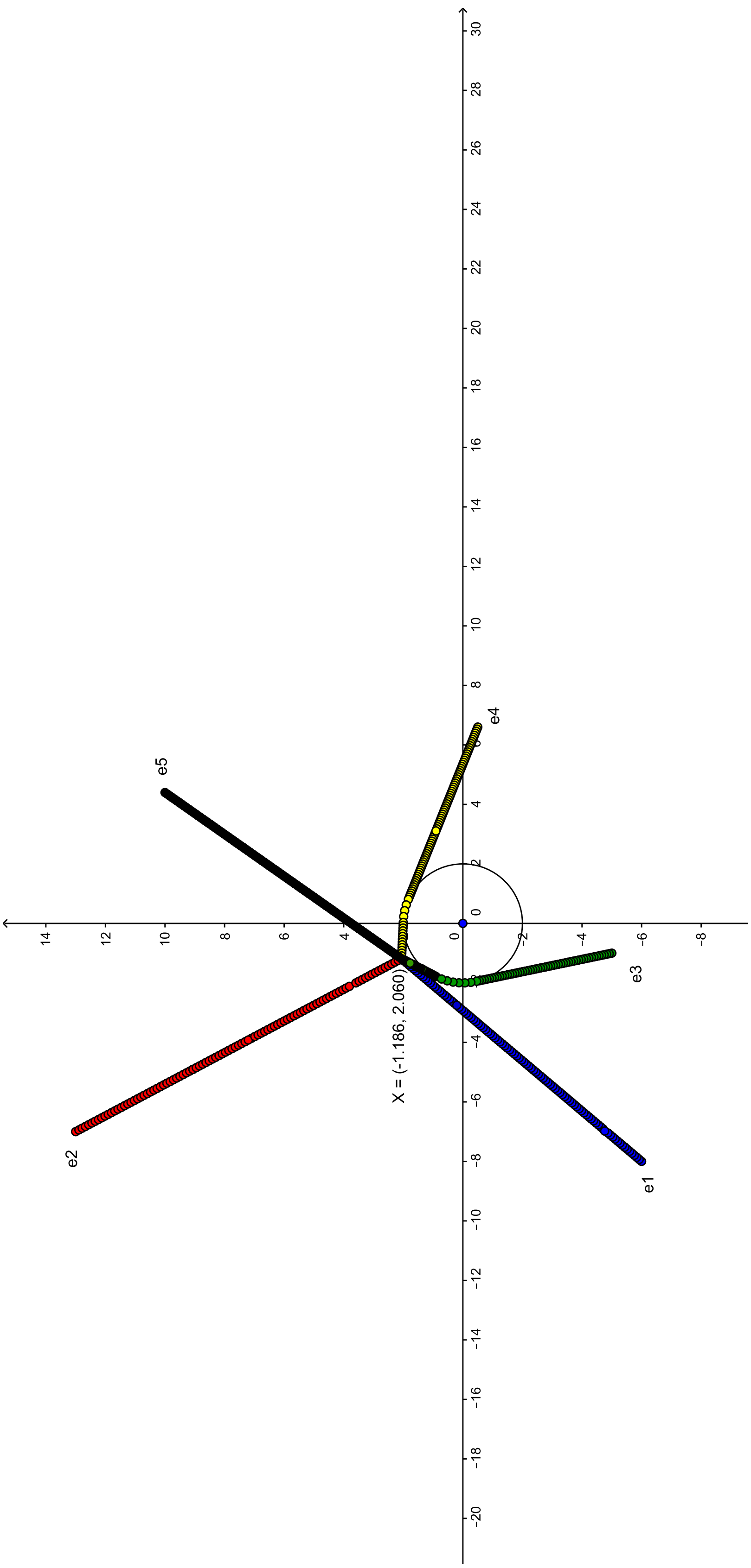


FIGURE 2.10: Katz and Cooper (1981) instance no. 1

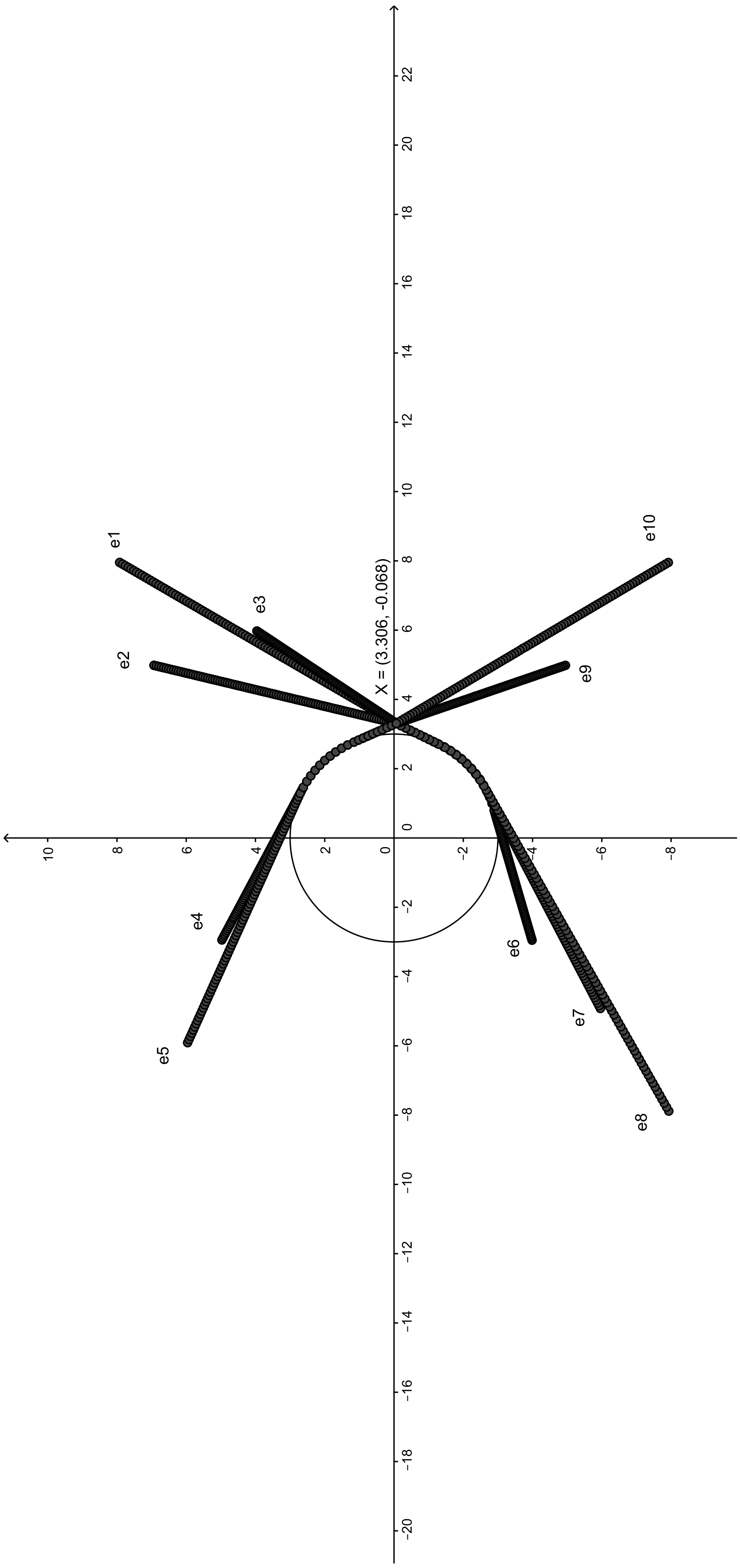


FIGURE 2.11: Katz and Cooper (1981) instance no. 2

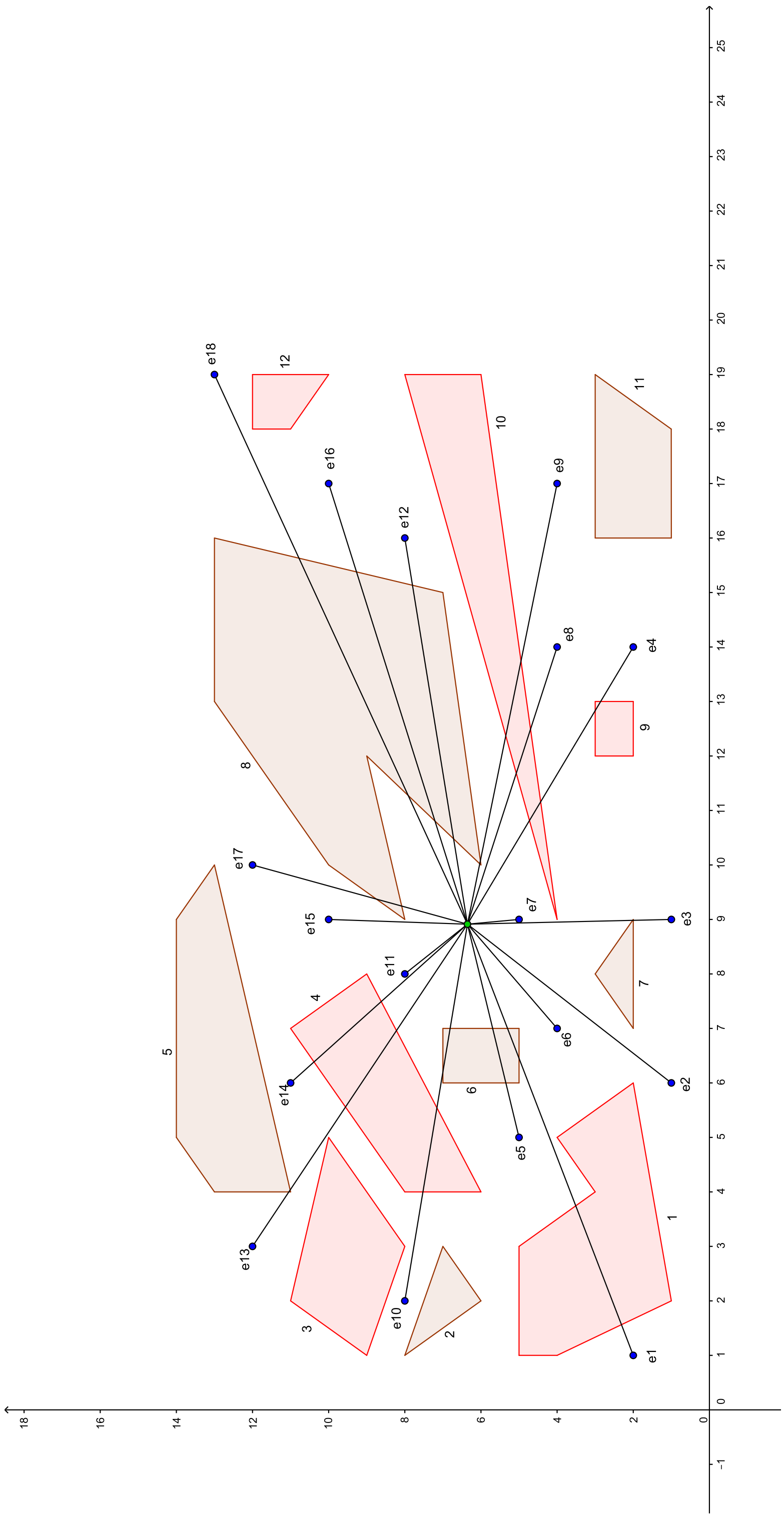


FIGURE 2.12: Cutting plane step 1

In Figure 2.12, it can be seen that certain barrier constraints are violated. For instance, a path from the first facility e1 to the location of a new facility intersects with barriers 1 and 6. In this case, I add the constraint sets that define regions 1 and 6 as barriers. Other infeasibilities are shown in Table 2.3, for which the corresponding constraints are also added and the model is resolved. The resulting solution, shown in Figure 2.13, shows other barriers being crossed, namely 10, 8 and 12. I iteratively add the relevant cuts and resolve the model until a feasible solution is obtained, as shown in Figure 2.14, after four iterations. By applying the cutting plane approach, the total number of equations in the model is decreased from $27,035$ to $4,635$.

TABLE 2.3: Barriers for which constraints are added in step 2

Customer Site	Barrier
1	1, 6
2	-
3	7
4	9, 10
5	6
6	-
7	-
8	10
9	8, 10
10	4, 6
11	-
12	8
13	4, 5
14	4
15	-
16	8
17	8
18	8

TABLE 2.4: Barriers for which constraints are added in step 3

Customer Site	Barrier
3	10
15	8
18	12

Once all the violated constraints are added, there is one other potential issue that needs to be addressed to reach optimality for this instance. Even though constraints are added to force the points that form the paths to be outside of the barriers, a path may well contain a line segment that crosses the corners of a barrier. This infeasibility can be addressed by adding some constraints to the model, which dictate that one of the two points at either end of such a line segment should be placed on the vertex. For example,

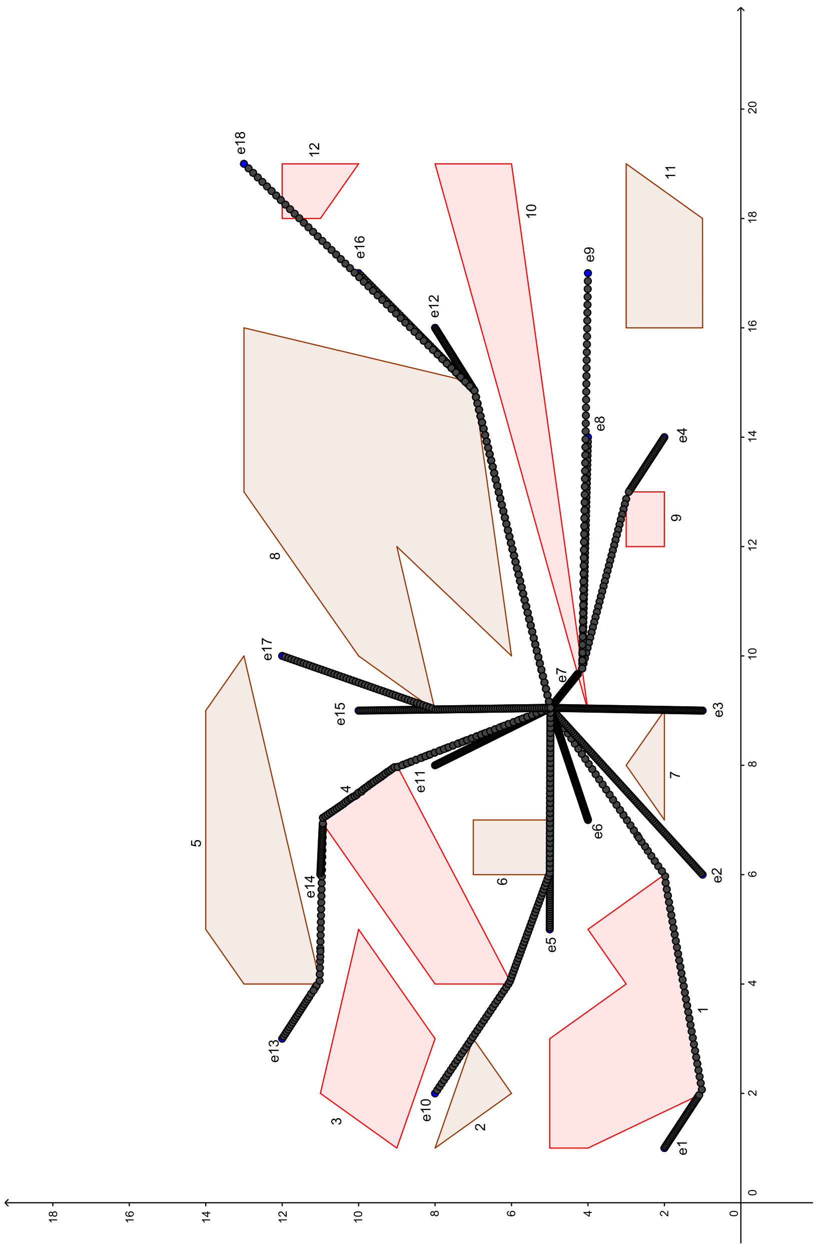


FIGURE 2.13: Cutting plane step 2

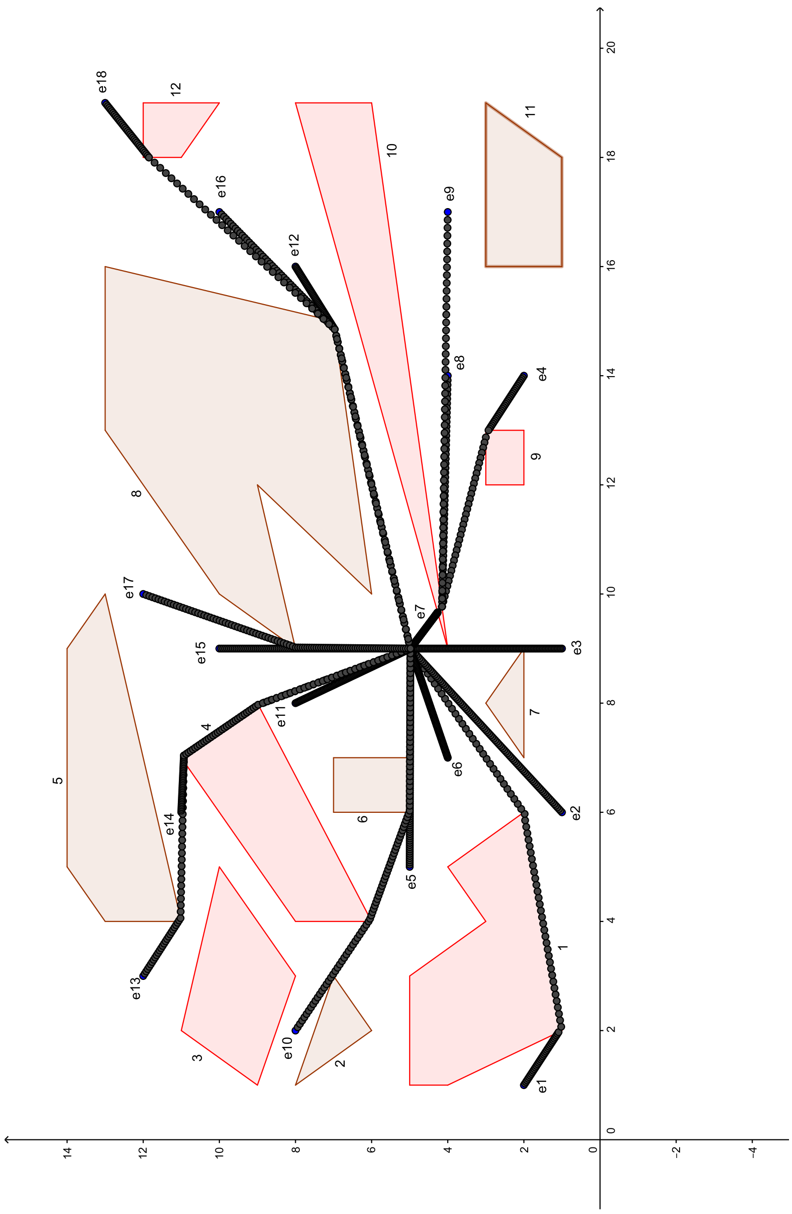


Figure 2.14: Cutting plane step 3

the 13th point of Path 1 shown in Table 2.5 will be placed on the vertex coordinate $(2, 1)$ of barrier 1, and will therefore prohibit the formation of a path that passes through it.

Table 2.5: Vertex points

Customer Site (Path)	Point (J)	Barrier	Vertex Coordinate
1	13th	1	(2,1)
1	54th	1	(6,2)
3	45th	7	(9,2)
4	18th	9	(13,3)
4	60th	10	(9,4)
8	58th	10	(9,4)
9	60th	10	(9,4)
10	28th	4	(4,6)
10	46th	6	(6,5)
12	26th	8	(15,7)
13	15th	5	(4,11)
13	47th	4	(7,11)
13	65th	4	(8,9)
14	8th	4	(7,11)
14	27th	4	(8,9)
16	38th	8	(15,7)
17	56th	8	(9,8)
18	10th	12	(18,12)
18	52nd	8	(15,7)

For this instance, optimality is reached after adding these constraints. After applying the cutting plane technique, moving specific points to barrier vertecies can be called as post-processing to reach a feasible optimal solution. An optimal solution is presented in Figure 2.15.

The model for this problem is a NLP with max and min operators in the constraints. It may be transformed into a MINLP, but to keep the size as small as possible, I opted to keep it as a NLP and used IPOPT 3.11 as a solver using its internal functions to model the max and min operators. The optimal solution reported by IPOPT 3.11 has a value 119.176 where the new facility is placed at (8.752, 4.979). The execution time is 0.936 seconds. The values are slightly different than the results found in Aneja and Parlar (1994) and Bischoff and Klamroth (2007), both of which report $(8.7667, 4.9797)$ for the location of the new facility and 119.1387 for the objective value.

2.6 Case Study: New Single and Multi-facility Instances

In this section, I introduce, model and solve new instances based on real geography, for both single and multiple facilities. The new set of instances are derived from an

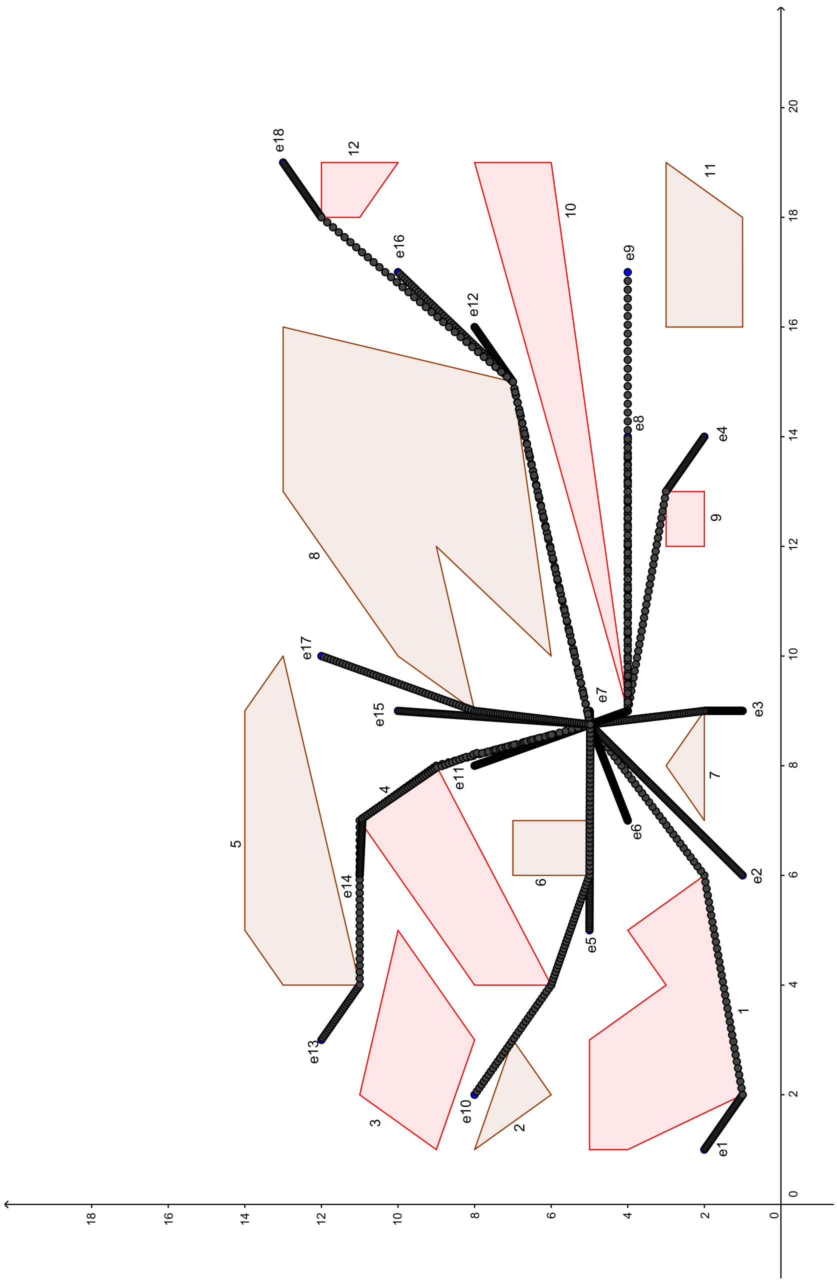


Figure 2.15: Final solution

archipelago called Prince Islands in the Marmara Sea near Istanbul, Turkey. These islands have historical and touristic importance. There are nine of these islands, namely Tavsanadasi, Sedefadasi, Buyukada, Heybeliada, Kasikadasi, Burgazada, Kinaliada, Sivriada and Yassiada, but only six are populated. A satellite view of Prince Islands taken from Google Earth software is shown in Figure 2.16. Prince Islands are in close proximity to an active fault line and face the danger of earthquakes. For this archipelago, I focused on the problem of locating a relief center with a helicopter field. Sending supplies from Istanbul to Prince Islands may take considerable time, if it is also hit by an earthquake. Therefore, it is vital for the residents and the visitors of these islands that they have their own autonomous emergency response system. After a possible earthquake, relief items from the new depot can be sent to each of the populated islands using helicopters. Given the nature of the problem, an appropriate objective here would be to minimise the maximum distance between each populated island and its closest new relief depot.

For this instance, I solve 1-center and 2-center problems using the modelling framework where the Marmara Sea is modeled as the forbidden region, and the Prince Islands as the feasible regions. In contrast to previous instances, the phi-objects in this case model the feasible regions implying that the forbidden region is non-convex shaped and unbounded. To approximate these phi-objects, 105 half planes, 10 circles and 14 circular holes are used as primitive objects, as shown in Figure 2.17.

The helicopter fields on each populated island can be thought of as customer sites. The weights are calculated as follows. The highly populated islands in this archipelago are Buyukada and Heybeliada, the less populated islands are Sedefadasi and Yassiada. Population in Burgazada and Kinaliada is on medium level compared to the other islands. It is, therefore, assumed that the weights of Heybeliada and Buyukada are both 3, Burgazada and Kinaliada are both 2 and Sedefadasi and Yassiada are both 1. The coordinates of the customer sites are assumed to be $(1, 7)$, $(5, 5)$, $(9, 6)$, $(13, 7)$, $(17, 4)$ and $(19, 14)$, which are shown with blue dots in Figure 2.17. Since the transportation of relief items will be done by helicopter, Euclidean distance is considered as the distance metric. The representation of Prince Islands as phi-objects, the locations of existing helicopter fields and the solutions for 1-center and 2-center problems are shown in Figure 2.17. The 1-center problem for these islands is solved by BONMIN 1.7, which yielded the location of the depot as $(9.837, 5.192)$, which is shown by a red dot on Heybeliada. The objective value is 14.513 and the execution time for this model is 0.215 seconds. The 2-center problem is again solved by BONMIN 1.7, according to which the new depots are to be placed at $(13.599, 6.978)$ and $(8, 6.813)$, which are shown by green dots. The objective value is 10.517. These new facilities are on Burgazada and Heybeliada. The execution time for this model is 0.437 seconds.

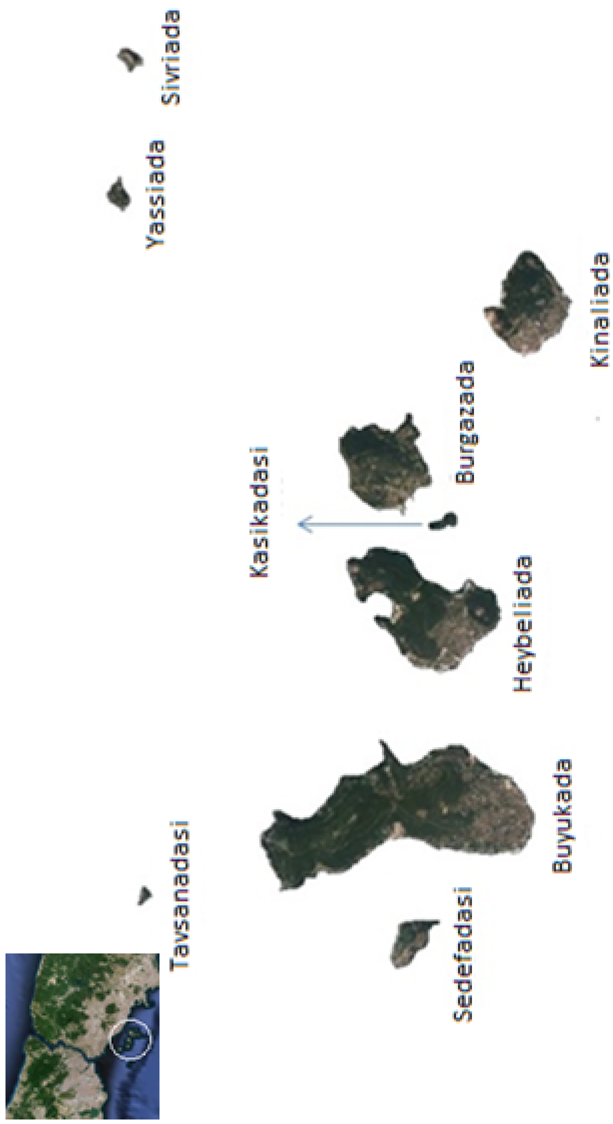


Figure 2.16: Prince Islands

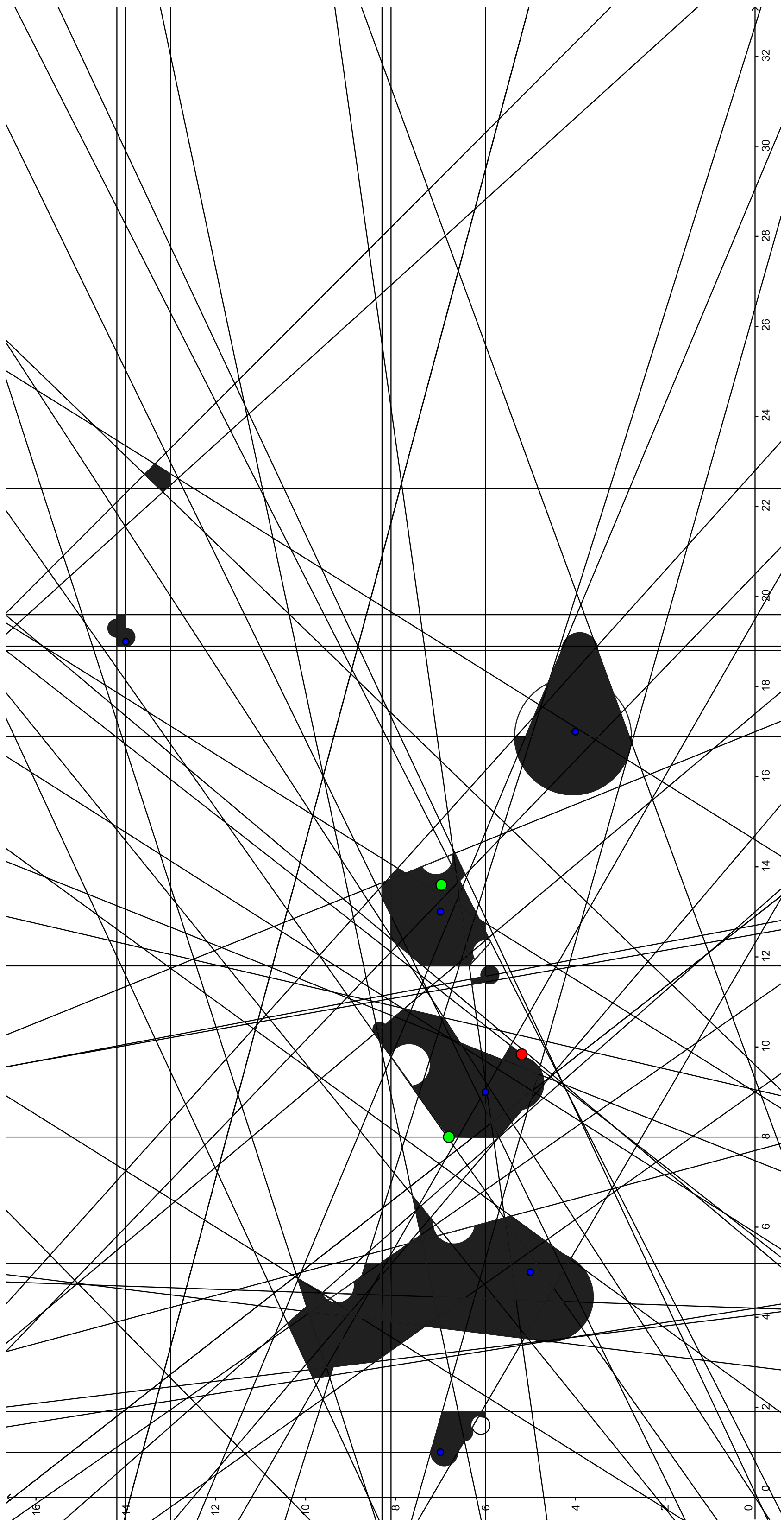


FIGURE 2.17: Phi-objects representing the Prince Islands and the solutions of the problems

2.7 Conclusions

This paper has described a general framework for modelling continuous facility location problems with either forbidden regions or barriers by using the concept of phi-objects. The use of phi-objects allows one to model either a restricted region or a feasible region, thus an unbounded restricted region can also be taken into consideration. Another advantage of using phi-objects is the flexibility in modelling various geometric shapes. Many complicated geographical shapes (e.g., archipelago, bays, lakes) can be modeled with negligible approximations. Within this framework, it is also possible to formulate both median and center problems with various types of distance metrics such as Euclidean, rectilinear and Chebyshev. Using the models, I have successfully solved various instances from the literature for this class of problems to optimality using state-of-the-art linear and nonlinear programming solvers.

Building from the framework I present in this paper, there are two clear research challenges. To the best of our knowledge, there is limited research on multi-facility location problems with barriers. I believe that a solution approach to these problems is needed. Another topic that warrants further research is a similar general modelling framework for stochastic versions of the restricted location problems. There are many real-life examples of this problem, where random obstacles are created by natural disasters, especially flooding. Many countries around the globe, including the United Kingdom, face high risks of floods which can occur during heavy storms. Such restricted regions can be treated as 2D obstacles on a plane with a stochastic formation pattern. Although some research on this variant has appeared, there is still a need for a general modelling approach.

Chapter 3

Discretisation, Multi-commodity Flows and Benders Decomposition for Restricted Stochastic Continuous Location Problems

Abstract

This study presents a new formulation and a solution algorithm for continuous restricted location problems. The model is defined on a discretisation of the space and is based on multi-commodity flows with unknown destinations. The discretisation can be applied to deterministic and stochastic continuous restricted location problems using any distance metric. The solution algorithm is based on Benders Decomposition. This paper presents extensive computational results on multi-facility restricted location problems derived from a well-known and complex instance from literature.
Keywords: mathematical modelling, facility location, network design, multi-commodity network flow, Benders decomposition

3.1 Introduction

Restricted continuous facility location problems have attracted attention in the literature due to their complexity and applications, including assembly of printed circuit boards (Foulds and Hamacher 1994), obnoxious facility planning (Carrizosa and Plastria 1993), urban planning, and, in particular, location of emergency facilities. These problems are concerned with locating one or more facilities on a continuous plane, containing restricted regions. The restriction types can be classified into three categories, namely forbidden regions, barriers and congested regions (Canbolat and Wesolowsky 2010). In problems with forbidden regions, locating new facilities within the region is not permitted but traveling through them is possible. In the case of barriers, neither locating a facility within the barrier nor travel through the barrier is permitted. In problems with congested regions, traveling through these regions is possible with some penalty, however locating new facilities within such region is not allowed.

A restricted continuous location problem is said to be deterministic, if the locations of all restricted regions are known in advance of locating the facilities and do not change over time. In stochastic restricted location problems, restricted regions occur randomly. This happens in real world situations such as sudden rail road failures, accidents, unplanned construction and natural disasters. As a result of which, random obstacles may arise. A very common example is when restricted regions are created by natural disasters, in particular floods. Other natural disasters such as earthquakes, typhoons and hurricanes can also cause the formation of barriers. The main reason for the formation of barriers as a result of natural disasters is the collapse of buildings, viaducts and bridges, crushed cars and trucks, and rubble in the streets.

Many countries around the globe, including the United Kingdom, suffer from flooding caused by heavy rainfall, rising ground water, coastal floods caused by wave actions and storm surges, and pluvial floods. Floods have adverse effects on transportation activities due to the formation of large puddles, mudslides and possible debris after the natural disaster, all of which can be seen as barriers in the context of restricted location problems. According to Prinos et al. (2009), a flood hazard map is a detailed map, which indicates geographical areas (flood zones) that could be covered by a flood according to several probabilities: flood scenarios with low, medium and high probability. As an example, Figure 3.1 shows a flood hazard map of Europe (Alfieri et al. 2014). The blue regions on the map show the locations of possible flood occurrences. Considering the definition of a flood hazard map (Prinos et al. 2009), the probabilistic behavior of the flood zones is the main motivation why I consider stochastic restriction, since locating a facility within flood zones would be unsafe and costly. Furthermore, transportation of goods or people through them would be difficult after a possible flood.

In 2002, a flood event occurred in Central Europe, affecting several countries such as Germany, the Czech Republic, Austria, Italy and Slovakia, due to unusual meteorological conditions. The total number of fatalities caused by flooding was 100. Total economic loss was estimated at 9 billion Euros for Germany, 3 billion Euros for Austria, and 2.5 billion Euros for the Czech Republic (Ulbrich et al. 2003). The flooding had an enormous negative effect on transportation in the region. All bridges across the main river in Prague malfunctioned, many railway routes were closed across the Elbe river in Germany, and many roads in the region were blocked due to direct flooding (Toothill 2002). In 2009, there was another negative impact on transportation infrastructure in Cumbria, UK due to the floods caused by heavy rainfall (Cumbria County Council 2009a). For over 6 weeks, 25 bridges were closed, and the damage on the regional highway was significant. The total number of businesses that were affected by this natural disaster was 3057 (Cumbria County Council 2009b). More recently, in May

FIGURE 3.1: European flood hazard map (Alfieri et al. 2014)

2015, a flood event due to severe storms in Texas caused at least 27 million dollars in infrastructure damage, including damage to 167 roads in the counties of Texas and 24 causalities (Gallucci 2015b).

Observations from real life cases show that the problem at hand is significant in two ways. First, climate change due to global warming will likely lead to an increase in the number of flooding instances in many places (Milly et al. 2002); (Gallucci 2015a). Thus, further preparedness is needed to decrease the risks associated with this natural disaster. Second, locating new facilities outside the effected regions would decrease the death toll and the financial costs resulting from the disaster.

The problem at hand is to find the optimal locations of new facilities that serve existing demand points located on a continuous plane, by minimising a function of the distance between these demand points and new facilities, while considering uncertainty of barrier occurrences. This problem is challenging to solve, since the random occurrence of barriers implies a stochastic structure. In this paper, a deterministic version of the problem, where the barriers are static, is also considered as a special case. To address this challenge, this paper makes four unique contributions which are listed below:

- A new modelling approach for both deterministic and stochastic restricted facility location problems is presented that is based on discretisation and multi-commodity flows. The novelty of this modelling approach is that the destination nodes of each flow are unknown.
- The proposed modelling approach is applied to single and multi-facility stochastic restricted location problems with forbidden regions and barriers in combination with various objective functions. The shape of the restricted regions can be convex or non-convex and there is no restriction on the distance metric.
- To obtain an approximation of the original problem, a way to construct a network by discretising the continuous space is described.
- An exact algorithm based on Benders decomposition is described to solve the proposed models to optimality.

The remainder of this paper is organised as follows. A brief literature review on stochastic restricted facility location problems is presented in Section 3.2. In Section 3.3, I introduce a formal definition of the general problem. Section 3.4 presents the design of the network that will be used to transform the continuous plane to its discrete counterpart, the modelling framework based on a multi-commodity network flow formulation with unknown destination, and the way in which it can be applied to facility location problems with restrictions. In Section 3.5, an exact algorithm based on Benders decomposition is described for solving the problem. I present extensive computational results for evaluating the efficiency of the modelling and solution approach by comparing results of the both discretised deterministic and stochastic restricted facility location instances in the literature in Section 3.6. The paper concludes in Section 3.7.

3.2 Literature Review

The deterministic restricted facility location problem has been studied in the literature to a good extent. For a review of single and multiple facility problems with forbidden regions and single facility location problems with barriers see Oğuz et al. (2016). To the best of our knowledge, Bischoff et al. (2009) is the only paper to study restricted multi-facility location problems with barriers. Their solution algorithm is a heuristic method based on genetic algorithms.

In contrast, only a few studies exist on restricted location problems with probabilistic barriers. The first of these papers, to best of our knowledge, is by Canbolat and Wesolowsky (2010), who consider the barrier in the form of a probabilistic line, where the objective is to find the location of a single facility which minimises the total expected rectilinear distance from a set of demand points to the location of the new facility. The authors describe an algorithm based on dividing the problem into two, and comparing their solution to obtain a solution for the original problem. The authors have several strict assumptions, including the uniformity of the barrier location and the invariability of the y-coordinate of the line barrier. Amiri-Aref et al. (2011) also consider a probabilistic line barrier that is uniformly distributed in the plane, and used rectilinear distance as the metric. Unlike Canbolat and Wesolowsky (2010), the objective here is to minimise the maximum expected distance from the demand points to the new facility. The authors propose a heuristic to solve the problem. Shiripour et al. (2012) considered the multi-facility version of the same problem, where the objective is to minimise the total expected rectilinear distance from new facilities to the demand points. For small size instances, the authors use a commercial solver to solve a mixed integer quadratic programming formulation of the problem. For larger instances, they propose metaheuristic algorithms, namely a genetic algorithm and an imperialist competitive algorithm (a type of evolutionary algorithm). Amiri-Aref et al. (2013) describe a heuristic for the problem with a probabilistic polyhedral barrier, extending the aforementioned problems involving only a probabilistic line barrier and a rectilinear distance metric. Finally, Javadian et al. (2014) studied a problem under a multi-period planning horizon, using rectilinear distances and a uniformly distributed line barrier, and proposed a genetic algorithm and imperialist competitive algorithm.

3.3 Problem Description

Consider a set M of demand points located on a plane, each with a positive weight v_m corresponding to the amount of demand at each point $m \in \mathsf{M}$. These demand points need to be served by θ new facilities that will be located on the plane so as to minimise a given function, which is the expected value of the weighted distance from the demand points to the new facilities. I am also given a finite set of potential locations of the possible barrier occurrences that may arise on the continuous plane (e.g. flood hazard maps), where each realization of the set of occurrences is modeled as a scenario. Let S be the set of scenarios and p_s is the probability of occurrence for scenario $s \in \mathsf{S}$. The problem is to find an optimal placement of the new facilities that can serve the demand points in any possible scenario by considering the associated probabilities of each scenario.

3.4 Modelling with Discretisation and Network Flows

In this section, I first present two different network constructions in order to obtain approximations of the solution of the original continuous problem. I briefly describe multi-commodity flow problems and discuss how they can be applied to restricted facility location problems using a discretised continuous plane. Finally, I introduce the general model.

3.4.1 Construction of the Discrete Network

The basic idea is to divide the space into unit squares where each vertex is connected to each other with an edge to be able to represent various distance metrics (e.g., Euclidean). Each unit square contains four vertices and six edges, and its shape is similar to that of a sealed envelope. An example of this network with four unit squares, nine nodes and 20 edges is given in Figure 3.2.

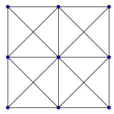


Figure 3.2: Unit square network

When the number of nodes is increased, the size of the unit square will decrease, with the expectation that the approximation of the continuous facility location problem will improve.

An alternative way to model the discrete network and refine the approximation is to increase the number of edges by using unit rectangles that are formed by unit squares, instead of unit squares alone, as shown in Figure 3.3. These unit rectangles form the columns and rows of the general network, where each node in each unit rectangle is connected in a cross-cutting manner. This increases the number of diagonal edges on the general network, which is in contrast to the discretisation shown in Figure 3.2, which has only one type of diagonal edge.

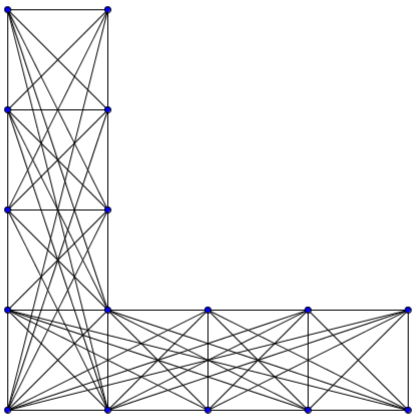


Figure 3.3: Unit rectangle network

In Figure 3.3, two unit rectangles made up of unit squares are shown. One of the rectangles represents a single column of the network, the other rectangle represents a single row of the network. It is clearly seen that the number of edges leaving and entering each node has increased, making more links available for flows compared to a unit square network with the same number of nodes. The application and evaluation of quality of the two discretisations are presented in Section 3.6.

It is also possible to develop a tessellation based on other geometric shapes, such as triangles and hexagons, but in this thesis the tessellation is limited to unit squares and rectangles.

One point of consideration is regarding the convex hull of the network formed by the locations of the existing facilities and the vertices of the barriers. It is sufficient to construct the discretised network within the convex hull since it is known that the global optimal facility location lies in this convex hull (Butt and Cavalier 1996).

3.4.2 Multi-commodity Network Flows for Restricted Facility Location Problems

The nature of multi-commodity network flow problems and facility location problems are, in a sense, similar. Multi-commodity network flow problems determine the flows of commodities from their origins to destinations in order to minimise the total cost of flow routing along the arcs of a given graph (Moradi et al. 2015). In the standard version of the problem, the origin and destination nodes for each commodity are known. In the variant I propose, the destination nodes, which correspond to the locations of the facilities, are unknown.

I model the origin nodes of each commodity flow as the locations of the demand points and the unknown destination nodes as the locations of the new facilities. Consider a network G formed by a set of nodes $\mathbf{N}$ and a set of uncapacitated arcs $\mathbf{A}$ in which there will be a flow for each element of a set $\mathbf{M}$ of commodities that has a one-to-one mapping with demand points. Each vertical or horizontal arc has a unit non-negative cost, which in our case, is the unit distance between a node pair connected by a single vertical or horizontal arc. The values that diagonal arcs hold are calculated by considering the values that horizontal and vertical links hold. In this problem, a set of origin nodes $\mathbf{O}$ for each commodity $m \in \mathbf{M}$ is defined, which is also a subset of $\mathbf{N}$. While locations of the destination nodes are unknown, the total number is known. I wish to send a single unit of flow from each source to unknown sinks using a least costly path, where the cost of the path is the total cost of the arcs forming the path. In the stochastic variant of the problem I represent the set of available arcs by $\mathbf{A}_s \subseteq \mathbf{A}$, for each scenario $s \in \mathbf{S}$.

3.4.3 A multi-commodity network flow formulation

There are two nonnegative continuous and one binary variables in the model, where the first continuous variable, x_{smij}, denotes the flow of commodity m through the arc $(i,j) \in \mathbf{A}_s$ in scenario s, and $X = \{x_{smij}, s \in \mathbf{S}, m \in \mathbf{M}, \{i,j\} \in \mathbf{A}_s\}$. The second continuous variable, z_{smi}, indicates the decisions for the locations of destination nodes for each commodity $m \in \mathbf{M}$ in scenario $s \in \mathbf{S}$, and $Z = \{z_{smi}, s \in \mathbf{S}, m \in \mathbf{M}, i \in \mathbf{N})\}$. The binary variable, y_i, takes the value of 1 if the destination is located at $i \in \mathbf{N}$ and $Y = \{y_i \in \{0,1\}, i \in \mathbf{N}\}$. The parameter v_m is the positive weight of the commodity flow, c_{smij} is the per unit cost for routing commodity $m \in \mathbf{M}$ through the directed arc (i,j) in scenario $s \in \mathbf{S}$, $O(m)$ denotes the origin node of commodity $m \in \mathbf{M}$ and θ is the total number of destination nodes. Since there is a finite set of scenarios and the objective is to optimise an expected value, it is possible to form a deterministic equivalent of a stochastic model (DESM), as shown below.

$$\underset{X,Y,Z}{\text{minimise}} \quad \mathrm{E}\left\{F\left(v_m, c_{smij}, x_{smij}\right)_{m\in\mathbf{M},\{i,j\}\in\mathbf{A}_s}\right\} \tag{3.1}$$

subject to

$$\sum_{j:(i,j)\in\mathbf{A}_s} x_{smji} - \sum_{j:(i,j)\in\mathbf{A}_s} x_{smij} = \begin{cases} -1, & \text{if } i = O(m) \\ z_{smi}, & \text{otherwise} \end{cases} \qquad \forall s \in \mathbf{S}, \forall i \in \mathbf{N}, m \in \mathbf{M} \tag{3.2}$$

$$z_{smi} \le y_i \qquad \forall s \in \mathbf{S}, \forall i \in \mathbf{N}, \forall m \in \mathbf{M} \tag{3.3}$$

$$\sum_{i\in\mathbf{N}} y_i = \theta \tag{3.4}$$

$$z_{smi}, x_{smij} \ge 0 \qquad \forall s \in \mathbf{S}, \forall i \in \mathbf{N}, \forall \{i,j\} \in \mathbf{A}_s, \forall m \in \mathbf{M}. \tag{3.5}$$

$$y_i \in \{0,1\} \qquad \forall i \in \mathbf{N}, \tag{3.6}$$

The objective function (3.1) minimises the expected value of a function defining the total weighted distance between the origin nodes (demand points) and destination nodes (new facilities). For example, the expected value of the objective function of a median problem can be written as $\sum_{s \in \mathsf{S}} \sum_{m \in \mathsf{M}} \sum_{\{i,j\} \in \mathsf{A}_s} v_m p_s c_{mij} x_{mij}$. Other expected objective functions can be applied in a similar manner, but in this paper I only consider the objective of the median problem as a linear objective function. Constraints (3.2) model flow conservation for each commodity from demand point $m \in \mathbf{M}$ in scenario $s \in \mathbf{S}$. The flow conservation equations in this model are different as compared the usual network flow conservation equations shown below (Balakrishnan et al. 1987);

$$\sum_{j \in \mathbf{N}} x_{smji} - \sum_{j \in \mathbf{N}} x_{smij} = \begin{cases} -1 & \text{if } i = O(m) \\ 1 & \text{if } i = D(m) \\ 0 & \text{otherwise} \end{cases} \qquad \forall i \in \mathbf{N}, m \in \mathbf{M}, \tag{3.7}$$

where $D(m)$ is the destination node of the mth commodity flow. In constraints (3.2), the variable z_{smi} gives a flexibility to the model to choose optimal destinations for each commodity from demand point $m \in \mathbf{M}$ in scenario $s \in \mathbf{S}$. In other words, when $z_{smi} = 1, \forall s \in \mathbf{S}, m \in \mathbf{M}, i \in \mathbf{N}$, it means the destination point of the flow starting at demand point $m \in \mathbf{M}$ in scenario $s \in \mathbf{S}$ is located on ith node. Constraints (3.3) state that the destination node of each commodity $m \in \mathbf{M}$ under scenario $s \in \mathbf{S}$ can be located on a node $i \in \mathbf{N}$, if $y_i = 1$. Constraints (3.4) set the number of destination nodes, or equivalently, the number of facilities to be located in the problem. Constraints (3.5) and (3.6) define the non-negativity and binary restrictions on the decision variables, respectively. The resulting model is a mixed integer linear program.

There is one important characteristic of this modelling framework, which is that the binary variable y_i can be relaxed to its continuous counterpart, $0 \leq y_i \leq 1$, and the optimal solution of the model will still be integer and same as the original model. The constraint matrix of the model is obviously totally unimodular for each fixed set of decision variable y_i. This means that at optimality the values of both $x^{\star}_{smij}$ and $z^{\star}_{smi}$ are integers. First, consider the case where $y_i < 1$, $\forall i \in \mathbf{N}$. In this case, due to constraint (3.3), the maximum integer value that z_{smi}, $\forall s \in \mathbf{S}, \forall m \in \mathbf{M}, i \in \mathbf{N}$ can have is zero, which makes the model infeasible, where at least one of the z_{smi}, $\forall s \in \mathbf{S}, \forall m \in \mathbf{M}$ has to be equal to one to make flow conservation constraints hold, and a single facility can serve to all the demand points under each scenario $s \in \mathbf{S}$. The only way to guarantee this is to have at least one y_i, $\forall i \in \mathbf{N}$ has to be equal to one. If the number of y_i that are equal to one increases, the number of new facilities will increase as same. Second, consider the objective function of the model, which is a linear function of total distances between demand points and new facilities. To open as much as new facilities possible automatically decrease the objective function value. Thus, at optimality, due to constraint (3.4), the number of $y_i^\star$ that are equal to 1 is θ, and the rest are equal to zero. As a result it is possible to form DESM as a linear program, as shown below.

$$\text{minimise} \quad \sum_{s \in \mathsf{S}} \sum_{m \in \mathsf{M}} \sum_{\{i,j\} \in \mathsf{A}_s} v_m p_s c_{mij} x_{mij} \tag{3.8}$$

subject to

$$\sum_{j:(i,j) \in \mathbf{A}_s} x_{smji} - \sum_{j:(i,j) \in \mathbf{A}_s} x_{smij} = \begin{cases} -1, & \text{if } i = O(m) \\ z_{smi}, & \text{otherwise} \end{cases} \qquad \forall s \in \mathbf{S}, \forall i \in \mathbf{N}, m \in \mathbf{M} \tag{3.9}$$

$$z_{smi} \leq y_i \qquad \forall s \in \mathbf{S}, \forall i \in \mathbf{N}, \forall m \in \mathbf{M} \tag{3.10}$$

$$y_i \leq 1 \qquad \forall i \in \mathbf{N} \tag{3.11}$$

$$\sum_{i \in \mathbf{N}} y_i = \theta \tag{3.12}$$

$$y_i, z_{smi}, x_{smij} \geq 0 \qquad \forall s \in \mathbf{S}, \forall i \in \mathbf{N}, \forall \{i,j\} \in \mathbf{A}_s, \forall m \in \mathbf{M}. \tag{3.13}$$

The proposed modelling framework is of the form of a two-stage stochastic program. The examples of successful applications of this form to various problem types are abundant in literature. Readers are referred to Noyan (2008) and Barbarosoglu and Arda (2004) for applications in disaster management, Li et al. (2006) and Maqsood et al. (2005) for applications in water resources management, Liu et al. (2009) for applications in transportation network protection, Zhou et al. (2013) for applications in energy systems design and Tsiakis et al. (2001) for applications in supply chain network design problems.

The proposed model can be applied to the problems with arbitrarily shaped probabilistic barriers. It is also possible to model problems considering different distance metrics such as Euclidean and rectilinear distance metrics. The deterministic problem arises as a special case for $|\mathbf{S}| = 1$. When the number of scenarios increases, the size of the model will increase which in turn will increase the complexity of solving the problem. To overcome this difficulty, an exact algorithm based on Benders decomposition is described in the next section.

3.5 Benders Decomposition

3.5.1 Overview

Benders decomposition (Benders 1962) is a technique to efficiently solve a problem to optimality by decomposing a model into a subproblem and a master problem and iteratively solving the two to identify an optimal solution to the original model. While decomposing the model, the aim is to keep complicating variables (i.e. integer variables) in the master problem. To describe Benders decomposition in more detail, consider the following notation of the mixed integer linear program (MILP).

$$\text{minimise} \quad c^T x + f^T y \tag{3.14}$$

subject to

$$Ax + By = b \tag{3.15}$$

$$y \in \mathbb{Z}^+ \cap Y \tag{3.16}$$

$$x \geq 0. \tag{3.17}$$

where x is a vector of continuous variables having dimension p and y is a vector of integer variables having dimension q, Y is a polyhedron, A and B are matrices, and b, c and f are vectors with appropriate dimensions. Since y-variables are integer in this notation, it is possible categorise them as complicating variables. If y-variables are fixed to $\bar{y}$, it can clearly be seen that solving the obtained problem is significantly easier, since the model reduces to an LP. This LP is named as a subproblem, which only has continuous variables. The subproblem is:

$$\min_{\bar{y} \in Y} \{ f^T \bar{y} + \min_{x \geq 0} \{ c^T x : Ax = b - B\bar{y} \} \}. \tag{3.18}$$

Since the inner minimisation in (3.18) is a continuous linear problem, it is possible to dualise by using dual variables π associated with the constraint set of the inner minimisation:

$$\min_{\bar{y} \in Y} \{ f^T \bar{y} + \max_{\pi \in \mathbb{R}} \{ (b - B\bar{y})^T \pi : A^T \pi \leq c \} \}. \tag{3.19}$$

The feasible space of the dual problem is independent from the choice of $\bar{y}$. Assuming that the dual feasible space is not empty, the dual problem can either be unbounded or feasible for any $\bar{y}$. It is possible to enumerate all extreme rays, $(\gamma_1, ..., \gamma_{|\mathbf{Q}|})$, of the dual feasible space, where $\mathbf{Q}$ is the set of extreme rays. To avoid the unboundedness due to the choice of $\bar{y}$, the following cut, named as a feasibility cut, can be added:

$$\gamma_q^T (b - B\bar{y}) \leq 0 \qquad \forall q \in \mathbf{Q}. \tag{3.20}$$

Like extreme rays, it is also possible to enumerate all extreme points, $(\pi_1, ..., \pi_{|\mathbf{E}|})$, of the dual feasible space, where $\mathbf{E}$ is the set of extreme points. After adding all the feasible cuts, the value of the dual problem one of these extreme points, since the feasibility will be guaranteed. Thus, the problem (3.19) can be reformulated as;

$$\min_{\bar{y} \in Y} f^T \bar{y} + \max_{e \in \mathbf{E}} \{ \pi_e^T (b - B\bar{y}) \} \tag{3.21}$$

subject to

$$\gamma_q^T (b - B\bar{y}) \leq 0 \qquad \forall q \in \mathbf{Q}. \tag{3.22}$$

This problem can be linearised by defining a continuous variable, $\alpha \in \mathbb{R}$, as shown below. The resulting model will be an equivalent formulation to the problem (3.14)-(3.17):

$$\text{minimise} \quad \alpha + f^T y \tag{3.23}$$

$$\alpha \geq \pi_e^t(b - By) \qquad \forall e \in \mathbf{E} \tag{3.24}$$

$$\gamma_q^t(b - B\bar{y}) \leq 0 \qquad \forall q \in \mathbf{Q} \tag{3.25}$$

$$y \in \mathbb{Z}^+ \cap Y. \tag{3.26}$$

This problem is referred to as a Master Problem (MP). As it can be seen that the formulation has only the complicating variables from the original formulation. The constraint (3.24) is the optimality cuts. It is possible to enumerate all cuts to solve the model, but it is computationally expensive. Instead of enumerating all cuts, Benders (1962) introduced an approach based on the relaxation of both optimality and feasibility cuts in the MP. The Benders decomposition algorithm starts by solving relaxed MP to obtain $\bar{y}$ and a lower bound for the optimal solution, because of the relaxation. The obtained $\bar{y}$ is then used to solve the subproblem (3.19) to reach an upper bound for the optimal solution and an information needed to generate optimality or feasibility cuts, if the subproblem is unbounded. These cuts added to the relaxed MP in the next iteration. This iterative process continues until the lower bound and upper bound converge to reach optimality.

In the next section, I show how standard Benders decomposition can be applied to the model DESM.

3.5.2 Application of Benders Decomposition to DESM

To illustrate the application of the technique, I will use the objective of a median problem as the objective function for DESM. This formulation yields the following Benders subproblem (SP) when variables y_i are fixed to $\bar{Y} = \{\bar{y}_i, i \in \mathbf{N}\}$, such that $\bar{y}_i \in \{0, 1\}$:

$$\sum_{i \in \mathsf{N}} \bar{y}_i = \theta \tag{3.27}$$

$$\text{minimise} \sum_{s \in \mathsf{S}} \sum_{m \in \mathsf{M}} \sum_{\{i,j\} \in \mathsf{A}_s} v_m p_s c_{smij} x_{smij} \tag{3.28}$$

subject to

$$\sum_{j:(i,j) \in \mathbf{A}_s} x_{smji} - \sum_{j:(i,j) \in \mathbf{A}_s} x_{smij} = \begin{cases} -1 & \text{if } i = O(m), \\ z_{smi} & \text{otherwise;} \end{cases} \qquad \forall s \in \mathbf{S}, \forall i \in \mathbf{N}, m \in \mathbf{M} \tag{3.29}$$

$$z_{smi} \leq \bar{y}_i \qquad \forall s \in \mathbf{S}, \forall i \in \mathbf{N}, \forall m \in \mathbf{M} \tag{3.30}$$

$$z_{smi}, x_{smij} \geq 0 \qquad \forall s \in \mathbf{S}, \forall \{i,j\} \in \mathbf{A}_s, \forall m \in \mathbf{M}, \forall i \in \mathbf{N}. \tag{3.31}$$

SP further decomposes by each scenario $s \in \mathbf{S}$ and each commodity $m \in \mathbf{M}$, resulting in the following subproblem shown by FDSP$_{sm}(\bar{Y})$:

$$\text{minimise } v_m p_s \sum_{\{i,j\} \in \mathsf{A}_s} c_{smij} x_{smij} \tag{3.32}$$

subject to

$$\sum_{j:(i,j) \in \mathbf{A}_s} x_{smji} - \sum_{j:(i,j) \in \mathbf{A}_s} x_{smij} = \begin{cases} -1 & \text{if } i = O(m), \\ z_{smi} & \text{otherwise;} \end{cases} \qquad \forall i \in \mathbf{N} \tag{3.33}$$

$$z_{smi} \leq \bar{y}_i \qquad \forall i \in \mathbf{N}, \tag{3.34}$$

$$z_{smi}, x_{smij}, x_{smji} \geq 0 \qquad \forall \{i,j\} \in \mathbf{A}_s. \tag{3.35}$$

Formulation of FDSP$_{sm}(\bar{Y})$ (3.32) – (3.35) is the mathematical formulation of the well-known shortest path problem. In other words, since the locations of new facilities in SP are fixed by the $\bar{y}$ variables, the optimal objective value of SP is the summation of the weighted shortest path distances from demand points to the closest new facilities for each scenario $s \in \mathbf{S}$.

The dual $\mathrm{DFDSP}_{sm}(\bar{Y})$ of $\mathrm{FDSP}_{\mathrm{sm}}(\bar{Y})$ for each $s \in \mathbf{S}$ and $m \in \mathbf{M}$ is:

$$\text{maximise} \sum_{i \in \mathbf{N}} \bar{y}_i w_{smi} - \sum_{\substack{i \in \mathbf{N} \\ i = O(m)}} u_{smi} \tag{3.36}$$

subject to

$$u_{smj} - u_{smi} \leq v_m p_s c_{smij} \qquad \forall \{i, j\} \in \mathbf{A}_s \tag{3.37}$$

$$w_{smi} - u_{smi} \leq 0 \qquad \text{if } i \neq O(m), \forall i \in \mathbf{N}, \tag{3.38}$$

$$w_{smi} \leq 0 \qquad \forall i \in \mathbf{N}. \tag{3.39}$$

Using optimal solutions $(\bar{U}, \bar{W}) = \{(\bar{u}_{smi}, s \in \mathbf{S}, m \in \mathbf{M}, i \in \mathbf{N}), (\bar{w}_{smi}, s \in \mathbf{S}, m \in \mathbf{M}, i \in \mathbf{N})\}$ from DFDSP_{sm}, the relaxed Benders Master Problem (MP) for the restricted stochastic facility location problem can be stated as follows:

$$\text{minimise} \sum_{s \in \mathbf{S}} \sum_{m \in \mathbf{M}} \alpha_{sm} \tag{3.40}$$

subject to

$$\alpha_{sm} \geq \sum_{i \in \mathbf{N}} y_i \bar{w}^{(k)}_{smi} - \sum_{\substack{i \in \mathbf{N} \\ i = O(m)}} \bar{u}^{(k)}_{smi} \qquad \forall k \in \mathbf{K}_{sm}, \forall s \in \mathbf{S}, \forall m \in \mathbf{M} \tag{3.41}$$

$$\sum_{i \in \mathbf{N}} y_i \bar{w}^{(l)}_{smi} - \sum_{\substack{i \in \mathbf{N} \\ i = O(m)}} \bar{u}^{(l)}_{smi} \leq 0 \qquad \forall l \in \mathbf{L}_{sm}, \forall s \in \mathbf{S}, \forall m \in \mathbf{M} \tag{3.42}$$

$$\sum_{i \in \mathbf{N}} y_i = \theta \tag{3.43}$$

$$0 \leq y_i \leq 1 \qquad \forall i \in \mathbf{N}, \tag{3.44}$$

where α_{sm} is a continuous variable and $\mathbf{K}_{sm}$ and $\mathbf{L}_{sm}$ are the subsets of extreme points and extreme rays of feasible space of the DFDSP_{sm} for each $s \in \mathbf{S}$ and $m \in \mathbf{M}$, respectively. The constraints (3.41) are defined for each extreme point in the feasible region of the DFDSP_{sm} and are known as Benders optimality cuts. The constraints (3.42) are defined for each extreme ray of the $\mathrm{DFDSP}_{\mathrm{sm}}$ for each $s \in \mathbf{S}$ and $m \in \mathbf{M}$, whenever it is infeasible and are named Benders feasibility cuts. For the problem at hand, it is possible to replace the Benders feasibility cuts with the following constraint in the MP in each iteration.

$$\sum_{i \in \mathbf{N}_b} y_i = 0 \qquad \forall b \in \mathbf{B}, \tag{3.45}$$

where $\mathbf{N}_b$ is the subset of nodes that fall within the barriers and $\mathbf{B}$ is the set of barriers. Constraints (3.45) in the MP ensure that any solution $\bar{Y}$ will be feasible with respect to constraints (3.44), since they forbid locating any new facilities within the barriers. Thus, with constraints (3.45), there will always be a feasible $\text{FDSP}_{\text{sm}}(\bar{Y})$ and $\text{DFDSP}_{sm}(\bar{Y})$, $\forall s \in \mathbf{S}, m \in \mathbf{M}$, and therefore always an optimal solution to these models, meaning that the algorithm will only require Benders optimality cuts to converge to an optimal solution to the original problem.

The Benders decomposition algorithm starts with a relaxed $MP(Initial)$ with a (possibly empty) subset of constraints (3.41) and iterates between the MP and the subproblems until an optimal solution is identified. In each iteration of the algorithm, the relaxed MP yields a lower bound (LB) with a solution $\bar{Y}$ for the global optimum of the original problem, whereas the summation of the objective values of each DFDSP_{sm} yields an upper bound (UB) to the optimal value of the original model with solutions $(\bar{U}, \bar{W})$. Pseudo-code of the standard Benders decomposition algorithm (SBDA) is presented below.

Algorithm 1 SBDA for Restricted Facility Location Problems

$NCut$: Number of Iterations; $MaxCut$: Maximum Number of Cuts; $z[NCut]$: Optimal value of MP at iteration $NCut$.

Initialization: $NCut \longleftarrow 0$.

START

Solve $MP(Initial)$. Let the solution be $\bar{Y}$ and the optimal value be $v_{MP}(\bar{U}, \bar{W})$.

$z[NCut] \longleftarrow v_{MP}(\bar{U}, \bar{W})$

While $(NCut < MaxCut)$

Solve each $DFDSP_{sm}\left(\bar{Y}\right)$ $\forall s \in \mathsf{S}$ and $m \in \mathsf{M}$.

Let the solutions be $\left(\bar{U}, \bar{W}\right)_{sm}$ and the optimal values be $v_{FDSP_{sm}}(\bar{Y})$

$\forall s \in \mathsf{S}$, $m \in \mathsf{M}$.

$v_{SP}(\bar{Y}) = \sum\limits_{s \in \mathsf{S}} \sum\limits_{m \in \mathsf{M}} v_{FDSP_{sm}}(\bar{Y})$

If $(v_{SP}(\bar{Y}) <= z[NCut] + \epsilon)$

$STOP$

$NCut \longleftarrow NCut + 1$

Solve $MP\left(\bar{U}, \bar{W}\right)$. Let the solution be $\bar{Y}$ and the optimal value be

$v_{MP}(\bar{U}, \bar{W})$.

$z[NCut] \longleftarrow v_{MP}(\bar{U}, \bar{W})$

End While

END

For a more efficient algorithm, $\mathrm{FDSP_{sm}}(\bar{Y})$ can be solved by Dijkstra's algorithm for each commodity $m \in \mathsf{M}$, under each scenario $s \in \mathsf{S}$ in each iteration of the Benders decomposition algorithm. Dijkstra's algorithm finds the shortest distance between each node pair $(i, j) \in A_s$ in the given network (Dijkstra 1959). I refer to Benders decomposition algorithm using Dijkstra's algorithm to solve the shortest path problems as DBDA.

Proposition 3.1 states and proves the way to obtain the dual solution needed in the MP directly from using Dijkstra's algorithm.

Proposition 3.1. *Given $\bar{y}_i, \forall i \in \mathsf{N}$, and optimal solutions $x^\star = \{x^\star_{smij} | \forall s \in \mathsf{S}, m \in \mathsf{M}, (i,j) \in \mathsf{A}_s\}$ and $z^\star = \{z^\star_{smi} | \forall s \in \mathsf{S}, m \in \mathsf{M}, i \in \mathsf{N}\}$ to the SP, then an optimal* *solution* $(u^\star, w^\star)$, *where* $u^\star = \{u^\star_{smi} | \forall s \in \mathbf{S}, m \in \mathbf{M}, i \in \mathbf{N}\}$ *and* $w^\star = \{w^\star_{smi} | \forall s \in \mathbf{S}, m \in \mathbf{M}, i \in \mathbf{N}\}$, *to the DFDSP is one that satisfies:*

$$u^\star_{smj} - u^\star_{smi} = v_m p_s c_{smij}, \qquad \text{if } x^\star_{smij} = 1, \forall s \in \mathbf{S}, m \in \mathbf{M}, i, j \in \mathbf{N} \tag{3.46}$$

$$u^\star_{smj} - u^\star_{smi} \leq v_m p_s c_{smij}, \qquad \text{if } x^\star_{smij} = 0, \forall s \in \mathbf{S}, m \in \mathbf{M}, i, j \in \mathbf{N} \tag{3.47}$$

$$w^\star_{smi} = 0, \qquad \text{if } \bar{y}_i = 1, \forall s \in \mathbf{S}, m \in \mathbf{M}, i \in \mathbf{N} \tag{3.48}$$

$$w^\star_{smi} - u^\star_{smi} \leq 0 \qquad \text{if } \bar{y}_i = 0 \text{ and } i \neq O(m), \forall i \in \mathbf{N}, \tag{3.49}$$

$$w^\star_{smi} \leq 0 \qquad \text{if } \bar{y}_i = 0, \forall i \in \mathbf{N}. \tag{3.50}$$

Proof. Point 1 below proves equations (3.46) and inequalities (3.47). Point 2 presents the proof for the equation (3.48) and point 3 is the proof for the inequalities (3.49) and (3.50).

1. If the shortest path between the origin and destination nodes of commodity $m \in \mathbf{M}$ passes through the arc $(i, j) \in \mathbf{A}_s$ in scenario $s \in \mathbf{S}$, then $x^\star_{smij} = 1$, $\forall s \in \mathbf{S}, m \in \mathbf{M}, (i, j) \in \mathbf{A}_s$. Due to the complementary slackness, it is also known that $(u^\star_{smj} - u^\star_{smi} - v_m p_s c_{smij}) x^\star_{smij} = 0$, $\forall s \in \mathbf{S}, m \in \mathbf{M}, i, j \in \mathbf{N}$. Since, $x^\star_{smij} = 1$, $\forall s \in \mathbf{S}, m \in \mathbf{M}, (i, j) \in \mathbf{A}_s$, the equation, $u^\star_{smj} - u^\star_{smi} = v_m p_s c_{smij}$, $\forall s \in \mathbf{S}, m \in \mathbf{M}, i, j \in \mathbf{N}$ must hold. Let $u^\star_{smi} = 0$, if $\bar{y}_i = 1$, $\forall i \in \mathbf{N}$ and $z^\star_{smi} = 1$, $\forall s \in \mathbf{S}, m \in \mathbf{M}, i \in \mathbf{N}$. This means that the source node of the shortest path between the new facility that is serving the demand point $m \in \mathbf{M}$ and the demand point $m \in \mathbf{M}$ is the node where this new facility is located. If $x^\star_{smij} = 0, \forall s \in \mathbf{S}, m \in \mathbf{M}, (i, j) \in \mathbf{A}_s$, then there is not any flow passing through the arc $(i, j) \in \mathbf{A}_s$, which means any feasible u_{smi} in this condition will not effect the optimality.

2. If $\bar{y}_i = 1$, then $z^{\star}_{smi}$, $\forall s \in \mathbf{S}, m \in \mathbf{M}, i \in \mathbf{N}$ can either be 0 or 1, due to constraint set (3.3) and integrality property. First, consider the case where $z^{\star}_{smi} = 0$, $\forall s \in \mathbf{S}, m \in \mathbf{M}, i \in \mathbf{N}$. Due to complementary slackness, $(v^{\star}_{smi} - \bar{y}_i)w^{\star}_{smi} = 0$, $\forall s \in \mathbf{S}, m \in \mathbf{M}, i \in \mathbf{N}$. In this case, this equation holds only if $w^{\star}_{smi} = 0$. Second, consider $z^{\star}_{smi} = 1$, $\forall s \in \mathbf{S}, m \in \mathbf{M}, i \in \mathbf{N}$. Due to complementary slackness, $(w^{\star}_{smi} - u^{\star}_{smi})v^{\star}_{smi} = 0$, where $i \neq O(m), \forall s \in \mathbf{S}, m \in \mathbf{M}, i \in \mathbf{N}$. Thus, $w^{\star}_{smi} = u^{\star}_{smi}$, where $z^{\star}_{smi} = 1$. As it is mentioned before, $u^{\star}_{smi} = 0$, if $\bar{y}_i = 1$, $\forall i \in \mathbf{N}$ and $z^{\star}_{smi} = 1$, $\forall s \in \mathbf{S}, m \in \mathbf{M}, i \in \mathbf{N}$. Similarly, if $\bar{y}_i = 1$, $\forall i \in \mathbf{N}$ and $z^{\star}_{smi} = 1$, $\forall s \in \mathbf{S}, m \in \mathbf{M}, i \in \mathbf{N}$, then $w^{\star}_{smi} = 0$, $\forall s \in \mathbf{S}, m \in \mathbf{M}, i \in \mathbf{N}$.
3. Consider the first summation term of the objective function (3.31) of the DFDSP. If $\bar{y}_i = 0$, $\forall i \in \mathbf{N}$, then any feasible value for w_{smi}, $\forall s \in \mathbf{S}, m \in \mathbf{M}$, will not effect the optimality, since $w^{\star}_{smi} = 0$, $\forall s \in \mathbf{S}, m \in \mathbf{M}$ provided that $\bar{y}_i = 1$, $\forall s \in \mathbf{S}, m \in \mathbf{M}, i \in \mathbf{N}$.

□ □

It is proven that either $\bar{y}_i$, $\forall i \in \mathbf{N}$, is 0 or 1, at optimality the first summation term of (3.31) will be zero. In this work, when $\bar{y}_i = 0$, $\forall i \in \mathbf{N}$, I use the tightest values for $w^{\star}_{smi}$, $\forall s \in \mathbf{S}, m \in \mathbf{M}, i \in \mathbf{N}$ by;

$$w^{\star}_{smi} = \begin{cases} u^{\star}_{smi} & \text{if } u^{\star}_{smi} \leq 0, \\ 0 & \text{otherwise;} \end{cases} \qquad \forall s \in \mathbf{S}, \forall i \in \mathbf{N}, m \in \mathbf{M} \tag{3.51}$$

There are various ways to improve the performance of the Benders decomposition algorithm. The Pareto-optimality cut approach (Magnanti and Wong 1981) is a way to generate stronger cuts for the master problem, which is expected to yield a decrease in the number of iterations to reach optimality, but requires an auxilary problem to be solved in each iteration. For further details on this approach, readers are referred to Cordeau et al. (2001), Papadakos (2008) and Papadakos (2009). This approach, referred as DPBDA, is tested in the following section.

3.6 Computational Results

In this section, I present results of computational experiments to show the efficiency of the proposed solution approaches. I first describe how the instances are generated in Section 3.6.1. The quality of the discrete network is tested in Section 3.6.2. Sections 3.6.3 and 3.6.4 present computational studies for both deterministic and stochastic restricted multi-facility location problems, respectively.

All experiments in Section 3.6.2 were run on a PC with an Intel(R) Core(TM) 2.60 GHz processor and 4.00 GB of RAM using CPLEX 12.5. Those reported in Sections 3.6.3 and 3.6.4 were solved on a single node of IRIDIS 4 computer cluster at the University of Southampton with dual 2.6 GHz Intel Sandybridge processors and 64 GB of RAM using CPLEX 12.5. Both SBDA and DBDA were implemented in C++.

3.6.1 Description of the Instances

In the existing literature, I have identified three restricted facility location problem instances with barriers and four instances with forbidden regions. Tables 3.1 and 3.2 show the characteristics of these instances including their source, type, number of demand points ($|\mathsf{M}|$), the type of restricted regions (RRT), the distance metric and the names for use in the remainder of the paper.

TABLE 3.1: Barrier instances

Name	Source	Type	$\|\mathsf{M}\|$	RRT	Distance
B1	Katz and Cooper (1981)	1-median	5	Circular	Euclidean
B2	Katz and Cooper (1981)	1-median	10	Circular	Euclidean
B3	Aneja and Parlar (1994)	1-median	18	Polygonal	Euclidean

TABLE 3.2: Forbidden region instances

Name	Source	Type	$\|\mathsf{M}\|$	RRT	Distance
FR1	Aneja and Parlar (1994)	1-median	4	Polygonal	Euclidean
FR2	Hamacher and Nickel (1994)	1-median	3	Polygonal	Euclidean
FR3	Hamacher and Nickel (1994)	1-median	3	Polygonal	Rectilinear
FR4	Hamacher and Nickel (1994)	1-median	3	Polygonal	Chebyshev

These instances are shown in Tables 3.1 and 3.2 are that of single facility location and used to evaluate the quality of the discretisation presented in Section 3.4.2.

As for multi-facility instances, I have generated both deterministic and stochastic instances using a complex barrier instance described by Aneja and Parlar (1994), shown in Figure 3.4, where each barrier is identified by the number shown in it.

In this instance, there are 12 polygonal barriers and 18 demand points, which are located at coordinates $(1, 2)$, $(6, 1)$, $(9, 1)$, $(14, 2)$, $(5, 5)$, $(7, 4)$, $(9, 5)$, $(14, 4)$, $(17, 4)$, $(2, 8)$, $(8, 8)$, $(16, 8)$, $(3, 12)$, $(6, 11)$, $(9, 10)$, $(17, 10)$, $(10, 12)$ and $(19, 13)$. The distance metric is Euclidean. New instances were generated by considering 36, 54 and 72 demand points. For the 36 demand points case, 18 additional points were located on coordinates $(19, 2)$, $(2, 14)$, $(4, 9)$, $(5, 11)$, $(7, 11)$, $(8, 6)$, $(9, 12)$, $(10, 8)$, $(12, 1)$, $(12, 13)$, $(13, 14)$, $(15, 7)$, $(16, 4)$, $(17, 5)$, $(18, 1)$, $(19, 5)$, $(19, 9)$ and $(19, 10)$. For the 54 demand points case, a further 18 additional points located on coordinates $(1, 6)$, $(1, 10)$, $(3, 6)$, $(3, 1)$, $(4, 4)$, $(4, 14)$, $(6, 3)$, $(7, 8)$, $(8, 12)$, $(9, 7)$, $(10, 4)$, $(11, 8)$, $(13, 6)$, $(14, 14)$, $(15, 1)$, $(16, 12)$, $(17, 5)$ and $(5, 6)$ were added to the instance with 36 demand points. Finally, the 72 demand point instance was constructed by adding a new set of 18 demand points located at coordinates $(1, 13.5)$, $(3, 11)$, $(3, 14)$, $(4, 1)$, $(5, 10)$, $(9, 8)$, $(9, 14)$, $(10, 3)$, $(10, 11)$, $(11, 14)$, $(12, 6)$, $(13, 13)$, $(14, 6)$, $(15, 14)$, $(17, 14)$, $(19, 1)$, $(10, 12)$ and $(19, 5)$ to the instance with 54 demand points.

For the stochastic instances, three different probability sets were generated corresponding to the occurrence of barriers. Table 3.3 shows the number of scenarios (NS), the corresponding barriers and respective probabilities for each probability set. For example, in the first scenario of NS $= 2$, the barriers numbered 1, 3, 5, 7, 9 and 11, occur with 50% probability in probability set 1, with 60% probability in probability set 2, and with 80% probability in probability set 3, whereas in the second scenario of NS $= 2$, barriers numbered 2, 4, 6, 8, 10 and 12 in Figure 3.4 occur with probabilities of 50%, 40% and 20% in probability sets 1, 2 and 3, respectively. The rest of the scenarios and their related probabilities can be found in Table 3.3.

With three probability sets, four instance sizes (18, 36, 54, 72 demand points), six various θ values ($\theta = 1, ..., 6$), and five different combinations of scenarios, the total number of stochastic instances generated is $4 \times 6 \times 3 \times 5 = 360$.

TABLE 3.3: Probability sets for the stochastic restricted facility location instances

NS	Scenarios(s)	Barriers	p_s (Set 1)	p_s (Set 2)	p_s (Set 3)
2	1	1-3-5-7-9-11	0.5	0.6	0.8
	2	2-4-6-8-10-12	0.5	0.4	0.2
3	1	1-4-7-10	0.333	0.6	0.4
	2	2-5-8-11	0.333	0.2	0.3
	3	3-6-9-12	0.333	0.2	0.3
4	1	1-5-9	0.25	0.3	0.4
	2	2-6-10	0.25	0.3	0.1
	3	3-7-11	0.25	0.2	0.4
	4	4-8-12	0.25	0.2	0.1
5	1	1-6-11	0.2	0.3	0.4
	2	2-7-12	0.2	0.2	0.2
	3	3-8	0.2	0.1	0.1
	4	4-9	0.2	0.3	0.1
	5	5-10	0.2	0.1	0.2
6	1	1-7	0.1667	0.2	0.4
	2	2-8	0.1667	0.1	0.2
	3	3-9	0.1667	0.2	0.1
	4	4-10	0.1667	0.1	0.1
	5	5-11	0.1667	0.3	0.1
	6	6-12	0.1667	0.1	0.1

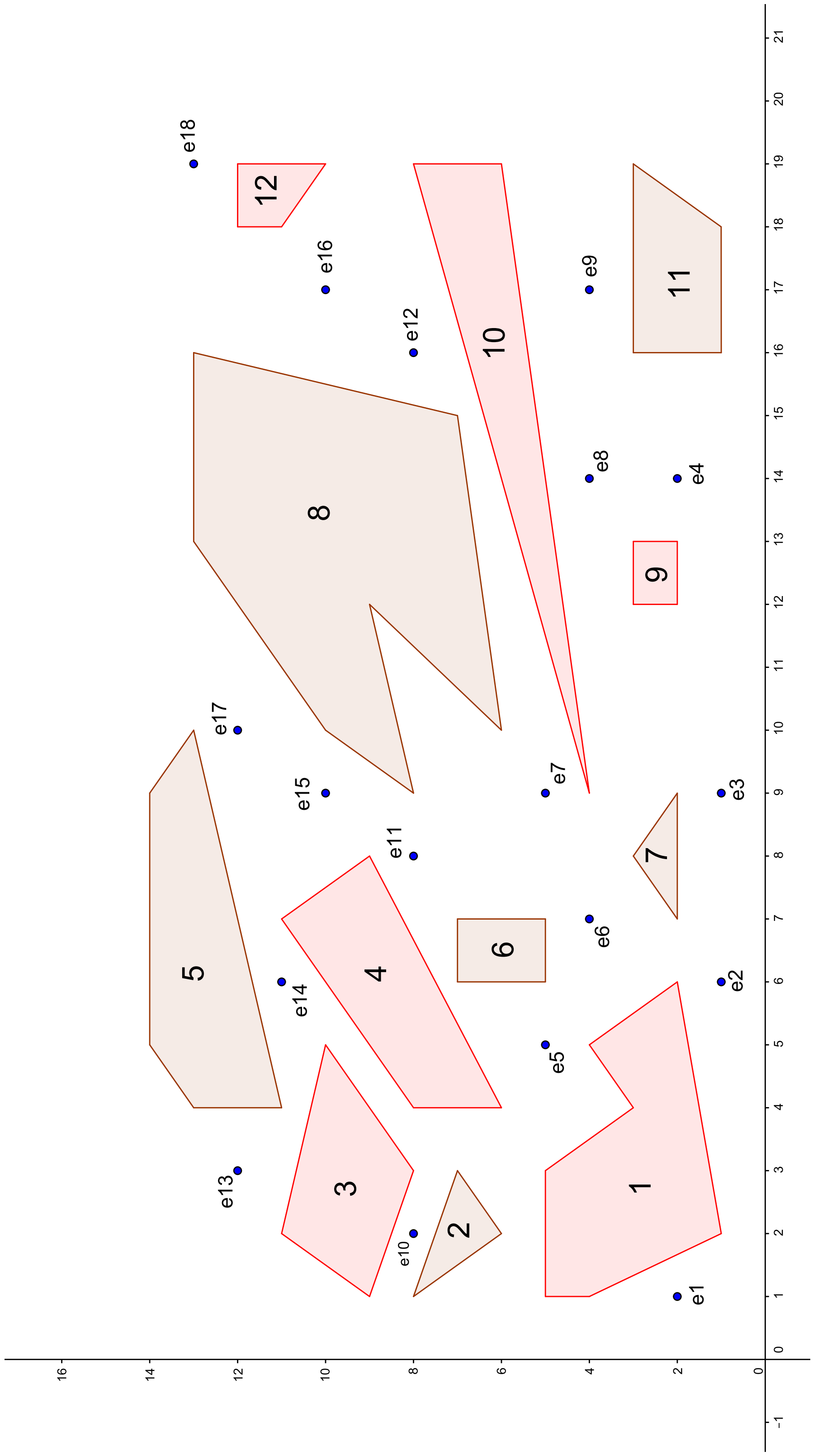


FIGURE 3.4: Aneja and Parlar (1994) barrier instance

3.6.2 Quality of the Discretisation

In this section, I assess the quality of the two discretisations, namely the quality of both a unit square network and a unit rectangle network described in Section 3.4.1, by solving the deterministic instances shown in Tables 3.1 and 3.2. Solutions to the continuous problems were obtained using the models proposed by Oğuz et al. (2016). The solutions obtained from the discretisation and the original solutions are compared by analyzing the percentage gap between them.

3.6.2.1 Unit Square Network

As a preliminary test, I initially consider instance B1 and solve it using different discretisation points. The best-known objective value for this instance is 48.257 and is confirmed by several papers such as Butt and Cavalier (1996), Klamroth (2001), Bischoff and Klamroth (2007), Klamroth (2004) and Oğuz et al. (2016), which allows us to test the quality of a unit square network approximation. Table 3.4 presents the number of nodes (NN) used in the unit square network and the percentage gap (PG) between the best-known objective value of the continuous problem (OVCP) and the optimal objective value of the discretised problem (OVDP), calculated as $(100 \times \dfrac{OVDP - OVCP}{OVCP})$.

Table 3.4: B1 results with unit square network

NN	PG
400	6.76%
1600	5.74%
6400	5.68%
25,600	5.59%
102,400	5%

Table 3.4 and Figure 3.5 show that increasing the number of nodes improves the approximation. However, even with $102,400$ nodes on the network, the percentage gap is 5%, which is unsatisfactory for the purposes of deriving an exact algorithm. The results suggest that the unit square network discretisation is inefficient at producing near-optimal approximations.

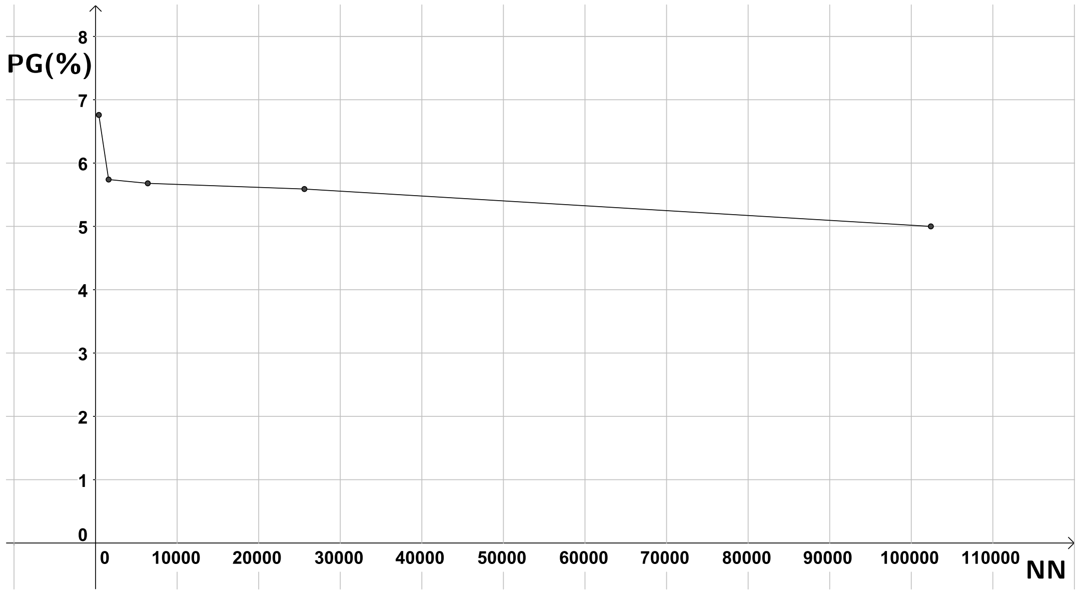


FIGURE 3.5: Percentage gap vs. number of nodes for Katz and Cooper (1981) first instance modeled on unit square network

3.6.2.2 Unit Rectangle Network

Similar to Section 3.6.2.1, B1 is solved by using a unit rectangle network. Table 3.5 shows that the percentage gaps are much lower compared the unit square network. The advantages of using the unit rectangle network are;

TABLE 3.5: B1 results with unit rectangle network

NN	PG
400	1.2%
1600	0.67%

- With 1600 nodes on the unit rectangle network, the percentage gap is less than 1%, which is satisfactory for deriving an exact algorithm, in contrast to 5.74% obtained using the unit square network.
- Compared to the unit square network, a fewer number of nodes is needed to achieve satisfactory percentage gaps for the unit rectangle network. This leads to a significant drop in the size of the model, which increases the performance of the solution algorithm.

To further test the quality of the unit rectangle network, the rest of the instances shown in Tables 3.1 and 3.2 are discretised and solved using this network.

Each of the instances described in Table 3.1 was constructed with 400 and 1600 nodes in the resulting graph. An objective value of 88.326 is the best-known solution for B2 reported by Bischoff and Klamroth (2007) and Oğuz et al. (2016). For B3, 119.139 is the best-known objective value, which is reported by Aneja and Parlar (1994), Bischoff and Klamroth (2007) and Oğuz et al. (2016). Table 3.6 presents comparative results.

Table 3.6: Barrier instances results comparison

Name	NN	OVCP	OVDP	PG
B1	400	48.257	48.817	1.2%
B1	1600	48.257	48.581	0.67%
B2	400	88.326	90.188	2.1%
B2	1600	88.326	89.218	0.99%
B3	400	119.139	120.573	1.2%
B3	1600	119.139	119.53	0.33%

The forbidden region instances are solved in the same way using two networks of 400 and 1600 nodes, except FR1. In order for there to be a meaningful comparison, the unit cost of each arc has to be the same for all instances. To discretise FR1 under this condition, a larger network is needed. Instead of 400 and 1600 nodes network, 484 and 1936 nodes network are used to discretise FR1. In other words, diagonals assumed to have the same cost. The best known solution for instance FR1 is found by Aneja and Parlar (1994) and Oğuz et al. (2016), and the best known solutions for instances FR2-FR4 are confirmed by Hamacher and Nickel (1994) and Oğuz et al. (2016). Table 3.7 shows the comparison results.

Table 3.7: Forbidden region instances results comparison

Name	NN	OVCP	OVDP	PG
FR1	484	48.5	48.738	0.49%
FR1	1936	48.5	48.655	0.32%
FR2	400	8.566	8.595	0.33%
FR2	1600	8.566	8.594	0.32%
FR3	400	10	10	0%
FR3	1600	10	10	0%
FR4	400	8	8	0%
FR4	1600	8	8	0%

Both Tables 3.6 and 3.7 confirm that all instances in Tables 3.1 and 3.2 solved using the unit rectangle network have satisfactory gaps. It is also shown that the increase in number of nodes in the network leads to a better approximation, with gaps that are all below 1%. For this reason, all results presented in the following section will use the unit rectangle network with 1600 nodes.

3.6.3 Deterministic Restricted Multi-facility Location Problem Instances

I discretised the continuous multi-facility instances derived from Aneja and Parlar (1994) by using the unit rectangle network with 1600 nodes. These are the deterministic instances, which arise as a special case of the stochastic restricted facility location problem in which there is a single scenario. A total of 24 instances were generated and solved for the deterministic case. To solve the resulting model, I used four different methods, namely DESM using CPLEX 12.5, SBDA, DBDA and DPBDA. For all 24 deterministic instances, Table 3.8 presents the total number of demand points considered in each instance, the number of new facilities (θ), the optimal objective value ($v^\star$), the total CPU time in seconds (t_C) spent by CPLEX 12.5 in solving formulation DESM, the total solution CPU time in seconds (t_S) required by SBDA, the total CPU time in seconds (t_D) required by DBDA, the average CPU time in seconds (t_M) for the master problem in each iteration of DBDA, the total number of iterations (η_i) in DBDA, the total CPU time in seconds (t_P) needed to solve each instance by DPBDA, and the total number of iterations (η_i) for DPBDA.

TABLE 3.8: Results for the deterministic restricted multi-facility location instances

$\lvert\mathbf{M}\rvert$	θ	$v^\star$	t_C	t_S	t_D	t_M	η_i	t_P	η_p
18	1	119.53	29.02	171.03	1.21	0.008	3	296.43	2
	2	90.68	33.35	220.16	2.01	0.008	5	372.89	4
	3	66.38	30.91	210.38	2.37	0.008	6	366.93	5
	4	50.22	29.08	232.39	2.57	0.004	6	559.1	5
	5	42.64	29.06	228.21	3.74	0.005	9	427.2	8
	6	36.56	28.42	236.08	4.04	0.004	9	370.18	6
36	1	270.69	66.33	461.85	3.21	0.013	4	697.49	2
	2	176.51	63.73	640.33	5.45	0.012	7	1272.07	6
	3	139.9	63.58	1089.96	7.35	0.014	7	966.63	7
	4	111.91	60.11	768.49	8.13	0.011	9	937.38	6
	5	95.53	58.48	725.3	6.91	0.011	8	952.4	6
	6	86.06	57.74	902.67	18.86	0.012	17	1212.22	10
54	1	388.48	94.28	727.02	3.43	0.015	3	1257.49	2
	2	279.05	107.71	1101.82	12.55	0.014	8	2650.29	6
	3	222.12	99.37	1863.27	12.68	0.013	8	2525.77	9
	4	181.24	90.05	1744.21	12.53	0.016	11	1653.59	7
	5	159.52	89.98	2370.45	16.85	0.017	13	2965.54	13
	6	141.09	90.61	2215.2	16.87	0.018	11	1890.73	9
72	1	538.08	140.56	1438.24	4.65	0.02	3	2153.84	2
	2	395.52	153.47	2235.6	9.21	0.02	6	2801.23	5
	3	314.2	151.79	2947.55	17.75	0.019	9	2990.04	7
	4	256.98	128.24	2862.33	13.52	0.017	8	3521.19	8
	5	224.57	133.44	3519.89	22.68	0.018	14	LIMIT	0
	6	196.95	119.61	2392.38	20.33	0.014	9	LIMIT	0

The bar chart in Figure 3.6 shows the average number of iterations required by the DBDA with a purple bar and by the DPBDA with a red bar to reach optimality for instances with 18, 36, 54 and 72 demand points. As expected, the stronger cuts generated by the DPBDA require a fewer iterations as compared to the number of iterations in the DBDA. The instances with 72 demand points with 5 and 6 new facilities are not included in the bar chart, since these instances could not be solved by the DPBDA, due to the limited RAM available.

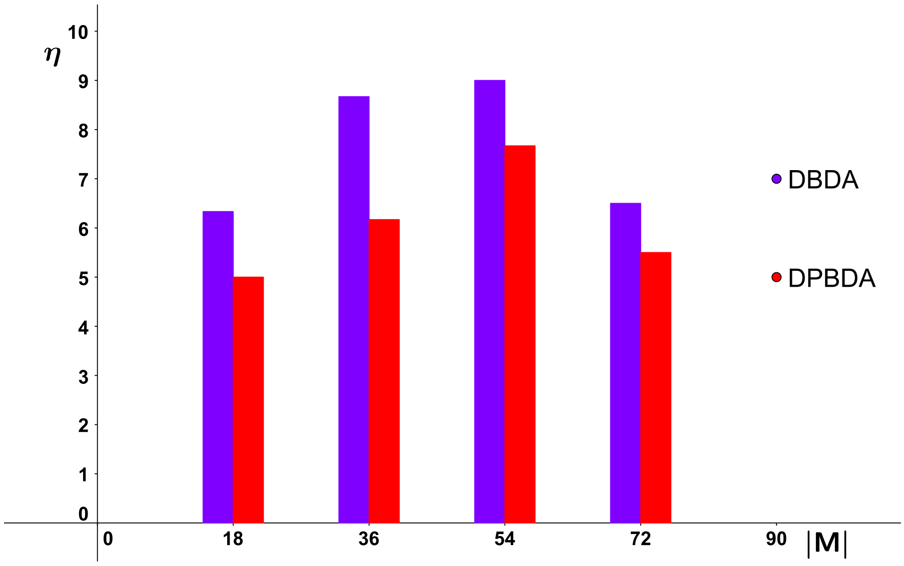


Figure 3.6: Average number of iterations vs. number of demand points

Table 3.9 provides average CPU times in seconds for solving instances with 18, 36, 54 and 72 demand points by CPLEX 12.5, SBDA, DBDA and DPBDA. The average CPU times are also visualised by line charts in Figures 3.7a-3.7d for 18, 36, 54, and 72 demand points, respectively. In Figure 3.7d, the CPU times obtained by the DPBDA are not included since two of the instances could not be solved due to the RAM availability, as mentioned before. The average savings in time between CPLEX 12.5 and the DBDA (SAVC), and the SBDA and the DBDA (SAVS) are also presented in Table 3.9.

Table 3.9: Average savings gained from the DBDA over the SBDA and CPLEX 12.5

$\lvert M \rvert$	AVG. t_C	AVG. t_S	AVG. t_D	AVG. t_P	SAVC	SAVS	SAVP
18	29.97	216.38	2.66	398.79	91.14%	98.77%	99.33%
36	61.83	764.77	8.32	1006.37	86.55%	98.91%	99.17%
54	95.33	1670.33	12.49	2157.24	86.90%	99.25%	99.42%
72	137.85	2566	14.69		89.34%	99.43%	

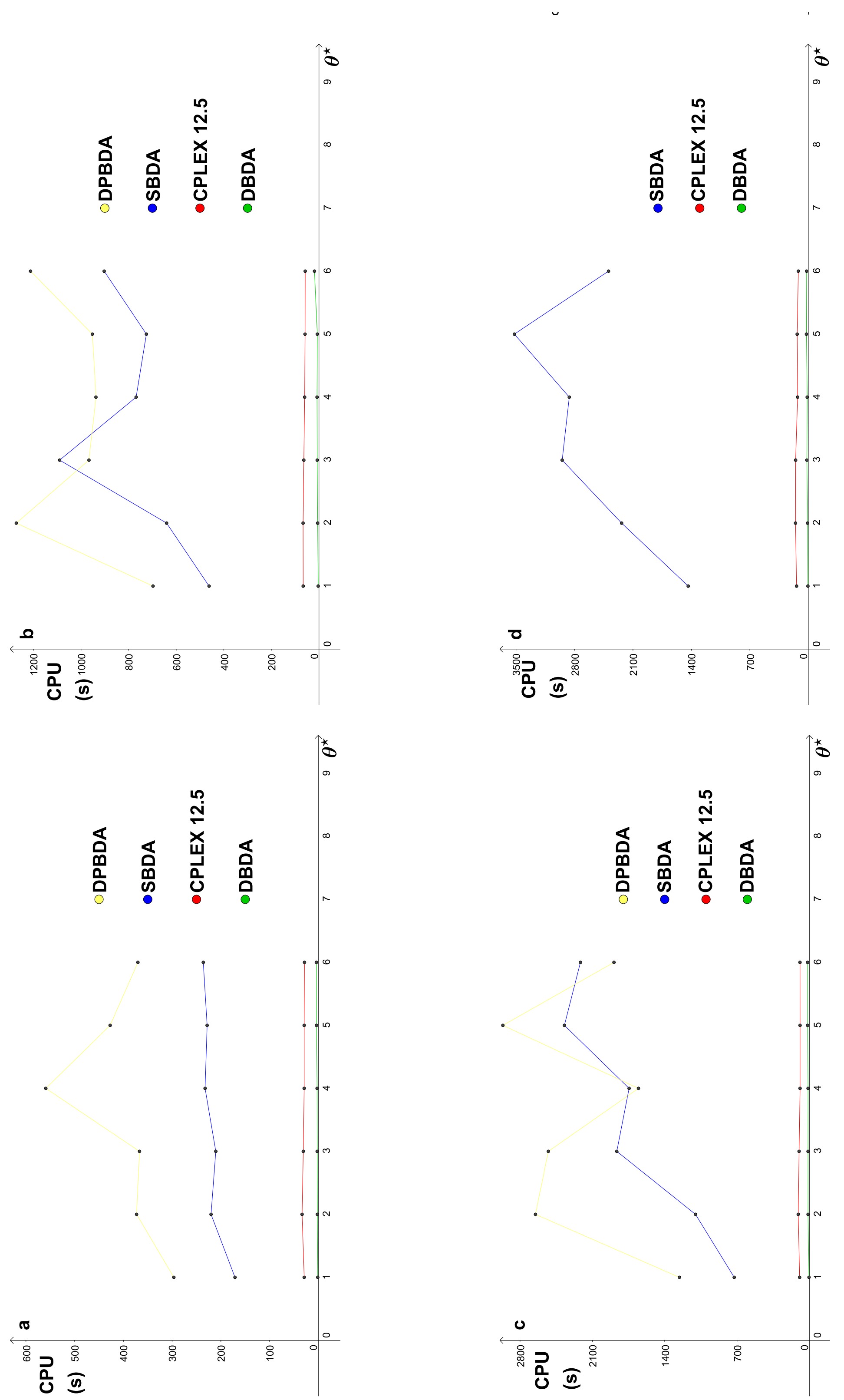


Figure 3.7: CPU time comparison over the proposed solution techniques

Despite a higher number of iterations for DBDA compared to DPBDA, Figures 3.7a-3.7d and Table 3.9 show that the average solution CPU times for instances obtained by DBDA are much less than the times obtained for CPLEX 12.5, SBDA and DPBDA. The average savings by DBDA are more than 85%, 98% and 99% over CPLEX 12.5, SBDA and DPBDA, respectively. These results confirm that the general performance of DBDA is superior compared to the other three techniques.

For the instances, Table 3.10 shows the average for t_M in each iteration, the average for η_i, the average total t_M for DBDA, the average for t_D, and the ratio between average total t_M and average t_D, which indicates the solution time needed for the MP as a fraction of the total solution time.

Table 3.10: MP average solution times

$\|\mathsf{M}\|$	AVG. t_M	AVG. η_i	AVG. Total t_M	AVG. t_D	Ratio
18	0.006	6.333	0.038	2.66	1.42%
36	0.012	8.667	0.106	8.32	1.28%
54	0.015	9	0.139	12.49	1.11%
72	0.018	8.167	0.147	14.69	1%

As can be seen in Table 3.10, solving the relaxed MP in the DBDA consumes significantly less time compared to the other steps of the algorithm. Also, it requires a small fraction of total time.

3.6.4 Stochastic Restricted Location Problem Instances

A total of 360 stochastic restricted location problem instances were generated according to Table 3.3. Up to 6 facilities were considered. For these instances, the SBDA is not used to solve problems since it was established in the previous section that the performance of DBDA is superior. Computational results for the entire set of instances are presented in Tables A.1- A.12 in Appendix A.

As an example of a single instance from the full set of 360 instances, the solution of a two scenario instance with 36 demand points and 6 new facilities under probability set 1 is shown in Figures 3.8a and 3.8b. In Figure 3.8a, the locations of the new facilities and the paths from demand points to these facilities are presented when the barriers in scenario 1 occurs, whereas Figure 3.8b shows the locations of the new facilities and the paths between new facilities and the demand points under scenario 2.

A summary of the results for 18 and 36 demand point instances are presented in Tables 3.11 and 3.12, respectively, with results averaged across the scenarios for each probability set.

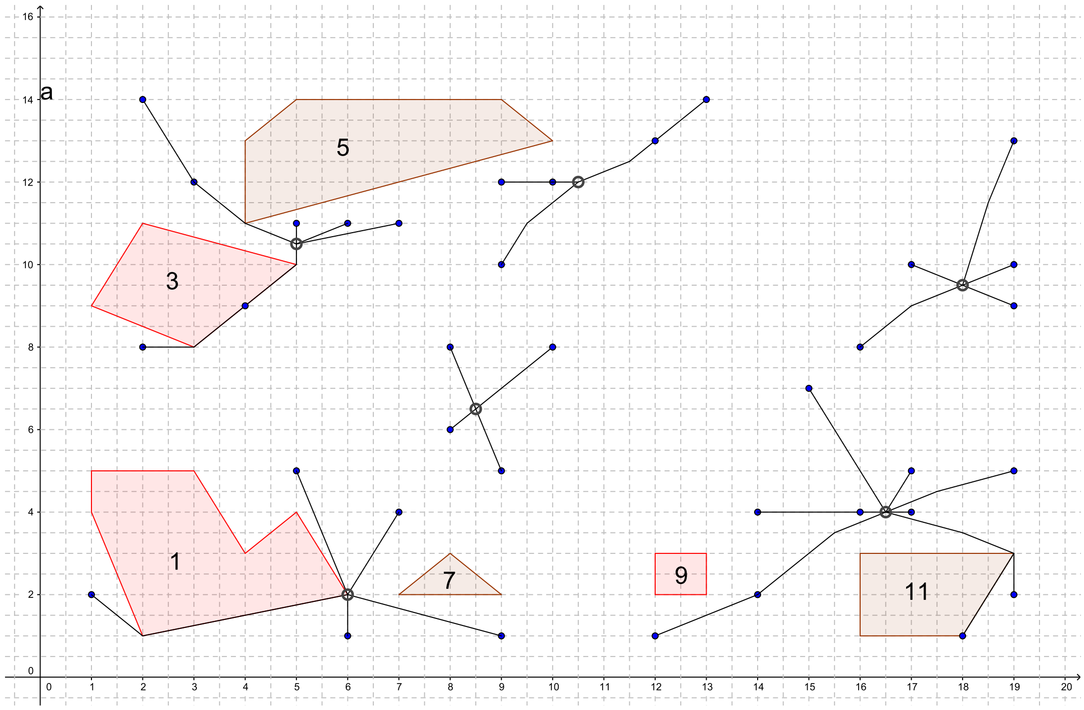
TABLE 3.11: Summary of results for 18 demand points instances

	θ	AVG. $v^\star$	AVG. t_C	AVG. t_D	SAVC	AVG. t_M	AVG. η_i	AVG. Total t_M	Ratio
Set 1	1	113.21	181.1	5.27	97.09%	0.026	2.8	0.073	1.38%
	2	83.18	212.06	11.41	94.62%	0.024	6.4	0.15	1.32%
	3	62.76	194.11	14.53	92.52%	0.021	7.8	0.166	1.14%
	4	48.68	169.69	11.40	93.28%	0.021	6	0.127	1.12%
	5	41.36	164	16.24	90.1%	0.047	8.6	0.401	2.47%
	6	35.56	153.01	19.78	87.07%	0.023	9.4	0.217	1.09%
Set 2	1	113.5	184.51	5.55	96.99%	0.022	3	0.066	1.19%
	2	83.58	219.89	11.25	94.88%	0.022	6.4	0.14	1.25%
	3	62.91	191.94	13.48	92.98%	0.023	7.4	0.168	1.25%
	4	48.93	166	16.16	90.26%	0.021	9	0.19	1.18%
	5	41.48	162.88	15.7	90.36%	0.02	8.6	0.17	1.08%
	6	35.62	154.6	23.63	84.71%	0.021	11.2	0.238	1.01%
Set 3	1	112.95	189.06	5.55	97.07%	0.023	3	0.069	1.24%
	2	83.48	222.52	10.83	95.13%	0.022	6.2	0.134	1.24%
	3	62.40	194.3	13.12	93.25%	0.028	7.4	0.208	1.59%
	4	48.92	165.83	13.93	91.60%	0.021	7.4	0.155	1.11%
	5	41.57	162.95	16.04	90.16%	0.021	8.8	0.186	1.16%
	6	35.71	154.63	27.18	82.42%	0.02	13.6	0.276	1.02%

TABLE 3.12: Summary of results for 36 demand points instances

	θ	AVG. $v^\star$	AVG. t_C	AVG. t_D	SAVC	AVG. t_M	AVG. η_i	AVG. Total t_M	Ratio
Set 1	1	249.14	471.17	10.74	97.72%	0.038	3	0.114	1.06%
	2	161.93	534.27	22.38	95.81%	0.043	6.4	0.275	1.23%
	3	130.57	478.45	36.66	92.34%	0.041	10.2	0.416	1.13%
	4	107.12	430.67	31.67	92.65%	0.039	8.6	0.335	1.06%
	5	91.42	364.85	28.2	92.27%	0.038	7.6	0.289	1.03%
	6	83.136	362	50.69	86%	0.04	14.4	0.569	1.12%
Set 2	1	249.53	475.25	11.04	97.68%	0.042	3	0.126	1.14%
	2	162.56	502.19	19.38	96.14%	0.04	5.8	0.233	1.2%
	3	130.97	478.98	26.16	94.54%	0.04	7.2	0.29	1.11%
	4	107.5	423.11	31.93	92.45%	0.039	8.6	0.334	1.05%
	5	91.66	356.95	26.53	92.57%	0.039	7.6	0.293	1.11%
	6	83.38	364.03	47.55	86.94%	0.04	10.8	0.427	0.9%
Set 3	1	246.62	451.97	10.54	97.67%	0.04	3	0.12	1.14%
	2	160.54	495.16	18.46	96.27%	0.04	5.4	0.217	1.18%
	3	130.03	473.4	28.94	93.89%	0.041	7.8	0.319	1.1%
	4	107.71	381.33	31.72	91.68%	0.041	8.8	0.359	1.13%
	5	91.9	362.38	53.45	85.25%	0.043	11.8	0.511	0.96%
	6	83.58	370.38	100.14	72.96%	0.046	23	1.066	1.06%

Similar to the results obtained in the previous section, Tables 3.11 and 3.12 confirm that the savings gained from the DBDA over CPLEX 12.5 are significant in the multiple scenario case. The savings gained by the DBDA for 18 and 36 demand points instances



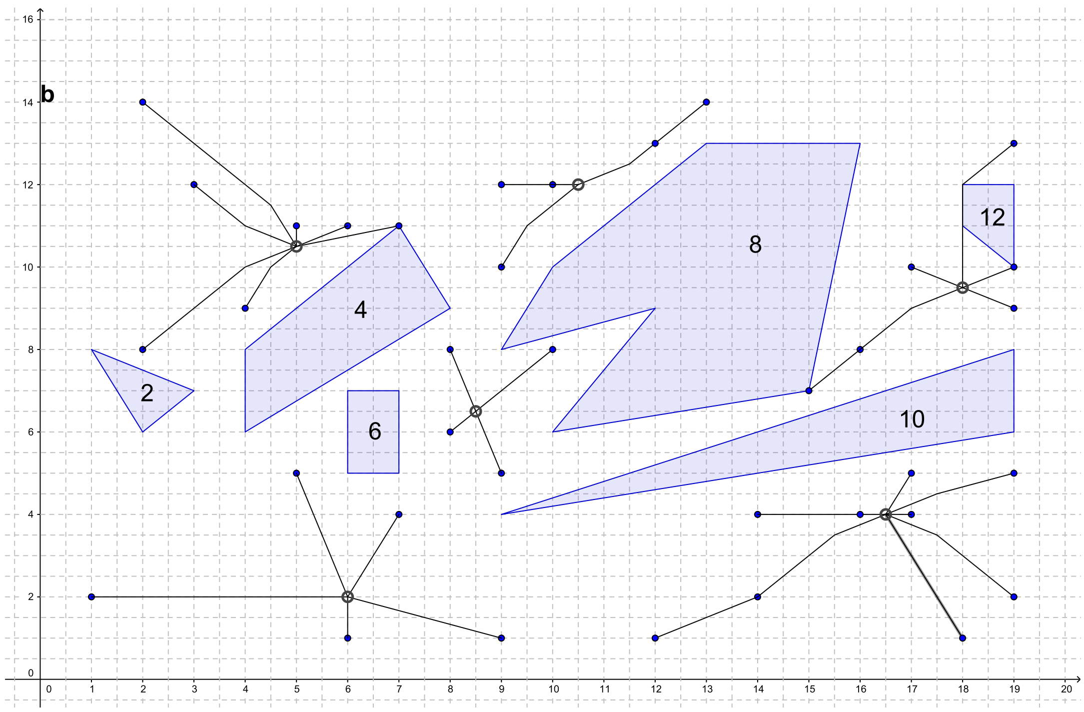


FIGURE 3.8: Solutions for the instance with 36 demand points and 6 new facilities using probability set 1

on probability sets 1, 2 and 3 considering 1-6 new facilities are presented in a bar chart in Figure 3.9a and 3.9b, respectively. Another interesting point is that the highest levels of savings are in single facility problem instances, varying between 96.99% and 97.72%. When the number new facilities increases, the performance gap between the DBDA and CPLEX 12.5 decreases, but still saving for DBDA are at least 72.96%, which is obtained for the instances with 36 demand points, 6 new facilities, and probability set 3.

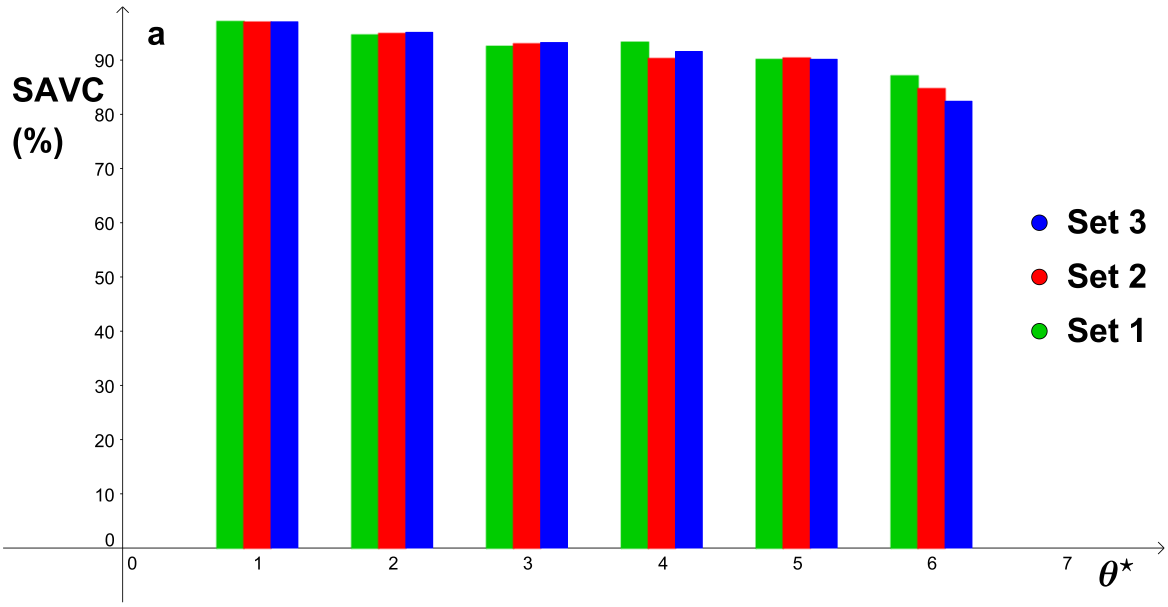
To solve the instances with 54 and 72 demand points using CPLEX 12.5, RAM was increased to 256 GB. Due to the size of the models, a large amount of memory is needed to run CPLEX 12.5. Table 3.13 shows the total number of variables (NV) and the total number constraints (NC) of DESM for 54 and 72 demand points instances for each scenario.

TABLE 3.13: Total number of variables and constraints for 54 and 72 demand points instances

$\lvert \mathsf{M} \rvert$	NS	NV	NC
54	2	276,654,400	347,201
	3	414,980,800	520,001
	4	553,307,200	692,801
	5	691,633,600	865,601
	6	829,960,000	1,038,401
72	2	368,872,000	462,401
	3	553,307,200	692,801
	4	737,742,400	923,201
	5	922,177,600	1,153,601
	6	1,106,612,800	1,384,001

Only 39 of the 90 instances could be solved for the instance with 54 demand points and none of the instances with 72 demand points could be solved by CPLEX 12.5; these results are reported in Tables A.7 - A.12 in Appendix. On the other hand, all of the instance could be solved to optimality by DBDA even with no increase in RAM.

Tables 3.14 and 3.15 show the average objective function value, average solution CPU time in seconds by DBDA, average CPU time for solving MP in each iteration, average number of iterations, average total CPU time for solving MPs, and the ratio between average total CPU time for solving MPs and average solution CPU time for DBDA across the scenarios for each combination of probability sets and problems with 54 and 72 demand points, respectively. Since 56.67% of the instances with 54 demand points



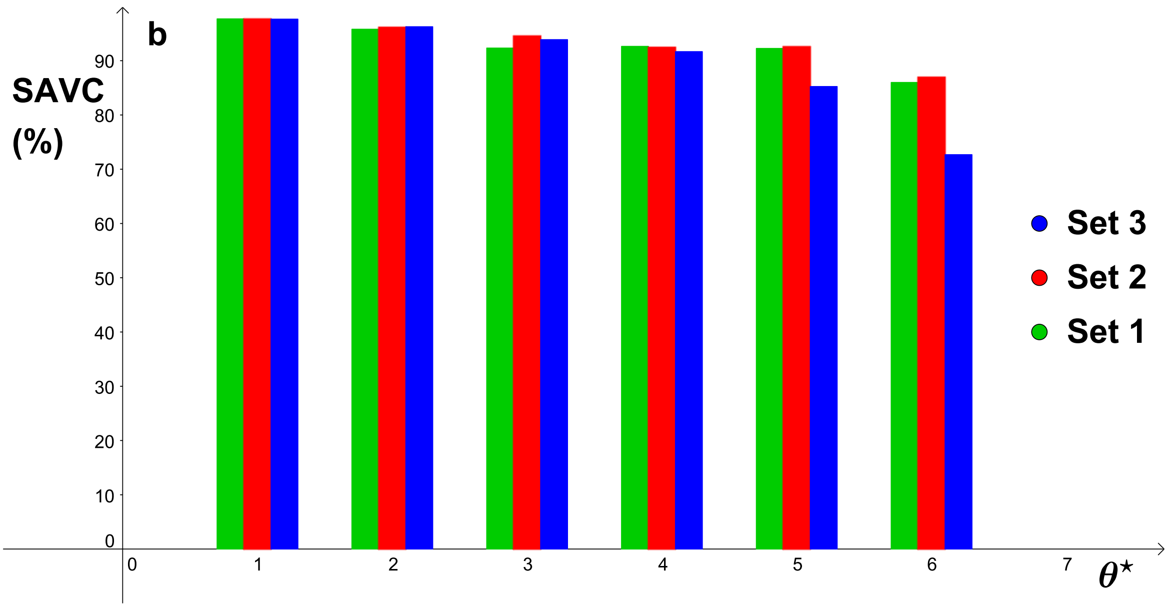


FIGURE 3.9: Savings (%) vs. number of demand points for 18 and 36 demand points instances

and none of the instance with 72 demand points could be solved by CPLEX 12.5, average solution CPU time for CPLEX 12.5 are not included in Tables 3.14 and 3.15.

Table 3.14: Summary of results for 54 demand points instances

	θ	AVG. $v^\star$	AVG. t_D	AVG. t_M	AVG. η_i	AVG. Total t_M	Ratio
Set 1	1	359.95	23.28	0.063	3.8	0.238	1.02%
	2	253.15	31.02	0.059	6.2	0.366	1.18%
	3	204	46.32	0.059	8.4	0.497	1.07%
	4	171.6	57.34	0.061	10.2	0.627	1.09%
	5	150.04	159.23	0.079	19.8	1.555	0.98%
	6	133.61	67.30	0.056	11	0.614	0.91%
Set 2	1	360.25	17.7	0.057	3.2	0.183	1.04%
	2	254.34	34.15	0.06	6.6	0.397	1.16%
	3	204.77	49.04	0.06	8.8	0.526	1.07%
	4	172.24	49.66	0.057	9	0.514	1.04%
	5	150.74	99.07	0.061	15.6	0.952	0.96%
	6	134.42	53.17	0.059	9.6	0.565	1.06%
Set 3	1	357.52	21.1	0.059	3.6	0.213	1.01%
	2	250.78	34.74	0.064	6.6	0.419	1.21%
	3	203.47	82.04	0.067	13.6	0.917	1.12%
	4	172.28	66.75	0.062	12.2	0.752	1.13%
	5	151.15	136.84	0.069	18.4	1.278	0.93%
	6	134.96	60.55	0.059	10.2	0.597	0.99%

Table 3.15: Summary of results for 72 demand points instances

	θ	AVG. $v^\star$	AVG. t_D	AVG. t_M	AVG. η_i	AVG. Total t_M	Ratio
Set 1	1	493.31	21.60	0.076	3	0.228	1.06%
	2	363.17	59.97	0.087	8.2	0.716	1.19%
	3	290.89	90.96	0.085	11.4	0.97	1.07%
	4	239.54	71.55	0.076	10.2	0.771	1.08%
	5	208.37	176.24	0.093	17.2	1.592	0.9%
	6	185.45	104.43	0.076	12	0.908	0.87%
Set 2	1	493.63	36.39	0.083	4.6	0.382	1.05%
	2	364.53	62.17	0.088	8.8	0.778	1.25%
	3	291.99	87.89	0.083	11.2	0.926	1.05%
	4	240.21	133.76	0.081	10.8	0.875	0.65%
	5	209.26	146.14	0.084	15.6	1.308	0.89%
	6	186.57	102.81	0.078	11.4	0.889	0.87%
Set 3	1	489.48	21.53	0.076	3	0.228	1.06%
	2	360.60	70.81	0.086	9.4	0.804	1.14%
	3	290.47	394.71	0.145	37.2	5.395	1.37%
	4	240.01	79.76	0.075	10	0.753	0.94%
	5	209.20	132.18	0.08	14.8	1.187	0.9%
	6	187.17	148.40	0.081	15.2	1.232	0.83%

Similar to the the deterministic case, Tables 3.11-3.12 and 3.14-3.15 show that the solution times of each master problem in each iteration is significantly lower. The ratios between average total time for solving MPs and average solution time for DBDA varies from 0.65% to 2.47% with an overall average of 1.11%. This shows that decomposing the model and solving DFDSP with Dijkstra's Algorithm leads to a significant gain in performance. Also, the large sized models cannot be solved by CPLEX 12.5 to optimality even with 256 GB of memory. These results along with those in the previous section, show that DBDA is a very efficient algorithm to solve both deterministic and stochastic restricted location problems that have been discretised using the proposed modelling framework.

3.7 Conclusions

This paper describes modelling and solution approaches for restricted facility location problems based on discretising the continuous plane and using a multi-commodity flow network with unknown destinations. The discretisation approach was tested on instances from the literature to evaluate the approximation quality. It was found that the percentage gaps between the global optimal solutions and the approximated results were less than 1% for all instances. An exact solution algorithm based on Benders decomposition that takes advantage of using a discrete network structure is also described. Both stochastic and deterministic restricted facility location problems with forbidden regions and barriers, with single and multiple facilities, and various with distance metrics can be solved. In total, 384 instances were generated and solved to evaluate the efficiency of the algorithm. In each instance, the performance of the proposed algorithm was superior, in terms of computational time to the off-the-shelf solver CPLEX 12.5. The performance of CPLEX 12.5 drops significantly as the number of demand points increases. In all, 56.67% of the instances with 54 demand points and none of the instances with 72 demand points could be solved by the off-the-shelf solver, whereas all of these instances could be solved by the proposed solution technique in a matter of 10s to 100s of seconds.

Chapter 4

Accelerated Benders Decomposition for the Capacitated Restricted Facility Location Problem

Abstract

This paper presents an exact solution algorithm and its variations for restricted capacitated facility location problems based on discretising the continuous plane and using a model for multi-commodity flows with unknown destinations. The solution algorithm is based on Benders decomposition, which takes advantage of the discrete network structure. Deterministic instances for restricted capacitated continuous facility location problems with barriers, are generated from well-known problems in literature. The performance of the proposed algorithm is tested on these benchmark instances. A comparison between an off-the-shelf solver and the proposed algorithm is also presented.

Keywords. mixed integer linear programming, capacitated facility location, multi-commodity network flow, Benders decomposition, restricted location

4.1 Introduction

Facility location problems are well-known in the field of operational research. The overarching goal is to find the optimal placement of new facilities considering various factors such as the customers that will be served, costs related to shipment of goods, and the feasible locations for facility sites.

Within the literature on facility location problems, it is possible to distinguish two categories. These are the uncapacitated facility location problems and the capacitated facility location problems. In uncapacitated facility location problems, it is assumed that a new facility can meet demand of customers without any shortage of supplies, whereas in capacitated facility location problems, a new facility has a limited supply. In practice, a shortage of supplies can be faced in many occasions, due to limited capacity of facilities. Thus, sensible precautions have to be considered to prevent possible interruptions on satisfying the demands of customers. Due to these reasons, capacitated facility location problems consider the optimal placements of the new facilities, while taking into account the capacity that each new facility has.

Another branch of facility location problems is termed restricted facility location problems. In these problems, there are restricted regions on continuous plane that a facility cannot be located inside. Three types of restricted regions are defined in the literature, namely forbidden regions, barriers and congested regions. In forbidden regions, travelling through them is permitted, whereas for barriers, passing through is not allowed. As a third restriction type, congested regions, travelling through them is only allowed by paying a penalty. These problems have been studied in the literature for nearly 30 years, but there are still many uninvestigated issues. Past studies of this problem class have focused on uncapacitated location problems. Readers are referred to Oğuz et al. (2016) for an extensive review of deterministic restricted location problems and to Chapter 3 for a review of stochastic problems.

To the best of our knowledge, capacitated versions of restricted facility location problems have not been studied before. In this paper, I used the discretisation technique proposed in Chapter 3 to transform the continuous plane into a network using a unit rectangle network. Based on the unit rectangle network, a general modelling framework for capacitated facility location problem is presented involving the use of mixed integer linear programming (MILP). I also devise various exact algorithms based on Benders decomposition for the solution of the model at hand. The performance of these algorithms is evaluated to find the best algorithm for capacitated restricted facility location problem.

The contributions of this paper, which are mainly computational, are listed below:

- An adaptation of the modelling approach in Research Paper 2 for capacitated restricted facility location problem is presented.
- Various algorithmic enhancements are applied to standard Benders decomposition to accelerate the performance in solving capacitated restricted facility location problems.
- I present extensive computational results of instances generated from the two main instances in the literature and one new instance to test the performance of the proposed exact algorithms.

s The remainder of this paper is organised as follows. In Section 4.2, I introduce a definition of the problem at hand with notation and the modelling framework for the capacitated facility location problems with restrictions. In Section 4.3, exact algorithms based on Benders decomposition are introduced for solving the capacitated restricted facility location problem. I present computational results of instances generated from the literature in Section 4.4. In Section 4.5, conclusions are given.

4.2 The Capacitated Restricted Facility Location Problem

In this section, the problem at hand is defined below. A discretisation of the continuous space is provided in Section 4.2.1 and a way of modelling the problem is presented in Section 4.2.2.

Consider a set M of demand points that will be served by a number of new facilities with each point $m \in \mathsf{M}$ having a positive weight, v_m, reflecting the amount of demand at point m. The considered restriction type is barriers, where new facilities cannot be located inside the barriers and travelling through them is not allowed. The new facilities are capacitated and each new facility can only serve a limited number μ of demand points due to its storage capacity. There is also a fixed cost to open any new facility with the feasible region, i.e. outside the barriers. In this thesis, it is assumed that the facilities are identical with respect to capacity but the fixed cost is location-dependent, to account for different installation costs at different geographical locations.

Considering the restrictions on the locations of the new facilities and the transportation between the demand points and the new facilities, the cost of opening new facilities and the available serving capacities of the new facilities, the problem aims to find the optimal number and the locations of new facilities to be opened to serve the demand points by minimising a function of the distance between the demand points and the new facilities plus the opening cost of facilities.

4.2.1 Discretisation of the Continuous Space

To model and solve the capacitated restricted facility location problem, the continuous space is discretised by forming a unit rectangle network, which was first presented in Chapter 3. Each unit rectangle constructs the rows and the columns of the network, where each unit rectangle is formed by unit squares. Each vertex of the unit square is a node, $i \in \mathbf{N}$, where $\mathbf{N}$ is the set of nodes. Figure 4.1 shows two unit rectangles, one forms a single row of the network, and the other one forms a single column of the network.

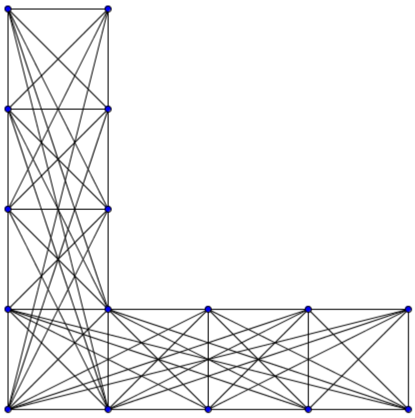


FIGURE 4.1: Unit rectangle network

As it can be seen in the Figure 4.1, the nodes of the each unit rectangle are connected by arcs in a cross-cutting manner. In this formation, when the number of nodes is increased, the number of arcs will automatically increase, leading to a better approximation of the continuous space. The quality of this discretisation technique is tested in Chapter 3 by solving various uncapacitated continuous restricted single facility location problems from the literature. It is concluded that the approximation quality is satisfactory in Chapter 3.

4.2.2 A Model for the Discretised Capacitated Restricted Facility Location Problem

The proposed modelling framework for the discretised capacitated restricted facility location problem is based on a multi-commodity network formulation. In multi-commodity flow problems, the locations of the origin and destination of the commodity flow are known a priori and the aim is to find the optimal routing of each commodity with a minimum cost between origin and destination pairs. In facility location problems, however, the locations of the demand points are known and the aim is to find the optimal placement of new facilities to serve demand points efficiently. In our proposed modelling framework, the locations of the destination nodes are determined by the locations of the new facilities. The origin nodes for each commodity are given by the locations of the demand points.

Let $\mathbf{O}$ be the set of nodes, which is also a subset of $\mathbf{N}$, where the demand points are located. $O(m)$ gives the node of demand point $m \in \mathbf{M}$. The parameter c_{mij} is the unit distance of arc $(i,j) \in \mathbf{A}$, where flow from demand point $m \in \mathbf{M}$ to a new facility may pass through. Parameter μ is the maximum number of demand points which a single new facility can serve. The parameter f_i is the fixed cost to open a new facility at node $i \in \mathbf{N}$. There are three variables in the proposed modelling framework, including two continuous and one binary. The first continuous variable x_{mij} indicates the unit flow from the demand point $m \in \mathbf{M}$ through the arc $(i,j) \in \mathbf{A}$, and $X = \{x_{mij}, m \in \mathbf{M}, \{i,j\} \in \mathbf{A}_s\}$, whereas the second continuous variable z_{mi} denotes which new facility serves demand point $m \in \mathbf{M}$, and $Z = \{z_{mi}, m \in \mathbf{M}, i \in \mathbf{N}\}$. Binary variable y_i takes the value 1 if a new facility is opened at node $i \in \mathbf{N}$, 0 otherwise, and $Y = \{y_i \in \{0,1\}, i \in \mathbf{N}\}$. A formulation for the capacitated restricted facility location problem is shown below.

$$\underset{X,Y,Z}{\text{minimise}}\; F\left(v_m, c_{mij}, x_{mij}\right)_{m\in\mathbf{M},\{i,j\}\in\mathbf{A}} + \sum_{i\in\mathbf{N}} f_i y_i \tag{4.1}$$

subject to

$$\sum_{j:(i,j)\in\mathbf{A}} x_{mji} - \sum_{j:(i,j)\in\mathbf{A}} x_{mij} = \begin{cases} -1 & \text{if } i = O(m), \\ z_{mi} & \text{otherwise}; \end{cases} \qquad \forall i \in \mathbf{N}, m \in \mathbf{M} \tag{4.2}$$

$$\sum_{m\in\mathbf{M}} z_{mi} \le \mu y_i \qquad \forall i \in \mathbf{N} \tag{4.3}$$

$$z_{mi} \le 1 \qquad \forall i \in \mathbf{N}, m \in \mathbf{M} \tag{4.4}$$

$$y_i \in \{0,1\} \qquad \forall i \in \mathbf{N}, \tag{4.5}$$

$$z_{mi}, x_{mij} \geq 0 \qquad \forall\forall i \in \mathbf{N}, \forall m \in \mathbf{M}, \{i,j\} \in \mathbf{A}. \tag{4.6}$$

The objective function (4.1) minimises a linear function $F(v_m, c_{mij}, x_{mij})$, which defines a weighted cost of transportation between the demand points and new facilities, plus the total fixed cost of opening new facilities. It is expected that a high fixed cost will lead to a fewer number of new facilities opened, whereas a low fixed cost will do the opposite. In Section 4.4, various fixed cost values, from low to high, will be tested. In a case of a median problem, where the objective is to minimise the total distance between the demand points and new facilities, $F(v_m, c_{mij}, x_{mij})_{m \in \mathbf{M}, \{i,j\} \in \mathbf{A}} = \sum_{m \in \mathsf{M}} \sum_{\{i,j\} \in \mathsf{A}} v_m c_{mij} x_{mij}$. In this paper, the objective of the median problem is considered, but others such as that of the center problem can be considered using this framework. Constraints (4.2) are the flow conservation equations that are imposed on the flow from each demand point $m \in \mathbf{M}$. Capacity constraints (4.3) gives the maximum number of demand points that can be served by a single new facility. Constraints (4.4) puts an upper bound of 1 on variable z_{mi} to ensure a unit flow from each demand point $m \in \mathbf{M}$ to its assigned facility. Constraints (4.5) and (4.6) define the binary and non-negativity restrictions on the decision variables, respectively.

The mathematical model for the Capacitated Restricted Facility Location Problem is a mixed integer linear program (MILP). In the next section, I presented several algorithms based on Benders decomposition to solve this type of problem to optimality.

4.3 Accelerated Benders Decomposition Algorithms for Capacitated Restricted Facility Location Problems

The Benders decomposition algorithm is a well-known exact algorithm to efficiently solve linear models to optimality (Benders 1962). In order to apply Benders decomposition algorithm to the capacitated restricted location problem, the variables of the model can be classified into two groups, namely linking variables and complicating variables. The main idea of the algorithm is to decompose the MILP into two smaller problems, one as a linear program (LP), named the sub-problem (SP), with linking variables and fixed complicating variables, and the other an MILP, named the master problem (MP), with complicating variables. By iteratively solving the dual of the SP and adding constraints to the MP, an optimal solution is eventually found. In each iteration of the algorithm, the relaxed master problem yields a lower bound (LB) for the global optimum of the original MILP, whereas the dual of SP yields an upper bound (UB) to the global optimum of the original model.

Even if there is no stochasticity in the capacitated problem, Benders decomposition is still a suitable method to solve the associated formulation given the special structure. In particular, given a fixing of the location decisions, the SP can be further decomposed for each demand point. This decomposition will form basic shortest path problems for each demand point that each can be solved by Dijkstra's algorithm efficiently as shown in Chapter 3.

In this paper, three acceleration techniques are tested to improve the performance of the algorithm, namely valid inequalities, disaggregated cuts and the two-phase method. In the next subsection, I provide more details on Benders decomposition, its variants, and how I implement them for the problem at hand.

4.3.1 Standard Benders Decomposition

For illustration purposes, the objective of a median problem will be considered in this paper for the formulation (4.1)-(4.6). The flow variables x_{mij} are categorised as linking variables, whereas the decision variables for the new facilities z_{mi} and y_i are classified as complicating variables. The Benders sub-problem is obtained by fixing the complicating variables z_{mi} and y_i to values $\bar{z}_{mi}$ and $\bar{y}_i$, respectively, such that constraints (4.3) - (4.6) are satisfied.

$$\text{minimise} \sum_{m \in \mathsf{M}} \sum_{\{i,j\} \in \mathsf{A}} v_m c_{mij} x_{mij} \tag{4.7}$$

subject to

$$\sum_{j:(i,j)\in \mathbf{A}} x_{mji} - \sum_{j:(i,j)\in \mathbf{A}} x_{mij} = \begin{cases} -1 & \text{if } i = O(m), \\ \bar{z}_{mi} & \text{otherwise;} \end{cases} \qquad \forall i \in \mathbf{N}, m \in \mathbf{M} \tag{4.8}$$

$$x_{mij} \geq 0 \qquad \forall m \in \mathbf{M}, \forall \{i,j\} \in \mathbf{A}. \tag{4.9}$$

The dual of SP (DSP) is:

$$\text{maximise} \sum_{m\in \mathsf{M}} \sum_{\substack{i\in \mathsf{N} \\ i\neq O(m)}} \bar{z}_{mi} u_{mi} - \sum_{m\in \mathsf{M}} \sum_{\substack{i\in \mathsf{N} \\ i=O(m)}} u_{mi} \tag{4.10}$$

subject to

$$u_{mj} - u_{mi} \leq v_m c_{mij} \qquad \forall m \in \mathbf{M}, \forall \{i,j\} \in \mathbf{A}. \tag{4.11}$$

Based on the dual of SP, the relaxed Benders Master Problem (MP) for the capacitated restricted multi-facility location problem can be stated as follows in terms of the y and v variables:

$$\text{minimise } \alpha + \sum_{i\in \mathsf{N}} f_i y_i \tag{4.12}$$

subject to

$$\alpha \geq \sum_{m\in \mathsf{M}} \sum_{\substack{i\in \mathsf{N} \\ i\neq O(m)}} z_{mi} \bar{u}^{(k)}_{mi} - \sum_{m\in \mathsf{M}} \sum_{\substack{i\in \mathsf{N} \\ i=O(m)}} \bar{u}^{(k)}_{mi} \qquad \forall k \in \mathbf{K} \tag{4.13}$$

$$\sum_{m\in \mathsf{M}} \sum_{\substack{i\in \mathsf{N} \\ i\neq O(m)}} z_{mi} \bar{u}^{(l)}_{mi} - \sum_{m\in \mathsf{M}} \sum_{\substack{i\in \mathsf{N} \\ i=O(m)}} \bar{u}^{(l)}_{mi} \leq 0 \qquad \forall l \in \mathbf{L} \tag{4.14}$$

$$\sum_{m\in \mathbf{M}} z_{mi} \leq \mu y_i \qquad \forall i \in \mathbf{N} \tag{4.15}$$

$$z_{mi} \leq 1 \qquad \forall i \in \mathbf{N}, m \in \mathbf{M} \tag{4.16}$$

$$y_i \in \{0,1\} \qquad \forall i \in \mathbf{N}, \tag{4.17}$$

$$z_{mi} \geq 0 \qquad \forall i \in \mathbf{N}, m \in \mathbf{M}. \tag{4.18}$$

Sets K and L are the extreme points and extreme rays of the feasible space of DSP. Constraints (4.13) are defined for each extreme point in the feasible region of the dual of the SP and termed Benders optimality cuts. Constraints (4.14) are defined for each extreme ray of DSP, and, called Benders feasibility cuts.

Since the decision variables for the location of the new facilities are fixed in SP, the objective of SP is to find the shortest path distances between the new facilities and demand points. This means that the SP can be separated into independent shortest path problems SP_m for each demand point $m \in \mathsf{M}$, which can be formulated as follows;

$$\text{minimise} \sum_{\{i,j\} \in \mathsf{A}} v_m c_{mij} x_{mij} \tag{4.19}$$

subject to

$$\sum_{j \in \mathsf{N}} x_{mji} - \sum_{j \in \mathsf{N}} x_{mij} = \begin{cases} -1 & \text{if } i = O(m), \\ \bar{z}_{mi} & \text{otherwise;} \end{cases} \qquad \forall i \in \mathsf{N} \tag{4.20}$$

$$x_{mij}, x_{mji} \geq 0 \qquad \forall \{i,j\} \in \mathsf{A}. \tag{4.21}$$

This problem can be efficiently solved by Dijkstra's algorithm and the dual variables u_{mi} needed for cut generation in the MP directly obtained from Dijkstra's algorithm, as shown in Chapter 3.

4.3.2 Valid Inequalities

Introducing valid inequalities to the MP may decrease the number of iterations needed to reach optimality. One has to be careful about the series of valid inequalities, since they must not change the structure of the problem (Saharidis et al. 2010). In this paper, I introduce a set of valid inequalities considering the number of demand points and the capacity of each new facility. In this problem, it is assumed that each new facility can serve at most μ demand points. Since the total number of demand points is known, $|\mathsf{M}|$, it is possible to add the following inequality to the MP.

$$\sum_{i \in \mathbf{N}} y_i \geq \lceil \frac{|\mathbf{M}|}{\mu} \rceil. \tag{4.22}$$

By adding inequality (4.22), the structure of the problem will not be changed, since the terms in the right hand side of the inequality are already known. This valid inequality will be added to all algorithms that are presented above to minimise the number of iterations and prevent the infeasibility that may come out during solution process.

Besides adding the valid inequality (4.22) to the MP, another infeasibility that can be eliminated. Since a new facility cannot be located inside the restricted region, addition of the following equality will automatically prevent this. This equality was first presented in Chapter 3.

$$\sum_{i \in \mathbf{N}_b} y_i = 0 \qquad \forall b \in \mathbf{B}, \tag{4.23}$$

where $\mathbf{N}_b$ is the subset of nodes that fall within the restricted region b and $\mathbf{B}$ is the set of barriers. This equality means that there will not be any placement of facilities inside the barriers.

Similar to the valid inequality (4.22), the equalities (4.23) must be also added to the MP.

Another reason for infeasibility during the solution procedure is if a demand point is not served due to the limited number of facilities available. This infeasibility can be prevented by the following set of valid inequalities.

$$\sum_{i \in \mathbf{N}} z_{mi} \geq 1 \qquad \forall m \in \mathbf{M}. \tag{4.24}$$

The inequality (4.24) ensures that each demand point $m \in \mathbf{M}$ is served by one new facility and prevents any infeasibility caused by an unserved demand point.

By adding (4.23) and (4.24) to the MP, all possible conditions that can cause infeasibility for the problem are eliminated, so there is no need to generate infeasibility cuts, which will increase the performance of the algorithm.

4.3.3 Disaggregated Cuts

Disaggregated cuts cannot be applied to every MILP. To apply disaggregated cuts, the SP should be decomposable into a set of independent subproblems. Minoux (2001) and Gabrel et al. (1999) use this multi-generations of cuts method in variants of multi-commodity flow problems. As shown in Section 4.3.1, it is possible to decompose the SP into independent shortest path problems, one for each demand point. Each of these shortest path problems will form a disaggregated cut for the MP at any iteration. Instead of having an aggregated primal Benders cut (4.13), in the MP, I can add the following disaggregated cuts, which have the same information as their counterpart;

$$\alpha_m \geq \sum_{\substack{i \in \mathsf{N} \\ i \neq O(m)}} z_{mi}\bar{u}_{mi}^{(k)} - \sum_{\substack{i \in \mathsf{N} \\ i = O(m)}} \bar{u}_{mi}^{(k)} \qquad \forall k \in \mathbf{K}, m \in \mathbf{M}. \tag{4.25}$$

Thus, the objective function of the MP (4.12) can be transformed into:

$$\text{minimise} \sum_{m \in \mathsf{M}} \alpha_m + \sum_{i \in \mathsf{N}} f_i y_i \tag{4.26}$$

This method generates $|\mathbf{M}|$ cuts in each iteration of the algorithm, such that the solution space of the MP is restricted more efficiently. The pseudo-code of standard Benders Decomposition with disaggregated cuts (DIS) is shown as Algorithm 1.

The Benders decomposition algorithm with aggregated cuts (AGG) is same as Algorithm 1, except for the generation of cuts and the objective of the MP. Tang et al. (2013) stated that disaggregated cuts improves the convergence rate of the algorithm, but at the same time, the generation of multiple cuts in a single iteration may cause a high amount of time spent in a single iteration. To investigate this trade-off, both algorithms will be tested in Section 4.4.

Algorithm 1

$NCut$: Number of Iterations; $MaxCut$: Maximum Number of Cuts; $z[NCut]$: Optimal value of MP in iteration $NCut$.

Initialization: $NCut \longleftarrow 0$.

START

  Solve $MP(Initial)$. Let the solutions be $(\bar{Y}, \bar{Z})$ and the optimal value be

  $v_{MP}(\bar{U})$.

    $z[NCut] \longleftarrow v_{MP}(\bar{U})$

  While $(NCut < MaxCut)$

    Solve each $SP_m\left(\bar{Y}, \bar{Z}\right)$ $\forall m \in \mathsf{M}$.

    Let the solution be $\bar{U}$ and the optimal values be $v_{SP_m}(\bar{Y}, \bar{Z})$ $\forall m \in \mathsf{M}$.

      $v_{SP}(\bar{Y}, \bar{Z}) = \sum_{m \in \mathsf{M}} v_{SP_m}(\bar{Y}, \bar{Z})$

    If $(v_{SP}(\bar{Y}, \bar{Z}) \leq z[NCut] + \epsilon)$

      $STOP$

    $NCut \longleftarrow NCut + 1$

    Solve $MP\left(\bar{U}\right)$. Let the solutions be $(\bar{Y}, \bar{Z})$ and the optimal value be

    $v_{MP}(\bar{U})$.

      $z[NCut] \longleftarrow v_{MP}(\bar{U})$

  End While

END

4.3.4 Two-Phase Method

Two-phase Benders decomposition algorithm was first proposed by McDaniel and Devine (1977). The main idea is to relax the complicating variables in the MP to initially solve a sequence of LPs before reverting to the integer form of the MP. The formulation of the MP in an LP form (LPMP) is:

$$\text{minimise } \alpha + \sum_{i \in \mathsf{N}} f_i y_i \tag{4.27}$$

subject to

$$\alpha \geq \sum_{m \in \mathsf{M}} \sum_{\substack{i \in \mathsf{N} \\ i \neq O(m)}} z_{mi} \bar{u}_{mi}^{(k)} - \sum_{m \in \mathsf{M}} \sum_{\substack{i \in \mathsf{N} \\ i = O(m)}} \bar{u}_{mi}^{(k)} \qquad \forall k \in \mathbf{K} \tag{4.28}$$

$$\sum_{m \in \mathsf{M}} \sum_{\substack{i \in \mathsf{N} \\ i \neq O(m)}} z_{mi} \bar{u}_{mi}^{(l)} - \sum_{m \in \mathsf{M}} \sum_{\substack{i \in \mathsf{N} \\ i = O(m)}} \bar{u}_{mi}^{(l)} \leq 0 \qquad \forall l \in \mathbf{L} \tag{4.29}$$

$$\sum_{m \in \mathbf{M}} z_{mi} \leq \mu y_i \qquad \forall i \in \mathbf{N} \tag{4.30}$$

$$z_{mi} \leq 1 \qquad \forall i \in \mathbf{N}, m \in \mathbf{M} \tag{4.31}$$

$$y_i \leq 1 \qquad \forall i \in \mathbf{N}, \tag{4.32}$$

$$z_{mi} \geq 0 \qquad \forall i \in \mathbf{N}, m \in \mathbf{M}, \tag{4.33}$$

$$y_i \geq 0 \qquad \forall i \in \mathbf{N}. \tag{4.34}$$

The algorithm starts by solving the relaxed LPMP until the SP and LPMP converge. It is important to note that at this point of the algorithm, the optimal value of the SP in each iteration does not have to be an upper bound for the optimal solution of the original problem. It is also obvious that the value where the SP and the LPMP converge is a lower bound for the optimal solution of the original problem. After the first convergence, algorithm uses the original, unrelaxed MP, but with the cuts generated from the LPMP. After this point, the optimal solution of the SP in each iteration is definitely an upper bound for the optimal solution of the original problem. Finally the algorithm stops, when the SP and the MP converge at the optimality. Both disaggregated and aggregated cuts can be used in the algorithm. 2PDIS is the acronym for the two-phase Benders decomposition algorithm with disaggregated cuts, whereas 2PAGG is an acronym for two-phase Benders decomposition algorithm with aggregated cuts. The pseudo code of the two-phase method is shown in Algorithm 2.

Algorithm 2 Two-Phase Method

$NCut$: Number of Iterations; $MaxCut$: Maximum Number of Cuts; $z[NCut]$: Optimal value of MP in iteration $NCut$. $Z[NCut]$: Optimal value of LPMP in iteration $NCut$.

Initialization: $NCut \longleftarrow 0$, $z[NCut] \longleftarrow -\infty$

START

  Solve $LPMP(Initial)$. Let the solution be $(\bar{Y}, \bar{Z})$ and the optimal value be

  $v_{LPMP}(\bar{U})$.

    $Z[NCut] \longleftarrow v_{MP}(\bar{U})$

  While $(NCut < MaxCut)$

    Solve each $SP_m\left(\bar{Y}, \bar{Z}\right)$ $\forall m \in \mathsf{M}$.

    Let the solution be $\bar{U}$ and the optimal values be $v_{SP_m}(\bar{Y}, \bar{Z})$ $\forall m \in \mathsf{M}$.

      $v_{SP}(\bar{Y}, \bar{Z}) = \sum\limits_{m \in \mathsf{M}} v_{SP_m}(\bar{Y}, \bar{Z})$

    If $(v_{SP}(\bar{Y}, \bar{Z}) \leq z[NCut] + \epsilon)$

      $STOP$

      If $(v_{SP}(\bar{Y}, \bar{Z}) <= Z[NCut] + \epsilon)$

        $Z[NCut] \longleftarrow \infty$

        $NCut \longleftarrow NCut + 1$

        Solve $MP\left(\bar{U}\right)$. Let the solution be $(\bar{Y}, \bar{Z})$ and the optimal value

        be $v_{MP}(\bar{U})$.

        $z[NCut] \longleftarrow v_{MP}(\bar{U})$

      Else

        $NCut \longleftarrow NCut + 1$

        Solve $LPMP\left(\bar{U}\right)$. Let the solution be $(\bar{Y}, \bar{Z})$ and the optimal

        value be $v_{MP}(\bar{U})$.

          $Z[NCut] \longleftarrow v_{MP}(\bar{U})$

  End While

END

4.4 Computational Results

This section presents the results of extensive computational experiments to test the algorithms presented in Section 4.3. In Section 4.4.1, the generation of the instances is described. The generated instances are grouped into three and the analysis for each group of presented in Sections 4.4.2, 4.4.3 and 4.4.4, respectively.

All experiments in Section 4.4 were solved on a single node of the IRIDIS 4, computer cluster, at the University of Southampton, with dual 2.6 GHz Intel Sandybridge processors and 256 GB of RAM, using CPLEX 12.5. All presented algorithms were implemented in C++.

4.4.1 Description of the Instances

I have generated capacitated restricted facility location instances on the basis of two complex barrier instances described by Katz and Cooper (1981) and Aneja and Parlar (1994), and another instance that is newly proposed in this paper. Table 4.1 shows the characteristics of these core instances, including the names used in the remainder of the paper, their source, type, number of demand points ($|\mathbf{M}|$), the type of restricted regions (RRT), and the distance metric.

Table 4.1: Core instances

Name	Source	Type	$\|\mathbf{M}\|$	RRT	Distance	Figure
C1	This paper	1-median	5	Polygonal	Euclidean	Figure 4.2a
C2	Aneja and Parlar (1994)	1-median	18	Polygonal	Euclidean	Figure 4.2b
C3	Katz and Cooper (1981)	1-median	10	Circular	Euclidean	Figure 4.2c

Figure 4.2 shows the shapes and locations of the barriers in each instance. The locations of the demand points in the instances are also depicted in Figure 4.2.

The generation of instances from these core instances is achieved by varying the parameters such as the number of demand points $|\mathbf{M}|$, the maximum number of demand points that a single new facility can serve μ, and the fixed cost f_i to open a new facility at node $i \in \mathbf{N}$.

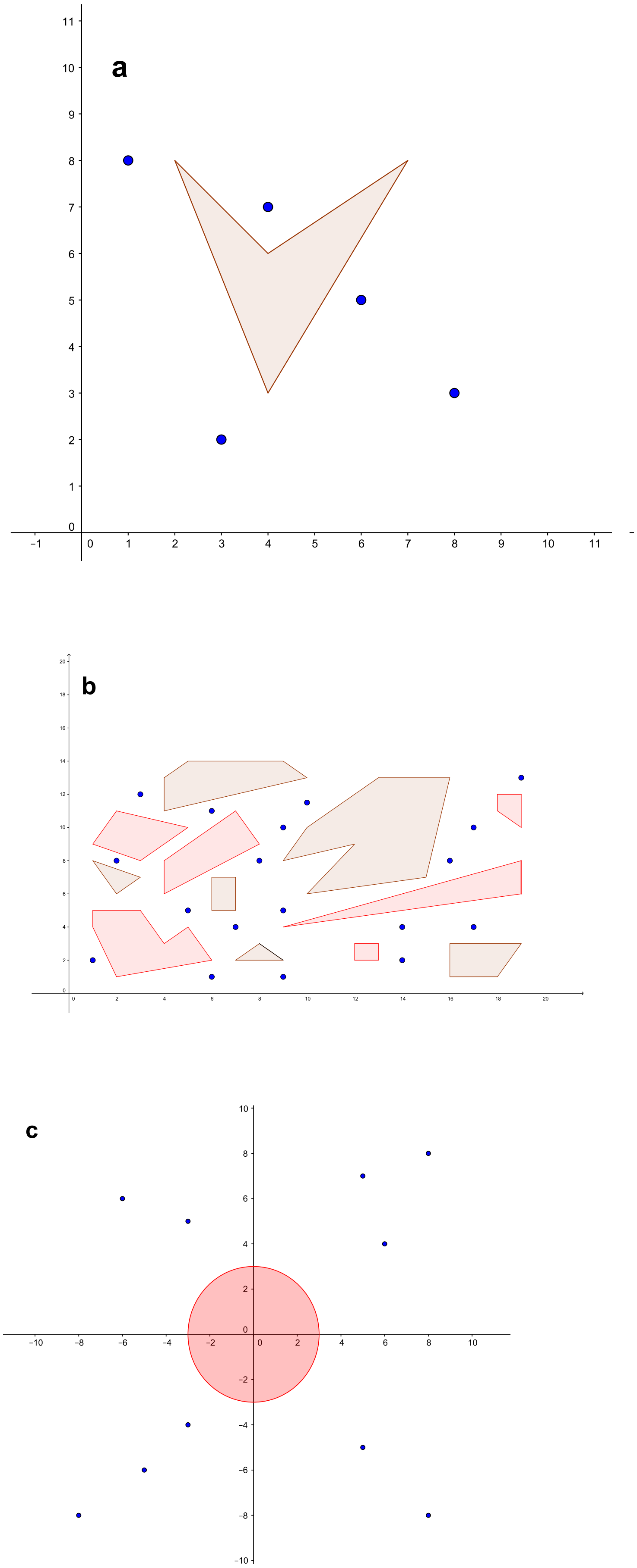


FIGURE 4.2: Core instances

For fixed facility costs, six different uniform values for f_i are considered. These values are 1, 5, 10, 15, 20 and 25 unit costs. The reason to choose these values is to observe the effect of the increase in f_i on the performance of the proposed solution techniques. The values of μ are set equal to various percentages of the total number of demand points $|\mathbf{M}|$. For example, if $|\mathbf{M}| = 7$ and the percentage is 50%, then $\mu = 3$, which means a single new facility can serve 3 demand points at most. In this paper, I test the following percentages: 30%, 50%, 70% and 90%.

The last step of the instance generation is to decide on $|\mathbf{M}|$. Originally, C1, C2 and C3 have 5, 18 and 10 demand points, respectively. In addition to the original demand points, 10, 15 and 20 new demand points for C1, 36, 54 and 72 new demand points for C2, and 20, 30 and 40 new demand points for C3 are considered. The locations of the demand points in the C2 instances are the same as in Chapter 3. Table 4.2 shows the coordinates of the demand point $m \in \mathbf{M}$ for the C1 and C3 instances.

TABLE 4.2: Restricted location problems literature overview

C1 Instances				C3 Instances			
$\lvert\mathbf{M}\rvert = 5$	$\lvert\mathbf{M}\rvert = 10$	$\lvert\mathbf{M}\rvert = 15$	$\lvert\mathbf{M}\rvert = 20$	$\lvert\mathbf{M}\rvert = 10$	$\lvert\mathbf{M}\rvert = 20$	$\lvert\mathbf{M}\rvert = 30$	$\lvert\mathbf{M}\rvert = 40$
(1,8)	(4,9)	(1,6)	(0,0)	(8,8)	(-7,-2)	(-6,3)	(-6,-4)
(3,2)	(4,1)	(2,3)	(1,4)	(5,7)	(-4,-1)	(-5,0)	(-8.5,8)
(4,7)	(3,5)	(6,1)	(6,3)	(6,4)	(-3,2)	(-3,-2)	(-2,-3)
(6,5)	(7,6)	(8,9)	(7,0)	(-3,5)	(-1,-3)	(-1,-6)	(-1,8)
(8,3)	(0,2)	(9,5)	(9,8)	(-6,6)	(0,3)	(1,5)	(4,-3)
				(-3,-4)	(1,-4)	(2,-5)	(5,3)
				(-5,6)	(2.5,-7)	(6,-7)	(6,-2)
				(-8,-8)	(4,0)	(0,-8)	(7,-5.5)
				(5,-5)	(4,4)	(7,6)	(8,-1)
				(8,-8)	(7,1)	(8,3)	(9,5)

In Table 4.2, each column shows the locations of newly added demand points. For example, the demand points of the C1 instance with 15 demand points are located on coordinates (1,8), (4,9), (1,6), (3,2), (4,1), (2,3), (4,7), (3,5), (6,1), (6,5), (7,6), (8,9), (8,3), (0,2) and (9,5).

The instance generation process described above has 96 instances each for C1, C2 and C3, yielding a total of 288 instances for testing.

Another important point to consider is the unit rectangle networks and their properties that are used to discretise the continuous space of the instances. Table 4.2 shows the total number of arcs ($|\mathbf{A}|$), total number of nodes ($|\mathbf{N}|$) and the number of available nodes (AN) that a new facility can be opened on for the C1, C2 and C3 instances.

TABLE 4.3: Network properties for each core instance

Name	$\|\mathbf{A}\|$	$\|\mathbf{N}\|$	AN
C1	2146	100	89
C2	96796	1600	1290
C3	193126	1600	1481

As can be seen from Table 4.3, a smaller sized network is used for the instances generated from C1. It is obvious that networks with a fewer number of nodes and arcs can effect the approximation quality in a negative manner. On the other hand, it may be healthier to analyze and compare the performance of the algorithms on a smaller instance first. Another significant point in Table 4.3 that has to be taken into consideration is that the solution space for C3 instances is larger than the C2 instances. In other words, the more $|\mathbf{A}|$ and AN will lead to a bigger solution space for the instances in this problem type.

Finally, it should be kept in mind that the reason why C3 has more available arcs and nodes is that there is only one barrier in C3, which is in contrast to C2 which has 12 barriers.

4.4.2 C1 Instances

The 96 C1 instances are modelled and solved by five different methods, namely, CPLEX 12.5, DIS, AGG, 2PDIS and 2PAGG. The results obtained can be found in Tables B.1-B.4 in Appendix B. As an example, Figure 4.3 illustrates the solution of an instance with 15 demand points, a maximum serving capacity of 10 demand points for a single facility, and a fixed cost of 20 units. Under these conditions, the optimal number of new facilities that serve the demand points is two and they are located on the coordinates $(3, 3)$ and $(7, 7)$.

For the instances with 5, 10, 15 and 20 demand points, Table 4.4 shows the average solution CPU times in seconds (t_C) required by CPLEX 12.5, the average CPU time in seconds (t_{2PDIS}) spent by the 2PDIS, the average CPU time in seconds (t_{2PAGG})

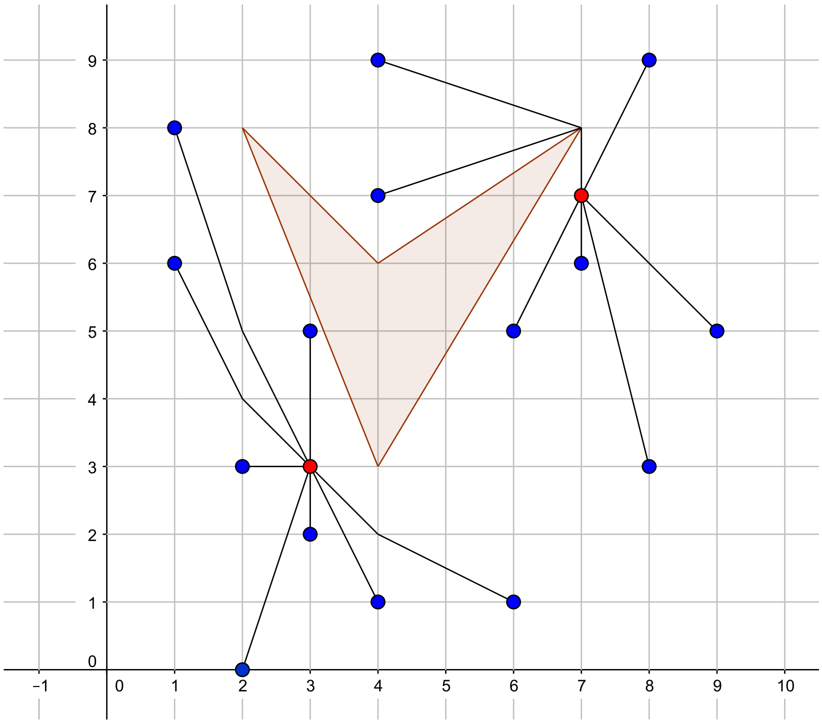


FIGURE 4.3: A C1 instance with $|\mathbf{M}| = 15$, $\mu = 10$, $f_i = 20$

required by the 2PAGG, the average CPU time in seconds (t_{DIS}) spent by the DIS and the average CPU time in seconds (t_{AGG}) required by the AGG.

TABLE 4.4: A summary of average CPU times for C1 instances

$\|\mathbf{M}\|$	μ	AVG. t_C	AVG. t_{2PDIS}	AVG. t_{2PAGG}	AVG. t_{DIS}	AVG. t_{AGG}
5	1	0.105	0.020	0.025	0.012	0.012
	2	0.297	0.203	0.238	0.125	0.127
	3	0.350	0.137	0.183	0.093	0.100
	4	0.523	0.240	0.268	0.172	0.173
10	3	1.678	0.133	0.197	0.182	0.188
	5	1.153	0.263	0.323	0.220	0.225
	7	1.207	0.373	0.643	0.385	0.377
	9	1.325	0.427	0.482	0.235	0.508
15	4	1.920	0.390	4.038	2.975	2.997
	7	1.652	0.237	0.858	0.168	0.178
	10	2.520	0.377	0.667	0.445	0.353
	13	3.647	1.318	1.073	0.730	0.755
20	6	2.088	0.643	5.463	0.678	0.683
	10	3.608	0.525	1.062	0.595	0.472
	14	3.562	0.707	0.715	0.728	0.748
	18	4.647	1.330	0.862	1.122	1.145

Table 4.4 shows that the results obtained from C1 instances are similar to each other, most likely due to the small sized structure of the instances. The results shown in Table 4.4 are also depicted in a line chart in Figure 4.4, to observe the small but important differences among the considered solution techniques.

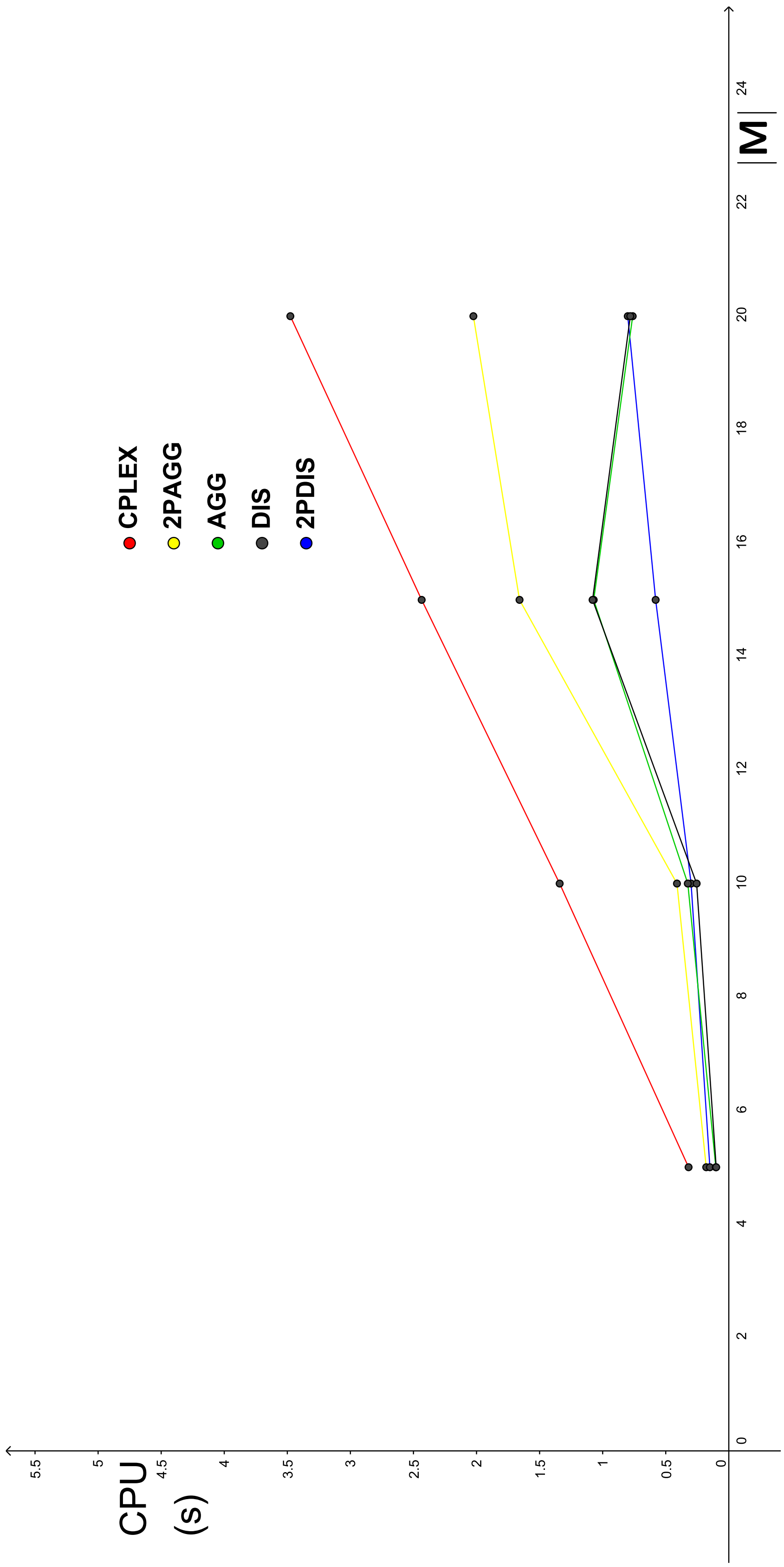


FIGURE 4.4: CPU time comparison for C1 instances

As can be seen from Figure 4.4, all presented solution algorithms based on Benders decomposition are more efficient than CPLEX 12.5 in terms of solution times. Among the proposed algorithms, 2PDIS is the most efficient and 2PAGG the least efficient. The performance of DIS and AGG are similar for this set of instances. Table 4.5, which presents the average number of iterations η_{2PDIS}, η_{2PAGG}, η_{DIS} and η_{AGG} needed for 2PDIS, 2PAGG, DIS and AGG, also supports the obtained findings from the solution time analysis. The average number of iterations needed for 2PAGG is more than the other algorithms.

Table 4.5: A summary table of average number of iterations

$\vert\mathsf{M}\vert$	μ	AVG. η_{2PDIS}	AVG. η_{2PAGG}	AVG. η_{DIS}	AVG. η_{AGG}
5	1	3	4.50	2.00	2.00
	2	3	5.67	2.00	2.00
	3	3	7.00	2.00	2.00
	4	3	7.83	2.00	2.00
10	3	3	6.67	2.17	2.17
	5	3	8.67	2.17	2.00
	7	3	10.17	2.33	2.17
	9	3	10.83	2.17	3.00
15	4	3	7.50	2.17	2.00
	7	3	9.67	2.00	2.00
	10	3	11.33	2.17	2.17
	13	3	12.67	2.00	2.00
20	6	3	7.67	2.00	2.00
	10	3	11.33	2.33	2.17
	14	3	12.83	2.00	2.00
	18	3	14.00	2.00	2.00

4.4.3 C2 Instances

A total of 96 instances were solved with CPLEX 12.5, DIS, AGG, 2PDIS and 2PAGG. The results can be found in Tables B.5-B.8 in Appendix B. None of the instances can be solved by DIS or AGG, furthermore, only 76 instances (79.17% of the total) can be solved by CPLEX 12.5, and only 90 instances (93.75% of the total) can be solved by 2PAGG due to limited RAM. All 96 instances can be solved to optimality by 2PDIS.

Using CPLEX 12.5, all the instances with 18 demand points, 21 instances with 36 demand points, 15 instances with 54 demand points , and 12 instances with 72 demand points can be solved. All instances with 18, 36 and 54 demand points can be solved by 2PAGG, whereas six instances with 72 demand points cannot be. Table 4.8 presents the average CPU solution times for CPLEX 12.5, 2PDIS and 2PAGG for the 18 demand point instances and the average CPU times for 2PDIS and 2PAGG for the 36 and 54 demand point instances. To have a meaningful performance comparison of 2PDIS, 2PAGG, and CPLEX 12.5, Figure 4.5a only shows the finding in Table 4.8 on a line chart for instances with 18 demand points since all of the these instances can be solved by all three proposed solution techniques. Figure 4.5b and 4.5c present performance comparisons of 2PAGG and 2PDIS for instances with 36 and 54 demand points, respectively. The instances with 72 demand points are not included in this comparison because of the 2PAGG's failure of solving six instances with 72 demand points.

As can be clearly seen in Figure 4.5a, the performance of 2PDIS and 2PAGG are better than CPLEX 12.5. For the rest of the instances, CPLEX 12.5 only performed better in solving instances with $f_i = 1, \forall i \in \mathbf{M}$. For all other instances, 2PDIS and 2PAGG performed better than CPLEX 12.5.

Figure 4.5 also shows that 2PDIS performs better than 2PAGG on the instances with 18 demand point instances. On the other hand, 2PAGG performs better on the 54 demand point instances. In the case of the 36 demand point instances, an increase in the capacity of a new facility leads to a performance drop for 2PAGG. Specifically, the performance of 2PDIS is higher than that of 2PAGG when $\mu = 32$. For smaller μ values, 2PAGG is more efficient. It can be concluded that 2PAGG can solve instances to optimality more efficiently, when number of demand points is small. However, it has to be noted that 2PDIS is more robust than 2PAGG, since all 96 instances can be solved by 2PDIS, which makes it unique among the proposed solution techniques.

Another interesting point is the time spent in seconds for solving MP in each iteration. To solve the 18, 36, 54 and 72 demand point instances by 2PDIS, the average number of iterations needed, the average number of iterations needed (η_{2PDIS}), the average number of iteration that the MP is modelled and solved as MILP (η_{MILP}), the average time spent to solve the MP modelled as an LP (t_{LP}), the average time required to solve the MP modelled as an MILP (t_{MILP}) and the ratio between the average t_{LP} and the average t_{MILP} are presented in Table 4.6.

Table 4.6 and Table 4.7 give similar pictures 2PAGG.

Table 4.6: Average MP solution times obtained by 2PDIS

$\lvert\mathsf{M}\rvert$	AVG. η_{2PDIS}	AVG. η_{MILP}	AVG. t_{LP}	AVG. t_{MILP}	Ratio
18	23.75	1	0.24	117.57	0.2%
36	59.25	1	1.14	385.78	0.3%
54	73.25	1	1.80	875.94	0.2%
72	86.83	1	2.73	1213.52	0.2%

Table 4.7: Average MP solution times obtained by 2PAGG

$\lvert\mathsf{M}\rvert$	AVG. η_{2PDIS}	AVG. η_{MILP}	AVG. t_{LP}	AVG. t_{MILP}	Ratio
18	25.29	1	0.55	228.67	0.2%
36	40.66	1	2.02	594.15	0.3%
54	49.25	1	3.93	611.06	0.6%

TABLE 4.8: Average CPU times of CPLEX, 2PDIS and 2PAGG for instances with 18, 36 and 54 demand points

		$\lvert\mathbf{M}\rvert = 18$			$\lvert\mathbf{M}\rvert = 36$			$\lvert\mathbf{M}\rvert = 54$	
μ	AVG. t_C	AVG. t_{2PDIS}	AVG. t_{2PAGG}	μ	AVG. t_{2PDIS}	AVG. t_{2PAGG}	μ	AVG. t_{2PDIS}	AVG. t_{2PAGG}
5	1471.16	96.18	127.89	10	184.51	121.69	16	539.01	526.96
9	1260.96	143.24	227.28	18	427.34	381.10	27	857.20	745.77
12	2586.35	161.15	374.22	25	585.88	347.80	37	1403.14	1189.24
16	1467.45	249.22	314.80	32	919.52	2044.49	48	1752.28	1094.50

Table 4.6 and Table 4.7 show that an average time spent to solve the MP as an MILP is significantly more than solving the MP as an LP for both 2PDIS and 2PAGG. From this observation, the reason why DIS and AGG fails to solve large instances is the need to solve an MILP in each iteration of the algorithm, which causes RAM problems. On the other hand, while solving these instances with 2PDIS and 2PAGG, for each instance, only one iteration is needed to solve the MILP, which saves from memory and time.

Finally, a solution to one of the 96 generated instances is shown in Figure 4.6. In this instance, the total number of demand points is 36, the fixed cost to open a new facility is 20, and the maximum number of demand point that a single facility can serve is 25.

4.4.4 C3 Instances

As is shown in Table 4.3, the C3 instances are the largest instance in terms of $|\mathbf{M}|$ and the number of available nodes. These 96 generated instances were only solved with 2PDIS.

The results, the full set of which are given in Tables B.9-B.12 in Appendix B, show that the limits of 2PDIS are also reached. For the instances with 10 and 20 demand points, only the instances with a fixed cost of 1 can be solved by this solution technique. For the instances with 30 and 40 demand points, the algorithm performs better. Overall, 23 out of 24 instances can be solved with the proposed algorithm.

Finally, an optimal solution of the instance with a 30 demand points, $\mu = 15$ and a fixed cost of 5 is presented in Figure 4.7 as an example.

4.5 Conclusions

There are two main issues that are investigated in this paper. Firstly, to the best of our knowledge, the capacitated version of restricted continuous facility location problems have never been studied before in the existing literature. To do that, the discretisation method presented in Chapter 3 is used and a new modelling framework for this problem type is introduced. Secondly, there are various types of Benders decomposition algorithm

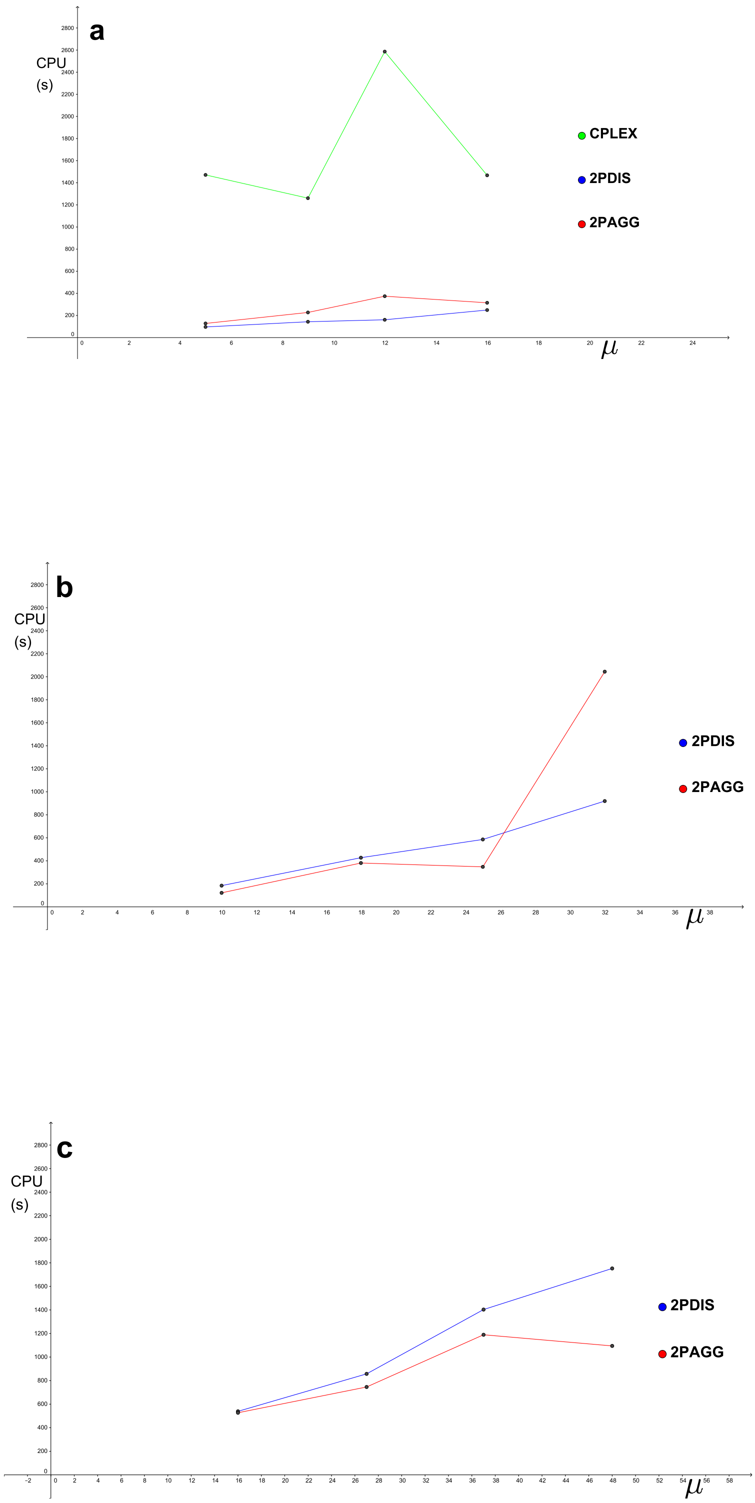


FIGURE 4.5: CPU time comparison for C2 instances with 18 demand points

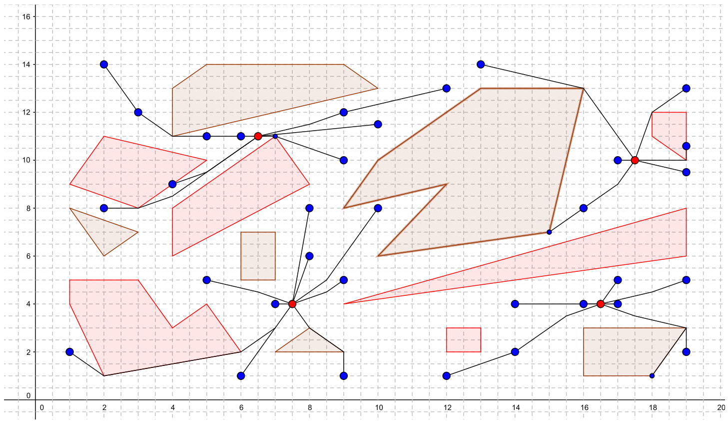


FIGURE 4.6: A C2 instance with $|\mathbf{M}| = 36$, $\mu = 25$, $f_i = 20$

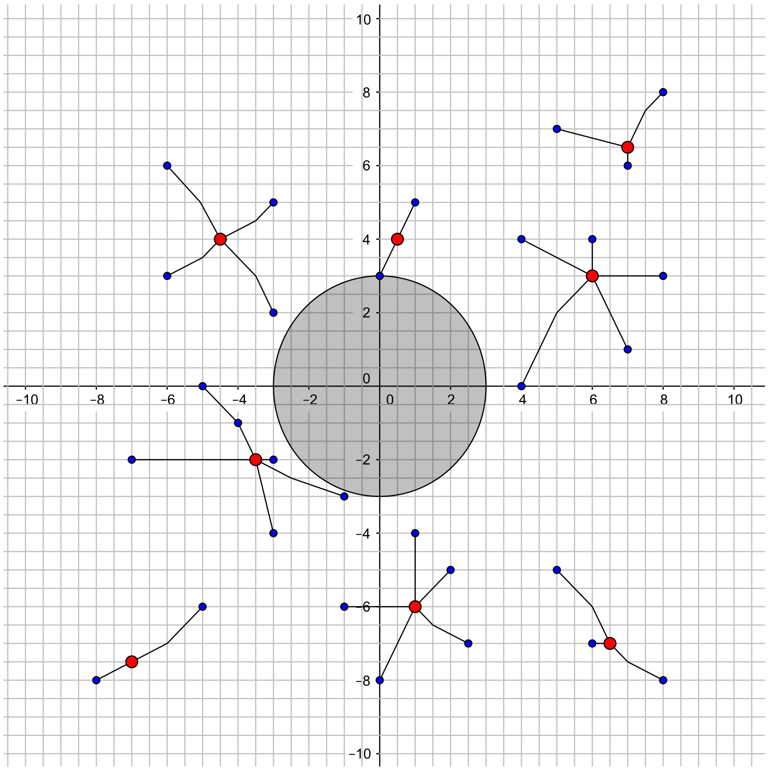


FIGURE 4.7: A C3 instance with $|\mathbf{M}| = 30$, $\mu = 15$, $f_i = 5$

introduced in the literature to improve the performance of the solution process. These methods are adapted to the current problem and applied to a large set of instances that were generated from two previous instances from the literature and a newly presented instance. Computational results show the most efficient algorithm to solve the problem at hand is a 2-phase Benders decomposition with disaggregated cuts. Limitations of this algorithm are also presented. I observe that the when the total number of the arcs used to discretise the continuous plane increases, the performance of the algorithm drops, causing memory problems that prevent algorithm from finding a solution, even with a high memory computer. It can be concluded that this method may be more useful for instances with a large number of barriers, which leads to a drop in the number of arcs needed to discretise the continuous space.

Chapter 5

Conclusions

5.1 Overview

This thesis has studied a number of the continuous restricted facility location problems. The content and the scientific contributions of each chapter are briefly summarised in Section 5.2. An overview of the research outputs from the thesis is presented in Section 5.3. The limitations of the proposed techniques and an identification of future research directions are presented in Sections 5.4 and 5.5, respectively.

5.2 Summary of the Main Findings of the Thesis

As mentioned in Section 1.2, the aim of the thesis is to investigate modelling and solution techniques for both uncapacitated and capacitated facility location problems that have deterministic and stochastic restrictions on locations and transportations. To do so, three research papers were presented in the three main body chapters of the thesis to reach the research objectives presented in Section 1.2.

A literature review of both deterministic and stochastic uncapacitated continuous restricted facility location problems are presented in Chapters 2 and 3, respectively. The extensive review on these problems detected several interesting points. Firstly, the deterministic case is studied more than the stochastic case, however, the attention on the stochastic case has been increasing in recent years. To the best of our knowledge, the first study on a deterministic case was published by Katz and Cooper (1981), whereas the first study on a stochastic case was first published by Canbolat and Wesolowsky (2010). I believe that there is still more to build on the stochastic cases since it is still in an early stage of research. As far as I know, there is also one study (Bischoff et al. 2009) focused on the deterministic multi-facility location problem, which shows that there is still room to investigate and improve on this variant of the problem. Secondly, for both deterministic and stochastic cases, each problem had been addressed individually based on various characteristics such as the shape of a restricted region, the distance, and the objective function. Table 2.1 visualises this finding for the deterministic problems and this was one of the motivations for focusing on these problems'. Thirdly, I realise that the capacitated version of the restricted facility location problems have not considered yet in the literature. This problem type can also be an interesting area for future research.

In Chapter 2, a general modelling framework for continuous facility location problems with either forbidden regions or barriers is described using the concept of phi-objects. It was shown that the flexibility of phi-objects allows one to model problems with both arbitrarily shaped forbidden regions and barriers, with different objective functions, e.g. median and centre problem, and with various types of distance metrics in a single modelling framework. To test the proposed approach, instances from the existing literature and a real life instance inspired by an archipelago near the Turkish city Istanbul were modelled and solved to optimality. The resulting models were MILP, NLP and MILNP depending on the shape of the restricted region and the considered distance metric. The results obtained showed that all instances could be solved in less than one second. The ability to model and solve the real-life instance showed the advantage of using phi-objects to model various random geometric shapes with negligible approximation errors. It was also shown that solving the proposed modelling framework is suitable by an appropriate off the shelf solver for multi-facility problems with forbidden regions and single facility problems with barriers. This finding from Chapter 2 motivated us to focus more on the multi-facility cases with barriers in Chapters 3 and 4.

In Chapter 3, two different networks were defined to transform the continuous space into a discrete counterpart to obtain high quality approximated solutions for the original continuous multi-facility problem. All the continuous single facility instances with forbidden regions and barriers were approximated on these two defined networks, namely the unit square network and the unit rectangle network. Since the global optimal solutions of these continuous problems were obtained in Chapter 2, it was possible to evaluate the approximation quality of the defined networks. The findings showed that the percentage gap between the optimal solutions of the approximations for the unit rectangle network was less than 1% for all instances considered. Thus, discretisation with the unit rectangle network could be used to solve multi-facility problems with barriers with low approximation error. Based on this finding, the continuous space of the both stochastic and capacitated versions were discretised by the unit rectangle network and solved to optimality in Chapters 3 and 4, respectively.

Besides the discretisation technique, modelling and solution techniques were presented for the stochastic restricted facility location problem with probabilistic barriers in Chapter 3. The modelling formulation was based on multi-commodity flows with unknown destinations, where the origin nodes of the flows were thought as the locations of the demand points and the unknown destination nodes were thought as the locations of the new facilities. Computational experiments in Chapter 3 showed that the adaptation of multi-commodity flow formulations to a facility location problem was an efficient approach to model both single and multi-facility problems, considering either deterministic or probabilistic restrictions. There were several solution techniques proposed to solve the problem based on Benders decomposition. It was also shown that the subproblem could further be decomposed into multiple shortest path problems. Thus, these subproblems could be solved by Dijkstra's algorithm instead of using an off the shelf solver. In total, 24 deterministic and 360 stochastic instances were solved. It was observed that using Dijkstra's algorithm to solve subproblems boosts the performance of the algorithm. The reason is that the average time spent on solving the subproblem with Dijkstra's algorithm was less than the average time spent solving the subproblem with a commercial solver. The presented algorithm was also more efficient than the commercial solver on solving the general model, which showed an advantage of using it as a solution approach for the problem modelled as multi-commodity flow formulations. As a managerial insight, the solutions showed the optimal paths from each demand point to the new facilities under each scenario, thus a decision maker could choose the optimal distribution path of goods from new facilities to demand points for any considered possible scenario.

In Chapter 4, the main concentration was on the capacitated version of the continuous restricted facility location problem. The discretisation technique, proposed in Chapter 3, was also used in Chapter 4 to transform the continuous problem into a network optimisation problem. An adaptation of the modelling technique presented in Chapter 3 was also used to model the capacitated version. Since a capacity constraint existed in the model, the integrality property did not hold. Thus, the problem was modelled as an MILP. An exact algorithm based on Benders decomposition and variants that were expected to increase performance were proposed as solution techniques for the problem at hand. Extensive computational experiments were done to test the performance and capabilities of the solution techniques. The results showed that 2PDIS was the most capable algorithm among the presented solution techniques, since 2PDIS could solve most instances. Another important finding was that using a 2-phase method improves the performance of Benders decomposition significantly for these problems, because this method minimised the number of iterations that were needed to solve the master problem as a MILP. As a managerial insight, the modelling technique showed the tradeoff between the cost of opening new facilities and the number of new facilities need to serve the demand points.

5.3 Research Outputs

An overview of the research outputs of the thesis is given below.

Publication:

Oğuz M., Bektaş T., Bennell J., Fliege J. (2016) ”A modeling framework for solving restricted planar location problems using phi-objects.” *Journal of the Operational Research Society* 67(8):1080–1096.

Conference Presentations:

Oğuz M, Bektaş T, Bennell J ”Discretisation, Multi-commodity Flows and Benders Decomposition for Restricted Stochastic Continuous Location Problems.” 2016 *INFORMS International Conference*, June 2016. Waikoloa Village, HI, USA

Oğuz M, Bektaş T, Bennell J ”Solving Restricted Facility Location Problems on Discrete Networks using Multi-commodity Flows.” *Southampton Business School PhD Conference*, March 2016. Southampton, United Kingdom

Oğuz M, Bektaş T, Bennell J, Fliege J ”A modeling framework for solving restricted planar location problems using phi-objects.” 20^{th} *Conference of the International Federation of Operational Research Societies*, July 2014. Barcelona, Spain

Oğuz M, Bektaş T, Bennell J "Literature Survey on Disaster Management Applications." 1^{st} *Young CORMSIS Conference*, June 2013. Southampton, United Kingdom

5.4 Limitations of the Research Results

I acknowledge some limitations and shortcomings of this research.

- Due to the difficulty of the problems considered especially in Chapter 3 and Chapter 4, the continuous space was discretised so that an exact algorithm could be used to solve an approximation of the original problem. The performance of the algorithm was shown to be promising, but optimality of the original problem is not necessarily guaranteed.
- I have assumed that all problems considered here are time-independent, however, particularly in emergency management, the occurrence time of barriers due to disasters may have a significant effect on planning an effective disaster response plan. For example, the probability of a flood is more in a rainy season than the dry season.
- I have assumed that all parameters are deterministic. Even in the stochastic cases, I discretise the random events by defining a set of scenarios, and developed a deterministic equivalent of the stochastic model. This study has not ventured into a full exploration of dynamic environments.

5.5 Future Research Directions

I identify the following three areas as promising lines of future research:

- Data: Dynamism can be considered in the problem definition. Dynamic problems arise over a certain period of time. In dynamic facility location problems, a decision can be made to change capacity or open facilities at various points in time. The main aim of the dynamic facility location problems is to decide where and when to use more capacity to satisfy the customer demand at a cost as low as possible. Most of the dynamic facility location problems are the multi period extensions of classical facility location problems. To the best of my knowledge, the restrictions on location and transportation have not been considered in dynamic facility location yet, and these problems can be considered as a multi period extension.

- Techniques: Further studies should be done to develop exact algorithms, specifically methods for continuous restricted problem without the need for discretisation. A method based on Lagrangean relaxation can be used to obtain strong lower bounds for the global optima of the problem with a larger scale. Especially, in the capacitated version of the restricted problem, both CPLEX 12.5 and Benders decomposition approach could not obtain the optimality in some instances, due to the size of the problems. Lagrangean relaxation can be an effective approach to obtain strong lower bounds for these large instances, and also, a heuristic approach can be constructed to obtain an upper bound.

- Case Study: This study mainly focuses on the instances defined previously in the literature. On the other hand, restricted location problems have wide application. An extensive realistic case study can be considered to see the effects of both deterministic and stochastic restrictions on operations. As a case study of a deterministic problem, one could consider distributing goods from optimally located new facilities to demand points in a mountainous terrain. In this case, mountains are the barriers that restricts the facility location and transportation of goods. As a case study for the stochastic problem, locating and transporting in areas that are prone to flooding, such as Cumbria, UK, would be interesting, and might help to improve transportation in the region.

Appendix A

Supplement to Chapter 3

The results of the 360 instances of Section 3.6.4 can be found in Tables A.1-A.12. LIMIT means the memory requirements to solve that instance is too high.

Table A.1: Probability set 1 with 18 demand points

θ	NS	$v^\star$	t_C	t_D	AVG. t_M	η_i
1	2	115.53	78.09	3.73	0.01	4
	3	112.77	122.23	4.25	0.02	3
	4	113.05	178.41	5.85	0.035	3
	5	112.53	231.17	7.09	0.025	3
	6	112.15	295.61	5.45	0.04	1
2	2	86.11	92.32	4.34	0.015	5
	3	83.29	141.15	13.82	0.02	10
	4	82.86	198.53	8.86	0.023	5
	5	82.12	276.8	13.87	0.028	6
	6	81.52	351.52	16.17	0.032	6
3	2	64.74	84.93	7.68	0.01	8
	3	62.71	138.72	15.8	0.017	10
	4	62.55	190.03	10.87	0.02	6
	5	62.08	245.03	16.12	0.027	7
	6	61.72	311.84	22.17	0.033	8
4	2	49.37	75.35	6.09	0.01	6
	3	48.48	116.52	9.52	0.016	6
	4	48.68	166.66	11.51	0.026	6
	5	48.5	214.53	13.46	0.024	6
	6	48.38	275.37	16.44	0.03	6
5	2	41.98	72.94	6.69	0.013	7
	3	41.12	113.3	17.91	0.017	12
	4	41.35	156.24	17.36	0.023	9
	5	41.23	210.03	19.04	0.029	8
	6	41.14	267.49	20.19	0.032	7
6	2	36.01	68.82	7.32	0.018	5
	3	35.33	111.5	21.49	0.02	13
	4	35.56	150.8	22.04	0.021	11
	5	35.48	191.23	24.22	0.023	10
	6	35.43	242.69	23.85	0.033	8

Table A.2: Probability set 2 with 18 demand points

θ	NS	$v^\star$	t_C	t_D	AVG. t_M	η_i
1	2	114.96	78.96	2.78	0.01	3
	3	115.44	124.75	4.08	0.015	3
	4	113.14	180.75	5.42	0.025	3
	5	112.29	238.33	7.14	0.03	3
	6	111.67	299.75	8.35	0.03	3
2	2	85.52	85.52	5.22	0.01	6
	3	86.8	86.8	10.91	0.02	8
	4	83.27	83.27	8.83	0.023	5
	5	81.79	81.79	15.28	0.025	7
	6	80.52	80.52	16.02	0.032	6
3	2	64.29	83.65	7.91	0.014	8
	3	64.78	134.11	9.61	0.017	7
	4	62.83	185.88	10.96	0.02	6
	5	61.5	248.19	21.12	0.033	9
	6	61.13	307.87	17.79	0.03	7
4	2	49.49	73.45	11.19	0.013	12
	3	49.48	111.89	7.85	0.012	6
	4	48.66	158.84	12.51	0.022	7
	5	48.76	222.96	29.93	0.028	13
	6	48.37	262.87	19.33	0.032	7
5	2	42.06	71.81	12.2	0.011	12
	3	41.61	112.27	12.82	0.016	9
	4	41.3	157.04	16.43	0.02	8
	5	41.2	213.85	14.67	0.024	6
	6	41.21	259.45	22.39	0.029	8
6	2	36.13	69.96	7.54	0.013	7
	3	35.65	108.47	14.28	0.017	10
	4	35.48	153.16	14.79	0.02	7
	5	35.38	196.32	44.19	0.027	19
	6	35.46	245.09	37.37	0.029	13

Table A.3: Probability set 3 with 18 demand points

θ	NS	$v^\star$	t_C	t_D	AVG. t_M	η_i
1	2	113.69	80.97	2.79	0.015	3
	3	114.37	128.87	4.05	0.02	3
	4	112	180.33	5.48	0.02	3
	5	112.48	247.36	7.01	0.025	3
	6	112.21	307.76	8.4	0.035	3
2	2	83.5	91.94	5.32	0.014	6
	3	84.87	152.87	7.63	0.016	6
	4	80.89	215.19	10.12	0.022	6
	5	82.27	294.78	13.08	0.026	6
	6	80.89	357.84	18.01	0.03	7
3	2	63.11	84.07	6.85	0.015	7
	3	63.87	135.32	12.21	0.018	9
	4	61.48	183.75	12.65	0.018	7
	5	62.15	259.71	16.02	0.028	7
	6	61.38	308.65	17.85	0.062	7
4	2	49.48	75.58	5.59	0.012	6
	3	49.1	114.54	12.63	0.019	8
	4	48.76	160.56	12.85	0.02	7
	5	48.62	208.86	17.08	0.026	8
	6	48.66	269.61	21.49	0.029	8
5	2	42.19	70.5	12.4	0.015	11
	3	41.55	110.77	17.28	0.019	12
	4	41.62	160.61	13.48	0.02	7
	5	41.22	211.63	16.47	0.023	7
	6	41.28	261.22	20.56	0.028	7
6	2	36.35	71	10.09	0.013	8
	3	35.68	108.48	19.08	0.018	11
	4	35.73	149.98	67.1	0.022	35
	5	35.36	194.38	16.37	0.022	7
	6	35.41	249.31	23.27	0.027	7

Table A.4: Probability set 1 with 36 demand points

θ	NS	$v^{\star}$	t_C	t_D	AVG. t_M	η_i
1	2	256.76	181.03	5.18	0.02	3
	3	248.76	304.07	8.1	0.035	3
	4	248.36	483.1	10.59	0.035	3
	5	246.61	598.13	13.58	0.04	3
	6	245.21	789.51	16.25	0.06	3
2	2	166.93	179.37	10.46	0.026	6
	3	161.63	326.32	15.12	0.03	6
	4	161.52	503.33	25.13	0.045	7
	5	160.26	719.59	30.31	0.05	7
	6	159.32	942.73	30.87	0.064	6
3	2	134.29	172.39	15.39	0.02	8
	3	130.22	289.68	30.55	0.03	11
	4	130.21	436.18	33.96	0.042	10
	5	129.41	618.52	37.99	0.05	9
	6	128.73	875.5	65.39	0.062	13
4	2	108.85	156.07	14.2	0.02	7
	3	106.51	264.47	21.45	0.033	8
	4	107.07	375.98	37.36	0.039	11
	5	106.71	569.4	41.84	0.049	9
	6	106.45	787.43	43.52	0.054	8
5	2	92.81	148.6	13.04	0.02	7
	3	91.22	239.84	21.53	0.029	8
	4	91.39	326.82	24.08	0.035	7
	5	90.99	466.4	43.14	0.05	9
	6	90.69	642.59	39.2	0.057	7
6	2	84.31	147.59	45.65	0.023	25
	3	82.77	239.12	31.2	0.03	11
	4	83.14	338.87	64.58	0.038	15
	5	82.85	472.14	43.14	0.049	11
	6	82.61	612.28	68.87	0.058	10

Table A.5: Probability set 2 with 36 demand points

θ	NS	$v^\star$	t_C	t_D	AVG. t_M	η_i
1	2	253.9	192.21	5.2	0.025	3
	3	257.3	303.75	7.78	0.03	3
	4	248.81	462.73	10.29	0.04	3
	5	245.21	616.36	16.09	0.055	3
	6	242.23	801.72	15.86	0.06	3
2	2	165.46	178.02	8.44	0.018	5
	3	167.95	317.14	18.46	0.033	7
	4	161.63	448.68	23.19	0.04	7
	5	159.82	692.86	21.4	0.05	5
	6	157.94	874.23	25.43	0.06	5
3	2	133.45	171.82	14.84	0.02	8
	3	134.61	304.05	18.93	0.032	7
	4	130.81	436.34	23.29	0.038	7
	5	127.95	609.2	37.94	0.055	7
	6	128.02	873.49	35.79	0.057	7
4	2	109.12	155.61	20.61	0.024	8
	3	107.74	253.39	31.15	0.03	12
	4	107	388.55	29.54	0.04	8
	5	106.84	561.73	30.54	0.048	7
	6	106.81	756.25	47.83	0.051	8
5	2	93.07	142.72	17.34	0.021	9
	3	91.94	230.77	22.7	0.029	9
	4	91.28	342.89	23.33	0.038	6
	5	90.99	450.68	31.16	0.05	7
	6	91	617.71	38.1	0.055	7
6	2	84.52	148.06	21.39	0.019	11
	3	83.43	229.01	60.37	0.036	14
	4	83.07	349.95	56.56	0.039	12
	5	82.89	459.91	47.63	0.046	9
	6	82.97	633.22	51.82	0.057	8

Table A.6: Probability set 3 with 36 demand points

θ	NS	$v^\star$	t_C	t_D	AVG. t_M	η_i
1	2	247.72	186.96	5.18	0.02	3
	3	252.81	298.36	7.78	0.035	3
	4	242.87	442.03	10.34	0.04	3
	5	245.47	586.26	13.63	0.045	3
	6	244.21	746.26	15.75	0.06	3
2	2	162.27	181.26	11.04	0.026	6
	3	164.43	320.36	16.5	0.03	6
	4	158.36	462.18	17.06	0.04	5
	5	159.79	639.54	21.94	0.048	5
	6	157.84	872.45	25.74	0.058	5
3	2	131.72	165.2	16.21	0.026	8
	3	132.31	285.37	21.27	0.029	8
	4	128.53	422.22	40.63	0.045	9
	5	129.73	613.19	25.26	0.048	6
	6	127.86	881.03	41.33	0.057	8
4	2	109.68	157.94	14	0.021	8
	3	107.63	250.12	23.51	0.03	9
	4	107.45	400.81	49.32	0.043	13
	5	107.21	687.44	37.72	0.047	8
	6	106.56	716.43	34.04	0.062	6
5	2	93.58	144.92	14.81	0.021	8
	3	91.97	235.69	16.84	0.032	6
	4	91.66	359.52	33.33	0.039	9
	5	91.43	460.28	165.27	0.069	29
	6	90.85	611.5	36.99	0.055	7
6	2	84.93	147.07	25.14	0.02	10
	3	83.6	234.24	25.03	0.031	8
	4	83.55	359.87	221.33	0.062	54
	5	83.28	453.72	159.53	0.06	32
	6	82.52	656.98	69.68	0.058	11

Table A.7: Probability set 1 with 54 demand points

θ	NS	$v^{\star}$	t_C	t_D	AVG. t_M	η_i
1	2	369.56	LIMIT	7.81	0.03	3
	3	358.74	LIMIT	12.05	0.05	3
	4	359	LIMIT	15.79	0.06	3
	5	356.97	LIMIT	35.78	0.083	5
	6	355.48	LIMIT	44.98	0.09	5
2	2	262.04	363.15	14.54	0.032	6
	3	252.98	865.07	27.63	0.045	7
	4	252.18	LIMIT	35.27	0.063	7
	5	250.04	LIMIT	38.6	0.07	6
	6	248.05	LIMIT	39.04	0.085	5
3	2	210.99	326.89	26.37	0.033	8
	3	203.79	532.9	38	0.045	9
	4	203.19	LIMIT	43.18	0.056	8
	5	201.59	LIMIT	52.55	0.07	8
	6	200.45	LIMIT	71.49	0.093	9
4	2	175.2	280.4	26.63	0.031	10
	3	170.94	444.24	43.58	0.045	11
	4	171.39	909.1	56.68	0.057	10
	5	170.54	LIMIT	70.14	0.073	10
	6	169.92	LIMIT	89.65	0.1	10
5	2	153.33	246.59	30.8	0.029	11
	3	149.36	424.18	64.37	0.046	14
	4	149.78	630.17	54.29	0.056	10
	5	149.12	LIMIT	89.95	0.076	13
	6	148.59	LIMIT	556.75	0.186	51
6	2	136.07	236.94	25.43	0.029	10
	3	132.97	382.36	35.14	0.043	8
	4	133.47	591.61	49.83	0.056	9
	5	132.97	LIMIT	63.85	0.065	9
	6	132.58	LIMIT	162.27	0.086	19

Table A.8: Probability set 2 with 54 demand points

θ	NS	$v^{\star}$	t_C	t_D	AVG. t_M	η_i
1	2	366.37	LIMIT	7.79	0.02	3
	3	368.67	LIMIT	12.31	0.045	3
	4	359.37	LIMIT	15.93	0.055	3
	5	354.19	LIMIT	27.92	0.077	4
	6	352.67	LIMIT	24.54	0.09	3
2	2	259.56	368.92	15.37	0.036	6
	3	263.51	894.06	27.93	0.047	7
	4	252.59	LIMIT	35.72	0.055	7
	5	249.65	LIMIT	45.4	0.073	7
	6	246.41	LIMIT	46.32	0.09	6
3	2	209.97	316.56	22.35	0.029	8
	3	210.05	668.4	38.26	0.043	8
	4	204.4	LIMIT	49.04	0.061	9
	5	199.66	LIMIT	65.04	0.072	10
	6	199.78	LIMIT	70.52	0.094	9
4	2	175.44	284.54	20.92	0.027	8
	3	173.7	464.3	35.89	0.044	9
	4	171.52	712.61	54.28	0.061	10
	5	170.36	LIMIT	61.01	0.073	9
	6	170.19	LIMIT	76.18	0.081	9
5	2	153.77	249.04	41.38	0.031	14
	3	151.88	403.49	43.26	0.04	11
	4	149.9	666.02	136.65	0.065	19
	5	149.14	LIMIT	72.2	0.068	11
	6	149	LIMIT	201.87	0.101	23
6	2	136.77	230.83	21.57	0.03	10
	3	135.38	366.63	43.67	0.046	11
	4	133.6	628.19	59.56	0.058	9
	5	133.31	LIMIT	59.03	0.066	10
	6	133.05	LIMIT	82.02	0.094	12

Table A.9: Probability set 3 with 54 demand points

θ	NS	$v^\star$	t_C	t_D	AVG. t_M	η_i
1	2	359.98	LIMIT	7.78	0.03	3
	3	363.94	LIMIT	11.99	0.05	3
	4	353.48	LIMIT	15.92	0.06	3
	5	356.01	LIMIT	35.64	0.073	5
	6	354.17	LIMIT	34.19	0.083	4
2	2	253.41	346.9	15.64	0.036	6
	3	257.54	567.92	26.96	0.045	7
	4	247.24	LIMIT	36.93	0.07	7
	5	249.26	LIMIT	46.68	0.077	7
	6	246.45	LIMIT	47.5	0.09	6
3	2	207.11	314.87	24	0.03	9
	3	206.92	567.92	67.37	0.055	14
	4	201.08	LIMIT	63.6	0.061	12
	5	202.54	LIMIT	72.06	0.069	11
	6	199.72	LIMIT	183.18	0.0122	22
4	2	175.62	272.22	45.08	0.041	16
	3	172.98	445.73	44.73	0.046	11
	4	171.4	764.63	60.16	0.059	11
	5	171.15	LIMIT	69.85	0.07	10
	6	170.23	LIMIT	113.94	0.093	13
5	2	154.58	264.39	34.97	0.033	13
	3	151.14	418.36	79.52	0.045	16
	4	150.55	680.94	58.78	0.058	11
	5	150.12	LIMIT	73.28	0.069	11
	6	149.34	LIMIT	437.63	0.142	41
6	2	138.18	248.54	28.28	0.032	10
	3	134.58	386.31	43.24	0.043	10
	4	134.52	631.39	74.15	0.064	11
	5	134.14	LIMIT	55.98	0.069	8
	6	133.36	LIMIT	101.1	0.085	12

Table A.10: Probability set 1 with 72 demand points

θ	NS	$v^\star$	t_C	t_D	AVG. t_M	η_i
1	2	507.56	LIMIT	10.69	0.04	3
	3	492.53	LIMIT	15.68	0.05	3
	4	491.88	LIMIT	21.45	0.08	3
	5	488.49	LIMIT	27.26	0.095	3
	6	486.1	LIMIT	32.91	0.115	3
2	2	374.44	LIMIT	23.12	0.038	7
	3	362.88	LIMIT	53.9	0.072	10
	4	362.14	LIMIT	64.08	0.091	8
	5	359.23	LIMIT	91.4	0.109	10
	6	357.17	LIMIT	64.08	0.126	6
3	2	300.15	LIMIT	55	0.052	13
	3	290.67	LIMIT	55.01	0.057	10
	4	289.89	LIMIT	73.47	0.077	10
	5	287.66	LIMIT	135.22	0.108	15
	6	286.06	LIMIT	136.08	0.133	9
4	2	246.61	LIMIT	43.05	0.041	12
	3	238.51	LIMIT	56.17	0.054	11
	4	238.76	LIMIT	68.98	0.072	10
	5	237.4	LIMIT	94.95	0.097	10
	6	236.4	LIMIT	94.62	0.114	8
5	2	214.42	LIMIT	49.59	0.037	7
	3	207.68	LIMIT	89.46	0.059	8
	4	207.77	LIMIT	111.86	0.082	7
	5	206.47	LIMIT	135.51	0.103	9
	6	205.52	LIMIT	494.79	0.183	7
6	2	189.57	LIMIT	51.5	0.038	12
	3	184.9	LIMIT	78.45	0.059	13
	4	185.2	LIMIT	85.57	0.071	11
	5	184.17	LIMIT	110.66	0.09	11
	6	183.4	LIMIT	195.97	0.12	13

Table A.11: Probability set 2 with 72 demand points

θ	NS	$v^{\star}$	t_C	t_D	AVG. t_M	η_i
1	2	502.59	LIMIT	11.16	0.045	3
	3	506.02	LIMIT	27.84	0.06	5
	4	492.15	LIMIT	21.43	0.075	3
	5	484.62	LIMIT	77.3	0.119	8
	6	482.75	LIMIT	44.23	0.117	4
2	2	371.86	LIMIT	42.34	0.044	12
	3	374.9	LIMIT	37.35	0.062	7
	4	362.73	LIMIT	76.13	0.086	10
	5	358.53	LIMIT	75.57	0.113	8
	6	354.65	LIMIT	79.44	0.138	7
3	2	298.9	LIMIT	28.42	0.039	8
	3	299.01	LIMIT	124.43	0.069	17
	4	291.67	LIMIT	77.07	0.081	10
	5	284.91	LIMIT	114.66	0.115	12
	6	285.44	LIMIT	94.89	0.111	9
4	2	245.74	LIMIT	40.52	0.042	11
	3	243.89	LIMIT	47.44	0.055	9
	4	239.48	LIMIT	70.99	0.078	10
	5	236.03	LIMIT	120.77	0.101	8
	6	235.93	LIMIT	389.08	0.129	16
5	2	214.11	LIMIT	58.85	0.041	11
	3	211.96	LIMIT	130.77	0.058	14
	4	208.38	LIMIT	129.68	0.085	16
	5	206.19	LIMIT	276.11	0.123	26
	6	205.65	LIMIT	135.31	0.112	11
6	2	190.22	LIMIT	73.37	0.038	11
	3	188.38	LIMIT	91.95	0.06	11
	4	185.55	LIMIT	122.95	0.075	14
	5	184.66	LIMIT	101.59	0.1	11
	6	184.02	LIMIT	124.19	0.117	10

Table A.12: Probability set 3 with 72 demand points

θ	NS	$v^{\star}$	t_C	t_D	AVG. t_M	η_i
1	2	492.65	LIMIT	10.87	0.035	3
	3	499.63	LIMIT	17.17	0.055	3
	4	483.91	LIMIT	21.61	0.08	3
	5	486.65	LIMIT	26.67	0.095	3
	6	484.57	LIMIT	31.35	0.115	3
2	2	365.11	LIMIT	16.44	0.038	5
	3	368.7	LIMIT	36.47	0.058	7
	4	356.05	LIMIT	113.95	0.089	17
	5	358.41	LIMIT	64.29	0.113	7
	6	354.71	LIMIT	122.91	0.13	11
3	2	295.73	LIMIT	35.97	0.042	10
	3	295.14	LIMIT	62.01	0.063	11
	4	287.44	LIMIT	145.93	0.098	19
	5	288.94	LIMIT	100.91	0.101	11
	6	285.08	LIMIT	1628.75	0.421	135
4	2	244.02	LIMIT	48.57	0.042	13
	3	244.02	LIMIT	39.01	0.046	10
	4	237.57	LIMIT	59.01	0.077	8
	5	238.35	LIMIT	171.01	0.102	11
	6	236.11	LIMIT	81.21	0.11	8
5	2	213.48	LIMIT	73.38	0.041	16
	3	210.51	LIMIT	87.13	0.058	13
	4	207.78	LIMIT	98.88	0.074	13
	5	207.82	LIMIT	185.84	0.114	18
	6	206.42	LIMIT	215.67	0.114	14
6	2	191.5	LIMIT	70.02	0.039	15
	3	187.18	LIMIT	90.49	0.061	14
	4	186.41	LIMIT	135.37	0.074	16
	5	185.97	LIMIT	83.83	0.104	9
	6	184.79	LIMIT	362.31	0.128	22

Appendix B

Supplement to Chapter 4

The results of the 288 instances of Section ?? can be found in the following subsections, each named after the original instances, where the rest are generated from.

B.0.1 C1 Instances

The results of the instances in Section 4.4.2 can be found in Tables B.1-B.4.

TABLE B.1: Results of C1 with 5 demand points

μ	f_i	$v^\star$	t_C	t_{2PDIS}	η_{2PDIS}	t_{2PAGG}	η_{2PAGG}	t_{DIS}	η_{DIS}	t_{AGG}	η_{AGG}
1	1	10	0.08	0.02	3	0.05	10	0.01	2	0.02	2
1	5	30	0.11	0.02	3	0.02	4	0.01	2	0.02	2
1	10	55	0.11	0.02	3	0.02	4	0.01	2	0.02	2
1	15	80	0.11	0.02	3	0.02	3	0.01	2	0.02	2
1	20	105	0.11	0.02	3	0.02	3	0.02	2	0.02	2
1	25	130	0.11	0.02	3	0.02	3	0.01	2	0.02	2
2	1	9.83	0.16	0.06	3	0.14	13	0.04	2	0.06	2
2	5	22.06	0.42	0.12	3	0.1	6	0.1	2	0.12	2
2	10	37.06	0.34	0.22	3	0.3	4	0.11	2	0.22	2
2	15	52.06	0.27	0.24	3	0.26	4	0.16	2	0.24	2
2	20	67.06	0.3	0.25	3	0.31	4	0.2	2	0.25	2
2	25	82.06	0.29	0.33	3	0.32	3	0.14	2	0.33	2
3	1	9.83	0.13	0.08	3	0.16	17	0.07	2	0.08	2
3	5	20.12	0.38	0.09	3	0.15	8	0.05	2	0.09	2
3	10	30.12	0.34	0.07	3	0.11	5	0.05	2	0.07	2
3	15	40.12	0.39	0.26	3	0.11	4	0.07	2	0.08	2
3	20	50.12	0.41	0.17	3	0.19	4	0.17	2	0.17	2
3	25	60.12	0.45	0.33	3	0.38	4	0.15	2	0.33	2
4	1	9.83	0.15	0.08	3	0.18	19	0.08	2	0.08	2
4	5	20.12	0.55	0.14	3	0.21	9	0.09	2	0.14	2
4	10	30.12	0.51	0.18	3	0.26	6	0.13	2	0.18	2
4	15	40.12	0.57	0.26	3	0.32	5	0.22	2	0.26	2
4	20	50.12	0.58	0.44	3	0.31	4	0.27	2	0.44	2
4	25	60.12	0.78	0.34	3	0.33	4	0.24	2	0.34	2

Table B.2: Results of C1 with 10 demand points

μ	f_i	$v^\star$	t_C	t_{2PDIS}	η_{2PDIS}	t_{2PAGG}	η_{2PAGG}	t_{DIS}	η_{DIS}	t_{AGG}	η_{AGG}
3	1	17	0.3	0.11	3	0.19	16	0.09	2	0.1	2
3	5	35.05	5.55	0.14	3	0.22	8	0.14	2	0.13	2
3	10	55.05	1.12	0.14	3	0.2	4	0.13	2	0.13	2
3	15	75.05	1.08	0.14	3	0.19	4	0.24	2	0.28	3
3	20	95.05	1.03	0.13	3	0.2	4	0.36	3	0.36	2
3	25	115.05	0.99	0.14	3	0.18	4	0.13	2	0.13	2
5	1	17	0.29	0.16	3	0.23	17	0.1	3	0.1	2
5	5	32.46	1.09	0.2	3	0.33	12	0.18	2	0.17	2
5	10	47.46	1.74	0.35	3	0.42	7	0.32	2	0.32	2
5	15	57.73	1.29	0.28	3	0.37	6	0.25	2	0.27	2
5	20	67.73	1.31	0.28	3	0.3	5	0.24	2	0.26	2
5	25	82.06	1.2	0.31	3	0.29	5	0.23	2	0.23	2
7	1	17	0.27	0.13	3	0.25	21	0.1	3	0.11	2
7	5	32.46	1.01	0.37	3	0.23	14	0.52	2	0.44	3
7	10	45.67	1.41	0.45	3	0.92	8	0.3	2	0.29	2
7	15	55.67	1.66	0.52	3	0.52	7	0.71	2	0.73	2
7	20	65.67	1.51	0.41	3	0.81	6	0.32	3	0.32	2
7	25	75.67	1.38	0.36	3	1.13	5	0.36	2	0.37	2
9	1	17	0.27	0.14	3	0.23	21	0.11	3	0.11	2
9	5	32.46	0.97	0.46	3	0.23	14	1.16	2	1.18	6
9	10	45.67	1.49	0.81	3	1.11	10	0.28	2	0.28	2
9	15	55.67	1.83	0.39	3	0.44	8	0.28	2	0.29	2
9	20	65.67	1.73	0.42	3	0.45	6	0.88	2	0.92	4
9	25	75.67	1.66	0.34	3	0.43	6	0.27	2	0.27	2

Table B.3: Results of C1 with 15 demand points

μ	f_i	$v^\star$	t_C	t_{2PDIS}	η_{2PDIS}	t_{2PAGG}	η_{2PAGG}	t_{DIS}	η_{DIS}	t_{AGG}	η_{AGG}
4	1	24.41	0.53	0.16	3	0.71	17	0.05	3	0.05	2
4	5	47.07	1.98	0.48	3	1.1	8	0.26	2	0.27	2
4	10	67.07	2.45	0.74	3	0.27	6	0.35	2	0.35	2
4	15	87.07	2.32	0.32	3	1.59	5	1.08	2	1.11	2
4	20	107.07	2.04	0.33	3	5.75	5	3.83	2	3.91	2
4	25	127.07	2.2	0.31	3	14.81	4	12.28	2	12.29	2
7	1	24.41	0.54	0.19	3	0.34	20	0.07	2	0.08	2
7	5	46.12	2.04	0.28	3	0.36	12	0.14	2	0.15	2
7	10	61.12	1.43	0.27	3	0.8	8	0.16	2	0.17	2
7	15	76.12	2.19	0.26	3	2.24	7	0.23	2	0.25	2
7	20	91.12	1.91	0.21	3	1.12	6	0.22	2	0.22	2
7	25	106.12	1.8	0.21	3	0.29	5	0.19	2	0.2	2
10	1	24.41	0.75	0.1	3	0.35	22	0.07	2	0.07	2
10	5	46.12	1.89	0.19	3	0.4	14	0.16	2	0.16	2
10	10	61.12	2.64	0.28	3	0.72	11	0.94	2	0.35	3
10	15	72.85	3.53	0.36	3	0.78	8	0.28	3	0.29	2
10	20	82.85	3.22	0.56	3	0.87	7	0.49	2	0.49	2
10	25	92.85	3.09	0.77	3	0.88	6	0.73	2	0.76	2
13	1	24.41	0.54	0.1	3	0.93	22	0.06	2	0.07	2
13	5	46.12	1.99	0.26	3	0.8	17	0.18	2	0.19	2
13	10	61.12	2.53	1.9	3	0.94	13	0.34	2	0.36	2
13	15	72.85	3.37	2.41	3	0.75	9	0.76	2	0.81	2
13	20	82.85	4.94	1.27	3	1.02	8	1.23	2	1.3	2
13	25	92.85	8.51	1.97	3	2	7	1.81	2	1.8	2

Table B.4: Results of C1 with 20 demand points

μ	f_i	$v^\star$	t_C	t_{2PDIS}	η_{2PDIS}	t_{2PAGG}	η_{2PAGG}	t_{DIS}	η_{DIS}	t_{AGG}	η_{AGG}
6	1	30.41	1.01	0.1	3	0.38	18	0.07	2	0.07	2
6	5	57.94	2.5	0.14	3	0.81	5	0.11	2	0.12	2
6	10	77.94	1.72	0.3	3	0.53	7	0.26	2	0.27	2
6	15	97.94	2.65	2.16	3	2.89	6	1.94	2	1.96	2
6	20	117.94	2.46	0.55	3	26.75	5	0.76	2	0.76	2
6	25	137.94	2.19	0.61	3	1.42	5	0.93	2	0.92	2
10	1	30.41	0.97	0.09	3	0.89	22	0.07	3	0.07	2
10	5	57.94	2.24	0.16	3	0.43	14	0.99	2	0.27	3
10	10	76.41	3.67	0.47	3	0.45	11	0.32	3	0.3	2
10	15	90.73	4.17	1.17	3	3.07	8	0.97	2	0.97	2
10	20	100.73	5.27	0.69	3	0.7	7	0.72	2	0.71	2
10	25	110.73	5.33	0.57	3	0.83	6	0.5	2	0.51	2
14	1	30.41	1.19	0.1	3	0.47	22	0.07	2	0.08	2
14	5	57.94	1.98	0.16	3	0.51	17	0.14	2	0.14	2
14	10	76.41	3.86	0.65	3	0.51	13	0.86	2	0.9	2
14	15	89.62	3.52	1.31	3	0.81	9	1.08	2	1.09	2
14	20	99.62	5.84	1.02	3	0.83	8	1.06	2	1.08	2
14	25	109.62	4.98	1	3	1.16	8	1.16	2	1.2	2
18	1	30.41	1.47	0.11	3	0.47	22	0.06	2	0.07	2
18	5	57.94	2.03	0.19	3	0.55	18	0.15	2	0.16	2
18	10	76.41	3.51	1.16	3	0.73	14	0.88	2	0.89	2
18	15	89.62	5.37	2.46	3	0.85	12	1.41	2	1.42	2
18	20	99.62	6.17	2.17	3	1.14	9	1.79	2	1.79	2
18	25	109.62	9.33	1.89	3	1.43	9	2.44	2	2.54	2

B.0.2 C2 Instances

The results of the instances in Section 4.4.3 can be found in Tables B.5-B.8.

Table B.5: Results of C2 with 18 demand points

μ	f_i	$v^\star$	t_C	t_{2PDIS}	η_{2PDIS}	AVR. 2PDIS t_{MLP}	2PDIS t_{MMILP}	t_{2PAGG}	η_{2PAGG}	AVR. 2PAGG t_{MLP}	2PAGG t_{MMILP}
5	1	27	205.07	123.59	42	0.343	93.44	108.9	42	0.591	57.86
5	5	66.56	452.96	55.05	16	0.194	46.02	157.82	16	0.596	144.46
5	10	92.64	1072.5	74.9	11	0.178	68.84	155.64	11	0.372	147.56
5	15	114.8	1767.49	86.05	9	0.169	81.21	84.09	9	0.476	77.93
5	20	134.8	2371.95	96.82	8	0.138	92.73	56.02	8	0.469	51.17
5	25	154.8	2956.98	140.65	7	0.134	137.12	204.88	7	0.085	200.68
9	1	27	198.83	191.56	56	0.389	35.55	178.57	66	1.089	63.88
9	5	66.56	463.18	93.51	28	0.328	73.82	367.84	27	0.608	341.72
9	10	90.22	632.93	85.97	17	0.232	75.66	277.65	17	0.396	263.54
9	15	110.22	1319.92	93	13	0.182	85.82	129.71	13	0.500	120.08
9	20	126.38	2021.56	155.39	11	0.201	149.1	149.31	11	0.333	141.8
9	25	141.38	2929.32	240.03	10	0.184	234.49	260.6	10	0.250	253.66
12	1	27	277.51	294.59	58	0.378	60.5	158.92	69	1.276	33.09
12	5	66.56	588.77	111.62	39	0.323	84.7	419.36	39	0.954	371.35
12	10	90.22	615.58	95.91	22	0.238	82.62	403.04	22	0.432	382.69
12	15	110.22	1481.79	103.46	16	0.203	94.38	527.04	16	0.195	514.63
12	20	126.38	2562.79	215.37	13	0.183	208.27	496.69	13	0.041	487.26
12	25	141.38	9991.66	145.97	11	0.172	140.07	240.28	11	0.346	232.49
16	1	27	312.3	502.63	62	0.411	62.27	210.31	81	1.425	44.58
16	5	66.56	474.18	144.73	44	0.356	112.91	321.77	42	1.079	268.52
16	10	90.22	888.73	137.65	26	0.260	121.35	310.78	26	0.670	285.33
16	15	110.22	1540.52	189.47	21	0.228	177	274.34	21	0.541	255.84
16	20	126.38	1856.61	123.53	16	0.211	114.34	229.92	16	0.442	216.85
16	25	140.68	3732.34	397.28	14	0.190	389.47	541.7	14	0.070	531.04

Table B.6: Results of C2 with 36 demand points

μ	f_i	$v^\star$	t_C	t_{2PDIS}	η_{2PDIS}	AVR. 2PDIS t_{MLP}	2PDIS t_{MMILP}	t_{2PAGG}	η_{2PAGG}	AVR. 2PAGG t_{MLP}	2PAGG t_{MMILP}
10	1	47.27	161.72	298.92	106	1.427	62.47	200.75	54	2.082	41.03
10	5	109.79	1369.68	179.21	42	0.831	110.79	123.77	23	1.694	83.03
10	10	145.53	3135.76	183.75	27	0.552	147.55	128.7	16	1.306	105.51
10	15	170.53	LIMIT	158.24	21	0.421	132.81	93.59	13	1.250	75.97
10	20	191.91	LIMIT	166.66	17	0.445	145.78	99.71	11	1.187	85.05
10	25	211.91	LIMIT	120.3	15	0.415	102.21	83.6	10	1.323	70.19
18	1	47.27	204.02	466.87	121	2.097	114.34	435.13	85	3.430	40
18	5	109.79	1780.9	445.08	73	1.265	292.88	400.32	51	3.042	236.99
18	10	145.53	1742.06	527.24	42	0.860	456.68	238.4	29	2.059	174.06
18	15	170.53	9918.98	425.65	32	0.693	377.2	444.27	23	1.389	401.27
18	20	191.91	4574.47	357.6	26	0.578	321.13	210.64	18	1.591	177.7
18	25	211.91	6627.02	341.61	23	0.537	310.24	557.85	16	1.095	531.38
25	1	47.27	194.9	432.97	130	1.841	90.86	564.9	102	3.747	47.44
25	5	109.79	1095.78	544.36	81	2.893	378.92	350.83	57	3.251	158.94
25	10	145.53	3216.78	770.11	63	1.021	655.79	397.87	45	2.655	269.89
25	15	170.53	7515.58	661.16	40	0.718	601	216.87	29	1.903	157.51
25	20	191.91	6002.12	497.73	34	0.628	449.42	283.53	25	1.653	235.92
25	25	211.91	7876.62	608.92	28	0.558	571.01	272.81	54	0.541	239.21
32	1	47.27	189.68	1241.24	210	3.555	298.99	552.22	103	3.607	48.48
32	5	109.79	1177.94	734.73	97	1.969	464.85	434.62	68	3.639	178.63
32	10	145.53	2081.5	1227.62	66	1.338	1085.78	9121.45	48	0.834	8963.69
32	15	170.53	6232.02	861.8	54	1.186	753.95	322.82	42	2.517	210.59
32	20	191.91	6168.53	716.69	39	0.903	649.81	269.93	29	1.899	209.2
32	25	211.91	5307.91	735.02	35	0.662	684.23	1565.87	25	0.826	1517.87

Table B.7: Results of C2 with 54 demand points

μ	f_i	$v^\star$	t_C	t_{2PDIS}	η_{2PDIS}	AVR. 2PDIS t_{MLP}	2PDIS t_{MMILP}	t_{2PAGG}	η_{2PAGG}	AVR. 2PAGG t_{MLP}	2PAGG t_{MMILP}
16	1	67.93	328.29	551.71	118	3.003	69.59	554.36	76	4.757	61.2
16	5	145.47	3962.1	445.33	66	1.848	252.03	412.07	49	4.464	186.24
16	10	195.76	7594.05	428.71	39	1.245	336.49	318.53	28	3.591	212.66
16	15	231.09	LIMIT	592.38	31	1.046	525.44	357.54	23	2.870	284.38
16	20	259.53	LIMIT	471.74	24	0.845	425.38	618.97	17	2.006	569.54
16	25	279.71	LIMIT	744.18	20	0.773	706.32	900.28	15	1.555	862.3
27	1	67.93	232.13	600.16	130	2.596	123.98	860.28	99	5.857	74.65
27	5	145.47	1731.32	740.81	84	1.786	501.2	558.27	58	5.056	254.42
27	10	195.76	2067.92	822.43	63	1.422	665.64	539.33	47	4.288	326.96
27	15	231.09	10135.5	1371.17	42	1.034	1282.8	486.68	31	3.159	378.27
27	20	259.53	LIMIT	873.92	35	0.912	804.7	1268.5	25	1.707	1192.01
27	25	279.71	LIMIT	734.68	31	0.817	676.17	761.58	23	2.358	696
37	1	67.93	377.98	1049.57	167	4.156	17.62	898.89	104	5.909	69.17
37	5	145.47	2386.8	1112.03	100	2.612	742.77	686.42	69	5.721	278.15
37	10	195.76	3055.07	1402.61	73	2.007	1177.33	657.18	53	4.678	395.51
37	15	231.09	10588.8	2454.94	63	1.762	2275.81	1761.32	46	3.622	1549.01
37	20	259.53	LIMIT	1360.31	43	1.282	1258.65	2272.63	31	1.701	2157.98
37	25	279.71	LIMIT	1039.36	37	1.165	955.98	859.01	27	2.870	768.94
48	1	67.93	385.39	1350.09	221	4.289	166.43	933.5	105	6.038	75.9
48	5	145.47	1815.49	1474.44	125	2.675	1007.11	889.57	78	6.709	348.1
48	10	195.76	2759.46	1513.9	80	1.832	1282.2	697.57	56	5.033	401.22
48	15	231.09	LIMIT	2229.6	66	1.558	2056.58	859.49	49	4.450	621.82
48	20	259.53	9184.18	2678.03	57	1.381	2538.44	1760.66	42	3.134	1582.88
48	25	279.71	LIMIT	1267.61	43	1.107	1173.98	1426.21	31	2.746	1318.15

Table B.8: Results of C2 with 72 demand points

μ	f_i	$v^\star$	t_C	t_{2PDIS}	η_{2PDIS}	AVR. 2PDIS t_{MLP}	2PDIS t_{MMILP}	t_{2PAGG}	η_{2PAGG}	AVR. 2PAGG t_{MLP}	2PAGG t_{MMILP}
21	1	82.93	415.7	795.27	130	3.680	133.42	949.62	79	7.373	140.93
21	5	171.7	2297.61	623.34	75	2.288	345.82	714.73	53	7.329	313.63
21	10	235.59	9431.68	705.58	55	1.795	529.24	734.02	40	5.859	484.97
21	15	282.68	10882.7	895.45	36	1.292	798	712.1	26	3.939	593.69
21	20	316.54	LIMIT	730.68	31	1.156	650.99	773.75	23	3.601	672.87
21	25	346.54	LIMIT	573.4	24	0.961	516.31	706.73	18	3.568	633.36
36	1	82.93	367.67	1035.39	139	4.222	252.7	1425.9	96	9.125	208.29
36	5	171.7	1833.93	1212.98	96	3.037	786.26	1144.77	65	10.099	466.35
36	10	235.59	4555.45	1335.71	73	2.390	1058.37	917.81	53	7.119	521.9
36	15	282.66	LIMIT	2082.3	57	1.944	1891.2	1947.08	42	5.529	1665.87
36	20	316.54	LIMIT	1240.71	43	1.550	1113.28	1311.39	30	3.984	1159.13
36	25	346.54	LIMIT	1444.62	37	1.375	1341.49	1306.28	27	3.924	1180.75
50	1	82.93	363.11	1290.62	168	5.115	194.64	1340.36	107	8.342	126.43
50	5	171.7	1841.72	1576.97	177	3.746	973.99	1025.01	78	8.992	303.9
50	10	235.59	3498.99	1807.44	81	2.721	1473.06	901.22	57	7.163	474.87
50	15	282.66	LIMIT	2201.55	67	2.296	1953.26	LIMIT	LIMIT	LIMIT	LIMIT
50	20	316.54	LIMIT	2403.08	63	2.193	2176.17	LIMIT	LIMIT	LIMIT	LIMIT
50	25	346.54	LIMIT	1853.68	47	1.699	1707.43	LIMIT	LIMIT	LIMIT	LIMIT
64	1	82.93	641.1	2158.59	226	6.716	322.44	1410.38	110	8.250	164.17
64	5	171.7	1309.47	2352.54	157	4.931	1357.18	1181.67	85	9.333	364.57
64	10	235.59	3010.12	2366.79	97	3.290	1911.57	1144.34	67	7.843	594.63
64	15	282.66	LIMIT	2893.54	76	2.634	2586.19	LIMIT	LIMIT	LIMIT	LIMIT
64	20	316.54	LIMIT	2784.35	66	2.318	2538.09	LIMIT	LIMIT	LIMIT	LIMIT
64	25	346.54	LIMIT	2743.05	63	2.237	2513.39	LIMIT	LIMIT	LIMIT	LIMIT

B.0.3 C3 Instances

The results of the instances in Section 4.4.4 can be found in Tables B.9-B.12.

TABLE B.9: Results of C3 with 10 demand points

μ	f_i	$v^\star$	t_{2PDIS}	η_{2PDIS}
3	1	15	31.2	44
3	5	LIMIT	LIMIT	LIMIT
3	10	LIMIT	LIMIT	LIMIT
3	15	LIMIT	LIMIT	LIMIT
3	20	LIMIT	LIMIT	LIMIT
3	25	LIMIT	LIMIT	LIMIT
5	1	15	130.71	58
5	5	LIMIT	LIMIT	LIMIT
5	10	LIMIT	LIMIT	LIMIT
5	15	LIMIT	LIMIT	LIMIT
5	20	LIMIT	LIMIT	LIMIT
5	25	LIMIT	LIMIT	LIMIT
7	1	15	222.87	64
7	5	LIMIT	LIMIT	LIMIT
7	10	LIMIT	LIMIT	LIMIT
7	15	LIMIT	LIMIT	LIMIT
7	20	LIMIT	LIMIT	LIMIT
7	25	LIMIT	LIMIT	LIMIT
9	1	15	82.18	109
9	5	LIMIT	LIMIT	LIMIT
9	10	LIMIT	LIMIT	LIMIT
9	15	LIMIT	LIMIT	LIMIT
9	20	LIMIT	LIMIT	LIMIT
9	25	LIMIT	LIMIT	LIMIT

Table B.10: Results of C3 with 10 demand points

μ	f_i	$v^\star$	t_{2PDIS}	η_{2PDIS}
6	1	30	185.49	105
6	5	LIMIT	LIMIT	LIMIT
6	10	LIMIT	LIMIT	LIMIT
6	15	LIMIT	LIMIT	LIMIT
6	20	LIMIT	LIMIT	LIMIT
6	25	LIMIT	LIMIT	LIMIT
10	1	30	259.81	130
10	5	LIMIT	LIMIT	LIMIT
10	10	LIMIT	LIMIT	LIMIT
10	15	LIMIT	LIMIT	LIMIT
10	20	LIMIT	LIMIT	LIMIT
10	25	LIMIT	LIMIT	LIMIT
14	1	30	373.83	168
14	5	LIMIT	LIMIT	LIMIT
14	10	LIMIT	LIMIT	LIMIT
14	15	LIMIT	LIMIT	LIMIT
14	20	LIMIT	LIMIT	LIMIT
14	25	LIMIT	LIMIT	LIMIT
18	1	30	468.47	184
18	5	LIMIT	LIMIT	LIMIT
18	10	LIMIT	LIMIT	LIMIT
18	15	LIMIT	LIMIT	LIMIT
18	20	LIMIT	LIMIT	LIMIT
18	25	LIMIT	LIMIT	LIMIT

Table B.11: Results of C3 with 30 demand points

μ	f_i	$v^\star$	t_{2PDIS}	η_{2PDIS}
9	1	43.83	305.83	132
9	5	96.85	164.48	43
9	10	126.7	2826.26	25
9	15	146.7	990.63	18
9	20	166.7	1171.91	14
9	25	LIMIT	LIMIT	LIMIT
15	1	43.82	506.65	162
15	5	96.85	313.55	56
15	10	126.7	2366.78	33
15	15	146.7	4580.02	25
15	20	163.86	746.6	20
15	25	178.86	14758.5	17
21	1	43.82	648.81	190
21	5	96.85	462.11	82
21	10	126.701	3157.11	50
21	15	146.7	3668.6	34
21	20	163.86	3794.22	28
21	25	178.86	4841.52	23
27	1	43.82	804.81	200
27	5	96.85	659.17	96
27	10	126.7	6087.02	59
27	15	146.7	3519.31	43
27	20	163.86	17862	34
27	25	LIMIT	LIMIT	LIMIT

Table B.12: Results of C3 with 40 demand points

μ	f_i	$v^\star$	t_{2PDIS}	η_{2PDIS}
12	1	57.89	576.87	154
12	5	123.22	275.33	56
12	10	160.91	519.66	34
12	15	184.78	1339.74	24
12	20	204.78	1194.49	20
12	25	224.78	604.14	16
20	1	57.89	975.29	209
20	5	123.22	519.63	78
20	10	160.91	802.13	45
20	15	184.78	2339.61	33
20	20	204.78	9860.45	26
20	25	224.78	16451.8	22
28	1	57.89	1229.55	239
28	5	123.22	808.17	112
28	10	160.91	876.22	68
28	15	184.78	3225.53	49
28	20	204.78	1568.29	39
28	25	224.78	9053.56	32
36	1	57.89	1413.86	248
36	5	123.22	1119.18	133
36	10	160.91	1198.9	88
36	15	184.78	2345.45	56
36	20	204.78	719.27	48
36	25	LIMIT	LIMIT	LIMIT

Bibliography

Akyüz MH, Öncan T, Altinel IK (2010) The multi-commodity capacitated multi-facility Weber problem: heuristics and confidence intervals. *IIE Transactions* 42(11):825–841.

Alfieri L, Salamon P, Bianchi A, Neal J, Bates P, Feyen L (2014) Advances in pan-European flood hazard mapping. *Hydrological Processes* 28(1):4067–4077.

Amiri-Aref M, Javadian N, Tavakkoli-Moghaddam R, Aryanezhad MB (2011) The center location problem with equal weights in the presence of a probabilistic line barrier. *International Journal of Industrial Engineering Computations* 2(1):793–800.

Amiri-Aref M, Javadian N, Tavakkoli-Moghaddam R, Baboli A (2013) A new mathematical model for the Weber location problem with a probabilistic polyhedral barrier. *International Journal of Production Research* 51(20):6110–6128.

Aneja YP, Parlar M (1994) Technical notealgorithms for Weber facility location in the presence of forbidden regions and/or barriers to travel. *Transportation Science* 28(1):70–76.

Arabani AB, Farahani RZ (2012) Facility location dynamics: An overview of classifications and applications. *Computers & Operations Research* 62(1):408–420.

Balakrishnan A, Magnanti TL, Wong RT (1987) A dual-ascent procedure large-shape uncapacitated network design. Working paper, OR 161-87, Massachusetts Institute of Technology, Cambridge.

Barbarosoglu G, Arda Y (2004) A two-stage stochastic programming framework for transportation planning in disaster response. *Journal of the Operational Research Society* 55(1):43–53.

Batta R, Ghose A, Palekar US (1989) Locating facilities on the manhattan metric with arbitrarily shaped barriers and convex forbidden regions. *Transportation Science* 23(1):26–36.

Bazaraa MS, Jarvis JJ, Sherali HD (1990) *Linear programming and network flows.* John Wiley & Sons, New York.

Benders J (1962) Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik* 4(3):238–252.

Bennell JA, Scheithauer G, Stoyan Y, Romanova T (2010) Tools of mathematical modeling of arbitrary object packing problems. *Annals of Operations Research* 179(1):343–368.

Bennell JA, Scheithauer G, Stoyan Y, Romanova T, Pankratov A (2015) Optimal clustering of a pair of irregular objects. *Journal of Global Optimization* 497–524.

Bischoff M, Klamroth K (2007) An efficient solution method for Weber problems with barriers based on genetic algorithms. *European Journal of Operational Research* 177(1):22–41.

Bischoff M, Fleischmann T, Klamroth K (2009) The multi-facility location-allocation problem with polyhedral barriers. *Computers & Operations Research* 36(1):1376–1392.

Butt SE, Cavalier TM (1996) An efficient algorithm for facility location in the presence of forbidden regions. *European Journal of Operational Research* 90(1):56–70.

Butt SE, Cavalier TM (1997) Facility location in the presence of congested regions with the rectilinear distance metric. *Socio-Economic Planning Sciences* 31(2):103–113.

Canbolat MS, Wesolowsky GO (2010) The rectilinear distance Weber problem in the presence of a probabilistic line barrier. *European Journal of Operational Research* 202(1):114–121.

Carrizosa E, Plastria F (1993) Planar minquantile and maxcovering location problems. *Studies in Locational Analysis* 4:29–33.

Chernov N, Stoyan Y, Romanova T, Pankratov A (2012) Phi-functions for 2D objects formed by line segments and circular arcs. *Advances in Operations Research* 1–26.

Cordeau JF, Stojkovic G, Soumis F, Desrosiers J (2001) Benders' decomposition for simultaneous aircraft routing and crew scheduling. *Transportation Science* 35(4):375–388.

Cumbria County Council (2009a) Floods explanation. ⟨http://www.cumbria.gov.uk/floods/explanationandtimeline/explanationandtimeline.asp⟩ (accessed 01.06.2015).

Cumbria County Council. (2009b), Impact on transport infrastructure. ⟨http://www.cumbria.gov.uk/floods/damageanalysis/transportinfrastructure.asp⟩ (accessed 01.06.2015).

Dasci A, Verter V (2001) A continuous model for production-distribution system design. *European Journal of Operational Research* 129(2):287–298.

Dearing PM, Klamroth K, Segars Jr. R (2005) Planar location problems with block distance and barriers. *Annals of Operations Research* 136(1):117–143.

Dearing PM, Segars Jr. R (2002a) An equivalence result for single facility planar location problems with rectilinear distance and barriers. *Annals of Operations Research* 111(1-4):89–110.

Dearing PM, Segars Jr. R (2002b) Solving rectilinear planar location problems with barriers by a polynomial partitioning. *Annals of Operations Research* 111(1-4):111–133.

Dearing PM, Hamacher HW, Klamroth K (2002) Dominating sets for rectilinear center location problems with polyhedral barriers. *Naval Research Logistics* 49(7):647–665.

Dijkstra EW (1959) A note on two problems in connexion with graphs. *Numerische Mathematik* 1:269–271.

Drezner Z (1984) The p-centre problem-heuristic and optimal algorithms. *Journal of Operational Research Society* 35(8):741–748.

Drezner Z (1988) Location strategies for satellites' orbits. *Naval Research Logistics* 35:503–512.

Drezner Z (1995) *Facility location: a survey of applications and methods.* Springer, Verlag, New York.

Farahani RZ, SteadieSeifi M, Asgari N (2010) Multiple criteria facility location problems: A survey. *Applied Mathematical Modelling* 6(6):1689–1709.

Foulds LR, Hamacher HW (1993) Optimal bin location and sequencing in printed circuit board Assembly . *European Journal of Operational Research* 66:279–290.

Gabrel V, Knippel A, Minoux M (1999) Exact solution of multicommodity network optimization problems with general step functions. *Operations Research Letters* 25(1):15–23.

Gallucci M (2015a) Texas And Oklahoma Floods 2015: How Global Warming Makes The Flooding More Destructive. *International Business Times* 26 May. Available from: ⟨http://www.ibtimes.com/texas-oklahoma-floods-2015-how-global-warming-makes-flooding-more-destructive-1938597⟩ (accessed 02.06.2015)

Gallucci M (2015b) Texas Flooding 2015: Severe Storms In Texas Cause At Least $27M In Infrastructure Damage, Continuing Trend Of Increasing Natural Disasters. *International Business Times* 31 May. Available from: ⟨http://www.ibtimes.com/texas-flooding-2015-severe-storms-texas-cause-least-27m-infrastructure-damage-1945594⟩ (accessed 31.05.2015)

Hakimi S (1965) Optimum location of switching centers in a communications network and some related graph theoretic problems. *Operations Research* 13:462–475.

Hamacher HW, Klamroth K (2000) Planar Weber location problems with barriers and block norms. *Annals of Operations Research* 96(1-4):191–208.

Hamacher HW, Nickel S (1994) Combinatorial algorithms for some 1-facility median problems in the plane. *European Journal of Operational Research* 79(2):340–351.

Hamacher HW, Nickel S (1995) Restricted planar location problems and applications. *Naval Research Logistics* 42(6):967–992.

Hamacher HW, Schöbel A (1997) A note on center problems with forbidden polyhedra. *Operations Research Letters* 20(4):165–169.

Handler GY, Mirchandani PB (1979) *Location on Networks: Theory and Algorithms.* The MIT Press, Cambridge, Massachusetts.

Hewitt M, Nemhauser GL, Savelsbergh MWP (2010) Combining exact and heuristic approaches for the capacited fixed-charge network problem. *INFORMS Journal on Computing* 22(2):314–325.

Hodgson JM, Wong RT, Honsaker J (1987) The p-centroid problem on an inclined plane. *Operations Research* 35:221–233.

Hogan K (1990) Reducing errors in rainfall estimates through rain gauge location. *Geographical Analysis* 22:33–49.

Javadian N, Tavakkoli-Moghaddam R, Amiri-Aref M, Shiripour S (2014) Two meta-heuristics for a multi-period minisum location-relocation problem with line restriction. *The International Journal of Advanced Manufacturing Technology* 71(1):1033–1048.

Katz IN, Cooper L (1981) Facility location in the presence of forbidden regions, i: Formulation and the case of Euclidean distance with one forbidden circle. *European Journal of Operational Research* 6(2):166–173.

Klamroth K (2001) A reduction result for location problems with polyhedral barriers. *European Journal of Operational Research* 130(3):486–497.

Klamroth K (2004) Algebraic properties of location problems with one circular barrier. *European Journal of Operational Research* 154(1):20–35.

Larson RC, Sadiq G (1983) Facility locations with the manhattan metric in the presence of barriers to travel. *Operations Research* 31(4):652–669.

Li YP, Huang GH, Nie SL (2006) An interval-parameter multi-stage stochastic programming model for water resources management under uncertainty. *Advances in Water Resources* 29(5):776–789.

Liu C, Fan Y, Ordonez F (2009) A two-stage stochastic programming model for transportation network protection. *Computers & Operations Research* 36(5):1582–1590.

Maqsood I, Huang GH, Yeomans JS (2005) An interval-parameter fuzzy two-stage stochastic program for water resources management under uncertainty. *European Journal of Operational Research* 167(1):208–225.

Magnati TL, Wong RT (1981) Accelerating Benders decomposition: algorithmic enhancement and model selection criteria. *Operations Research* 29(3):464–484.

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

McGarvey RG, Cavalier TM (2003) A global optimal approach to facility location in the presence of forbidden regions. *Computers & Industrial Engineering* 45(1):1–15.

Milly PCD, Wetherald RT, Dunne KA, Delworth TL (2002) Increasing risk of great floods in a changing climate. *Nature* 415:514–517.

Minoux M (2001) Discrete cost multicommodity network optimization problems and exact solution methods. *Annals of Operations Research* 106(1):19–46.

Moradi S, Raith A, Ehrgott M (2015) A bi-objective column generation algorithm for the multi-commodity minimum cost flow problem. *European Journal of Operational Research* 244(2):369–378.

Noyan N (2008) Risk-averse two-stage stochastic programming with an application to disaster management. *Computers & Operations Research* 39(3):541–559.

Oğuz M, Bektaş T, Bennell JA, Fliege J (2016) A modeling framework for solving restricted planar location problems using phi-objects. *Journal of the Operational Research Society* 67(8):1080–1096.

Papadakos N (2008) Practical enhancements to the Magnanti–Wong method. *Operations Research Letters* 444–449.

Papadakos N (2009) Integrated airline scheduling. *Computers & Operations Research* 36(1):176–195.

Paraskevopoulos DC, Bektaş T, Crainic TG, Potts CN (2016) A cycle based evolutionary algorithm for the fixed-charge capacitated multi-commodity network design problem. *European Journal of Operational Research* 253(2):265–279.

Prinos P, Kortenhaus A, Swerpel B, Jimenez J (2009) Review on flood hazard mapping. *FLOODsite report* Available from: ⟨http://www.floodsite.net/html/partner_area/project_docs/t03_07_01_review_hazard_mapping_v4_3_p01.pdf⟩ (accessed 05.12.2016).

ReVelle CS, Eiselt HA, Daskin MS (2008) A bibliography for some fundamental problem categories in discrete location science. *European Journal of Operational Research* 184:817–848.

Saharidis GKD, Minoux M, Ierapetritou MG (2010) Accelerating Benders method using covering cut bundle generation. *International Transactions in Operations Research* 17(2):221–237.

Sarkar A, Batta R, Nagi R (2004) Commentary on facility location in the presence of congested regions with the rectilinear distance metric. *Socio-Economic Planning Sciences* 38:291–306.

Schouwenaars T, Moor BD, Feron E, How J (2001) Mixed integer programming for multi-vehicle path planning. In: *European Control Conference*, Porto, 4-7 September 2001. 2603-2608.

Schumacher C, Chandler PR, Rasmussen SR (2002) Task allocation for wide area search munitions via iterative network flow. In: *American Control Conference*, Anchorage, 8-10 May 2002. pp. 1917–1922.

Shiripour S, Mahdavi I, Amiri-Aref M, Mohammadnia-Otaghsara M, Mahdavi-Amiri N (2012) Multi-facility location problem in the presence of a probabilistic line barrier: a mixed integer quadratic programming model. *International Journal of Production Research* 50(15):3988–4008.

Tang L, Jiang W, Saharidis GKD (2013) An improved Benders decomposition algorithm for the logistics facility location problem with capacity expansions. *Annals of Operations Research* 210(1):165–190.

Toothill J (2002) Central European Flooding August 2002. Technical Report. EQECAT, London.

Toregas C, Swain R, ReVelle C, Bergman L (1971) The location of emergency service facilities. *Operations Research* 19:1363–1373.

Tsiakis P, Shah N, Pantelides CC (2001) Design of multi-echelon supply chain networks under demand uncertainty. *Industrial & Engineering Chemistry Research* 40(1):3585–3604.

Ulbrich U, Brücher T, Fink AH, Leckebusch GC, Krüger A, Pinto JG (2003) The central European floods of August 2002: Part 1 - Rainfall periods and flood development. *Weather* 58:371–377.

Wang QJ (1991) The POT model described by the generalized Pareto distribution with Poisson arrival rate. *Journal of Hydrology* 129:263–280.

Woeginger GJ (1998) A comment on a minmax location problem. *Operations Research Letters* 23(1):41–43.

Zhou Z, Zhang J, Liu P, Li Z, Georgiadis MC (2013) A two-stage stochastic programming model for the optimal design of distributed energy systems. *Applied Energy* 103(1):135–144.