UNIVERSITY OF GRANADA

DOCTORATE PROGRAM IN MATHEMATICS

PhD Dissertation

UNIVERSIDAD
DE GRANADA

ADVANCES AND APPLICATIONS IN CONTINUOUS LOCATION AND RELATED PROBLEMS

RICARDO GÁZQUEZ TORRES

Thesis supervised by

Prof. Dr. Víctor Blanco Izquierdo

Granada 2022

Editor: Universidad de Granada. Tesis Doctorales
Autor: Ricardo Gázquez Torres
ISBN: 978-84-1117-361-2
URI: http://hdl.handle.net/10481/75448

A mi familia y Blanca.
A MW Ogíjares.
A 1M.

Agradecimientos

Desde que comencé a leer tesis en mis trabajos de investigación, siempre me fijé en la sección de agradecimientos. He visto agradecimientos muy diversos, desde puramente agradecimientos destinados a aquellas personas que acompañaron al doctorando hasta breves repasos de la vida académica del mismo. Es por ello, que más allá del interés investigador que contiene una tesis, los agradecimientos es una de las partes que siempre me han llamado la atención.

Desde mucho antes de saber que quería realizar una tesis, siempre me imaginé escribiendo unas palabras a aquellas personas que me han ayudado a llegar hasta aquí. Permitidme, por tanto, extenderme fuera del rigor y la precisión del lenguaje matemático para llegar a aquellas personas que lean estas palabras.

En primer lugar, quiero agradecer a mi director de tesis, Víctor Blanco, su confianza depósitada en mí, su grandísimo trabajo, su conocimiento, su magnífica predisposición a ayudarme, sus constantes ánimos y motivación, su infinita paciencia, sus horas de explicaciones y sus consejos. Me llevo una enorme experiencia y aprendizaje al haber trabajado a su lado y la referencia de cómo un investigador debe desarrollar su trabajo. Gracias por haber estado ahí y gracias por darme la oportunidad de haber trabajado con usted.

Agradezco a mis padres por su constante apoyo, en las buenas y en las no tan buenas, por ser el pilar donde baso mi vida, por su ayuda infinita. Gracias por ser un modelo a seguir, a inculcarme los valores que me han formado y que me han dado la oportunidad de presentar esta tesis. Gracias por estar ahí cuando lo necesito. Gracias por ayudarme a alcanzar mis metas.

A mi hermano, por alegrarme los días no tan buenos, por compartir conmigo esos buenos momentos, por estar ahí siempre, por todo lo que nos das día a día. Gracias por animarme a embarcarme en nuevos proyectos. Gracias por estar ahí.

A Blanca, sin duda esta tesis se apoya en ella. Gracias Blanca por ayudarme en todo lo que has podido y más. Gracias por acompañarme en esta aventura. Gracias por estar ahí y motivarme cada día. Gracias por ser como eres y por compartirlo conmigo. Esta tesis es el resultado de tu ánimo desinteresado hacia a mí.

A MW Ogíjares por su constante apoyo y ánimo. Gracias por darme la oportunidad de compartir esta experiencia con vosotros, por soportar las idas y venidas en los momentos más difíciles y tensos. Gracias por estar ahí. Esta tesis es gracias también a vosotros.

Permitidme hacer un breve repaso a mi vida académica para terminar esta sección de agradecimientos. Empecé la carrera de matemáticas animado por mi padre y enamorado por la satisfacción de resolver problemas. Esa satisfacción me la dieron distintos profesores en mi etapa en el instituto, pero sobre todo a Laura Camacho, gracias por la magnífica labor que hizo en el IES Blas Infante de Ogíjares. Ya en bachillerato, el Dr. Diego Torrecillas me mostró que las matemáticas *escondían* algo más que la resolución de problemas. Ese algo más me fascinó y desde entonces supe que las matemáticas deberían estar siempre conmigo.

Recuerdo que en bachillerato fuimos a una charla en la que nos comentaban el funcionamiento de los distintos grados y estudios que la Universidad de Granada ofrecía. Cómo era el camino para llegar a investigar y resolver esos problemas que tanto me satisfacían. Sabiendo el largo camino que escondía ese objetivo, comencé mi carrera de matemáticas queriendo ser investigador, y tras años de estudios en carreras, másteres y años de doctorado, hoy entrego esta tesis.

Gracias por tanto, a todos los maestros y profesores de matemáticas en toda esta carrera académica. En algún lugar y momento contribuistéis a que eligiera este camino.

Finalmente, quiero agradecer a la Universidad de Granada por permitirme llegar hasta aquí, al programa de doctorado en Matemáticas por darme la oportunidad y por último quiero agradecer la acogida que tuvo el Dpto. de Métodos Cuantitativos para la Economía y Empresa de la Universidad de Granada para poder desarrollar mi investigación. Por último, hacer extensiva este agradecimiento a mis compañeros del departamento, por sus consejos y experiencias que me han ayudado a llegar hasta aquí. Al Centro de Matemática, Aplicações Fundamentais e Investigação Operacional (CMAFcIO), en la facultad de ciencias de la Universidad de Lisboa por darme la posibilidad de realizar la estancia. A todos mis coautores, por sus valiosas lecciones en los problemas que hemos tratado. Agradecer el apoyo del grupo de investigacion SEJ-534, de los proyectos a los que he tenido la suerte de pertenecer, PID2020-114594GB-C21 y B-FQM-322-UGR20, y la beca del ministerio PEJ2018-002962-A.

Resumen

Esta tesis se centra en la resolución de problemas de localización de instalaciones en espacios continuos. Un problema de localización surge cuando queremos responder la pregunta de dónde ubicar alguna instalación que provee un servicio a un conjunto de usuarios. Este tipo de problemas pertenecen a la Teoría de la Localización, que ha tenido un gran desarrollo desde la década de los 60. Está definida principalmente por los problemas de localización de instalaciones que consisten en la búsqueda de las posiciones óptimas, según algún o algunos criterios dados, para un conjunto de instalaciones con respecto a un conjunto de usuarios.

Existen distintas clasificaciones de los problemas de localización, una de ellas es basada en la naturaleza del conjunto potencial donde ubicar los servicios. En Localización Discreta, la localización de las instalaciones se escoge en un conjunto finito de lugares potenciales; el espacio continuo considera el espacio donde esta definido el problema y hay infinitas posiciones para localizar; y el cuando consideramos una red como espacio de localización, las instalaciones pueden ser localizadas en los nodos o en los arcos de la misma. El uso de cada espacio vendrá dado por la aplicación real del problema. El caso discreto solemos utilizarlo cuando localizamos servicios físicos como colegios, hospitales, supermercados o ATM; el continuo cuando la localización puede ser más flexible, como en ruters, cámaras de vigilancias o sensores; y la red cuando los elementos a localizar son usados en aplicaciones con redes como paradas de autobús o gasolineras.

El Capítulo 1 se trata de un capítulo introductorio en el que se desarrollan los contenidos básicos que contiene los problemas estudiados en esta tesis. Su estructura está organizda en base a los cuatro conceptos principales que motivan esta tesis. Comenzaremos explicando la Teoría de la Localización, donde nos centraremos en sus orígenes y en los problemas que consideran espacios continuos como el espacio de localización para las instalaciones. Introduciremos el *Problema de Weber* y su extensión a más de una instalación. El segundo concepto que desarrollaremos son los problemas de cubrimiento, que se trata de un tipo de problema de localización muy estudiado en esta disciplina. Los problemas de cubrimiento surgen cuando decidimos localizar instalaciones que proveen un servicio y el cliente solo puede recibirlo si está a una cierta distancia de su instalación más cercana. Después mostraremos los operadores que son usados para tratar con problemas

multicriterio y aquellos que son usados en el contexto de la localización. Terminaremos el capítulo introduciendo el concepto de justicia y cómo la literatura ha trabajado con él.

En el Capítulo 2 presentamos un nuevo procedimiento para resolver un problema estudiado en la literatura, el problema de localización continua usando mediana ordenada monótona. El objetivo es localizar un conjunto de servicios en el espacio considerado que minimice una función de mediana ordenada monótona de las distancias entre los puntos de demanda y su instalación más cercana. Este problema fue estudiando en la literatura y se propuso una reformulación entera mixta usando el cono de segundo orden que fue capaz de resolver instancias de tamaño mediano (hasta 50 puntos) usando los solvers comerciales. En este capítulo introducimos un procedimiento de *branch-&-price* y tres familias de matheuristicas. Mostramos la superioridad de este enfoque con respecto a la formulación presente en la literatura y reportamos unas extensas baterías de pruebas computacionales.

En los siguientes capítulos nos centramos en extensiones del problema de máximo cubrimiento, uno de los paradigmas considerados cuando se clasifican los problemas de cubrimiento. En este tipo de problemas, tenemos instalaciones que proveen su servicio en un área restringida y queremos maximizar la demanda cubierta por las instalaciones a localizar. Este área suele ser definida como la distancia máxima a la que las instalaciones dan el servicio, como por ejemplo en la localización de antenas y señales, o el tiempo máximo que un usuario esta dispuesto a emplear para ir a esa instalación, como en las paradas de autobús y metro.

En particular, el problema que abordamos en el Capítulo 3 trata de localizar instalaciones con servicio restringido en un espacio continuo y que las instalaciones estén conectadas mediante una estructura de grafo dadas. Entendemos que dos instalaciones pueden estar conectadas siempre que no se supere una distancia máxima dada. Este tipo de situación aparece cuando localizamos estaciones de bomberos forestales que deben estar comunicadas con un servidor central a una distancia máxima dada o en la localización de sensores que deben estar conectados unos con otros. Se propone un formulación entera mixta no lineal que resuelve el problema y se deriva una reformulación entera pura basada en la geometría del problema. Además proponemos dos enfoques de *branch-&-cut* relajando algunas restricciones de la reformulación entera y desarrollamos un matheuristico capaz de resolver instancias más grandes.

La mayoría de la literatura en la Teoría de la Localización considera un mismo tipo de instalación. Sin embargo, en algunas situaciones, este no puede ser el caso, como por ejemplo cuando tenemos que localizar equipos antiguos y modernos con características diferentes, pero que proveen el mismo servicio. En el Capítulo 4 consideramos este tipo de situaciones aplicado a los problemas de máximo cubrimiento. El objetivo es localizar distintos tipos de instalaciones que proveen el mismo servicio pero con diferentes áreas de cobertura y en diferentes espacios métricos. El uso de distintos espacios métricos representa la posibilidad de localizar en espacios discretos y continuos. Presentamos un modelo general no lineal para cualquier tipo de espacio de localización, discretos y continuos, y lo reformulamos como un problema entero lineal. En particular, centramos nuestro estudios en la localización de instalaciones en conjunto discretos y en el plano Euclídeo. En este caso derivamos otra reformulación entera no lineal basada en la geometría del espacio. Por último, estudiamos las tres formulaciones en una extensa batería computacional donde consideramos conjunto de datos reales de hasta 920 puntos de demanda.

En el Capítulo 5 introducimos el concepto de justicia en los problemas de máximo cubrimiento y cómo afecta a la localización de las instalaciones con servicio restringido. Con este propósito, introducimos una función nueva en la literatura que generaliza dos operadores que han sido usados en la literatura. Esta nueva función depende de dos parámetros y modela un reparto justo de las demandas a las instalaciones para que según el decisor pueda reducir la diferencia entre la que más cubre y la que menos. Esta función influye en la decisión final de dónde localizar las instalaciones para un reparto más justo de la demanda cubierta. Proponemos una formulación general para los espacios de localización discretos y continuos, y particularizamos para cada uno de ellos. Proponemos un modelo no lineal y derivamos reformulaciones enteras mixtas de cono de segundo orden. Finalmente, probamos la eficiencia de los modelos en un conjunto de datos real y mostramos que la inclusión de ambos operadores usados en la literatura proporciona distintas soluciones que si se tienen en cuenta por separado.

Esta tesis doctoral finaliza en el Capítulo 6 con las conclusiones de la investigación realizada y se presentan futuras líneas de trabajo abiertas.

Publicaciones relacionadas

Los problemas estudiados en los distintos capítulos de esta tesis doctoral están basado en los siguientes trabajos donde alguno ha sido publicado y otros han sido envíados para su publicación en revistas científicas de impacto internacional en el área de Investigación Operativa:

Capítulo 2

Blanco, V., Gázquez, R., Ponce, D., & Puerto, J. (2021). A Branch-and-Price approach for the Continuous Multifacility Monotone Ordered Median Problem. Preprint disponible en arXiv:2108.00407.

Capítulo 3

Blanco, V. & Gázquez, R. (2021). Continuous maximal covering location problems with interconnected facilities. *Computers & Operations Research*, 132, 105310, https://doi.org/hqwx

Capítulo 4

Blanco, V., Gázquez, R., & Saldanha-da-Gama, F. (2021). Multitype Maximal Covering Location Problems: Hybridizing discrete and continuous problems. Preprint disponible en arXiv:2111.14494.

Capítulo 5

Blanco, V. & Gázquez, R. (2022). Fairness in Maximal Covering Location Problems. Preprint disponible en arXiv:2204.06446.

Otras contribuciones

Fuera de los contenidos de esta tesis doctoral pero durante el transcurso de ella se ha realizado otra contribución que ha sido enviada para su publicación:

Blanco, V., Gázquez, R., & Leal, M. (2020). Reallocating and sharing health equipments in sanitary emergency situations: The COVID-19 case in Spain. Preprint disponible en arXiv:2012.02062.

Abstract

This thesis focuses on the family of the continuous location problems. A location problem arises whenever a question of where to locate something is raised. This kind of problems belongs to one of the research areas of Operations Research which has had a greatest development since the 1960s, the Location Science. This discipline is mainly defined by the facility location problems which consist of finding the optimal locations for a set of facilities with respect a set of demand nodes and a given objective function.

There are many classifications of location problems, one of them is the one that considers the location space as a classifier. In a discrete location problem, facilities can be located in a finite set of potential locations; the continuous space consider the space where the problem is defined and there are infinite positions to locate; and when we consider a network as a location space, facilities can be located at the nodes or at the arcs of the network. The use of each space will be given by the real application of the problem. We usually use the discrete case when we locate physical services such as schools or ATMs; the continuous one when the location can be more flexible, as in routers or sensors; and the network when the elements to be located are used in applications with networks such as bus stops or gas stations.

Chapter 1 is devoted to provide a background of the theoretical contents that are used in the problems studied in this thesis. It is organized based on the main pillars of this thesis. We start by providing a general framework for Location problems, fixing the notation, but mainly focused on continuous location problems. We introduce the *Weber Problem* and its extension to several facilities. The second main ingredient are covering location problems. These problems arise when deciding where to locate facilities that provide a service, the client can only receive the service if he/she is at a certain distance from his closest facility. After, we show the operators used in the literature to deal with multicriteria problems and those used in the context of location. Finally, we finish the chapter by giving some insights of the notion of fairness and how it has been treated in the literature.

In Chapter 2 we propose a new procedure to solve the Continuous Multifacility Monotone Ordered Median Problem whose goal is to locate a given number of facilities in a continuous space minimizing a monotone ordered weighted median function of the distances between given demand points and its closest facility. This problem was studied in

the literature where it was provided a mixed integer second order cone optimization reformulation for the problem and it was able to solve problems of small to medium size (up to 50 demand points) using commercial solvers. In this chapter we propose a *branch-and-price* procedure and three families of matheuristics to solve it. We report the superiority of this new approach over the existing in the literature solving an extensive battery of computational experiments.

In the following chapters we focus on different extensions of continuous maximal covering location problems whose goal is to locate a given number of services to maximize the amount of demand covered within a maximal service distance or time standard by locating a fixed number of facilities. This maximal distance is used when the facilities to locate are antennas and sensors, or in case the maximal time is when a user goes to its closest bus stop.

In particular, in Chapter 3 we analyze a continuous version of the maximal covering location problem, in which the facilities are required to be linked by means of a given graph structure (provided that two facilities are allowed to be linked if a given distance is not exceed). This type of situation arises when in the design of forest fire-fighters centers that must be communicated to a central server at a give radius or in the location of sensors that have to be connected to each others. We provide a Mixed Integer Non Linear Programming formulation for the problem and derive some geometrical properties that allow us to reformulate it as an equivalent pure integer linear programming problem. We propose two branch-and-cut approaches by relaxing some sets of constraints of the former formulation, and we develop a matheuristic algorithm for the problem capable to solve instances of larger sizes.

On the other hand, one may note that most of the existing literature on maximal covering location problems consider a single type of facility to locate. However in some situations, this may not be the case, for example when one has old equipment and one desires to locate new ones with other characteristics but providing the same service. In Chapter 4 we consider this type of situations applied to maximal covering location problems. The goal is to locate different types of facilities that provide the same service but with different coverage areas and in different metric spaces. The use of different metric spaces allows one to connect the location of facilities in discrete and continuous spaces. We present a general modeling framework for a multitype maximal covering location problem and we provide a non-linear model for which an integer linear programming reformulation is derived. In particular, we strengthen the general methodology by assuming that the continuous facilities are to be located in the Euclidean plane. In this case, taking advantage of some geometrical properties of the problem, an alternative integer linear programming model is proposed. Finally, we report the results of an extensive battery of computational experiments with data up to 920 demand nodes.

In Chapter 5 we introduce the concept of fairness in maximal covering problems and how it affects on the location of facilities. For this purpose, we introduce a novel function generalizing two operators that have been used in the literature with the same purpose. This new function depends on two parameters and incorporate fairness measures from the facilities' perspective so that, according to the decision-maker, they can reduce the difference between the one that covers the most and the one that covers the least. This function determines the final decision of where to locate the facilities for a fairer distribution of the covered demand. We provides a general mathematical programming based framework to incorporate fairness measures from the facilities' perspective to discrete and continuous location spaces. The models are firstly formulated as Mixed Integer Non-Linear programming problems for both the discrete and the continuous frameworks. Suitable Mixed Integer Second Order Cone programming reformulations are derived using geometric properties of the problem. Finally, we test the efficiency of the models on a real data set and the obtained results support that the inclusion of both operators used in the literature provides different solutions than if they are considered separately.

This PhD dissertation ends in Chapter 6 with the conclusions of the research carried out and future open lines of work are presented.

Related publications

The problems studied in the different chapters of this doctoral thesis are based on the following works where some have been published and others have been sent for publication in scientific journals of international impact in the area of Operations Research:

Chapter 2

Blanco, V., Gázquez, R., Ponce, D., & Puerto, J. (2021). A Branch-and-Price approach for the Continuous Multifacility Monotone Ordered Median Problem. Preprint available in arXiv:2108.00407.

Chapter 3

Blanco, V. & Gázquez, R. (2021). Continuous maximal covering location problems with interconnected facilities. *Computers & Operations Research*, 132, 105310, https://doi.org/hqwx

Chapter 4

Blanco, V., Gázquez, R., & Saldanha-da-Gama, F. (2021). Multitype Maximal Covering Location Problems: Hybridizing discrete and continuous problems. Preprint available in arXiv:2111.14494.

Chapter 5

Blanco, V. & Gázquez, R. (2022). Fairness in Maximal Covering Location Problems. Preprint available in arXiv:2204.06446.

Other contributions

Outside the contents of this doctoral thesis but during the course of it, another contribution has been sent for publication:

Blanco, V., Gázquez, R., & Leal, M. (2020). Reallocating and sharing health equipments in sanitary emergency situations: The COVID-19 case in Spain. Preprint available in arXiv:2012.02062.

Contents

Chapter 1

Background

A mathematical model trying to describe a real-world situation often calls for maximizing, or minimizing (the word optimizing includes both), some objective function of the variables which describe the problem (Craven, 2012). This could be the case when a factory requires to calculate the conditions of operation of any process which maximize the benefits or minimize the costs. The technical requirements (usually complex) of the problem required using advanced mathematical optimization tools for its resolution beyond the popular *gradient equal to zero* method.

A mathematical problem in which it is required to calculate the optimum of some function subject to constraints, is called *mathematical programming* problem. Mathematical optimization or mathematical programming is a important branch of Mathematics which has observed an intensive growth since the last century. The origins date back to the studies of Pierre de Fermat and Joseph-Louis Lagrange founding calculus-based formulas to identify optima of functions while Sir Isaac Newton and Johann Carl Friedrich Gauss proposed iterative methods for moving towards an optimum. However, the term "*linear programming*" for certain optimization cases was due to George Bernard Dantzig in 1947, although much of the theory had been introduced by Leonid Vitalyevich Kantorovich (Kantorovich, 1960).

Dantzig (1949) carried out an analysis of military operations and proposed that the interrelationships between the activities of a large organization should be viewed as a type of linear programming model and proposed the *Simplex* algorithm to solve them. It was soon recognized that such models can be used in many other contexts, and this brought about the great interest in the theorical as well as the computational aspects of this field (Vajda, 2009). On the other hand, Neumann (1947) developed the duality theory of the linear programming models being crucial contribution to this field.

Dantzig (2014) defines mathematical programming as the study or use of mathematical programs which comprises theorems about the forms of a solution; algorithms to seek a solution (or certifications that none exists); formulations of problems into mathematical programs, including understanding the quality of one formulation in comparison with another; analysis of results; theorems about the model structure, including properties pertaining to feasibility, redundancy and/or implied relations; theorems about approximation arising from imperfections of models, levels of aggregation, computational error, and other deviations; and developments in connection with other disciplines.

In mathematical programming models, it is required to find the optimum (maximum or minimum) of an *objective function* $F : \mathbb{S} \longrightarrow \mathbb{R}$ on the domain $\mathbb{S} \subset \mathbb{R}^d$, and subject to a set of constraints defined by functions $g_i : \mathbb{S} \longrightarrow \mathbb{R}$, for $i = 1, \ldots, d$. A general way to express a mathematical program is:

$$\begin{aligned} \operatorname*{opt}_{x\in\mathbb{S}} \ & F(x), \\ \text{s.t. } & g_i(x) \leq b_i, \ i = 1, \ldots, d, \\ & x \in \mathbb{S}, \end{aligned}$$

where opt means maximize or minimize the objective function and $b_i \in \mathbb{R}$, for $i = 1, \ldots, d$.

Since these formulations represent real world problems, there are different applications of mathematical optimization in all fields of research. Some examples are Data Analysis (Blanco et al., 2021d; Marín et al., 2022), Project scheduling and Management Science (Correia et al., 2012), Medicine (Carrizosa et al., 1992), among others.

Part of the objectives of this thesis is to derive and solve mathematical programs for problems that arise in the context of the location of facilities. This chapter shows a brief background of the theoretical contents that we use in this PhD dissertation. It is defined by four main concepts. We will start in Section 1.1 with the contextualization of Location Science, which studies location problems. We study its roots and define the main elements involved in continuous location problems, that is, the location space for the facilities is the metric space where our set of clients is located. Part of the problems studied in this thesis are the maximal covering location problems that belong to the branch of covering problems which is a core in Location Science. We explain in Section 1.2 what the covering problems are and the two paradigms considered in the literature when to classify this type of problems. Finally we focus on the continuous maximal covering problems. Later in Section 1.3 we define multicriteria optimization problems and the popular aggregation operators used in the literature to deal with them, and those that have been used in the context of localization. The chapter ends in Section 1.4 by explaining the notion of fairness and its use in the literature and in location problems.

1.1 Location Science

One of the research areas of Operations Research that has had its greatest development since the 1960s has been Location Science (Smith et al., 2009). Thanks to this growing activity it has been recognized by the American Mathematical Society with the code 90B85. A location problem arises whenever a decision maker desires to locate something is raised. This discipline is mainly defined by the facility location problems. These problems consist, in a general and non-rigorous manner, of finding the optimal locations for one or more facilities with respect a set of demand nodes, existing facilities, feasible domains, and a given objective function.

Although the development of the Location Science dates since the 1960s, the origins of this area of research are much older, from the time of the Greeks. Wesolowsky (1993) states that the Greek geometers had already given at least three solutions to the location problem that is nowadays known as the *Weber Problem*.

More recently, according to Kuhn (1967), it may be argued that location analysis was originated in the 17th century with Pierre de Fermat's (1601–1665)[1] problem: given three points in the plane, find a fourth point minimizing the sum of its distances to the three given points. During the 17th century different solutions were proposed to solve the problem. Evangelista Torricelli (1608–1647) was the first to propose a geometric approach to find that fourth point, the so-called *Fermat-Torricelli point* (see Laporte et al., 2019a, for details).

However in the last century, the *Weber Problem* introduced by Weber (1909) is the one that is assumed to determine the era of modern location analysis (Smith et al., 2009). This problem consists of finding a point in the plane that minimizes the sum of weighted Euclidean distances to a set of fixed demand points.

Although the Weber Problem is considered as the problem founding location analysis, it is a natural extension of the problem previously proposed by Launhardt (1900) (see Laporte et al., 2019a; Pinto, 1977, for an extended discussion). It consists of the three-node Weber Problem, that is, finds the fourth point which minimizes the sum of weighted Euclidean distances to three given points.

Both the Weber Problem and the Laundhard Problem were motivated by the location of a facility in industrial context that minimize the weighted sum from suppliers and customers, where weights represented relative volumes of interactions. Launhardt (1900) proposed a geometric solution scheme for the problem, while Weber (1909) presented a deeper analysis of the problem. The problem was solved using a different approach but this resulted in the same solution.

Weiszfeld (1937) provide an algorithm to solve the Weber Problem with an arbitrary number of fixed nodes. This is a least squares method with iteratively changing weights, that converges to an optimal solution.

The 1960s set the foundations of Location Science as new scientific area with the natural extension of the Weber Problem from locating a single facility to the multi-facility case (Cooper, 1963; Miehle, 1958, among others). Indeed, Cooper (1963) introduced the planar p-median problem being a fundamental problem in Location Science, which still attracts the attention of the scientific community (Alcaraz et al., 2012; Labbé et al., 2017, among others).

After that, different versions of the problem were introduced which involve location spaces: the p-median on a network by Hakimi (1964, 1965), and the single facility location problem in a discrete setting by Balinski (1965), among others; or the objective function: Hakimi (1964) proposes p-centre problem, which finds the location of p facilities on a network to minimize the maximum distance from demand to the closest facility, and the

[1]The problem is presented in his essay on maxima and minima.

covering location problems introduced by Toregas et al. (1971). These are used when the facilities that provide a service, a costumer can receive it only if is located close-enough to the facilities (Garcia-Quiles and Marín, 2019).

There are different classifications schemes for location problems. Concretely, ReVelle and Eiselt (2005) characterizes location problems based on four components (costumers, facilities, a space, and a metric). Daskin (1995) classifies the location models into: p-median, p-center and covering problems. Others classify the models depending the space: discrete, network or continuous. The more general classification scheme is defined by Hamacher and Nickel (1998): (1) number and type of new facilities, (2) solution space characteristics, (3) a set of costumers and the relation with new facilities, (4) a metric that indicates distances or times to measure how far are facilities from users, and (5) the objective function.

The new facilities are characterize by: the number of them to locate, which it could be specified before solve the problem, as p-median or p-center, or determined by the problem as, for instance, set covering problems or uncapacitated facility location problems; the nature of service, that is, the facility could be attractive services, or instead it can be obnoxious facilities and the decision maker might want it to be as far as possible from the demand nodes; and finally, by the size of the facility, sometimes the problem is to locate capacitated or uncapacitated facilities as, for instance, in the location of warehouses.

Different solution spaces can be considered for location problems. The most popular are the discrete, the networks and continuous frameworks. One has a discrete spaces when a finite set of potential locations for the facilities is provided, this could be the case when the facilities to locate is physical services like ATM or schools. Networks spaces appears when the facilities are to be located geometrically in a graph. The are useful to represent communication networks, where the nodes represent the important elements of the communication network as cities, and the arcs represent the connections between the nodes like roads. In this case it could be located physical services of a network such as bus stops or subway entrances. The continuous framework is used when the problem can not be discretized or the facilities can be more flexibly located like routers or antennas. Other spaces used in the literature are the sphere (Drezner, 1985) or functional spaces (Puerto and Rodríguez-Chía, 1999).

Normally, two types are usually considered for the set of users: a finite set of demand clients or by regions in the solution space. When a finite set of demand clients is given, we also have a set of associated weight representing the value of demand on this costumer. In case we have regions, each one has associated a probability measure which gives the demand in each point of the region.

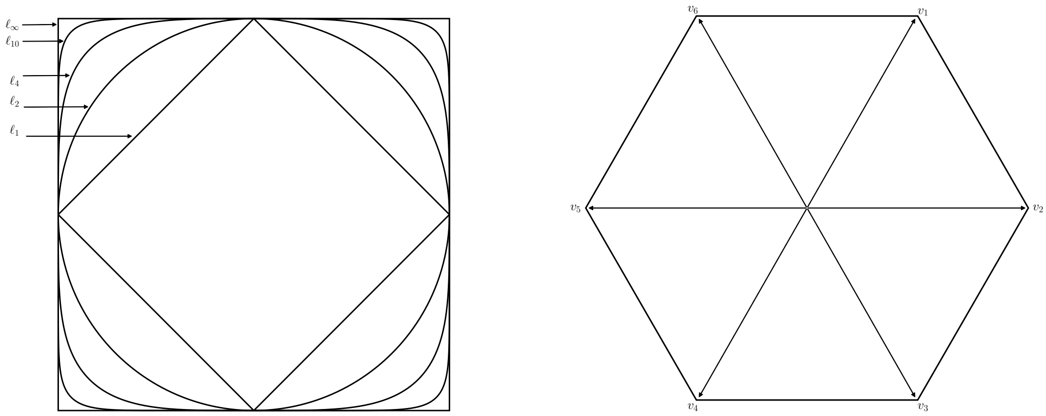
The metric allows one to determine the the relationship between the new facilities and the costumers. This relationship is fundamental in location problems. The most popular metrics are the ℓ_τ-norms, with $\tau \geq 1$, given by

$$\|x\|_\tau = \left(\sum_{l=1}^{d} |x|^\tau\right)^{\frac{1}{\tau}},$$

or the polyhedral norms with a symmetric polytope B (with respect to the origin) (Ward and Wendell, 1985),

$$\|x\|_B = \min\{\sum_{g=1}^{G} |\beta_g| : x = \sum_{g=1}^{G} \beta_g v_g\},$$

where $\{\pm v_1, \ldots, \pm v_G\}$ are the extreme points of B. Figure 1.1 show different examples of unit balls of the considered norms.



(a) ℓ_τ-norms with $\tau \in \{1, 2, 4, 10, \infty\}$ (b) A example of block norm

Figure 1.1: Examples of norms

Due to the evolution of computer technologies, Location Science has observed a rapid growth in last decades (Church, 1999), and more realistic versions of facility location problems have been proposed in the literature.

In a general p-facility location problem, we are given a set $\mathcal{A} = \{a_1, \ldots, a_n\}$ of clients in a given space $\mathbb{S}$ endowed with a norm measure $\|\cdot\|$. Location problems look for the locations of a set $\mathcal{X} = \{X_1, \ldots, X_p\}$ of p new facilities in the defined space such as the quality of a solution is evaluated by a function on the relation between both sets, $\mathcal{A}$ and $\mathcal{X}$. Following the shape of a mathematical programming model defined at the beginning of this Chapter, a general shape to represent objective functions in location analysis is

$$\operatorname*{opt}_{\mathcal{X}=\{X_1,\ldots,X_p\}\in\mathbb{S}} F(\{\|a - \mathcal{X}\|_{a\in\mathcal{A}}\}),$$

where F is a globalizing function, and opt means optimize (minimize or maximize). Note that if $p = 1$ we have a single-facility model or if $p > 1$ a multi-facility model.

The determination of the function F is a crucial decision to represent the problem. Some examples of these objective functions which turns into classical location problems are:

- The Weber Problem (Weber, 1909) is a single-facility, $p = 1$, location problem in the plane, $\mathbb{S} = \mathbb{R}^2$, which tries to minimize the weighted Euclidean distance to a set of fixed points. Let be $w(a)$ the associated weight for each $a \in \mathcal{A}$, and ℓ_2-norm, $\|\cdot\|_2$, as metric. The objective function states as follows,

$$\min_{X \in \mathbb{R}^2} \sum_{a \in \mathcal{A}} w(a) \|a - X\|_2.$$

- The planar p-median problem introduced by Cooper (1963) where each demand node must be served by one out of p new facilities to be located. This is the natural extension of the Weber problem to multiple facilities. Thus, the objective function is quite similar to the presented before,

$$\min_{\mathcal{X}=\{X_1,\ldots,X_p\} \subset \mathbb{R}^2} \sum_{a \in \mathcal{A}} w(a) \min_{j=1,\ldots,p} \|a - X_j\|_2.$$

- The center problem introduced by Hakimi (1964) look for a facility location problem on a network minimizing the maximum distance from the demand nodes to its closest facility. In this case, we are in a graph, $G = (V, E)$, so that the vertices of the graph are the demand nodes, $\mathcal{A} = V$, the facility belongs to edges or vertices of the graph, $\mathbb{S} = V \cup E$, and the space is endowed with $\|\cdot\|_2$-norm. The objective function is:

$$\min_{X \in \mathbb{S}} \max_{v \in V} \|v - X\|_2.$$

- The set covering problem introduced by Toregas et al. (1971) minimizes the number of open restricted facilities such that all demand nodes are covered. In this case the facilities belong to a finite set of potential facilities $\mathbb{S} \subset \mathbb{R}^2$, the Euclidean distance as metric, and the facilities have a restricted area to provide the service, that is, a node is said to be covered if there is at least one open facility that is less than a given value away R. The objective function can be written as,

$$\min_{\mathcal{X} \subset \mathbb{S}: d(\mathcal{X},a) \leq R, a \in \mathcal{A}} |\mathcal{X}|,$$

  where $|\mathcal{X}|$ representing the cardinality of $\mathcal{X}$, that is, the number of opened facilities.

- The maximal covering location problem introduced by Church and ReVelle (1974) assumes the existence of a budget for opening facilities and the goal is to accommodate it to satisfy as much demand of the users as possible. Here, the authors consider the Euclidean space, $\mathbb{S} = \mathbb{R}^2$ and $\|\cdot\|$ the Euclidean norm, and the facilities have a restricted area to provide the service. Let be $w(a)$ the associated weight for each $a \in \mathcal{A}$, and R the maximum distance to provide service from the facility. The objective function is:

$$\max_{\mathcal{X} \subset \mathbb{S}} \sum_{a: d(\mathcal{X}, a) \leq R, a \in \mathcal{A}} w(a).$$

For further examples in location analysis, see the recent book by Laporte et al. (2019b).

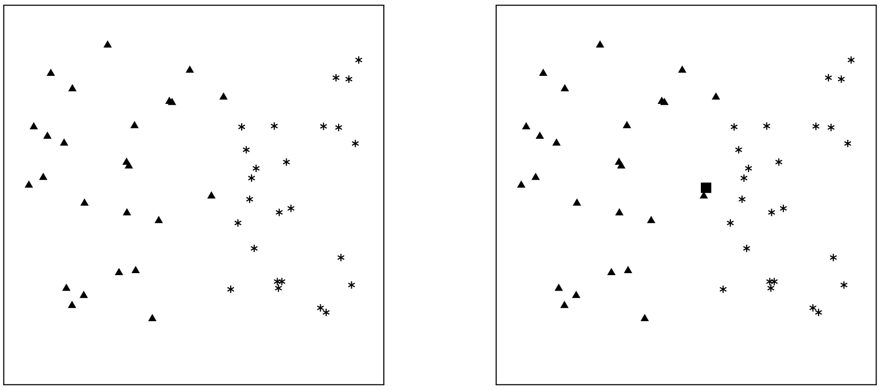
This dissertation aims to solve particular problems in continuous metric spaces. In the next section we introduce continuous problems and how to represent them as mathematical programming problems.

1.1.1 Continuous location problems

This section focuses in a particular type of location problems where the facilities sites are to be located in a continuous space and demands is defined as point into the space. Concretely, we describe the so-called continuous multi-facility Weber Problem mentioned in the previous section. We first review the related literature, and we formulate the problem under mathematical programming lens.

In the last years, a lot of attention has been paid to the discrete aspects of location analysis and a large body of literature has been published on this topic (see, e.g., Beasley, 1985; Elloumi et al., 2004; Espejo et al., 2009; García et al., 2011; Marín et al., 2009, 2010; Puerto et al., 2013; Puerto and Tamir, 2005). One of the reasons of this flourish is the recent development of integer programming and the success of MIP solvers. In spite of that, as we said in the previous section, the mathematical origins of this theory emerged very close to some classical continuous problems as the well-known Fermat or Weber Problem (see, e.g., Laporte et al., 2019a; Nickel and Puerto, 2006, and the references therein). However, the continuous counterparts of location problems have been mostly analyzed and solved using geometric constructions, valid on the plane and the three dimensional space, that are difficult to extend when the dimensions grow or the problems are slightly modified to include some side constraints (Blanco et al., 2017; Carrizosa et al., 1995, 1998; Fekete et al., 2005; Nickel et al., 2003; Puerto and Rodríguez-Chía, 2011). These problems, although very interesting, quickly fall within the field of global optimization and they become very hard to solve. Even those problems that might be considered as *easy*, as for instance the classical Weber Problem with Euclidean norms, are most of the times solved with constructive algorithms (as the Weiszfeld algorithm, Weiszfeld (1937)). Moreover, most problems studied in continuous location assume that a single facility is to be located, since their multifacility counterparts lead to difficult non-convex problems (Blanco, 2019; Blanco et al., 2014; Brimberg, 1995; Carrizosa et al., 1998; Mallozzi et al., 2019; Manzour-al Ajdad et al., 2012; Puerto, 2020; Reinelt, 1992a; Valero-Franco et al., 2013).

The Weber Problem was introduced at the core of industrial location. In fact, it consists of locating an industrial plant that minimizes the transport cost of bringing the needed raw materials to the plant from a fixed number of suppliers as well as the costs of transporting the final product to a given set of markets on the Euclidean plane. Figure 1.2 shows this case where triangles and stars represent the markets and the suppliers respectively, and the optimal location for the industrial plant is represented as a square.



(a) Set example (b) Weber Problem solution

Figure 1.2: Example of Weber Problem with markets (triangles), raw materials (stars), and optimal facility location (square)

Although Weber introduced the problem in the industrial context, mathematical model under this problem consists of finding the coordinates of a facility by minimizing the weighted Euclidean distance to the given demand points, in this case a set of raw materials and a set of market locations. The multi-facility version of the Weber Problem consists of locating p facilities in the euclidean space which minimize the weighted distance to their closest point.

Here, we present the generalized version of the multi-facility problem, that is, locate p facilities in a metric space $\mathbb{S} \subseteq \mathbb{R}^d$ endowed by a $\|\cdot\|$-norm. Let us assume that we are given a set of demand nodes $\mathcal{A} = \{a_1, \ldots, a_n\} \subset \mathbb{S}$ indexed by the set $N = \{1, \ldots, n\}$. Each demand point $a_i \in \mathcal{A}$ has associated a non-negative demand weight ω_i. Throughout the thesis we often call a demand point interchangeably by the node a_i or by the index i. Demand points may represent users or regions and the weights allow one to give more importance to different users or take into account the size/population of each of the regions. Historically, the Weber Problem considers the Euclidean distance, $\|\cdot\|_2$, as metric, but we assume any ℓ_τ or polyhedral norm.

The goal for the multi-facility location problem (MFLP) is to locate p facilities, $\mathcal{X} = \{X_1, \ldots, X_p\}$ in $\mathbb{S}$, indexed by $P = \{1, \ldots, p\}$, minimizing the weighted distances to their closest demand nodes, that is,

$$\min_{\mathcal{X}=\{X_1,\dots,X_p\}} \sum_{i\in N} \omega_i \min_{j\in P} \|a_i - X_j\|. \tag{MFLP}$$

The following set of binary variables is defined to provide a suitable mathematical programming formulation for (MFLP)

$$z_{ij} = \begin{cases} 1, & \text{if node } a_i \text{ is allocated to facility } X_j, \\ 0, & \text{otherwise,} \end{cases} \quad \text{for all } \; i \in N, j \in P.$$

A formulation for the multi-facility location problem (MFLP) is,

$$\min \sum_{i\in N} r_i, \tag{MFLP$_1$}$$

$$\text{s.t. } r_i \geq D_{ij} - M(1 - z_{ij}), \;\; \forall i \in N, \forall j \in P, \tag{MFLP$_2$}$$

$$D_{ij} \geq \|a_i - X_j\|, \;\; \forall i \in N, \forall j \in P, \tag{MFLP$_3$}$$

$$\sum_{j\in P} z_{ij} = 1, \;\; \forall i \in N, \tag{MFLP$_4$}$$

$$X_j \in \mathbb{S}, \;\; \forall j \in P, \tag{MFLP$_5$}$$

where D_{ij} is an auxiliary variable which represents the distance between a demand node and its closest facility, and r_i is also an auxiliary variable which represents the distance between the demand node a_i and its closest facility. Thus, the objective function (MFLP$_1$) minimizes the the weighted distances to their closest demand nodes. The set of constraints (MFLP$_2$) assures the correct allocation for a demand node to its closest facility, where M is a big constant. Family of constraints (MFLP$_3$) gives the correct value of the distance to variable D. Finally, constraints (MFLP$_4$) ensure that each demand node is allocated to a single facility. Note that we only have a set of continuous variables represented as X_j, for $j \in P$.

Theorem 1 (Megiddo and Supowit (1984))**.** *The MFLP is a NP-hard problem.*

Proof. Megiddo and Supowit (1984) prove the NP hardness of this problem for the particular case of the Euclidean plane, that is, $d = 2$ and $\|\cdot\|_2$. □

One of the most challenging aspects to deal with this problem is to adequately describe the nonlinear constraint (MFLP$_3$) to solve the problem.

Weiszfeld (1937) was the first author to solve the problem when there were more than 3 points. The author proposed an iterative gradient type algorithm to find or to approximate the solutions of the (MFLP). Weiszfeld was unaware of Weber's work and for several decades this algorithm remains forgotten until in 1973, when Kuhn (1973) rediscovered it and proved its convergence, under some conditions in the Euclidean case. After, Katz (1974) gives another convergence result. With the generalization of (MFLP) to ℓ_τ or polyhedral norms (this is changing the Euclidean norm in (MFLP$_3$) for any norm), several authors tried to generalize the Weiszfeld's algorithm (see Chandrasekaran and Tamir, 1989, for interesting questions concerning resolubility of the algorithm). Morris and Verdini (1979) gave the generalization of the algorithm for ℓ_τ-norms with $\tau \in [1, 2]$. There are studies of its local and global convergence given by Brimberg and Love (1992, 1993), and it was extended by Frenk et al. (1994) under more general assumptions of quasinconvexity of the objective function. After proving local and global convergence, there are also several works that sought to accelerate its convergence (see, e.g, Brimberg et al., 1998; Drezner, 1995, among others).

Other authors derived the same approach as Weiszfeld being aware of the Weber's work independently (Cooper, 1963; Kuhn and Kuenne, 1962; Vergin and Rogers, 1967). The work of these authors together with the Weiszfeld algorithm has motivated the development of numerous extensions of the Weber Problem.

Examples of these extensions are: Drezner and Weslowsky (1980) dealt with the single-facility location problem with demand areas instead of points, after improved by Carrizosa et al. (1998); the work by O'Kelly (1992) extend facilities to hubs; Carrizosa et al. (1995) gave a generalized Weber Problem where both demand locations and the facility to be located may be regions and are assumed to be distributed according to some probability measures inside each region; Butt and Cavalier (1996) gave an algorithm to solve the single-facility version of (MFLP) in the presence of convex polygonal forbidden regions, i. e., the facility can not belong to these regions; or Klamroth (2001) considered linear barriers in the Weber Problem, that is, the costumers are divided on two sides of a linear barrier and the barrier has a finite of possible points to go from one side to another. These are some extensions in the vast literature related about this topic.

The interest in solving Weber's problem using geometric approaches was due to the fact that optimization solvers were not capable of returning solutions to problems with nonlinear constraints. With the development of the second-order cone optimization (SOC) and its use in mathematical programming problems, it allowed solvers to incorporate solution approaches to these problems. Thus, in case the norm used is Euclidean, the solvers are capable of solving problem with constraints of the type (MFLP$_3$). In fact, the family of constraints (MFLP$_3$) when the Euclidean norm is used, it can be rewritten as,

$$t_{ijl} \geq a_{il} - X_{jl}, \ \forall i \in N, j \in P, l = 1, \ldots, d, \quad (\ell_2\text{-norm}_1)$$

$$t_{ijl} \geq -a_{il} + X_{jl}, \ \forall i \in N, j \in P, l = 1, \ldots, d, \quad (\ell_2\text{-norm}_2)$$

$$\sum_{l=1}^{d} t_{ijl}^2 \leq D_{ij}^2, \ \forall i \in N, j \in P, l = 1, \ldots, d, \quad (\ell_2\text{-norm}_3)$$

$$t_{ijl} \geq 0,$$

where t_{ijl} are auxiliary variables that allow to model the absolute values $|a_{il} - X_{jl}|$ with constraints (ℓ_2-norm$_1$) and (ℓ_2-norm$_2$), being a_{il} and X_{jl} the l-th coordinates of the demand point a_i and the facility X_j, respectively. Finally, the family of constraints (ℓ_2-norm$_3$) assures the value of the Euclidean distance given by the auxiliary variable defined above, D_{ij}. Note that for ℓ_1, the Manhattan distance, the last constraint is only defined by the same expression without the squares over the variables.

After, Blanco et al. (2014) defined a general framework to represent any ℓ_τ-norm for $\tau = \frac{r}{s} \geq 1$ as a set of SOC constraints. A general overview for any value of r and s is stated as follow,

$$t_{ijl} \geq a_{il} - X_{jl}, \ \forall i \in N, j \in P, l = 1, \ldots, d, \quad (\ell_\tau\text{-norm}_1)$$

$$t_{ijl} \geq -a_{il} + X_{jl}, \ \forall i \in N, j \in P, l = 1, \ldots, d, \quad (\ell_\tau\text{-norm}_2)$$

$$t_{ijl}^r \leq \xi_{ijl}^s D_{ij}^{r-s}, \ \forall i \in N, j \in P, l = 1, \ldots, d, \quad (\ell_\tau\text{-norm}_3)$$

$$\sum_{l=1}^{d} \xi_{ijl} \leq D_{ij}, \ \forall i \in N, j \in P, \quad (\ell_\tau\text{-norm}_4)$$

$$t_{ijl}, \xi_{ijl} \geq 0, \ \forall i \in N, j \in P, l = 1, \ldots, d, \quad (\ell_\tau\text{-norm}_5)$$

where t_{ijl} are the same auxiliary variables defined earlier. The variables ξ_{ijl} are auxiliary variables that allow to adequately represent the $\ell_{\frac{r}{s}}$-norm. Note that the family of constraints (ℓ_τ-norm$_4$) must be still reformulated as SOC constraints (see Blanco et al., 2014, for further details on this representation). Table 1.1 shows some examples of reformulation of (ℓ_τ-norm$_3$) for different norms, in particular, $\tau \in \{\frac{3}{2}, 3, 4\}$.

$\ell_{\frac{3}{2}}$	ℓ_3	ℓ_4
$t_{ijl}^2 \leq \psi_{ijl}\xi_{ijl}$,	$t_{ijl}^2 \leq \psi_{ijl}D_{ij}$,	$t_{ijl}^2 \leq \psi_{ijl}D_{ij}$,
$\psi_{ijl}^2 \leq D_{ij}t_{ijl}$,	$\psi_{ijl}^2 \leq \xi_{ijl}t_{ijl}$,	$\psi_{ijl}^2 \leq \xi_{ijl}D_{ij}$,

Table 1.1: Constraints for different values of τ to represent (ℓ_τ-norm$_3$)

Therefore, any ℓ_τ-norm for $\tau \geq 1$ can be rewritten as SOC constraints and most of the commercial optimization solvers can solve this kind of problems. Other useful norms used in the literature are the polyhedral norms (see, e.g., Nickel and Puerto, 2006; Ward and Wendell, 1985). If $\|\cdot\|$ is a polyhedral norm, then (MFLP$_3$) is equivalent to:

$$\sum_{l=1}^{g} e_{gl}(a_{il} - X_{jl}) \leq D_{ij},\ \forall i \in N, j \in P, e \in \mathrm{Ext}_{\|\cdot\|^o}, \tag{Pol-norm}$$

where $\mathrm{Ext}_{\|\cdot\|^o} = \{e_1^o, \ldots, e_g^o\}$ are the extreme points of the unit ball of the dual norm of $\|\cdot\|$.

This reformulation has allowed to model and solve some other versions of continuous location problems as Blanco et al. (2017) located a facility when the distance measure is different at each one of the sides of a given hyperplane; Blanco et al. (2018) fitted hyperplanes with respect to a given set of points minimizing the different distance-based errors; the work by Blanco (2019) in which it is assumed that the facilities must be located in a region around their initially assigned location (the neighborhood); or Blanco and Puerto (2021b) proposed an extension of hub location problem where the positions of the hubs are allowed to belong to a region around an initial set of potential positions, among others.

As we have seen, the representation of the norm in an optimization programming problem has been a challenging task on the part of the existing literature being solved by geometric algorithms until the incorporation of SOC constraints in the solvers. Most of the problems considered in this thesis solve location problems in which the facilities can be located in any part of the considered space. We will see that these SOC constraints can be applied to any location problem that involves the use of the rules considered in this chapter.

1.2 Covering problems

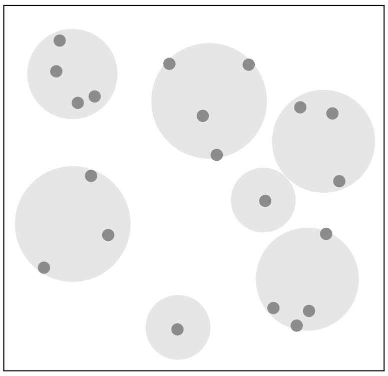
Covering location problem is a particular type of problems within Location Science. These problems arise when deciding where to locate facilities that provide a service but the user can only receive the service if he/she is at a certain distance from his closest facility. When the customer is within that certain distance, the customer is said to be *covered*. An example of this type of problem is when we must locate ambulances at a maximum distance of 7 minutes from people for them to be covered. This section presents a historical background of this kind of problems and we focus on a particular type of them that will be used in the following chapters of this thesis.

The first mentions to covering problems are attributed to Berge (1957), where the author provides an algorithm to find the minimum cover in a graph. After, Hakimi (1965) solved the minimum number of policed patrols required to protect a highway network. The problems were solved using algorithms based on geometric properties. It was not until the work proposed by Toregas et al. (1971) that the problem was formulated mathematically for the first time in the context of location and by Roth (1969) outside of this context.

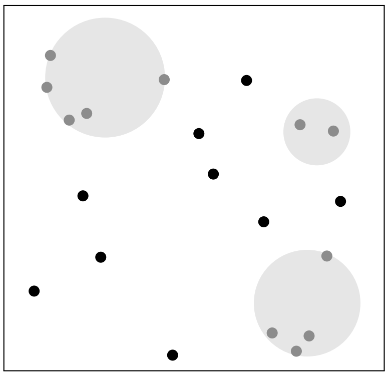
Two different paradigms have been considered when classifying this type of problems. The first considers a cost-oriented objective and the main goal is to satisfy the demand of *all* the users by minimizing the setup costs of the facilities. These problems are referred to in the literature as *Set Covering Location Problems* (SCLP), and were mathematically introduced by Toregas et al. (1971). Particularly, if all the facilities have the same setup costs, the problem is equivalent to minimizing the number of opened facilities. The most popular problem in this family is the p-center problem (Hakimi, 1965).

Normally, SCLP solutions show that one can cover an important percentage the demand with a few facilities, and the total coverage of the demand is usually achieved by a large number of facilities. Therefore, this gives rise to problems that are not in line with reality by assuming that there are sufficient resources to open this large number of facilities. Thus, the second paradigm assumes the existence of a budget for opening facilities and the goal is to accommodate it to satisfy as much demand of the users as possible. These problems belong to the family of *Maximal Covering Location Problems* (MCLP) that have attracted the attention of many researchers since its introduction by Church and ReVelle (1974), both because its practical interest in different disciplines (see Chung, 1986) and the mathematical challenges it poses.

Figure 1.3 shows the differences between these two kind of paradigms. Figure 1.3a shows the solution for the SCLP in which 7 have been selected to cover all demand points and Figure 1.3b shows the solution for MCLP where only three facilities can be selected and have to maximize the covered demand.



(a) SCLP solution.



(b) MCLP solution for 3 facilities.

Figure 1.3: Example of the two paradigms considered to classify the covering problems

Due to its applicability in real-world situations, covering problems will appear in all those location problems in which customers and facilities have to be close enough to provide the service. Applications of those are in different areas such as deployment of emergency services (ReVelle, 1989; Toregas et al., 1971), in natural disaster management (Mandal et al., 2021), humanitarian logistics (Li et al., 2018), in mail advertising (Dwyer and Evans, 1981), fire stations (Schilling et al., 1980), bus stops (Gleason, 1975), in subway and highways (Boffey and Narula, 1998), among others. The Table 1.2 given by Church and Murray (2018b) lists examples of coverage in different applications.

Standard	Context
$3 - 50$ km	Cellular antenna
$5 - 6$ min	Emergency 911 call
70 dB audibility	Outdoor warning or message
400 m	Reasonable waking distance for bus access
70 miles	Essential air service access for rural communities
800 m	Suitable acces for rail/subway
1 day ride	Mail delivery
120 miles	Doppler radar moisture detection
1500 m	Visibility distance of camera mounted on tower
60 min	Areomedical response for trauma care

Table 1.2: Some coverage standards examples from Church and Murray (2018b)

Like most location problems, the SCLP and MCLP may be defined as continuous problems (in which facilities may be located anywhere on the plane), as discrete problems (in which they may be located only at a set of potential facility locations) or as network problems (in which they may be located anywhere on the network). This distinction in the decision space is motivated by the installation to locate, it is known that in covering problems the discrete space is used for the location of physical services (as ATM, schools, hospitals, among others), the continuous for facilities that allow a flexible location (as sensors, routers, radars) and the networks for facilities that belong to a connected network (as bus stops, subway entrances). The literature has several examples and applications on different spaces, in the discrete for the assignment of fire equipment to fire house (Walker, 1974); in the continuous as the work by Goodchild and Lee (1989) in which involving the placement or siting of fire watch tower for quick detection of fire in forests; and in networks as Gendreau et al. (1997) where a set of vertices must be covered; among others.

In the rest of this section, we explicitly introduce the two paradigms for a complete view of covering problems. Section 1.2.1 details the mathematical formulation for the SCLP in the discrete context as it was first formulated by Toregas et al. (1971), albeit with the notation developed in the previous sections. After, Section 1.2.2 introduces discrete MCLP and develops some applications. Finally, Section 1.2.3 focuses on the possible formulations of the continuous MCLP that will be used in the following chapters.

1.2.1 Set covering location problem

The SCLP where the coverage of all demand is required tries to minimize location cost of the facilities. Usually SCLP considers a finite set of demand points in the plane, here we consider in some metric space $\mathbb{S} \subset \mathbb{R}^d$. Let $\mathcal{A} = \{a_1, \ldots, a_n\} \subseteq \mathbb{S}$ be a finite set of demand points indexed by the set $N = \{1, \ldots, n\}$. We are also given a potential set of facilities, $\mathcal{B} = \{b_1, \ldots, b_m\} \subseteq \mathbb{S}$, where the *services* are to be chosen, indexed by the set $M = \{1, \ldots, m\}$. Each potential facility $b_j \in \mathcal{B}$ has associated a non-negative opening cost c_j.

The literature on covering problems considers different points of view regarding coverage areas. On some occasions, the coverage area can be seen as the limit that the facility has to provide its service or as the limit that a user is willing to move to obtain the service. Therefore, the areas can be defined from the point of view of the facility or the client. In this chapter we consider the point of view of the facility, but throughout this thesis we will exchange this point of view according to the problem studied, and we will define it when it comes.

Thus, each potential position for the facilities, $b_j \in \mathcal{B}$ could have associated a coverage area R_j, but sometimes the radius is the same for all facilities, R, in the literature. It is usual to define the coverage areas as Euclidean balls with certain *coverage radia*. In this dissertation, we consider ball-shaped coverage areas centered in a point b in the form:

$$\mathbb{B}_R(b) = \{z \in \mathbb{R}^d : \|z - b\| \leq R\},$$

where $\|\cdot\|$ is the metric of the space $\mathbb{S}$. A demand node a_i is *covered by the facility* if the distance between a_i and the facility does not exceed R. The node a_i is said to be *covered* if there is at least one open facility covering it. For any finite subset of open facilities, $\mathcal{X} = \{X_1, \ldots, X_p\} \subseteq \mathcal{B}$, we denote by $\mathcal{C}(X) \subseteq N$ the set of indices of covered nodes by the open facility X in $\mathcal{X}$, i.e.,

$$\mathcal{C}(X) = \{i \in N : a_i \in \mathbb{B}_{R(X)}(X), \text{ for some } X \in \mathcal{X}\}, \tag{Cov-Set}$$

we also denote $\mathcal{C}(\mathcal{X}) = \bigcup_{j\in=1}^{p} \mathcal{C}(X_j)$ the set of indices of covered nodes by at least one open facility in $\mathcal{X}$.

The goal of the SCLP is to find a subset of facilities $\mathcal{X} = \{X_1, \ldots, X_p\} \subseteq \mathcal{B}$ minimizing the opening cost of the facilities and covering all demand clients, that is,

$$\min_{\substack{\mathcal{X}=\{X_1,\ldots,X_p\}\subseteq\mathcal{B}\,: \\ i\in\mathcal{C}(\mathcal{X}),\ \forall i\in N}} \sum_{j=1}^{p} c_j. \tag{SCLP}$$

For the sake of deriving a suitable mathematical programming formulation for the SCLP, it is required to introduce the following decision variable:

$$y_j = \begin{cases} 1, & \text{if facility } b_j \in \mathcal{B} \text{ is selected,} \\ 0, & \text{otherwise,} \end{cases} \quad \text{for all , } j \in M.$$

Using the above decision variables, the discrete SCLP can be formulated as follows:

$$\min \sum_{j \in M} c_j y_j, \tag{d-SCLP$_1$}$$

$$\text{s.t.} \sum_{\substack{j \in M: \\ a_i \in \mathbb{B}_{R_j}(b_j)}} y_j \geq 1, \; \forall i \in N, \tag{d-SCLP$_2$}$$

$$y_j \in \{0, 1\}, \; \forall j \in M. \tag{d-SCLP$_3$}$$

The objective function (d-SCLP$_1$) minimizes the total fixed cost of opening y_j facilities at site $j \in M$. In the particular case where the cost are equal for each potential location, $c_j = 1 \; \forall j \in M$, the goal is to minimize the number of open centers. The family of constraints (d-SCLP$_2$) assures that all demand nodes are covered by at least one facility and each demand node is covered if and only if it belongs to the ball of a open facility. Finally (d-SCLP$_3$) define the domain of y-variables.

In (d-SCLP) we assume that we have a potential facility locations $\mathcal{B}$, but in the literature is usually used $\mathcal{B} = \mathcal{A}$, that is, the potential facility locations are the same as the demand nodes. That problem is known as the *node covering problem* and it has been widely studied (see, e.g, Balinski, 1965).

Theorem 2 (Karp (1972)). *The SCLP is an NP-complete problem.*

Although the SCLP is NP-complete, the linear relaxations of the problem provide good lower bounds (Garcia-Quiles and Marín, 2019). One of the main drawbacks of covering problems, in particular of the LSCP is that they have usually many different optimal solutions, that is, sets of facilities with the same solution, we see this property in some solutions of the proposed problem in Chapter 5.

The constant growth of this area of Location Science has brought an important and rich literature on this topic. Thanks to this, extensions of covering problems within the two paradigms explained have been developed by the research community. Some example are: capacited set covering problems (Current and Storbeck, 1988) where the problem has facility capacity restrictions; the probabilistic set covering problems (Revelle and Hogan, 1989) where consider some of deterministic parameters as probabilistic; the anti-covering problems (Moon and Chaudhry, 1984) that maximizes the set of selected location sites so that there are no two selected sites within a pre-specified distance. For interested readers about other types of covering problems see Church and Murray (2018b) or Garcia-Quiles and Marín (2019) and references therein.

ReVelle (1989) gives a review focused on emergency service, Plastria (2002) provides a review in the context of continuous covering models, Snyder (2011) and Farahani et al. (2012) review models and extensions of covering problems, Murray (2016) is a survey on MCLP and Garcia-Quiles and Marín (2019) give a general view of covering models.

The rest of this section focuses in the MCLP which is the main topic of the following chapters of this thesis. We will present a general background of these problems, the general formulation and extensions of this work in the literature. Finally, we will focus our attention on the case where the location space is continuous.

1.2.2 Maximal covering location problems

Sometimes, SCLP leads to unrealistic problems since the number of facilities needed to cover all the points can be excessive. The demand points are required to be covered, regardless of their quantity and density, resulting in solutions that could be not economically feasible. On the other hand, the SCLP considers all demand points equally. These concerns lead to the Maximal Covering Location Problem (MCLP) model introduced by Church and ReVelle (1974). The MCLP does not require that all demand points be covered. Instead, the MCLP maximize the amount of demand covered within a maximal service distance by locating a fixed number of facilities (Church and Murray, 2018a).

The MCLP was introduced by Church and ReVelle (1974), although, at the same time White and Case (1974) also realized that there needed to be more flexibility on covering location problems. The authors defined the same problem, a covering location problem with a budget of facilities to be located, but they considered all the demand points identically.

Since its introduction, the MCLP and its extensions of it have been studied in many works and it has attracted significant attention from both researchers and practitioners (Wei and Murray, 2015), both by its technical merit and practical interest. Indeed, since its first publication and until 2015 Church and ReVelle (1974) had around 1550 citations (Murray, 2016). At the days of writing this thesis it has more than 3200 citations in Google Scholar, which represents an increase in research and interesting about the problem.

There are tons of applications of the MCLP in the literature, including the location of health clinics (Bennett et al., 1982), positioning ambulances (Saydam and McKnew, 1985), cluster analysis (Chung, 1986), placement of emergency warning sirens (Current and O'Kelly, 1992), selecting sites for nature reserve (Church et al., 1996), cellular network design (Kalvenes et al., 2005), designing police patrol areas (Curtin et al., 2010), locating fire stations (Murray, 2013), just to cite a few. For interested readers on applications are referred to the recent survey in MCLP by Murray (2016) or the book on covering problems by Church and Murray (2018b).

As usual, the use of MCLP has been applied in different location spaces based on the applications. The classical MCLP was defined in a network space, where the facilities can be located in any node of the graph. For the sake of readability, we formulate the classical one as a discrete problem.

Similar to the SCLP, a demand point or area is said to be covered if it is within a predefined service distance or time from at least one facility. Here, as we defined in previous sections, we assume that we are given a set of demand nodes in some metric space $\mathbb{S} \subseteq \mathbb{R}^d$. Thus, we are given the set of demand nodes $\mathcal{A} = \{a_1, \ldots, a_n\}$, indexed by $N = \{1, \ldots, n\}$, and unlike the SCLP, each demand point $a_i \in \mathcal{A}$ has associated a non-negative demand weight ω_i. This is one of the main differences between both problems. Furthermore the number of facilities to be located is limited, all demand in the region may not be covered. A budget constraint is incorporated in the MCLP to relax the rigid requirement of complete coverage of all demand in the SCLP. We are also given a finite set of candidate facilities, $\mathcal{B} = \{b_1, \ldots, b_m\}$, indexed by $M = \{1, \ldots, m\}$, with covering area R_j, $j \in M$, albeit in this case there are no cost for opening facilities.

Using the notation defined by (Cov-Set), the goal of the MCLP is locate p facilities $\mathcal{X} = \{X_1, \ldots, X_p\}$ in $\mathcal{B}$ maximizing the covered demand on the finite set $\mathcal{A}$, that is,

$$\max_{\mathcal{X}=\{X_1,\ldots,X_p\}\subseteq\mathcal{B}} \sum_{i\in\mathcal{C}(\mathcal{X})} \omega_i. \tag{D-MCLP}$$

For the sake of deriving a suitable mathematical programming formulation for the problem we use the following sets of decision variables:

$$y_j = \begin{cases} 1, & \text{if facility } b_j \in \mathcal{B} \text{ is selected,} \\ 0, & \text{otherwise,} \end{cases} \quad \text{for all }, j \in M.$$

$$x_i = \begin{cases} 1, & \text{if demand node } a_i \text{ is covered by the facilities} \\ 0, & \text{otherwise,} \end{cases} \quad \text{for all }, i \in N.$$

Using the above variables, the discrete MCLP can be formulated as follows:

$$\max \sum_{i\in N} \omega_i x_i, \tag{d-MCLP$_1$}$$

$$\text{s.t.} \sum_{j\in M} y_j = p, \tag{d-MCLP$_2$}$$

$$x_i \leq \sum_{\substack{j\in M:\\ a_i\in\mathbb{B}_{R_j}(b_j)}} y_j, \ \forall i \in N, \tag{d-MCLP$_3$}$$

$$x_i \in \{0,1\}, \ \forall i \in N, \tag{d-MCLP$_4$}$$

$$y_j \in \{0,1\}, \ \forall j \in M. \tag{d-MCLP$_5$}$$

In the above model, the objective function (d-MCLP$_1$) maximizes the weighted coverage of the nodes in $\mathcal{A}$ by the facilities. Constraints (d-MCLP$_2$) ensure that exactly p facilities are selected in the finite set $\mathcal{B}$. Inequalities (d-MCLP$_3$) state whether a demand point is covered by open facilities. Specifically, if a demand point a_i is covered, i.e., $x_i = 1$, then there must be at least one facility at a distance less than its coverage radius R_j. Finally, constraints (d-MCLP$_4$) and (d-MCLP$_5$) state the domain of the decision variables, which, with the linear objective function, make the MCLP a linear mixed integer optimization program.

Theorem 3 (Megiddo et al. (1983)). *The MCLP is NP-hard.*

Although the MCLP was introduced in the context of networks, the location of the facilities was limited to the nodes of the network and could therefore be considered as a discrete setting. These assumptions could result in unrealistic problems and the location within the arcs of the graphs began to be considered. Some recent examples of these problems are those made by Berman et al. (2016); Blanquero et al. (2016) or Baldomero-Naranjo et al. (2021).

1.2.3 Continuous maximal covering location problems

An important distinction in location analysis and modeling has long been discrete versus continuous approaches. In previous sections we have reviewed the literature in both contexts, firstly explaining the roots of the location science with the Weber Problem, and finally presenting discrete extensions like the SCLP. The discrete frameworks have enabled discrete integer programming formulations for the problems, allowing for efficient exact approaches. In some situations, however, neither potential facility sites are necessarily known and finite. Thus, one aspect of a continuous space location model is that facilities may be sited anywhere in the decision space. In this section, we focus in the continuous version of the MCLP.

Two paradigms have been considered in the literature when dealing with continuous MCLP: (1) consider the demand set as a finite set of nodes in the metric space; or (2) as a demand distributed throughout the feasible region defined by some function.

Mehrez and Stulman (1982) were the first authors to consider the first paradigm, where the demand is distributed by a finite set of nodes. The authors are credited as the ones that introduced the continuous MCLP using the Euclidean distance, although they did not formulate it mathematically. They exploited the geometric properties of coverage and gave a method to solve the problem based on constructing a finite set where the optimal solution could be found. Later, the works of Mehrez (1983) and Mehrez and Stulman (1984) proposed a discrete formulation for the continuous MCLP using the finite set they discovered. Church (1984) also formulated the continuous MCLP as discrete using the same set, but for rectilinear and Euclidean distances. The sets were named Diamond Intersect Points Set (DIPS) if rectilinear distance was used, and Circle Intersect Points Set (CIPS) in case the Euclidean distance was used. These three works were published independently at the same time, and the strategy to solve it was the same, identifying a finite dominating set of locations as potential facility sites. Afterwards, the concept of discrete demand was generalized. Murray and Tong (2007) proposed a generalization in which the demand could be points, lines and polygons. This type of demand can be treated as discrete since the lines and polygons are defined as a set of points and an object will be covered if and only if the set of points that defines it is covered (see Murray and

Tong, 2007, for further details). Again, the geometric properties of the problem allows for the construction of finite dominating sets, and the authors introduced a method for identifying a finite set of potential facility locations called Polygon Intersection Point Set (PIPS). Thus, the continuous location problem could be formulated as a mixed integer optimization problem.

Later, the scientific community realized that the representation of demand spaces as points could lead to precision errors (Current and Schilling, 1990; Daskin et al., 1989; Murray and O'Kelly, 2002). This is how the second paradigm arises, the study of covering problems where the demand is found throughout the space. Murray and O'Kelly (2002) analyze the problem of locating warning sirens where demand for service exists everywhere in the region and sirens can be located anywhere in the region. The authors gave a general framework which can be applied to general continuous MCLP problems. Since then, other works have been published with different solution strategies and extensions of the problem. For instance, when complete coverage of an entire space is required, it is possible to solve as a p-center problem. The extension of the center problem, introduced in Section 1.1, is the p-center problem which looks for the location of p facilities minimizing the maximum distance to any point in the space to its closest facility. In the case of covering problems, if the optimal solution for the p-center, it satisfies the coverage requirements then the solution is optimal. Suzuki and Okabe (1995), and Suzuki and Drezner (1996) gave an heuristic approach for solving the p-center problem based on Voronoi diagrams, to solve the location of the facilities anywhere in order to serve polygonal-shaped regions. Wei et al. (2006) relaxed the assumption of the polygon shape extending the heuristic. In case the complete regional coverage is not possible. Murray et al. (2008) and Matisziw and Murray (2009a,b) introduced the use of a medial axis to indicate where an optimal facility location can be found in a continuous space. Preparata (1977) defined the medial axis of an arbitrary simple polygon as the set of points of the plane internal to the polygon which have more than one closest point on the boundary of its (the medial axis was introduced by Blum, 1967).

Most of the the proposed approaches for solving continuous covering location are based on constructing finite dominating sets, since they allow one to use tools from linear integer programming to solve the problem (see e.g., Blanco and Puerto, 2021a; Cordeau et al., 2019), even at the price of incurring on errors when computing the dominating set. This use is motivated by the fact that continuous location in covering problems is usually applied in problems of locating routers, sensors or alarms, among others, and the covered demand is usually a finite set of users or buildings. This is why, in this thesis, we analyze covering problems in which we will locate in metric spaces to serve a finite set of demands.

Similar to the discrete counterpart, in the continuous MCLP that we consider during this thesis, we are given a metric space $\mathbb{S} \subseteq \mathbb{R}^d$ endowed by a $\|\cdot\|$-norm as distance measure, and a finite set of demand points $\mathcal{A} = \{a_1, \dots, a_n\} \subseteq \mathbb{S}$, indexed by $N = \{1, \dots, n\}$, and each demand point $a_i \in \mathcal{A}$ has associated a non-negative demand weight ω_i. Thus, the goal of the continuous MCLP is locate p facilities $\mathcal{X} = \{X_1, \dots, X_p\}$ in the metric space $\mathbb{S}$ maximizing the covered demand, that is,

$$\max_{\mathcal{X}=\{X_1,\dots,X_p\}\subseteq\mathbb{S}} \sum_{i\in\mathcal{C}(\mathcal{X})} \omega_i. \tag{C-MCLP}$$

Theorem 4. *The* (C-MCLP) *is NP-hard.*

Proof. The problem reduces to a discrete MCLP by Church (1984) which is NP-hard (Megiddo et al., 1983). □

There are different formulations available in the literature to model the continuous MCLP. Unlike the discrete case, we must allocate the point to the facility that covers it so that its demand is only counted once.

Let us denote by $P = \{1, \dots, p\}$ the index set for the centers. A mathematical programming formulation for the problem can be derived by using the following set of variables:

$$z_{ij} = \begin{cases} 1, & \text{if node } a_i \text{ is covered by facility } X_j, \\ 0, & \text{otherwise,} \end{cases} \quad \text{for all } \ i \in N, j \in P,$$

and $X_j \in \mathbb{S} \subseteq \mathbb{R}^d$: coordinates of the jth facility in $\mathcal{X}$ for all $j \in P$.

A 'natural' nonlinear formulation, similar to the presented (MFLP) is the following,

$$\max \sum_{i\in N}\sum_{j\in P} \omega_i z_{ij}, \tag{c-MCLP$_1^{\text{NL}}$}$$

$$\text{s.t.} \sum_{j\in P} z_{ij} \leq 1, \ \forall i \in N, \tag{c-MCLP$_2^{\text{NL}}$}$$

$$\|a_i - X_j\| \leq R_j, \ \text{if } z_{ij} = 1, \ \forall i \in N, \tag{c-MCLP$_3^{\text{NL}}$}$$

$$z_{ij} \in \{0, 1\}, \ \forall i \in N, j \in P, \tag{c-MCLP$_4^{\text{NL}}$}$$

$$X_j \in \mathbb{S}, \ \forall j \in P, \tag{c-MCLP$_5^{\text{NL}}$}$$

where the objective function (c-MCLP$_2^{\text{NL}}$) accounts for the weighted number of covered points. Constraints (c-MCLP$_2^{\text{NL}}$) enforce that covered demand points are accounted only once in the objective function, even if it can be covered by more than one center. The family of constraints (c-MCLP$_3^{\text{NL}}$) ensure that covered points are those with a center in their coverage radius. Finally, (c-MCLP$_4^{\text{NL}}$) and (c-MCLP$_5^{\text{NL}}$) fix the domain of the variables. Note that constraint (c-MCLP$_3^{\text{NL}}$) can be equivalently rewritten as:

$$\|a_i - X_j\| \leq R_j + M(1 - z_{ij}), \forall i \in N, j \in P, \tag{c-MCLP$_6^{\text{NL}}$}$$

for a big enough constant $M > \max_{i,k\in N} \|a_i - a_k\|$.

The above formulation is clearly discrete and non linear, but the nonlinear constraints (c-MCLP$_6^{\text{NL}}$) can be efficiently reformulated as a set of second order cone constraints, resulting in a Mixed Integer Second Order Cone Optimization (MISOCO) problem (see Blanco et al., 2014). The value of the distance in the nonlinear constraint must be changed by the one given by the set of constraints (ℓ_τ-norm$_1$)-(ℓ_τ-norm$_5$), in addition to adding all of them in case of ℓ_τ-norms, or (Pol-norm) if polyhedral norms are used.

However, in case the problem uses discrete demand (points, lines, polygons or other geometric objects), it is possible to derive discrete potential facility locations that sufficiently represent continuous space. Such discrete sets of potential facility locations are known as a *finite dominating set* (FDS). Church (1984) proved that for the plane ($\mathbb{S} = \mathbb{R}^2$), and the rectilinear and Euclidean norms (ℓ_1- and ℓ_2-norm) it is enough to inspect a explicit finite set of potential centers to find the optimal location of the problem. In particular, the so-called DIPS and CIPS which consists of the demand points and the pairwise intersection of the balls (disks) centered at the demand points and the corresponding covering radii. Murray and Tong (2007) proved that for any demand objects, points, lines or polygons, a finite set of potential facility sites could be identified.

In general, Murray and Tong (2007) generalized the process of generating the FDS for discrete demand as follows:

1. Identify demand objects to be covered.
2. Derive covering areas around each demand object.
3. Find the intersection points of covering areas.

The first step (1) is trivial whatever the representation of the demand, however it must be done for a good representation of the problem. Once the demand has been established, the second step is to find the area where the facilities can cover the demand. For this, in each demand we place the coverage area defined by the facilities. This will give us as a result a set of areas where the facilities can be located to cover that demand. Finally, step 3 looks for the intersection of all those demand-centered coverage areas, giving rise to a set of finite points where at least one optimal solution of the problem is found. The difficulty of this step highly depend on the dimension of the decision space and the metric used to construct the coverage areas.

This dissertation only considers points in the location space as demand. An example of the FDS generation process for demand points and considered by Church (1984) is illustrated in Figure 1.4. Church consider the Euclidean plane, that is, $\mathbb{S} = \mathbb{R}^2$ and $\|\cdot\|_2$ the Euclidean distance. The author also considers the same coverage radius for all the facilities to locate. In Figure 1.4a we show a set of given demand points. In Figure 1.4b we draw the areas around each of these demand points where one should locate a facility to cover it. Note the Euclidean ball shape centered in each demand nodes. With these balls, the points of intersection can be derived giving the set of potential facilities in Figure 1.4c. This is the FDS of defined demand nodes, called by Church (1984) as CIPS. Thus, the problem turns into the discrete version of the MCLP, and the (d-MCLP) can be used, resulting in other suitable formulation for the continuous MCLP.

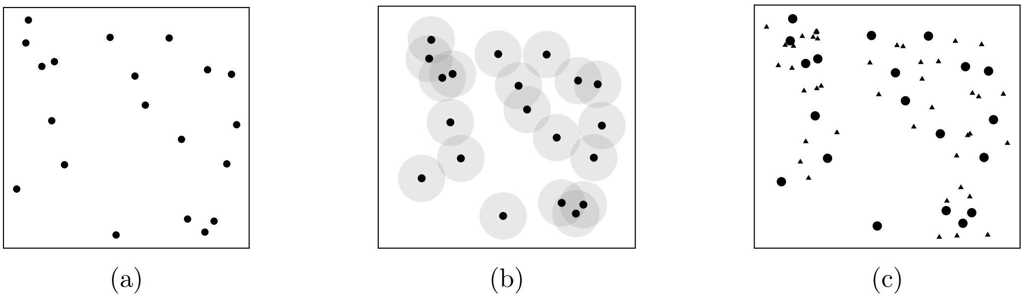


Figure 1.4: Depiction of major steps of FDS process following Murray and Tong (2007). (a) Demand points. (b) Covering areas of demand. (c) FDS

The discrete version of the continuous MCLP, although allows a linear representation of the planar MCLP may has a large number of x-variables. If we assume that all covering radius are equal for all the facilities ($R_j = R,\ \forall j \in P$), we could have a large FDS (in worst case, $O(n^2)$). On the other hand, if we assume different covering radius we could have in worst case $O(pn^2)$.

Another equivalent linear representation of the problem, in case we assume that centers must cover at least one demand point, and that we exploit in this dissertation, is based on the following straightforward observation.

Lemma 1. *Let $\mathcal{C}(X_1), \ldots, \mathcal{C}(X_p) \subseteq N$ be p nonempty disjoint subsets of N. Then, they induce a solution to the MCLP if and only if:*

$$\bigcap_{i \in \mathcal{C}(X_j)} \mathbb{B}_{R_j}(a_i) \neq \emptyset, \forall j \in P.$$

The above result allows us to rewrite constraints (c-MCLP$_3^{\text{NL}}$) as linear constraints and formulate the MCLP as:

$$\max \sum_{i\in N}\sum_{j\in P} \omega_i z_{ij} \tag{c-MCLP$_1^{\rm IP}$}$$

$$\text{s.t.} \sum_{j\in P} z_{ij} \leq 1,\ \forall i \in N, \tag{c-MCLP$_2^{\rm IP}$}$$

$$\sum_{i\in N} z_{ij} \geq 1,\ \forall j \in P, \tag{c-MCLP$_3^{\rm IP}$}$$

$$\sum_{i\in S} z_{ij} \leq |S|-1,\ \forall j \in P \text{ and } S \subseteq N : \bigcap_{i\in S} \mathbb{B}_{R_j}(a_i) = \emptyset, \tag{c-MCLP$_4^{\rm IP}$}$$

$$z_{ij} \in \{0,1\},\ \forall i \in N, j \in P, \tag{c-MCLP$_5^{\rm IP}$}$$

where the set of constraints (c-MCLP$_3^{\rm IP}$) give the assumption of the center must cover at least one demand point, and (c-MCLP$_4^{\rm IP}$) enforces that the set of points covered by a center must verify the condition of Lemma 1. Once the solution of the problem above is obtained, z^*, explicit coordinates of the centers, can be found in the following sets:

$$X_j \in \bigcap_{\substack{i\in N:\\ z^*_{ij}=1}} \mathbb{B}_{R_j}(a_i), \forall j \in P,$$

which can be formulated as a convex feasibility problem:

$$\begin{aligned} \min\ & 0 \\ \text{s.t. } & \|X_j, a_i\| \leq R_j, \quad \forall i \in N : z^*_{ij} = 1, \\ & X_j \in \mathbb{R}^d, \end{aligned}$$

for all $j \in P$.

The convex problem above can be efficiently handled for the most common distance measures using the reformulation given by (ℓ_τ-norm) in the particular case of ℓ_τ-norms or by (Pol-norm) in case of polyhedral norms. Thus, for this wide family of distance measures, and once the discrete part of the problem is solved, the coordinates of the centers can be solved in polynomial-time by using interior-point techniques (Nesterov and Nemirovskii, 1994).

1.3 Ordered Weighted Averaging operators

The previous sections establish the criteria used to locate new facilities. However, in many decision and planning problems involve multiple objectives that should be considered simultaneously due to more than one viewpoint or scenario (multiple conflicting criteria). Such problems are generally known as multiple criteria decision making problems and their solutions should be optimal to several criteria at the same time. This section recall a general class of parameterized aggregation operators to address with different objective functions in multi-objective problems.

The Ordered Weighted Averaging (OWA) operators were introduced by Yager (1988) to provide a means for aggregating scores associated with the satisfaction to multiple criteria. Subsequently they have proved to be a useful family of aggregation operators for many different types of problems and have attracted much interest among researchers. They provide a general class of parameterized aggregation operators that include the *min, max, average*. Since its introduction, the OWA operators have been successfully used in various fields such as multicriteria and group decision making (Chiclana et al., 2003; Herrera et al., 1996), database query management and data mining (Torra, 2004; Yager, 2003), classification problems using support vector machine (Maldonado et al., 2018; Marín et al., 2022), among others. See Yager and Kacprzyk (1997) and Yager et al. (2011) for further applications.

Formally, an OWA operator is a mapping $\Phi_\lambda : \mathbb{R}^p \to \mathbb{R}$ with associated weighting vector $\lambda = (\lambda_1, \ldots, \lambda_p) \in \mathbb{R}^p$ satisfying $\lambda_j \in [0, 1]$, $\forall j \in \{1, \ldots, p\}$ and $\sum_{j=1}^{p} \lambda_j = 1$. For a given vector $\mathbf{x} = (x_1, \ldots, x_p) \in \mathbb{R}^p$, the OWA operator is defined as:

$$\Phi_\lambda(x_1, \ldots, x_p) = \sum_{j=1}^{p} \lambda_j x_{(j)}, \tag{OWA}$$

where $x_{(j)}$ is the jth-largest input vector component of $\mathbf{x}$, i.e., $x_{(j)} \in \{x_1, \ldots, x_p\}$ such that $x_{(1)} \leq \ldots \leq x_{(p)}$. These operators are weighted sums of the different criteria, but where weights are associated with the position of the criteria when they are sorted in non decreasing order. According to Yager, the key feature of this operator is the ordering of the arguments by value, a process that introduces a nonlinearity into the operation (Yager and Kacprzyk, 1997, Chapter 2).

Therefore, the use of an OWA operator is generally composed of the following three steps (Xu, 2005):

1. Reorder the input arguments in non descending order.
2. Determine the weights associated with the operator by using a proper method.
3. Use the weights to aggregate these reordered arguments.

When aggregating different criteria by means of an OWA operator, a crucial step is to provide the adequate weights λ that are assigned to the sorted sequence. Weight vector determination is often a prerequisite step in many OWA-related applications. The most popular OWA operators ($\min - \lambda = (1, 0, \ldots, 0)$ and $\max\text{–}\ \lambda = (0, \ldots, 0, 1)$) serve as reference weights to define the notion of *orness* of a vector of λ-weights defining an OWA operator:

$$\text{orness}(\lambda) = \sum_{j=1}^{p} \frac{p-i}{p-1}\lambda_i.$$

The degree of orness emphasizes the higher (better) values or the lower (worse) values in a set of attributes associated with the different agents/services. Given a vector of λ weights, as closer its orness to 1, closer to the min-operator while as closer to 0, closer to the max-operator. Assuming that all the criteria are to be minimized, the min-operator allows one to generate solutions protected under worst-case scenario (pessimistic), while the max-operator produces solution in which the best situation for all the criteria is assumed (optimistic). In the middle, one can find an equilibrium between those extreme choices. In particular, for $\lambda = (\frac{1}{p}, \ldots, \frac{1}{p})$– the mean operator, its orness degree takes value 0.5. In Table 1.3 we show a list with some of the most popular OWA operators and their orness degree.

OWA	λ-vector	Operator	orness
Average	$\lambda_j = \frac{1}{p}$	$\frac{1}{p}\sum_{j\in P} W_j$	$\frac{1}{2}$
Minimum	$\lambda_1 = 1, \lambda_j = 0\ (j \geq 2)$	$\min_{j\in P} W_j$	1
k-Average	$\lambda_j = \frac{1}{k}\ (j \leq k),\ \lambda_j = 0\ (j > k)$	$\frac{1}{k}\sum_{j=1}^{k} W_{(k)}$	$1 - \frac{k-1}{2(p-1)}$
α-Min-Average	$\lambda_1 = \frac{1}{1+(p-1)\alpha},\ \lambda_j = \frac{\alpha}{1+(p-1)\alpha}\ (j \geq 2)$	$\frac{1-\alpha}{1+(p-1)\alpha}\min_{j\in P} W_j + \frac{\alpha}{1+(p-1)\alpha}\sum_{j\in P} W_j$	$\frac{-p\alpha+p+2\alpha}{2p\alpha-2\alpha+2}$
Gini	$\lambda_j = \frac{2(p-j)+1}{p^2}$ for all j	$\frac{1}{p^2}\sum_{j\in P} W_j + \frac{2}{p^2}\sum_{j\in P}(p-j)W_{(j)}$	$\frac{4p+1}{6p}$
Harmonic	$\lambda_j = \frac{1}{p}\left(H(p) - H(j-1)\right)\ (H(k) = \sum_{\ell=1}^{k}\frac{1}{\ell})$	$\frac{1}{p}\sum_{j\in P}\left(H(p) - H(j-1)\right)W_{(j)}$	$\frac{3}{4}$

Table 1.3: Some examples of OWA operators

It is clear that an OWA operator (identified with a λ-weight) is not uniquely determined by its orness degree (unless its orness degree is in $\{0, 1\}$ or $p = 2$). Thus, several optimization-based methods have been proposed in order to construct, with different paradigms, λ-weights with a given orness degree $\beta \in (0, 1)$ (see, e.g., Liu and Chen, 2004; Filev and Yager, 1995; Fullér and Majlender, 2001). The main idea when searching for λ-weights with a given orness degree is to solve problems in the form:

$$\begin{aligned} &\min \mathcal{L}(\lambda) \\ &\text{s.t. orness}(\lambda) = \beta, \\ &\lambda \in \mathbb{R}^p_+, \end{aligned}$$

where $\mathcal{L}$ is a loss function measuring some properties of the weights. For instance, if $\mathcal{L}(\lambda) = -\sum_{j=1}^{p} \lambda_j \log \lambda_j$ one obtain the maximal entropy monotone OWA (O'Hagan, 1988), or choosing $\mathcal{L}(\lambda) = \sum_{j=1}^{p}(\lambda_j - \overline{\lambda})^2$ one obtains the minimum variance weights (Fullér and Majlender, 2003), where $\overline{\lambda}$ stands for the mean of the vector λ.

This operator has been applied in the literature for other purposes such as the ordered median function introduced by Puerto and Fernández (1994) to provide a common framework for most of the classical location problems, or SAND studied by Francis et al. (2000) functions to study aggregation errors in multifacility location models.

1.3.1 Ordered Median Location Problem

The Ordered Median location problem (OMP) has been recognized as a powerful tool from a modeling point of view within the field of location analysis (Puerto and Rodríguez-Chía, 2019). This problem provides a common framework for most of the classical location problems such as the median, center, k-center, among others. The goal of the OMP is to optimize the ordered weighted average of the considered measure between the set of clients and the set of facilities to locate, once we have applied rank dependent compensation factors on them. For an extensive definition of these kind of problems see the books of Nickel and Puerto (2006) and Puerto and Rodríguez-Chía (2019).

The objective function in the OMP is called as ordered median function and it is a weighted average of ordered elements. The ordered median function is a mapping $\Phi_\lambda : \mathbb{R}^p \longrightarrow \mathbb{R}$ for some $\lambda = (\lambda_1, \ldots, \lambda_p) \in \mathbb{R}^p$. For $\mathbf{x} = (x_1, \ldots, x_p) \in \mathbb{R}^p$, the ordered median function is defined as:

$$\Phi_\lambda(x_1, \ldots, x_p) = \sum_{j=1}^{p} \lambda_j x_{(j)}, \tag{OMf}$$

where $x_{(j)}$ is the jth-largest input vector component of $\mathbf{x}$, i.e., $x_{(j)} \in \{x_1, \ldots, x_p\}$ such that $x_{(1)} \leq \ldots \leq x_{(p)}$. As it happened in the OWA operators defined by Yager, we have that ordered median functions are nonlinear functions induced by the sorting.

As we already mentioned, for different values of λ-vector lead different class of problems. If λ-vectors satisfy the requirement of the OWA operators defined by Yager, the Table 1.3 gives the same examples for ordered median functions. However, we can consider other examples like the range objective function where $\lambda = (-1, \ldots, 1)$.

This type of objective function has been successfully applied to different problems within the literature of location analysis. Examples of applications are in locating hyperplanes to fitting set of points (Blanco et al., 2021c), in segmentation of 2D and 3D Scanning-Transmission Electron Microscope (Calvino et al., 2022), covering with polyellpisoids (Blanco and Puerto, 2021a), in hub location problems (Puerto et al., 2011, 2016), in p-median problems both discrete (Deleplanque et al., 2020; Marín et al., 2020) and continuous (Blanco et al., 2014, 2016), in locating facilities with neighborhoods (Blanco, 2019), among others. See Nickel and Puerto (2006) and Puerto and Rodríguez-Chía (2019) for other applications of ordered median functions.

1.3.2 Representation of the sorting

As mentioned above, both OWA operators and ordered median functions are nonlinear objective functions induced by the sorting of the vector to which it is applied. The literature is rich in attempts to represent this sorting in mathematical programming models. In this section we present the best known and most used in a general way to represent the sorting of the vector in a model.

Let (1.8) be a general model which includes any of the operators for a defined vector $\mathbf{x}$ in some domain $\mathcal{D}(\mathbf{x})$.

$$\begin{aligned} \max_{\mathbf{x}} \ & \Phi_\lambda(\mathbf{x}) = \sum_{j=1}^{p} \lambda_j x_{(j)} \\ \text{s.t. } & \mathbf{x} \in \mathcal{D}(\mathbf{x}). \end{aligned} \tag{1.8}$$

The value of $\mathbf{x}$ can represent a vector of real numbers, a variable which define the problem or a result given by some relation between variables like, for example, the value of variable D_{ij} in the problem (MFLP).

For any value of λ-vector, the book by Domínguez-Marín (2003) gives different models and solution methods for the discrete ordered median problem. Ordering the values of a vector is equivalent to find a permutation providing the correct order. Since any permutation can be represented by an assignment problem, the sorting can be formulated as an Integer Linear Program (ILP), adding some additional constraints to obtain the correct permutation. To this aim, given a permutation $\sigma \in \mathcal{P}(\{1, \ldots, p\})$, the following binary variables are defined,

$$s_{jk} = \begin{cases} 1, & \text{if } \sigma(j) = k, \text{ i.e., if } x_j \text{ is the } k\text{th smallest value,} \\ 0, & \text{otherwise,} \end{cases} \quad \text{for all } \ j, k \in \{1, \ldots, p\}.$$

Then, the (1.8) can be reformulated on the following ILP formulation,

$$
\begin{aligned}
\max \ & \sum_{j=1}^{p} \lambda_j s_{jk} x_j \\
\text{s.t.} \ & \sum_{j=1}^{p} s_{jk} = 1, \ \forall k \in \{1, \ldots, p\}, \\
& \sum_{k=1}^{p} s_{jk} = 1, \ \forall j \in \{1, \ldots, p\}, \\
& \sum_{k=1}^{p} s_{jk} x_j \leq \sum_{k=1}^{p} s_{j+1k} x_j, \ \forall j \in \{1, \ldots, p-1\}, \\
& s_{jk} \in \{0, 1\}, \ \forall j, k \in \{1, \ldots, p\}, \\
& \mathbf{x} \in \mathcal{D}(\mathbf{x}),
\end{aligned}
\tag{2I}
$$

where the first family of constraints ensures that each j is placed at only one position, and the second set of constraints assures each position is assigned to a single coordinate of the vector x. Finally, the third family guarantee the non-decreasing order of the sorted real numbers. The last two sets of constrains are the domain of the variables. Note that the product of variable s with the vector $\mathbf{x}$ could be the product between two variables if the vector represents a variable for the problem. This can be linearized by reformulations presented in the literature (see, e.g, Domínguez-Marín, 2003, for further details on reformulations).

In case the λ-vector is monotone other formulations have been proposed. Ogryczak and Tamir (2003) provide a suitable linear programming representation of the problem of minimizing the sum of the k largest (equivalently, smallest) linear functions on a polyhedral set in $\mathbb{R}^d$. This representation is extended to the minimization of monotone OWA functions by means of a telescopic sum of k-sum functions.

If we use $\lambda = (\lambda_1, \ldots, \lambda_p)$ such that $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_p$, the (**OT**) reformulation given by Ogryczak and Tamir (2003) is,

$$
\begin{aligned}
\max \ & \sum_{k=1}^{p-1} (\lambda_k - \lambda_{k+1}) \left(k t_k - \sum_{j=1}^{p} z_{jk}^{+} \right) \\
\text{s.t.} \ & z_{jk}^{+} \geq t_k - x_j, \ \forall j, k \in \{1, \ldots, p\}, \\
& z_{jk}^{+} \geq 0, \ \forall j, k \in \{1, \ldots, p\}, \\
& t_k \in \mathbb{R}, \ \forall k \in \{1, \ldots, p\}, \\
& x_j \in \mathcal{D}(\mathbf{x}), \ \forall j \in \{1, \ldots, p\}.
\end{aligned}
\tag{OT}
$$

Also for the same monotone λ-vector, Blanco et al. (2014) provide a linear programming formulation of the problem. This formulation is based on an assignment problem whose dual problem allows to compute the value of the ordered median function.

$$\max \sum_{j=1}^{p} u_j - \sum_{j=1}^{p} v_k \tag{BEP}$$
$$\text{s.t. } u_j - v_k \leq \lambda_k x_j, \ \forall j, k \in \{1, \dots, p\},$$
$$u_j, v_k \geq 0, \ \forall j, k \in \{1, \dots, p\},$$
$$x_j \in \mathcal{D}(\mathbf{x}), \ \forall j \in \{1, \dots, p\}.$$

Both (**2I**) and (**BEP**) are based on reformulating the sorting problem as an assignment problem, but the difference is the auxiliary variables to represent the sorting.

The reader can observe that while the general λ-case results in a non–concave problem, the use of monotone weights, $\lambda_1 \geq \cdots \geq \lambda_p$, gives rise to ordered median function is concave (see proposition 1.1 in Nickel and Puerto (2006). In that proposition, the lambdas are defined as non-increasing monotones, and thus, the ordered median function is convex).

1.4 Fairness

The term *fairness* is defined as "the quality of treating people equally or in a way that is right or reasonable" (Cambridge Dictionary). It is an abstract but widely studied concept in Decision Sciences in which some type of indivisible resources are to be shared among different agents. The importance of fairness issues in resource allocation problems has been recognized and well studied in a variety of settings with tons of applications in different fields (see e.g., Jiang et al., 2021; Kelly et al., 1998; Li and Li, 2006; Luss, 1999). Fair solutions should imply impartiality, justice and equity allocation patterns, which are usually quantified by means of inequality measures that are minimized. Several measures have been proposed in the literature to this end, although the most popular one is the max-min (or min-max) approach which assures that the most *damaged* agent in the share is as less damaged as possible (see e.g., Bertsekas et al., 1992; Hayden, 1981; Jaffe, 1981; Megiddo, 1974). Other proposals of fairness measures are the minimum envy (Caragiannis et al., 2009; Espejo et al., 2009; Lipton et al., 2004; Netzer et al., 2016) or certain families of ordered weighted averaging criteria (Hurkała and Hurkała, 2013; Ogryczak et al., 2014; Ogryczak and Trzaskalik, 2006), among others.

This section briefly introduces the concept of fairness, its applications in various fields and some of the proposed measures to quantify the fairness of the solution of a mathematical optimization problem.

In many resource reallocation problems there has been special attention to the notion of fairness. For this reason, that importance has been widely recognized in the literature and has been well studied by the academic research in a wide variety of ways. These applications of fairness range from social or humanitarian contexts to engineering applications or location problems.

The social context is an important field where the fairness is used. Gross (2008) highlighted the importance of studying these theoretical concepts in the literature of equity and justice to apply them to real contexts such as the fair allocation of water for irrigation farms. Huang and Rafiei (2019) consider the notion of fairness in the context of humanitarian resources and summarize the related literature.

Furthermore, fairness has been widely analyzed in communication networks where it is usual to share limited resources among a large number of users (see e.g., Bonald and Massoulié, 2001; Luss, 1999; Kleinberg et al., 1999; Ogryczak et al., 2014).

The allocation of public resources is one of the most known application for equity and fairness. One can find applications in emergency medical services resources such as the allocation of beds among patients or ambulances in emergency calls. In (Leclerc et al., 2012, Chapter 4), the authors collect certain measurements from the literature and study them in this particular application, the equity allocation of ambulances. Another application of equity allocation is in the healthcare scheduling, where beds and other resources should be fairly allocated to reduce the mortality (see e.g, Zhou et al., 2020).

In the context of location, concretely, in the public sector, equity and fairness is crucial. However, one can find few papers incorporating fairness in facility location (see e.g., Chanta et al., 2014; Espejo et al., 2009). A review of measures for equity facility location problems was early done by Marsh and Schilling (1994) and recent done by Barbati and Piccolo (2016).

However, due to the fuzzy nature of notion of fairness and different possible interpretations of equity, there is no principle that is universally accepted as “the most fair” (Bertsimas et al., 2012). One approach to quantify the degree of fairness associated with a solution of any problem is through a fairness measure. This fairness measure is a function that maps the solution into a real number. Therefore, different measures have been proposed in the literature to this end together with an axiomatic theory of desired properties of these operators.

The most popular one is the max-min (or min-max) approach which assures that the most *damaged* agent in the share is as less damaged as possible (see e.g., Jiang et al., 2021; Kelly et al., 1998; Li and Li, 2006; Luss, 1999; Megiddo, 1974). The max-min ratio is given by the maximum ratio of any two user’s resource allocation.

Other proposals of fairness measures are the minimum envy which is a measure that considers the differences in service quality between all possible pairs of agents (see e.g., Caragiannis et al., 2009; Chanta et al., 2014; Espejo et al., 2009; Lipton et al., 2004; Netzer et al., 2016).

Jain et al. (1984) proposed a different operator to measure the *equality* of an allocation, that is, if all users get the same amount of resources. The Jain’s index takes value 1 for the most fair allocation pattern while it takes value 0 for the less fairer one. If a system allocates resources to p agents and we denote as $\mathbf{x} \in \mathbb{R}^p$ the vector of resource allocation between the p agents such as x_j is the allocation of resources for the agent j, the Jain's index is defined as:

$$\Psi(\mathbf{x}) = \frac{\left(\sum_{j=1}^{p} x_j\right)^2}{\sum_{j=1}^{p} x_j^2}, \; x_j \geq 0 \; \forall j \in \{1, \ldots, p\}. \tag{J-Index}$$

This index is also generalized by Lan et al. (2010). In that work, they developed a method to construct fairness schemes based on some axioms. The authors proposed generalized Jain's index using power functions.

An alternative approach that is the α-fairness approach that has become popular in the last two decades (Bertsimas et al., 2011, 2012; Kelly et al., 1998; Lan et al., 2010; Mo and Walrand, 2000, see e.g.,). The α-fairness measure was early introduced by Atkinson (1970) and according to him, this function maximizes the constant elasticity social welfare. Again, for a vector $\mathbf{x} \in \mathbb{R}^p$ of resource allocation between p agents, the α-fairness operator for $\alpha \geq 0$ is defined as

$$\Psi_\alpha(\mathbf{x}) = \begin{cases} \frac{1}{1-\alpha} \sum_{j=1}^{p} x_j^{1-\alpha} & \text{if } \alpha \geq 0, \, \alpha \neq 1, \\ \sum_{j=1}^{p} \log(x_j) & \text{if } \alpha = 1. \end{cases} \tag{α-fairness}$$

The parameter α is known as the *inequality aversion parameter* since it controls the rate of the difference between the allocation resources x_j. Consider an agent k with lower amount of resources x_k than the agent k' with amount of resources $x_{k'}$. If we increase the resources in x_k then we would have a higher welfare than if we increase in $x_{k'}$. Thus, an increase in the resources of x_k would be more desirable to reduce the unfairness in the allocation of the resources. This can be performed by tuning, adequately, the parameter α since when it increases, the difference between the values of resources for the agents decreases (for a constructive proof of this, see Lan et al., 2010). This property, also known Principle of transfer or Pigou-Dalton (Erkut, 1993), yielding then to fair solutions.

For different values of $\alpha \geq 0$, we get different measures. For instance, for $\alpha = 0$ we get the utilitarian principle, which is neutral toward inequalities; for $\alpha = 1$ the scheme corresponds to proportional fairness (introduced by Nash, 1950); and when $\alpha \to \infty$ the allocation converges to the max-min fairness.

The above mentioned fairness measures allow the decision maker to focus the objective of an allocation problem to different fairness schemes. In the literature, this approach can be done modeling as a multiobjective mathematical programming problem, which are usually hard to solve (Ehrgott and Gandibleux, 2000). Also, the goal of finding solutions satisfying all the decision makers is usually unattainable in practice, and some services must sacrifice their resources for the benefit of others. Thus, it is widely accepted that the use of aggregation functions may yield compromise solutions for the different criteria, and then, an adequate procedure to provide solutions to hard multiobjective problems. OWA operators defined in Section 1.3 have been also used in the literature to guide the optimization problem to obtain fairer solutions. For instance, Ogryczak and Trzaskalik (2006) show that fairly efficient solution of LP-based resource allocation problem can be identified with an OWA optimal solution with appropriate strictly monotonic weights; Hurkała and Hurkała (2013) present a multiple types of facilities and the authors want to provide fairness solutions maximizing the production; and Ogryczak et al. (2014) review fair optimization models and methods for the location problems and for the resource allocation problems in communication networks.

Chapter 2

A branch-and-price approach for the continuous multifacility monotone ordered median problem

In this chapter, we address the Continuous Multifacility Monotone Ordered Median Problem. The goal of this problem is to locate p facilities in $\mathbb{R}^d$ minimizing a monotone ordered weighted median function of the distances between given demand points and its closest facility. We propose a new branch-and-price procedure for this problem, and three families of matheuristics based on: solving heuristically the pricer problem, aggregating the demand points, and discretizing the decision space. We give detailed discussions of the validity of the exact formulations and also specify the implementation details of all the solution procedures. Besides, we assess their performance in an extensive computational experience that shows the superiority of the branch-and-price approach over the compact formulation in medium-sized instances. To handle larger instances it is advisable to resort to the matheuristics that also report rather good results.

2.1 Introduction

Motivated by the recent advances on discrete multifacility location problems with ordered median objectives (Deleplanque et al., 2020; Espejo et al., 2021; Fernández et al., 2014; Labbé et al., 2017; Marín et al., 2020), and the available results on conic optimization (Blanco et al., 2014; Puerto, 2020), we analyze here a family of difficult continuous multifacility location problems with ordered median objectives and distances induced by a general family of norms. These problems gather the essential elements of discrete and continuous location analysis, making their solution a challenging question.

In this chapter, we develop an *ad hoc* branch-and-price algorithm for solving this general family of continuous location problems. The continuous multifacility Weber problem has been already studied using branch-and-price methods (Krau, 1997; Du Merle et al., 1999; Righini and Zaniboni, 2007; Venkateshan and Mathur, 2015). In addition, in discrete location, these techniques have also been applied to the p-median problem (see, e.g., Avella et al., 2007). However a branch-and-price approach for location problems with ordered median objectives has been only developed for the discrete version by Deleplanque et al. (2020) beyond a multisource hyperplanes application (Blanco et al., 2021c).

Our goal is to analyze the *Continuous Multifacility Monotone Ordered Median Problem* (MFMOMP, for short), in which we are given a finite set of demand points, $\mathcal{A}$, and the goal is to find the optimal location of p new facilities such that: (1) each demand point is allocated to a single facility; and (2) the measure of the goodness of the solution is an ordered weighted aggregation of the distances of the demand points to their closest facility (see, e.g., Nickel and Puerto, 2006). We consider a general framework for the problem, in which the demand points (and the new facilities) lie in $\mathbb{S} = \mathbb{R}^d$, the distances between points and facilities are polyhedral- or ℓ_τ-norms for $\tau \geq 1$, and the ordered median functions are assumed to be defined by non-decreasing monotone weights. These problems are analyzed in Blanco et al. (2016), in which the authors provide a Mixed Integer Second

Order Cone Optimization (MISOCO) reformulation of the problem able to solve, for the first time, problems of small to medium size (up to 50 demand points), using off-the-shell solvers.

The family of problems under analysis has a broad range of applications in different fields. On the one hand, continuous location has been proven to be an adequate tool in case the services to be located are sensors, surveillance cameras, etc., that are allowed to be flexibly positioned in the space. Also, multifacility location problems can be seen as a unified modelling tool to extend classical clustering algorithms, as the k-means or k-median approaches, or more general approaches (Blanco et al., 2021c). The use of ordered median objective functions determines, at the same time, the positions of the *optimal* location of the services balancing equity and efficiency of the list of distances from the demand points to their closest facilities (see e.g. Aouad and Segev, 2019; Calvino et al., 2022; Espejo et al., 2009; Fourour and Lebbah, 2020; Muñoz-Ocaña et al., 2020; Ogryczak et al., 2011; Olender and Ogryczak, 2019; Tamir, 2001). The connection between discrete location and its continuous counterpart has been a topic of study since the introduction of the continuous problem (Cooper, 1963; Kalczynski et al., 2021). Thus, the extension of some facility location problems that have been analyzed in a discrete space (voting, exam qualifications, etc.) to the continuous framework, is a topic of interest in the Location Science field (Drezner and Nickel, 2009; Espejo et al., 2009; Ponce et al., 2018). We also refer the reader to Bruno et al. (2014); Drezner and Hamacher (2004); Love et al. (1988); Mirchandani and Francis (1990); and the references therein to find more applications in the fields of industry, urban or regional planning, clustering, mobile location, commerce, public service facilities, or transport facilities.

Our contribution is to introduce a new set partitioning-like (with side constraints) reformulation for this family of problems that allows us to develop a branch-and-price algorithm for solving it. This approach gives rise to a decomposition of the original problem into a master problem (set partitioning with side constraints), and a pricing problem that consists of a special form of the maximal weighted independent set problem combined with a single facility location problem. We compare this new strategy with the one obtained by solving Mixed Integer Non-Linear Programming (MINLP) formulations using standard solvers. Our results show that it is worth to use the new reformulation since it allows us to solve larger instances and reduce the gap when the time limit is reached. Moreover, we also exploit the structure of the branch-and-price approach to develop some new matheuristics for the problem that provide good quality feasible solutions for fairly large instances of several hundreds of demand points.

The chapter is organized in six sections and two appendixes. Section 2.2 formally describes the problem considered in this chapter, namely the MFMOMP, and develops MISOCO formulations for it. Section 2.3 is devoted to present the new set partitioning-like formulation and all the details of the branch-and-price algorithm proposed to solve it. There, we present how to obtain initial variables for the restricted master problem, we discuss and formulate the pricing problem and set properties for handling it, and describe the branching strategies and variable selection rules implemented in our algorithm. The next section, namely Section 2.4, deals with some heuristic algorithms proposed to provide solutions for large-sized instances. In this section, we also describe how to solve heuristically the pricing problem which gives rise to a matheuristic algorithm consisting of applying the branch-and-price algorithm but solving the pricing problem only heuristically. Obviously, since in this case the optimality of the pricing problem is not guaranteed, we cannot ensure optimality for the solution of the master problem, although we always obtain feasible solutions. In addition, we also present two heuristics more: the aggregation heuristic based on clustering strategies that allows us to provide bounds for the problem, and the discretization heuristic based on discretizing the space to become it in a discrete p-median problem. Section 2.5 reports the results of an exhaustive computational study with real-world instances of different nature. There, we compare the standard formulations with the branch-and-price approach and also with the heuristic algorithms. Finally, 2.6 reports the details of the computational experiment for different norms showing the usefulness and generality of our approach, and 2.7 shows the computational results disaggregated by different parameters of the instances. The chapter ends with some conclusions in Section 2.8.

2.2 The Continuous Multifacility Monotone Ordered Median Problem

In this section, we describe the problem under study and fix the notation for the rest of the chapter. We are given a metric space $\mathbb{S} = \mathbb{R}^d$ with a metric $\|\cdot\|$ associated, a set of n demand points in $\mathbb{R}^d$, $\mathcal{A} = \{a_1, \ldots, a_n\} \subset \mathbb{R}^d$, and $p \in \mathbb{N}$ ($p > 0$). Our goal is to find p new facilities located in $\mathbb{R}^d$ that minimize a function of the closest distances from the demand points to the new facilities. We denote the index sets of demand points and facilities by $N = \{1, \ldots, n\}$ and $P = \{1, \ldots, p\}$, respectively. Several elements are involved when finding the *best* p new facilities to provide service to the n demand points. In what follows, we describe them:

- *Closeness Measure:* Given a demand point a_i, $i \in N$, and a facility $x \in \mathbb{R}^d$, we use norm-based distances to measure the point-to-facility closeness. Thus, we consider the following measure for the distance between a_i and x:

$$\delta_i(x) = \|a_i - x\|,$$

  where $\|\cdot\|$ is a polyhedral- or an ℓ_τ-norm (with $\tau \geq 1$).

- *Allocation Rule*: Given a set of p new facilities, $\mathcal{X} = \{x_1, \ldots, x_p\} \subset \mathbb{R}^d$, and a demand point a_i, $i \in N$, once all the distances between a_i and x_j ($j \in P$) are calculated, one has to allocate the point to a single facility. As usual in the literature, we assume that each point is allocated to its closest facility, i.e., the closeness measure between a_i and $\mathcal{X}$ is:
$$\delta_i(\mathcal{X}) = \min_{x \in \mathcal{X}} \delta_i(x),$$
and the facility $x \in \mathcal{X}$, reaching such a minimum is the one where a_i is allocated to (in case of ties among facilities, a random assignment is performed).

- *Aggregation of Distances*: Given the set of demand points $\mathcal{A}$, the distances $\{\delta_i(\mathcal{X}) : i \in N\} = \{\delta_1, \ldots, \delta_n\}$ must be aggregated (abusing of notation, and unless necessary, we will avoid the dependence of $\mathcal{X}$ in the δ-values). To this end, we use the family of ordered median criteria. Given $\lambda \in \mathbb{R}^n_+$ the λ-ordered median function is defined as:
$$\mathsf{OM}_\lambda(\mathcal{A}; \mathcal{X}) = \sum_{i \in N} \lambda_i \, \delta_{(i)}, \tag{OM}$$
where $(\delta_{(i)})_{(i \in N)}$ is a permutation of $(\delta_i)_{(i \in N)}$ such that $\delta_{(1)} \leq \cdots \leq \delta_{(n)}$. Some particular choices of λ-weights are shown in Table 2.1. Note that most of the classical continuous location problems can be cast under this *ordered median* framework, e.g., the multisource Weber problem, $\lambda = (1, \ldots, 1)$, or the multisource p-center problem, $\lambda = (0, \ldots, 0, 1)$.

Summarizing all the above considerations, given a set of n demand points in $\mathbb{R}^d$, $\mathcal{A} = \{a_1, \ldots, a_n\} \subset \mathbb{R}^d$ and $\lambda \in \mathbb{R}^n_+$ (with $0 \leq \lambda_1 \leq \cdots \leq \lambda_n$), the Continuous Multifacility Monotone Ordered Median Problem (**MFMOMP**$_\lambda$) can be stated as:

$$\min_{\mathcal{X} = \{x_1, \ldots, x_p\} \subset \mathbb{R}^d} \mathsf{OM}_\lambda(\mathcal{A}; \mathcal{X}). \tag{$\mathbf{MFMOMP}_\lambda$}$$

Observe that the problem above is NP-hard since the multisource p-median problem is just a particular instance of (**MFMOMP**$_\lambda$) where $\lambda = (1, \ldots, 1)$ (see Sherali and Nordai, 1988). In the following result we provide a suitable Mixed Integer Second Order Cone Optimization (MISOCO) formulation for the problem.

Theorem 5. *Let $\| \cdot \|$ be an ℓ_τ-norm in $\mathbb{R}^d$, where $\tau = \frac{r}{s}$ with $r, s \in \mathbb{N} \setminus \{0\}$, $r > s$ and $\gcd(r, s) = 1$ or a polyhedral norm. Then, (**MFMOMP**$_\lambda$) can be formulated as a MISOCO problem.*

Proof. First, assume that $\{\delta_i(\mathcal{X}) : i \in N\} = \{\delta_1, \ldots, \delta_n\}$ are given. Then, sorting the elements and multiplying them by the λ-weights can be equivalently written as the following assignment problem (see Blanco et al., 2014, 2016), whose dual problem (right side) allows to compute the value of the ordered median function:

$$\sum_{k\in N}\lambda_k\delta_{(k)} = \max \sum_{i,k\in N}\lambda_k\delta_i\sigma_{ik} \quad = \quad \min \sum_{i\in N}u_i + \sum_{k\in N}v_k$$
$$\text{s.t.} \sum_{k\in N}\sigma_{ik} = 1,\ \forall i\in N, \qquad \text{s.t.}\ u_i + v_k \geq \lambda_k\delta_i,\ \forall i,k\in N,$$
$$\sum_{i\in N}\sigma_{ik} = 1,\ \forall k\in N, \qquad u_i, v_k \in \mathbb{R},\ \forall i,k\in N.$$
$$\sigma_{ik}\in[0,1],\ \forall i,k\in N.$$

The formulation is similar to the represented in Section 1.3.2 given by (**BEP**), although in this case we have a minimization problem with non-increasing monotone vector of λ.

Now, we can embed the above representation of the ordered median aggregation of $\delta_1,\dots,\delta_n$, into (**MFMOMP**$_\lambda$). On the other hand, we have to represent the allocation rule (closest distances). This family of constraints is given by

$$\delta_i = \min_{j\in P}\|a_i - x_j\|,\ \forall i\in N.$$

In order to represent it, we use the following set of decision variables: $w_{ij}=1$ if a_i is allocated to facility j, $w_{ij}=0$ otherwise, $\forall i\in N, j\in P$; in addition, z- and r-variables are auxiliary variables.

Then, a *Compact* formulation for (**MFMOMP**$_\lambda$) is:

$$\min \sum_{i\in N}u_i + \sum_{k\in N}v_k$$
$$\text{s.t.}\ u_i + v_k \geq \lambda_k r_i,\ \forall i,k\in N, \tag{C$_1$}$$
$$z_{ij} \geq \|a_i - x_j\|,\ \forall i\in N, j\in P, \tag{C$_2$}$$
$$r_i \geq z_{ij} - M(1-w_{ij}),\ \forall i\in N, j\in P, \tag{C$_3$}$$
$$\sum_{j\in P}w_{ij} = 1,\ \forall i\in N, \tag{C$_4$}$$
$$x_j\in\mathbb{R}^d,\ \forall j\in P,$$
$$w_{ij}\in\{0,1\},\ \forall i\in N, j\in P,$$
$$z_{ij}\geq 0,\ \forall i\in N, j\in P,$$
$$r_i\geq 0,\ \forall i\in N,$$

where (C$_3$) allows to compute the distance between the points and its closest facility and (C$_4$) assures single allocation of points to facilities. Here M is a big enough constant $M > \max_{i,k\in N}\|a_i - a_k\|$.

Finally, in case $\|\cdot\|$ is the $\ell_{\frac{r}{s}}$-norm, constraint (C$_2$) can be rewritten as the set of constraints given in (ℓ_τ-norm), or in case the norm is a polyhedral norm can be rewritten as the family of constraints given by (Pol-norm). Therefore, the final compact formulation is a MISOCO reformulation for (**MFMOMP**$_\lambda$). □

Note that (**MFMOMP**$_\lambda$) is an extension of the single-facility ordered median location problem (see, e.g., Blanco et al., 2014), where apart from finding the location of p new facilities, the allocation patterns between demand points and facilities are also determined. In the rest of the chapter, we will exploit the combinatorial nature of the problem by means of a set partitioning-like formulation which is based on the following observation:

Proposition 1. *Any optimal solution of* (**MFMOMP**$_\lambda$) *is characterized by p pairs $(S_1, x_1), \dots, (S_p, x_p)$ with $S_j \subset N$ and $x_j \in \mathbb{R}^d$, $\forall j \in P$, such that:*

1. $\bigcup_{j \in P} S_j = N$.
2. $S_j \cap S_{j'} = \emptyset$, $j, j' \in P : j \neq j'$.
3. *For each* $j \in P$, $x_j \in \arg \min\limits_{x \in \{x_1,\dots,x_p\}} \|a_i - x\|$, $\forall i \in S_j$.
4. $(x_1, \dots, x_p) \in \arg \min\limits_{y_1,\dots,y_p} \sum\limits_{j \in P} \sum\limits_{i \in S_j} \lambda_{(i)} \|a_i - y_j\|$, *where* $(i) \in N$ *such that* $\|a_i - y_j\|$ *is the* (i)*-th smallest element in* $\{\|a_i - y_j\| : i \in N, j \in P\}$.

From the structure of the optimal solutions of (**MFMOMP**$_\lambda$) described in Proposition 1, we can conclude that there exists a finite candidate set of admissible solutions of this problem given by the different partitions of N in p subsets and one of their associated p best facilities, as defined in Proposition 1 (4). In addition, if the demand points $\mathcal{A}$ are non-collinear and $\tau > 1$ the solution of the problem in Proposition 1 (4) is unique; otherwise, we can always restrict the choices of $x_1, \dots, x_p$ to the extreme points of the set of optimal solutions which is finite. From the above discussion, we conclude that there exists a finite dominating set of candidates, that we will denote as $\mathcal{FDS}$, to optimal solutions of (**MFMOMP**$_\lambda$).

From now on, we will call a pair (S, x) with $S \subset N$ and $x \in \mathbb{R}^d$ a *suitable pair if*

1. There exist $(S_2, x_2), \dots, (S_p, x_p)$ such that $\bigcup\limits_{j=2}^{p} S_j = N \setminus S$, $S_j \cap S_{j'} = \emptyset$ for $j, j' \in \{2, \dots, p\} : j \neq j'$, $x_j \in \mathbb{R}^d$, $j = 2, \dots, p$.
2. $(S, x), (S_2, x_2), \dots, (S_p, x_p) \in \mathcal{FDS}$.

In words, a suitable pair is any pair (S, x) that can be part of a candidate solution of (**MFMOMP**$_\lambda$) within the set $\mathcal{FDS}$. By the finiteness of the sets of admissible solutions, it also follows that the number of suitable pairs is finite as well.

2.3 A set partitioning-like formulation

The compact formulation shown in the previous section is affected by the size of p and d, and it exhibits the same limitations as many other compact formulations for continuous location models even without ordering constraints. For this reason, in the following we propose an alternative set partitioning-like formulation (Du Merle et al., 1999; du Merle and Vial, 2002) for (**MFMOMP**$_\lambda$).

Let $S \subset N$ be a subset of demand points that are assigned to the same facility. Let $R = (S, x)$ be a suitable pair composed by a subset $S \subset N$ and a facility $x \in \mathbb{R}^d$. We denote by δ_i^R the contribution of demand point $i \in S$ in the subset with respect to the facility x. Finally, for each suitable pair $R = (S, x)$ we define the variable

$$y_R = \begin{cases} 1 & \text{if subset } S \text{ is selected and its associated facility is } x, \\ 0 & \text{otherwise.} \end{cases}$$

We denote by $\mathcal{R} = \{(S, x) : \text{suitable pairs}, S \subset N \text{ and } x \in \mathbb{R}^d\}$.

The set partitioning-like formulation is

$$\min \quad \sum_{i \in N} u_i + \sum_{k \in N} v_k \tag{MP$_1$}$$

$$\text{s.t.} \quad \sum_{R=(S,x) \in \mathcal{R} : i \in S} y_R = 1, \ \forall\, i \in N, \tag{MP$_2$}$$

$$\sum_{R \in \mathcal{R}} y_R = p, \tag{MP$_3$}$$

$$u_i + v_k \geq \lambda_k \sum_{R=(S,x) \in \mathcal{R} : i \in S} \delta_i^R y_R, \ \forall\, i, k \in N, \tag{MP$_4$}$$

$$y_R \in \{0, 1\}, \ \forall\, R \in \mathcal{R}, \tag{MP$_5$}$$

$$u_i, v_k \in \mathbb{R}, \ \forall i, k \in N. \tag{MP$_6$}$$

The objective function (MP$_1$) and constraints (MP$_4$) give the correct ordered median function of the distances from the demand points to the closest facility (see Section 2.2). Constraints (MP$_2$) ensure that all demand points appear in exactly one set S in each feasible solution. Exactly p facilities are open due to constraint (MP$_3$). Finally, (MP$_5$) define the variables as binary.

The reader might notice that this formulation has an exponential number of variables, and therefore in the following we describe the necessary elements to address its solution by means of a branch-and-price scheme, namely:

1. *Initial Pool of Variables:* Generation of initial feasible solutions induced by a set of initial subsets of demand points (and their costs).

2. *Pricing Problem:* In set partitioning problems, instead of solving initially the problem

with the whole set of variables, the variables have to be incorporated *on-the-fly* by solving adequate pricing subproblems derived from previously computed solutions until the optimality of the solution is guaranteed. The pricing problem is derived from the relaxed version of the master problem and using the strong duality properties of the induced Linear Programming Problem.

3. *Branching:* The rule that creates new nodes of the branch-and-bound tree when a fractional solution is found at a node of the search tree. We have adapted the Ryan and Foster branching rule to our problem.

4. *Stabilization*: The convergence of column generation approaches can be sometimes erratic since the values of dual variables in the first iterations might oscillate, leading to variables of the master problem that will never appear in the optimal solution of the problem. Stabilization tries to avoid that behavior.

In what follows, we describe how each of the above items is implemented in our proposal.

2.3.1 Initial variables

In the solution process of the set partitioning-like formulation using a branch-and-price approach, it is convenient to generate an initial pool of variables before starting solving the problem. The adequate selection of these initial variables might help to reduce the CPU time required to solve the problem. We apply an iterative strategy to generate this initial pool of y-variables. In the first iteration, we randomly generate p positions for the facilities. The demand points are then allocated to their closest facilities, and at most p subsets of demand points are generated. We incorporate the variables associated with these subsets to the master problem (MP). In the next iterations, instead of generating p new facilities, we keep those with more associated demand points, and randomly generate the remainder. After a fixed number of iterations, an initial set of columns is generated to define the restricted master problem, and also an upper bound of our problem. Since the optimal position of the facilities belongs to a bounded set contained in the rectangular hull of the demand points, the random facilities are generated in the smallest hyperrectangle containing $\mathcal{A}$.

2.3.2 The pricing problem

To apply the column generation procedure we define the restricted relaxed master of (MP), in the following (RRMP).

$$\rho^*_{\text{MP}} := \min \quad \sum_{i\in N} u_i + \sum_{k\in N} v_k \qquad \textbf{Dual Multipliers} \tag{RRMP}$$

$$\begin{aligned}
\text{s.t.} \quad & \sum_{R=(S,x)\in\mathcal{R}: i\in S} y_R \geq 1, \ \forall\, i \in N, && \alpha_i \geq 0\\
& -\sum_{R\in\mathcal{R}_0} y_R \geq -p, && \gamma \geq 0\\
& u_i + v_k - \lambda_k \sum_{R=(S,x)\in\mathcal{R}: i\in S} \delta_i^R y_R \geq 0, \ \forall\, i,k \in N, && \epsilon_{ik} \geq 0\\
& y_R \geq 0, \ \forall\, R \in \mathcal{R}_0,\\
& u_i, v_k \in \mathbb{R}, \ \forall i,k \in N,
\end{aligned}$$

where $\mathcal{R}_0 \in \mathcal{R}$ represents the initial pool of columns used to initialize the set partitioning-like formulation (MP). Constraints (MP$_2$) and (MP$_3$) are slightly modified from equations to inequalities in order to get nonnegative dual multipliers. This transformations keeps the equivalence with the original formulation since coefficients affecting the y-variables in constraint (MP$_4$) are nonnegative. The notation for the dual variables associated with each family of constraints is written in the right column $(\alpha, \gamma, \epsilon)$.

The value of the distances is unknown beforehand because the location of facilities can be anywhere in the continuous space. Hence, its determination requires solving continuous optimization problems.

By strong duality, the objective value of the continuous relaxation (RRMP), can be obtained as:

$$\begin{aligned}
\rho^*_{\text{MP}} = \max \quad & \sum_{i\in N} \alpha_i - p\gamma \qquad\qquad \text{(Dual RRMP)}\\
\text{s.t.} \quad & \sum_{i\in N} \epsilon_{ik} = 1, \ \forall\, k \in N,\\
& \sum_{k\in N} \epsilon_{ik} = 1, \ \forall\, i \in N,\\
& \sum_{i\in S} \alpha_i - \gamma - \sum_{i\in S}\sum_{k\in N} \delta_i^R \lambda_k \epsilon_{ik} \leq 0, \ \forall\, R = (S,x) \in \mathcal{R}_0,\\
& \alpha_i, \gamma, \epsilon_{ik} \geq 0, \ \forall\, i,k \in N.
\end{aligned}$$

Hence, for any variable y_R in the master problem, its reduced cost is

$$c_R - z_R = -\sum_{i\in S} \alpha_i^* + \gamma^* + \sum_{i\in S}\sum_{k\in N} \delta_i^R \lambda_k \epsilon_{ik}^*,$$

where $(\alpha^*, \gamma^*, \epsilon^*)$ is the dual optimal solution of the current (RRMP).

To certify optimality of the relaxed problem one has to check implicitly that all the reduced costs for the variables not currently included in the (RRMP) are nonnegative. Otherwise, new variables must be added to the pool of columns. This can be done solving the so-called pricing problem.

The pricing problem consists of finding the minimum reduced cost among the variables that have not yet been included in the pool. That is, we have to find the set $S \subset N$ and the position of the facility x (its coordinates) which minimizes the reduced cost.

For a given set of dual multipliers, $(\alpha^*, \gamma^*, \epsilon^*) \geq 0$, the problem to be solved is

$$\begin{aligned} \min_{\substack{S\subset N\\ x\in\mathbb{R}^d}} \quad & -\sum_{i\in S} \alpha_i^* + \gamma^* + \sum_{i\in S}\sum_{k\in N} \delta_i^S \lambda_k \epsilon_{ik}^* \\ \text{s.t.} \quad & \delta_i^S \geq \|x - a_i\|, \ \forall i \in S. \end{aligned}$$

The above formal problem can be reformulated by means of a mixed integer program. We define variables $w_i = 1, \ i \in N$ if the demand point belongs to S, and zero otherwise. We also define variables $r_i, \ i \in N$ to represent the distance from demand point i to facility x and zero in case $w_i = 0$. Finally, $z_i, \ i \in N$ are auxiliary variables to represent the distances from demand point i to facility x in any case.

$$\min \quad -\sum_{i\in N} \alpha_i^* w_i + \gamma^* + \sum_{i\in N} c_i r_i \tag{2.3}$$

$$\text{s.t.} \quad z_i \geq \|x - a_i\|, \ \forall i \in N, \tag{2.4}$$

$$r_i + M(1 - w_i) \geq z_i, \ \forall i \in N, \tag{2.5}$$

$$w_i \in \{0, 1\}, \ \forall i \in N, \tag{2.6}$$

$$z_i, r_i \geq 0, \ \forall i \in N, \tag{2.7}$$

where M is a big enough constant ($M > \max\{\|a_i - a_{i'}\| : i, i' \in N, i \neq i'\}$) and $c_i = \sum_{k\in N} \lambda_k \epsilon_{ik}^*, i \in N$.

Objective function (2.3) is the minimum reduced cost associated with the optimal solution of the pricing problem. Constraints (2.4) define the distances. As in Section 2.2, this family of constraints is defined *ad hoc* for a given norm. Constraints (2.5) set correctly the r-variables. Finally, constraints (2.6) and (2.7) are the domain of the variables.

As it has been shown in the proof of Theorem5, the above problem can be formulated as a MISOCO problem in case of polyhedral or ℓ_τ-norms.

The so-called *Farkas pricing* should be adequately defined in case the feasibility of (RRMP) is not ensured. That strategy allows one to detect such an infeasibility by means of solving a pricing problem similar to (RRMP). However, we avoid the use of the Farkas pricing applying the following strategy: (a) we introduce in the firstly solved master problem the initial pool of variables as described in Section 2.3.1; and (b) since the feasibility of the master problem might be lost along the branching process of our branch-and-price approach, we add an *artificial* variable $y_{(N,x_0)}$ whose local lower bound is never set to zero and with $\delta_i^{(N,x_0)}$ being a big enough value. This strategy allows us to assure that (MP_2) is satisfied by this variable, and the overall master problem is always feasible.

When the pricing problem is optimally solved, one can obtain a theoretical lower bound even if more variables must be added. The following remark explains how the result is applied to our particular problem.

Remark 1. *Desrosiers and Lübbecke (2005) provide theoretical lower bounds for binary programming problems that are embedded into branch-and-price approaches, in case the number of binary variables that can take value one is upper bounded . In our case, the number of y-variables in* (MP) *that take value one is exactly p. Thus, one can compute a lower bound for* (MP) *as:*

$$LB = z_{\text{RRMP}} + p \min_{S,x} \overline{c}_{(S,x)}, \tag{2.8}$$

where z_{RRMP} is the objective value of any of the relaxed problems (RRMP) *and $\overline{c}_{(S,x)}$ is the reduced cost of the variable defined by (S,x).*

It is important to remark that this bound can be computed at each node of the branch-and-bound tree. The bounds are particularly useful at the root node since they may help to accelerate the optimality certification, or for large instances where the linear relaxation is not solved within the time limit.

Observe also that for adding a variable to the master problem, it suffices to find one variable y_R with negative reduced cost. This search can be performed by solving exactly the pricing problem, although that might have a high computational load. Alternatively, one could also solve heuristically the pricing problem, hoping for variables with negative reduced costs. In what follows, this approach will be called the *heuristic pricer*. The key observation is to check if a candidate facility is promising to this end.

Given the coordinates of a facility, x, we construct a set of demand points, S, compatible with the conditions of the node of the branch-and-bound tree by allocating demand points in S to x whenever the reduced cost $c_{(S,x)} - z_{(S,x)} = \gamma^* + \sum_{i\in S} e_i < 0$, where $e_i = -\alpha_i^* + \sum_{k\in N} \delta_i(x)\lambda_k \epsilon_{ik}^*$. In that case, the variable $y_{(S,x)}$ is candidate to be added to the pool of columns. Here, we detail how the heuristic pricer algorithm is implemented at the root node. For deeper nodes in the branch-and-bound tree we refer the reader to Section 2.3.3.

For the root node, there is a very easy procedure to solve this problem, just selecting the negative ones, i.e., we define $S = \{i \in N : e_i < 0\}$ and, in case $c_{(S,x)} - z_{(S,x)} < 0$, the variable $y_{(S,x)}$ could be added to the problem. Additionally, the region where the facility is generated can be significantly reduced, in particular to the hyperrectangle defined by demand points with negative e_i.

In both exact and heuristic pricer, we use multiple pricing, i.e., several columns are added to the pool at each iteration, if possible. In the exact pricer, we take advantage that the solver saves different solutions besides the optimal one, so it might provide us more than one column with negative reduced cost. In the heuristic pricer, we add the best variables in terms of reduced cost as long as their associated reduced costs are negative.

2.3.3 Branching

When the relaxed (MP) is solved, but the solution is not integer, the next step is to define an adequate branching rule to explore the searching tree. In this problem, we apply an adaptation of the Ryan and Foster branching rule (Ryan and Foster, 1981). Given a solution with fractional y-variables in a node, it might occur that

$$0 < \sum_{R=(S,x)\in\mathcal{R}:i_1,i_2\in S} y_R < 1, \text{ for some } i_1, i_2 \in N,\ i_1 < i_2. \tag{2.9}$$

Provided that this happens, in order to find an integer solution, we create the following branches from the current node:

- **Left branch:** i_1 and i_2 must be served by different facilities.

$$\sum_{R=(S,x)\in\mathcal{R}:i_1,i_2\in S} y_R = 0.$$

- **Right branch:** i_1 and i_2 must be served by the same facility.

$$\sum_{R=(S,x)\in\mathcal{R}:i_1,i_2\in S} y_R = 1.$$

Remark 2. *The above information is easily translated to the pricing problem adding one constraint to each one of the branches: 1) $w_{i_1} + w_{i_2} \leq 1$ for the left branch; and 2) $w_{i_1} = w_{i_2}$ for the right branch.*

It might also happen that being some y_R fractional, $\sum_{R=(S,x)\in\mathcal{R}:i_1,i_2\in S} y_R$ is integer for all $i_1, i_2 \in N, i_1 < i_2$. The following result allows us to use this branching rule and provides a procedure to recover a feasible solution encoded in the current solution of the node.

Theorem 6. *If $\sum_{R=(S,x)\in\mathcal{R}:i_1,i_2\in S} y_R \in \{0,1\}$, for all $i_1, i_2 \in N$, such that $i_1 < i_2$, then there exists an integer feasible solution of* (MP) *with the same objective function value.*

Proof. Let $\mathcal{X}_S$ be the set of all facilities which are part of a variable $y_{(S,x)}$ belonging to the pool of columns. We define $\mathcal{X}_S$ for all used partitions S. First, it is proven in Barnhart et al. (1998) that, under the hypothesis of the theorem, the following expression holds for any set S in a partition.

$$\sum_{x\in\mathcal{X}_S} y_{(S,x)} \in \{0,1\}.$$

If $\sum_{x\in\mathcal{X}_S} y_{(S,x)} = 0$, then $y_{(S,x)} = 0$, for all $x \in \mathcal{X}_S$, because of the nonnegativity of the variables. However, if

$$\sum_{x\in\mathcal{X}_S} y_{(S,x)} = 1, \tag{2.10}$$

$y_{(S,x)}$ could be fractional, for some $x \in \mathcal{X}_S$.

Observe that, currently, the distance associated with demand point $i \in S$ in the problem is

$$\delta_i^S = \sum_{x\in\mathcal{X}_S} y_{(S,x)}\delta_i(x).$$

Thus, from the above we construct a new facility x^* for S.

$$x_l^* = \sum_{x\in\mathcal{X}_S} y_{(S,x)}x_l, \ \forall l = 1,\dots,d, \tag{2.11}$$

so that $\delta_i(x^*) \le \delta_i^S$, $\forall i \in S$.

Indeed, by the triangular inequality and by (2.10),

$$\delta_i(x^*) = \|x^* - a_i\| = \|\sum_{x\in\mathcal{X}_S} y_{(S,x)}(x-a_i)\| \le \sum_{x\in\mathcal{X}_S} y_{(S,x)}\|x-a_i\| = \delta_i^S,$$

for all $i \in S$. The inequality being strict unless $x - a_i$, for all $x \in \mathcal{X}_S$, are collinear.

Finally, we have constructed the variable $y_{(S,x^*)} = 1$ as part of a feasible integer solution of the master problem (MP). Therefore, it ensures that either the solution is binary or there exists a binary feasible solution with the same objective function value. □

Among all the possible choices of pairs i_1, i_2 verifying (2.9), we propose to select the one provided by the following rule:

$$\arg \max_{\substack{i_1,i_2:\\ 0<\sum_{\substack{R=(S,x)\in\mathcal{R}:\\ i_1,i_2\in S}} y_R<1}} \left\{ \theta \min\left\{ \sum_{\substack{R=(S,x)\in\mathcal{R}:\\ i_1,i_2\in S}} y_R, 1 - \sum_{\substack{R=(S,x)\in\mathcal{R}:\\ i_1,i_2\in S}} y_R \right\} + \frac{1-\theta}{\|a_{i_1}-a_{i_2}\|} \right\}. \tag{θ-rule}$$

This rule uses the most fractional y-solution, but also pays attention to the pairs of demand points which are close to each other in the solution, assuming they will be part of the same variable with value one at the optimal solution. It has been successfully applied in a related Discrete Ordered Median Problem (Deleplanque et al., 2020). The parameter θ is chosen in $[0,1]$, where for $\theta = 0$, the closest demand points among the pairs with fractional sum will be selected, while for $\theta = 1$, the most fractional branching will be applied.

The above branching rule affects the heuristic pricer procedure, since not all $S \subset N$ are compatible with the branching conditions leading to a node. In case that we have to respect some branching decisions, the pricing problem gains complexity. Therefore, we develop a greedy algorithm which generates heuristic variables respecting the branching decision in the current node. This algorithm uses the information from the branching rule to build the new variable to add.

The candidate set S is built by means of a greedy algorithm similar to the one presented in Sakai et al. (2003). First, we construct a graph of incompatibilities $G = (V, E)$, with V and E defined as follows: for each maximal subset of demand points $i_1 < i_2 < \cdots < i_m$, that according to the branching rule have to be assigned to the same subset, we include a vertex v_{i_1} with weight $\omega_{i_1} = \sum_{i \in \{i_1,\ldots,i_m\}} e_i$; next, for each $v_i, v_{i'} \in V$, such that i and i' cannot be assigned to the same subset at the current node, we define $\{v_i, v_{i'}\} \in E$. The subset S minimizing the reduced cost for a given x can be calculated solving the Maximum Weighted Independent Set Problem over G. The algorithm solves this problem heuristically applying the GGWMIN selection vertex rule proposed by Sakai et al. (2003).

2.3.4 Convergence

Due to the huge number of variables that might arise in column generation procedures, it is very important checking the possible degeneracy of the algorithm. Accelerating the convergence has been traditionally afforded by means of stabilization techniques. In recent papers, it has been shown how heuristic pricers avoid degeneracy (e.g., Benati et al., 2022; Blanco et al., 2021c). Stabilization and heuristic pricers have in common that both do not add in the first iterations variables with the minimum associated reduced cost. This idea has been empirically shown to accelerate convergence (see, e.g., Du Merle et al., 1999; Pessoa et al., 2010).

For the sake of readability, all the computational analysis is included in Section 2.5. There, the reader can see how our heuristic pricer needs less variables to certify optimality than the exact pricer for medium- and large-sized instances, therefore, accelerating the convergence.

2.4 Matheuristic approaches

(**MFMOMP**$_\lambda$) is an NP-hard combinatorial optimization problem, and both the compact formulation and the proposed branch-and-price approach are limited by the number of demand points (n) and facilities (p) to be considered. Actually, as we will see in Section 2.5, the two exact approaches are only capable of solving, optimally, small- and medium-sized instances. In this section, we derive three different matheuristic procedures, capable to handle larger instances in reasonable CPU times. The first approach is based on using the branch-and-price scheme but solving only heuristically the pricing problem. The second is an aggregation based-approach that will also allow us to derive theoretical error bounds on the approximation. A third heuristic based on discretizing the space is proposed.

2.4.1 Heuristic pricer

The matheuristic procedure described here has been successfully applied in the literature. See, e.g., Albornoz and Zamora (2021); Benati et al. (2022); Deleplanque et al. (2020), and the references therein. Recall that our pricing problem is NP-hard. In order to avoid the exact procedure for large-sized instances, where not even a single iteration could be solved exactly, we propose a matheuristic. It consists of solving each pricing problem heuristically. The inconvenience of doing that is that we do not have a theoretic lower bound during the process. Nevertheless, for instances where the time limit is reached, we are able to visit more nodes in the branch-and-bound tree which could allow us to obtain better incumbent solutions than the unfinished exact procedure.

2.4.2 Aggregation schemes

The second matheuristic approach that we propose is based on applying aggregation techniques to the input data (the set of demand points). This type of approaches has been successfully applied to solve large-scale continuous location problems (see Blanco et al., 2021c, 2018; Current and Schilling, 1990; Daskin et al., 1989; Irawan, 2016a).

Let $\mathcal{A} = \{a_1, \ldots, a_n\} \subset \mathbb{R}^d$ be a set of demands points. In an aggregation procedure, the set $\mathcal{A}$ is replaced by a multiset $\mathcal{A}' = \{a'_1, \ldots, a'_n\}$, where each point a_i in $\mathcal{A}$ is assigned to a point a'_i in $\mathcal{A}'$. In order to be able to solve (**MFMOMP**$_\lambda$) for $\mathcal{A}'$, the cardinality of the different elements of $\mathcal{A}'$ is assumed to be smaller than the cardinality of $\mathcal{A}$, and then, several a_i might be assigned to the same a'_i.

Once the points in $\mathcal{A}$ are aggregated into $\mathcal{A}'$, the procedure consists of solving (**MFMOMP**$_\lambda$) for the demand points in $\mathcal{A}'$. We get a set of p optimal facilities for the aggregated problem, $\mathcal{X}' = \{x'_1, \ldots, x'_p\}$, associated with its objective value $\mathsf{OM}_\lambda(\mathcal{A}'; \mathcal{X}')$. These positions can also be evaluated in the original objective function of the problem for the demand points $\mathcal{A}$, $\mathsf{OM}_\lambda(\mathcal{A}; \mathcal{X}')$. The following result allows us to get upper bound of the error incurred when aggregating demand points.

Theorem 7. *Let $\mathcal{X}^*$ be the optimal solution of* (**MFMOMP$_\lambda$**) *and* $\Delta = \max_{i \in N} \|a_i - a'_i\|$. *Then*

$$|\mathsf{OM}_\lambda(\mathcal{A}; \mathcal{X}^*) - \mathsf{OM}_\lambda(\mathcal{A}; \mathcal{X}')| \leq 2\Delta \sum_{i \in N} \lambda_i. \tag{2.12}$$

Proof. By the triangular inequality and the monotonicity and sublinearity of the ordered median function we have that $\mathsf{OM}_\lambda(\mathcal{A}; \mathcal{X}) \leq \mathsf{OM}_\lambda(\mathcal{A}'; \mathcal{X}) + \mathsf{OM}_\lambda(\mathcal{A}'; \mathcal{A})$ for all $\mathcal{X} = \{x_1, \ldots, x_p\} \subset \mathbb{R}^d$. Since $\Delta \geq \|a_i - a'_i\|$ for all $i \in I$ we get that $|\mathsf{OM}_\lambda(\mathcal{A}; \mathcal{X}) - \mathsf{OM}_\lambda(\mathcal{A}'; \mathcal{X})| \leq \Delta \sum_{i \in I} \lambda_i$ for all $\mathcal{X} = \{x_1, \ldots, x_p\} \subset \mathbb{R}^d$. Applying (Geoffrion, 1977, Theorem 5), we get that $|\mathsf{OM}_\lambda(\mathcal{A}; \mathcal{X}^*) - \mathsf{OM}_\lambda(\mathcal{A}'; \mathcal{X}')| \leq \Delta \sum_{i \in I} \lambda_i$, and then:

$$\begin{aligned} |\mathsf{OM}_\lambda(\mathcal{A}; \mathcal{X}^*) - \mathsf{OM}_\lambda(\mathcal{A}; \mathcal{X}')| &\leq |\mathsf{OM}_\lambda(\mathcal{A}; \mathcal{X}^*) - \mathsf{OM}_\lambda(\mathcal{A}'; \mathcal{X}')| + \\ &\quad\ |\mathsf{OM}_\lambda(\mathcal{A}'; \mathcal{X}') - \mathsf{OM}_\lambda(\mathcal{A}; \mathcal{X}')| \\ &\leq 2\Delta \sum_{i \in I} \lambda_i. \end{aligned}$$

□

There are different strategies to reduce the dimensionality by aggregating points. In our computational experiments we consider two differentiated approaches: the *k-Means Clustering* (KMEANS) and the *Pick The Farthest* (PTF). In KMEANS, we replace the original points by the centroids. Alternatively, in PTF, an initial random demand point from $\mathcal{A}$ is chosen and the rest are selected as the farthest demand point from the last one chosen, until a predefined number of points is reached (Daskin et al., 1989).

2.4.3 Discretization

We propose a third heuristic algorithm which consists of solving a discrete version of our problem, also known as the Discrete Ordered Median Problem (DOMP for short) (Nickel and Puerto, 2006). In this matheuristic, the potential facilities are chosen among the demand points to solve the DOMP with the solution methods developed in Deleplanque et al. (2020). This approach produces suboptimal solutions since the feasible domain of the DOMP is a discrete set contained in the solution space of (**MFMOMP$_\lambda$**). The reader can see in Section 2.5.2 that, for large-sized instances, this methodology provides rather good results.

2.5 Computational study

In order to compare the performance of our branch-and-price and our matheuristic approaches, we report the results of our computational experiments. We consider different sets of instances used in the location literature with size ranging from 20 to 654 demand points in the plane. In all of them, the number of facilities to be located, p, ranges in

$\{2, 5, 10\}$ and we solve the instances for the λ-vectors in Table 2.1, {W, C, K, D, S, A}. We set $k = \frac{n}{2}$ for the k-center and k-entdian, and $\alpha = 0.9$ for the centdian and k-entdian.

Notation	λ-vector	Name
W	$(1, \ldots, 1)$	p-median
C	$(0, \ldots, 0, 1)$	p-center
K	$(0, \ldots, 0, \overbrace{1, \ldots, 1}^{k})$	k-center
D	$(\alpha, \ldots, \alpha, 1)$	centdian
S	$(\alpha, \ldots, \alpha, \overbrace{1, \ldots, 1}^{k})$	k-entdian
A	$(0 = \frac{0}{n-1}, \frac{1}{n-1}, \frac{2}{n-1}, \ldots, \frac{n-2}{n-1}, \frac{n-1}{n-1} = 1)$	ascendant

Table 2.1: Examples of Ordered Median aggregation functions

For the sake of readability, we restrict the computational study of this document to ℓ_1-norm based distances. However, the reader can find extensive computational results for other norms in 2.6 and 2.7.

The models were coded in C and solved with SCIP v.7.0.2 (Gamrath et al., 2020) using as optimization solver SoPLex 5.0.2 in a Mac OS Catalina with a Core Intel Xeon W clocked at 3.2 GHz and 96 GB of RAM memory.

2.5.1 Computational performance of the branch-and-price procedure

In this section we report the results for our branch-and-price approach based on the classical dataset provided by Eilon et al. (1974). From this dataset, we randomly generate five instances with sizes $n \in \{20, 30, 40, 45\}$ and we also consider the entire complete original instance with $n = 50$. Together with the number of facilities p and the different ordered weighted median functions (`type`), a total of 378 instances has been considered.

Firstly, concerning convergence (Section 2.3.4), each line in Table 2.2 shows the average results of 45 instances, five for each type of ordered median objective function to be minimized {W, D, S} and $p \in \{2, 5, 10\}$, solved to optimality. The results has been split by size (n) and by `Heurvar`: `FALSE` when only the exact pricer is used; `TRUE` if the heuristic pricer is used and the exact pricer is called when it does not provide new columns to add. The reader can see a significant reduction of the CPU time (`Time`) caused by a decrease of the number of calls to the exact pricer (`Exact`) even though the number of total iterations (`Total`) increases. Additionally, a second effect is that the necessary number of variables to certify optimality (`Vars`) is slightly less when the heuristic is applied for $n = 40$. Hence, we will use the heuristic pricer for the rest of the experiments.

Secondly, we have tuned the values of θ for the branching rule (θ-rule) for each of the objective functions (different values for the λ-vector) based in our computational experi-

n	Heurvar	Iterations		Vars	Time
		Exact	Total		
20	FALSE	13	13	2189	64.92
	TRUE	4	23	2219	18.02
30	FALSE	15	15	2827	1034.97
	TRUE	3	60	2856	191.84
40	FALSE	50	50	4713	9086.33
	TRUE	13	136	4511	2229.21

Table 2.2: Average number of pricer iterations, variables and time using the combined heuristic and exact pricers or only using the exact pricer

type	$\theta = 0.0$	$\theta = 0.1$	$\theta = 0.3$	$\theta = 0.5$	$\theta = 0.7$	$\theta = 0.9$	$\theta = 1.0$
W	0.04	0.04	0.04	0.04	0.04	0.04	**0.02**
C	**27.94**	28.34	28.29	28.47	28.64	28.74	28.19
K	12.83	12.63	12.80	**12.46**	12.73	13.15	12.88
D	0.09	0.07	0.09	0.09	0.09	0.09	**0.02**
S	0.11	0.14	0.14	0.14	0.14	0.13	**0.10**
A	7.73	7.66	7.69	7.71	7.64	7.73	**7.33**

Table 2.3: GAP(%) for ℓ_1-norm, Eilon et al. (1974) instances

ence. In Table 2.3, we show the average gap at termination of the above-mentioned 378 instances when we apply our branch-and-price approach fixing a time limit of 2 hours.

Therefore we set $\theta = 0$ for the center problem (C), $\theta = 0.5$ for the k-center problem (K), and $\theta = 1$ for the p-median (W), centdian (D), k-entdian (S), and ascendant problems (A). Recall that when we use $\theta = 0$, we are selecting a pure distance branching rule. In contrast, when $\theta = 1$, we select the most fractional variable. On the other hand, when $\theta = 0.5$, we use a hybrid selection between the two extremes of the (θ-rule). In the following, the above fixed parameters will be used in the computational experiments for exact and matheurisitic methods.

The average results obtained for the Eilon et al. (1974) instances, with a CPU time limit of 2 hours, are shown in Table 2.4. There, for each combination of n (size of the instance), p (number of facilities to be located) and type (ordered median objective function to be minimized), we provide the average results for ℓ_1-norm with a comparison between the compact formulation (C) (Compact) and the branch-and-price approach (B&P). The table is organized as follows: the first column gives the CPU time in seconds needed to solve the problem (`Time`) and within parentheses the number of unsolved instances (`#Unsolved`), i.e., those for which the lower and upper bound do not coincide within the time limit; the second column shows the gap at the root node; the third one gives the gap at termination, i.e., the remaining MIP gap in percentage (`GAP(%)`) when the time limit is reached, 0.00 otherwise; in the fourth column we show the number of variables (`Vars`) needed to solve the problem; in the fifth column we show the number of nodes (`Nodes`) explored in the branch-and-bound tree; and, in the last one, the RAM memory (`Memory (MB)`) in Megabytes required during the execution process is reported. Within each column, we highlight in bold the best result between the two formulations, namely Compact or B&P.

The branch-and-price algorithm is able to solve optimally 58 instances more than the compact formulation. However, for some instances (mainly Center and k-center problems or when $p = 2$) the solved instances with the compact formulation need less CPU time. Thus, the first conclusion could be that when p increases decomposition techniques become more important because the number of variables is not so dependant of this parameter. The second conclusion from the results is that the branch-and-price is a very powerful tool when the gap at the root node is close to zero which does not happen when a big percentage of the positions of the λ-vector are zeros. Concerning the memory used by the tested formulations, the compact formulation needs bigger branch-and-bound trees to deal with fractional solutions whereas that branch-and-price uses more variables.

Since the average gap at termination for the branch-and-price algorithm is much smaller than the one obtained by the compact formulation (7.98% against 35.81%), we will use decomposition-based algorithms to study medium- and large-sized instances.

2.5.2 Computational performance of the matheuristics

In this section, we show the performance of our matheuristic procedures. Firstly, we will test them for $n = 50$ (Eilon et al., 1974) where the solutions can be compared with the theoretic bounds provided by the exact method. Secondly, we will compare them using larger instances. Specifically, we use two instances from the TSP library (Reinelt, 1991), which is a well known repository of complex instances for the the TSP and related problems (Goldengorin and Krushinsky, 2011): `att532.tsp` and `p654.tsp`. They contain the coordinates of 532 cities of the continental US (Padberg and Rinaldi, 1987) and 654 points from a drilling problem example (Reinelt, 1992b), respectively. The spatial distribution of the demand points for each of the instances that we use is shown in Figure 2.1, where one can see the different nature of the datasets that we test.

Tables 2.5, 2.6, and 2.7 present a similar layout. The instances are solved with 18 different configurations of ordered weighted median functions and number of open facilities. Each of these 18 problems has been solved by means of the following strategies: branch-and-price procedure (B&P); the heuristic used to generate initial columns (InitialHeur); the decomposition-based heuristic (Matheur); the aggregation-based approaches described in Section 2.4.2 (KMEANS-20, KMEANS-30, PTF-20, PTF-30) for $|\mathcal{A}'| = \{20, 30\}$; and the discretization strategy explained in Section 2.4.3 (Discretization). The reported results are the CPU time and:

1. the gap ($\texttt{GAP}_{LB}\texttt{(\%)}$) which is calculated with respect to the lower bound of the branch-

n	type	p	Time (#Unsolved)				GAProot(%)		GAP(%)		Vars		Nodes		Memory (MB)	
			Compact		B&P		Compact	B&P	Compact	B&P	Compact	B&P	Compact	B&P	Compact	B&P
20	W	2	**1.59**	**(0)**	22.90	(0)	93.92	**0.00**	0.00	0.00	**224**	2131	9518	**1**	**4**	103
		5	1588.99	(1)	**8.34**	**(0)**	100.00	**0.00**	3.38	**0.00**	**470**	2408	10967305	**1**	1278	**49**
		10	—	(5)	**3.95**	**(0)**	100.00	**0.46**	43.84	**0.00**	**880**	2127	19785215	**2**	3425	**28**
	C	2	**0.06**	**(0)**	237.96	(4)	78.92	**22.59**	**0.00**	10.78	**224**	97635	**7**	4652	**4**	2239
		5	**12.58**	**(0)**	—	(5)	100.00	**29.46**	**0.00**	17.16	**470**	15251	40379	**18660**	**12**	464
		10	**511.69**	**(2)**	1831.83	(4)	100.00	**37.64**	**7.59**	20.28	**880**	4243	7928195	**21617**	725	**160**
	K	2	**0.35**	**(0)**	1412.69	(1)	91.43	**7.55**	**0.00**	1.42	**224**	37917	**630**	670	**3**	953
		5	**243.88**	**(0)**	404.99	(3)	100.00	**15.40**	**0.00**	3.85	**470**	9363	657827	**6642**	**77**	279
		10	32.22	(4)	**3156.63**	**(2)**	100.00	**18.53**	36.95	**3.26**	**880**	4071	12150962	**9244**	2265	**111**
	D	2	**2.18**	**(0)**	30.36	(0)	93.78	**0.03**	**0.00**	0.00	**224**	2135	9222	**1**	**5**	108
		5	1535.82	(1)	**12.18**	**(0)**	100.00	**0.00**	6.69	**0.00**	**470**	2401	8972062	**1**	1225	**49**
		10	5030.79	(4)	**6.88**	**(0)**	100.00	**0.46**	48.19	**0.00**	**880**	2127	15660031	**4**	2798	**28**
	S	2	**2.24**	**(0)**	54.23	(0)	93.77	**0.16**	0.00	0.00	**224**	2119	7677	**1**	**5**	106
		5	1238.87	(1)	**15.75**	**(0)**	100.00	**0.06**	4.40	**0.00**	**470**	2401	8141244	**2**	745	**50**
		10	—	(5)	**7.61**	**(0)**	100.00	**0.53**	50.12	**0.00**	**880**	2126	16072018	**5**	2835	**28**
	A	2	**0.85**	**(0)**	783.95	(1)	91.63	**4.45**	**0.00**	0.35	**224**	16973	1340	**400**	**4**	738
		5	**411.21**	**(0)**	2304.77	(0)	100.00	**10.18**	**0.00**	0.00	**470**	7405	878975	**1697**	**126**	222
		10	60.27	(4)	**883.79**	**(1)**	100.00	**17.10**	31.87	**1.73**	**880**	3288	10637608	**2721**	1723	**79**
30	W	2	**139.91**	**(0)**	526.92	(0)	93.86	**0.00**	0.00	0.00	**334**	3142	787145	**1**	**38**	260
		5	—	(5)	**64.66**	**(0)**	100.00	**0.00**	52.05	**0.00**	**700**	2963	17382888	**1**	8647	**109**
		10	—	(5)	**19.51**	**(0)**	100.00	**0.00**	76.26	**0.00**	**1310**	2472	12250097	**1**	4692	**55**
	C	2	**0.11**	**(0)**	39.44	(4)	79.19	**21.41**	**0.00**	15.46	**334**	125429	**66**	931	**8**	1443
		5	**30.64**	**(0)**	1564.58	(4)	100.00	**31.68**	**0.00**	22.73	**700**	30216	69019	**2817**	**19**	389
		10	**4212.55**	**(3)**	—	(5)	100.00	**34.18**	**16.67**	27.51	**1310**	12928	9619002	**6027**	1823	**190**
	K	2	**4.44**	**(0)**	409.69	(4)	90.88	**8.65**	**0.00**	7.58	**334**	45846	8511	**147**	**10**	1696
		5	2956.65	(4)	**5199.43**	**(3)**	100.00	**12.01**	17.79	**5.82**	**700**	18893	12169516	**815**	2570	**534**
		10	—	(5)	**2740.67**	**(4)**	100.00	**18.84**	69.60	**12.31**	**1310**	7416	9299590	**2992**	3105	**187**
	D	2	**201.28**	**(0)**	454.39	(0)	93.77	**0.00**	0.00	0.00	**334**	3087	757445	**1**	**49**	258
		5	—	(5)	**65.46**	**(0)**	100.00	**0.00**	57.16	**0.00**	**700**	2957	9914066	**1**	7439	**111**
		10	—	(5)	**21.25**	**(0)**	100.00	**0.00**	79.34	**0.00**	**1310**	2464	10108803	**1**	4631	**55**
	S	2	**203.04**	**(0)**	370.63	(0)	93.68	**0.00**	**0.00**	0.00	**334**	3184	566382	**1**	**41**	263
		5	—	(5)	**160.85**	**(0)**	100.00	**0.03**	56.47	**0.00**	**700**	2963	9283122	**2**	7054	**112**
		10	—	(5)	**42.86**	**(0)**	100.00	**0.09**	79.91	**0.00**	**1310**	2469	9530286	**3**	4686	**56**
	A	2	**21.89**	**(0)**	3640.13	(2)	91.15	**4.46**	**0.00**	3.26	**334**	12721	26764	**41**	**12**	845
		5	5403.72	(4)	**2750.01**	**(3)**	100.00	**7.76**	28.99	**2.60**	**700**	8615	8044660	**188**	2288	**357**
		10	—	(5)	**804.71**	**(4)**	100.00	**13.38**	70.51	**6.64**	**1310**	5529	7159232	**1465**	2364	**165**
40	W	2	4028.70	(4)	**1675.34**	**(0)**	93.79	**0.01**	12.34	**0.00**	**444**	5211	26828725	**1**	2515	**645**
		5	—	(5)	**1647.86**	**(0)**	100.00	**0.02**	67.11	**0.00**	**930**	4028	12240990	**3**	10977	**229**
		10	—	(5)	**348.57**	**(0)**	100.00	**0.09**	81.57	**0.00**	**1740**	4001	7841923	**2**	4267	**125**
	C	2	**0.25**	**(0)**	—	(5)	75.52	**30.52**	**0.00**	29.73	**444**	136451	**237**	259	**15**	1541
		5	**116.02**	**(0)**	—	(5)	100.00	**42.30**	**0.00**	41.65	**930**	27041	195892	**158**	**42**	224
		10	**3022.45**	**(4)**	—	(5)	100.00	**36.47**	**31.47**	33.88	**1740**	12733	7126207	**667**	2024	**110**
	K	2	**58.78**	**(0)**	—	(5)	90.67	**14.52**	**0.00**	14.52	**444**	14164	93918	**11**	**27**	897
		5	—	(5)	—	(5)	100.00	**21.45**	56.58	**21.44**	**930**	10132	6803627	**28**	5632	**360**
		10	—	(5)	—	(5)	100.00	**19.04**	75.08	**17.71**	**1740**	8823	5436226	**280**	2606	**198**
	D	2	5908.68	(4)	**436.48**	**(1)**	93.67	**0.02**	15.22	**0.01**	**444**	5669	16542227	**2**	3164	**709**
		5	—	(5)	**855.62**	**(1)**	100.00	**0.11**	68.93	**0.08**	**930**	4094	7984937	**2**	10233	**233**
		10	—	(5)	**331.54**	**(0)**	100.00	**0.07**	83.85	**0.00**	**1740**	4004	5704188	**2**	4413	**126**
	S	2	4977.44	(4)	**429.96**	**(1)**	93.60	**0.47**	14.33	**0.47**	**444**	5195	12853124	**1**	2430	**657**
		5	—	(5)	**2159.56**	**(1)**	100.00	**0.14**	70.18	**0.02**	**930**	4082	7715457	**4**	9805	**233**
		10	—	(5)	**615.35**	**(0)**	100.00	**0.17**	84.62	**0.00**	**1740**	3999	5409994	**4**	4687	**126**
	A	2	**533.79**	**(0)**	—	(5)	90.85	**8.30**	**0.00**	8.19	**444**	6506	455652	**3**	**48**	769
		5	—	(5)	—	(5)	100.00	**14.76**	58.65	**14.17**	**930**	5538	3684557	**10**	3409	**331**
		10	—	(5)	—	(5)	100.00	**12.35**	74.52	**10.21**	**1740**	6285	4403838	**161**	2000	**214**
45	W	2	—	(5)	**483.59**	**(1)**	94.05	**0.04**	27.06	**0.02**	**499**	7219	24989615	**2**	5854	**1085**
		5	—	(5)	**1745.55**	**(2)**	100.00	**0.32**	71.65	**0.27**	**1045**	4855	11473640	**4**	11171	**374**
		10	—	(5)	**635.43**	**(0)**	100.00	**0.03**	83.54	**0.00**	**1955**	4239	5717627	**1**	3767	**168**
	C	2	**0.46**	**(0)**	—	(5)	74.99	**39.01**	**0.00**	38.99	**499**	109398	628	**104**	**17**	1364
		5	**144.75**	**(0)**	—	(5)	100.00	**40.69**	**0.00**	40.62	**1045**	20483	215522	**31**	**44**	176
		10	—	(5)	—	(5)	100.00	**32.80**	37.16	**31.54**	**1955**	14204	6469050	**219**	1915	**110**
	K	2	**342.18**	**(0)**	—	(5)	91.23	**16.93**	**0.00**	16.93	**499**	10490	497310	**4**	**59**	845
		5	—	(5)	—	(5)	100.00	**22.98**	64.55	**22.98**	**1045**	6631	5434589	**11**	6520	**295**
		10	—	(5)	—	(5)	100.00	**16.68**	77.74	**16.48**	**1955**	8738	4555667	**78**	2681	**220**
	D	2	—	(5)	**364.11**	**(1)**	93.96	**0.02**	29.64	**0.02**	**499**	6473	14042725	**1**	6338	**973**
		5	—	(5)	**1744.42**	**(2)**	100.00	**0.17**	73.98	**0.11**	**1045**	4731	7322624	**3**	10301	**365**
		10	—	(5)	**667.13**	**(0)**	100.00	**0.02**	84.73	**0.00**	**1955**	4231	5228591	**1**	4875	**169**
	S	2	—	(5)	**623.35**	**(2)**	93.87	**0.10**	28.98	**0.09**	**499**	7260	10776521	**2**	4776	**1093**
		5	—	(5)	—	(5)	100.00	**0.72**	76.35	**0.62**	**1045**	4899	7356115	**7**	10281	**378**
		10	—	(5)	**1848.85**	**(0)**	100.00	**0.18**	85.38	**0.00**	**1955**	4258	4378226	**4**	4283	**168**
	A	2	**4681.25**	**(0)**	—	(5)	91.28	**11.43**	**0.00**	11.43	**499**	6849	3057137	**2**	**121**	975
		5	—	(5)	—	(5)	100.00	**17.39**	65.13	**17.17**	**1045**	5476	2139768	**4**	2517	**415**
		10	—	(5)	—	(5)	100.00	**10.28**	76.58	**8.88**	**1955**	6105	3577556	**56**	1976	**244**
50	W	2	—	(1)	**331.87**	**(0)**	94.13	**0.00**	34.44	**0.00**	**554**	8094	24416531	**1**	6585	**1464**
		5	—	(1)	**410.87**	**(0)**	100.00	**0.00**	76.08	**0.00**	**1160**	5292	9438723	**1**	9646	**466**
		10	—	(1)	**1005.02**	**(0)**	100.00	**0.00**	84.68	**0.00**	**2170**	4914	5017512	**1**	3878	**225**
	C	2	**0.34**	**(0)**	—	(1)	75.02	**30.33**	**0.00**	30.31	**554**	80356	367	**37**	**22**	1064
		5	**379.06**	**(0)**	—	(1)	100.00	**41.37**	**0.00**	41.37	**1160**	15314	443313	**14**	**137**	143
		10	—	(1)	—	(1)	100.00	**37.49**	46.67	**36.94**	**2170**	12538	5837062	**213**	2937	**98**
	K	2	**1135.78**	**(0)**	—	(1)	91.28	**15.46**	**0.00**	15.46	**554**	10042	1334361	**3**	**84**	926
		5	—	(1)	—	(1)	100.00	**24.86**	68.28	**24.86**	**1160**	6541	4368607	**4**	6095	**324**
		10	—	(1)	—	(1)	100.00	**23.05**	79.98	**23.04**	**2170**	7164	2448072	**15**	1851	**205**
	D	2	—	(1)	**328.07**	**(0)**	94.07	**0.00**	37.30	**0.00**	**554**	8035	12235346	**1**	7096	**1485**
		5	—	(1)	**4430.48**	**(0)**	100.00	**0.08**	78.70	**0.00**	**1160**	5415	5502769	**5**	8618	**485**
		10	—	(1)	**1408.05**	**(0)**	100.00	**0.00**	86.81	**0.00**	**2170**	4914	3617149	**2**	4412	**219**
	S	2	—	(1)	**516.57**	**(0)**	93.95	**0.00**	37.29	**0.00**	**554**	7579	8797114	**1**	4912	**1387**
		5	—	(1)	—	(1)	100.00	**0.57**	79.68	**0.57**	**1160**	5704	5750451	**5**	9004	**508**
		10	—	(1)	**3413.97**	**(0)**	100.00	**0.00**	87.36	**0.00**	**2170**	4962	3126980	**5**	5175	**230**
	A	2	—	(1)	—	(1)	91.49	**10.36**	19.20	**10.36**	**554**	8056	3163853	**2**	**1161**	1369
		5	—	(1)	—	(1)	100.00	**19.06**	67.25	**18.75**	**1160**	5872	2177800	**4**	3290	**542**
		10	—	(1)	—	(1)	100.00	**10.17**	78.48	**9.36**	**2170**	5764	2718050	**16**	1862	**268**
Total Average:			645.10	(229)	**772.80**	**(171)**	96.71	**10.19**	35.81	**7.98**	**897**	13958	6581088	**1111**	3014	**427**

Table 2.4: Results for Eilon et al. (1974) instances for ℓ_1-norm

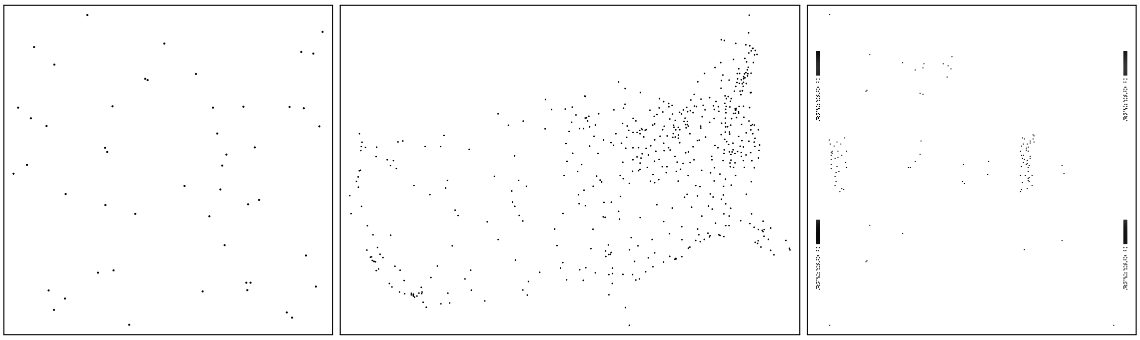


Figure 2.1: Demand points of each of the instances that we use in our computational study: $n = 50$, $n = 532$, and $n = 654$ (from left to right)

type	p	B&P		InitialHeur		Matheur		KMEANS-20		KMEANS-30		PTF-20		PTF-30		Discretization	
		Time	GAP_{LB}(%)	Time	GAP_{LB}(%)	Time	GAP_{LB}(%)	Time	GAP_{LB}(%)	Time	GAP_{LB}(%)	Time	GAP_{LB}(%)	Time	GAP_{LB}(%)	Time	GAP_{LB}(%)
W	2	331.87	**0.00**	0.00	1.44	134.54	**0.00**	74.60	3.10	5204.85	2.80	35.14	7.58	7200.18	2.50	12.94	1.23
	5	410.87	**0.00**	0.00	11.53	9.51	**0.00**	21.47	12.55	398.93	11.25	16.38	17.88	187.03	7.90	2680.48	1.17
	10	1005.02	**0.00**	0.00	21.57	2.84	**0.00**	4.23	17.16	59.08	8.79	16.11	23.45	57.35	9.88	4.95	3.11
C	2	7200.64	43.49	0.00	43.49	6.61	39.68	7200.16	47.05	7200.52	32.36	7200.21	**31.46**	7200.47	41.39	7205.96	43.11
	5	7200.19	70.56	0.00	73.45	7200.00	**46.66**	7200.09	82.76	7200.10	76.52	7200.09	94.74	7200.09	72.70	7203.66	70.86
	10	7200.19	58.58	0.00	101.43	7200.18	**28.93**	7200.16	104.98	7200.19	73.44	7200.16	154.72	7200.18	70.54	7203.84	87.77
K	2	7200.13	18.28	0.00	21.40	227.64	19.79	3589.02	21.19	7200.40	22.00	7200.35	20.05	7200.57	18.81	7203.55	**16.83**
	5	7200.34	33.09	0.00	33.09	392.07	**22.37**	7200.10	26.84	7200.15	36.08	7200.10	42.99	7200.13	30.43	7203.81	26.59
	10	7200.28	29.94	0.00	41.73	864.26	**11.79**	7200.17	35.38	7200.20	19.91	7200.17	41.97	7200.19	15.93	7202.90	38.36
D	2	328.07	**0.00**	0.00	1.45	130.00	**0.00**	42.82	3.76	5213.35	2.82	18.77	7.70	5105.83	2.47	5730.24	1.25
	5	4430.48	**0.00**	0.00	11.43	10.73	**0.00**	7.87	12.53	162.45	10.95	17.50	16.36	191.88	8.13	4714.13	1.12
	10	1408.05	**0.00**	0.00	21.38	3.06	0.43	4.76	17.00	50.22	10.18	22.26	23.27	135.57	9.07	811.58	3.03
S	2	516.57	**0.00**	0.00	1.62	111.73	**0.00**	25.43	2.97	7200.20	2.59	35.77	7.34	7200.27	2.33	7203.79	1.24
	5	7200.43	**0.57**	0.00	12.28	14.96	0.64	36.68	12.60	498.07	11.81	41.88	18.33	746.57	8.26	7203.80	1.77
	10	3413.97	**0.00**	0.00	21.24	3.22	0.66	10.99	16.86	29.98	7.98	50.52	23.29	114.10	8.97	2756.51	4.06
A	2	7200.39	11.56	0.00	11.56	155.18	10.91	2611.77	**9.05**	7200.15	10.62	7200.41	13.49	7200.09	13.69	7203.98	10.27
	5	7200.40	23.08	0.00	23.61	325.35	**14.00**	7200.10	19.96	7200.16	18.63	7200.11	26.03	7200.13	22.18	7204.38	16.41
	10	7200.24	10.33	0.00	31.65	129.77	**6.29**	7200.17	17.48	7200.21	16.63	7200.17	33.49	7200.21	21.82	7204.03	22.39
Total Average:		4658.23	16.64	0.00	26.97	940.09	**11.23**	3157.25	25.73	4645.51	20.85	3614.23	33.56	4763.38	20.39	5330.81	19.48

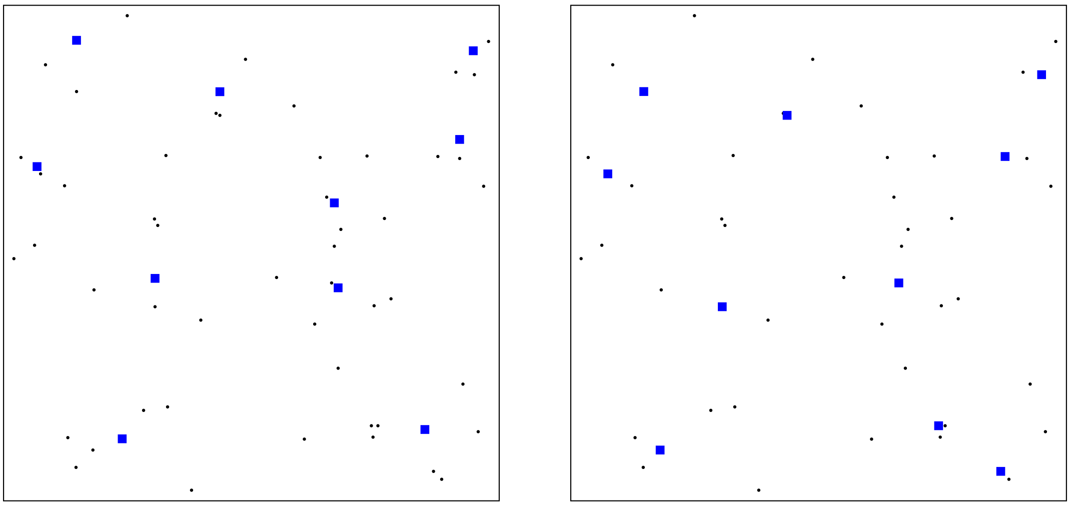
Table 2.5: Heuristic results for instances of $n = 50$, Eilon et al. (1974)

and-price algorithm when the time limit is reached. Thereby, we have a theoretic gap knowing exactly the room for improvement of our heuristics;

2. or $\mathtt{GAP}_{Best}\mathtt{(\%)}$ as the gap with respect to the best known integer solution. We calculate it for large-sized instances since the branch-and-price provides poor lower bounds even using (2.8).

In order to obtain Table 2.5, a time limit of 2 hours was fixed for this experiment with $n = 50$. For these instances, B&P and Matheur report the best performance in most of the cases. In general they present less gap and, in average, it is better not wasting the time solving the exact pricer letting the algorithm go further adding columns or branching before certifying optimality. Thus, with the Matheur strategy we obtain an 11.23% of average gap. In fact, this matheuristic finds the optimal solution (certified by the exact method) at least in six instances. Concerning the time, the other heuristics obtain good quality solution in much smaller CPU times. For the Eilon dataset, the aggregation schemes exhibit that the larger the aggregated set the smaller the gap and the larger the CPU time, as expected.

Figure 2.2 illustrates the optimal solution (square points) of a particular instance (B&P) and the solution when the solution space is limited to the demand points coordinates (Discretization). One can observe in that figure that although the continuous nature of



(a) B&P (b) Discretization

Figure 2.2: Solutions for $n = 50$ (Eilon et al., 1974), S, $p = 10$, and ℓ_1-norm

type	p	B&P		InitialHeur		Matheur		KMEANS-20		KMEANS-30		PTF-20		PTF-30		Discretization	
		Time	GAP_{Best}(%)	Time	GAP_{Best}(%)	Time	GAP_{Best}(%)	Time	GAP_{Best}(%)	Time	GAP_{Best}(%)	Time	GAP_{Best}(%)	Time	GAP_{Best}(%)	Time	GAP_{Best}(%)
W	2	86427.79	3.78	0.04	3.78	86427.79	3.78	214.12	13.36	607.50	6.06	29.81	3.09	807.38	12.08	86465.01	**0.00**
	5	86430.70	3.16	0.04	3.16	86430.70	3.16	15.25	27.80	115.15	21.13	38.83	15.75	296.68	18.38	86417.45	**0.00**
	10	86431.06	1.62	0.04	2.59	86431.06	1.62	7.16	27.35	310.76	35.82	7.42	40.26	51.25	36.48	86423.45	**0.00**
C	2	86406.87	0.08	0.04	0.08	38146.48	**0.00**	86403.24	4.55	86400.09	12.48	86428.73	9.69	86401.42	8.36	86474.06	1.70
	5	86407.73	6.65	0.04	6.65	12456.56	6.65	86400.07	18.75	86400.10	31.08	86400.07	17.39	86400.10	3.09	86427.62	**0.00**
	10	86407.25	28.72	0.04	28.72	86405.98	28.72	15064.77	33.06	86400.16	**0.00**	14624.77	59.63	86400.16	18.58	86460.43	7.32
K	2	86419.13	2.23	0.04	2.23	86419.13	2.23	1601.48	7.51	86406.88	9.01	3974.78	7.09	86403.41	8.29	86445.75	**0.00**
	5	86418.39	2.64	0.04	2.64	86418.39	2.64	86400.08	16.05	86400.12	15.28	86400.09	21.98	86402.98	21.14	86462.28	**0.00**
	10	86418.91	4.83	0.04	4.83	86418.91	4.83	13699.21	83.01	86400.17	9.35	86400.13	37.08	86400.18	31.92	86446.91	**0.00**
D	2	86428.55	3.78	0.04	3.78	86428.55	3.78	234.79	12.45	809.15	5.78	17.49	2.85	206.49	12.99	86448.31	**0.00**
	5	86427.84	3.16	0.04	3.16	86427.84	3.16	35.12	26.28	119.24	21.67	75.10	14.79	241.96	15.48	86466.61	**0.00**
	10	86427.84	1.61	0.04	2.59	86427.84	1.61	8.04	24.60	656.42	23.75	14.80	45.41	37.98	35.53	86464.28	**0.00**
S	2	86428.95	3.71	0.04	3.71	86428.95	3.71	518.28	13.83	7700.72	5.74	49.64	2.85	408.46	12.02	86451.83	**0.00**
	5	86428.30	2.95	0.04	2.95	86428.30	2.95	29.61	26.42	1267.17	20.84	87.31	15.90	661.87	13.14	86427.65	**0.00**
	10	86428.12	2.63	0.04	2.63	86428.12	2.63	19.91	30.84	1681.79	38.23	40.72	44.90	76.69	35.11	86468.68	**0.00**
A	2	86428.30	2.79	0.04	2.79	86428.30	2.79	798.53	4.91	86402.27	2.72	1291.04	7.08	74404.87	11.16	86434.48	**0.00**
	5	86426.24	2.28	0.04	2.28	86426.24	2.28	66616.11	15.03	86400.12	18.50	20039.72	26.95	86400.25	16.98	86419.49	**0.00**
	10	86426.11	4.03	0.04	4.03	86426.11	4.03	17972.81	10.69	86400.17	16.75	86400.13	31.32	86400.18	29.85	86482.04	**0.00**
Total Average:		86423.23	4.48	0.04	4.59	79633.63	4.48	20891.03	22.03	43937.66	16.34	26240.03	22.45	42689.02	18.92	86449.24	**0.50**

Table 2.6: Heuristic results for instances of $n = 532$, Padberg and Rinaldi (1987)

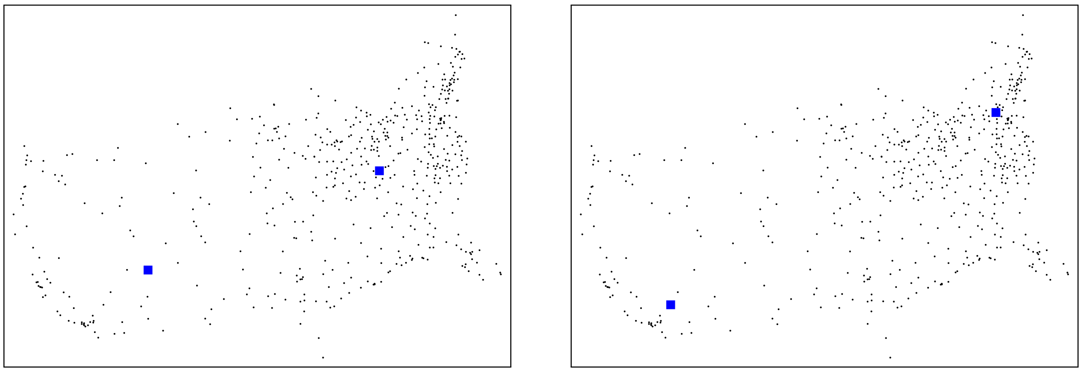
the problem is not completely captured by the discrete version of the problem, the structure of the clusters of demand points obtained by discretizing the space is similar to the one obtained by the exact approach, being this method an adequatate heuristic for larger instances in which the exact approach is not able to certify optimality.

For large-sized problems (Tables 2.6 and 2.7) we set the time limit to 24 hours. The best solutions are found by the discretization matheuristic except for the center problems (see Figure 2.3 where the solutions obtained with the Matheur and the Discretization approaches clearly differ). Among the other strategies, decomposition-based matheuristic stands out, but the improvement from the initial heuristic is null for some cases.

Some instances have the best performance using KMEANS-20 or PTF-20 matheusristics. It is not appreciated a big improvement taking 30 points instead of 20 for the aggregation method. To find an explanation for that, Figure 2.4 depicts the aggregation (triangular points) and the solution for a particular instance. The reader can see how the demand points are concentrated by zones. Adding more points to $\mathcal{A}'$ gives an importance

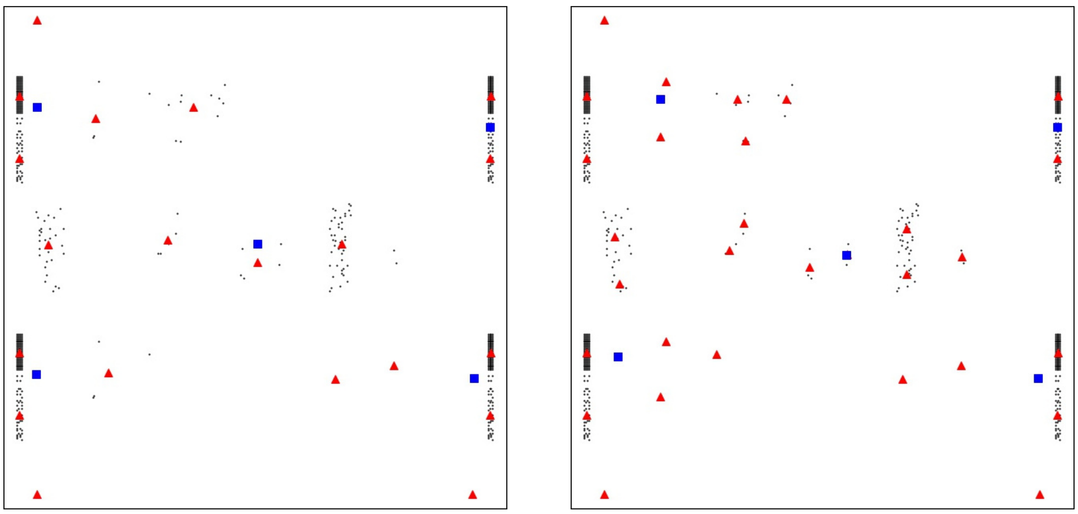
type	p	B&P		InitialHeur		Matheur		KMEANS-20		KMEANS-30		PTF-20		PTF-30		Discretization	
		Time	GAP_{Best}(%)	Time	GAP_{Best}(%)	Time	GAP_{Best}(%)	Time	GAP_{Best}(%)	Time	GAP_{Best}(%)	Time	GAP_{Best}(%)	Time	GAP_{Best}(%)	Time	GAP_{Best}(%)
W	2	86441.18	18.56	0.06	18.56	86441.18	18.56	13.31	8.04	755.99	40.81	32.59	19.19	697.98	40.02	86426.03	**0.00**
	5	86444.11	13.10	0.06	13.68	86444.11	13.10	6.56	45.35	97.79	79.07	7.46	68.54	74.85	76.34	86408.63	**0.00**
	10	86439.34	15.13	0.06	32.85	86439.34	15.13	2.64	101.43	31.49	112.39	1.58	157.47	68.11	133.94	86437.47	**0.00**
C	2	86407.88	4.16	0.07	4.16	86407.32	**0.00**	420.97	8.91	86400.83	3.43	5765.05	**0.00**	86401.06	**0.00**	86436.12	2.27
	5	86407.52	6.54	0.06	8.92	51003.35	4.47	86424.42	32.61	86400.09	**0.00**	86400.15	20.98	86400.09	1.64	86420.51	12.42
	10	86408.46	20.39	0.06	20.39	83212.74	3.23	42425.62	31.02	86400.17	1.36	86400.16	63.01	86400.13	2.00	86419.39	**0.00**
K	2	86424.57	5.30	0.06	5.30	86424.57	5.30	274.39	4.34	40549.49	14.86	365.79	13.07	15942.12	10.65	86404.35	**0.00**
	5	86424.89	11.27	0.06	11.27	86424.89	11.27	854.51	38.53	86400.13	64.01	2868.28	49.15	86400.26	42.60	86441.63	**0.00**
	10	86425.31	32.55	0.06	32.55	86425.31	32.55	6695.50	149.41	86400.19	105.65	13189.85	162.33	86400.14	126.61	86436.97	**0.00**
D	2	86440.98	18.55	0.07	18.55	86440.98	18.55	29.22	6.78	737.65	40.14	23.55	18.85	669.43	39.84	86429.90	**0.00**
	5	86440.05	13.09	0.06	13.67	86440.05	13.09	15.08	46.33	123.71	81.13	6.47	71.68	96.06	82.16	86444.51	**0.00**
	10	86439.96	15.12	0.06	32.79	86439.96	15.12	7.19	99.40	21.26	101.71	3.05	156.24	106.33	133.98	86428.90	**0.00**
S	2	86440.27	17.23	0.07	17.23	86440.27	17.23	19.02	8.52	427.39	37.97	27.17	18.58	515.89	36.72	86424.95	**0.00**
	5	86439.85	12.69	0.07	13.27	86439.85	12.69	9.14	45.19	200.77	77.05	6.68	68.55	143.32	79.93	86421.39	**0.00**
	10	86440.73	15.01	0.06	33.18	86440.73	15.01	6.69	102.11	32.23	108.25	2.44	156.15	107.15	134.09	86437.97	**0.00**
A	2	86439.83	7.51	0.06	7.51	86439.83	7.51	264.51	3.79	11439.37	13.09	416.03	10.15	6083.81	11.98	86432.48	**0.00**
	5	86443.12	9.69	0.06	9.69	86443.12	9.69	256.67	35.94	40907.89	51.62	414.57	43.77	15986.23	42.06	86405.38	**0.00**
	10	86438.72	33.58	0.06	33.58	86438.72	33.58	4315.73	86.09	86400.18	121.59	29377.52	127.07	86400.14	121.93	86407.30	**0.00**
Total Average:		86432.60	14.97	0.06	18.18	84288.13	13.67	7891.18	47.43	34095.92	58.56	12517.13	68.04	31049.62	62.03	86425.77	**0.82**

Table 2.7: Heuristic results for instances of $n = 654$, Reinelt (1992b)



(a) Matheur (b) Discretization

Figure 2.3: Solutions for $n = 532$ (Padberg and Rinaldi, 1987), C, $p = 2$, and ℓ_1-norm



(a) KMEANS-20 (b) KMEANS-30

Figure 2.4: Solutions for $n = 654$ (Reinelt, 1992b), W, $p = 5$, and ℓ_1-norm

to some aggregated points that does not represent properly the original data of this instance of $n = 654$. In this case, we can see an example for which the aggregation algorithm works better under the *less is more* paradigm.

2.6 Computational results for alternative ℓ_τ-norms

In this section, we show the results of our computational experiments for other ℓ_τ-norms, in particular, we have considered $\tau \in \{\frac{3}{2}, 2, 3\}$. We have shown in Theorem 5 the general way to reformulate (**MFMOMP**$_\lambda$) as MISOCO formulation for general values of τ using (ℓ_τ-norm) or polyhedral norms using (Pol-norm). In (ℓ_2-norm) and Table 1.1, the reader can see the sets of constraints for the τ considered, for all $i \in N, j \in P, l \in \{1, \dots, d\}$. Tables 2.8, 2.9, and 2.10 report the results.

n	type	p	Time (#Unsolved)				GAProot(%)		GAP(%)		Vars		Nodes		Memory (MB)	
			Compact		B&P		Compact	B&P	Compact	B&P	Compact	B&P	Compact	B&P	Compact	B&P
20		2	**75.10**	**(0)**	966.81	(0)	100.00	**0.00**	0.00	0.00	524	**500**	31161	**1**	187	**25**
	W	5	—	(5)	**332.73**	**(0)**	100.00	**0.00**	58.11	**0.00**	1270	**397**	2323815	**1**	11580	**12**
		10	—	(5)	**152.66**	**(0)**	100.00	**0.00**	97.34	**0.00**	2480	**377**	1280023	**1**	7438	**10**
		2	**0.70**	**(0)**	520.73	(4)	100.00	**21.46**	**0.00**	17.11	**524**	13259	**317**	650	**9**	174
	C	5	**57.47**	**(0)**	74.10	(4)	100.00	**34.21**	**0.00**	33.65	**1270**	1333	15338	**302**	44	**13**
		10	**3863.55**	**(1)**	1270.00	(2)	100.00	**18.56**	**2.45**	17.81	2480	**1638**	1164203	**1936**	1159	**15**
		2	**6.23**	**(0)**	1227.76	(2)	100.00	**6.46**	**0.00**	2.58	**524**	7247	3052	**197**	**20**	212
	K	5	—	(5)	**824.52**	**(3)**	100.00	**10.90**	27.82	**6.21**	**1270**	2719	3205611	**1120**	7563	**57**
		10	—	(5)	**2409.37**	**(3)**	100.00	**17.66**	96.76	**8.78**	2480	**1235**	1093174	**3003**	8496	**24**
		2	**63.65**	**(0)**	724.11	(0)	100.00	**0.00**	0.00	0.00	524	**437**	31224	**1**	153	**22**
	D	5	—	(5)	**284.21**	**(0)**	100.00	**0.00**	65.95	**0.00**	1270	**383**	2204675	**1**	12441	**11**
		10	—	(5)	**163.89**	**(0)**	100.00	**0.08**	95.40	**0.00**	2480	**370**	1140802	**3**	9200	**9**
		2	**58.41**	**(0)**	1040.72	(0)	100.00	**0.00**	0.00	0.00	524	**519**	25556	**1**	155	**25**
	S	5	—	(5)	**423.48**	**(0)**	100.00	**0.11**	59.03	**0.00**	1270	**397**	2398328	**3**	12161	**12**
		10	—	(5)	**175.32**	**(0)**	100.00	**0.34**	98.74	**0.00**	2480	**383**	878394	**4**	8683	**10**
		2	**16.20**	**(0)**	2802.48	(1)	100.00	**3.44**	**0.00**	0.35	**524**	2987	5372	**54**	**37**	146
	A	5	—	(5)	**2131.87**	**(2)**	100.00	**10.91**	47.25	**3.22**	**1270**	2548	1996233	**413**	8655	**69**
		10	—	(5)	**3440.13**	**(1)**	100.00	**11.84**	99.32	**2.01**	2480	**1288**	705456	**2001**	7031	**30**
30		2	**3007.98**	**(0)**	854.86	(4)	86.97	**19.38**	**0.00**	19.38	**784**	852	1101778	**1**	2823	**71**
	W	5	—	(5)	**3389.46**	**(3)**	87.13	**22.60**	79.53	**22.60**	1900	**845**	744808	**1**	12424	**47**
		10	—	(5)	**2960.90**	**(0)**	89.05	**0.00**	88.07	**0.00**	3710	**773**	341608	**2**	3070	**32**
		2	**1.27**	**(0)**	110.41	(4)	81.07	**30.05**	**0.00**	26.62	**784**	10754	416	**190**	**15**	137
	C	5	**311.13**	**(0)**	—	(5)	81.71	**45.91**	**0.00**	44.26	**1900**	3340	57541	**200**	116	**33**
		10	**4.95**	**(4)**	—	(5)	81.93	**45.96**	75.89	**45.96**	3710	**1758**	407909	**231**	1166	**17**
		2	**77.50**	**(0)**	19.84	(4)	85.63	**39.16**	**0.00**	38.71	**784**	1708	28803	**5**	115	**80**
	K	5	—	(5)	**20.48**	**(4)**	85.80	**18.37**	66.83	**17.99**	**1900**	1986	956059	**19**	11102	**61**
		10	—	(5)	**194.29**	**(4)**	86.29	**22.05**	84.71	**21.13**	3710	**1574**	390091	**234**	2707	**42**
		2	**2441.14**	**(1)**	300.79	(4)	86.55	**22.86**	**2.74**	22.86	**784**	918	1208553	**1**	3490	**77**
	D	5	—	(5)	**4831.03**	**(2)**	86.99	**24.04**	74.25	**24.04**	1900	**841**	798302	**3**	10071	**48**
		10	—	(5)	**2546.68**	**(0)**	87.46	**0.05**	86.29	**0.00**	3710	**774**	343890	**2**	4196	**32**
		2	**2363.80**	**(0)**	470.41	(4)	86.86	**32.35**	**0.00**	32.35	**784**	895	822524	**1**	2187	**72**
	S	5	—	(5)	**4534.78**	**(2)**	86.78	**18.98**	76.88	**18.98**	1900	**840**	726885	**1**	10244	**48**
		10	—	(5)	**2666.23**	**(0)**	88.10	**0.04**	87.16	**0.00**	3710	**776**	410399	**5**	4551	**31**
		2	**327.89**	**(0)**	93.44	(4)	85.31	**49.21**	**0.00**	49.18	**784**	1259	70967	**2**	365	**104**
	A	5	—	(5)	**95.22**	**(4)**	86.39	**7.72**	70.59	**7.37**	1900	**1346**	626210	**9**	8166	**74**
		10	—	(5)	**169.25**	**(4)**	86.14	**13.71**	84.74	**10.99**	3710	**1487**	307003	**141**	2647	**54**
40		2	—	(5)	—	(5)	100.00	**32.33**	45.10	**32.33**	**1044**	1292	1282229	**1**	15092	**131**
	W	5	—	(5)	—	(5)	100.00	**67.86**	96.95	**67.86**	2530	**1298**	364182	**1**	8105	**102**
		10	—	(5)	—	(5)	100.00	**92.83**	100.00	**92.83**	4940	**1339**	165306	**1**	2712	**83**
		2	**4.22**	**(0)**	—	(5)	100.00	**44.51**	**0.00**	44.51	**1044**	2468	1043	**2**	**20**	35
	C	5	**3879.78**	**(3)**	—	(5)	100.00	**77.56**	**41.01**	77.56	2530	**2525**	606460	**2**	4626	**27**
		10	—	(5)	—	(5)	100.00	**78.16**	91.15	**78.16**	4940	**2206**	237096	**13**	883	**22**
		2	**2889.87**	**(0)**	—	(5)	100.00	**61.39**	**0.00**	61.39	**1044**	2226	614033	**1**	2098	**125**
	K	5	—	(5)	—	(5)	100.00	**83.30**	92.54	**83.30**	2530	**2213**	451617	**1**	4187	**94**
		10	—	(5)	—	(5)	100.00	**75.79**	100.00	**75.79**	4940	**2169**	163856	**3**	1240	**83**
		2	—	(5)	—	(5)	100.00	**32.13**	40.95	**32.13**	**1044**	1529	1548255	**1**	11470	**153**
	D	5	—	(5)	—	(5)	100.00	**68.37**	99.29	**68.37**	2530	**1310**	360994	**1**	12324	**99**
		10	—	(5)	—	(5)	100.00	**91.94**	100.00	**91.94**	4940	**1357**	141565	**1**	4573	**79**
		2	—	(5)	—	(5)	100.00	**30.53**	42.63	**30.53**	**1044**	1443	1109568	**1**	14496	**145**
	S	5	—	(5)	—	(5)	100.00	**72.37**	98.23	**72.37**	2530	**1299**	358058	**1**	11144	**97**
		10	—	(5)	—	(5)	100.00	**90.94**	100.00	**90.94**	4940	**1362**	176603	**1**	2825	**76**
		2	**2886.85**	**(4)**	—	(5)	100.00	**61.20**	**14.49**	61.20	**1044**	1876	1127650	**1**	3383	**212**
	A	5	—	(5)	—	(5)	100.00	**82.57**	93.62	**82.57**	2530	**1656**	402572	**1**	4383	**134**
		10	—	(5)	—	(5)	100.00	**75.96**	100.00	**75.96**	4940	**1782**	145643	**1**	2437	**114**
45		2	—	(5)	—	(5)	100.00	**34.65**	46.87	**34.65**	**1174**	1462	974796	**1**	11024	**145**
	W	5	—	(5)	—	(5)	100.00	**83.70**	97.65	**83.70**	2845	**1444**	367765	**1**	3877	**126**
		10	—	(5)	—	(5)	100.00	100.00	100.00	100.00	5555	**1522**	122910	**1**	1081	**101**
		2	**6.63**	**(0)**	—	(5)	100.00	**57.24**	**0.00**	57.24	**1174**	2438	1160	**1**	**23**	34
	C	5	**4337.92**	**(3)**	—	(5)	100.00	**84.59**	**49.89**	84.59	2845	**2696**	467399	**1**	1217	**33**
		10	—	(5)	—	(5)	100.00	**64.84**	96.12	**64.84**	5555	**2327**	135558	**3**	578	**24**
		2	**6688.87**	**(4)**	—	(5)	100.00	**63.44**	**14.13**	63.44	**1174**	2443	1718920	**1**	6733	**142**
	K	5	—	(5)	—	(5)	100.00	**71.76**	95.10	**71.76**	2845	**2663**	453635	**1**	3364	**169**
		10	—	(5)	—	(5)	100.00	**91.29**	99.70	**91.29**	5555	**2231**	158192	**1**	781	**97**
		2	—	(5)	—	(5)	100.00	**30.83**	48.64	**30.83**	**1174**	1508	933735	**1**	10637	**153**
	D	5	—	(5)	—	(5)	100.00	**91.29**	98.36	**91.29**	2845	**1470**	374986	**1**	5117	**129**
		10	—	(5)	—	(5)	100.00	100.00	100.00	100.00	5555	**1528**	159745	**1**	1184	**106**
		2	—	(5)	—	(5)	100.00	**31.76**	45.32	**31.76**	**1174**	1556	1059244	**1**	8872	**160**
	S	5	—	(5)	—	(5)	100.00	**86.56**	99.18	**86.56**	2845	**1477**	452467	**1**	4738	**125**
		10	—	(5)	—	(5)	100.00	100.00	100.00	100.00	5555	**1522**	131588	**1**	976	**100**
		2	—	(5)	—	(5)	100.00	**60.42**	**25.43**	60.42	**1174**	1992	861937	**1**	4928	**221**
	A	5	—	(5)	—	(5)	100.00	**84.00**	97.37	**84.00**	2845	**1822**	297805	**1**	4261	**172**
		10	—	(5)	—	(5)	100.00	**93.77**	100.00	**93.77**	5555	**2008**	141643	**1**	845	**175**
50		2	—	(1)	—	(1)	100.00	**36.39**	57.48	**36.39**	**1304**	1749	759779	**1**	11084	**186**
	W	5	—	(1)	—	(1)	100.00	**88.42**	97.49	**88.42**	3160	**1659**	270228	**1**	3125	**135**
		10	—	(1)	—	(1)	100.00	100.00	100.00	100.00	6170	**1777**	122705	**1**	1843	**127**
		2	**6.06**	**(0)**	—	(1)	100.00	**57.05**	**0.00**	57.05	**1304**	2813	879	**1**	**28**	36
	C	5	—	(1)	—	(1)	100.00	**94.44**	**86.11**	94.44	3160	**2767**	489141	**1**	1274	**41**
		10	—	(1)	—	(1)	100.00	**66.91**	100.00	**66.91**	6170	**2678**	117338	**2**	535	**32**
		2	—	(1)	—	(1)	100.00	**66.85**	**22.51**	66.85	**1304**	2554	1624297	**1**	9722	**141**
	K	5	—	(1)	—	(1)	100.00	**95.48**	**92.64**	95.48	3160	**2458**	501803	**1**	1496	**132**
		10	—	(1)	—	(1)	100.00	**99.57**	100.00	**99.57**	6170	**2418**	128940	**1**	669	**115**
		2	—	(1)	—	(1)	100.00	**31.81**	51.21	**31.81**	**1304**	1704	1059488	**1**	10438	**182**
	D	5	—	(1)	—	(1)	100.00	**91.53**	97.42	**91.53**	3160	**1671**	405884	**1**	1222	**156**
		10	—	(1)	—	(1)	100.00	100.00	100.00	100.00	6170	**1710**	156712	**1**	1360	**127**
		2	—	(1)	—	(1)	100.00	**23.29**	56.18	**23.29**	**1304**	2040	757576	**1**	10234	**215**
	S	5	—	(1)	—	(1)	100.00	**46.71**	98.40	**46.71**	3160	**1772**	404250	**1**	2374	**166**
		10	—	(1)	—	(1)	100.00	100.00	100.00	100.00	6170	**1739**	144453	**1**	764	**131**
		2	—	(1)	—	(1)	100.00	**58.75**	**40.71**	58.75	**1304**	2006	519352	**1**	6759	**215**
	A	5	—	(1)	—	(1)	100.00	**86.23**	95.15	**86.23**	3160	**2175**	396890	**1**	2119	**208**
		10	—	(1)	—	(1)	100.00	100.00	100.00	100.00	6170	**2127**	127758	**1**	672	**183**
Total Average:			1016.47	(282)	**1438.77**	**(277)**	96.64	**44.54**	59.19	**43.82**	2451	**1902**	628495	**143**	4813	**85**

Table 2.8: Results for Eilon et al. (1974) instances for $\ell_{\frac{3}{2}}$-norm

n	type	p	Time (#Unsolved)				GAProot(%)		GAP(%)		Vars		Nodes		Memory (MB)	
			Compact		B&P		Compact	B&P	Compact	B&P	Compact	B&P	Compact	B&P	Compact	B&P
20	W	2	26.03	(0)	24.22	(0)	100.00	0.00	0.00	0.00	204	2385	50632	1	41	120
		5	—	(5)	20.33	(0)	100.00	0.00	52.53	0.00	470	2504	8992191	1	15633	53
		10	—	(5)	9.15	(0)	100.00	0.00	89.75	0.00	880	2178	5306625	1	13982	28
	C	2	0.16	(0)	757.86	(4)	100.00	18.45	0.00	10.76	204	45528	119	13798	4	962
		5	16.30	(0)	—	(5)	100.00	32.78	0.00	21.37	470	7002	13102	7867	31	148
		10	983.51	(0)	2223.12	(3)	100.00	27.39	0.00	10.97	880	3717	650055	9690	1569	75
	K	2	5.03	(0)	1648.82	(1)	100.00	7.02	0.00	1.49	204	11126	6787	349	11	301
		5	4231.23	(4)	2850.11	(2)	100.00	13.04	23.89	2.37	470	5627	8109266	1830	7136	117
		10	—	(5)	1822.27	(2)	100.00	16.01	85.45	6.61	880	2925	4227203	4287	11279	68
	D	2	25.62	(0)	20.60	(0)	100.00	0.00	0.00	0.00	204	2365	45501	1	37	119
		5	—	(5)	15.53	(0)	100.00	0.00	50.45	0.00	470	2495	9156476	1	11659	52
		10	—	(5)	17.64	(0)	100.00	0.05	87.20	0.00	880	2175	5927899	2	15624	28
	S	2	25.31	(0)	28.86	(0)	100.00	0.00	0.00	0.00	204	2445	42178	1	37	126
		5	—	(5)	49.89	(0)	100.00	0.03	49.72	0.00	470	2491	9117840	3	12277	53
		10	—	(5)	24.45	(0)	100.00	0.25	87.53	0.00	880	2183	5399238	4	15145	29
	A	2	15.72	(0)	1663.90	(1)	100.00	3.21	0.00	0.17	204	6400	13975	133	19	288
		5	—	(5)	1603.15	(2)	100.00	9.93	36.09	1.74	470	5066	6284524	703	9720	138
		10	—	(5)	2420.97	(1)	100.00	15.86	87.63	0.94	880	2987	3478065	2125	11095	60
30	W	2	1214.67	(0)	264.29	(0)	86.79	0.00	0.00	0.00	304	3787	1814638	1	559	339
		5	—	(5)	80.27	(0)	86.91	0.00	72.61	0.00	700	2914	3104800	1	12866	107
		10	—	(5)	34.14	(0)	87.86	0.00	84.94	0.00	1310	2474	1626671	1	18935	54
	C	2	0.43	(0)	104.99	(4)	81.07	14.80	0.00	11.58	304	45929	422	1794	6	643
		5	76.89	(0)	503.59	(4)	81.71	27.87	0.00	20.17	700	8380	40315	1409	165	100
		10	1.89	(4)	—	(5)	81.93	38.05	41.89	33.92	1310	4996	2709802	2978	4156	69
	K	2	54.31	(0)	744.55	(4)	85.50	6.60	0.00	6.45	304	7743	62564	44	58	368
		5	—	(5)	5040.71	(4)	86.00	11.54	67.36	8.59	700	5932	2838928	295	15993	170
		10	—	(5)	—	(5)	87.68	17.84	83.29	13.27	1310	4040	1799130	1270	12831	88
	D	2	1247.78	(0)	222.14	(0)	86.38	0.00	0.00	0.00	304	3810	1731910	1	529	335
		5	—	(5)	139.30	(0)	86.94	0.00	73.46	0.00	700	2939	2824067	1	12906	108
		10	—	(5)	38.85	(0)	87.51	0.00	85.28	0.00	1310	2480	1519282	1	16694	55
	S	2	1271.98	(0)	186.78	(0)	86.69	0.00	0.00	0.00	304	3699	1631950	1	518	324
		5	—	(5)	540.52	(0)	87.12	0.02	73.42	0.00	700	2905	3042832	3	13666	109
		10	—	(5)	94.57	(0)	87.66	0.08	84.50	0.00	1310	2465	1569091	2	13710	55
	A	2	220.34	(0)	3254.28	(3)	85.20	2.15	0.00	1.69	304	4997	136052	16	110	403
		5	—	(5)	1745.13	(4)	86.59	5.37	71.23	3.06	700	4216	1909608	51	10892	174
		10	—	(5)	1207.09	(4)	86.97	11.27	83.28	6.38	1310	3735	1103129	465	10680	106
40	W	2	—	(5)	252.49	(0)	100.00	0.00	31.03	0.00	404	6365	7312331	1	5864	817
		5	—	(5)	364.88	(0)	100.00	0.00	94.01	0.00	930	4168	1925927	1	11549	234
		10	—	(5)	355.67	(0)	100.00	0.00	99.63	0.00	1740	3922	862095	1	8294	120
	C	2	1.55	(0)	—	(5)	100.00	23.26	0.00	22.73	404	29032	1164	419	9	448
		5	594.78	(0)	—	(5)	100.00	36.17	0.00	35.69	930	5983	271138	92	862	57
		10	4207.08	(1)	—	(5)	100.00	37.29	11.40	35.81	1740	4780	1357589	266	5116	40
	K	2	719.29	(0)	—	(5)	100.00	8.69	0.00	8.69	404	5651	701127	4	429	412
		5	—	(5)	—	(5)	100.00	16.32	89.79	16.31	930	4724	1771444	4	11673	171
		10	—	(5)	—	(5)	100.00	16.50	99.43	16.47	1740	4836	700210	44	13117	100
	D	2	—	(5)	259.07	(0)	100.00	0.00	30.59	0.00	404	6694	6885185	1	7289	865
		5	—	(5)	485.52	(0)	100.00	0.00	94.27	0.00	930	4240	2040736	1	9260	238
		10	—	(5)	473.11	(0)	100.00	0.01	99.90	0.00	1740	3891	692838	1	10352	119
	S	2	—	(5)	691.25	(0)	100.00	0.02	29.89	0.00	404	6224	6075474	1	5159	806
		5	—	(5)	244.29	(0)	100.00	0.00	94.33	0.00	930	4120	2196020	1	11994	230
		10	—	(5)	1223.93	(0)	100.00	0.07	99.90	0.00	1740	3910	925128	3	11056	120
	A	2	3347.72	(4)	—	(5)	100.00	4.09	9.28	4.09	404	5760	2823594	2	1885	763
		5	—	(5)	—	(5)	100.00	9.56	91.43	9.08	930	4410	1346465	5	6365	258
		10	—	(5)	—	(5)	100.00	9.54	99.86	8.60	1740	4404	677970	41	9029	141
45	W	2	—	(5)	388.20	(0)	100.00	0.00	41.73	0.00	454	9960	4536228	1	6272	1570
		5	—	(5)	207.18	(0)	100.00	0.00	96.23	0.00	1045	4741	1544401	1	10363	360
		10	—	(5)	282.84	(0)	100.00	0.00	100.00	0.00	1955	4318	494940	1	10023	172
	C	2	2.49	(0)	—	(5)	100.00	22.33	0.00	22.32	454	4517	1618	6	13	67
		5	398.50	(0)	—	(5)	100.00	35.75	0.00	35.62	1045	4850	162356	10	453	46
		10	—	(5)	—	(5)	100.00	35.00	76.29	34.76	1955	4710	1628650	35	5534	36
	K	2	5287.03	(2)	—	(5)	100.00	12.13	5.66	12.12	454	6140	4935475	3	2537	535
		5	—	(5)	—	(5)	100.00	17.38	93.84	17.36	1045	5116	2086094	4	9338	226
		10	—	(5)	—	(5)	100.00	14.67	100.00	14.67	1955	4781	618710	5	11268	120
	D	2	—	(5)	358.37	(0)	100.00	0.00	40.50	0.00	454	9220	4961123	1	5740	1483
		5	—	(5)	207.01	(0)	100.00	0.00	96.12	0.00	1045	4756	1837767	1	8596	363
		10	—	(5)	483.48	(0)	100.00	0.00	100.00	0.00	1955	4310	486390	1	9703	172
	S	2	—	(5)	370.88	(0)	100.00	0.00	40.04	0.00	454	9566	4705056	1	5430	1507
		5	—	(5)	1332.07	(0)	100.00	0.20	95.75	0.00	1045	4809	2080272	4	10194	370
		10	—	(5)	1487.71	(0)	100.00	0.02	99.90	0.00	1955	4315	659990	3	10857	173
	A	2	—	(5)	—	(5)	100.00	6.72	25.73	6.26	454	6957	2069919	4	3364	1078
		5	—	(5)	—	(5)	100.00	11.97	91.39	11.12	1045	4895	1615563	6	3655	373
		10	—	(5)	—	(5)	100.00	7.59	100.00	7.46	1955	4486	483721	6	8786	172
50	W	2	—	(1)	456.01	(0)	100.00	0.00	48.21	0.00	504	10414	4777385	1	6240	2007
		5	—	(1)	615.74	(0)	100.00	0.03	96.50	0.00	1160	5458	1930033	3	6233	470
		10	—	(1)	204.96	(0)	100.00	0.00	100.00	0.00	2170	4971	355938	1	7946	232
	C	2	4.78	(0)	—	(1)	100.00	22.61	0.00	22.61	504	4342	2935	2	17	59
		5	—	(1)	—	(1)	100.00	34.42	67.17	34.37	1160	5142	2940681	4	5728	51
		10	—	(1)	—	(1)	100.00	40.27	83.64	40.21	2170	5198	1333733	18	3970	41
	K	2	—	(1)	—	(1)	100.00	11.24	18.49	11.24	504	6665	4998109	3	3635	673
		5	—	(1)	—	(1)	100.00	17.43	91.41	17.43	1160	5674	3154623	3	5249	284
		10	—	(1)	—	(1)	100.00	14.01	100.00	13.91	2170	5463	567705	7	10827	156
	D	2	—	(1)	405.36	(0)	100.00	0.00	47.33	0.00	504	9675	4367022	1	6329	1789
		5	—	(1)	209.77	(0)	100.00	0.00	94.73	0.00	1160	5302	2598004	1	4692	469
		10	—	(1)	605.10	(0)	100.00	0.00	100.00	0.00	2170	5000	548086	1	7504	233
	S	2	—	(1)	451.37	(0)	100.00	0.00	48.39	0.00	504	10286	3512436	1	6039	1986
		5	—	(1)	815.35	(0)	100.00	0.07	96.27	0.00	1160	5236	3055646	3	6241	459
		10	—	(1)	204.63	(0)	100.00	0.01	100.00	0.01	2170	4900	386889	1	9984	231
	A	2	—	(1)	—	(1)	100.00	5.95	32.08	5.36	504	8326	1790159	4	2035	1513
		5	—	(1)	—	(1)	100.00	12.91	91.62	11.92	1160	5559	1874166	7	3311	483
		10	—	(1)	—	(1)	100.00	7.13	100.00	6.88	2170	5089	462300	9	10198	231
Total Average:			673.68	(267)	557.40	(157)	96.65	8.44	53.08	6.79	886	6179	2347788	663	7186	310

Table 2.9: Results for Eilon et al. (1974) instances for ℓ_2-norm

n	type	p	Time (#Unsolved)				GAProot(%)		GAP(%)		Vars		Nodes		Memory (MB)	
			Compact		B&P		Compact	B&P	Compact	B&P	Compact	B&P	Compact	B&P	Compact	B&P
20	W	2	**151.03**	**(0)**	743.17	(0)	100.00	**0.00**	0.00	0.00	524	**471**	56126	**1**	418	**24**
		5	—	(5)	**267.56**	**(0)**	100.00	**0.00**	62.95	**0.00**	1270	**415**	2700255	**1**	13112	**14**
		10	—	(5)	**133.91**	**(0)**	100.00	**0.00**	98.54	**0.00**	2480	**406**	1259405	**1**	13418	**13**
	C	2	**0.91**	**(0)**	—	(5)	100.00	**30.56**	**0.00**	29.19	**524**	9181	507	**417**	**11**	121
		5	**71.86**	**(0)**	—	(5)	100.00	**35.58**	**0.00**	30.56	**1270**	3061	17235	**1473**	62	**32**
		10	**2142.39**	**(0)**	—	(5)	100.00	**47.17**	**0.00**	47.00	2480	**912**	261767	**553**	879	**8**
	K	2	**16.05**	**(0)**	5363.15	(2)	100.00	**9.14**	**0.00**	2.07	**524**	12397	7592	**424**	**40**	352
		5	—	(5)	**2946.11**	**(3)**	100.00	**14.87**	38.13	**5.02**	**1270**	3466	2839547	**1556**	8104	**75**
		10	—	(5)	**3206.51**	**(2)**	100.00	**8.25**	93.62	**5.18**	2480	**1348**	1021924	**2630**	6013	**23**
	D	2	**125.29**	**(0)**	810.43	(0)	100.00	**0.00**	0.00	0.00	524	**498**	53138	**1**	300	**26**
		5	—	(5)	**230.99**	**(0)**	100.00	**0.00**	60.85	**0.00**	1270	**382**	2732256	**1**	12124	**11**
		10	—	(5)	**112.30**	**(0)**	100.00	**0.09**	99.24	**0.00**	2480	**377**	1083407	**2**	11367	**10**
	S	2	**104.35**	**(0)**	973.36	(0)	100.00	**0.05**	0.00	0.00	**524**	530	45754	**1**	239	**27**
		5	—	(5)	**255.50**	**(0)**	100.00	**0.11**	64.54	**0.00**	1270	**398**	2470849	**2**	10763	**12**
		10	—	(5)	**125.04**	**(0)**	100.00	**0.26**	99.51	**0.00**	2480	**378**	1131966	**4**	11380	**10**
	A	2	**33.54**	**(0)**	2705.24	(1)	100.00	**4.22**	**0.00**	0.15	**524**	3049	11374	**71**	**74**	150
		5	—	(5)	**1648.71**	**(2)**	100.00	**8.18**	42.12	**1.90**	**1270**	2262	2117999	**329**	6961	**64**
		10	—	(5)	**4194.46**	**(1)**	100.00	**13.35**	99.33	**1.41**	2480	**1216**	842028	**2054**	8107	**29**
30	W	2	2404.31	(4)	**435.00**	**(4)**	86.64	**14.53**	**12.54**	14.53	**784**	843	2130171	**1**	10112	**65**
		5	—	(5)	**3934.80**	**(2)**	86.35	**5.57**	77.53	**5.57**	1900	**860**	830494	**1**	12706	**48**
		10	—	(5)	**2479.98**	**(0)**	86.89	**0.03**	86.14	**0.00**	3710	**816**	306492	**3**	4764	**37**
	C	2	**2.30**	**(0)**	1631.39	(4)	81.07	**29.54**	**0.00**	29.53	**784**	7137	**675**	7559	**17**	94
		5	**571.24**	**(0)**	2360.18	(4)	81.71	**48.42**	**0.00**	46.18	**1900**	2388	71418	**386**	212	**27**
		10	**8.07**	**(4)**	1801.17	(4)	81.93	**48.01**	71.16	**47.51**	3710	**1236**	316323	**33**	1003	**10**
	K	2	**211.31**	**(0)**	17.36	(4)	85.39	**24.11**	**0.00**	23.96	**784**	1808	86793	**7**	303	**81**
		5	—	(5)	**27.59**	**(4)**	86.11	**14.05**	63.23	**13.64**	1900	**1707**	1143862	**12**	9716	**52**
		10	—	(5)	**34.09**	**(4)**	85.92	**22.77**	85.05	**22.13**	3710	**1409**	365197	**168**	4413	**35**
	D	2	4446.26	(3)	**3777.09**	**(3)**	86.24	**18.03**	**10.14**	18.03	**784**	946	2197143	**1**	9689	**71**
		5	—	(5)	**3574.66**	**(2)**	86.38	**9.19**	74.99	**9.19**	1900	**853**	958686	**1**	12995	**45**
		10	—	(5)	**2472.72**	**(0)**	86.60	**0.06**	85.43	**0.00**	3710	**796**	375899	**3**	5968	**35**
	S	2	**3256.59**	**(3)**	305.27	(4)	86.54	**17.51**	**9.82**	17.51	**784**	850	1889060	**1**	8632	**70**
		5	—	(5)	**3697.87**	**(1)**	86.19	**1.29**	78.49	**1.23**	1900	**760**	873284	**2**	12851	**40**
		10	—	(5)	**2525.29**	**(0)**	87.92	**0.12**	87.51	**0.00**	3710	**791**	351574	**4**	4490	**32**
	A	2	**1365.87**	**(0)**	61.55	(4)	85.11	**21.30**	**0.00**	21.04	**784**	1141	327100	**3**	1255	**96**
		5	—	(5)	**11.83**	**(4)**	85.51	**7.43**	72.18	**6.75**	1900	**1251**	643473	**8**	9003	**67**
		10	—	(5)	**100.35**	**(4)**	85.84	**15.80**	84.79	**13.75**	3710	**1385**	306764	**101**	4475	**51**
40	W	2	—	(5)	—	(5)	100.00	**26.96**	51.29	**26.96**	**1044**	1248	1344963	**1**	16445	**121**
		5	—	(5)	—	(5)	100.00	**83.14**	95.28	**83.14**	2530	**1267**	482105	**1**	5041	**104**
		10	—	(5)	—	(5)	100.00	**92.94**	100.00	**92.94**	4940	**1350**	188692	**1**	3381	**85**
	C	2	**6.66**	**(0)**	—	(5)	100.00	**38.00**	**0.00**	37.95	**1044**	2502	1762	**5**	**22**	37
		5	**2658.74**	**(2)**	—	(5)	100.00	**81.92**	**39.02**	81.92	2530	**2235**	434526	**1**	3523	**25**
		10	—	(5)	—	(5)	100.00	**75.34**	100.00	**75.34**	4940	**1921**	349858	**12**	1348	**18**
	K	2	**4990.55**	**(1)**	—	(5)	100.00	**50.04**	**3.02**	50.04	**1044**	2202	1651655	**1**	5136	**113**
		5	—	(5)	—	(5)	100.00	**74.12**	97.64	**74.12**	2530	**2018**	479613	**1**	7167	**86**
		10	—	(5)	—	(5)	100.00	**69.68**	100.00	**69.68**	4940	**1934**	210396	**3**	2563	**72**
	D	2	—	(5)	—	(5)	100.00	**20.20**	42.95	**20.20**	**1044**	1247	1505294	**1**	12264	**125**
		5	—	(5)	—	(5)	100.00	**89.82**	98.91	**89.82**	2530	**1248**	463628	**1**	8186	**98**
		10	—	(5)	—	(5)	100.00	**90.59**	100.00	**90.59**	4940	**1350**	153036	**1**	5215	**87**
	S	2	—	(5)	—	(5)	100.00	**25.20**	43.94	**25.20**	**1044**	1323	1360254	**1**	12167	**126**
		5	—	(5)	—	(5)	100.00	**70.38**	96.48	**70.38**	2530	**1294**	462667	**1**	6780	**91**
		10	—	(5)	—	(5)	100.00	**84.09**	100.00	**84.09**	4940	**1334**	139121	**1**	6489	**81**
	A	2	—	(5)	—	(5)	100.00	**24.46**	**22.07**	24.46	**1044**	1810	1460678	**1**	5826	**191**
		5	—	(5)	—	(5)	100.00	**62.67**	99.14	**62.67**	2530	**1646**	345643	**1**	7108	**134**
		10	—	(5)	—	(5)	100.00	**85.25**	100.00	**85.25**	4940	**1767**	155190	**1**	3426	**113**
45	W	2	—	(5)	—	(5)	100.00	**26.30**	54.64	**26.30**	**1174**	1398	1133697	**1**	11416	**134**
		5	—	(5)	—	(5)	100.00	**92.43**	99.51	**92.43**	2845	**1413**	401143	**1**	5947	**113**
		10	—	(5)	—	(5)	100.00	100.00	100.00	100.00	5555	**1469**	153297	**1**	2343	**104**
	C	2	**6.40**	**(0)**	—	(5)	100.00	**46.67**	**0.00**	46.67	**1174**	2326	1132	**2**	**24**	34
		5	**5003.88**	**(3)**	—	(5)	100.00	**75.71**	**60.00**	75.71	2845	**2532**	500043	**1**	5891	**29**
		10	—	(5)	—	(5)	100.00	**75.66**	100.00	**75.64**	5555	**2236**	171358	**7**	722	**22**
	K	2	—	(5)	—	(5)	100.00	**44.43**	**28.39**	44.43	**1174**	2477	1611616	**1**	11832	**146**
		5	—	(5)	—	(5)	100.00	**80.52**	99.07	**80.52**	2845	**2482**	426670	**1**	6214	**157**
		10	—	(5)	—	(5)	100.00	**84.97**	100.00	**84.97**	5555	**2165**	160867	**1**	1944	**90**
	D	2	—	(5)	—	(5)	100.00	**24.77**	54.89	**24.77**	**1174**	1406	1171833	**1**	12252	**135**
		5	—	(5)	—	(5)	100.00	**98.15**	98.67	**98.15**	2845	**1412**	424560	**1**	4188	**113**
		10	—	(5)	—	(5)	100.00	100.00	100.00	100.00	5555	**1466**	152963	**1**	2822	**104**
	S	2	—	(5)	—	(5)	100.00	**23.69**	56.00	**23.69**	**1174**	1479	1069851	**1**	11291	**141**
		5	—	(5)	—	(5)	100.00	**72.98**	99.95	**72.98**	2845	**1450**	514193	**1**	5809	**107**
		10	—	(5)	—	(5)	100.00	100.00	100.00	100.00	5555	**1467**	159205	**1**	2847	**101**
	A	2	—	(5)	—	(5)	100.00	**24.73**	37.56	**24.73**	**1174**	1731	695531	**1**	6228	**178**
		5	—	(5)	—	(5)	100.00	**65.14**	97.05	**65.14**	2845	**1800**	295359	**1**	4504	**160**
		10	—	(5)	—	(5)	100.00	**95.05**	100.00	**95.05**	5555	**2027**	131122	**1**	1477	**172**
50	W	2	—	(1)	—	(1)	100.00	**75.57**	**61.16**	75.57	**1304**	1616	1047768	**1**	11081	**161**
		5	—	(1)	—	(1)	100.00	**99.83**	100.00	**99.83**	3160	**1623**	406553	**1**	2969	**141**
		10	—	(1)	—	(1)	100.00	100.00	100.00	100.00	6170	**1709**	186637	**1**	934	**128**
	C	2	**11.71**	**(0)**	—	(1)	100.00	**62.36**	**0.00**	62.36	**1304**	2575	2112	**1**	38	**37**
		5	—	(1)	—	(1)	100.00	**90.14**	**87.76**	90.14	3160	**2577**	540242	**1**	1441	**34**
		10	—	(1)	—	(1)	100.00	**99.12**	100.00	**99.12**	6170	**2587**	112265	**1**	572	**26**
	K	2	—	(1)	—	(1)	100.00	**36.29**	46.27	**36.29**	**1304**	2810	957166	**1**	10628	**164**
		5	—	(1)	—	(1)	100.00	**91.13**	98.89	**91.13**	3160	**2450**	434558	**1**	4069	**127**
		10	—	(1)	—	(1)	100.00	**90.03**	100.00	**90.03**	6170	**2410**	142538	**1**	1231	**108**
	D	2	—	(1)	—	(1)	100.00	**31.05**	63.12	**31.05**	**1304**	1666	878351	**1**	10528	**162**
		5	—	(1)	—	(1)	100.00	**74.10**	100.00	**74.10**	3160	**1695**	285611	**1**	4410	**134**
		10	—	(1)	—	(1)	100.00	100.00	100.00	100.00	6170	**1777**	83410	**1**	545	**129**
	S	2	—	(1)	—	(1)	100.00	**36.71**	60.33	**36.71**	**1304**	1577	983547	**1**	10718	**148**
		5	—	(1)	—	(1)	100.00	**94.34**	99.55	**94.34**	3160	**1635**	583530	**1**	2230	**129**
		10	—	(1)	—	(1)	100.00	100.00	100.00	100.00	6170	**1737**	153870	**1**	775	**135**
	A	2	—	(1)	—	(1)	100.00	**24.67**	48.59	**24.67**	**1304**	2258	514472	**1**	5470	**226**
		5	—	(1)	—	(1)	100.00	**75.58**	100.00	**75.58**	3160	**1998**	403520	**1**	2368	**182**
		10	—	(1)	—	(1)	100.00	**91.23**	100.00	**91.23**	6170	**2051**	117660	**1**	1968	**170**
Total Average:			928.16	(292)	**1680.95**	**(276)**	96.54	**41.27**	61.20	**40.52**	2451	**1819**	711082	**237**	5823	**82**

Table 2.10: Results for Eilon et al. (1974) instances for ℓ_3-norm

2.7 Aggregated results

In order to show the influence of p, the Ordered Median aggregation function, and the ℓ_τ-norm, we have aggregated the results of the 1512 instances used in Tables 2.4, 2.8, 2.9, and 2.10.

n	p	Time (#Unsolved)				GAProot(%)		GAP(%)		Vars		Nodes		Memory (MB)	
		Compact		B&P		Compact	B&P	Compact	B&P	Compact	B&P	Compact	B&P	Compact	B&P
20	2	**31.53**	**(0)**	995.23	(27)	97.64	**5.78**	**0.00**	3.18	**352**	11676	19115	**909**	**76**	311
	5	598.89	(77)	**645.63**	**(36)**	100.00	**9.40**	31.41	**5.29**	**841**	3424	4014722	**1775**	6812	**86**
	10	1887.11	(100)	**944.99**	**(27)**	100.00	**10.49**	68.18	**5.25**	**1623**	1837	4961903	**2579**	7318	**38**
30	2	**800.60**	**(11)**	782.84	(68)	86.56	**15.67**	**1.47**	14.99	**527**	12187	724493	**448**	1706	**341**
	5	605.00	(98)	**1716.72**	**(59)**	89.26	**13.33**	54.38	**11.70**	**1258**	4621	3293952	**259**	8505	**124**
	10	1688.00	(115)	**1273.04**	**(52)**	89.74	**12.60**	77.60	**10.90**	**2427**	2794	3021553	**672**	5907	**64**
40	2	**1269.05**	**(71)**	637.73	(92)	97.42	**22.37**	**18.71**	22.30	**702**	10587	3899173	**30**	5302	**452**
	5	1285.99	(105)	**920.48**	**(92)**	100.00	**43.95**	76.39	**43.86**	**1674**	4274	2224552	**13**	7266	**162**
	10	3970.16	(115)	**558.03**	**(90)**	100.00	**47.30**	88.85	**46.92**	**3228**	3561	1806857	**63**	4752	**102**
45	2	**1404.32**	**(86)**	417.24	(94)	97.47	**24.07**	27.13	**24.05**	**789**	9011	3575284	**6**	5658	**592**
	5	1528.56	(106)	**914.34**	**(99)**	100.00	**47.27**	79.86	**47.20**	**1883**	4121	2010197	**4**	5773	**216**
	10	—	(120)	**900.91**	**(90)**	100.00	**50.95**	92.38	**50.81**	**3630**	3778	1503232	**18**	3886	**130**
50	2	231.73	(19)	**414.88**	**(18)**	97.50	**26.53**	34.60	**26.50**	**877**	8218	3270850	**3**	5870	**733**
	5	379.06	(23)	**1296.44**	**(19)**	100.00	**49.11**	85.88	**49.05**	**2091**	4208	2014876	**3**	4056	**261**
	10	—	(24)	**1140.29**	**(18)**	100.00	**53.29**	93.65	**53.22**	4033	**3983**	1167240	**13**	3434	**158**
Total Average:		788.01	(1070)	**950.76**	**(881)**	96.64	**26.11**	52.32	**24.78**	**1614**	5964	2567113	**538**	5209	**226**

Table 2.11: Results for Eilon et al. (1974) instances disaggregated by p

n	type	Time (#Unsolved)				GAProot(%)		GAP(%)		Vars		Nodes		Memory (MB)	
		Compact		B&P		Compact	B&P	Compact	B&P	Compact	B&P	Compact	B&P	Compact	B&P
20	W	317.70	(36)	**223.81**	**(0)**	99.49	**0.04**	42.20	**0.00**	**939**	1358	4396856	**1**	6710	**40**
	C	**586.30**	**(3)**	1167.87	(50)	98.24	**29.65**	**0.84**	22.22	**939**	16897	840935	**6801**	376	**368**
	K	208.19	(33)	**2302.80**	**(26)**	99.29	**12.07**	33.55	**4.07**	**939**	8287	2776965	**2663**	4251	**214**
	D	490.31	(35)	**202.43**	**(0)**	99.48	**0.05**	42.83	**0.00**	**939**	1345	3918058	**2**	6411	**39**
	S	246.13	(36)	**264.52**	**(0)**	99.48	**0.16**	42.80	**0.00**	**939**	1364	3810920	**3**	6202	**41**
	A	94.15	(34)	**2244.67**	**(14)**	99.30	**9.39**	36.97	**1.16**	**939**	4622	2247746	**1058**	4463	**168**
30	W	1513.57	(44)	**1106.95**	**(13)**	89.87	**5.18**	52.47	**5.17**	**1404**	1895	3535132	**1**	7636	**102**
	C	**298.00**	**(15)**	1014.47	(52)	84.44	**34.66**	**17.14**	30.95	**1404**	21207	1107742	**2046**	725	**263**
	K	**223.54**	**(39)**	1637.34	(48)	88.77	**18.00**	44.82	**15.96**	**1404**	8338	2429087	**501**	5244	**283**
	D	1618.90	(44)	**1283.18**	**(11)**	89.57	**6.19**	52.42	**6.18**	**1404**	1905	2728170	**1**	7388	**103**
	S	1512.19	(43)	**1267.56**	**(11)**	89.80	**5.88**	52.85	**5.84**	**1404**	1883	2558116	**2**	6886	**101**
	A	**718.27**	**(39)**	1701.10	(44)	88.68	**13.30**	47.19	**11.06**	**1404**	3974	1721747	**207**	4355	**208**
40	W	4028.70	(59)	**774.14**	**(30)**	99.48	**33.01**	72.86	**33.00**	**1868**	2957	5069956	**1**	7853	**233**
	C	**980.10**	**(20)**	—	(60)	97.96	**50.13**	**26.17**	49.58	**1868**	19156	881914	**158**	1541	**215**
	K	**2015.89**	**(41)**	—	(60)	99.22	**42.57**	59.51	**42.46**	**1868**	5091	1589810	**32**	4656	**226**
	D	5908.68	(59)	**461.24**	**(32)**	99.47	**32.77**	72.91	**32.76**	**1868**	3053	3668574	**1**	8229	**244**
	S	4977.44	(59)	**865.44**	**(32)**	99.47	**31.20**	72.88	**31.17**	**1868**	2965	3231789	**2**	8253	**232**
	A	**1271.93**	**(53)**	—	(60)	99.24	**37.56**	63.59	**37.20**	**1868**	3620	1419121	**19**	4108	**281**
45	W	—	(60)	**545.90**	**(33)**	99.50	**36.46**	76.57	**36.45**	**2101**	3670	4325838	**1**	6928	**371**
	C	**631.76**	**(26)**	—	(60)	97.92	**50.86**	**34.95**	50.71	**2101**	14393	812873	**35**	1369	**165**
	K	**2695.65**	**(51)**	—	(60)	99.27	**44.77**	64.85	**44.75**	**2101**	4696	1888145	**9**	5273	**253**
	D	—	(60)	**565.54**	**(33)**	99.50	**37.10**	77.13	**37.10**	**2101**	3542	3091420	**1**	6813	**355**
	S	—	(60)	**1176.85**	**(37)**	99.49	**34.68**	77.24	**34.64**	**2101**	3672	2778561	**2**	6696	**369**
	A	**4681.25**	**(55)**	—	(60)	99.27	**40.71**	68.02	**40.45**	**2101**	3846	1280588	**7**	3555	**361**
50	W	—	(12)	**504.08**	**(6)**	99.51	**41.69**	79.67	**41.68**	**2333**	4106	4060816	**1**	5964	**479**
	C	**80.39**	**(7)**	—	(12)	97.92	**56.37**	**47.61**	56.32	**2333**	11574	985006	**25**	1392	**139**
	K	**1135.78**	**(11)**	—	(12)	99.27	**48.78**	68.21	**48.77**	**2333**	4721	1721732	**3**	4630	**280**
	D	—	(12)	**1231.14**	**(6)**	99.51	**35.71**	79.72	**35.71**	**2333**	4047	2644819	**1**	5596	**464**
	S	—	(12)	**1080.38**	**(7)**	99.50	**33.47**	80.29	**33.47**	**2333**	4097	2304729	**2**	5704	**477**
	A	—	(12)	—	(12)	99.29	**41.84**	72.76	**41.59**	**2333**	4273	1188832	**4**	3434	**466**
Total Average:		788.01	(1070)	**950.76**	**(881)**	96.64	**26.11**	52.32	**24.78**	**1614**	5964	2567113	**538**	5209	**226**

Table 2.12: Results for Eilon et al. (1974) instances disaggregated by `type`

n	norm	Time (#Unsolved)				GAProot(%)		GAP(%)		Vars		Nodes		Memory (MB)	
		Compact		B&P		Compact	B&P	Compact	B&P	Compact	B&P	Compact	B&P	Compact	B&P
20	ℓ_1	436.31	(27)	**536.32**	**(21)**	96.86	**9.14**	12.95	**3.27**	**387**	12007	6217790	**3685**	959	**322**
	$\ell_{\frac{3}{2}}$	431.87	(51)	**988.85**	**(22)**	100.00	**7.55**	41.56	**5.10**	**1425**	2112	1027930	**538**	5279	**49**
	ℓ_2	237.07	(49)	**695.93**	**(21)**	100.00	**8.00**	36.12	**3.13**	**518**	6200	3712315	**2266**	6961	**154**
	ℓ_3	330.68	(50)	**1287.62**	**(26)**	100.00	**9.55**	42.16	**6.80**	**1425**	2264	1036285	**529**	5743	**56**
30	ℓ_1	507.49	(51)	**719.49**	**(33)**	96.81	**8.47**	33.60	**5.77**	**582**	16294	6498700	**857**	2749	**394**
	$\ell_{\frac{3}{2}}$	**1149.07**	**(55)**	2365.89	(57)	85.90	**22.91**	48.76	**22.36**	2131	**1818**	519097	**58**	4414	**59**
	ℓ_2	567.61	(54)	**450.17**	**(37)**	85.92	**7.53**	45.62	**5.84**	**771**	6524	1636955	**463**	8071	**200**
	ℓ_3	1098.91	(64)	**2343.41**	**(52)**	85.46	**16.54**	49.94	**16.14**	2131	**1499**	731911	**461**	6256	**53**
40	ℓ_1	895.06	(66)	**941.95**	**(49)**	96.56	**11.16**	44.14	**10.67**	**772**	14886	7295651	**89**	3794	**429**
	$\ell_{\frac{3}{2}}$	**1932.07**	**(77)**	—	(90)	100.00	**67.76**	69.78	**67.76**	2838	**1742**	514263	**2**	5889	**101**
	ℓ_2	1337.71	(70)	**483.36**	**(45)**	100.00	**8.97**	59.71	**8.75**	**1025**	6284	2142580	**49**	7183	**330**
	ℓ_3	**2330.98**	**(78)**	—	(90)	100.00	**63.60**	71.65	**63.60**	2838	**1650**	621616	**2**	6227	**95**
45	ℓ_1	1292.16	(70)	**984.00**	**(58)**	96.63	**11.66**	49.03	**11.45**	**868**	13141	6512940	**30**	4305	**523**
	$\ell_{\frac{3}{2}}$	**1924.73**	**(82)**	—	(90)	100.00	**73.90**	**72.99**	73.90	3191	**1895**	489638	**1**	3902	**123**
	ℓ_2	1374.31	(77)	**568.64**	**(45)**	100.00	**9.10**	66.84	**8.98**	**1151**	5691	1939349	**5**	6785	**490**
	ℓ_3	**1434.25**	**(83)**	—	(90)	100.00	**68.40**	76.98	**68.40**	3191	**1819**	509691	**1**	5431	**113**
50	ℓ_1	505.06	(15)	**1480.61**	**(10)**	96.66	**11.82**	53.46	**11.72**	**966**	11475	5577448	**18**	4265	**634**
	$\ell_{\frac{3}{2}}$	**6.06**	**(17)**	—	(18)	100.00	**74.63**	77.52	**74.63**	3545	**2101**	443749	**1**	3651	**140**
	ℓ_2	4.78	(17)	**440.92**	**(9)**	100.00	**9.23**	73.10	**9.11**	**1278**	6261	2147547	**4**	5899	**632**
	ℓ_3	**11.71**	**(17)**	—	(18)	100.00	**76.23**	81.43	**76.23**	3545	**2042**	435212	**1**	3999	**130**
Total Average:		788.01	(1070)	**950.76**	**(881)**	96.64	**26.11**	52.32	**24.78**	**1614**	5964	2567113	**538**	5209	**226**

Table 2.13: Results for Eilon et al. (1974) instances disaggregated by norm

2.8 Conclusions

In this chapter, the Continuous Multifacility Monotone Ordered Median Problem is analyzed. This problem finds solutions in a continuous space and to solve it we have proposed two exact methods, namely a compact formulation and a branch-and-price procedure, using binary variables. Along the chapter, we give full details of the branch-and-price algorithm and all its crucial steps: master problem, restricted relaxed master problem, pricing problem, initial pool of columns, feasibility, convergence, and branching.

Moreover, theoretic and empirical results have proven the utility of the obtained lower bound. Using that bound, we have tested three matheuristics that we propose. The decomposition-based heuristics have shown a very good performance on the computational experiments. For large-sized instances, the best known solutions have been obtained reducing the solution space by means of a discretization of the continuous problem.

Among the extensive computational experiments and configurations of the problem, we highlight the usefulness of the branch-and-price approach for medium- to large-sized instances, but also the utility of the compact formulation and the aggregation-based heuristics for small values of p or for some particular ordered weighted median functions.

Further research on the topic includes the design of similar branch-and-price approaches to other continuous facility location and clustering problems. Specifically, the application of set-partitioning column generation methods to hub location and covering problems with generalized upgrading (see, e.g., Blanco and Marín, 2019) where the index set for the y-variables must be adequately defined.

Chapter 3

Continuous maximal covering location problems with interconnected facilities

In this chapter we analyze a continuous version of the maximal covering location problem, in which the facilities are required to be linked by means of a given graph structure (provided that two facilities are allowed to be linked if a given distance is not exceed). We propose a mathematical programming framework for the problem and different resolution strategies. First, we provide a Mixed Integer Non Linear Programming formulation for the problem and derive some geometrical properties that allow us to reformulate it as an equivalent pure integer linear programming problem. We propose two branch-and-cut approaches by relaxing some sets of constraints of the former formulation. We also develop a matheuristic algorithm for the problem capable to solve instances of larger sizes. We report the results of an extensive battery of computational experiments comparing the performance of the different approaches.

3.1 Introduction

In many practical situations in location analysis, the facilities to be located are required to be interconnected. This is the case of the optimal design of forest fire-fighters centers that must be communicated to a central server at a give radius (Demaine et al., 2009) or in the location of sensors that have to be connected to each others (Romich et al., 2015). Some facility location problems have been analyzed with some kind of interconnection between facilities (see, e.g., Blanco et al., 2016), although interconnection between services has been mostly studied in the context of hub location, in which the routing costs induce such a connectivity between the hub nodes (see Contreras and O'Kelly, 2019, and the references therein). Another possibility is to consider that an interconnection graph for the centers has to be constructed provided that two facilities are allowed to be linked if the distance between them does not exceed a given limit. The incorporation of this type of interconnection to discrete facility location problems has been recently proposed by Cherkesly et al. (2019). There, the authors provide a model to simultaneously decide the facilities to open as well as spanning tree of the open facilities (where connection between nodes is allowed only when a given distance is not exceed) with both the p-median and the p-center problems. However, as far as we know, there are no previous attempts to deal with this notion of interconnection in continuous facility location problems.

In this chapter we analyze a novel version of the continuous maximal covering location problem, called the Maximal Covering Location Problem with Interconnected Facilities whose goals are: (1) to determine the positions of p facilities in a d-dimensional space in order to maximize the weighted number of covered points; and (2) to decide the activated links between the new facilities following the particularities of a given graph structure, provided that two facilities are allowed to be linked if the distance between them does not exceed a given radius. Cherkesly et al. (2019) analyze the discrete version of this problem in case the given graph structure is a spanning tree of the facilities. In contrast, we consider a more general framework in which different graph structures can be given to construct the network of facilities. This model allows the decision maker to choose the most appropriate graph to link the facilities. For instance, one may consider λ-connected graphs as the input structure, being the tree-shaped interconnection a particular case of our problem. Thus, we provide here a flexible modeling approach to incorporate interconnection in continuous facility location problems, which can be adapted to different useful situations.

We further analyze the following useful input types of spanning graphs for the interconnected facilities: complete, cycle, matching, star, ring-star and line, although one may also consider spanning trees or other λ-connected structures. The importance of the graphs that we analyze in more detail comes from the usefulness of them in different situations. For instance, complete graphs induce conservative robust networks of facilities under failures since all the facilities are at most at a limit distance from the others. Matching-shaped structures allow also robust distribution networks in which each facility is supported by its *paired facility*. Cycle- and ring-star-shaped graphs are useful in the design of telecommunications networks where an alternative path is assured to provide the service in case of failure (Contreras et al., 2017; Labbé et al., 2004). Star-shaped interconnections are appropriate when one facility acts as a main server linked to all the others. This is the case of a main switch that provide service to wifi routers. All the routers are linked to this switch and a star structure is adequate to design a network with high quality connections of users. Also, in Digital Data Service in which one desires to connect terminals to hub servers by point-to-point links and also to connect the servers through a ring structure (Labbé et al., 2004; Xu et al., 1999).

The main contributions of this chapter are:

- We propose a general framework for the Maximal Covering Location Problem with Interconnected Facilities (MCLPIF) in a d-dimensional space, in which the distance/costs are measured by means of ℓ_τ ($\tau \geq 1$) or polyhedral based-norms and where general spanning graph structures are required to link the facilities.

- We derive a Mixed Integer Non-Linear Programming (MINLP) formulation for the problem and show how it can be reformulated as a Mixed Integer Second Order Cone Optimization (MISOCO) problem that can be solved using any of the available off-the-shelf solvers (Gurobi, CPLEX, XPRESS, etc.).

- We provide a decomposition of the problem by exploiting its geometry. We prove that the MCLPIF is equivalent to solve a pure Integer Linear Programming (ILP) problem and use its solution to derive a Second Order Cone programming problem, whose optimal solution coincides with the one of the MCLPIF.

- We further analyze the planar Euclidean MCLPIF and derive two branch-and-cut approaches for solving it.

- We develop a novel matheuristic approach for the problem by using aggregation techniques and simplified ILP formulations.

The rest of the chapter is organized as follows. The MCLPIF is introduced Section 3.2. We provide a MINLP formulation for the MCLPIF and reformulate it as a MISOCO problem. Section 3.3 is devoted to analyze the geometry of the MCLPIF and reformulate it as an ILP problem. In Section 3.4 we derive two branch-and-cut approaches for solving the planar Euclidean MCLPIF by means of relaxing subsets of constraints of the ILP formulation. In Section 3.5 we describe our matheuristic approach for solving the problem. The results of our computational experience are reported in Section 3.6 showing the performance of the different proposed approaches. Finally, we draw some conclusions and point out some further research topics.

3.2 The Maximal Covering Location Problem with Interconnected Facilities

In this section we introduce the Maximal Covering Location Problem with Interconnected Facilities (MCLPIF) and fix the notation for the rest of the chapter. In this problem, apart from the requisites of the MCLP defined in Section 1.2.3, the centers are required to be linked, within a maximum allowed distance, $r \geq 0$, and following the specifications of a given graph structure. As already mentioned, this interconnection is useful in many practical situations in which communication between facilities is required for the adequate performance of the system.

As in Section 1.2.3, we are given metric space $\mathbb{R}^d$ with a metric $\|\cdot\|$, a finite set of demand points $\mathcal{A} = \{a_1, \ldots, a_n\} \subset \mathbb{R}^d$, indexed by $N = \{1, \ldots, n\}$ and each point $a_i \in \mathcal{A}$ has associated a positive demand weight ω_i. As we said, the coverage radius can be seen as the limit to provide service by the facility or as the limit that the user is willing to reach. In this chapter, we assume the later, thus each demand point $a_i \in \mathcal{A}$ has also associated a nonnegative coverage radius R_i. We denote by G the undirected complete graph with nodes $P = \{1, \ldots, p\}$ and $\mathcal{S}(G)$ certain undirected spanning subgraph structure of G. In this subgraph structure the decision maker incorporates the desired particularities for the topology of the network of facilities.

With the notation above, the goals of the MCLPIF are:

- To determine the positions of p facilities in a d-dimensional space, $X_1, \ldots, X_p \in \mathbb{R}^d$, that maximize the weighted covered points, provided that the coverage area of the facilities is determined by the radius R_i, for $i \in N$, and

- To decide the activated links between the new facilities following the particularities of the structure $\mathcal{S}(G)$, provided that two facilities can be interconnected if the distance between them does not exceed the radius r.

Lemma 2. *The MCLPIF is NP-hard.*

Proof. The MCLPIF reduces to the MCLP in case $\mathcal{S}(G)$ is a 0-connected subgraph or $\mathcal{S}(G)$ is any spanning subgraph and r big enough, i.e., any link is possible between the facilities. Thus, by the work of Church (1984), the problem reduces to a discrete MCLP which is known to be NP-hard (Megiddo et al., 1983). □

We provide here a very general framework for the problem that can be adapted to the different situations in which it may be applied. In what follows, we give a suitable mathematical programming formulation for the MCLPIF. Apart from the decisions on the demand points that are covered by the facilities, i.e., the z-variables used in Section 1.2.3, we use the following set of binary variables to decide on the activated links between the facilities:

$$x_{jk} = \begin{cases} 1 & \text{if facilities } k \text{ and } j \text{ are linked in } \mathcal{S}(G), \\ 0 & \text{otherwise,} \end{cases} \quad \text{for all } j,k \in P, j < k.$$

With these sets of variables, the MCLPIF can be formulated as the following Mixed Integer Non Linear Programming problem, that we call (MCLPIF$^{\text{NL}}$):

$$\begin{aligned}
\max & \sum_{i\in N} \omega_i \sum_{j\in P} z_{ij} \\
\text{s.t.} & \sum_{j\in P} z_{ij} \le 1, \; \forall i \in N, && (\text{MCLPIF}_1^{\text{NL}})\\
& \|X_j - a_i\| \le R_i, \text{ if } z_{ij} = 1, \; \forall i \in N, j \in P, && (\text{MCLPIF}_2^{\text{NL}})\\
& \|X_j - X_k\| \le r, \text{ if } x_{jk} = 1, \; \forall j,k \in P, j < k, && (\text{MCLPIF}_3^{\text{NL}})\\
& x \in \mathcal{S}(G), && (\text{MCLPIF}_4^{\text{NL}})\\
& z_{ij} \in \{0,1\}, \; \forall i \in N, j \in P, \\
& x_{jk} \in \{0,1\}, \; \forall j,k \in P, j < k, \\
& X_j \in \mathbb{R}^d, \; \forall j \in P.
\end{aligned}$$

In the above formulation, apart from the constraints of the MCLP (constraints (MCLPIF$_1^{\text{NL}}$) and (MCLPIF$_2^{\text{NL}}$)), constraints (MCLPIF$_3^{\text{NL}}$) ensure that the facilities are allowed to be linked only in case the distance between them is smaller or equal to r. In (MCLPIF$_4^{\text{NL}}$) we incorporate all the constraints ensuring the desired properties of the spanning subgraph of facilities $\mathcal{S}(G)$. Similarly to constraints defined en Chapter 1.2.3, the family of constraints (c-MCLP$_6^{\text{NL}}$), (MCLPIF$_3^{\text{NL}}$) can be equivalently rewritten as:

$$\|X_j - X_k\| \le r + M(1 - x_{jk}), \; \forall j,k \in P, j < k,$$

for a big enough constant $M > \max_{i,j\in N} \|a_i - a_j\| + 2r$.

3.2.1 Spanning subgraphs of facilities

The spanning subgraph of the facilities, $\mathcal{S}(G)$, represents the requirements of the decision maker on the desired network. Several options are possible when designing such a network. We will focus on network structures that can be modeled by fixing the values of the x-variables in (MCLPIF$^{\text{NL}}$). However, at the end of this section we will describe other possibilities that can be considered.

Observe that in the continuous MCLP (and also in the MCLPIF), in contrast to its discrete counterpart, the *labels* $\{1,\ldots,p\}$ assigned to the facilities are arbitrary, in the sense that any permutation of the labels is an alternative solution of the problem. Taking into account this observation we consider the following six spanning graph structures and their incorporation, in terms of the x-variables, into the problem:

- **Complete**. A complete graph can be model by fixing the values of all the x-values to one, i.e., $x_{jk} = 1$ for all $j, k \in P$, $j < k$.
- **Cycle**. A spanning cycle of facilities can be set as:

$$x_{jk} = \begin{cases} 1 & \text{if } (k = j+1) \text{ or } (j = 1 \text{ and } k = p), \\ 0 & \text{otherwise.} \end{cases}$$

- **Line**. Similarly to cycles, one can enforce a line (path) by setting:

$$x_{jk} = \begin{cases} 1 & \text{if } k = j+1, \\ 0 & \text{otherwise.} \end{cases}$$

- **Star**. The star shape on the sets of nodes $\{1,\ldots,p\}$ can be enforced by fixing the central node of the star to 1 and the links as:

$$x_{jk} = \begin{cases} 1 & \text{if } j = 1 \text{ and } k \neq 1, \\ 0 & \text{otherwise.} \end{cases}$$

- **Ring-Star**. The structure of a ring-star graph on $\{1,\ldots,p\}$ can be incorporated, similarly to the star as:

$$x_{jk} = \begin{cases} 1 & \text{if } (j = 1 \text{ and } k \neq 1) \text{ or } (j \neq 1 \text{ and } k = j+1) \text{ or } (j = 2 \text{ and } k = p), \\ 0 & \text{otherwise.} \end{cases}$$

- **Matching**. A perfect matching on $\{1,\ldots,p\}$ (with even p) is a pairwise group of the vertices, and then, in our case, can be set as:

$$x_{jk} = \begin{cases} 1 & \text{if } j \text{ is odd and } k = j+1, \\ 0 & \text{otherwise.} \end{cases}$$

In Figure 3.1 we show the shapes of the six spanning subgraphs described above.

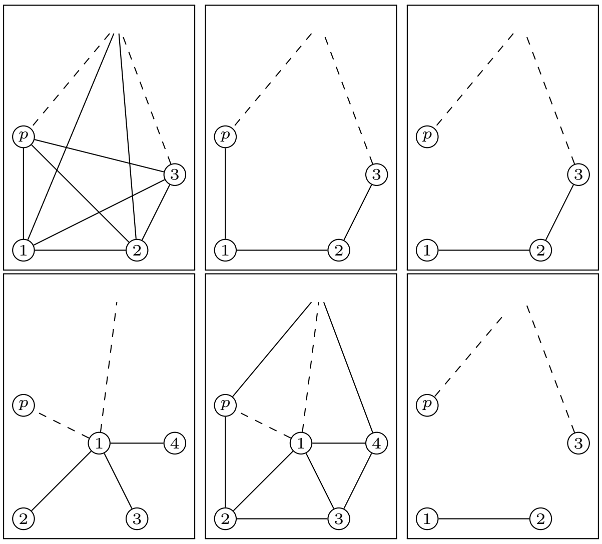


Figure 3.1: Shapes for the spanning graphs of facilities. From left to right: Complete, Cycle, Line, Star, Ring-Star, and Matching.

Example 1. *We consider a set of* 15 *demand points on the plane,* $\mathcal{A} = \{(0.34, 0.59), (0.13, 0.90), (0.67, 0.53), (0.41, 0.03), (0.36, 0.20), (0.09, 0.10), (0.29, 1.), (0.68, 0.56), (0.08, 0.50), (0.86, 0.71), (0.66, 0.63), (0.87, 0.05), (0.22, 0.44), (0.22, 0.11), (0.11, 0.53)\}$, $R_i = 0.1$, *for all* $i = 1, \ldots, 15$, $r = 0.3$, $\omega_i = 1$, *for all* i *and* $\|\cdot\|$ *the Euclidean distance. A solution of the classical MCLP for* $p = 6$ *is drawn in Figure 3.2. On the other hand, the solutions for the MCLPIF for the same number of facilities and shapes (Complete, Cycle, Line, Star, Ring-Star and Matching) are drawn in Figure 3.3. Demand points are drawn with solid circles while optimal facilities are drawn with asterisks. Links in the solutions are represented by segments joining the asterisks in the pictures.*

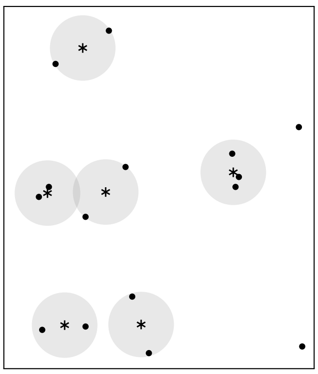


Figure 3.2: Solution to the MCLP for the data of Example 1.

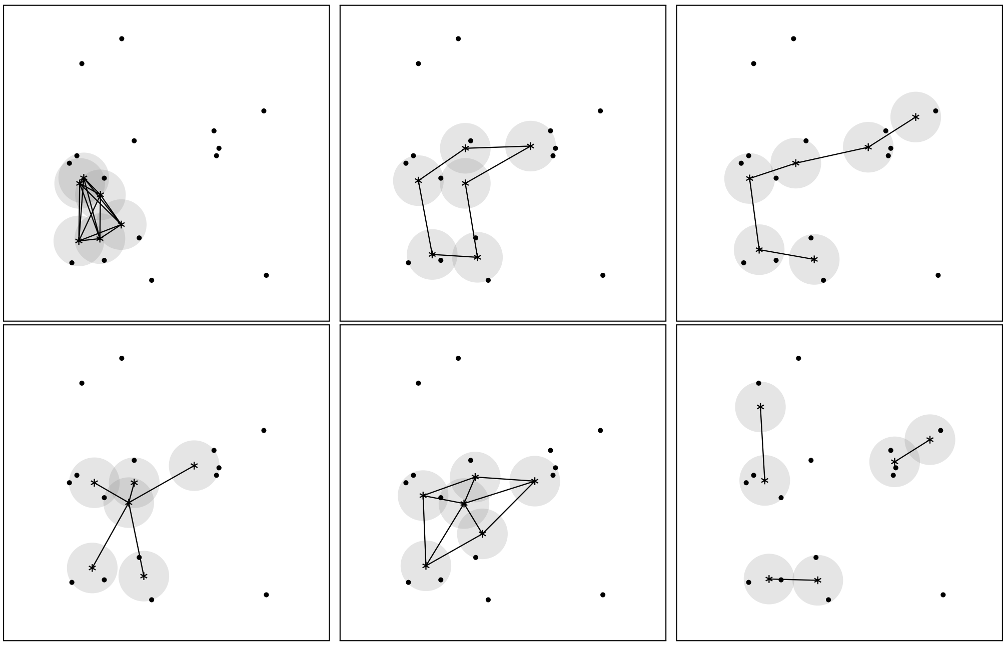


Figure 3.3: Solutions of the MCLPIF for the data of Example 1 and shapes (from left to right): Complete, Cycle, Line, Star, Ring-Star and Matching.

Other graph structures can be modeled and incorporated to the problem. In particular the structure required by Cherkesly et al. (2019) for the interconnected discrete maximal covering location problem, spanning trees. For instance, one may use the subtour elimination constraints (SEC) formulation of the problem:

$$\sum_{\substack{j,k \in P:\\ k<j}} x_{jk} = p - 1, \tag{ST$_1$}$$

$$\sum_{\substack{j,k \in S:\\ j<k}} x_{jk} \leq |S| - 1, \ \forall S \subseteq P, S \neq \emptyset. \tag{ST$_2$}$$

The SEC constraints, (ST$_2$), can be efficiently separated, avoiding the use of exponentially many constraints in the formulation. Other *compact* formulations are the ones proposed by Miller et al. (1960) or the following flow-based formulation:

$$\sum_{\substack{j,k\in P:\\ k<j}} x_{jk} = p-1, \tag{FlowST$_1$}$$

$$\sum_{\substack{\ell\in P\\ \ell\neq 1}} f_{j1\ell} = 1, \quad \forall j\in P, j\neq 1, \tag{FlowST$_2$}$$

$$\sum_{\substack{\ell\in P\\ \ell\neq j}} f_{j\ell j} = 1, \quad \forall j\in P, j\neq 1, \tag{FlowST$_3$}$$

$$\sum_{\ell\in P} f_{jk\ell} - \sum_{\ell\in P} f_{j\ell k} = 0, \quad \forall j,k\in P, j\neq 1, k\neq 1, \ell\neq j, \tag{FlowST$_4$}$$

$$f_{jk\ell} + f_{j\ell k} \leq x_{k\ell}, \quad \forall j,k,\ell\in P, j\neq 1, k<\ell, \tag{FlowST$_5$}$$

where $f_{jk\ell}$ indicates the amount of flow to sent from node 1 to node j using arc (k,ℓ) in the graph of facilities. (FlowST$_1$) indicates that the tree must have $p-1$ edges. Constraints (FlowST$_2$), (FlowST$_3$) and (FlowST$_4$) are the flow conservation constraints. Finally, constraints (FlowST$_5$) avoid using links that are not activated.

Finally, we would like to highlight that the strategy explained in Chapter 1.2.3 to reformulate the continuous MCLP as a discrete MCLP by means of the DFS is no longer valid for the MCLPIF as shown in the following simple counter example.

Example 2. *Let us consider the set of five demand points on the plane* $\mathcal{A} = \{(0,0), (1,0), (3.25,0), (5,0), (6,0)\}$, $R_i = 0.5$ *for* $i = 1,\dots,5$, $r = 2.5$, $p = 3$, $\|\cdot\|$ *the Euclidean norm, and as* $\mathcal{S}(G)$ *a line. The DFS is a Circle Intersection Points set (CIPS) defined by Church (1984), which is* $\text{CIPS} = \mathcal{A}\cup\{(0.5,0),(5.5,0)\}$. *However, the unique optimal solution to the problem to cover the five demand points is to locate the facilities in* $X_1^* = (0.5,0)$, $X_2^* = (3,0)$ *and* $X_3^* = (5.5,0)$ *as shown in Figure 3.4.*

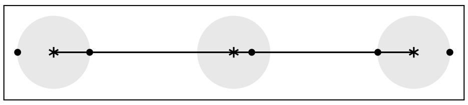


Figure 3.4: Unique optimal solution of the instance of Example 2.

3.3 An Integer Programming Formulation for the MCLPIF

The formulation (MCLPIF$^{\text{NL}}$) for the MCLPIF is a Mixed Integer Non Linear Programming problem, which can be reformulated as a Mixed Integer Second Order Cone Optimization (MISOCO) problem and then, solved using any of the available off-the-shelf software (CPLEX, Gurobi, XPress, ...). However, the capability of MISOCO solvers to solve large-size instances is nowadays limited compared to the efficiency of the routines for solving Mixed Integer Linear Programs. In this section we provide a novel pure Integer Linear Programming formulation for the MCLPIF that only uses the z-variables already used in (c-MCLP$^{\text{NL}}$) and (MCLPIF$^{\text{NL}}$).

First, observe that the *difficult* decisions of the MCLPIF are those associated to the z and the x-variables, since the continuous variables (coordinates of the centers) can be calculated from them in polynomial time.

Theorem 8. *Let $(\bar{z}, \bar{x})$ be a feasible solution for the MCLPIF. Then, optimal positions for the facilities, $X_1, \ldots, X_p \in \mathbb{R}^d$, can be computed in polynomial time.*

Proof. Note that once the z and the x variables are fixed in (MCLPIF$^{\text{NL}}$), the problem turns into:

$$\xi(\bar{x}, \bar{z}) := \max \sum_{\substack{i \in N, j \in P: \\ \bar{z}_{ij}=1}} \omega_i \tag{SP$_1$}$$

$$\text{s.t. } \|a_i - X_j\| \leq R, \ \forall i \in N, j \in P : \bar{z}_{ij} = 1, \tag{SP$_2$}$$

$$\|X_j - X_k\| \leq r, \ \forall j, k \in P, j < k : \bar{x}_{jk} = 1, \tag{SP$_3$}$$

$$X_j \in \mathbb{R}^d, \ \forall j \in P, \tag{SP$_4$}$$

which can be reformulated as a (continuous) SOC problem, and then, solved by interior point methods in polynomial time for any desired accuracy (see Nesterov and Nemirovskii, 1994). □

In what follows we analyze the feasible region of the problem (SP) above, and detail how this information can be exploited and incorporated to the x and z variables in order to project out the X-variables in (MCLPIF$^{\text{NL}}$) in a similar way to the presented in Chapter 1.2.3. For the sake of this analysis, from now on, we assume that each facility covers at least one demand point, which is assured by constraints c-MCLP$_2^{\text{NL}}$ that we recall by:

$$\sum_{i \in N} z_{ij} \geq 1, \ \forall j \in P.$$

Let $(\bar{z}, \bar{x}) \in \{0,1\}^{n \times p} \times \{0,1\}^{p \times p}$ be a feasible solution for the MCLPIF. Denote by $C_j = \{i \in N : \bar{z}_{ij} = 1\}$, the demand points allocated to facility j, and $K_j = \{k \in P : \bar{x}_{\min\{j,k\}\max\{j,k\}} = 1\}$, the set of facilities linked to facility j (according to the solution). Then, we get that:

- By constraints (SP$_2$):

$$X_j \in \bigcap_{i \in C_j} \mathbb{B}_{R_i}(a_i), \ \forall j \in P, \tag{Cov}$$

  that is, X_j must belong to the intersection of all $\|\cdot\|$-balls centered at the points covered by the facility and their radii.

- By constraints (SP$_3$):
$$X_j \in \bigcap_{k\in K_j} \mathbb{B}_r(X_k),\ \forall j \in P, \tag{Link}$$
that is, the j-th facility must be reachable (at distance r) to all the facilities linked to it.

The above conditions fully characterize all the feasible solutions of (SP). However, although (Cov) is clearly determined from the input data (the demand points and the radius R_i), (Link) depends on the coordinates of the X-variables, whose values are unknown. In what follows we derive a necessary and sufficient condition for the existence of feasible values of the X-variables in terms of the z and x-variables.

Let us denote by $\oplus$ the Minkowski sum operator [1] in $\mathbb{R}^d$. Given $\mathbf{C} = \{C_1,\dots,C_p\}$ with $C_1,\dots,C_p \subset N$ and $\mathbf{K} = \{K_1,\dots,K_p\}$ with $K_1,\dots,K_p \subset P$, we denote:

$$z_{ij}^{\mathbf{C}} = \begin{cases} 1 & \text{if } i \in C_j, \\ 0 & \text{otherwise} \end{cases} \text{ and } x_{jk}^{\mathbf{K}} = \begin{cases} 1 & \text{if } k \in K_j, \\ 0 & \text{otherwise} \end{cases},$$

for all $i \in N, j, k \in P$.

The following result allows us to reformulate (MCLPIF$^{\text{NL}}$) using the variables z and x.

Theorem 9. *Let $\mathbf{C} = \{C_1,\dots,C_p\}$ with $C_1,\dots,C_p \subset N$ be nonempty disjoint sets of demand points and $\mathbf{K} = \{K_1,\dots,K_p\}$, with $K_1,\dots,K_p \subset P$, the sets defining the graph structure $\mathcal{S}(G)$. Then, the following conditions are equivalent:*

1. *The set*
$$\mathcal{M}_j(\mathbf{C};\mathbf{K}) := \bigcap_{i\in C_j} \mathbb{B}_{R_i}(a_i) \cap \bigcap_{k\in K_j} \Big(\Big(\bigcap_{i\in C_k} \mathbb{B}_{R_i}(a_i)\Big) \oplus \mathbb{B}_r(0)\Big), \tag{$\mathcal{M}$-SETS}$$
is non empty, for all $j \in P$.

2. *There exists $\mathbf{X} = (X_1,\dots,X_p) \in \mathbb{R}^d$ such that $(z^{\mathbf{C}}, x^{\mathbf{K}}, \mathbf{X})$ is a feasible solution for MCLPIF.*

Proof. Let us assume that $\mathbf{C} = \{C_1,\dots,C_p\}$ and $\mathbf{K} = \{K_1,\dots,K_p\}$ are given such that the $\mathcal{M}$-sets in ($\mathcal{M}$-SETS) are nonempty. Then, we construct the $\bar z$ and $\bar x$-values as:

$$\bar z_{ij} = z_{ij}^{\mathbf{C}} \quad \text{and} \quad \bar x_{jk} = x_{jk}^{\mathbf{K}}.$$

Let us denote by $L_j = \bigcap_{i\in C_j} \mathbb{B}_{R_i}(a_i)$ for all $j \in P$.

Assume that there not exist $\mathbf{X} = (X_1,\dots,X_p)$ such that $(\bar x, \bar z, \mathbf{X})$ is feasible for MCLPIF, i.e., for all $\mathbf{X} = (X_1,\dots,X_p) \in L_1 \times \cdots \times L_p$ there exist $j_X \in P$ and $k_X \in K_{j_X}$

[1] $A \oplus B = \{a + b : a \in A, b \in B\}\ \forall A, B \subset \mathbb{R}^d$

such that $\|X_{j_X} - X_{k_X}\| > r$, or equivalently, $X_{j_X} \notin \mathbb{B}_r(X_{k_X})$. Thus, we have that for all $\mathbf{X} \in L_1 \times \cdots \times L_p$:

$$\Big(L_1 \times \bigcap_{k \in K_1} (L_k \oplus \mathbb{B}_r(0))\Big) \times \cdots \times \Big(L_p \times \bigcap_{k \in K_p} (L_k \oplus \mathbb{B}_r(0))\Big) = \emptyset$$

Then, any of the sets $\mathcal{M}_j = L_j \cap \bigcap_{k \in K_j} (L_k \oplus \mathbb{B}_r(0))$ is empty, contradicting the non-emptiness of the $\mathcal{M}$-sets.

The other implication is straightforward. □

From the above result we can reformulate the MCLPIF as an Integer Linear Programming problem, projecting out the continuous variables in (MCLPIF$^{\text{NL}}$).

Corollary 1. *A solution to the MCLPIF can be obtained by solving the following integer linear programming formulation:*

$$\max \sum_{i \in N} \sum_{j \in P} \omega_i z_{ij} \tag{3.3}$$

$$s.t. \sum_{j \in P} z_{ij} \leq 1, \forall i \in N, \tag{3.4}$$

$$\sum_{i \in N} z_{ij} \geq 1, \forall j \in P, \tag{3.5}$$

$$\sum_{i \in C_j} z_{ij} + \sum_{k \in K_j} \sum_{i \in C_k} z_{ik} + \sum_{k \in K_j} x_{jk} \leq |C_j| + \sum_{k \in L} |C_k| + |K_j| - 1,$$

$$\forall C_1, \ldots, C_p \subset N, K_1, \ldots, K_p \subset P, \text{ with } \mathcal{M}_j(\mathbf{C}; \mathbf{K}) = \emptyset, \tag{3.6}$$

$$x \in \mathcal{S}(G), \tag{3.7}$$

$$z_{ij} \in \{0, 1\}, \forall i \in N, j \in P, \tag{3.8}$$

$$x_{jk} \in \{0, 1\}, \forall j, k \in P, j < k. \tag{3.9}$$

Proof. By Theorem 9, any feasible solution, in the X-variables, given the values of the z and the x-variables, verifies that

$$X_j \in \mathcal{M}_j := \mathcal{M}_j(\mathbf{C}, \mathbf{K}) = \bigcap_{i \in C_j} \mathbb{B}_R(a_i) \cap \bigcap_{k \in K_j} \Big(\Big(\bigcap_{i \in C_k} \mathbb{B}_R(a_i)\Big) \oplus \mathbb{B}_r(0)\Big)$$

and also that the optimal solution of MCLPIF can be obtained by choosing adequately $X_j \in \mathcal{M}_j$ for $j \in P$. Thus, in order to ensure that the z and x variables induce non empty sets $\mathcal{M}_1, \ldots, \mathcal{M}_p$, one must require that the sets $C_1, \ldots, C_p$ and $K_1, \ldots, K_p$ inducing empty $\mathcal{M}$-sets are not allowed.

Constraints (3.6) enforce that in case $\mathcal{M}_j = \emptyset$ the solution if no longer valid.

Once the clusters of points and links are obtained with the formulation above, one is assured that (SP) is feasible, and its solutions are the desired coordinates of the centers. □

The above mathematical programming formulation reduces the search of an optimal solution of the MCLPIF to the binary variables, avoiding continuous variables and non linear constraints, at the price of adding the exponentially many constraints in (3.6). Also, note that constraints (3.6) are added just in case the sets $\mathcal{M}_j(\mathbf{C};\mathbf{K})$ ($\mathcal{M}$-sets, for short) are empty. These sets are constructed as described in ($\mathcal{M}$-SETS), and although they are convex sets, they are not computationally easy to handle. These sets belong to the family of generalized Minkowski sets (see Peters and Herrmann, 2019).

The following result allows us to reduce the *emptiness tests* on the $\mathcal{M}$-sets. We denote by $\mathbb{O}_{i_1i_2} = \Big(\mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}}(a_{i_2})\Big) \oplus \mathbb{B}_r(0)$ for any $i_1, i_2 \in N$. Note that in case $|S| = 1$ the set $\mathbb{O}_{i_1i_1}$ reduces to $\mathbb{B}_{R_{i_1}+r}(a_{i_1})$.

Lemma 3. *Let $S \subseteq N$ then:*

$$\Big(\bigcap_{i\in S} \mathbb{B}_{R_i}(a_i)\Big) \oplus \mathbb{B}_r(0) = \bigcap_{i_1,i_2\in S} \mathbb{O}_{i_1i_2}$$

Proof. First, observe that the intersection of the R_i-disks centered at the points indexed by S is identical to the pairwise intersections in that index set, i.e.

$$\bigcap_{i\in S} \mathbb{B}_R(a_i) = \bigcap_{i_1,i_2\in S} \Big(\mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}}(a_{i_2})\Big). \tag{3.10}$$

Let us now check the identity in the result. On the one hand, let us assume that $z \in \bigcap_{i\in S} \mathbb{B}_R(a_i) \oplus \mathbb{B}_r(0)$. Then, by (3.10), there exist $x \in \bigcap_{i_1,i_2\in S} \Big(\mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}}(a_{i_2})\Big)$ and $y \in \mathbb{B}_r(0)$ such that $z = x + y$. It implies that $x \in \mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}}(a_{i_2})$, $\forall i_1, i_2 \in S$ and $z \in \Big(\mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}}(a_{i_2})\Big) \oplus \mathbb{B}_r(0) = \mathbb{O}_{i_1i_2}$. Thus, $z \in \bigcap_{i_1,i_2\in S} \mathbb{O}_{i_1i_2}$.

On the other hand, let $z \in \bigcap_{i_1,i_2\in S} \mathbb{O}_{i_1i_2}$. Then, $\forall i_1, i_2 \in S$, there exists $x_{i_1i_2}, y_{i_1i_2} : z = x_{i_1i_2} + y_{i_1i_2}$, implying that $x_{i,i'} \in \mathbb{B}_r(z)$, $\forall i, i' \in C_k, i \neq i'$. Then, $\mathbb{B}_r(z) \cap (\mathbb{B}_R(a_i) \cap \mathbb{B}_R(a_{i'})) \neq \emptyset$ for all $i_1, i_2 \in S$. By Helly's Theorem, it assures that $\mathbb{B}_r(z) \cap \bigcap_{i_1,i_2\in S} (\mathbb{B}_R(a_i) \cap \mathbb{B}_R(a_{i'}) \neq \emptyset$. Then, there exists $x \in \mathbb{B}_r(z) \cap \Big(\mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}}(a_{i_2})\Big)$ $\forall i_1, i_2 \in S$. Thus, z can be written as $z = x + (z - x)$, with $x \in \bigcap_{i_1,i_2\in S} \Big(\mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}}(a_{i_2})\Big)$ and $(z - x) \in \mathbb{B}_r(0)$, being then $z \in \bigcap_{i,i'\in S} \Big(\mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}}(a_{i_2})\Big) \oplus \mathbb{B}_r(0)$.

The case $|S| = 1$ follows analogously. □

The following Theorem allows us to replace the exponential number of constraints (3.6) by a polynomial number of them (for fixed dimension d).

Theorem 10. *Let $\mathbf{C} = \{C_1, \ldots, C_p\}$ with $C_1, \ldots, C_p \subset N$ and $\mathbf{K} = \{K_1, \ldots, K_p\}$ with $K_1, \ldots, K_p \subset P$. Then, $\mathcal{M}_j(\mathbf{C};\mathbf{K}) = \emptyset$ if and only:*

$$\bigcap_{i\in S^0} \mathbb{B}_{R_i}(a_i) \cap \bigcap_{i_1,i_2\in S^1} \mathbb{O}_{i_1i_2} = \emptyset,$$

for all $S^0, S^1 \subseteq N$ *with* $|S^0| + |S^1| = d + 1$.

Proof. First, observe that by Lemma 3 we get that:

$$\mathcal{M}_j(\mathbf{C}, \mathbf{K}) = \bigcap_{i \in C_j} \mathbb{B}_R(a_i) \cap \bigcap_{k \in K_j} \Big(\bigcap_{i_1, i_2 \in C_k} \mathbb{O}_{i_1 i_2} \Big)$$

Next, note that $\mathcal{M}_j(\mathbf{C}; \mathbf{K})$ is a convex set since it is the intersections of convex sets (D-balls and Minkowski sums of intersection of balls with a ball). By Helly's Theorem (Helly, 1930), only intersection of $(d+1)$-wise sets is needed. □

In particular, the above theorem allows simplifying constraints (3.6) in the planar $(d = 2)$ case, as stated in the following result, whose proof is straightforward.

Corollary 2. *Let* $\mathbf{C} = \{C_1, \ldots, C_p\}$ *with* $C_1, \ldots, C_p \subset N$, $\mathbf{K} = \{K_1, \ldots, K_p\}$ *with* $K_1, \ldots, K_p \subset P$ *and* $j \in P$. *Then, in the planar case,* $\mathcal{M}_j(\mathbf{C}; \mathbf{K}) = \emptyset$ *if and only any of the following conditions is verified:*

1. $\mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}}(a_{i_2}) \cap \mathbb{B}_{R_{i_3}}(a_{i_3}) = \emptyset$, *for all* $i_1, i_2, i_3 \in C_j$.
2. $\mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}}(a_{i_2}) \cap \mathbb{O}_{i_3 i_4} = \emptyset$, *for all* $i_1, i_2 \in C_j$, $i_3, i_4 \in C_\ell$ *for some* $\ell \in C_k$ *for some* $k \in K_j$.
3. $\mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{O}_{i_2 i_3} \cap \mathbb{O}_{i_4 i_5} = \emptyset$, *for all* $i_1 \in C_j$, $i_2, i_3 \in C_{\ell_1}$, $i_4, i_5 \in C_{\ell_2}$ *for* $\ell_1, \ell_2 \in K_j$.
4. $\mathbb{O}_{i_1 i_2} \cap \mathbb{O}_{i_3 i_4} \cap \mathbb{O}_{i_5 i_6} = \emptyset$, *for all* $i_1, i_2 \in C_{\ell_1}$, $i_3, i_4 \in C_{\ell_2}$, $i_5, i_6 \in C_{\ell_3}$, *for* $\ell_1, \ell_2, \ell_3 \in K_j$.

As a consequence of Corollary 2, in the planar case, constraints (3.6) can be replaced by the following set of constraints. Let $i, i_1, \ldots, i_6 \in N$ and $j, k, k_1, k_2, k_3 \in P$.

- If $\mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}}(a_{i_2}) \cap \mathbb{B}_{R_{i_3}}(a_{i_3}) = \emptyset$:
  - If $\mathbb{B}_{R_{\ell_1}}(a_{\ell_1}) \cap \mathbb{B}_{R_{\ell_2}}(a_{\ell_2}) = \emptyset$ for some $\ell_1, \ell_2 \in \{i_1, i_2, i_3\}$:

    $$z_{\ell_1 j} + z_{\ell_2 j} \leq 1, \ \forall j \in P, \tag{Int$_1$}$$

    that is, points a_{ℓ_1} and a_{ℓ_2} cannot be covered by the same facility.
  - If all pairwise intersections are non empty:

    $$z_{i_1 j} + z_{i_2 j} + z_{i_3 j} \leq 2, \ \forall j \in P, \tag{Int$_2$}$$

    which assures that the three points cannot be simultaneously covered by the same facility.

- If $\mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}}(a_{i_2}) \cap \mathbb{O}_{i_3 i_4} = \emptyset$ (with $\mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}(a_{i_2})} \neq \emptyset$):

- If $\mathbb{B}_{R_\ell}(a_\ell) \cap \mathbb{O}_{i_3 i_4} = \emptyset$ for some $\ell \in \{i_1, i_2\}$:

$$z_{\ell j} + z_{i_3 k} + z_{i_4 k} + x_{jk} \leq 3, \ \forall j, k \in P, j < k, \tag{Int$_3$}$$

ensuring that a_ℓ is not allowed to be allocated to a facility, j, that is linked to the facility, k, which covers a_{i_3} and a_{i_4}.

- If all pairwise intersections are non empty:

$$z_{i_1 j} + z_{i_2 j} + z_{i_3 k} + z_{i_4 k} + x_{jk} \leq 4, \ \forall j, k \in P, j < k, \tag{Int$_4$}$$

avoiding that a_{i_1} and a_{i_2} are allocated to a facility, j, that is linked to the facility, k, which covers a_{i_3} and a_{i_4}.

- If $\mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{O}_{i_2 i_3} \cap \mathbb{O}_{i_4 i_5} = \emptyset$ (with $\mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{O}_{i_2 i_3}, \mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{O}_{i_4 i_5} \neq \emptyset$):

  - If $\mathbb{O}_{i_2 i_3} \cap \mathbb{O}_{i_4 i_5} = \emptyset$:

$$z_{i_2 k_1} + z_{i_3 k_1} + z_{i_4 k_2} + z_{i_5 k_2} + x_{j k_1} + x_{j k_2} \leq 5, \ \forall j, k_1, k_2 \in P, j < k_1, k_2, \tag{Int$_5$}$$

assuring that the facility covering a_{i_2} and a_{i_3} (k_1) and the facility covering a_{i_4} and a_{i_5} (k_2) cannot share a common linked facility (j).

  - If all pairwise intersections are non empty:

$$z_{i_1 j} + z_{i_2 k_1} + z_{i_3 k_1} + z_{i_4 k_2} + z_{i_5 k_2} + x_{j k_1} + x_{j k_2} \leq 6, \ \forall j, k_1, k_2 \in P, j < k_1, k_2, \tag{Int$_6$}$$

avoiding to link the facility that covers a_{i_1}, j, both with the facility covering a_{i_2} and a_{i_3}, k_1, and the facility covering a_{i_4} and a_{i_5}, k_2.

- If $\mathbb{O}_{i_1 i_2} \cap \mathbb{O}_{i_3 i_4} \cap \mathbb{O}_{i_5 i_6} = \emptyset$ (with nonempty pairwise intersections):

$$\begin{aligned} & z_{i_1 k_1} + z_{i_2 k_1} + z_{i_3 k_2} + z_{i_4 k_2} + z_{i_5 k_3} + z_{i_6 k_3} + \\ & \qquad x_{j k_1} + x_{j k_2} + x_{j k_3} \leq 8, \ \forall j, k_1, k_2, k_3 \in P, j < k_1, k_2, k_3, \end{aligned} \tag{Int$_7$}$$

which is the generalization of (Int$_5$) to the case of three facilities (k_1, k_2, k_3) sharing a fourth common linked facility (j).

Summarizing the above comments we obtain the following result for the planar MCLPIF.

Theorem 11. *The planar MCLPIF can be equivalently formulated as the following Integer*

Linear Programming problem:

$$\max \sum_{i\in N}\sum_{j\in P} \omega_i z_{ij} \tag{MCLPIF$_1^{\mathrm{IP}}$}$$

$$s.t.(\text{Int}_1) - (\text{Int}_7), \tag{MCLPIF$_2^{\mathrm{IP}}$}$$

$$\sum_{j\in P} z_{ij} \leq 1, \ \forall i \in N, \tag{MCLPIF$_3^{\mathrm{IP}}$}$$

$$\sum_{i\in N} z_{ij} \geq 1, \ \forall j \in P, \tag{MCLPIF$_4^{\mathrm{IP}}$}$$

$$x \in \mathcal{S}(G),$$

$$z_{ij} \in \{0,1\}, \ \forall i \in N, j \in P,$$

$$x_{jk} \in \{0,1\}, \ \forall j,k \in P, j < k.$$

The above integer linear programming formulation is a compact model for MCLPIF with a polynomial number of constraints ($O(p^3n^6)$). Apart from the large number of constrains it has, the main drawback of this formulation is that each of the constraints (Int_1)-(Int_7) requires checking either intersection of $\|\cdot\|$-balls, $\mathbb{O}$-sets or both.

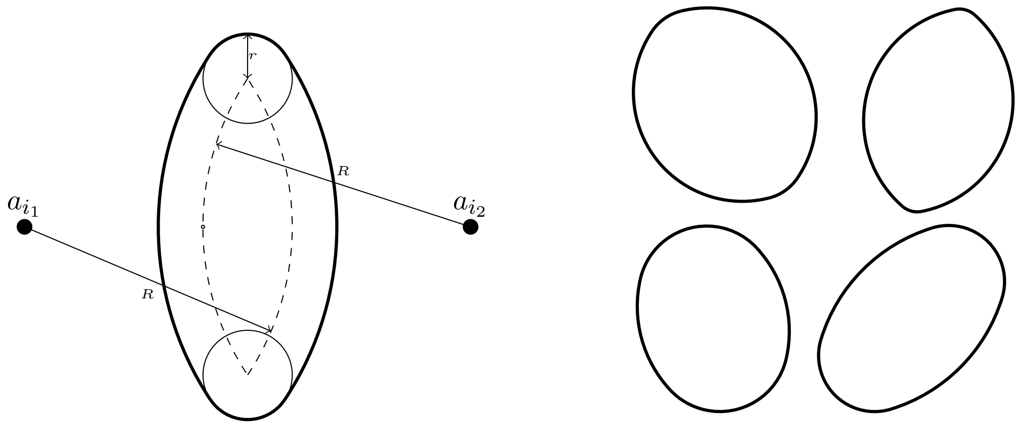
In the following section we give the details of the explicit construction of $\mathbb{O}$-sets on the plane that allows geometrically checking the emptiness of the intersections in practice.

3.3.1 Planar $\mathbb{O}$-sets

In what follows we study the geometry of the $\mathbb{O}$-sets defined above in order to exploit it in the Integer Programming Formulation for the MCLPIF. Given $a_{i_1}, a_{i_2} \in \mathbb{R}^d$, this set is defined as:

$$\mathbb{O}_{i_1 i_2} = \Big(\mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}}(a_{i_2})\Big) \oplus \mathbb{B}_r(0).$$

This set is the Minkowski sum of the intersection of two D-balls centered at demand points a_{i_1} and a_{i_2} and a D-ball centered at the origin. Since both sets are convex and bounded, $\mathbb{O}_{i_1 i_2}$ is also bounded and convex. In Figure 3.5a we illustrate the shape of this convex set in the planar case. There, we show the two centers of the Euclidean balls (disks), a_{i_1} and a_{i_2}, and the boundary of the intersection $\mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}}(a_{i_2})$ (drawn with a dashed line). The disks $\mathbb{B}_r(0)$ are then moved all around the points in the intersection of the disks. The border of $\mathbb{O}_{i_1 i_2}$ is drawn with a thick line in the picture. In Figure 3.5b we show different shapes for the planar $\mathbb{O}$-sets.



(a) Elements of a planar $\mathbb{O}$-set.

(b) Different shapes for planar $\mathbb{O}_{i_1 i_2}$.

Figure 3.5: Planar $\mathbb{O}$-sets.

Note that $\mathbb{O}_{i_1 i_2}$ is a non empty set provided that $\mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}}(a_{i_2})$ is non empty and $r > 0$. Also, observe that $\mathbb{O}_{i_1 i_2} \subset \mathbb{B}_{R_{i_1}+r}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}+r}(a_{i_2})$. This is clear since any element in $x \in \mathbb{O}_{i_1 i_2}$ can be written as $x = z + y$ where $z \in \mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}}(a_{i_2})$ and $y \in \mathbb{B}_r(0)$. Then, $\|x - a_{i_1}\| \leq \|(z + y) - a_{i_1}\| \leq \|z - a_{i_1}\| + \|y - 0\| = R_{i_1} + r$ (analogously for a_{i_2}). In case $a_{i_1} = a_{i_2}$, $\mathbb{O}_{i_1 i_2}$ coincides with $\mathbb{B}_{R_{i_1}+r}(a_{i_1})$. Otherwise ($a_{i_1} \neq a_{i_2}$), $\mathbb{O}_{i_1 i_2}$ is nothing but a *smoothing* of $\mathbb{B}_{R_{i_1}+r}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}+r}(a_{i_2})$ by two balls of radius r in the two peaks of such a shape (see Figure 3.6 where the boundary of $\mathbb{O}_{i_1 i 2}$ is drawn with a thick line and the boundary of the intersection $\mathbb{B}_{R_{i_1}+r}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}+r}(a_{i_2})$ is drawn with a thiner gray line). Actually, as one can also observe from Figure 3.6 that $\mathbb{O}_{i_1 i_2}$ can be decomposed in three parts. On the one hand, the middle part in the picture coincide with $\mathbb{B}_{R_{i_1}+r}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}+r}(a_{i_2})$ inside the strip delimited by the intersections points of the balls centered at $\{q_1, q_2\} = \partial\mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \partial\mathbb{B}_{R_{i_2}}(a_{i_2})$ and radius r (here, ∂A denotes the boundary of the bounded set A). On the other hand, the two other parts of $\mathbb{O}_{i_1 i_2}$ are just the balls with centers in $\{q_1, q_2\}$ and radius r.

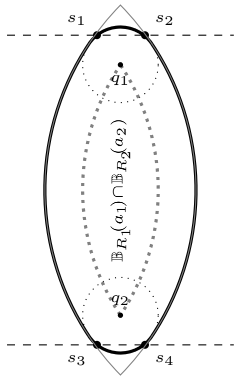


Figure 3.6: Comparison of $\mathbb{O}_{i_1 i_2}$ and $\mathbb{B}_{R+r}(a_{i_1}) \cap \mathbb{B}_{R+r}(a_{i_2})$.

Thus, $\mathbb{O}_{i_1 i_2}$ can be written as the following (non disjoint) union:

$$\mathbb{O}_{i_1 i_2} = \mathbb{B}_r(q_1) \cup \mathbb{B}_r(q_2) \cup \Big(\mathbb{B}_{R_{i_1}+r}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}+r}(a_{i_1}) \cap \mathcal{S}_{i_1 i_2} \Big),$$

where $\{q_1, q_2\} = \partial \mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \partial \mathbb{B}_{R_{i_2}}(a_{i_2})$ and $\mathcal{S}_{i_1 i_2} = \{z \in \mathbb{R}^d : \alpha_0 + \alpha^t z \leq 0 \leq \beta_0 + \beta^t z\}$ where $\alpha_0, \beta_0 \in \mathbb{R}, \alpha, \beta \in \mathbb{R}^d$ are the coefficients of the hyperplane passing through the points intersecting each of the balls centered at q_1 and q_2 and radius r and $\partial \mathbb{B}_{R_{i_1}+r}(a_{i_1}) \cap \partial \mathbb{B}_{R_{i_2}+r}(a_{i_2})$ and such that $q_1, q_2 \in \mathcal{S}_{i_1 i_2}$ (strip delimited by the dotted lines in Figure 3.6).

3.4 Branch-and-cut approaches for the MCLPIF

The main bottleneck of formulation (MCLPIF$^{\text{IP}}$) is the large number of constraints in the form (Int_1)-(Int_7) it has. Furthermore, checking the emptiness of the intersections of balls and $\mathbb{O}$-sets requires much memory and CPU time. We propose two exact methodologies for solving the planar MCLPIF in which the family of constraints (Int_1)-(Int_7) is initially relaxed. When solving these relaxed problems, one may obtain solutions which are not feasible for the MCLPIF. We provide an efficient separation strategy that allows either certifying the feasibility of a solution or generating a violated inequality for it. This procedure is embedded within an enumeration tree and it is applied not only at the root node but also at all generated nodes.

In what follows we describe the two incomplete formulations that we propose for the problem as well as the separation oracle that we apply to generate the violated constraints.

3.4.1 Incomplete formulation 1

In our first incomplete formulation, we initially incorporate the covering constraints (Int_1) and (Int_2). The solutions obtained when solving this relaxed model verify that the demand points are allocated to the p servers (taking into account the coverage radii), but the facilities are not assured to be adequately linked within a given distance r. This approach is particularly useful in instances where the radius r is large, compared to the coverage radii, R_i's, since the number of constraints induced by the $\mathbb{O}$-sets is small, and it may be convenient to incorporate them as long as they are violated. The incomplete formulation reads as follows:

$$\max \sum_{i \in N} \omega_i \sum_{j \in P} z_{ij} \tag{INC$^1_1$}$$

$$\text{s.t. } (\text{Int}_1), (\text{Int}_2), \tag{INC$^1_2$}$$

$$\sum_{j \in P} z_{ij} \leq 1, \ \forall i \in N, \tag{INC$^1_3$}$$

$$\sum_{i \in N} z_{ij} \geq 1, \ \forall j \in P, \tag{INC$^1_4$}$$

$$x \in \mathcal{S}(G),$$

$$z_{ij} \in \{0,1\}, \ \forall i \in N, j \in P,$$

$$x_{jk} \in \{0,1\}, \ \forall j,k \in P, j < k.$$

3.4.2 Incomplete formulation 2

In our second incomplete formulation, we consider a *relaxed* version of the $\mathcal{M}$-sets that appear in Theorem 9. In particular, we use the following straightforward result that states that these sets are subsets of the intersections of certain balls.

Proposition 2. *Let* $\mathbf{C} = \{C_1, \ldots, C_p\}$ *with* $C_j \subset N$*, for all* $j \in P$ *and* $\mathbf{K} = \{K_1, \ldots, K_p\}$ *with* $K_j \subset P$ *for all* $j \in P$*. Then:*

$$\mathcal{M}_j(\mathbf{C}, \mathbf{K}) \subset \mathcal{L}_j(\mathbf{C}, \mathbf{K}) := \bigcap_{i \in C_j} \mathbb{B}_{R_i}(a_i) \cap \bigcap_{k \in K_j} \Big(\bigcap_{i \in C_k} \mathbb{B}_{R_i + r}(a_i) \Big), \quad \textit{for all } j \in P.$$

Furthermore, the above inclusion is strict, except in case C_j *is a singleton.*

Using the above result, one may obtain a relaxed version of ($\text{MCLPIF}^{\text{IP}}$) by replacing constraints (Int$_3$)-(Int$_7$) by the following ones:

$$z_{i_1j} + z_{i_2k} + x_{jk} \leq 2, \text{ if } \mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{B}_{R'_{i_2}}(a_{i_2}) = \emptyset, \tag{Int$'_3$}$$

$$z_{i_1k_1} + z_{i_2k_2} + x_{jk_1} + x_{jk_2} \leq 3, \text{ if } \mathbb{B}_{R'_{i_1}}(a_{i_1}) \cap \mathbb{B}_{R'_{i_2}}(a_{i_2}) = \emptyset, \tag{Int$'_4$}$$

$$z_{i_1j} + z_{i_2j} + z_{i_3k} + x_{jk} \leq 3, \text{ if } \mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{B}_{R_{i_2}}(a_{i_2}) \cap \mathbb{B}_{R'_{i_3}}(a_{i_3}) = \emptyset, \tag{Int$'_5$}$$

$$z_{i_1j} + z_{i_2k_1} + z_{i_3k_2} + x_{jk_1} + x_{jk_2} \leq 4, \text{ if } \mathbb{B}_{R_{i_1}}(a_{i_1}) \cap \mathbb{B}_{R'_{i_2}}(a_{i_2}) \cap \mathbb{B}_{R'_{i_3}}(a_{i_3}) = \emptyset, \tag{Int$'_6$}$$

$$z_{i_1k_1} + z_{i_2k_2} + z_{i_3k_3} + x_{jk_1} + x_{jk_2} + x_{jk_3} \leq 5, \text{ if } \mathbb{B}_{R'_{i_1}}(a_{i_1}) \cap \mathbb{B}_{R'_{i_2}}(a_{i_2}) \cap \mathbb{B}_{R'_{i_3}}(a_{i_3}) = \emptyset. \tag{Int$'_7$}$$

for $i, i_1, \ldots, i_6 \in N, \forall j, k, k_, k_2, k_3 \in P, j < k, k_, k_2, k_3$ and where $R'_i = R_i + r$.

The second incomplete formulation reads as:

$$\max \sum_{i \in N} \sum_{j \in P} \omega_i z_{ij} \tag{INC$^2_1$}$$

$$\text{s.t. } (\text{Int}_1), (\text{Int}_2), (\text{Int}'_3) - (\text{Int}'_7), \tag{INC$^2_2$}$$

$$\sum_{j \in P} z_{ij} \leq 1, \ \forall i \in N, \tag{INC$^2_3$}$$

$$\sum_{i \in N} z_{ij} \geq 1, \ \forall j \in P, \tag{INC$^2_4$}$$

$$x \in \mathcal{S}(G),$$

$$z_{ij} \in \{0, 1\}, \ \forall i \in N, j \in P,$$

$$x_{jk} \in \{0, 1\}, \ \forall j, k \in P, j < k.$$

The main advantage of this incomplete formulation compared to (INC^1), is that its feasible region is *closer* to (MCLPIF$^{\text{IP}}$), and then, theoretically, a smaller number of violated should be added.

3.4.3 Separation of violated inequalities

Given a feasible solution, $(\bar{z}, \bar{x})$, obtained at a MIP node of any of the incomplete formulations ((INC^1) or (INC^2)), the separation oracle tries to obtain the positions of the p facilities with covering points provided by $\bar{z}$ and links derived from $\bar{x}$. Both incomplete formulations assure that one can construct the facilities covering the demand points at their coverage radii. However, we are not assured to be able to link the facilities within the required distance r. We check the feasibility of the obtained solution by solving the following problem:

$$\begin{aligned}
\rho(\bar{z}, \bar{x}) := \min & \sum_{j \in P} \sum_{k \in K_j} q_{jk} \\
\text{s.t. } & \|X_j - X_k\| \leq r + q_{jk}, \ \forall j \in P, k \in K_j, \\
& \|X_j - a_i\| \leq R_i, \ \forall j \in P, i \in C_j, \\
& q_{jk} \geq 0, \ \forall j \in P, k \in K_j, \\
& X_1, \ldots, X_p \in \mathbb{R}^2,
\end{aligned}$$

where $K_j = \{k \in P : \bar{x}_{\min\{j,k\}\max\{j,k\}} = 1\}$ and $C_j = \{i \in N : \bar{z}_{ij} = 1\}$ for all $j \in P$.

The set of nonnegative continuous slack variables q_{jk} allows us to account for the excess of distance (with respect to r) when linking facilities j and k. Since the overall sum of these variables is minimized, in case $\rho(\bar{z}, \bar{x}) = 0$, the solution is feasible for the MCLPIF (and the optimal values for $X_1, \ldots, X_p$ are valid coordinates for the servers). Otherwise, the solution violates one of the constraints (3.6) and we add the following constraint to the incomplete formulation:

$$\sum_{j \in P} \sum_{i \in C_j} z_{ij} + \sum_{j \in P} \sum_{\substack{k \in K_j: \\ j < k}} x_{jk} \leq \sum_{j \in P} (|C_j| + |K_j|) - 1. \tag{CUT($\mathbf{C}, \mathbf{K}$)}$$

This separation strategy is embedded within the branch-and-bound tree by checking feasibility, and eventually, adding lazy cuts, at each MIP node of the tree.

3.5 Matheuristic approach for larger instances

The MCLPIF is computationally costly, even for medium size instances, as we will see in Section 3.6. In this section we provide a family of heuristic approaches capable to solve larger instances in reasonable CPU times. One of the most popular families of heuristic algorithms in continuous location is the family of aggregation techniques (see e.g., Current and Schilling, 1990; Daskin et al., 1989; Emir-Farinas and Francis, 2005; Irawan, 2016b). Aggregation is a useful tool for manipulating data and determining the appropriate policies to employ for large-scale optimization models.

Recall that an instance of the MCLPIF consists of a tuple $(\mathcal{A}, w, R, r, \mathcal{S})$ where $\mathcal{A} = \{a_1, \ldots, a_n\}$ is the set of demand points, w is the vector of demand weights, R is the vector of coverage radii, r is the maximum allowed distance between the centers and $\mathcal{S}$ is the graph structure required for the facilities. Aggregating points of $(\mathcal{A}, w, R, r, \mathcal{S})$ means replacing it by another instance $(\mathcal{A}', w', R', r, \mathcal{S})$ in which obtaining the solution to the MCLPIF is easier than for the original one by reducing the number of demand points of $\mathcal{A}$, i.e., $|\mathcal{A}'| \ll |\mathcal{A}|$. Once the solution of this aggregated instance is obtained (the centers) the objective function of the original problem is evaluated by computing the weighted sum of covered points with the obtained centers. However, this strategy may reduce the accuracy of the model, incurring in different source errors (Current and Schilling, 1990). Below we describe the different phases of our strategy to aggregate the demand points and palliate the incurred aggregation errors. Figure 3.7 shows a flowchart of our matheuristic, which consists of four phases: Aggregation, Construction of Initial Solutions, Location-Allocation, and Improvement.

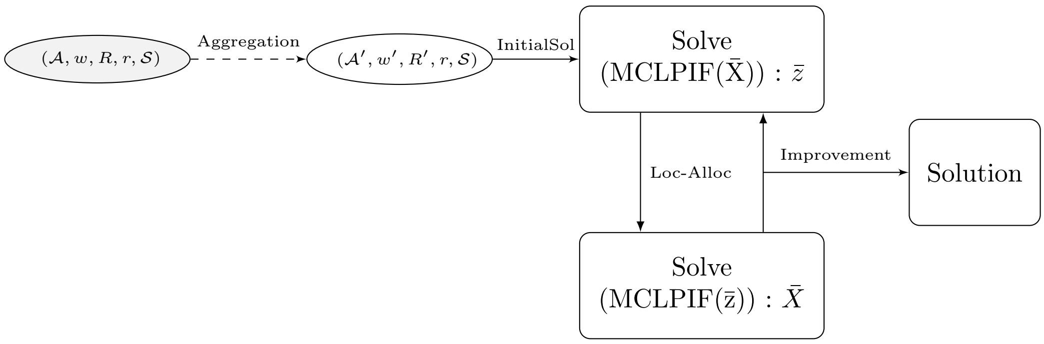


Figure 3.7: Flowchart of the matheuristic

3.5.1 Aggregation

The first phase of our procedure consists of aggregating the demand points. We apply two different strategies: the *k-Mean Clustering*, replacing the original points by the obtained centroids and the *Pick the Farthest* (PTF) strategy proposed by Daskin et al. (1989). We denote by $\mathcal{A}'$ the new set of demand points and by $N' = \{1, \ldots, |\mathcal{A}'|\}$ the index set for these points.

Once the points are aggregated, the weights of the aggregated demand points in $\mathcal{A}'$ are set to the sum of the weights of the original demand points allocated to it with any of the two aggregation strategies. The coverage radius of an aggregated demand point is set to the minimum of the coverage radii of the original points assigned to it.

3.5.2 Construction of initial solutions

In the next phase of our matheuristic, we construct initial feasible centers (X-variables) for the aggregated instance. First, the approach tries choosing the p centers out from the set $\mathcal{A}'$, as suggested Drezner (1984) and Callaghan et al. (2017) for the p-center problem, by solving the following Integer Linear Programming Problem:

$$\begin{aligned}
\min \quad & \sum_{\substack{j,k \in P: j<k \\ x_{jk}=1}} \tilde{D}_{jk} \\
\text{s.t.} \quad & \sum_{j \in P} h_{ij} \leq 1, \ \forall i \in N', && \text{(INI}_1\text{)} \\
& \sum_{i \in N'} h_{ij} = 1, \ \forall j \in P, && \text{(INI}_2\text{)} \\
& \tilde{D}_{jk} \geq \|a'_{i_1} - a'_{i_2}\|(h_{i_1k} + h_{i_2j} - 1), \ \forall k, j \in P \text{ with } x_{jk} = 1, j < k, && \text{(INI}_3\text{)} \\
& \tilde{D}_{jk} \leq r, \ \forall k, j \in P \text{ with } x_{jk} = 1, j < k, && \text{(INI}_4\text{)} \\
& h_{ij} \in \{0, 1\}, \ \forall i \in N', j \in P, \\
& \tilde{D}_{jk} \geq 0, \ \forall j \in P,
\end{aligned}$$

where we use the variables $h_{ij} = \begin{cases} 1 & \text{if } a_i \text{ is selected as the } j\text{th facility,} \\ 0 & \text{otherwise} \end{cases} \quad \forall i \in N', \forall j \in P$ and $\tilde{D}_{jk}$ is the distance between the jth and the kth facilities in case $x_{jk} = 1$. In the problem above a set of p demand points is selected from $\mathcal{A}'$ (constraints INI_1 and INI_2) verifying the distance limit constraints of the graph structure (constraints INI_3 and INI_4). There, if i_1 is selected as the jth facility and i_2 as the kth facility, the distance $\tilde{D}_{jk}$ is computed as the distance between demand points a'_{i_1} and a'_{i_2}, as desired. Although in this phase we only require to construct a feasible solution for the problem, we select the one with smallest overall sum of the distances between the linked facilities.

In case the above problem is feasible, with optimal values for the h-variables, $\bar{h}$, we set as initial centers for our problem the set $\bar{\mathrm{X}} = \{\bar{X}_1, \ldots, \bar{X}_p\} = \{a'_i \in \mathcal{A}' : \sum_{j \in P} \bar{h}_{ij} = 1\}$. In case the problem above is infeasible, we compute the initial solution by solving the MCLP of the aggregated points, with a single center ($p = 1$), c, and radius $\frac{r}{2}$. Then, we select, randomly, a set of p points in $\mathbb{B}_{\frac{r}{2}}(c)$ which are assured to verify that the distance between all the facilities is at most r and each of the selected facilities covers at least one demand point.

3.5.3 Location-allocation

The third phase consists of applying an alternate convex search strategy (Gorski et al., 2007), also known as location-allocation approach in Facility Location (see e.g., Gharaei et al., 2020). The rationale of this approach is to alternate in solving subproblems in the solution spaces of the two main sets of variables (X and z). We proceed iteratively, and each iteration consists of solving a pair of subproblems, one in each space of variables. When solving the subproblem in one solution space we fix the values of the variables of the other space.

Formally, let (MCLPIF($\bar{\mathrm{X}}$)) and (MCLPIF($\bar{\mathrm{z}}$)), denote the subproblems of a MCLPIF formulation, when $\bar{X}$ and $\bar{z}$ (for z-values equal to one) are fixed, respectively. We start with the initial centers obtained in the previous phase. In the k-th iteration, the allocation of demand points to centers is performed (by inspection) and the z-values are computed. Then, the z-variables with value one are fixed (while the other are still free to take any binary value) and the MCLPIF is solved, obtaining new centers. The procedure terminates when two consecutive iterations produce the same solution.

3.5.4 Improvement

When the Location-Allocation phase is terminated, a set of feasible centers, $X_1, \ldots, X_p$ is obtained. The objective function of the original MCLPIF can be evaluated to know the objective value for this solution.

In order to mitigate aggregation errors that affect the evaluation of the objective function (see Daskin et al., 1989), we apply a last step, in which slight translations of the centers $X_1, \ldots, X_p$ are allowed to cover a larger amount of (weighted) demand points. This phase is performed by solving the following MISOCO problem:

$$\begin{aligned}
\max & \sum_{i \in N} w_i y_i \\
\text{s.t. } & \|X_{j(i)} - a_i\| \leq R_i + M(1 - y_i), \forall i \in N, && (\mathrm{AUX}_1) \\
& \|X_j - X_k\| \leq r, \forall j, k \in P, j < k : \bar{x}_{jk} = 1, && (\mathrm{AUX}_2) \\
& y_i \in \{0, 1\}, \forall i \in N, \\
& X_j \in \mathbb{R}^d, \forall j \in P,
\end{aligned}$$

where we use the binary variables $y_i = \begin{cases} 1 & \text{if } a_i \text{ is covered by the } j(i) \text{ center,} \\ 0 & \text{otherwise} \end{cases}$ and $l(i) = \arg\min_{j \in P} \|X'_j - a_i\|$, for all $i \in N$.

To accelerate the resolution of the above problem, we fix the y-variables to zero in case $\|a_i - X'_j\| \geq R_i + 1.5\% R_i$, and to one if $\|a_i - X'_j\| \leq R_i/2$, for $i \in N, j \in P$.

Note that the error that most affect to the objective function of the MCLPIF is the one in which the centroid is covered by the center but some of the points assigned to it are not. The above auxiliary problem mitigates this effect by allowing the facilities to accommodate in the space (but close to the given centers) to cover the most demand as possible.

Finally, the objective function of the MCLPIF is evaluated with the obtained center in order to compute the effective covered demand.

3.6 Computational experiments

In this section, we report the results of our computational experience in order to evaluate the performance of the different proposed approaches for solving the planar MCLPIF. We consider different datasets from the location analysis literature with sizes ranging from 10 to 2863 demand points with coordinates normalized in the unit square (available in `github.com/vblancoOR/mclpif`). The number of centers to be located, p, ranges in $\{2, 6, 10\}$. We consider the same radius for all the demand points and ranging in $\{0.1, 0.2, 0.3\}$, and the limit distance between linked facilities $r \in \{0.3, 0.5\}$. The graph structures that we analyze are `Comp` (complete), `Cycle`, `Line`, `Matching`, `Star` and `Ring-Star` (see Section 3.2.1). Note that if $p = 2$ the MCLPIF coincides for all these graphs.

The models were coded in Python 3.7 in a MacBook Pro with a Core i5 CPU clocked at 2 GHz and 8GB of RAM memory. We use Gurobi 9.0 as optimization solver. A time limit of 1 hour was fixed for all the instances.

The complete results obtained in our computational experiments are available in the `github` repository `github.com/vblancoOR/mclpif`.

3.6.1 Computational performance of (MCLPIF$^{\text{IP}}$)

In order to evaluate formulation (MCLPIF$^{\text{IP}}$) we randomly generate 5 samples of sizes in $\{10, 20\}$ from the classical planar 50-points dataset provided by Eilon et al. (1974). In Table 3.1, the results are organized by graph structure (`Graph`), number of demand points (`n`) and number of centers to be located (`p`). We report the average consumed CPU time (in seconds) for solving the ILP problems (`IP_Time`), the average CPU time (in seconds) required to generate the constraints of (MCLPIF$^{\text{IP}}$) (`ContrGen_Time`) and the average total CPU time (in seconds) required for both tasks plus the CPU time (in seconds) of solving the continuous nonlinear problem (SP) (`Tot_Time`). We also report the information concerning the number of constraints of problem (MCLPIF$^{\text{IP}}$). In particular, we provide the average number of constraints involving only balls (`#Ball_Ctrs`), the constraints involving $\mathbb{O}$-sets (`#O_Ctrs`) and the overall number of constraints in the problem (`#All_Ctrs`). In column `%OoM` we report the percentage of instances that flagged "Out Of Memory" when trying to solve the problem.

Observe that the CPU times for solving the ILP formulation are small, but the CPU times needed to generate the constraints are huge compared to the size of the instances. Concretely, 99.04% of the total time is consumed generating the constraints of the problem. As expected, this approach is computationally inefficient because of the large number of constraints in the problem and the high computational cost needed to compute intersections of balls and $\mathbb{O}$-sets. Even for the small instances, the number of linear constraints needed to represent the nonlinear nature of the problem is excessive.

Graph	n	p	IP_Time	ContrGen_Time	Tot_Time	#Ball_Ctrs	#O_Ctrs	#All_Ctrs	%OoM
Complete	10	2	0.00	21.07	21.07	121	903	1036	0%
		6	0.30	24.66	24.96	1267	191936	193218	0%
	20	2	0.05	1314.88	1314.93	763	25142	25927	0%
		6	16.52	1387.05	1403.57	9155	3781055	3790236	3%
		10	48.78	1584.20	1632.98	13310	12221020	12234360	50%
Cycle	10	2	0.00	21.07	21.07	121	903	1036	0%
		6	0.03	22.76	22.79	566	20936	21518	0%
	20	2	0.05	1314.88	1314.93	763	25142	25927	0%
		6	1.80	1345.31	1347.12	3916	408220	412162	0%
		10	3.11	1355.91	1359.01	6527	750784	757341	0%
Line	10	2	0.00	21.07	21.07	121	903	1036	0%
		6	0.02	22.39	22.41	499	13843	14357	0%
	20	2	0.05	1314.88	1314.93	763	25142	25927	0%
		6	0.64	1341.90	1342.55	3374	196500	199899	0%
		10	1.42	1351.65	1353.07	5984	427510	433525	0%
Matching	10	2	0.00	21.07	21.07	121	903	1036	0%
		6	0.01	21.32	21.33	364	4498	4878	0%
	20	2	0.05	1314.88	1314.93	763	25142	25927	0%
		6	0.20	1323.10	1323.30	2288	86756	89070	0%
		10	0.37	1350.02	1350.38	3813	163321	167164	0%
Star	10	2	0.00	21.07	21.07	121	903	1036	0%
		6	0.05	21.87	21.92	515	33159	33689	0%
	20	2	0.05	1314.88	1314.93	763	25142	25927	0%
		6	4.37	1343.42	1347.79	3432	549085	552543	0%
		10	19.20	1380.24	1399.44	6259	2696969	2703257	7%
Ring-Star	10	2	0.00	21.07	21.07	121	903	1036	0%
		6	0.11	22.63	22.73	864	75528	76408	0%
	20	2	0.05	1320.04	1320.31	763	25142	25927	0%
		6	7.81	1353.78	1361.59	6196	1418769	1424991	0%
		10	27.95	1421.35	1449.30	11233	4313033	4324296	7%

Table 3.1: Results of computational experiment for solving the ($\text{MCLPIF}^{\text{IP}}$).

3.6.2 Computational performance of the incomplete formulations

We analyze now the two proposed branch-and-cut approaches for the MCLPIF. As in the previous experiments, we randomly generate 5 samples with sizes in $\{10, 20, 30, 40\}$, and the whole dataset of size 50 from the 50-points instance from the work of Eilon et al. (1974). 2088 instances were solved in this experiment.

We compare the performance of the three following approaches: `NL`: Compact Mixed Integer Non Linear formulation ($\text{MCLPIF}^{\text{NL}}$); $\texttt{B\&C}_1$: branch-and-cut approach based on the incomplete formulation (INC^1); and $\texttt{B\&C}_2$: branch-and-cut approach based on the incomplete formulation (INC^2)

The results are reported in Table 3.2. We report, for each of the three approaches: the average CPU times (in seconds) only for the optimality solved instances and the percentage of unsolved instances, the MIP Gaps obtained at the end of the time limit and also the average number of cuts required to solve the problem up to optimality in the branch-and-cut approaches. Next to the average CPU times, in parenthesis, we report the standard deviations of those times. In the last column we also report the percentage deviation of the best solution found, with any of the three approaches, with respect to the solution of the classical MCLP.

As can be observed, the branch-and-cut approaches clearly outperform (in CPU time) the Mixed Integer Programming Formulation ($\text{MCLPIF}^{\text{NL}}$). In 87% of the instances the CPU times required with the branch-and-cut algorithms are smaller than the CPU times required by the MISOCO formulation. The overall average of the CPU times (for those instances in which optimality was certified within the time limit) for ($\text{MCLPIF}^{\text{NL}}$) was 105 seconds, while for $\texttt{B\&C}_1$ was 23 seconds, i.e., in average, the incomplete formulation consumed 25% less of the time than that required for the non linear formulation. Concerning the two branch-and-cut schemes, one can observe that the results are similar, both in CPU times and in percentage of unsolved instances.

The nonlinear formulation was not able to solve 283 of the instances while the branch-and-cut approaches only failed in 53 of them. The results are more impressive for the complete graph structure with 10 facilities and $n = 50$ where none of the 6 instances were solved within the time limit with the non linear formulation while branch-and-cut algorithms solved each of them in at most 102.35 seconds. In contrast to the results reported in Section 3.6.1 for ($\text{MCLPIF}^{\text{IP}}$) where 99% of the CPU time was consumed checking intersection and loading constraints to the models, in the branch-and-cut algorithms this percentage is approximately 50% in average in all the instances. The number of cuts added through the execution of the two branch-and-cut approaches is also similar. The MIP Gaps of the compact approach are also greater than those obtained with the branch-and-cut approaches. In particular, in 86.69% of the instances, the compact formulation obtained higher gaps than the incomplete formulations.

We further analyze the results for the cases in which $p \geq 6$, where the the CPU times for solving the problem for $p = 6$ are, in average, larger than those for solving the problems for $p = 10$. On the one hand, note that our approaches are based on relaxing the MCLPIF to the MCLP. Then, as closer is the optimal solution of our problem to the one of the MCLP, less number of cuts are needed, and smaller CPU times are required for solving the problem. For larger values of p, in general, the covered demand will be larger and the density of the coverage areas within the convex hull of the demand points implies that the solution of the MCLP will be closer to the one of the MCLPIF. Otherwise, the representation of the *interconnection* of our problem is initially weak in the relaxed problem, and it has to be sequentially incorporated to the problem, increasing its resolution computational cost. Analyzing the last column in Table 3.2, one can observe that, in many cases (Cycle with $n = 30, 40$, Line with $n = 40, 50$, Matching with $n = 30, 40, 50$, or Ring-Star with $n = 30, 40, 50$) the solution of the MCLPIF is closer, in average, to the one of the MCLP for $p = 10$ than for $p = 6$, implying an increase on the CPU times for solving the problems for six facilities. On the other hand, we also observe that, since the instances were randomly generated, some of them seems to be more difficult to solve than others. It can be checked on the values of the standard deviations reported in parenthesis next to the CPU times.

Some of those values are particularly high, indicating that there is a significant difference between the difficulties of solving the same problem of instances with the same size.

From the last column, one can also observe that, as expected, the MCLPIF with a Complete graph is the most restrictive situation with an average deviation of 21% with respect to the solution of the classical MCLP, i.e., the MCLPIF covers, in average, only 79% of the points that were covered with the MCLP. The results for the Matching graph indicate that this graph is the most flexible one with an overall average deviation of 2.5% and with 75% of the instances coinciding, in optimal objective value with the MCLP.

We would like also to highlight that the MCLPIF is much more difficult to solve than the MCLP. The maximum CPU time that Gurobi consumed for solving the MCLP was 1.06 seconds while some of the instances of the MCLPIF were not able to be solved within the time limit of one hour.

			Average CPU Times (std)			UnSolved			MIP Gaps			Number of Cuts		MCLP
Graph	n	p	(MCLPIF^{NL})	B&C$_1$	B&C$_2$	(MCLPIF^{NL})	B&C$_1$	B&C$_2$	(MCLPIF^{NL})	B&C$_1$	B&C$_2$	B&C$_1$	B&C$_2$	Dev
Comp	10	2	0.11 (0.02)	0.01 (0.01)	0.03 (0.02)	0%	0%	0%	0%	0%	0%	0	0	5%
		6	3.09 (5.70)	0.09 (0.05)	0.09 (0.05)	0%	0%	0%	0%	0%	0%	0	0	30%
	20	2	0.31 (0.12)	0.13 (0.32)	0.41 (0.92)	0%	0%	0%	0%	0%	0%	14	14	8%
		6	44.96 (45.99)	0.83 (2.04)	0.85 (2.07)	0%	0%	0%	0%	0%	0%	12	12	25%
		10	585.15 (908.89)	4.74 (13.83)	4.79 (13.89)	40%	0%	0%	7%	0%	0%	27	27	38%
	30	2	0.74 (0.20)	0.55 (1.17)	1.67 (3.46)	0%	0%	0%	0%	0%	0%	52	52	4%
		6	578.80 (744.90)	64.15 (328.49)	62.05 (317.06)	0%	0%	0%	0%	0%	0%	791	791	24%
		10	1346.49 (986.54)	5.06 (9.97)	5.26 (10.83)	87%	0%	0%	9%	0%	0%	23	23	30%
	40	2	1.77 (0.48)	21.24 (69.87)	21.17 (69.36)	0%	0%	0%	0%	0%	0%	589	589	4%
		6	1224.02 (1069.98)	101.23 (286.68)	101.44 (287.16)	40%	0%	0%	9%	0%	0%	1408	1408	27%
		10	TL (–)	120.52 (346.62)	121.34 (348.42)	100%	0%	0%	16%	0%	0%	600	600	33%
	50	2	2.43 (0.87)	28.20 (54.64)	28.21 (54.54)	0%	0%	0%	0%	0%	0%	683	683	6%
		6	2482.97 (0)	533.88 (647.74)	529.82 (641.57)	83%	50%	50%	17%	36%	36%	20203	20261	28%
		10	TL (–)	57.41 (33.19)	57.88 (33.45)	100%	0%	0%	27%	0%	0%	270	270	34%
Cycle	10	2	0.11 (0.02)	0.01 (0.01)	0.03 (0.02)	0%	0%	0%	0%	0%	0%	0	0	5%
		6	1.08 (1.27)	0.09 (0.09)	0.10 (0.09)	0%	0%	0%	0%	0%	0%	1	1	13%
	20	2	0.31 (0.12)	0.13 (0.32)	0.41 (0.92)	0%	0%	0%	0%	0%	0%	14	14	8%
		6	29.06 (47.91)	2.37 (7.25)	2.39 (7.21)	0%	0%	0%	0%	0%	0%	73	73	4%
		10	15.62 (32.95)	0.97 (1.81)	0.99 (1.81)	3%	0%	0%	6%	0%	0%	3	3	1%
	30	2	0.74 (0.20)	0.55 (1.17)	1.67 (3.46)	0%	0%	0%	0%	0%	0%	52	52	4%
		6	309.24 (669.25)	14.57 (35.85)	14.82 (36.06)	7%	3%	3%	16%	7%	7%	1449	1454	4%
		10	240.64 (781.95)	9.91 (18.14)	9.97 (18.12)	27%	0%	0%	9%	0%	0%	140	140	1%
	40	2	1.77 (0.48)	21.24 (69.87)	21.17 (69.36)	0%	0%	0%	0%	0%	0%	589	589	4%
		6	642.03 (1128.76)	147.90 (579.12)	145.84 (569.60)	17%	17%	17%	40%	22%	22%	7327	7365	5%
		10	26.54 (27.56)	28.66 (47.35)	29.14 (50.48)	33%	3%	3%	22%	3%	3%	1333	1341	1%
	50	2	2.43 (0.87)	28.20 (54.64)	28.21 (54.54)	0%	0%	0%	0%	0%	0%	683	683	6%
		6	251.40 (302.48)	46.25 (43.98)	46.77 (44.39)	33%	17%	17%	53%	2%	2%	5465	5487	5%
		10	23.32 (24.00)	229.11 (250.34)	229.59 (250.18)	33%	17%	17%	25%	100%	100%	5601	5614	2%
Line	10	2	0.11 (0.02)	0.01 (0.01)	0.03 (0.02)	0%	0%	0%	0%	0%	0%	0	0	5%
		6	0.79 (1.05)	0.07 (0.06)	0.08 (0.06)	0%	0%	0%	0%	0%	0%	0	0	7%
	20	2	0.31 (0.12)	0.13 (0.32)	0.41 (0.92)	0%	0%	0%	0%	0%	0%	14	14	8%
		6	31.42 (57.77)	0.60 (0.88)	0.61 (0.88)	0%	0%	0%	0%	0%	0%	11	11	1%
		10	16.00 (34.51)	0.58 (0.47)	0.61 (0.49)	3%	0%	0%	11%	0%	0%	7	7	0%
	30	2	0.74 (0.20)	0.55 (1.17)	1.67 (3.46)	0%	0%	0%	0%	0%	0%	52	52	4%
		6	115.28 (204.42)	6.07 (12.37)	6.06 (12.33)	17%	0%	0%	20%	0%	0%	126	126	2%
		10	278.66 (853.01)	2.57 (1.91)	2.58 (1.91)	27%	0%	0%	9%	0%	0%	21	21	0%
	40	2	1.77 (0.48)	21.24 (69.87)	21.17 (69.36)	0%	0%	0%	0%	0%	0%	589	589	4%
		6	214.17 (665.38)	195.66 (597.97)	194.14 (591.21)	30%	3%	3%	32%	3%	3%	3347	3351	3%
		10	20.25 (12.50)	21.56 (45.95)	21.69 (45.99)	33%	0%	0%	22%	0%	0%	221	221	0%
	50	2	2.43 (0.87)	28.20 (54.64)	28.21 (54.54)	0%	0%	0%	0%	0%	0%	683	683	6%
		6	249.02 (318.41)	527.34 (899.32)	523.99 (893.52)	33%	17%	17%	58%	2%	2%	7740	7873	2%
		10	44.63 (51.95)	95.29 (159.3)	95.15 (158.87)	33%	0%	0%	27%	0%	0%	166	166	1%
Matching	10	2	0.11 (0.02)	0.01 (0.01)	0.03 (0.02)	0%	0%	0%	0%	0%	0%	0	0	5%
		6	0.63 (0.89)	0.05 (0.02)	0.06 (0.03)	0%	0%	0%	0%	0%	0%	0	0	2%
	20	2	0.31 (0.12)	0.13 (0.32)	0.41 (0.92)	0%	0%	0%	0%	0%	0%	14	14	8%
		6	79.95 (156.40)	0.38 (0.37)	0.40 (0.40)	0%	0%	0%	0%	0%	0%	5	5	1%
		10	4.53 (5.84)	0.35 (0.19)	0.37 (0.20)	7%	0%	0%	11%	0%	0%	3	3	0%
	30	2	0.74 (0.20)	0.55 (1.17)	1.67 (3.46)	0%	0%	0%	0%	0%	0%	52	52	4%
		6	271.20 (665.90)	110.58 (430.55)	109.45 (426.21)	23%	0%	0%	29%	0%	0%	1559	1559	1%
		10	4.93 (4.11)	2.13 (3.12)	2.14 (3.10)	33%	0%	0%	12%	0%	0%	23	23	0%
	40	2	1.77 (0.48)	21.24 (69.87)	21.17 (69.36)	0%	0%	0%	0%	0%	0%	589	589	4%
		6	57.05 (84.64)	4.18 (3.68)	4.25 (3.72)	33%	3%	3%	52%	3%	3%	989	992	2%
		10	10.39 (8.44)	10.88 (18.71)	10.72 (17.55)	33%	0%	0%	19%	0%	0%	85	85	0%
	50	2	2.43 (0.87)	28.20 (54.64)	28.21 (54.54)	0%	0%	0%	0%	0%	0%	683	683	6%
		6	260.67 (323.67)	58.34 (65.40)	58.63 (65.77)	33%	0%	0%	66%	0%	0%	643	643	1%
		10	18.84 (12.24)	10.50 (7.85)	10.68 (7.82)	33%	0%	0%	16%	0%	0%	19	19	0%
Star	10	2	0.11 (0.02)	0.01 (0.01)	0.03 (0.02)	0%	0%	0%	0%	0%	0%	0	0	5%
		6	1.03 (1.33)	1.52 (6.07)	1.52 (6.06)	0%	0%	0%	0%	0%	0%	85	85	14%
	20	2	0.31 (0.12)	0.13 (0.32)	0.41 (0.92)	0%	0%	0%	0%	0%	0%	14	14	8%
		6	48.58 (98.70)	4.83 (24.08)	4.83 (24.01)	0%	0%	0%	0%	0%	0%	179	179	5%
		10	310.28 (736.97)	0.49 (0.43)	0.51 (0.43)	23%	7%	7%	7%	7%	7%	3662	3797	7%
	30	2	0.74 (0.20)	0.55 (1.17)	1.67 (3.46)	0%	0%	0%	0%	0%	0%	52	52	4%
		6	312.57 (659.61)	2.93 (4.39)	2.97 (4.42)	17%	7%	7%	28%	4%	4%	3041	3066	4%
		10	204.10 (701.49)	2.17 (4.35)	2.23 (4.35)	43%	17%	17%	9%	4%	4%	8397	8481	7%
	40	2	1.77 (0.48)	21.24 (69.87)	21.17 (69.36)	0%	0%	0%	0%	0%	0%	589	589	4%
		6	57.05 (84.64)	4.18 (3.68)	4.25 (3.72)	33%	3%	3%	52%	3%	3%	989	992	2%
		10	10.39 (8.44)	10.88 (18.71)	10.72 (17.55)	33%	0%	0%	19%	0%	0%	85	85	0%
	50	2	2.43 (0.87)	28.20 (54.64)	28.21 (54.54)	0%	0%	0%	0%	0%	0%	683	683	6%
		6	260.67 (323.67)	58.34 (65.40)	58.63 (65.77)	33%	0%	0%	66%	0%	0%	643	643	1%
		10	18.84 (12.24)	10.50 (7.85)	10.68 (7.82)	33%	0%	0%	16%	0%	0%	19	19	0%
Ring-Star	10	2	0.11 (0.02)	0.01 (0.01)	0.03 (0.02)	0%	0%	0%	0%	0%	0%	0	0	5%
		6	1.68 (2.16)	0.10 (0.36)	0.30 (0.91)	0%	0%	0%	0%	0%	0%	7	8	19%
	20	2	0.31 (0.12)	0.13 (0.32)	0.41 (0.92)	0%	0%	0%	0%	0%	0%	14	14	8%
		6	17.07 (24.42)	3.41 (9.93)	7.17 (22.99)	0%	0%	0%	0%	0%	0%	242	227	9%
		10	143.76 (315.97)	22.89 (120.98)	24.06 (124.39)	0%	0%	0%	0%	0%	0%	1212	1209	7%
	30	2	0.74 (0.20)	0.55 (1.17)	1.67 (3.46)	0%	0%	0%	0%	0%	0%	52	52	4%
		6	292.19 (677.35)	4.43 (9.79)	5.48 (9.64)	0%	10%	10%	0%	7%	7%	8523	3271	8%
		10	439.26 (684.12)	15.35 (45.86)	19.16 (45.97)	20%	0%	0%	11%	0%	0%	676	651	7%
	40	2	1.77 (0.48)	21.24 (69.87)	21.17 (69.36)	0%	0%	0%	0%	0%	0%	590	590	4%
		6	392.52 (631.57)	32.68 (73.44)	32.72 (73.22)	13%	33%	33%	25%	34%	34%	10959	11044	10%
		10	378.73 (827.33)	79.19 (183.46)	79.86 (184.70)	40%	17%	17%	18%	24%	24%	6671	6753	8%
	50	2	2.43 (0.87)	28.20 (54.64)	28.21 (54.54)	0%	0%	0%	0%	0%	0%	683	683	6%
		6	394.67 (316.61)	79.50 (74.41)	80.29 (75.40)	33%	33%	33%	33%	7%	7%	9364	9493	10%
		10	70.98 (49.37)	69.46 (24.96)	69.81 (24.86)	50%	33%	33%	23%	52%	52%	9130	9155	8%

Table 3.2: Results of our computational experiments for the Non Linear and the branch-and-cut approaches.

In Figure 3.8 we show the comparison on the performance of the CPU Times averaged by number of demand points for each of the graph types for both the Non Linear formulation and the branch-and-cut procedure $\texttt{B\&C}_1$, showing again that the branch-and-cut approaches outperform the non linear formulation in most of the instances.

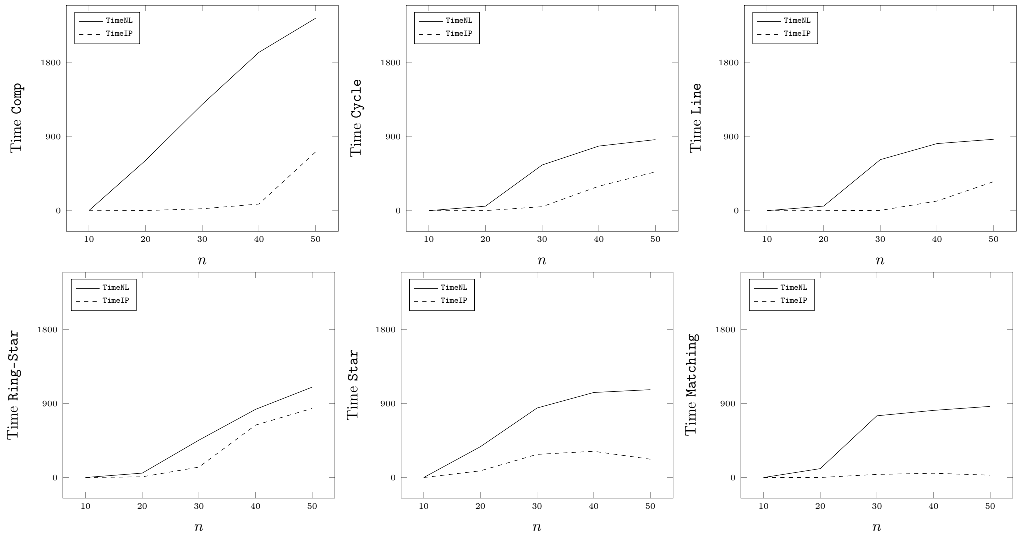


Figure 3.8: Graphics of averaged CPU times for the different graphs by number of demand points.

3.6.3 Computational performance of the matheuristic

Finally, we run our matheuristic algorithm (Section 3.5) on different larger size instances to test its performance. First, we run the approach for the 50-points instance from the work of Eilon et al. (1974), in order to test the accuracy of the obtained solution with respect to best solution obtained with the exact approach. In Table 3.3 we report the required CPU times for solving the instances as well as the deviation of the objective value with respect to the best solution obtained with $\texttt{B\&C}_1$.

We run the matheuristic for the two different aggregation schemes (Cluster and PTF) with 30 aggregated points.

As expected, the matheuristic approaches obtained solutions close to the best ones obtained with the exact approach. In fact, in 10.2% of these instances, the exact approach was not able to certify optimality within the time limit while the matheuristic was able to obtain good quality solutions in much lower CPU times. More concretely, in 9% of the instances the matheuristic obtained strictly better solutions than the exact approach. In 65% of the instances the matheuristic obtained solutions with a deviation less than 5% with respect to the exact approach and in 90% of the instances the deviation was smaller than 10%.

Concerning the two strategies that we used for the matheuristic approach, PTF outperforms the 30-Means in terms of both accuracy and CPU time. Thus, in the rest of the

		B&C$_1$	30-Means		PTF	
Graph	p	Time	Time	Dev	Time	Dev
Comp	2	28.20	1.09	4.10%	1.74	3.86%
	6	2069.58	10.76	-10.06%	32.88	-11.04%
	10	50.20	84.92	8.94%	44.64	4.29%
Cycle	2	28.20	1.09	4.10%	1.67	3.86%
	6	633.77	5.82	8.33%	8.01	4.4%
	10	755.27	26.80	-8.96%	30.07	-14.69%
Line	2	28.20	1.09	4.10%	1.69	3.86%
	6	953.57	4.95	9.85%	3.87	5.99%
	10	80.29	5.37	7.49%	12.40	3.13%
Matching	2	28.20	1.09	4.10%	1.91	1.92%
	6	49.48	2.38	2.88%	3.35	1.85%
	10	9.98	4.09	3.71%	5.95	3.49%
Star	2	28.20	1.09	4.10%	1.69	3.86%
	6	17.84	2.82	7.57%	3.26	5.67%
	10	624.22	25.01	8.03%	16.37	5.39%
Ring-Star	2	28.20	1.09	4.10%	1.72	5.38%
	6	1244.99	45.05	13.25%	27.11	4.21%
	10	1250.11	63.34	-6.14%	23.69	-12.44%

Table 3.3: Results of the matheuristic for the 50-points instance by Eilon et al. (1974).

computational experiments we use the PTF as the default scheme for our matheuristic algorithm.

We also analyze our matheuristic on larger real-world instances with sizes 200 (AP-dataset from Ernst and Krishnamoorthy, 1996), 324, 500, 708 and 818 (coordinates and demands in São José dos Campos from Senne et al., 2010) and 2863 (Taillard, 2003). We set in these experiments $p = 10$, but the same radii and distance limits as in the previous experiments. The matheuristic algorithm used the PTF aggregation strategy with 30 points.

In Table 3.4 we show the average results of our matheuristic on these 180 instances. We report the CPU times required by the approach in each of its phases: the construction of initial solutions (`Initial`), the location-allocation (`Loc-Alloc`) and the improvement (`Impr`). We also report the average total CPU time to solve the instances (`Total`). In order to check the convenience of the improvement phase, we also report the deviation of the solution obtained after the improvement phase with respect to the one constructed before this phase (`Dev_Impr`).

We also run B&C$_1$ for these instances, but the approach only obtained feasible solutions, within the time limit, for the dataset with 200 demand points, where it was able to solve 26 out of the 36 instances. In these solutions, the instances in which optimality was certified (15 instances), the average deviation of the heuristic approach with respect B&C$_1$ was 0.27%.

The last column reported in Table 3.4 shows that the improvement phase is very convenient since it clearly *improves* the solution obtained after the location-allocation phase. In this phase, the solution improves, in average, 4.88% with respect to the one obtained in the previous phase. Indeed, there is an instance in which in this phase improves 42.81%. On the other hand, in 17.59% of the instances the solution is not improved. The CPU times required for this phase are reasonable, needing tiny times for all the instances except for the largest one.

		Average times (sec.)				
Graph	n	Initial	Loc-Alloc	Impr	Total	Dev_Impr
200	Comp	11.23	3.15	0.47	14.85	2.53%
	Cycle	2.42	27.86	0.41	30.70	3.06%
	Line	1.94	18.54	0.35	20.83	2.57%
	Matching	1.16	1.66	0.38	3.20	5.8%
	Star	1.61	25.67	0.32	27.60	2.29%
	Ring-Star	3.72	126.32	0.55	130.59	1.53%
	Average	3.68	33.87	0.41	37.96	2.96%
324	Comp	120.56	148.02	0.66	269.24	6.52%
	Cycle	2.12	27.08	0.85	30.04	4.35%
	Line	1.77	26.08	0.64	28.50	3.81%
	Matching	1.16	1.68	0.72	3.56	9.47%
	Star	1.92	72.04	1.05	75.02	2.96%
	Ring-Star	3.37	39.37	0.63	43.37	3.26%
	Average	21.82	52.38	0.76	74.96	5.06%
500	Comp	356.95	8.15	1.21	366.31	5.47%
	Cycle	2.51	39.21	1.12	42.84	2.82%
	Line	2.06	2.76	1.15	5.97	4.75%
	Matching	1.48	4.30	1.19	6.97	6.67%
	Star	1.86	102.22	2.01	106.08	4.69%
	Ring-Star	3.73	21.30	1.56	26.59	3.3%
	Average	61.43	29.66	1.37	92.46	4.62%
708	Comp	365.90	5.54	3.68	375.11	4.71%
	Cycle	2.37	10	5.33	17.70	5%
	Line	1.71	8.74	3.78	14.23	4.02%
	Matching	1.32	1.21	1.53	4.07	9.76%
	Star	1.97	22.70	6.47	31.14	3.01%
	Ring-Star	3.87	29.91	11.98	45.77	3.65%
	Average	62.86	13.02	5.46	81.34	5.03%
818	Comp	371.25	223.11	2.57	596.93	7.72%
	Cycle	1.95	38.58	14.23	54.76	4.63%
	Line	1.62	2.94	4.87	9.42	3.78%
	Matching	1.13	2.36	2.86	6.35	13.15%
	Star	1.65	9.08	4.77	15.51	3.83%
	Ring-Star	3.54	47.11	7.30	57.95	3.82%
	Average	63.52	53.86	6.10	123.49	6.16%
2863	Comp	366.72	126.75	199.88	693.34	7.32%
	Cycle	2.29	2.44	120.08	124.82	4.01%
	Line	1.73	4.19	625.25	631.18	4.45%
	Matching	1.20	1.42	17.28	19.91	7.78%
	Star	1.84	60.32	74.57	136.73	4.68%
	Ring-Star	3.34	35.58	277.22	316.14	4.48%
	Average	62.85	38.45	219.05	320.35	5.45%

Table 3.4: Results of the Heuristic for different n data-sets.

One may note that the matheuristic is capable of obtaining feasible solutions for the MCLPIF in small times, even for instances for which the exact approach is not able to provide a single feasible solution within the time limit of one hour. As can be observed

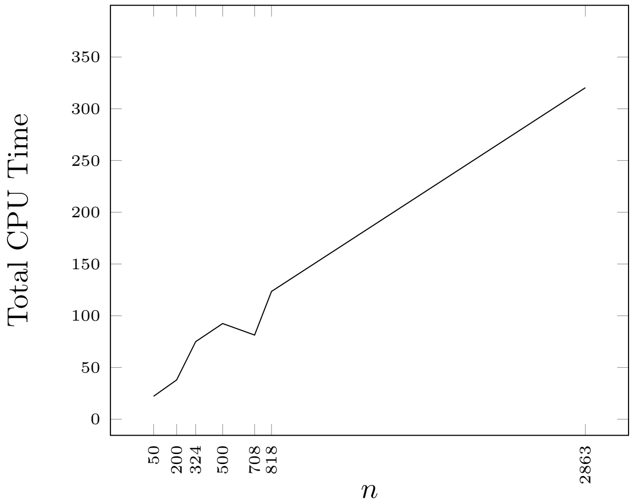


Figure 3.9: Average CPU times by instance size with the matheuristic algorithm.

from Table 3.4, the matheuristic scales very well. We represent in Figure 3.9 the average CPU times for each of the size instances run in these experiments where it seems that there is a linear trend in the CPU times when increasing the number of demand points.

3.7 Conclusions

In this chapter, we analyze a novel version of the Continuous Maximal Covering Location Problem in which the facilities are required to be linked through a given graph structure provided that the distance between the linked facilities does not exceed a given limit. We provide a general framework for the problem for any finite dimensional space and any ℓ_τ-norm based distance and we formulate it as a Mixed Integer Second Order Cone Optimization problem. We further analyze the geometry of the problem and prove that the continuous variables of the formulation can be projected out and the nonlinear constraints can be replaced by polynomially many linear constraints, resulting in a compact pure Integer Linear Optimization problem. We also derive two branch-and-cut solution strategies for solving the problem based on different relaxations for the MCLPIF. Finally, we develop a matheuristic algorithm which is capable to obtain good quality solutions for larger instances in reasonable CPU times. We test all the approaches in an extensive battery of computational experiments.

Further research on the topic includes, among others, the consideration of more sophisticated graph structures in the MCLPIF, as trees, or in general, λ-connected graphs. Although the geometrical analysis derived in this chapter can be exploited, unlike what happens for the graph structures studied in this chapter, the linear constraints defining the x-variables must be incorporated to the models, in the form of the so-called *connected* *subgraph polytope*, increasing considerably the difficulty of the problem. One possibility is to model the λ-connectivity of the spanning subgraphs of facilities via cut sets in the x-variables, i.e., constraints in the form $\sum_{l \in S: j<l} x_{jl} + \sum_{l \in S: l<j} x_{lj} \geq \lambda x_{jk}$, for all $j, k \in N$ and $S \subset P$ with $k \in S$ and $j \notin S$. These constraints are exponentially many, and its incorporation must be done via a branch-and-cut approach which in turns implies studying a separation oracle for the violation of λ-connectivity constraints. The separation of connectivity constraints is a topic that has been intensively applied in node and arc routing problems (see e.g., Padberg and Grötschel, 1985). One choice to detect one or more of the violated constraints is by computing the Gomory-Hu tree of the graph constructed with a solution of the relaxed problem, $\bar{x}$, using the procedure proposed by Gusfield (1990).

Also, it would be interesting to extend the *interconnection* framework to other continuous multifacility location problems. In particular, to continuous multifacility ordered median location problems (Blanco et al., 2016; Nickel and Puerto, 2006), which are of great interest in the location community since they allow unifying, in a single model, most of the existing continuous location problems in the literature. A first step could be done by using block norms (Ward and Wendell, 1985) since in that case, the distances can be represented by linear constraints (Nickel and Puerto, 2006).

Chapter 4

Multitype maximal covering location problems: hybridizing discrete and continuous problems

This chapter introduces a general modeling framework for a multitype maximal covering location problem in which the position of facilities in different metric spaces are simultaneously decided to maximize the demand generated by a set of points. From the need of intertwining location decisions in discrete and in continuous sets, a general hybridized problem is considered in which some types of facilities are to be located in finite sets and the others in continuous metric spaces. A natural non-linear model is proposed for which an integer linear programming reformulation is derived. A branch-and-cut algorithm is developed for better tackling the problem. The study proceeds considering the particular case in which the continuous facilities are to be located in the Euclidean plane. In this case, taking advantage from some geometrical properties it is possible to propose an alternative integer linear programming model. The results of an extensive battery of computational experiments performed to assess the methodological contribution of this chapter is reported on. The data consists of up to 920 demand nodes using real geographical and demographic data.

4.1 Introduction

A feature shared by most of the exiting literature focusing the MCLP concerns the existence of a single type of facility. However, in practice, this may not be the case. If not by other reasons, the progressive technology development often calls for older equipment that is still operational to be used together with more recent one. Another possibility emerges when two technologies can be looked at as complementing each other. For instance, when locating equipment for early fire detection in forests, surveillance facilities requiring human resources operating them may be complemented with equipment such as remotely controlled cameras to ensure a better coverage of the area of interest. When facilities can be installed in different phases (e.g. multi-period facility location) the facilities to be located in each phase can also be looked at as belonging to a different group (that we still call type) of facilities.

In this chapter we investigate maximal covering location problems with multiple facility types. We assume that the number of facilities of each type to be located is known beforehand, that each type of facilities is characterized by the shape of their coverage areas and the metric space from which they are selected. A plan is to be devised for a multi-stage process with each stage corresponding to installing one type of facility. The facilities opened in each stage perform the same tasks and thus complement the facilities installed in previous stages. In turn, they will be complemented by the facilities to be located in the subsequent stages. We show that instead of making sequential decisions for each facility type, coverage gains can be achieved by making an integrated decision involving all facility types.

We start by presenting a general formulation for the problem. Afterwards, motivated by some practical settings we investigate the hybridization of discrete and continuous facility location. We consider that several types of facilities are to be selected in finite sets of possibilities (one for each type) whereas the other types of facilities can be located continuously in the whole space.

By considering a hybrid setting it becomes possible to take advantage from choosing some services in finite sets of pre-specified preferred locations and then deciding flexible positions of the servers in the whole space. This setting can be useful, for instance, in telecommunications networks with a certain number of the servers (sensors, antennas, routers, etc.) being located inside adequately prepared infrastructures (buildings, offices, air-conditioned cabins, roofs, etc.) and additional servers being located at any place in the given space. The goal is of course to capture/cover as much demand as possible no matter the equipment doing it. The continuous facilities can be looked at as a set of servers to be located in the future and that must complement the equipment located in a discrete setting. To decide the positions of the centers, one could proceed by first locating the initial centers (in a fully discrete framework) maximizing the covered demand and then locate the future centers (in a fully continuous framework), maximizing the covered demand of the customers that are still uncovered by the initial servers. Although allowing the application of well-known existing tools in the context discrete and continuous maximal covering location, this procedure may easily lead to sub-optimal solutions: a better planning (i.e., covering more demand) can be obtained by considering an integrated approach which is what we propose.

The literature is quite rich when it comes to considering multitype facility location problems. Nevertheless, more often than not, we are led to problems stemming from logistics or telecommunications applications in which a multi-layered or a hierarchical facility structure is to be setup. In such a case, each facility type lies within a specific layer of the network or in the hierarchy and has a specialized function. The reader can refer to Contreras and Ortiz-Astorquiza (2019) and Heckmann and Nickel (2019) as well as to the references there in for overviews on many such problems.

In the current work, we are concerned with facilities that provide the same service, and thus can be used to complement each other although having some different characteristics. Wu et al. (2006) investigated one such problem in the context of capacitated facility location. In that paper, general setup costs are considered that depend on the size and location of the facility. The problem lies in the context of fixed-charge facility location (Fernández and Landete, 2019).

Mesa (1991) investigated several multi-period problems on networks. In particular, the author introduced the so-called absolute multi-period $(\alpha_1, \ldots, \alpha_{|T|})-$median problem where T stands for the number of periods in the planning horizon. This is possibly the first multi-period extension of the network p-median. We can look at the facilities to be located in each period as being of different types. Unlike we are considering in the current chapter, some facilities may just replace others (the former are closed when the latter are opened) and the location space is the same for all facilities.

Berman and Drezner (2008) consider a two-type discrete facility location problem. This is a p-median problem under uncertainty consisting of locating p initial facilities plus an uncertain number of extra additional ones. In both cases, the potential facilities to open belong to a finite set that coincides with the demand points and thus the problem is cast as a stochastic discrete p-median problem. In the current chapter we assume that we know beforehand the number of each type of facility to locate. Furthermore, we consider specific location spaces according to the different types of facilities.

Heyns and van Vuuren (2018) investigate a problem in which multiple types of facilities can be located in specific zones identified beforehand. Type-specific location requirements are assumed for the facilities. In each zone a finite set of candidate locations for the facilities are assumed, i.e., a pure discrete facility location setting is adopted. All facility types can in principle be located in all zones.

Considering also a finite set for locating the facilities, we find works considering a hierarchy between the facility-types in line to the models discussed by Contreras and Ortiz-Astorquiza (2019), i.e., the facilities in a higher level extend the service provided by the facilities in lower levels. Moore and ReVelle (1982) are possibly the first authors introducing such type of problem. In each potential facility location one must decide the type of facility to locate (if some). Other works deepening this type of analysis include Espejo et al. (2003), Ratick et al. (2009), and Xia et al. (2009). More recently, Küçükaydın and Aras (2020) also investigate a multitype discrete facility location problem but they consider so-called consumer preference. Each demand point has a preference for one facility type. The facilities are to be located in such a way that an optimal coverage in terms of the consumer preference is achieved.

In the current chapter we go beyond the existing literature by proposing a general modeling framework for a multitype maximal covering location problem. We do not restricted the problem to a discrete setting. Instead, we assume some location space for each facility type. The general framework proposed is motivated by some applications calling for hybridizing discrete and continuous facility location problems. For this reason, we deepen the study by considering that hybridized case: several types of facilities are to be located in a finite set of possible locations with their service being complemented by other facility types that can be located anywhere in underlying continuous space. We propose a 'natural' non-linear model that nonetheless, raises some computational difficulties. For this reason we also develop an integer linear model. Afterwards we focus on on the Euclidean plane. This allows using other types of modeling frameworks that we also investigate. We report on a series results obtained from a series of computational tests performed to assess the different models proposed. Real geographical data is consider in these tests.

The remainder of the chapter is organized as follows. In Section 4.2 the investigated problem is detailed and a general mathematical model is introduced. Section 4.3 specializes the general modeling framework to a hybridized discrete-continuous setting. Afterwards, in Section 4.4 we focuses on the the Euclidean plane provided additional insights in this case. The results of the computational tests performed to asses different developments proposed in the chapter are reported in Section 4.5. The chapter end with an further research line presented in Section 4.6 where some models were presented to capture uncertainty in multitype MCLP problems. And an overview of the work done and some other suggestions for further research were presented in Section 4.7.

4.2 The Multitype Maximal Covering Location Problem

Consider a finite set $\mathcal{A} = \{a_1, \ldots, a_n\}$ of demand points in $\mathbb{R}^d$ indexed in set $N = \{1, \ldots, n\}$, each of which with a weight given by a non-negative value c_i representing the demand of node a_i, for all $i \in N$. Throughout the chapter we often call a demand point interchangeably by the node a_i or by the index i.

Let us assume that there is a finite set of facility types, indexed in a set $\mathcal{T} = \{1, \ldots, T\}$. A facility of type $t \in \mathcal{T}$ can be located in some metric space that we denote by $\mathbb{S}^{(t)} \subseteq \mathbb{R}^d$. Consider a distance function of interest, say $\|\cdot\|^{(t)}$, in $\mathbb{S}^{(t)}$. Also for a facility of type $t \in \mathcal{T}$ we consider a coverage radius, say $\rho(t)$, $t \in \mathcal{T}$. Given one such facility, we say that node a_i, $i \in N$, is *covered by the facility* if the distance between a_i and the facility does not exceed $\rho(t)$. The node a_i is said to be *covered* if there is at least one open facility (no matter its type) covering it. For any finite subset of open facilities of type $t \in \mathcal{T}$, $\mathcal{X}^{(t)} \subseteq \mathbb{S}^{(t)}$, we denote by $\mathcal{C}(\mathcal{X}^{(t)}) \subseteq N$ the indices of the nodes in $\mathcal{A}$ covered by at least one point in $\mathcal{X}^{(t)}$, i.e.,

$$\mathcal{C}(\mathcal{X}^{(t)}) = \{i \in N : \|a_i - b\|^{(t)} \leq \rho(t), \text{for some } b \in \mathcal{X}^{(t)}\}.$$

Given $\mathbf{p} = (p_1, \ldots, p_T) \in N \times \overset{T}{\cdots} \times N$, the problem that we call the $\mathbf{p}$-Multitype Maximal Covering Location Problem ($\mathbf{p}$-MTMCLP, for short) seeks to locate p_t facilities of type $t \in \mathcal{T}$ so that the covered demand is maximized.

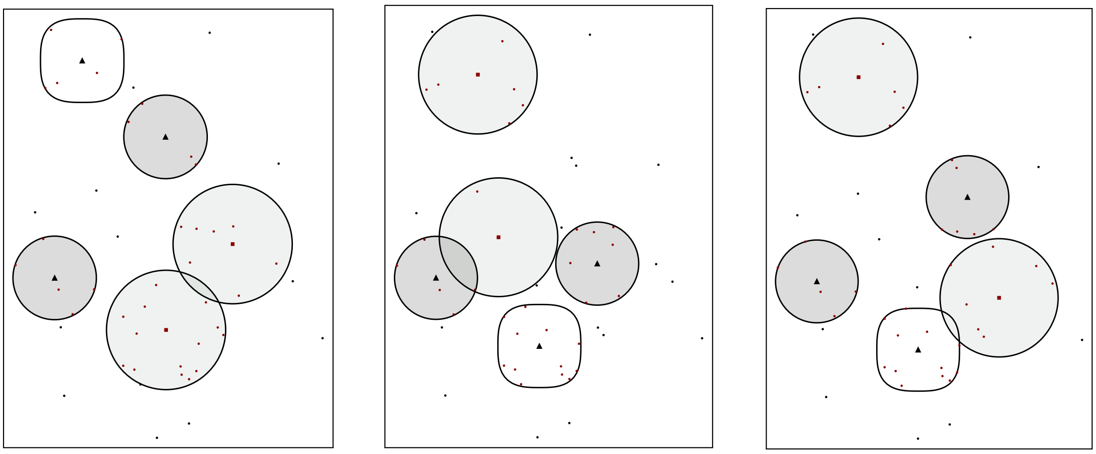
With the above notation, the $\mathbf{p}$-MTMCLP can be formally formulated as the following optimization problem:

$$\mathcal{V}(\mathbf{p}) := \max_{\mathcal{X}^{(t)} \subseteq \mathbb{S}^{(t)},\, |\mathcal{X}^{(t)}| = p_t} \sum_{i \in \bigcup_{t \in \mathcal{T}} \mathcal{C}(\mathcal{X}^{(t)})} c_i. \qquad (\mathbf{p}\text{-MTMCLP})$$

Observe that the difference between the different types of facilities to be located is the metric space where the locations are to be found as well as the coverage radii. In case the metric spaces coincide for two different types of facilities, one may also consider that they are of the same type and define different coverage radii, resulting in the same model.

The following example illustrates the problem we are investigating. Furthermore, it shows that a sequential decision making process in which we locate one type of facility in each step may lead to a sub-optimal solution.

Example 3. *We randomly generated a set of 50 demand nodes in* $[0,1] \times [0,1]$*—our set* $\mathcal{A}$*. Let us assume three types of facilities with* $\mathbb{S}^{(1)} = \mathcal{A}$ *and* $\mathbb{S}^{(2)} = \mathbb{S}^{(3)} = \mathbb{R}^2$*. Additionally, assume that* $\|\cdot\|^{(1)} = \|\cdot\|^{(2)}$ *are the Euclidean norm and* $\|\cdot\|^{(3)}$ *is the* ℓ_3*-norm. Regarding the coverage radii we take* $\rho(1) = 0.2$ *and* $\rho(2) = \rho(3) = 0.1$*. The number of facilities to open was fixed to* $\mathbf{p} = (2, 2, 1)$*. The weights* c_i *were all set equal to one.*



(a) $(\mathbb{S}^{(1)} \to \mathbb{S}^{(2)}\&\mathbb{S}^{(3)})$ (b) $(\mathbb{S}^{(2)}\&\mathbb{S}^{(3)} \to \mathbb{S}^{(1)})$ (c) **p**-MTMCLP

Figure 4.1: Sequential versus integrated decision making.

First, we look into locating two facilities in $\mathbb{S}^{(1)}$ *maximizing the number of covered points in* $\mathcal{A}$*, and after that we look for the best three additional facilities (two in* $\mathbb{S}^{(2)}$ *and one in* $\mathbb{S}^{(3)}$*) maximizing the number of covered points (i.e., covered demand) that were not already covered with the two initial facilities. We denote the resulting solution as* $(\mathbb{S}^{(1)} \to \mathbb{S}^{(2)}\&\mathbb{S}^{(3)})$*. Using the same instance, a different solution is obtained when we revert the sequence of decisions, i.e., we first look simultaneously for the two + one facilities in* $\mathbb{S}^{(2)}$ *and* $\mathbb{S}^{(3)}$*, respectively and after that we seek to find the additional two facilities in* $\mathbb{S}^{(1)}$ *maximizing the still not covered demand. The resulting solution is denoted by* $(\mathbb{S}^{(2)}\&\mathbb{S}^{(3)} \to \mathbb{S}^{(1)})$*. In a third experiment, we compute the solution using the integrated model* (**p**-MTMCLP)*.*

The solution $(\mathbb{S}^{(1)} \to \mathbb{S}^{(2)}\&\mathbb{S}^{(3)})$*, which is depicted in Figure 4.1a, is such that 74% of the demand nodes are covered. In case of solution* $(\mathbb{S}^{(2)}\&\mathbb{S}^{(3)} \to \mathbb{S}^{(1)})$*, depicted in Figure 4.1b, this percentage decreases to 66%. Finally, when we consider the solution obtained using model (***p***-MTMCLP)—Figure 4.1c—we obtain a 76% demand coverage. These results show that a sequential decision making process (even aggregating some types) may lead to a sub-optimal solution, which gives strength to the integrated modeling framework we are investigating.*

It is also worth noticing the change in the shape of the covered area associated with *facility to be located is* $\mathbb{S}^{(3)}$.

The above model is much general and accommodates many situations as discussed in the introductory section. In particular the location spaces $\mathbb{S}^{(t)}$, $t \in \mathcal{T}$, may correspond to finite sets, networks, continuous spaces or a combination of all. Heyns and van Vuuren (2018) and Küçükaydın and Aras (2020) consider a pure discrete setting. In the next section we focus on a case that raises some interesting challenges and that corresponds to having some types of facilities to be located in discrete spaces and the others located in continuous ones.

4.3 The hybridized discrete-continuous maximal covering location problem

In this section we propose suitable mathematical programming formulations for a wide family of problems in the shape of (**p**-MTMCLP) namely, the one that result when one assumes that the metric spaces $\mathbb{S}^{(t)}$ are either finite sets of points or the entire space $\mathbb{R}^d$.

We keep considering the set of demand points $\mathcal{A}$ already introduced as well as the index set for their elements, $N = \{1, \ldots, n\}$. We assume that two main families of types of facilities are to be located, say T_1 of type *discrete* and T_2 of type *continuous*, so we have a total of $T = T_1 + T_2$ types of facilities. In particular, we consider the set of type indices given by $\mathcal{T} = \mathcal{T}_1 \cup \mathcal{T}_2$ where $\mathcal{T}_1 = \{1, \ldots, T_1\}$, and $\mathcal{T}_2 = \{T_1 + 1, \ldots, T\}$.

For the first T_1 types of facilities, we consider that $\mathbb{S}^{(t)} = \{b_1^{(t)}, \ldots, b_{m^{(t)}}^{(t)}\}$ is a finite set of points in $\mathbb{R}^d$, indexed in set $M^{(t)} = \{1, \ldots, m^{(t)}\}$ that have been identified as potential locations for the facilities of type t, for $t \in \mathcal{T}_1$.

As in the general framework, a facility of type $t \in \mathcal{T}$ is endowed with a coverage radius $\rho(t)$. However, for the finite location spaces, we can go further in terms of coverage radii specification by assuming facility-dependent radii for the facilities in $\mathbb{S}^{(1)}, \ldots, \mathbb{S}^{(T_1)}$. We assume that a facility located at $b_j^{(t)}$ is endowed with a coverage radius equal to $\rho_j(t)$. In fact, the hybridized discrete-continuous setting we are considering is motivated by some practical applications (e.g. in telecommunication networks planning) an thus it makes sense to consider different coverage areas (typically larger) for the facilities chosen from the pre-specified sets $\mathbb{S}^{(1)}, \ldots, \mathbb{S}^{(T_1)}$ than those for the extra facilities—facilities located in $\mathbb{S}^{(T_1+1)}, \ldots, \mathbb{S}^{(T)}$ since the physical infrastructures may be prepared for a better service. Moreover, the fact that we know the potential locations in advance allows specifying a coverage radius that is location-specific. On the other hand, the common radii assumed for the different types of continuous facilities reflect a guaranteed coverage provided by the equipment no matter the point in the space it will end up being located.

For each $t \in \mathcal{T}$ we denote by P_t the index set for the facilities of type t to be located, i.e., $P_t = \{1, \ldots, p_t\}$.

For $t \in \mathcal{T}_2$, we consider metric spaces $\mathbb{S}^t = \mathbb{R}^d$ such that $p_{T_1+1}, \ldots, p_T$ locations are to be found respectively, in each of these spaces for installing additional facilities. These facilities have a coverage radius equal to $\rho(t)$, respectively for type $t \in \mathcal{T}_2$. In case one requires the continuous facilities to be located in a specific region of $\mathbb{R}^d$ instead of the entire space, most of the results presented in this chapter can be adapted conveniently in case the sets $\mathbb{S}^{(t)}$ are polyhedra or second order cone representable sets, by adding the suitable constraints defining each specific set.

In the following example we illustrate the problem on a three-dimensional instance.

Example 4. *We randomly generated a set of 50 demand nodes in* $[0,1]\times[0,1]\times[0,1]-$*our set* $\mathcal{A}$. *In this case, we consider that* $\|\cdot\|^{(1)} = \|\cdot\|^{(2)}$ *is the Euclidean norm. Finally, the rest of parameters are selected in the same way: two types of facilities with* $\mathbb{S}^{(1)} = \mathcal{A}$ *and* $\mathbb{S}^{(2)} = \mathbb{R}^3$, *the coverage radii we take* $\rho(1) = 0.2$ *and* $\rho(2) = 0.1$, *the number of facilities to open was fixed to* $\mathbf{p} = (2,3)$, *and the weights* c_i *were all set equal to one.*

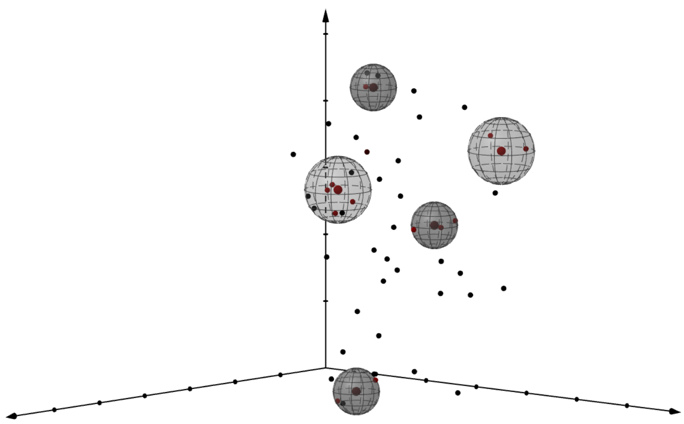


Figure 4.2: Solution of **p**-MTMCLP for the 3-dimensional instance of Example 4.

In the figure, the covered areas for the discrete facilities are drawn in light gray color while those of the continuous facilities are drawn in dark gray. Red dots indicate covered points and black dots the uncovered ones. The solution achieves an overall demand coverage of 32% *being* 16% *covered by each of the two types of facilities.*

4.3.1 A 'natural' non-linear model

The modeling framework (**p**-MTMCLP) derived in the previous section is of course valid in the hybridized discrete-continuous setting we are considering. The only distinguishing aspect is that in the definition of a set $\mathcal{X}^{(t)} \subseteq \mathbb{S}^{(t)}$ for the finite spaces (i.e., $t = 1, \ldots, T_1$) we must now consider location-specific radii namely, $\rho_j(t)$ for location $b_j^{(t)} \in \mathbb{S}^{(t)}$, $j \in M^{(t)}$.

For the sake of deriving a suitable mathematical programming formulation for the problem we introduce the following decision variables:

$$y_j^t = \begin{cases} 1, & \text{if facility } b_j^{(t)} \in \mathbb{S}^{(t)} \text{ is selected,} \\ 0, & \text{otherwise,} \end{cases} \quad \text{for all } t \in \mathcal{T}_1,\ j \in M^{(t)}.$$

$$x_i^t = \begin{cases} 1, & \text{if demand node } a_i \text{ is covered} \\ & \text{by the facilities located in } \mathbb{S}^{(t)}, \\ 0, & \text{otherwise,} \end{cases} \quad \text{for all } t \in \mathcal{T}_1,\ i \in N.$$

$$z_{ik}^t = \begin{cases} 1, & \text{if node } a_i \text{ is covered} \\ & \text{by the } k\text{-th facility in } \mathbb{S}^{(t)}, \\ 0, & \text{otherwise,} \end{cases} \quad \text{for all } i \in N,\ t \in \mathcal{T}_2,\ k \in P_t.$$

$X_k^t \in \mathbb{S}^{(t)}$: Coordinates of the k-th out of the p_t facilities located in $\mathbb{S}^{(t)}$, for $t \in \mathcal{T}_2$, $k \in P_t$.

Using the above decision variables, (**p**-MTMCLP) can be formulated as follows:

$$\max \sum_{i \in N} c_i \left[\sum_{t \in \mathcal{T}_1} x_i^t + \sum_{t \in \mathcal{T}_2} \sum_{k=1}^{p_t} z_{ik}^t \right], \tag{$\mathbf{p}$-MTMCLP$_1^{NL}$}$$

$$\text{s.t.} \sum_{j \in M^{(t)}} y_j^t = p_t, \ \forall t \in \mathcal{T}_1, \tag{$\mathbf{p}$-MTMCLP$_2^{NL}$}$$

$$x_i^t \leq \sum_{\substack{j \in M^{(t)}: \\ \|a_i - b_j^{(t)}\|^{(t)} \leq \rho_j(t)}} y_j^t, \ \forall i \in N, \ \forall t \in \mathcal{T}_1, \tag{$\mathbf{p}$-MTMCLP$_3^{NL}$}$$

$$\sum_{t=1}^{T_1} x_i^t + \sum_{t=T_1+1}^{T} \sum_{k=1}^{p_t} z_{ik}^t \leq 1, \ \forall i \in N, \tag{$\mathbf{p}$-MTMCLP$_4^{NL}$}$$

$$\|a_i - X_k^t\|^{(t)} \leq \rho(t) + U(1 - z_{ik}^t), \quad \forall i \in N, \ \forall t \in \mathcal{T}_2, \ \forall k \in P_t, \tag{$\mathbf{p}$-MTMCLP$_5^{NL}$}$$

$$x_i^t \in \{0,1\}, \ \forall t \in \mathcal{T}_1, \ \forall i \in N, \tag{$\mathbf{p}$-MTMCLP$_6^{NL}$}$$

$$y_j^t \in \{0,1\}, \ \forall t \in \mathcal{T}_1, \ \forall j \in M^{(t)}, \tag{$\mathbf{p}$-MTMCLP$_7^{NL}$}$$

$$z_{ik}^t \in \{0,1\}, \ \forall i \in N, \ \forall t \in \mathcal{T}_2, \ \forall k \in P_t, \tag{$\mathbf{p}$-MTMCLP$_8^{NL}$}$$

$$X_k^t \in \mathbb{S}^{(t)}, \ \forall t \in \mathcal{T}_2, \ \forall k \in P_t. \tag{$\mathbf{p}$-MTMCLP$_9^{NL}$}$$

In the above model, the objective function (**p**-MTMCLP$_1^{NL}$) measures the weighted coverage of the nodes in $\mathcal{A}$ using either the discrete or the continuous facilities; constraints (**p**-MTMCLP$_2^{NL}$) ensure that exactly p_t discrete facilities are selected in the finite set $\mathbb{S}^{(t)}$; Inequalities (**p**-MTMCLP$_3^{NL}$) state that node i is covered by a discrete facility iff there is such a facility covering it that has been opened. Inequalities (**p**-MTMCLP$_4^{NL}$) guarantee that at most one open facility "nominated" for each node and thus each weight c_i is accounted for at most once in the objective function. Constraints (**p**-MTMCLP$_5^{NL}$) ensure the proper definition of the z-variables. In these constraints, $\|a_i - X_k\|^{(t)}$ denotes the

$\|\cdot\|^{(t)}$-based distance between point a_j and the k-th continuous facility in $\mathbb{S}^{(t)}$ and U is a large enough value. Finally, constraints (**p**-MTMCLP$_6^{NL}$)–(**p**-MTMCLP$_9^{NL}$) state the domain of the decision variables.

The large U can be fine tuned. We can consider a common U for all $t \in \mathcal{T}$. In this case, any value $U \geq \max_{\substack{i,j \in N \\ t \in \mathcal{T}}} \|a_i - a_j\|^{(t)}$. Alternatively, for every $t \in \mathcal{T}$ we can set a t-specific value $U_t \geq \max_{i,j \in N} \|a_i - a_j\|^{(t)}$.

Observe that for $p_2 = \ldots, p_T = 0$, the problem to solve is a classical discrete p_1-MCLP, which can be formulated as above but omitting all terms involving facilities of types $2, \ldots, T$ and thus removing all the z- and X-variables. The case $p_t = 0 \; \forall t \in \mathcal{T} \backslash \{T_1 + 1\}$ and $p_{T_1+1} = 1$ reduces to the classical continuous p_{T_1+1}-MCLP which can be also formulated using a reduced set of variables.

Remark 3. (**p**-MTMCLP^{NL}) *is a Mixed-Integer Non-Linear Programming (MINLP) problem because of* (**p**-MTMCLP$_5^{NL}$)*. In case for some $t \in \mathcal{T}_2$ $\|\cdot\|^{(t)}$ is the Euclidean norm, such a type of constraints can be re-written as a set of linear and second-order cone inequalities given in* (ℓ_2-norm)*.*

Consequently, if all the types of continuous facilities use the Euclidean norm, (**p**-MTMCLP^{NL}) *simplifies to a mixed-integer second-order cone programming problem, which can be solved using any available off-the-shelf solver.*

Remark 4. *In Remark 3 one can replace Euclidean distances by polyhedral-norm based distances (deriving linear programming models using* (Pol-norm)*) or by ℓ_τ-norm (with $\tau \geq 1$) based distances inducing again mixed-integer second-order cone optimization problems (using the set of inequalities given in* (ℓ_τ-norm)*). One may even consider mixed distances (one for each demand point, if one desires to model different coverage areas).*

Remark 5. *Apart from regular coverage areas represented by convex surfaces, one could also represent non-convex coverage areas by means of unions of convex (second-order cone representable sets). This type of sets can be efficiently represented using disjunctive constraints that are usually modeled through binary variables (see Dolu et al., 2020, for further details on suitable representations of these regions in a location problem).*

4.3.2 An integer linear optimization model

The above mixed integer non-linear model becomes intractable for medium or large size instances of the problem even if Euclidean distances are considered. Therefore, to successfully tackle the problem other possibilities must to be considered.

Next, we derive an integer linear model based upon projecting out the X-variables —which represent the coordinates of the services—by ensuring that these can be *easily* found (in poly-time) once the different sets of demand points allocated to the same facility are known.

In what follows, we impose that every selected facility must cover at least one demand point, which, we believe, is a reasonable assumption in practice. Furthermore, we consider $T_1 = T_2 = 1$, i.e., a single type of discrete and a single type of continuous facilities are to be located. The presented results and models can be adapted to the general case. Nevertheless, this would make the contents of this section significantly more involved but not more informative. Accordingly, that there are p_1 discrete and p_2 continuous facilities to locate.

Lemma 1 defined in Chapter 1.2.3 allows us to rewrite constraints ($\mathbf{p}$-MTMCLP$_5^{NL}$) (for $T = 2$) as linear constraints and thus to formulate the *Hybridized* (p_1, p_2)*-Maximal Covering Location problem* (HMCLP, for short) as follows:

$$\max \sum_{i \in N} c_i \left[x_i + \sum_{k \in K} z_{ik} \right], \tag{HMCLP$_1^{IP}$}$$

$$\text{s.t.} \sum_{j \in M} y_j = p_1, \tag{HMCLP$_2^{IP}$}$$

$$x_i \leq \sum_{\substack{j \in M \\ \|a_i - b_j\|^{(1)} \leq \rho_j(1)}} y_j, \ \forall i \in N, \tag{HMCLP$_3^{IP}$}$$

$$x_i + \sum_{k=1}^{p_2} z_{ik} \leq 1, \ \forall i \in N, \tag{HMCLP$_4^{IP}$}$$

$$\sum_{i \in Q} z_{ik} \leq |Q| - 1, \forall k \in P_2, \ \forall Q \subseteq N : \bigcap_{i \in Q} \mathbb{B}_{\rho(2)}(a_i) = \emptyset, \tag{HMCLP$_5^{IP}$}$$

$$x_i \in \{0, 1\}, \ \forall i \in N,$$

$$y_j \in \{0, 1\}, \ \forall j \in M,$$

$$z_{ik} \in \{0, 1\}, \ \forall i \in N, \ \forall k \in P_2.$$

In this model, we have simplified some notation namely by removing the type index from the decision variables as well as from set M since we only have one type of discrete and one type of continuous facilities. Also in the above model, (HMCLP$_5^{IP}$) ensure that the set of points covered by a continuous facility verifies the condition of Lemma 1, i.e., sets of *incompatible* demand points are not allowed to be allocated to the same continuous facility. This constraint replaces the non-linear constraint in ($\mathbf{p}$-MTMCLP^{NL}).

Moreover, the above model does no include the variables X_k. In fact, once an optimal solution is obtained for (HMCLP^{IP}), we can use the values of the z-variables, say $\{\bar{\mathbf{z}}\}$, to find explicit optimal coordinates for the new facilities to install. In particular, the coordinates of the k-th facility to install ($k \in P_2$) can be given by any vector X_k satisfying:

$$X_k \in \bigcap_{\substack{i \in N: \\ \bar{z}_{ik} = 1}} \mathbb{B}_{\rho(2)}(a_i), \text{ for all } k \in P_2.$$

A center X_k in the above intersection can be found in polynomial time either solving a second order cone optimization problem or by solving a one-center facility location problem.

Solving (HMCLP^{IP}) requires incorporating exponentially many constraints—inequalities (HMCLP_5^{IP}). Interestingly, the above formulation can be simplified (reducing from exponential to polynomially many constraints in the form of (HMCLP_5^{IP})) by means of Helly's Theorem (Helly, 1923) (see also Danzer et al., 1963). Invoking that result, provided that the continuous space is $\mathbb{R}^d$, only intersections of $(d+1)$-wise balls are needed to check:

$$\mathbb{B}_{\rho(2)}(a_{i_1}) \cap \cdots \cap \mathbb{B}_{\rho(2)}(a_{i_{d+1}}),$$

for all $a_{i_1}, \ldots, a_{i_{d+1}} \in \mathcal{A}$.

Despite this simplification, the number of constraints may still be large and thus making the problem more difficult to solve. Instead, the problem can be tackled by considering an incomplete formulation (removing (HMCLP_5^{IP})) and iteratively incorporating these constraints *on-the-fly*, as needed.

The selection of the constraints to incorporate in each iteration is found using the following separation strategy: After solving (HMCLP^{IP}) with none or part of the constraints (HMCLP_5^{IP}) a solution, say $\bar{\mathbf{z}}$, is obtained. Then, for each $k \in P_2$ the define set $Q_k = \{i \in N : \bar{z}_{ik} = 1\}$. One can check for the validity of the set Q_k as a feasible cluster of demand points for our problem by solving the 1-center problem for the points in such a set. In case the optimal coverage radius obtained is less than or equal to $\rho(2)$, one knows that Q_k is a valid subset of demand points that can be covered by the same server. Otherwise, the solution violates the relaxed constraints, and thus we add the cut

$$\sum_{i \in Q_k} z_{ik'} \leq |Q_k| - 1, \ \forall k' \in P_2, \tag{4.3}$$

to ensure that such a solution is no further deemed feasible and thus obtained again.

The 1-center problem with Euclidean distances on the plane is known to be solvable in polynomial time (see e.g., Elzinga and Hearn, 1972). Extensions to higher dimensional spaces and generalized covering shapes have been recently proved to be also poly-time solvable (Blanco and Puerto, 2021a).

The above procedure can be embedded into a branch-and-cut approach by means of lazy constraints.

The following result holds, which helps finding (and thus ignoring) dominated cuts:

Proposition 3. *Let $\mathbf{z} \in \{0,1\}^{n \times p_2}$ and $Q, Q' \subseteq N$ be such that $Q \subset Q'$. Then, if $\mathbf{z}$ violates (HMCLP_5^{IP}) for the set Q violates, then, $\mathbf{z}$ also violates the constraint for the set Q'. Thus, the cut induced by Q strictly dominates the one induced by Q'.*

Proof. We suppose that Q violates the constraint (HMCLP_5^{IP}), this means $\bigcap_{i \in Q} \mathbb{B}_{\rho(2)}(a_i) =$

$\emptyset$ and we get $\sum_{i \in Q} z_{ik} > |Q| - 1$. Therefore,

$$\sum_{i \in Q'} z_{ik} = \sum_{i \in Q} z_{ik} + \sum_{i \in Q' \setminus Q} z_{ik} > |Q| - 1 + (|Q'| - |Q|) = |Q'| - 1, \ \forall k \in P_2.$$

Then, we have that Q' also violates the constraints and the cut induced by Q strictly dominates the one induced by Q'. □

In the next section we provide further details on how we can take advantage from these contents using a more specific setting.

4.4 The particular case of the Euclidean plane

In this section we focus on the particular case in which the continuous facilities are to be located in the Euclidean plane. This allows deepening the discussion already presented and also to consider an alternative model for the problem.

4.4.1 A branch-and-cut algorithm based on (HMCLP^{IP})

Let us take again the integer linear model (HMCLP^{IP}). In the case of the plane ($d = 2$), the application of Helly's Theorem described above guarantees that we only need to check intersection or 3-wise balls. In particular, we need to check if such intersections are empty. If so, we incorporate the adequate constraints to avoid searching for facilities in those intersections. Note that is true for every norm we adopt. Two cases may emerge for any set of three demand points $\{a_{i_1}, a_{i_2}, a_{i_3}\} \subset \mathcal{A}$:

Case 1: $\mathbb{B}_{\rho(2)}(a_{l_1}) \cap \mathbb{B}_{\rho(2)}(a_{l_2}) = \emptyset$ for some $l_1, l_2 \in \{i_1, i_2, i_3\}$.

In this case, points a_{l_1} and a_{l_2} are incompatible—they cannot be covered by the same center. Hence impose

$$z_{l_1 k} + z_{l_2 k} \leq 1, \ \forall k \in P_2. \qquad \text{(2-Wise)}$$

Case 2: The pairwise intersections are non-empty but $\mathbb{B}_{\rho(2)}(a_{i_1}) \cap \mathbb{B}_{\rho(2)}(a_{i_2}) \cap \mathbb{B}_{\rho(2)}(a_{i_3}) = \emptyset$.

In this case, the three points cannot be covered by the same facility and thus we impose

$$z_{i_1 k} + z_{i_2 k} + z_{i_3 k} \leq 2, \ \forall k \in P_2. \qquad \text{(3-Wise)}$$

Constraints (2-Wise) and (3-Wise) for all subsets of three points in $\mathcal{A}$ replace the constraint (HMCLP$_5^{IP}$) in formulation (HMCLP^{IP}) in the planar case. Although these constraints (in worst case) are $O(n^3)$, most of them are needless to construct the optimal solution of the problem.

The problem can now be solved considering an incomplete formulation (removing constraints (3-Wise) from (HMCLP^{IP})) and incorporating these constraints *on-the-fly*, as needed using a similar strategy than the one used for the general formulation and described above. It avoids checking the three-wise intersections of points which can be computationally costly for large instances. We embed this approach into a Branch-and-Cut scheme which is reinforced by the following elements:

Separation Oracle Given an optimal solution $\bar{\mathbf{z}}$ to the incomplete model, for every $k \in P_2$ we consider the set $Q_k = \{i \in N : \bar{z}_{ik} = 1\}$. The validity of the cluster Q_k is checked using the algorithm proposed by Elzinga and Hearn (1972) that computes (in polynomial time), the center and the minimum radius covering the demand points in Q_k.

To make this work self-contained, we recall that the algorithm proposed by Elzinga and Hearn (1972) is based on the construction of disks covering three points until the whole set is covered. Thus is accomplished sequentially in such a way that the covering radius increases at each step of the procedure.

We take further advantage from this procedure to incorporate more than a single cut in each iteration of the branch-and-cut algorithm. Considering a set of points Q_k, if at some step when applying the algorithm by Elzinga and Hearn (1972) the coverage radius becomes larger than $\rho(2)$ (it is not feasible to clustering the points in Q_k to be served by the same server), then all the sets of two or three points used to construct the minimum enclosing disks so far are used to generate constraints (2-Wise) and (3-Wise). This strategy has proven to alleviate the resolution of the problem.

Initial Pool of Constraints We propose procedure to generate an initial pool of constraints in the form of (3-Wise) to be included in the initial incomplete model. First, we construct clusters of demand points with maximum distance between two clustered points fixed to $\rho(2) + \varepsilon$, with $\varepsilon > 0$ (implemented in the Python module `scipy` through the function `fcluster`). At each of these clusters, we checked the validity of them as a solution of our MCLP using the same strategy used in our separation oracle. In case some of them is not valid, we incorporate the pool all the constraints of type (3-Wise) that are violated. The procedure is repeated for different values of ε.

Symmetries Our formulation (HMCLP^{IP}) is highly symmetric in the sense that any permutation of the k indices in the z-variables results in an alternative solution (with the same objective value). To break symmetries in our model thus hoping to speed up the resolution of the problem, we incorporate the following constraints that we can observe to be straightforwardly valid for the problem:

$$\sum_{i\in N} w_i z_{i(k-1)} \leq \sum_{i\in N} w_i z_{ik}, \ \forall k \in P_2\backslash\{1\}. \tag{4.4}$$

With these constraints, among all the possible sorting of facilities, we choose one producing a non-decreasing sequence of covered demand.

4.4.2 An alternative IP model

As largely explained, model (HMCLP^{IP}) is quite general; it is valid in metric spaces of any dimension $d \geq 2$ and for every distance of interest. In 2-dimensional spaces and in addition to the developments presented in Section 4.4.1, we can derive an alternative Integer Linear Programming formulation. This is accomplished by finding a finite dominating set, which is done using the discretization technique proposed by Church (1984) for the MCLP.

Let $\mathcal{B}$ be the set consisting of the demand points in $\mathcal{A}$ and also the intersection points of the pairwise intersections of the boundary of the $\|\cdot\|^{(2)}$-balls centered at the demand points with radius $\rho(2)$, i.e.,

$$\mathcal{B} = \mathcal{A} \cup \bigcup_{\substack{i,l\in N:\\ i<l}} \Big(\partial\mathbb{B}_{\rho(2)}(a_i) \cap \partial\mathbb{B}_{\rho(2)}(a_l)\Big),$$

where $\partial\mathbb{B}_{\rho(2)}(a)$ stands for the boundary of the ball $\mathbb{B}_{\rho(2)}(a)$. Inspired by the terminology used by Church (1984) the set $\mathcal{B}$ will be designated by the Balls Intersection Points Set (BIPS).

Lemma 4. *There exists an optimal solution to (p_1, p_2)-MTMCLP where the continuous facilities belong to the set $\mathcal{B}$.*

Proof. Let $\mathcal{X}^2 = \{\overline{X}_1, \ldots, \overline{X}_{p_2}\}$ be optimal positions for the continuous facilities. Clearly, $\overline{X}_k$ belongs to the intersection of the balls centered at the covered points, i.e.,

$$\overline{X}_k \in \bigcap_{i\in N\,:\, \|a_i - \overline{X}_k\|^{(2)} \leq \rho(2)} \mathbb{B}_{\rho(2)}(a_i).$$

Thus, we can replace $\overline{X}_k$ by any of those intersection points keeping the same coverage level. Clearly, these points belong to $\mathcal{B}$. □

Considering the set $\mathcal{B}$, it is possible to reformulate (p_1, p_2)-MTMCLP^{NL} as an integer linear programming problem, in which the selection of the continuous facilities is replaced by the search of the optimal $\mathcal{B}$-points to open. Let us denote by $\mathcal{B} = \{\gamma_1, \ldots, \gamma_{|\mathcal{B}|}\}$ and $L = \{1, \ldots, |\mathcal{B}|\}$, and consider the following sets of decision variables:

$$y_j^1 = \begin{cases} 1, & \text{if facility } b_j \text{ is selected,} \\ 0, & \text{otherwise,} \end{cases} \quad \forall j \in M,$$

$$y_l^2 = \begin{cases} 1, & \text{if point } \gamma_l \in \mathcal{B} \text{ is selected,} \\ 0, & \text{otherwise,} \end{cases} \quad \forall l \in L.$$

The problem can be reformulated as follows:

$$\max \sum_{i \in N} c_i x_i \tag{HMCLP$_1^{BIPS}$}$$

$$\text{s.t.} \sum_{j \in M} y_j^1 = p_1, \tag{HMCLP$_2^{BIPS}$}$$

$$\sum_{l \in L} y_l^2 = p_2, \tag{HMCLP$_3^{BIPS}$}$$

$$x_i \leq \sum_{\substack{j \in M \\ \|a_i - b_j\|^{(1)} \leq \rho_j(1)}} y_j^1 + \sum_{\substack{l \in L \\ \|a_i - \gamma_l\|^{(2)} \leq \rho(2)}} y_l^2, \ \forall i \in N, \tag{HMCLP$_4^{BIPS}$}$$

$$x_i \in \{0,1\}, \ \forall i \in N,$$

$$y_j^1 \in \{0,1\}, \ \forall j \in M,$$

$$y_l^2 \in \{0,1\}, \ \forall l \in L,$$

where constraint (HMCLP$_2^{BIPS}$) enforces opening exactly p_1 of the facilities from $\mathbb{S}^{(1)}$, while constraint (HMCLP$_3^{BIPS}$) ensures opening exactly p_2 of the additional (continuous) facilities in $\mathbb{S}^{(2)} = \mathbb{R}^2$. Constraints (HMCLP$_4^{BIPS}$) allow to determine whether a demand point is covered or not by any of the available open facilities (from $\mathbb{S}^{(1)}$ or $\mathcal{B}$). Note that in the above model the domain of the x-variables can be relaxed to the interval $[0,1]$.

The problem above is a particular version of the classical Discrete Maximal Covering Location problem in which two different types of facilities are desired to be open, p_1 of type discrete and p_2 of type continuous.

A major issue of concern in the above model is the number of y^2-variables, which coincides with the number of points in $\mathcal{B}$. This number is of order $O(n^2)$ considering one type of continuous facilities but will be of course larger if additional continuous facility types exist since we must add additional sets of $y-$variables—one for each facility type.

Still considering a single type of continuous facilities, the size of the set $\mathcal{B}$ can be reduced following the strategy proposed in Church (1984). However, again we must note that that author worked only with ℓ_1- and ℓ_2-norm for which a dominance relation between the points allows removing some elements from $\mathcal{B}$. In our case, although working in the Euclidean plane, we can consider distances other than the ℓ_1-norm and the ℓ_2-norm. This, again, creates some challenges to the above model since finding the BIPS is far from straightforward.

Overall, the model just proposed can be promising under a particular case: only one type of continuous facilities exist and the distances of interest reduce to ℓ_1 or ℓ_2 norms.

4.5 Computational experiments

In this section we report on the results of a series of computational experiments performed to empirically assess our methodological contribution for the hybridized MCLP presented in the previous sections.

4.5.1 The test data

To run the experiments, we made use of real geographic and demographic information from Manhattan Island, NY, USA. The data was collected from data.cityofnewyork.us. The main instance consists of the (planar) geographical coordinates of the 920 main buildings on the island with demand weights given by the number of people living in at each of the buildings. The complete data set used in our experiments is available in our repository github.com/vblancoOR/MTMCLP.

To test the scalability of the problem we are investigating, different subsets of the complete data we considered each with a different size. We sorted the location indices of the buildings according to demands and we made subsets from the buildings with the largest demands. We have considered subsets of cardinality $n \in \{400, 500, 700, 920\}$ with 920 corresponding to the complete data set. Figure 4.3 depicts the different sizes considered for the Manhattan data set.

(a) $n = 100$. (b) $n = 400$. (c) $n = 700$. (d) $n = 920$.

Figure 4.3: Different instances considered from the Manhattan dataset.

We assume one type of discrete facility and one type of continuous facility. Additionally, we suppose that the potential facility location for the discrete facilities correspond to buildings underlying the instance of the problem and we consider that the coverage radius is equal for all of them. As for facilities to locate we adopted $p_1, p_2 \in \{1, 2, 3, 4, 5\}$. Finally, we consider two radii for each type of facilities: for discrete facilities we consider

$\rho(1) \in \{0.008, 0.012\}$ and for the continuous radii the values are $\rho(2) \in \{0.005, 0.01\}$. These radii are adjusted to the real coordinates (latitude and longitude) of the demand points and are equivalent to $\{810, 1280\}$ (for $\rho(1)$) and $\{430, 1050\}$ (for $\rho(2)$) meters. Combining all the parameters, a total of 400 instances have been tested (four values for n, five values for p_1, five values for p_2, two values for ρ_1 and two values for ρ_2).

In the tests whose results we detail next, we run: (i) the non-linear formulation (**p**-MTMCLP^{NL}), (ii) the integer programming formulation (HMCLP^{IP}) (with constraints (2-Wise) and (3-Wise) and the Branch-and-Cut approach derived from an incomplete formulation of (HMCLP^{IP}) in which only the pair-wise intersection constraints are considered, and (iii) the integer programming formulation (HMCLP^{BIPS}),

All the experiments have been run on a virtual machine in a physical server equipped with 12 threads from a processor AMD EPYC 7402P 24-Core Processor, 64 Gb of RAM and running a 64-bit Linux operating system. The models were coded in Python 3.7 and we used Gurobi 9.1 as optimization solver. A time limit of 1 hour was fixed for all the instances.

4.5.2 Results

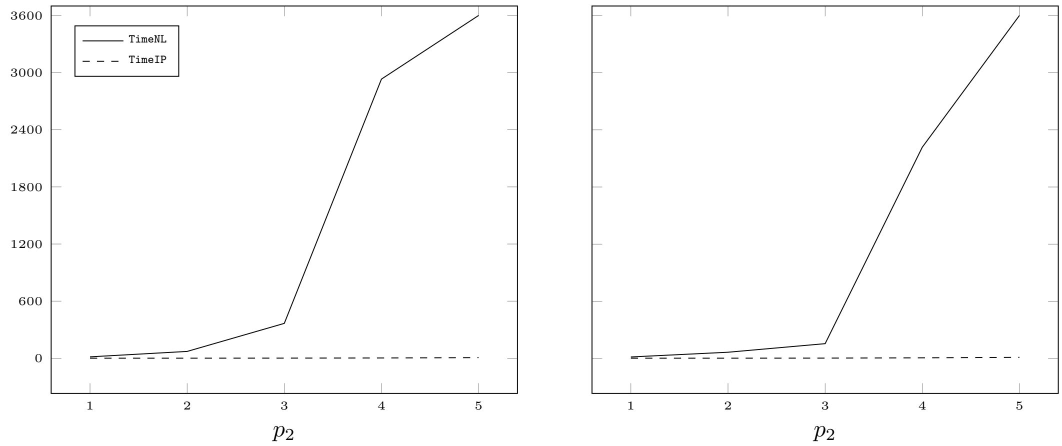
Figure 4.4 depicts results for the smallest instances with the purpose of comparing the non-linear formulation (**p**-MTMCLP^{NL}) and the complete pure integer formulation (HMCLP^{IP}). We show in that figure the average CPU times for both approaches by aggregating the different values of p_1 and $\rho(1)$. The instances with $n = 100$ demand nodes were considered in these tests.

We show how the computational times increase when the number of continuous facilities to locate increases for the two values of $\rho(2)$. In particular we observe that the CPU time reaches the time limit for $p_2 = 5$. These results give strong evidence to our intuition: the non-linear formulation is computationally demanding even for small instances of the problem.

Given the results presented in Figure 4.4 we decided to proceed the tests without the non-linear model.

Focusing on model (HMCLP^{IP}), Table 4.1 contains the number constraints (2-Wise) and (3-Wise) generated as well as the CPU time required for that generation. In this table we observe that the linear model grows a lot when the number of demand points increases, which makes is clearly more difficult to tackle. Moreover, the CPU time required to check all intersections increases significantly for constraints (3-Wise). Summing up, we easily conclude that making use of a branch-and-cut approach such as the one we propose in Section 4.4.1 is totally advisable in this case since we can avoid the clear burden that corresponds to computing and using all the constraints.

Next, we focus on the results corresponding to running the B&C procedure devised for tackling model (HMCLP^{IP}) and also the when formulation (HMCLP^{BIPS}) is adopted. The



(a) $\rho(2) = 0.005$ (b) $\rho(2) = 0.01$

Figure 4.4: Average of CPU times required by models (**p**-MTMCLP^{NL}) (straight line) and (HMCLP^{IP}) (dashed line) when solving the instances with 100 demand nodes.

common ground for this comparison is the Euclidean plane using the Euclidean distance. In fact, this is the setting in which we can use model (HMCLP^{BIPS}). Another possibility would be to use the ℓ_1-norm but then we would clearly be favor model (HMCLP^{BIPS}), which is already done using the Euclidean distance.

In tables (4.2) and (4.3) we present the results obtained. Each table refer to a different covering radius for the continuous facilities ($\rho(2) = 0.005$ and $\rho(2) = 0.01$, respectively) and are similarly organized. Both tables show the information aggregating the different values of the discrete facilities, p_1, and the radii of them, $\rho(1)$. The first three columns provide the details of the instance being solved: the radii used for the continuous facilities, the number of demand nodes, and the number of continuous facilities to locate. Next, seven blocks of columns are presented. In the blocks, the columns contains results for model (HMCLP^{BIPS}) (`BIPS`) and the branch-and-cut procedure devised for tackling model (HMCLP^{IP}) (`B&C`). Some blocks only contain information of the branch-and-cut approach since it is not applicable to the `BIPS` approach. The first block (columns 4 and 5) gives the overall CPU time in seconds (averaged only on the instances that were solved up to optimality within the time limit) required to solve the problem. This time includes: the time required by Gurobi to solve the IP models (second block) of solved instances; the pre-processing time—time for generating the `BIPS` in the case of Constraints (HMCLP$_4^{BIPS}$) and the time for building the initial pool of Constraints (2-Wise) in the case of model (HMCLP^{IP}) (third block); the time to generate the difficult constraint (HMCLP$_4^{BIPS}$) and its respective in the (HMCLP^{IP}) formulation (fourth block); callback times for the branch-and-cut approach (fifth block); percentage gap at termination and number of unsolved instances within the time limit– out of 10 (sixth block); and total number of constraints used (sixth block). The percentage gaps and the number of unsolved instances are not

		(2-Wise)		(3-Wise)	
n	$\rho(2)$	CPUTime	# Inequalities	CPUTime	# Inequalities
400	0.005	0.62	71780	490.70	1457
	0.01	0.55	61532	2658.11	5925
500	0.005	0.91	112404	1483.38	3701
	0.01	0.91	94279	9273.37	16600
600	0.005	1.35	161160	3997.49	6937
	0.01	1.29	133860	25277.75	30218
700	0.005	1.85	218260	9596.85	12171
	0.01	1.80	179211	61696.80	58717
920	0.005	3.14	369085	52788.62	40047
	0.01	3.09	290690	358966.03	184221

Table 4.1: Averaged number of Constraints (2-Wise) and (3-Wise) generated and the CPU time (seconds) required for their generation.

reported for the BIPS approach since all the instances were solved up to optimality within the time limit.

In view of the results of tables (4.2) and (4.3) we draw several conclusions. First, we realize that the difficulty in using model HMCLP^{BIPS} stems from loading constraints (HMCLP$_4^{BIPS}$). In fact, to accomplish this we need to check all the intersections. This takes a long time if the size of BIPS is large. Another challenge for that model regards the time to find the BIPS. When the number of demand nodes increases and the radius is large enough to require checking all (or a large majority of) points, then it takes a long time. From the above observations we conclude that in applications with a huge number of points and large radii (e.g. clustering problems easily lead to such cases) model HMCLP^{BIPS} may become intractable. Still concerning this model, we note that the CPU time required to solve it seems quite indifferent to the number of continuous facilities to locate.

We turn now our attention to the B&C algorithm devised for model (HMCLP^{IP}). From tables (4.2) and (4.3) we see that the model becomes more challenging when the number of continuous facilities to locate increase. Nevertheless, in those cases in which a proven optimal solution could not be found within the time limit imposed, the final gap is quite small. Still concerning the B&C procedure we observe that the pre-processing time as well as that for generating violated cuts and for the callback checks are all very low. This makes the whole algorithm more efficient. The number of continuous facilities to locate is clearly the factor influencing the most the performance of the algorithm since the number of incorporate cuts increases significantly. Still, we see that the B&C algorithm outperforms the plain use of model HMCLP^{BIPS} when the number of continuous facilities to locate is small (one or two). Nevertheless, we must recall that the comparison that can be observed in both tables can be made only because we are working on the Euclidean plane and using the Euclidean distance. As discussed above, if this was not the case, then we would certainly need to resort to the B&C algorithm devised since it is quite insensitive to the adopted norm, which is far from the case when we need to determine BIPS. The results show that the B&C approach is certainly a viable algorithm, which is quite general.

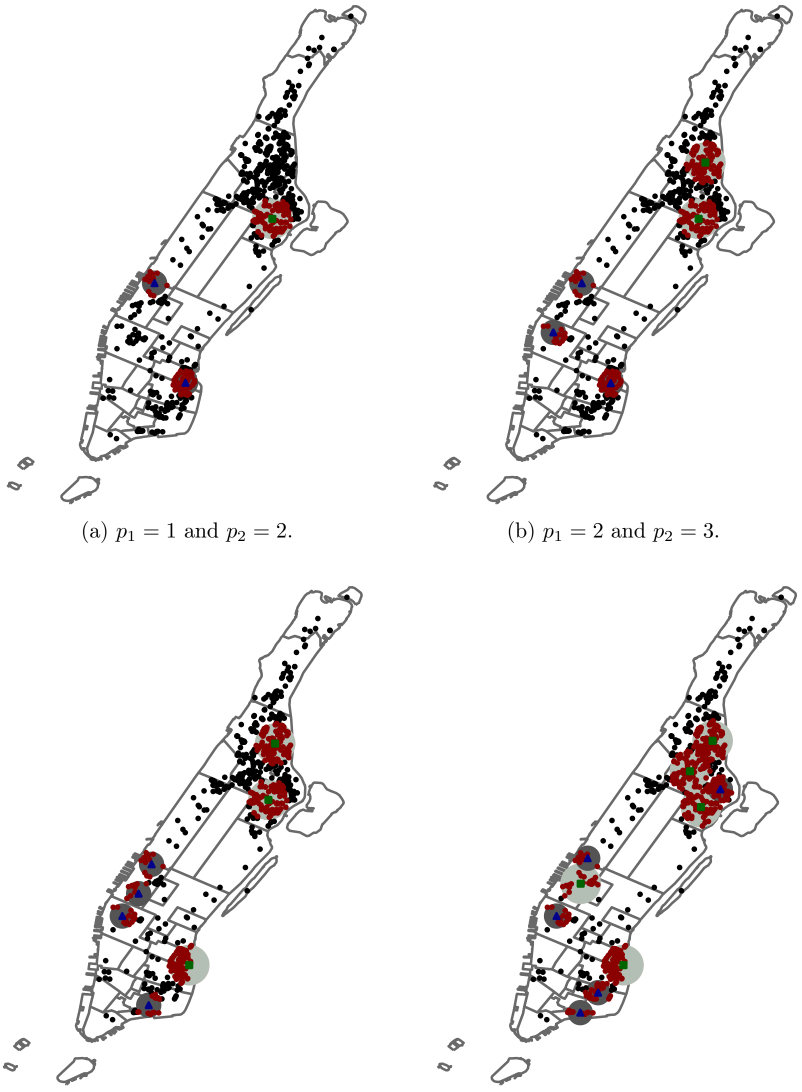
		CPUTime (secs.)												
		Total		Solving		Prepr.		Ctrs.	Gen.	Callback	MIPGAP	#Unsolved	#Ctrs	
n	p_2	BIPS	B&C	BIPS	B&C	BIPS	B&C	BIPS	B&C	B&C	B&C	B&C	BIPS	B&C
	1	53.45	6.24	0.27	3.75	7.10	1.38	46.09	1.10	0	0	0	402	72587
	2	53.28	51.91	0.23	49.41	7.10	1.36	45.96	1.10	0.04	0	0	402	144373
400	3	53.46	229.64	0.24	227.00	7.10	1.37	46.12	1.12	0.15	0	0	402	216159
	4	53.43	732.37	0.24	729.61	7.10	1.36	46.08	1.09	0.31	0	0	402	287945
	5	53.01	1516.04	0.23	1512.93	7.10	1.37	45.68	1.09	0.65	0	0	402	359731
	1	103.46	10.21	0.47	5.98	14.56	1.86	88.43	2.38	0	0	0	502	113418
	2	103.23	87.44	0.42	83.17	14.56	1.85	88.25	2.37	0.05	0	0	502	225835
500	3	102.28	473.58	0.46	469.20	14.56	1.85	87.26	2.32	0.21	0	0	502	338252
	4	102.60	1326.84	0.52	1322.02	14.56	1.85	87.51	2.34	0.63	0	0	502	450669
	5	103.73	2589.89	0.50	2584.69	14.56	1.85	88.67	2.34	1.65	0.01	5	502	563086
	1	318.71	56.62	1.60	50.02	59.45	3.23	257.66	3.36	0.01	0	0	702	219694
	2	318.78	391.64	1.52	384.99	59.45	3.23	257.81	3.34	0.07	0	0	702	437987
700	3	321.05	1631.95	1.57	1624.82	59.45	3.24	260.03	3.34	0.81	0.01	1	702	656280
	4	320.33	3165.19	1.57	3158.33	59.45	3.23	259.31	3.38	1.56	0.12	9	702	874573
	5	319.01	TL	1.57	TL	59.45	3.25	258.00	3.34	0.03	28.21	10	702	1092866
	1	937.45	548.75	14.23	535.54	216.81	5.05	706.40	8.01	0.15	0	0	922	370949
	2	927.92	2809.78	13.98	2795.74	216.81	5.06	697.13	7.99	1.03	0.12	2	922	740057
920	3	927.02	TL	14.26	TL	216.81	5.06	695.94	8.16	0.18	0.94	10	922	1109165
	4	928.00	TL	14.01	TL	216.81	5.04	697.18	8.07	0.08	26.68	10	922	1478273
	5	932.15	TL	13.64	TL	216.81	5.05	701.70	7.93	0.01	38.09	10	922	1847381

Table 4.2: Performance of the computational experiments for $\rho(2) = 0.005$

		CPUTime (secs.)												
		Total		Solving		Prepr.		Ctrs.	Gen.	Callback	MIPGAP	#Unsolved	#Ctrs	
n	p_2	BIPS	B&C	BIPS	B&C	BIPS	B&C	BIPS	B&C	B&C	B&C	B&C	BIPS	B&C
	1	130.38	11.92	3.05	9.90	24.49	0.89	102.84	1.11	0.02	0	0	402	62344
	2	130.23	109.60	2.59	107.50	24.49	0.89	103.14	1.16	0.05	0	0	402	123887
400	3	129.79	691.50	2.64	689.38	24.49	0.89	102.67	1.10	0.13	0	0	402	185430
	4	130.17	2252.92	2.87	2250.76	24.49	0.89	102.80	1.11	0.22	0.01	4	402	246973
	5	130.23	3299.38	2.98	3297.28	24.49	0.93	102.76	1.22	0.22	0.45	9	402	308516
	1	288.18	22.46	9.13	18.71	66.61	1.35	212.44	2.36	0.04	0	0	502	95286
	2	289.67	427.25	9.93	423.48	66.61	1.34	213.14	2.34	0.09	0	0	502	189571
500	3	289.07	1467.77	9.70	1463.84	66.61	1.34	212.76	2.34	0.26	0.01	1	502	283856
	4	289.85	2614.43	9.89	2610.61	66.61	1.34	213.36	2.34	0.24	0.42	9	502	378141
	5	289.29	TL	9.92	TL	66.61	1.34	212.76	2.36	0.03	28.37	10	502	472426
	1	1001.96	67.20	40.67	61.42	325.38	2.35	635.91	3.36	0.08	0	0	702	180619
	2	1000.09	1519.78	41.36	1513.80	325.38	2.35	633.35	3.36	0.28	0	1	702	359837
700	3	999.35	3289.82	40.67	3284.10	325.38	2.35	633.30	3.35	0.16	4.14	9	702	539055
	4	1000.46	TL	41.06	TL	325.38	2.35	634.02	3.33	0.01	30.58	10	702	718273
	5	999.80	TL	42.02	TL	325.38	2.35	632.40	3.35	0.01	36.89	10	702	897491
	1	3253.39	471.14	141.97	458.74	1401.62	3.84	1709.81	7.94	0.63	0	0	922	292544
	2	3270.67	3320.87	141.63	3303.30	1401.62	3.84	1727.42	8.07	0.97	5.50	9	922	583247
920	3	3267.82	TL	142.66	TL	1401.62	3.83	1723.55	7.95	0.01	22.84	10	922	873950
	4	3268.61	TL	145.35	TL	1401.62	3.85	1721.64	8.07	0.01	29.14	10	922	1164653
	5	3258.10	TL	147.56	TL	1401.62	3.83	1708.92	7.93	0.01	35.09	10	922	1455356

Table 4.3: Performance of the computational experiments for $\rho(2) = 0.01$

In Figure 4.5, we show the solutions of four instances of our testbed obtained with the BIPS approach painted on the Manhattan map. We draw the solutions for the larger instances ($n = 920$ demand points), radii $\rho(1) = 0.008$ and $\rho(2) = 0.005$ and different values of p_1 and p_2. In the figure, red dots represent the covered demand nodes, green squares the positions of the discrete facilities, and blue triangles the positions of continuous facilities; the coverage areas for the discrete facilities are drawn in light green color and those of the continuous facilities are colored in gray. The percentages of covered demand for these instances ranges between 29% (Figure 4.5a) and 75% (Figure 4.5d).



(a) $p_1 = 1$ and $p_2 = 2$.

(b) $p_1 = 2$ and $p_2 = 3$.

(c) $p_1 = 3$ and $p_2 = 4$.

(d) $p_1 = 5$ and $p_2 = 5$.

Figure 4.5: Solutions of some instances of our testbed for $n = 920$.

4.6 Hybridized maximal covering location problem under uncertainty

One of the possible future research lines is considering a first-stage set of facilities located in a discrete setting and a second-stage in a continuous one. Additionally we assume that the exact number of second-stage facilities to locate is not known beforehand—it corresponds to information to be disclosed in future moment in time, for instance, depending on a budget whose volume is still unknown. We consider both the cases in which uncertainty is not quantified using probabilities (a finite set of scenarios is assumed for the future) and that in which the random variable representing the number of future facilities to install is described by some known probability distribution.

Several authors have investigated uncertainty in the context of maximal covering location problems. Drezner and Goldstein (2010) analyze gradual covering location problems when the coverage radii are random variables. Several authors have also analyzed covering location models in case the capability of a center to cover a user is uncertain. In this case, the notion of congestion plays an important role (see Batta et al. 1989; de Assis Corrêa et al. 2009; Daskin 1983; ReVelle and Hogan 1989, and Vohra and Hall 1993). Possible disruptions in facilities or link have also been considered in the context of the MCLP (Berman et al. 2009).

Uncertainty has also been investigated in other covering problems such as the set covering location problem. This the case in the work by Marín et al. (2018) who consider a multi-period stochastic set covering location problem. The sources if uncertainty include (i) the coverage radius of the facilities (translated into the possibility of having some location covering some demand node) (ii) the minimum number of facilities required to cover each demand point, (iii) the marginal benefit for covering each demand point and (iv) the marginal penalty for a shortage of facilities in the coverage of each demand point.

As mentioned above, in this section we consider the hybridized discrete-continuous MCLP which it was explained above. For the sake of understanding, the problem will be designated in this section by the (p_1, p_2)-Maximal Covering Location Problem ((p_1, p_2)-MCLP, for short). Note that the p_2 continuous facilities can be seen as a set of servers to be located in the future and that must accommodate the first p_1 discrete positions.

Given that the continuous facilities will be located in the future, it may easily happen that their exact number is uncertain when a decision is made for p_1 initial facilities. For this reason, in the (p_1, p_2)-MCLP we assume that p_1 is a deterministic parameter known beforehand whereas p_2 is uncertain. A finite set of possibilities (scenarios) is assumed for it.

This research line can also be looked to a certain extent, a variation of that studied by Berman and Drezner (2008). These authors investigated a p-median problem under uncertainty consisting of locating p initial facilities plus an uncertain number of extra additional ones. They also assume that the extra facilities are located at sites that are optimal given the location of the original p facilities. In both cases, the potential facilities to open belong to a finite set with coincides with the demand points. Therefore the problem is cast as a stochastic discrete p-median problem.

The remainder of the section is organized as follows. In the following Section 4.6.1

we discuss the hybridized discrete-continuous MCLP under uncertainty— no probability is assumed for the future scenarios. Different attitudes of the decision maker towards risk are investigated. Finally, in Section 4.6.2 we assume uncertainty quantified by a probability law and, again we discuss several ways for hedging against uncertainty depending on the attitude of the decision maker towards risk.

4.6.1 Robust optimization models capturing uncertainty

Hereafter we assume $\mathbb{S}^{(2)} = \mathbb{R}^d$. Furthermore, we focus on a setting in which we do not know beforehand the number of additional facilities that can be located in the future. Instead, this number is represented by a parameter, say ω, taking values in a finite set $\Omega = \{0, 1, \ldots, q\}$. A value $\omega = 0$ indicates that no additional facility can be located in the future.

Our problem still consists of selecting p_1 facilities in the finite set $\mathbb{S}^{(1)}$ to open here-and-now while accounting for the possibility of locating in the future an uncertain number (ω) of additional facilities in $\mathbb{R}^d$. We still seek to maximize the weighted coverage of the nodes in N although the weighted coverage induced by a here-and-now solution is now itself uncertain.

In what follows, we discuss different modeling frameworks for capturing uncertainty in ω. The variables x and y introduced in the previous section are still useful. However, we now need to index both the z- and the X-variables in ω since they now depend on the exact number of future facilities to locate that can vary:

$$z_{ik}^{\omega} = \begin{cases} 1, & \text{if node } a_i \text{ is covered by the } k\text{-th additional facility when } \omega \text{ additional} \\ & \text{facilities can be installed,} \\ 0, & \text{otherwise,} \end{cases} \quad \text{for}$$

all $i \in N, \omega \in \Omega \setminus \{0\}$, and $k \in \{1, \ldots, \omega\}$.

$X_k^{\omega} \in \mathbb{S}^{(2)}$: coordinates of the k-th future facility to locate when ω such facilities can be opened, $k \in \{1, \ldots, \omega\}$.

These variables can be looked at as a scenario-indexed representation of the z- and X-variables.

If we knew the exact occurring scenario ω then the problem to solve would be (p_1, p_2)-MCLP with $p_2 = \omega$, i.e., we would need to solve (p_1, ω)-MCLP. Let us denote by $\mathcal{V}(p_1, \omega)$ the optimal value of this problem.

The following result holds which is a direct consequence of the maximal covering objective and the fact that by increasing the number of the future facilities we can open, the optimal value cannot deteriorate.

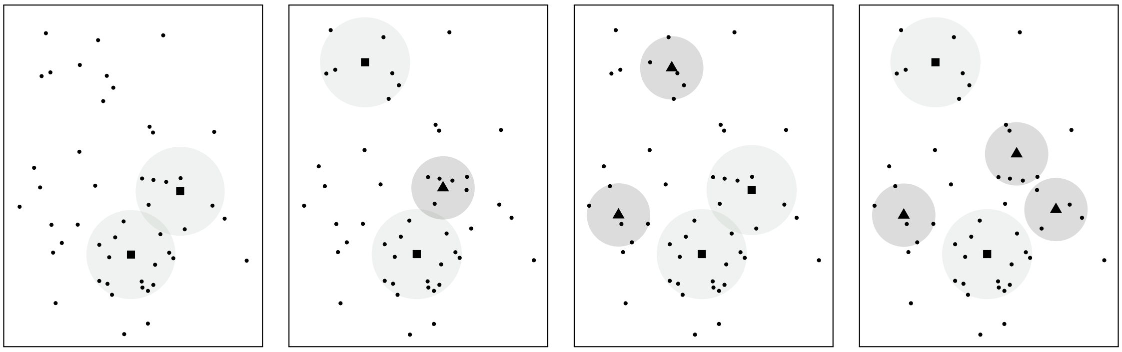
Proposition 4. *The following sorting holds:* $\mathcal{V}(p_1, 0) \leq \mathcal{V}(p_1, 1) \leq \cdots \leq \mathcal{V}(p_1, q)$.

Proof. The result follows directly from the fact that for every $\omega', \omega'' \in \Omega$ and $\omega' < \omega''$, an optimal solution to (p_1, ω')-MCLP is also feasible to (p_1, ω'')-MCLP with the same objective function value in both cases. □

The sorting stated in the previous result is illustrated in example 5.

Example 5. *We take the same set $\mathcal{A}$ introduced in Example 3. This set contains 50 nodes that were randomly generated in $[0,1] \times [0,1]$. We take $\mathbb{S}^{(1)} = \mathcal{A}$, $\mathbb{S}^{(2)} = \mathbb{R}^2$, and $\| \cdot \|^{(1)} = \| \cdot \|^{(2)} = \ell_2$, i.e., the Euclidean norm. For the coverage radii we take $\rho(1) = 0.2$ and $\rho(2) = \rho(3) = 0.1$. The number of facilities to open in $\mathbb{S}^{(1)}$ is fixed to 2. The weights c_i are all equal to one. Finally, we assume $\Omega = \{0, 1, 2, 3\}$.*

In Figure 4.6 we represent the optimal solution to models ($(p_1 = 2, \omega)$-MCLP), $\omega \in \Omega$.



(a) ((2, 0)-MCLP). (b) ((2, 1)-MCLP). (c) ((2, 2)-MCLP). (d) ((2, 3)-MCLP).

Figure 4.6: Optimal solution for different scenarios.

In Figures 4.6a and 4.6c we find the same initial facilities being selected. Nevertheless, in the former, we observe a 30% coverage whereas this number raises to 50% in the latter. In Figures 4.6b and 4.6d we observe again the same initial facilities being chosen but the coverage percentage is 58% for the first and 68% for the second.

Robust worst-case model

Since the number of future facilities to locate in uncertain, one possibility for making a here-and-now decision is to plan for the worst-case scenario. Given Proposition 4.6.1 such scenario corresponds to setting $\omega = 0$ and thus the problem resorts to solving (p_1, p_2)-MCLP with $p_2 = 0$. In turn, this is a classical discrete maximal covering location that can be formulated as:

$$\max \sum_{i \in N} c_i x_i, \tag{UMCLP$_1^{\text{WC}}$}$$

$$\text{s.t. (HMCLP}_2^{IP}\text{), (HMCLP}_3^{IP}\text{),}$$

$$x_i \in \{0,1\}, \; \forall i \in N, \tag{UMCLP$_2^{\text{WC}}$}$$

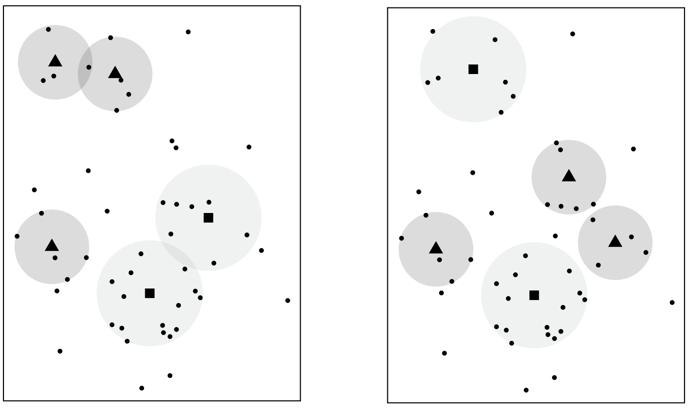
$$y_j \in \{0,1\}, \; \forall j \in M. \tag{UMCLP$_3^{\text{WC}}$}$$

Once this problem is solved the p facilities to install here-and-now are found. When the number of additional facilities to locate becomes eventually known, their locations are accommodated to the above initial p_1 facilities. This is a conservative approach that may be quite unfavorable from a covering perspective since the initial p_1 facilities may not provide room for adequately (i.e., in an advantageous way) locating the additional facilities. Example 6 illustrates this perspective.

Example 6. *We consider the instance from the previous example assuming again* $p_1 = 2$. *We start by solving the problem for the worst-case scenario. This leads to the location of the two discrete facilities represented by black squares in Figure 4.7a. In this case, a 42% coverage level is achieved by the discrete facilities (to locate here-and-now).*

Suppose that later we are informed about the possibility of locating three additional continuous facilities. In this case, we must condition the selection of these new facilities to the previously located discrete ones. The resulting solution can be seen in Figure 4.7a and includes the facilities represented by the black triangles. In this case a global 72% coverage level is attained.

However, if we knew beforehand that three additional facilities could be located in the future, then the best solution is the one depicted in Figure 4.7b. In this case, a global coverage level of 74% is achieved with the discrete facilities being responsible by 38%.



(a) (UMCLP^{WC}) (b) ((p_1, 3)-MCLP)

Figure 4.7: The conservatism of the solution for the worst-case scenario.

Planning for the worst-case scenario means to adopt a complete risk aversion attitude towards risk ad thus, somehow ignore how the future outcomes may positively influence a here-and-now solution.

Min-Max regret model

An alternative to the above extreme perspective consists of finding a here-and-now solution (the discrete facilities to locate) minimizing the maximum regret across all scenarios.

For any scenario $\omega \in \Omega$ we define a Regret function of the sets of initial and future facilities located, $\mathcal{X}^{(1)} \subseteq \mathbb{S}^{(1)}$ with $|\mathcal{X}^{(1)}| = p_1$ and $\mathcal{X}^{(2)}_\omega$ with $|\mathcal{X}^{(2)}_\omega| = \omega$:

$$\text{Regret}_\omega(\mathcal{X}^{(1)}; \mathcal{X}^{(2)}_\omega) = \mathcal{V}(p_1, \omega) - \sum_{i \in \mathcal{C}(\mathcal{X}^{(1)}) \cup \mathcal{C}(\mathcal{X}^{(2)}_\omega)} c_i.$$

The min-max regret problem can now be formulated as follows:

$$\min_{\substack{\mathcal{X}^{(1)} \subseteq \mathbb{S}^{(1)}: |\mathcal{X}^{(1)}| = p_1 \\ \mathcal{X}^{(2)}_\omega \subset \mathbb{R}^d: |\mathcal{X}^{(2)}_\omega| = \omega,\ \omega \in \Omega}} \max_{\omega \in \Omega} \text{Regret}_\omega(\mathcal{X}^{(1)}; \mathcal{X}^{(2)}_\omega).$$

Making use of the sets of variables already introduced this problem can be formulated as follows:

$$\min \max_{\omega \in \Omega} \left(\mathcal{V}(p_1, \omega) - \sum_{i \in N} c_i \left[x_i + \sum_{k=1}^{\omega} z^\omega_{ik} \right] \right) \tag{UMCLP$^{\text{R}}_1$}$$

s.t. (HMCLP$_2^{IP}$), (HMCLP$_3^{IP}$),

$$x_i + \sum_{k=1}^{\omega} z^\omega_{ik},\ \forall i \in N,\ \omega \in \Omega \setminus \{0\}, \tag{UMCLP$^{\text{R}}_2$}$$

$$||a_i - X^\omega_k|| \le \rho(2) + M(1 - z^\omega_{ik}),\ \forall i \in N,\ \omega \in \Omega \setminus \{0\},\ \forall k \in \{1, \ldots, \omega\}, \tag{UMCLP$^{\text{R}}_3$}$$

$$x_i \in \{0, 1\},\ \forall i \in N, \tag{UMCLP$^{\text{R}}_4$}$$

$$y_j \in \{0, 1\},\ \forall j \in M, \tag{UMCLP$^{\text{R}}_5$}$$

$$z^\omega_{ik} \in \{0, 1\},\ \forall i \in N,\ \forall \omega \in \Omega \setminus \{0\},\ \forall k \in \{1, \ldots, \omega\}, \tag{UMCLP$^{\text{R}}_6$}$$

$$X^\omega_k \in \mathbb{R}^d,\ \forall \omega \in \Omega \setminus \{0\},\ \forall k \in \{1, \ldots, \omega\}. \tag{UMCLP$^{\text{R}}_7$}$$

We can easily linearize this function by means of an auxiliary variable $\nu \ge 0$ representing the maximum regret to be minimized and imposing

$$\mathcal{V}(p, \omega) - \sum_{i \in N} c_i \left[x_i + \sum_{k=1}^{\omega} z^\omega_{ik} \right] \le \nu,\ \forall \omega \in \Omega. \tag{UMCLP$^{\text{R}}_{1'}$}$$

For each scenario $\omega \in \Omega$, the left-hand-side of the corresponding above constraint represents the regret of the solution in the scenario, i.e., the difference between the best weighted coverage we can achieve in that scenario and the actual weighted coverage (under this scenario) of the solution we are seeking.

Note that constraints (UMCLP$^{\text{R}}_3$) can be rewritten as explained in Section 4.3.2 leading to an alternative model.

Remark 6. *The above model allows finding a solution minimizing the maximum regret. This means that for the scenarios whose regret is not the maximum, every solution in the z-variables that leads to a different regret bellow the maximum provides an alternative* *optimal. To ensure the best coverage for each scenario (apart from the one corresponding to the maximum regret) we must fix the solution $\mathcal{X}^{(1)}$ found by the overall model and then solve the induced (continuous) MCLP maximizing the coverage of the nodes still uncovered by $\mathcal{X}^{(1)}$.*

Proposition 5. *Consider an optimal solution, say $\{\hat{\mathbf{x}}, \hat{\mathbf{X}}, \hat{\mathbf{y}}, \hat{\mathbf{z}}\}$, to model (UMCLP$^{\mathrm{R}}$). For each $\omega \in \Omega \setminus \{0\}$ denote $\gamma(\omega) = \sum_{i\in N} c_i \left[\hat{x}_i + \sum_{k=1}^{\omega} \hat{z}_{ik}^{\omega}\right]$ i.e., $\gamma(\omega)$ the weighted coverage of our solution in case scenario ω occurs. Then, we have that $\gamma(1) \leq \gamma(2) \cdots \leq \gamma(q)$.*

Proof. If for some $1 < \omega < q$ we have $\gamma(\omega) > \gamma(\omega - 1)$ then we can replace the values of variables z_{ik}^{ω} $(i \in I,\, k = 1, \dots, \omega - 1)$ by those of $z_{ik}^{\omega-1}$ $(i \in I,\, k = 1, \dots, \omega - 1)$ and setting $z_{i\omega}^{\omega} = 0$, $i \in I$. This change sets $\gamma(\omega)$ equal to $\gamma(\omega - 1)$, i.e., recovers the sorting stated in the proposition without deteriorating the total weighted coverage in scenario ω. □

4.6.2 Stochastic optimization models

Expected coverage model

Assume now that the number of facilities we can install in the future is a discrete random variable, $\boldsymbol{\xi}$. We directly assume that $\boldsymbol{\xi}$ has finite support, $\Omega = \{0, 1, \dots, q\}$, since other case would not make sense in our problem. Furthermore we denote $\pi_\omega = \mathbb{P}[\boldsymbol{\xi} = \omega]$, $\forall \omega \in \Omega$. Considering this probability distribution, other possibilities emerge for modeling the problem. To start with, we can consider a risk-neutral decision maker and plan for maximizing the expected weighted coverage of the nodes in $\mathcal{A}$, i.e., solving:

$$\max_{\mathcal{X}^{(1)} \subseteq \mathbb{S}^{(1)} \,:\, |\mathcal{X}^{(1)}| = p_1} \mathbb{E}_{\boldsymbol{\xi}} \left[\max_{\mathcal{X}_{\xi}^{(2)} \subset \mathbb{R}^d : |\mathcal{X}_{\xi}^{(2)}| = \xi} \sum_{i \in \mathcal{C}(\mathcal{X}^{(1)}) \cup \mathcal{C}(\mathcal{X}_{\xi}^{(2)})} c_i \right].$$

The extensive form of the deterministic equivalent can also be derived as the follow model that we call (UMCLP$^{\mathrm{EC}}$):

$$\begin{aligned} \max\ & \pi_0 \sum_{i\in N} c_i x_i + \sum_{\omega \in \Omega\setminus\{0\}} \pi_\omega \sum_{i \in N} c_i \sum_{k=1}^{\omega} z_{ik}^{\omega}, && (\mathrm{UMCLP}_1^{\mathrm{EC}}) \\ \text{s.t. } & (\mathrm{HMCLP}_2^{IP}), (\mathrm{HMCLP}_3^{IP}), \\ & (\mathrm{UMCLP}_2^{\mathrm{R}}) - (\mathrm{UMCLP}_7^{\mathrm{R}}). \end{aligned}$$

Expected coverage models with regret thresholds

The expected coverage model described above seeks to find a compromise (using the underlying probability distribution) for the coverage across the different scenarios. To enforce finding solutions that do not deviate too much from the ideal coverage for each scenario, the model can be enriched by imposing a constraint on the maximum regret across all scenarios (Snyder and Daskin 2006). Given a threshold, $\eta \in [0,1]$, this condition can be written as follows:

$$\mathcal{V}(p_1,\omega) - \sum_{i\in N} c_i \left[x_i + \sum_{k=1}^{\omega} z_{ik}^{\omega} \right] \leq \eta \mathcal{V}(p_1,\omega),\ \omega \in \Omega, \qquad (\text{UMCLP}^{\eta\text{T}})$$

where η indicates the maximum percentage deviation of the solution for each scenario ω, with respect to the ideal value $\mathcal{V}(p_1,\omega)$.

The best outcome would be to have all regrets as close to zero as possible. Unfortunately, due to the different future observations it is not possible to ensure that. Hence, the choice of η must result from a trade-off between the desire of the decision maker and what is actually feasible in each specific case. In fact, we expect that for every instance, there will be a lower threshold for η below which the no feasible solution can be found.

α-reliable Min-Max regret and regret thresholds

In some cases, both the min-max regret approach and the expected coverage with regret thresholds might be very restrictive since they measure the regret (either in the objective function or as threshold-type constraints) with respect to all the scenarios in Ω. Such requirement can be relaxed using the notion of α-reliability sets introduced by Daskin et al. (1997).

Definition 1 (α-reliability set)**.** *Given $\alpha \in [0,1]$, an α-reliability set, $\Omega^\alpha \subseteq \Omega$, is a subset of scenarios with joint probability at least α, i.e.,*

$$\mathbb{P}[\boldsymbol{\xi} \in \Omega^\alpha] \geq \alpha.$$

Using this concept, we can now seek to minimize the maximum regret only across the scenarios in some reliability set Ω^α. This ensures achieving a maximum regret with probability at least α. Note that all subsets of Ω define a 0-reliability sets.

An adequate model in this case is obtained from the min-max regret model above presented replacing ($\text{UMCLP}^{\text{R}}_{1'}$) with

$$\mathcal{V}(p_1,\omega) - \sum_{i\in N} c_i \left[x_i + \sum_{k=1}^{\omega} z_{ik}^{\omega} \right] \leq \nu,\ \omega \in \Omega^\alpha. \qquad (\text{UMCLP}^{\alpha\text{R}})$$

Considering a particular instance of the problem and assuming some value for α, several α-reliability sets may be available. This raises a query about which one to consider. In such a case we can plan for finding the best solution across all α-reliability sets. To accomplish this, the set of scenarios selected to take part of the reliability set Ω^α should be part of the decision to make. It can be determined by incorporating to the model the following binary variables indicating whether some scenario ω belongs or not to the selected Ω^α:

$$v_\omega = \begin{cases} 1, & \text{if } \omega \in \Omega^\alpha, \\ 0, & \text{otherwise,} \end{cases}$$

for all $\omega \in \Omega$. The following sets of constraints ensure that (UMCLP$^{\alpha\mathrm{R}}$) holds:

$$\sum_{\omega\in\Omega} \pi_\omega v_\omega \geq \alpha, \tag{UMCLP$_1^{\alpha\mathrm{R}}$}$$

$$\mathcal{V}(p,\omega) - \sum_{i\in N} c_i \left[x_i + \sum_{k=1}^{\omega} z_{ik}^\omega \right] \leq \nu + \overline{M}_\omega (1 - v_\omega),\ \omega \in \Omega,, \tag{UMCLP$_2^{\alpha\mathrm{R}}$}$$

$$v_\omega \in \{0,1\}, \forall \omega \in \Omega. \tag{UMCLP$_3^{\alpha\mathrm{R}}$}$$

For $\omega \in \Omega$, $\overline{M}_\omega$ is a large enough constant. The interested reader can refer to Daskin et al. (1997) for tighter choices concerning these bounds. Constraint(UMCLP$_1^{\alpha\mathrm{R}}$) ensures that the constructed set $\Omega^\alpha = \{\omega \in \Omega : v_\omega = 1\}$ is an α-reliability set. Constraint (UMCLP$_2^{\alpha\mathrm{R}}$) allows to activate constraint (UMCLP$^{\alpha\mathrm{R}}$) when $\omega \in \Omega^\alpha$.

The expected coverage model with regret thresholds can also be adapted to α-reliability sets by replacing constraints (UMCLP$^{\eta\mathrm{T}}$) by:

$$\mathcal{V}(p_1,\omega) - \sum_{i\in N} c_i \left[x_i + \sum_{k=1}^{\omega} z_{ik}^\omega \right] \leq \eta \mathcal{V}(p_1,\omega),\ \omega \in \Omega^\alpha. \tag{UMCLP$^{\alpha\mathrm{R}-\eta\mathrm{T}}$}$$

α-CVaR models

A risk-averse decision maker tends to focus the attention on less favorable scenarios since these are the most worrisome. The extreme case consists of planning for the worst scenario. Nevertheless, this may be too much conservative even because the worst scenario may hold with a negligible probability to justify having it influencing too much the adopted solution. Between the extreme perspective and a risk-neutral attitude towards risk, there is a good range of alternatives.

One possibility is to minimize a weighted sum of (i) the maximum regret across the reliability sets for some confidence and (ii) the regret for the other scenarios—optimization of the α-reliable mean-excess regret (see Chen et al. 2006). In fact, this corresponds to considering a risk-averse decision maker and optimizing the α conditional value at risk (α-CVaR) considering the regret of a solution as the loss function. This can be accomplished by minimizing a convex combination of the maximum regret with respect to the scenarios in an α-reliability set Ω^α and the expected regret with respect to the scenarios in $\Omega \backslash \Omega^\alpha$.

Denote $\lambda_{\Omega^\alpha} = \frac{\mathbb{P}[\boldsymbol{\xi}\in\Omega^\alpha]-\alpha}{1-\alpha}$. The problem can be stated as follows:

$$\min_{\substack{\mathcal{X}^{(1)}\subseteq\mathbb{S}^{(1)}:|\mathcal{X}^{(1)}|=p_1\\ \mathcal{X}^{(2)}_\omega\subset\mathbb{R}^d:|\mathcal{X}^{(2)}_\omega|=\omega,\forall\omega\in\Omega\\ \Omega^\alpha\subseteq\Omega}} \lambda_{\Omega_\alpha}\max_{\omega\in\Omega^\alpha}\text{Regret}_\omega(\mathcal{X}^{(1)},\mathcal{X}^{(2)}_\omega)+(1-\lambda_{\Omega_\alpha})\frac{\displaystyle\sum_{\omega\in\Omega\setminus\Omega^\alpha}\pi_\omega\text{Regret}_\omega(\mathcal{X}^{(1)},\mathcal{X}^{(2)}_\omega)}{\displaystyle\sum_{\omega\in\Omega\setminus\Omega^\alpha}\pi_\omega}.$$

Following Chen et al. (2006) and Rockafellar and Uryasev (2000, 2002), the problem can be equivalently rewritten as

$$\begin{aligned}\min_{\substack{\mathcal{X}^{(1)}\subseteq\mathbb{S}^{(1)}:|\mathcal{X}^{(1)}|=p_1\\ \mathcal{X}^{(2)}_\omega\subset\mathbb{R}^d:|\mathcal{X}^{(2)}_\omega|=\omega,\forall\omega\in\Omega\\ \Omega^\alpha\subseteq\Omega}} \quad & \zeta+\frac{1}{1-\alpha}\sum_{\omega\in\Omega}\pi_\omega\left[\text{Regret}_\omega(\mathcal{X}^{(1)},\mathcal{X}^{(2)}_\omega)-\zeta\right]^+,\\ \text{s.t.}\quad & \zeta\geq 0,\end{aligned}$$

where $[\Delta]^+ = \max\{0,\Delta\}$, and ζ is the Value-at-Risk (VaR) of confidence α, i.e., the α-quantile of the regret distribution. The second term of the objective function aims at minimizing the expected regret over the $(1-\alpha)\times 100\%$ worst scenarios, i.e., the $(1-\alpha)\times 100\%$ with a regret larger than or equal to the α-VaR.

In terms of our decision variables it is equivalent to:

$$\begin{aligned}&\min \zeta+\frac{1}{1-\alpha}\sum_{\omega\in\Omega}\pi_\omega\left[\left(\mathcal{V}(p_1,\omega)-\sum_{i\in N}c_i\left[x_i+\sum_{k=1}^{\omega}z^\omega_{ik}\right]\right)-\zeta\right]^+,\\ &\text{s.t. } (\text{HMCLP}^{IP}_2), (\text{HMCLP}^{IP}_3),\\ &\qquad (\text{UMCLP}^{\text{R}}_2)-(\text{UMCLP}^{\text{R}}_7).\end{aligned}$$

The model can be linearized leading to

$$\min \zeta+\frac{1}{1-\alpha}\sum_{\omega\in\Omega}\pi_\omega\varrho_\omega, \tag{UMCLP$_1^{\alpha-\text{CVaR}}$}$$

$$\text{s.t. } \varrho_\omega\geq\left(\mathcal{V}(p_1,\omega)-\sum_{i\in N}c_i\left[x_i+\sum_{k=1}^{\omega}z^\omega_{ik}\right]\right)-\zeta,\ \omega\in\Omega, \tag{UMCLP$_2^{\alpha-\text{CVaR}}$}$$

$$\varrho_\omega\geq 0,\ \omega\in\Omega, \tag{UMCLP$_3^{\alpha-\text{CVaR}}$}$$

$$(\text{HMCLP}^{IP}_2), (\text{HMCLP}^{IP}_3),$$
$$(\text{UMCLP}^{\text{R}}_2)-(\text{UMCLP}^{\text{R}}_7).$$

If we set $\alpha=0$ then all scenarios have a regret above the VaR and thus we obtain a model seeking to minimize the overall mean regret:

$$\min \sum_{\omega\in\Omega} \pi_\omega \left(\mathcal{V}(p,\omega) - \sum_{i\in N} c_i \left[x_i + \sum_{k=1}^{\omega} z_{ik}^{\omega} \right] \right), \tag{4.12}$$

$$\text{s.t. } (\text{HMCLP}_2^{IP}), (\text{HMCLP}_3^{IP}),$$
$$(\text{UMCLP}_2^{\text{R}}) - (\text{UMCLP}_7^{\text{R}}).$$

Given that for every problem instance, $\sum_{\omega\in\Omega} \pi_\omega \mathcal{V}(p_1,\omega)$ is a constant, it is trivial to conclude that minimizing the mean expected coverage regret (4.12) is equivalent to maximizing the expected weighted coverage ($\text{UMCLP}_1^{\text{EC}}$).

4.7 Conclusions

In this chapter we investigated the multitype maximal covering facility location problem. A general modeling framework was discussed, which was adapted to an hybridized discrete-continuous facility location problem. In this case we could go deeper in our analysis. For the particular case in which the space underlying the continuous location problem is the Euclidean space and when the euclidean norm is used a third model could be proposed. The results highlighted the viability of a branch-and-cut algorithm for dealing with the problem in its general form. In particular, instances with up to 920 demand nodes and two types of facilities (discrete and continuous) could be solved rather efficiently. This defines a new state-of-the-art in terms of maximal covering location problems with a large potential number of locations for the discrete facilities.

The work done encourages some other research lines. These include more work on the development of valid inequalities for the general integer linear programming model thus leading to an even better polyhedral description of the feasibility set, the inclusion of time as a decision dimension, and the inclusion of uncertainty either in the demand or in the number of facilities that can be open. This last one was presented in Section 4.6 where models to capture uncertainty were shown depending of the risk aversion by the decision maker.

Chapter 5

Fairness in maximal covering location problems

This chapter provides a general mathematical programming based framework to incorporate fairness measures from the facilities' perspective to Discrete and Continuous Maximal Covering Location Problems. The main ingredients to construct a function measuring fairness in this problem are the use of: (1) ordered weighted averaging operators, a family of aggregation criteria very popular to solve multiobjective combinatorial optimization problems, and (2) α-fairness operators which allow to generalize most of the equity measures. A general mathematical programming model is derived which captures the notion of fairness in maximal covering location problems. The models are firstly formulated as Mixed Integer Non-Linear programming problems for both the discrete and the continuous frameworks. Suitable Mixed Integer Second Order Cone programming reformulations are derived using geometric properties of the problem. Finally, the chapter concludes with the results obtained on an extensive battery of computational experiments. The obtained results support the convenience of the proposed approach.

5.1 Introduction

Section 1.4 introduces the notion of *fairness*, shows the vast literature on this term and tons of applications in different fields. We recall that this term is defined as "the quality of treating people equally or in a way that is right or reasonable". It is an abstract but widely studied concept in Decision Sciences in which some type of indivisible resources are to be shared among different agents.

Fair allocations should imply impartiality, justice and equity in the allocation patterns, which are usually quantified by means of inequality measures that are minimized. In Section 1.4 we show several proposed measures such as the max-min, the minimum envy, the Jain's index, α-fairness schemes, or certain families of ordered weighted averaging criteria.

In this chapter, we analyze a novel version of one of the core family of problems in Facility Location, Covering Location (CL) problems which we introduced them in Section 1.2. The location of the facilities in CL is characterized by the fact that the facilities are allowed to give service to the users at a limited distance from them. In particular, we answer the question of how to incorporate fairness in the Maximal Covering Location problem (MCLP). The MCLP was introduced in Section 1.2.2. We recall that in this problem, it is assumed the existence of a budget for opening facilities and the goal is to accommodate it to satisfy as much demand as possible. As usual in location problems, one can consider different frameworks based on the nature of the solution space for the facilities: in a discrete or continuous spaces. While the discrete setting is more adequate when locating physical services, (as ATMs, stores, hospitals, etc), the continuous framework is known to be more adequate to determine the positions of routers, alarms or sensors, that can be more flexibly positioned. The continuous framework is also useful to determine the set of potential facilities that serves as input for a discrete version of the problem. One of the main difference between these two families of problems (from the mathematical optimization viewpoint) is that in the discrete one the distances between the facilities and the users are given as input data (or can be preprocessed before solving the problem), while in the continuous case, the distances are part of the decision and they must incorporated to the optimization problem.

As far as we know, the incorporation of fairness measures into CL problems has been studied in Drezner and Drezner (2014), in which the max-min approach has been applied to the gradual maximal covering location problem. There, the authors incorporate the worst-case fairness criteria from the user's perspective, i.e. in order to enforce equity between the partial coverage of all users. In networks, Rahmattalabi et al. (2020) consider the selection of a subset of nodes to cover their adjacent nodes with fairness constraints with applications to social networks. Asudeh et al. (2020) analyze a covering location problem with fairness constraints minimizing the pairwise deviations between the different covered sets. Korani and Sahraeian (2013) study a hub covering problem with *equity* allocation constraints.

Despite of these applications, the efficiency measure used in the MCLP is the overall covered demand, that is, as much covered demand the better. However, when one looks at the individual utilities of each of the constructed facilities, one may obtain solutions with highly saturated facilities in contrast to others that only cover a small amount of demand, which results in unfair systems from the facilities' perspective. Moreover, in many situations this type of unfair solutions are also undesirable from the users' viewpoint which may see reduced the quality of the required service, as in the location of telecommunication servers which have a higher probability to fail in case of being saturated (being other capable to give service to these users) or in the student assignment process to schools, in which a higher number of alumni allocated to a school may deteriorate the education system. As far as we know, this problem has not been previously investigated in the literature in the context of Covering Location.

In this chapter, we provide a flexible mathematical programming based framework to incorporate fairness measures from the facilities' perspective to Discrete and Continuous MCLPs. This generalization of the fairness measure for the MCLP is based on adequately combining the two main tools mentioned : Ordered Weighted Averaging (OWA) operators in Section 1.3, and (α-fairness) operators in Section 1.4.

Our specific contributions in this chapter are:

1. To define a novel fairness measure combining OWA and (α-fairness) operators that can be incorporated to the objective function of the MCLP.

2. To describe a general mathematical programming model which captures the notion of fairness in MCLPs.

3. To provide a Mixed Integer Non-Linear programming formulation for the two main frameworks in the facility location: discrete and continuous spaces.

4. To derive MISOCO reformulations for the problem, suitable to be solved with off-the-shelf optimization softwares.

5. To test on a battery of computational experiments the performance of the formulations and their managerial insights.

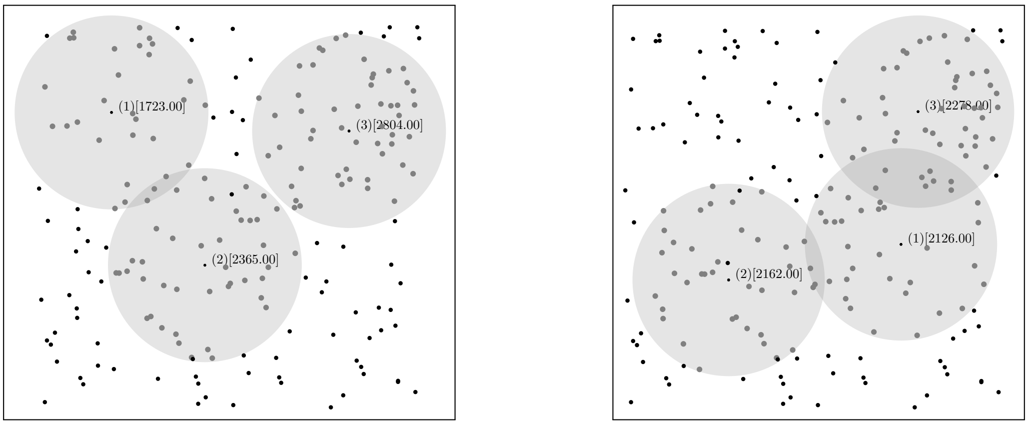
The remainder of the chapter is organized as follows. In Section 5.2 we introduce the generalized fair maximal covering problem. In Section 5.3 we present a mathematical programming formulation for the problem, both in the discrete and the continuous framework. The results of our computational experience are reported in Section 5.4. Finally, the chapter ends with some conclusions and future research lines.

5.2 The generalized Fair Maximal Covering Location Problem

As already mentioned, the Maximal Covering Location Problem (MCLP), in its different versions, can be seen as a resource allocation problem, in which the overall demand of the *covered* users is shared among the different services that are located. Thus, a high coverage of the total demand (no matter which service is providing the coverage) is appropriate from a global perspective, but from an individual viewpoint, one can easily get unfair allocation patterns. Furthermore, the MCLP usually exhibits multiple optimal solutions, that is, different subsets of p services covering the maximum possible covered demand, and then, optimization solvers output an arbitrary one, possibly not the fairest.

In the following example we illustrate this situation in a toy instance.

Example 7. *We consider a randomly generated set of* 200 *demand points with coordinates in* $[0, 20] \times [0, 20]$. *We assign to each of them a random integer weight in* $[0, 100]$. *We assume that three services are to be located chosen from the set of demand points. The coverage areas for all the potential location of the services are disks of radius* 5. *The solution obtained by Gurobi for the classical MCLP is shown in Figure 5.1a. In such a solution,* 65.7% *of the demand is covered, and the distribution of (weighted) users among the services is* $(1723, 2365, 2804)$, *that is, there are two services covering close to one thousand more clients than the other one. Other feasible (non optimal for the MCLP) solution for the problem is also show in Figure 5.1b, in which* 62.58% *of the demand is covered and whose distribution of covered demand is* $(2126, 2162, 2278)$. *This solution, although covers* 3% *less demand than the classical MCLP, is clearly much more equitable than the MCLP solution, since all the users cover approximately the same demand, but still efficient.*



(a) Classical MCLP (b) Fairer solution for MCLP

Figure 5.1: Optimal solutions obtained with the MCLP

In what follows we introduce a new fairness measure that we incorporate into the MCLP in order to provide fair coverage of the demands. This new measure is a generalization of the fairness measure based on adequately combining two main tools mentioned in Sections 1.3 and 1.4: OWA operators and (α-fairness) operators.

Definition 2. *Let $\alpha \geq 0$ and $\lambda \in \mathbb{R}^p_+$ with $\sum_{j=1}^p \lambda_j = 1$ and $\lambda_1 \geq \cdots \geq \lambda_p$. The (α, λ)-fair operator is a function $F_{\alpha,\lambda} : \mathbb{R}^p_+ \to \mathbb{R}^+$ defined as:*

$$F_{\alpha,\lambda}(W_1, \ldots, W_p) := \begin{cases} \frac{1}{1-\alpha} \displaystyle\sum_{j=1}^{p} \lambda_j W_{(j)}^{1-\alpha} & \text{if } \alpha \neq 1, \\ \displaystyle\sum_{j=1}^{p} \lambda_j \log(W_{(j)}) & \text{if } \alpha = 1. \end{cases} \qquad ((\alpha, \lambda)\text{-Fair})$$

Theorem 12. *$F_{\alpha,\lambda}$ is concave.*

Proof. Let $W_1, \ldots, W_p \in \mathbb{R}_+$. Assume that $W_1 \leq \cdots \leq W_p$. If $\alpha = 1$, since the log function is monotone, we get that $Z_1 := \log(W_1) \leq \cdots \leq Z_p := \log(W_p)$. Thus, $F_{1,\lambda}(W_1, \ldots, W_p) = \Phi_\lambda(Z_1, \ldots, Z_p)$ which is concave. In case $\alpha < 1$, defining $\widehat{\lambda}_j = \frac{1}{1-\alpha}\lambda_j$ and $Z_j = W_j^{1-\alpha}$ for all $j = 1, \ldots, p$, we get that $Z_1 \leq \cdots \leq Z_p$ and $F_{\alpha,\lambda}(W_1, \ldots, W_p) = \Phi_{\widehat{\lambda}}(Z_1, \ldots, Z_p)$, being then $F_{\alpha,\lambda}$ concave. Finally, in case $\alpha > 1$, we define $\widehat{\lambda}_j = -\frac{1}{1-\alpha}\lambda_j$ and $Z_j = -W_j^{\alpha-1}$ for all $j = 1, \ldots, p$, we get that $Z_1 \leq \cdots \leq Z_p$ and $F_{\alpha,\lambda}(W_1, \ldots, W_p) = \Phi_{\widehat{\lambda}}(Z_1, \ldots, Z_p)$. Thus, $F_{\alpha,\lambda}$ is concave. □

Proposition 6. *Let $\mathbf{W} = (W_1, \ldots, W_p) \in \mathbb{R}^d$. $F_{\alpha,\lambda}$ verifies the following properties:*

1. **Continuity:** *$F_{\alpha,\lambda}$ is a continuous function. This axiom assures that, locally, small changes in the allocation do not significantly affect the measure.*

2. **Population size independence**: *Equal resource allocations are, eventually, independent of the number of users, i.e., $\lim_{p\to\infty} \frac{F_{\alpha,\lambda}(\mathbf{1}_{p+1})}{F_{\alpha,\lambda}(\mathbf{1}_p)} = 1$.*

3. **Pareto optimality:** *If* $W_j \leq \bar{W}_j, \forall j \in \{1, \ldots, p\}$, *and* $W_j < \bar{W}_j$ *for at least some* j, *then* $F_{\alpha,\lambda}(\mathbf{W}) \leq F_{\alpha,\lambda}(\bar{\mathbf{W}})$.

4. **Symmetry:** $F_{\alpha,\lambda}(W_1, \ldots, W_p) = F_{\alpha,\lambda}(W_{\sigma(1)}, \ldots, W_{\sigma(p)})$, *where* σ *is an arbitrary permutation of the indices.*

5. **Bounded:** *The value of allocation given by the scheme is bounded.*

6. **Scale and metric independence:** *The measure is independent of scale, i.e, the unit of measurement does not matter.*

Proof. Some of this properties are stated by Lan et al. (2010) as axioms and others are from the literature (Jain et al., 1984; Barbati and Piccolo, 2016), and its proof for $F_{\alpha,\lambda}$ is straightforward. □

The ((α, λ)-Fair) operator depends of $p+1$ parameters (α and the λ-weights). Observe that this operator is a combination of the (α-fairness) measure introduced by Atkinson (1970) and the fair OWA operators introduced in Yager (1988). Actually, in case $\alpha = 0$, our operator turns into the λ-OWA operator. In case $\lambda = (\frac{1}{p}, \ldots, \frac{1}{p})$ it becomes the $\frac{1}{p}$-weighted (α-fairness) function $\frac{1}{p}\Psi_\alpha$. This combination of these two operators allows us to derive a unified framework to deal with fairness in maximal covering location problems, that we detail below.

We are given a set of demand points in a d-dimensional space, $\mathcal{A} = \{a_1, \ldots, a_n\} \subseteq \mathbb{R}^d$, indexed by the set $N = \{1, \ldots, n\}$, and each of the points endowed with a demand weight $\omega_i \geq 0$ for all $i \in N$. We are also given a metric space $\mathbb{S} \subseteq \mathbb{R}^d$ endowed with a distance measure $\|\cdot\|$, and a radius $R \in \mathbb{R}$ for each $X \in \mathbb{S}$, which is assumed to be the same for all the facilities to be located (although it is not a limitation of the results provided in this chapter). The goal of the Fair MCLP is to find the position of the p facilities to locate, $X_1, \ldots, X_p \in \mathbb{S}$, maximizing the ($(\alpha, \lambda)$-Fair) operator of the demands covered by these services. We denote by P the index set for the facilities, i.e., $P = \{1, \ldots, p\}$.

Formally, given λ and α, the (α, λ)-Fair Maximal Covering Location Problem ((α, λ)-FMCLP, for short) can be model as the following optimization problem:

$$F^*_{\alpha,\lambda} = \max_{X_1,\ldots,X_p \in \mathbb{S}} F_{\alpha,\lambda}(W(X_1), \ldots, W(X_p)) \qquad (\text{FMCLP}_{\alpha,\lambda})$$

where $W(X_j)$ is the covered demand of facility X_j, assumed that each demand point is accounted as covered by at most one of the facilities. We denote by $\mathcal{W}_{\alpha;\lambda} = (W(X_1^*), \ldots, W(X_p^*)) \in \mathbb{R}^p_+$ a coverage vector of each of the facilities in the problem above. We also denote by $\mathcal{W}^{\text{sum}}_{\alpha;\lambda} = \sum_{j=1}^p W(X_p^*)$ the total covered demand in the solution and by $\mathcal{W}^{\min}_{\alpha;\lambda} = \min_{j=1,\ldots,p} W(X_p^*)$ the demand covered by the service with smallest coverage in the solution.

The problem that we introduce above allows one to determine the position of the facilities that are fair from the individual viewpoint, but, how much is one willing to lose when imposing fairness to the MCLP? *The price of fairness* in any allocation rule is a notion studied in Bertsimas et al. (2011) in order to measure the efficiency loss under a fair allocation compared to the one that maximizes the overall sum of the users utilities. In our case, the solution of ($\text{FMCLP}_{\alpha,\lambda}$) is compared against the solution of the classical MCLP in order to know how far is a fair solution to the best coverage of the give demand.

Definition 3. *The price of fairness measure for* ($\text{FMCLP}_{\alpha,\lambda}$) *is defined as the index:*

$$\text{PoF}(\text{FMCLP}_{\alpha,\lambda}) = \frac{\mathcal{W}^{\text{sum}}_{0;(\frac{1}{p},\ldots,\frac{1}{p})} - \mathcal{W}^{\text{sum}}_{\alpha;\lambda}}{\mathcal{W}^{\text{sum}}_{0;(\frac{1}{p},\ldots,\frac{1}{p})}}, \tag{PoF}$$

The price of fairness indicates the relative deviation of the covered demand when solving the maximal (α, λ)-FMCLP with respect to the solution of the classical MCLP which attains the maximal possible coverage. Thus, the price of fairness is a value between 0 and 1 measuring how close is the effectiveness of the obtained fair solution with respect to the most effective covering solution. A price of fairness equal to 0 indicates that ($\text{FMCLP}_{\alpha,\lambda}$) is able to construct a fair allocation without loss of efficiency (at the maximum possible coverage). In contrast, a price of fairness with value 1 means that ($\text{FMCLP}_{\alpha,\lambda}$) obtains the worst global coverage. Thus, as closer this measure to 0 the better. In general it provides the percent loss of coverage with respect to the maximal possible coverage of an instance, allowing one to quantify the price one has to pay when imposing (α, λ)-fairness.

On the other hand, one can also measure how far is a fair solution from the fairest share, which is obtained when solving the max-min covering location problem, i.e., comparing the demand covered (in our fair MCLP) by the service covering the smallest demand in the solution with respect to the solution in which the coverage of the service covering the least demand is maximized. This measure was called in Bertsimas et al. (2012) as the *price of efficiency*.

Definition 4. *The price of efficiency measure for* ($\text{FMCLP}_{\alpha,\lambda}$) *is defined as:*

$$\text{PoE}(\text{FMCLP}_{\alpha,\lambda}) = \frac{\mathcal{W}^{\min}_{0;(1,0,\ldots,0)} - \mathcal{W}^{\min}_{\alpha;\lambda}}{\mathcal{W}^{\min}_{0;(1,0,\ldots,0)}}, \tag{PoE}$$

The Price of Efficiency is interpreted as the percent loss in the minimum demand coverage guarantee compared to the maximum minimum covered demand guarantee. This index also takes value in $[0, 1]$, in which a value of 0 means that the (α, λ)-FMCLP obtains the fairest solution, while a value of 1 indicates the least fair solution in which there is a service not covering any demand.

Other widely used measure of fairness is the *envy*. In the MCLP the envy of facility positioned in $X_j \in \mathbb{S}$, whose covered demand is $W(X_j)$, for a facility located at $X_k \in \mathbb{S}$, whose covered demand is $W(X_k)$, is defined as:

$$\text{envy}(X_j, X_k) = \max\{0, W(X_k) - W(X_j)\}$$

that is, facility j suffers envy of value $W(X_k) - W(X_j)$ from facility k in case k covers more demand than j. The overall envy of a set of p facilities $X_1, \ldots, X_p \in \mathbb{S}$ is the overall sum of all the pairwise envies. From this, the *Gini index* is defined as the ratio of this total envy and the all the covered demand by the facilities multiplied by $2p$ (the overall number of pairwise comparisons):

Definition 5. *The Gini index is defined as:*

$$\text{Gini}(X_1, \ldots, X_p) = \frac{\displaystyle\sum_{j,k \in P} \text{envy}(X_j, X_k)}{2p \displaystyle\sum_{j \in P} W(X_j)} \tag{Gini}$$

We will see in our computational experience that the family of (α, λ)-FMCLP exhibits differences when varying the values of α and λ with respect to the three measures that we described above (PoF, PoE and Gini). A trade-off solution between these measures will be desirable from the point of view of efficiency and also of fairness.

5.3 Mathematical Programming Formulations for α-FOWA MCLP

In this section we derive a suitable mathematical programming-based framework to model the (α, λ)-FMCLP. We will present different formulations for the problem for both the discrete case ($\mathbb{S}$ being a finite pre-specified set) and the continuous case ($\mathbb{S} = \mathbb{R}^d$). The nature of the domain of this problem directly affect the development of resolution strategies for it.

A general formulation for the problem considers the following decision variables:

$$z_{ik} = \begin{cases} 1, & \text{if node } a_i \text{ is covered by the } k\text{-th selected facility in } \mathbb{S}, \\ 0, & \text{otherwise,} \end{cases} \qquad \text{for all } i \in N,\ k \in P,$$

and $X_k \in \mathbb{R}^d :$ coordinates of the k-th selected facility in $\mathbb{S}$.

(FMCLP$_{\alpha,\lambda}$) can be formulated as follows:

$$\max F_{\alpha,\lambda}\Big(\sum_{i\in N}\omega_i z_{i1},\ldots,\sum_{i\in N}\omega_i z_{ip}\Big)$$

$$\text{s.t.} \sum_{k\in P} z_{ik} \leq 1,\ \forall i \in N, \tag{FMCLP$_1$}$$

$$a_i \in \mathbb{B}_R(X_k) \text{ if } z_{ik} = 1,\ \forall i \in N, k \in P, \tag{FMCLP$_2$}$$

$$z_{ik} \in \{0,1\},\ \forall i \in N, k \in P,$$

$$X_k \in \mathbb{S},\ \forall k \in P.$$

The objective function aims to maximize the (α,λ)-fairness of the demand coverage by each of the facilities. Constraint (FMCLP$_1$) assures that each covered demand point is counted at most once as covered. (FMCLP$_2$) ensures the adequate definition of the x-variables.

The feasible set of the problem above will described then by a set of linear and second-order cone representable inequalities on binary and continuous variables. One of the main difficulties of the model above stems on the representation of the objective function $F_{\alpha,\lambda}$ which consists of the following two ingredients:

Sorting: Representing the order given by the OWA operator into an optimization problem is a difficult challenge. In Section 1.3 we describe the two most popular formulations for this operator on the values $W_k^{1-\alpha}$ for $k \in P$. A third representation is based on the x-variables that we consider in our problem, by sorting the selected facilities in $\mathbb{S}$ in non-increasing order of the demand coverage, i.e, enforcing the following constraints:

$$\sum_{i\in N}\omega_i z_{ik} \leq \sum_{i\in N}\omega_i z_{ik+1},\ \forall k \in P. \tag{Sorting}$$

$(1-\alpha)$**-powers:** Observe that the $(1-\alpha)$ powers of the coverage of each facility appear in the objective function. Denoting by $W_k = \sum_{i\in N}\omega_i z_{ik}$ and by $Z_k = W_k^{1-\alpha}$, for $k \in P$, assuming that $W_1 \leq \ldots \leq W_p$, the objective function above can be written as the linear function:

$$F_{\alpha,\lambda}(W_1,\ldots,W_p) = \frac{1}{1-\alpha}\sum_{k\in P}\lambda_k Z_k,$$

as long as it is fulfilled that $Z_k \leq W_k^{1-\alpha}$ (for $\alpha < 1$) or $Z_k \geq W_k^{1-\alpha}$ (for $\alpha > 1$) for all $k \in P$. Assuming that α is rational, we get that there exists $p, q \in \mathbb{Z}_+$ with $p \geq q \geq 1$ and $\gcd(p,q) = 1$ such that:

$$\frac{1}{1-\alpha} = \begin{cases} \frac{p}{q} & \text{if } \alpha < 1, \\ -\frac{q}{p} & \text{if } \alpha > 1. \end{cases}$$

Thus, the power-constraints can be rewritten as:

$$Z_k^p \leq W_k^q, \text{for all } k \in P. \tag{Powers}$$

These constraints can be conveniently rewritten as a set of quadratic second-order cone constraints following a simplification of the results in Blanco et al. (2014).

In the rest of the section we describe how to represent constraints (FMCLP_2) in a suitable mathematical programming formulation. This representation highly depends on the nature of the set of potential coordinates for the facilities $\mathbb{S}$. We analyze the cases in which $\mathbb{S}$ is a finite set and the one where $\mathbb{S} = \mathbb{R}^d$.

5.3.1 Continuous framework

We analyze here the case where the potential set from which the coordinates of the services are chosen is the entire space, i.e., $\mathbb{S} = \mathbb{R}^d$. In this case, the norm-based *covering constraints* (FMCLP_2) can be rewritten as

$$\|X_k - a_i\| \leq R + U_i(1 - z_{ik}), \forall i \in N, k \in P. \tag{Norms}$$

where U_i is a big enough constant ($U_i > \|a_i - a_{i'}\|$ for all $i' \in N$). It ensures that in case i is allocated to the kth selected facility (X_k), then a_i must belong to $\mathbb{B}_R(X_k)$.

These constraints have been rewritten in 1.1.1 as set of linear and second-order cone inequalities by (ℓ_τ-norm) for ℓ_τ-norms or by (Pol-norm) for polyhedral norms inducing mixed-integer second-order cone optimization problems.

As already observed in Chapters 3 and 4 the norm-based constraints (Norms) can also be rewritten as linear constraints using Lemma 1 defined in Chapter 1.2.3. This *linearization* is based on projecting out the X-variables by ensuring that these can be constructed from the x-variables. See Section 4.3.2 for more details.

5.3.2 Discrete framework

Let us assume that the potential set of facilities if finite, that is, $\mathbb{S} = \{b_1, \ldots, b_m\} \subseteq \mathbb{R}^d$. We denote by $M = \{1, \ldots, m\}$ its index set. The model in this case can be simplified taking into account that the subset of potential facilities that are able to cover each single demand point can be pre-computed. It allows also to avoid the use of the X-variables, replacing them by the following decision variables to decide which of the potential facilities from $\{b_1, \ldots, b_m\}$ are open and and which is the position of the demand covered by each facility in the ordered vector.

$$y_{jk} = \begin{cases} 1, & \text{if the covered demand of facility } j \text{ is the } k\text{-th largest,} \\ 0, & \text{otherwise,} \end{cases} \quad \text{for all } j \in M, k \in P.$$

Then, (FMCLP) can be alternatively formulated as:

$$\max \frac{1}{1-\alpha} \sum_{k \in P} \lambda_k Z_k$$

$$\text{s.t. } (\text{FMCLP}_1), (\text{Sorting}), (\text{Powers})$$

$$z_{ik} \leq \sum_{j \in \mathcal{C}(i)} y_{jk}, \ \forall i \in N, k \in P, \qquad (\text{FMCLP}_1^{\text{D}})$$

$$\sum_{j \in M} y_{jk} = 1, \ \forall k \in P, \qquad (\text{FMCLP}_2^{\text{D}})$$

$$\sum_{k \in P} y_{jk} \leq 1, \ \forall j \in M, \qquad (\text{FMCLP}_3^{\text{D}})$$

$$z_{ik} \in \{0,1\}, \ \forall i \in N, \forall k \in P,$$

$$y_{jk} \in \{0,1\}, \ \forall j \in M, k \in P.$$

Apart from (FMCLP_1), (Sorting), (Powers), the *covering* constraints (FMCLP_2) are rewritten using the y-variables using ($\text{FMCLP}_1^{\text{D}}$)–($\text{FMCLP}_3^{\text{D}}$). Constraints ($\text{FMCLP}_1^{\text{D}}$) assure that the demand points can be assigned to a facility if it is sorted in any position (equivalently, if it is open). Constraints ($\text{FMCLP}_2^{\text{D}}$) enforces that a single facility is assigned to a position and ($\text{FMCLP}_3^{\text{D}}$) that at most one position is assigned to a facility (those facilities not assigned to a positions will be not open). Both constraints together with (Sorting) assure that exactly p facilities are open, each of them in a different order in the coverage sequence.

The classical formulations for the MCLP use one-index binary variables to determine the *open* facilities as the presented in Section 1.2.2. Note that in our case it is not enough, since the positions of the activated services in the sorting coverage sequence are needed to allow the allocation of the demand points to a facility only in case it is open.

5.4 Computational study

In this section, we report on the results of our computational experience of the presented models in this chapter. We consider the data set given by Orlandini (2019) with size 181 which contains the locations (latitude and longitude) of Residential Schools and student hostels operated by the federal government in Canada. The coordinates in this data set have been normalized in the unit square. From the whole dataset we construct different instances with sizes, n, ranging in $\{45, 90, 120\}$ (the first n demand nodes in the complete instance). The users' demands have been randomly generated by a uniform distribution in $(0, 1)$. The number of facilities to be located, p, ranges in $\{5, 10, 15, 20\}$, and we use the same radius for all facilities and for all instances, $R = 1$.

We run our models by choosing the λ-parameters of the OWA operator in $\{\mathtt{W}, \mathtt{C}, \mathtt{K}, \mathtt{D}, \mathtt{G}, \mathtt{H}\}$

(see Table 1.3, where `C` - Minimum, `D` - $\alpha-$Min-Average, `G` - Gini, `H` - Harmonic, `K` - $k-$Average, and `W` - Average). The α-parameter was chosen in $\{0, 0.5, 1, 2\}$.

As already mentioned the fairest solution is obtained for the max-min approach (`C`). Therefore, since a single facility (the most damaged) is accounted in such an objective function, the α-parameter does not affect the solution. Thus, for `C` we only solved the $\alpha = 0$ case. With that, a total of 252 instances for each framework (a total of 504) have been solved in our computational study.

The models were coded in Python 3.7 in an iMac with 3.3GHz with an Intel Core i7 with 4 cores and 16GB 1867 MHz DDR3 RAM. We use Gurobi 9.1.2 as optimization solver. A time limit of 2 hour was fixed for all the instances.

The set of generated instances and the complete results obtained in our computational experiments are available in the github repository `github.com/vblancoOR/fowa`.

In the following tables of this section we show the computational performance of each model for each frameworks that we consider: discrete and continuous. The tables show the average computational times for those solved instances within the time limit (`Time`), MIP GAP (`GAP`) when the time limit is reached, the total of instances which have reached the time limit (#`TL`), and the total of instances that flagged "Out of Memory" when trying to solve the problem (#`OoM`).

In Table 5.1 we show the results for the discrete framework this information for aggregated instances by the values of λ (Table 5.1a) and α (Table 5.1b). We observe that the model is not able to solve within the time limit of two hours 78 out of the 252 instances, and 12 of them can not be solved because of Out of Memory. Besides, we observe that the most difficult λ vector to solve was `K` with higher computational times and number of unsolved instances. Concerning the α-parameter, the computationally hardest one was $\alpha = 1$, that is, when the model uses the log to solve the problem.

λ	Time	GAP	#TL	#OoM
C	2223.46	10.96%	5	0
D	2586.8	6.83%	15	2
G	3265.55	9.27%	19	2
H	2930.06	11.24%	22	2
K	3071.63	10.04%	18	4
W	1865.88	3.85%	7	2

(a) Aggregated by λ

α	Time	GAP	#TL	#OoM
0	2019.45	9.09%	21	0
0.5	3472.56	7.26%	19	0
1	1795.86	11.05%	26	6
2	3228.64	8.60%	20	6

(b) Aggregated by α

Table 5.1: Averaged time and GAP, and total non-solved instances aggregating by parameters λ and α for the discrete framework

Table 5.2 shows the results disaggregated by size of the demand set (n) and by the number of facilities to locate (p). The reader can observe that for $p = 5$ we were able to optimally solve all the instances and for $p = 10$ most of the instances were also optimally solved within the time limit. In contrast, for the largest values of p, the model were no longer capable of certifying optimallity, but it provides very small MIPGAPs.

n	p	Time	GAP	#TL	#NS
45	5	3.41	0.00%	0	0
	10	359.71	3.14%	4	0
	15	5471.43	11.50%	17	0
	20	3302.87	8.22%	7	10
90	5	164.53	0.00%	0	0
	10	5264.63	3.42%	1	0
	15	4366.36	3.49%	16	0
	20	3607.89	16.39%	5	2
120	5	406.44	0.00%	8	0
	10	5465.62	4.61%	20	0
	15	225.87	4.72%	20	0
	20	3617.06	8.41%	20	0

Table 5.2: Averaged time and GAP, and total non-solved instances disaggregated by size n and number of centers p for the discrete framework

For the continuous MCLP, we observed that the nonlinear formulation (Norms) have a worse performance than the linear formulations proposed in Chapter 3 and 4, as expected. Thus, we provide the results of using formulation (4.3) in our experiments.

In Table 5.3 we show the results for the continuous framework. As usual in Location problems, this framework was more challenging to solve than its discrete counterpart. A total of 124 instances were not able to be solved within the time limit and 6 instances output with an Out of Memory flag. However, when the time limit is reached, the average gap for these instances is not very high. In this framework, we obtain similar results for all the λ-values, not being clear which problem more time consuming. However, we observe that the Average (W) and the minimum (C) are slightly easier to solve having more solved instances within the time limit. On the other hand, when the model tries to solve the logarithm for the value of $\alpha = 1$, there is a greater number of unsolved instances and the averaged MIPGAP is greater.

λ	Time	GAP	#TL	#OoM
C	1778.93	1.81%	2	0
D	1566.85	7.84%	26	1
G	1447.96	9.97%	28	1
H	1080.51	11.04%	29	1
K	1985.79	8.63%	25	2
W	1861.95	3.59%	14	1

(a) Aggregated by λ

α	Time	GAP	#TL	#OoM
0	1199.21	9.98%	29	0
0.5	2174.11	5.27%	32	0
1	941.67	12.67%	34	5
2	2286.11	6.36%	29	1

(b) Aggregated by α

Table 5.3: Averaged time and GAP, and total non-solved instances aggregating by parameters λ and α for the continuous framework

We show in Table 5.4 the computation times by disaggregating the values of the size of the demand set, n, and the number of centers to locate, p, for those instances for which optimality was certified within time limit. In this table we can see that both the size of n and the size of p have an impact in the difficulty of solving the instances. One can observe that for the largest values of n, the model is only capable to certify optimality in two instances for each p within the time limit.

n	p	Time	GAP	#TL	#OoM
45	5	3.41	0.00%	0	0
	10	359.71	1.38%	1	0
	15	5471.43	7.98%	9	0
	20	3302.87	6.07%	13	5
90	5	164.53	0.00%	0	0
	10	5264.63	10.60%	12	0
	15	4366.36	6.80%	19	0
	20	3607.89	9.20%	16	0
120	5	406.44	0.00%	19	0
	10	5465.62	8.05%	19	1
	15	225.87	10.45%	19	0
	20	3617.06	9.61%	19	0

Table 5.4: Averaged time and GAP, and total non-solved instances disaggregated by size n and number of centers p for the continuous framework

In what follows we analyze the solutions obtained with our models in terms of fairness and efficiency. As noted above, the MIPGAPs for the instances unsolved within the time limit is small, being most of the instances assumed to be optimally solved (with certain degree of accuracy). In order to show the quality of the solutions, we consider those instances with a MIPGAP smaller than 5%.

In Figures 5.2 and 5.3 we draw values of the price of fairness (PoF) and the price of efficiency (PoE) disaggregated by the values of the parameter α and n. Clearly, the fairest solution is the one that provides the minimum ($\mathtt{C}$). However, when the parameter α increases, we obtain a range of fair solutions between the classical MCLP ($\alpha = 0,\ \lambda = \mathtt{W}$) and the minimum. This shows that the inclusion of both measures provides different solutions in terms of fairness. On the other hand, Figure 5.3 shows the averaged values of PoE. One can note that the highest PoE is obtained, as expected, when solving the classical version of the MCLP and the smallest, with the max-min approach. However, if we look at both figures, we can conclude that the trade-off between fairness and efficiency is promising when considering $\alpha-$Min-Average ($\mathtt{D}$) and Gini ($\mathtt{G}$), obtaining fairer solutions by increasing the value of α. If one is willing to lose more efficiency to gain fairness the better operator seems to be the Harmonic (H).

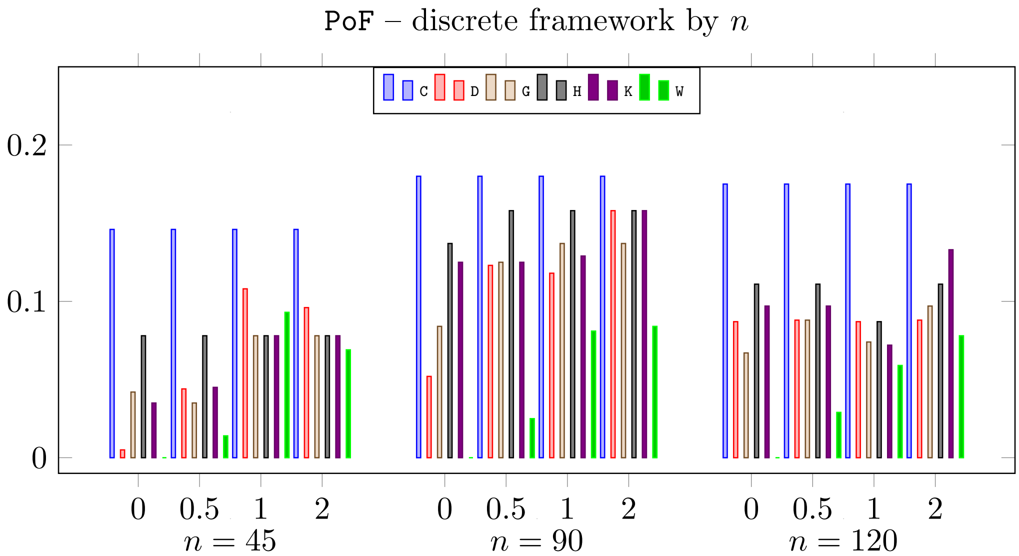


Figure 5.2: Price of fairness disaggregated by $\alpha \in \{0, 0.5, 1, 2\}$ and $n \in \{45, 90, 120\}$

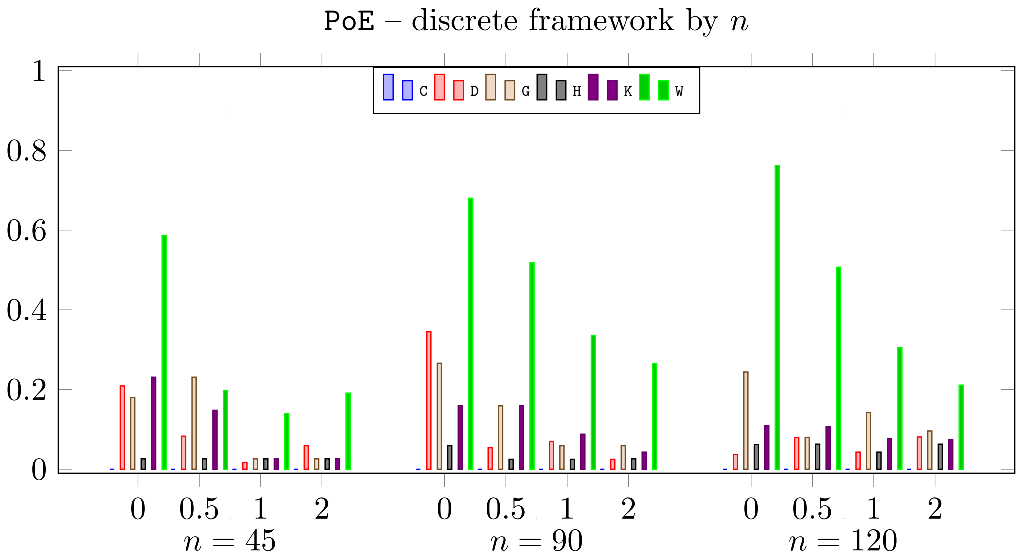


Figure 5.3: Price of efficiency disaggregated by $\alpha \in \{0, 0.5, 1, 2\}$ and $n \in \{45, 90, 120\}$

Disaggregating by p one can see, again, that the operator that gives good solutions of fairness and efficiency is the $\alpha-$Min-Average (D), and if we are willing to lose more efficiency to obtain better solutions in terms of fairness, a good choice is to use the Harmonic (H) OWA operator.

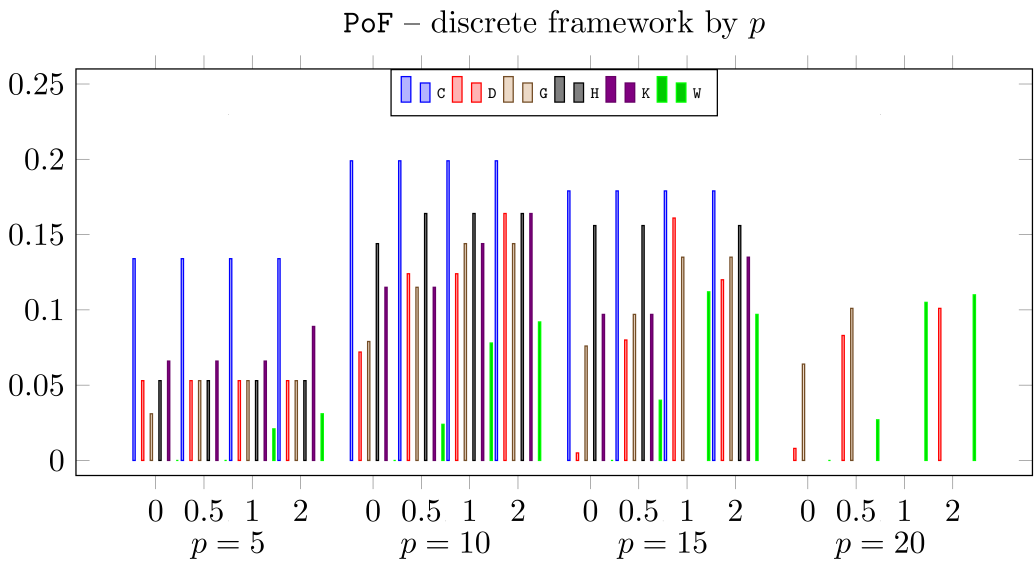


Figure 5.4: Price of fairness disaggregated by $\alpha \in \{0, 0.5, 1, 2\}$ and $p \in \{5, 10, 15, 20\}$

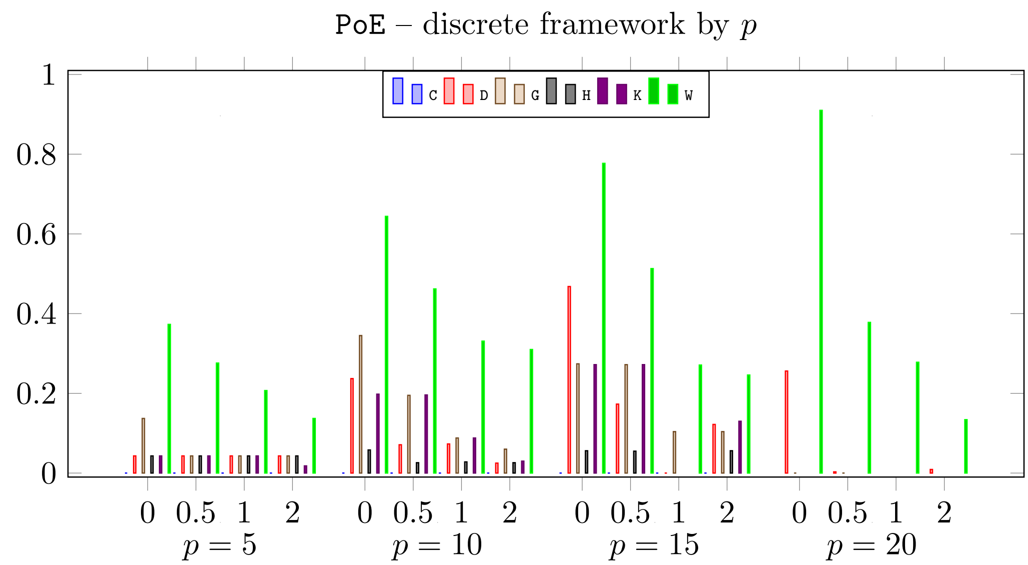
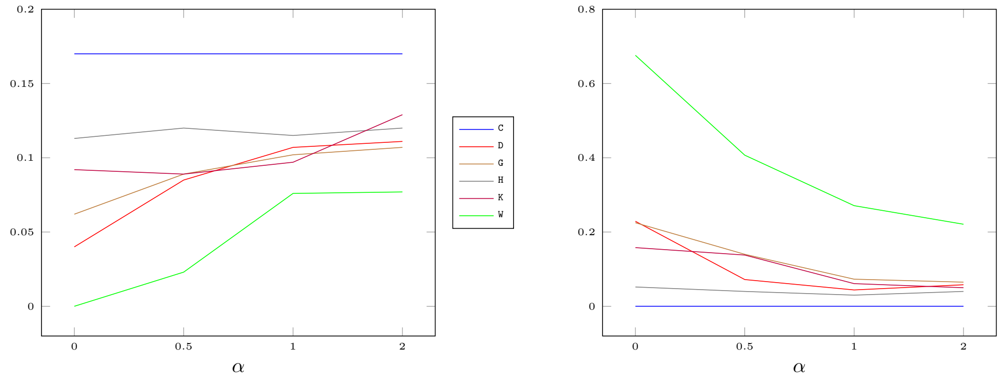


Figure 5.5: Price of efficiency disaggregated by $\alpha \in \{0, 0.5, 1, 2\}$ and $n \in \{5, 10, 15, 20\}$

Finally, if we observe the PoF and PoE values for the different values of the considered parameters, we can conclude that the harmonic operator returns the fairest solutions, that even not reaching the max-min fairness get balanced results in terms of efficiency and fairness. Other best choices to find a good trade off between fairness and efficiency are those provided by $\alpha-$Min-Average (D) and Gini (G).



(a) Price of fairness (b) Price of efficiency

Figure 5.6: Measures disaggregated by $\alpha \in \{0, 0.5, 1, 2\}$ and $p \in \{5, 10, 15, 20\}$

In the continuous counterpart, we obtained similar results. In figures 5.7 and 5.9 we observed that the level of fairness does not increase in some instances when α increase due to the average of the results with a 5% of GAP (not all the instances were solved up to optimallity).

When we disaggregate by n in figures 5.7 and 5.8, we see that the operators can be classified into three levels of fairness: on the one hand, the fairest, the minimum (C); on the other the least fair, the average (W); and lastly, a trade-off between fairness and efficiency for the rest of the operators in which we cannot conclude that one returns fairer results than another. Again, we observe that for the continuous case, the combination of both measures provides the decision maker with a range of fairness solutions different from those that the operators and α-fairness scheme would provide separately.

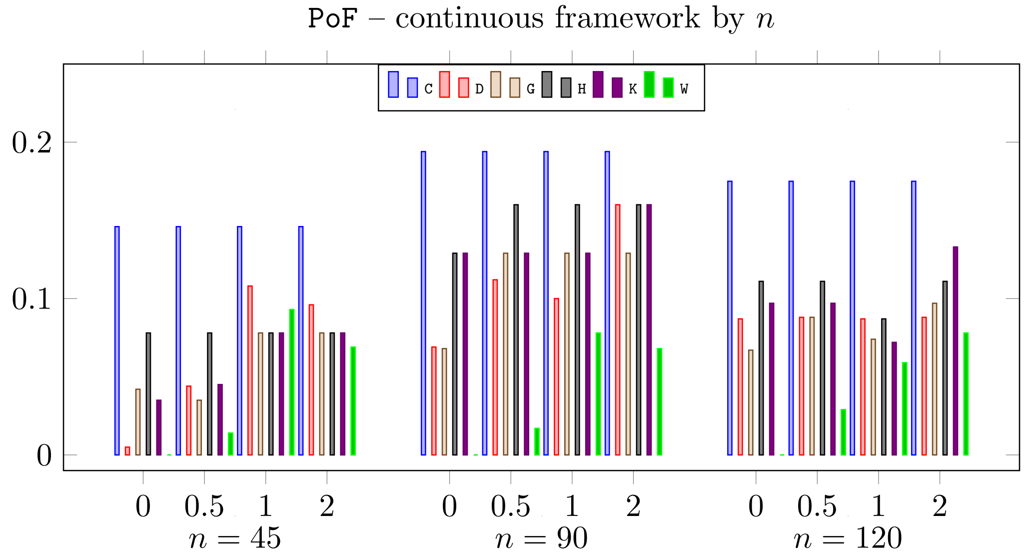


Figure 5.7: Price of fairness disaggregated by $\alpha \in \{0, 0.5, 1, 2\}$ and $n \in \{45, 90, 120\}$

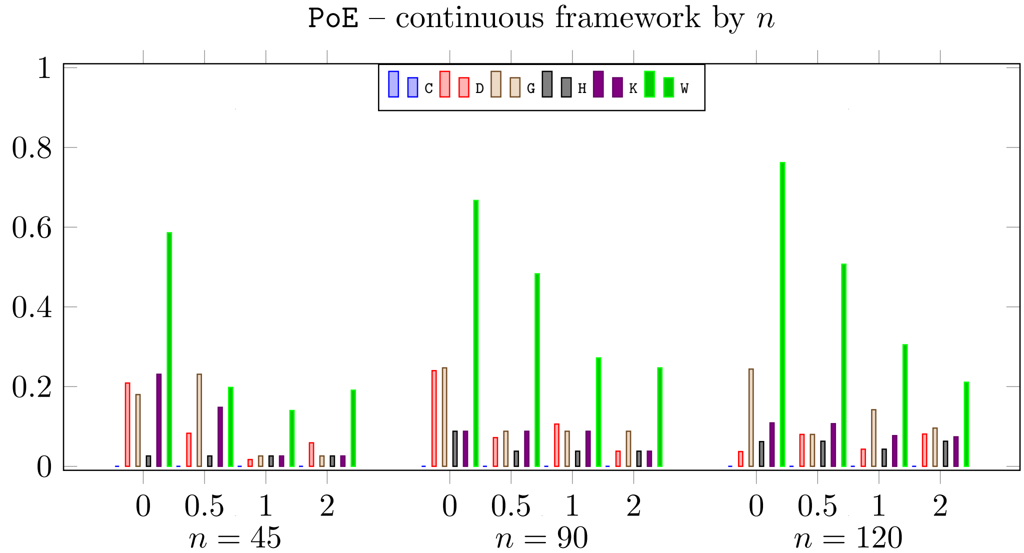


Figure 5.8: Price of efficiency disaggregated by $\alpha \in \{0, 0.5, 1, 2\}$ and $n \in \{45, 90, 120\}$

On the other hand, if we disaggregate by p, in the figures 5.9 and 5.10, we noted that the operators K and H give a good trade-off between fairness and efficiency.

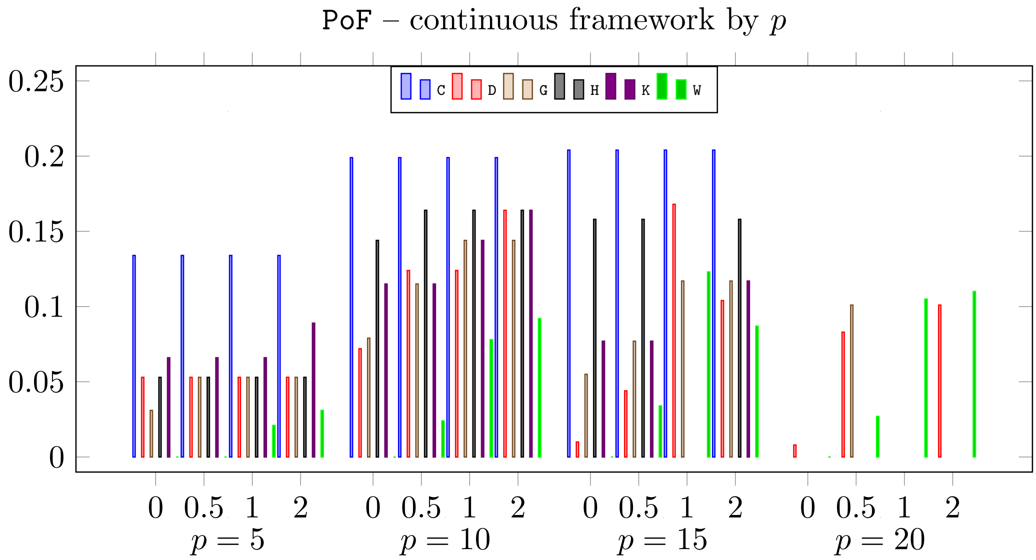


Figure 5.9: Price of fairness disaggregated by $\alpha \in \{0, 0.5, 1, 2\}$ and $p \in \{5, 10, 15, 20\}$

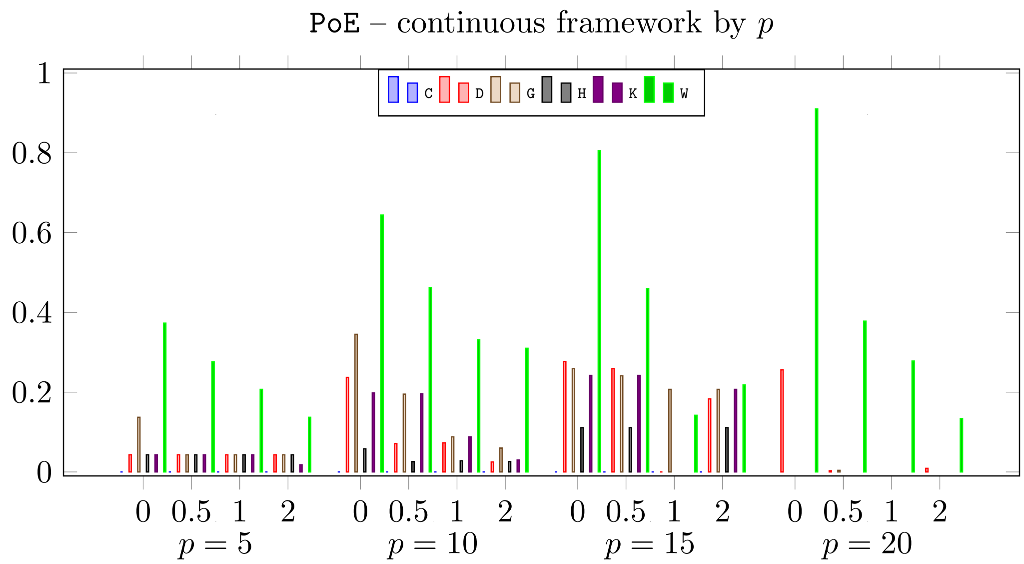
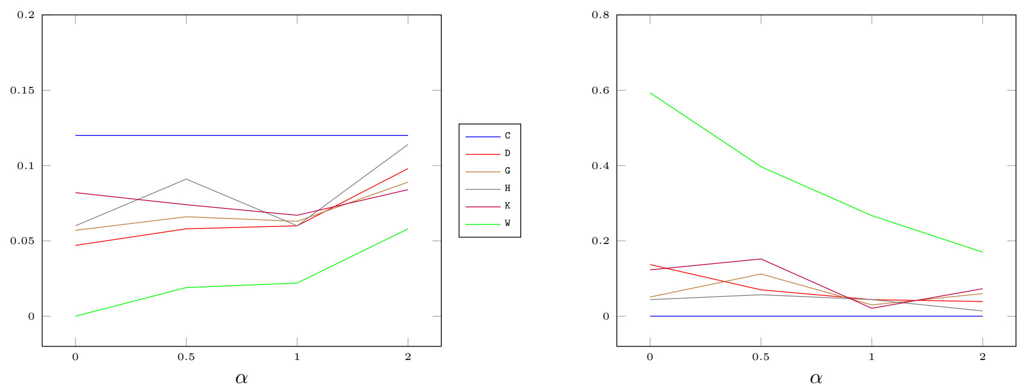


Figure 5.10: Price of efficiency disaggregated by $\alpha \in \{0, 0.5, 1, 2\}$ and $n \in \{5, 10, 15, 20\}$

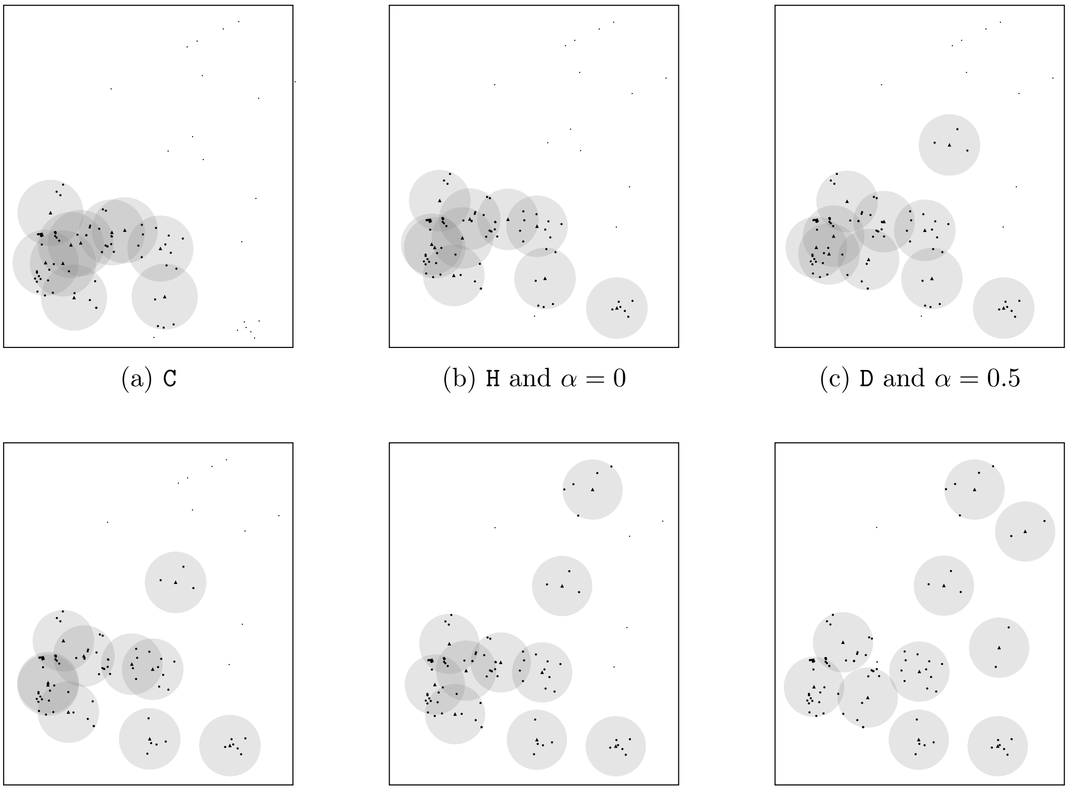
Disaggregating by the values of α we obtain a similar conclusion. The continuous case would return the three differentiated levels of fairness solutions discussed above. Being again as in the discrete case, the harmonic operator (H) the one that would gives us a fairest solution without reaching the one provided by the max-min approach.



(a) Price of fairness (b) Price of efficiency

Figure 5.11: Measures disaggregated by $\alpha \in \{0, 0.5, 1, 2\}$

Finally, in Figure 5.12 we draw the different solution obtained for a selected instance for different values of (λ, α) of our operator. From left top to right bottom we draw: (C, any α), (H, 0), (D, 0.5), (G, 0), (W, 0.5) and (W, 0). The reader can observe that, geometrically, a greater concentration of facilities is obtained for fairer solutions, being not only the position of the facilities the factor that affect the fairness of a solution, but also the allocation of the users to them, which can be more easily when different facilities are able to cover the same demand points.



(a) C (b) H and $\alpha = 0$ (c) D and $\alpha = 0.5$

(d) G and $\alpha = 0$ (e) W and $\alpha = 0.5$ (f) W and $\alpha = 0$

Figure 5.12: Example of fairness distribution for some results when $n = 90$ and $p = 10$

5.5 Conclusions and further research

We present in this chapter a novel fairness measure for Maximal Covering Location Problems, that combines the OWA operators, early introduced by Yager (1988), and the α-fairness scheme introduced by Atkinson (1970).

We develop suitable mathematical programming models that allow to capture the notion of fairness in the MCLP for the two main frameworks that are studied in the literature: the discrete and continuous. The models are then reformulated as MISOCO problem, using the geometrical insights of the problem, and then, the programs that we propose are able to be solved with the available off-the-shelf sofwares.

We have tested our models using a real data set containing the locations of residential schools and student hostels in Canada. Applying our new scheme to this dataset, we empirically observes that our models provide different solutions in terms of fairness than using the OWA operators and the α-fairness scheme separately. Therefore, we conclude that the inclusion of both in the same allocation scheme provides the decision maker with a wide range of options to find a trade-off between fairness and efficiency.

Further research lines on fairness topic include the using of this combination of schemes into different location problems, as the set covering problem, location problems with capacities and also its incorporation to queue problems where the facilities has to deal with the management of waiting users when the facility is saturated, and fair solutions of the problem could be a successfully too to deal with this problematic.

Chapter 6

Conclusions and future research lines

In this dissertation we present new contributions in different Continuous Location problems. We study novel extensions on maximal covering location problems and develope new approaches to solve large instances of a wide family of multifacility location problems. All the results reported in chapters 2,3, 4, and 5 are new and their contents are based on the published paper Blanco and Gázquez (2021) and the preprints Blanco et al. (2021a,b), that are, at present, under revision in different high-impact OR scientific journals.

Mathematical Optimization plays the most important role in this thesis, both to analyze and solve the proposed models. In particular, Mixed-Integer Non-Linear Programming (MINLP) formulations have been first developed as *'natural'* formulations to describe all problems. However, the intrinsic properties of each of the problems under analysis allow us to derive suitable (Mixed) Integer Linear Programming formulations (MILP), for which more efficient resolution strategies are implemented in the available commercial off-the-shelf solvers. Furthermore, all the Mathematical Programming formulations have been carefully analyzed in order to strengthen them with valid inequalities, strategies for fixing variables and generating initial solutions. Finally, all the proposed approaches have been empirically tested on extensive batteries of experiments on classical and new instances. Specifically, in Chapter 2 we provide a new dataset for continuous covering location which may serve as a benchmark to test further developments on this problem and its extensions.

As already mentioned, this thesis provides a step forward on analyzing Continuous Location problems. Although there is a wide literature on this family of problems since the Weber problem (Weber, 1909) and its extensions, the existence of efficient exact algorithms to solve medium to large size instances is still a challenge for most of the variants of the classical problem because their, in general, NP-hardness complexity. In Chapter 2 we address a difficult well-known family of continuous location problems which combines three ingredients that considerably increase the complexity (but at the same time, the applicability) of the problems: (a) multiple facilities are to be located instead on one, being each demand point allocated to its closest facility; (b) the measure that we consider of the closeness of a demand point in a d-dimensional space from a facility is a ℓ_τ-norm or a block-norm based distance in $\mathbb{R}^d$; and (c) in order to determine the goodness of a set of facilities with respect a set of given demand points we aggregate the distances by means of ordered median function. This family of problems was firstly analyzed in Blanco et al. (2016), in which the authors provided a Mixed Integer Second Order Cone Optimization (MISOCO) reformulation of the problem able to solve, for the first time, problems of small to medium size (up to 50 demand points), using off-the-shell solvers. In this chapter, we propose two exact methods, namely a compact formulation and a branch-and-price procedure, based on a set partitioning formulation of the problem. Moreover, we derive theoretical lower bounds for the problem, which also allow us to develop three three novel matheuristics. We perform an extensive computational battery of experiment that proved, empirically, the usefulness of the branch-and-price approach for medium- to large-sized instances.

On the other hand, in chapters 3, 4, and 5 we analyze practical extensions of the classical Continuous Maximal Covering Location Problem (MCLP). In Chapter 3 we study MCLP in which the facilities are required to be linked through a given graph structure provided that the distance between the linked facilities does not exceed a given limit. We propose a MINLP formulation and derive a MILP formulation based on the geometry of the problem, we also derive two branch-and-cut solution strategies for solving more efficiently the problem based on different relaxations of the problem. Furthermore, we develop a matheuristic algorithm which is capable to obtain good quality solutions for larger-size instances in reasonable CPU times. In total, 5 resolution methods have been proposed and tested in an extensive battery of computational experiments.

In Chapter 4 we analyze a multitype-version of the MCLP in which the position of facilities in different metric spaces are simultaneously decided. A general modeling framework is provided for any type and number of facilities. This model has been also adapted to an hybridized discrete-continuous facility location problem which is further analyzed. Specifically, we derive a novel pure binary linear programming formulation for the hybrid problem. In the Euclidean planar case, a third model is also proposed. The empirical results prove the validity of a branch-and-cut algorithm for dealing with the problem in its general form. In particular, instances with up to 920 demand nodes and two types of facilities (discrete and continuous) could be solved rather efficiently.

In Chapter 5 we present a novel fairness measure for allocation problems that combines the Ordered Weighted Averaging (OWA) operators, introduced by Yager (1988), and the α-fairness scheme introduced by Atkinson (1970). We apply this novel measure to capture fairness allocations for MCLP in both discrete and continuous frameworks. We develop suitable mathematical programming models and they are reformulated as MISOCO problem, using the geometrical insights of the problem. We have tested our models using a real data set containing the locations of residential schools and student hostels in Canada. Applying our new scheme to this dataset, we empirically observes that our models provide different solutions in terms of fairness than using the OWA operators and the α-fairness scheme separately.

This thesis serves also as a starting point to analyze both other Continuous Location Problems with similar tools to the proposed here or to develop new methodologies to solve the extensions that we introduce. Concretely, the branch-and-price approach presented in Chapter 2, may be adequately adapted to solve continuous covering location problems, or even discrete location problems that have been proven to be challenging, as the upgrading-based problems (see, e.g., Blanco and Marín, 2019). One of the main developments in this thesis is the ILP formulation (c-MCLP$^{\text{IP}}$) which serves as basis for modeling and solving different versions of the MCLP. This formulation includes constraints modeling incompatibilities of demand points to be covered by the same facility. Although the general formulation is valid for any d-dimensional spaces, in practice is difficult to implement for $d \geq 3$ since they simplify to check intersections of $(d+1)$ d-dimensional norm-based balls, which can be cumbersome. This is the reason because we develop different branch-and-cut approaches that incorporates this type constraints as needed in the solution procedure. Future lines of research include the development of efficient algorithms to handle these constraints with clique-based inequalities. Another practical extension of the models presented in this dissertation is the consideration of λ-connected graphs in the MCLPIF or the incorporation of interconnection between facilities to other continuous problems. Also, as a topic of forthcoming papers we are analyzing the viability of considering uncertainty and multiperiod settings in multitype maximal covering location problems as well as the development of fairness measures to other resource allocation problems.

References

Manzour-al Ajdad, S., Torabi, S.A., Eshghi, K., 2012. Single-source capacitated multi-facility weber problem—an iterative two phase heuristic algorithm. Computers & Operations Research 39, 1465–1476.

Albornoz, V.M., Zamora, G.E., 2021. Decomposition-based heuristic for the zoning and crop planning problem with adjacency constraints. Top 29, 248–265.

Alcaraz, J., Landete, M., Monge, J.F., 2012. Design and analysis of hybrid metaheuristics for the reliability p-median problem. European Journal of Operational Research 222, 54–64.

Aouad, A., Segev, D., 2019. The ordered k-median problem: surrogate models and approximation algorithms. Mathematical Programming 177, 55–83.

Asudeh, A., Berger-Wolf, T., Dasgupta, B., Sidiropoulos, A., 2020. Maximizing coverage while ensuring fairness: a tale of conflicting objective. ArXiv abs/2007.08069.

Atkinson, A., 1970. On the measurement of inequality. Journal of Economic Theory 3, 244–263.

Avella, P., Sassano, A., Vasil'ev, I., 2007. Computational study of large-scale p-median problems. Mathematical Programming 109, 89–114.

Baldomero-Naranjo, M., Kalcsics, J., Rodríguez-Chía, A.M., 2021. Minmax regret maximal covering location problems with edge demands. Computers & Operations Research 130, 105181.

Balinski, M.L., 1965. Integer programming: methods, uses, computations. Management science 12, 253–313.

Barbati, M., Piccolo, C., 2016. Equality measures properties for location problems. Optimization Letters 10, 903–920.

Barnhart, C., Johnson, E.L., Nemhauser, G.L., Savelsbergh, M.W., Vance, P.H., 1998. Branch-and-price: Column generation for solving huge integer programs. Operations research 46, 316–329.

Batta, R., Dolan, J., Krishnamurthy, N., 1989. The maximal expected covering location problem: Revisited. Transportation science 23, 277–287.

Beasley, J.E., 1985. A note on solving large p-median problems. European Journal of Operational Research 21, 270–273.

Benati, S., Ponce, D., Puerto, J., Rodríguez-Chía, A.M., 2022. A branch-and-price procedure for clustering data that are graph connected. European Journal of Operational Research 297, 817–830.

Bennett, V.L., Eaton, D.J., Church, R.L., 1982. Selecting sites for rural health workers. Social Science & Medicine 16, 63–72.

Berge, C., 1957. Two theorems in graph theory. Proceedings of the National Academy of Sciences of the United States of America 43, 842.

Berman, O., Drezner, T., Drezner, Z., Wesolowsky, G., 2009. A defensive maximal covering problem on a network. International Transactions in Operational Research 16, 69–86.

Berman, O., Drezner, Z., 2008. The p-median problem under uncertainty. European Journal of Operational Research 189, 19–30.

Berman, O., Kalcsics, J., Krass, D., 2016. On covering location problems on networks with edge demand. Computers & Operations Research 74, 214–227.

Bertsekas, D.P., Gallager, R.G., Humblet, P., 1992. Data networks. volume 2. Prentice-Hall International New Jersey.

Bertsimas, D., Farias, V.F., Trichakis, N., 2011. The price of fairness. Operations research 59, 17–31.

Bertsimas, D., Farias, V.F., Trichakis, N., 2012. On the efficiency-fairness trade-off. Management Science 58, 2234–2250.

Blanco, V., 2019. Ordered p-median problems with neighbourhoods. Computational Optimization and Applications 73, 603–645.

Blanco, V., Gázquez, R., 2021. Continuous maximal covering location problems with interconnected facilities. Computers & Operations Research 132, 105310.

Blanco, V., Gázquez, R., Saldanha-da Gama, F., 2021a. Multitype maximal covering location problems: Hybridizing discrete and continuous problems. arXiv preprint arXiv:2111.14494 .

Blanco, V., Gázquez, R., Ponce, D., Puerto, J., 2021b. A branch-and-price approach for the continuous multifacility monotone ordered median problem. arXiv preprint arXiv:2108.00407 .

Blanco, V., Japón, A., Ponce, D., Puerto, J., 2021c. On the multisource hyperplanes location problem to fitting set of points. Computers & Operations Research 128, 105124.

Blanco, V., Japón, A., Puerto, J., 2021d. Multiclass optimal classification trees with svm-splits. arXiv preprint arXiv:2111.08674 .

Blanco, V., Marín, A., 2019. Upgrading nodes in tree-shaped hub location. Computers & Operations Research 102, 75–90.

Blanco, V., Puerto, J., 2021a. Covering problems with polyellipsoids: A location analysis perspective. European Journal of Operational Research 289, 44–58.

Blanco, V., Puerto, J., 2021b. On hub location problems in geographically flexible networks. International Transactions in Operational Research .

Blanco, V., Puerto, J., ElHaj-BenAli, S., 2014. Revisiting several problems and algorithms in continuous location with ell_τ-norms. Computational Optimization and Applications 58, 563–595.

Blanco, V., Puerto, J., ElHaj-BenAli, S., 2016. Continuous multifacility ordered median location problems. European Journal of Operational Research 250, 56–64.

Blanco, V., Puerto, J., Ponce, D., 2017. Continuous location under the effect of 'refraction'. Mathematical Programming 161, 33–72.

Blanco, V., Puerto, J., Salmerón, R., 2018. Locating hyperplanes to fitting set of points: A general framework. Computers & Operations Research 95, 172–193.

Blanquero, R., Carrizosa, E., Boglárka, G., et al., 2016. Maximal covering location problems on networks with regional demand. Omega 64, 77–85.

Blum, H., 1967. A transformation for extracting new descriptors of shape. Models for Perception of Speech and Visual Forms, 1967 , 362–380.

Boffey, B., Narula, S.C., 1998. Models for multi-path covering-routing problems. Annals of Operations Research 82, 331–342.

Bonald, T., Massoulié, L., 2001. Impact of fairness on internet performance. SIGMETRICS Perform. Eval. Rev. 29, 82–91. URL: https://doi.org/10.1145/384268.378438, doi:10.1145/384268.378438.

Brimberg, J., 1995. The fermat—weber location problem revisited. Mathematical programming 71, 71–76.

Brimberg, J., Chen, R., Chen, D., 1998. Accelerating convergence in the fermat–weber location problem. Operations Research Letters 22, 151–157.

Brimberg, J., Love, R.F., 1992. Local convergence in a generalized fermat-weber problem. Annals of Operations Research 40, 33–66.

Brimberg, J., Love, R.F., 1993. Global convergence of a generalized iterative procedure for the minisum location problem with lp distances. Operations Research 41, 1153–1163.

Bruno, G., Genovese, A., Improta, G., 2014. A historical perspective on location problems. BSHM Bulletin: Journal of the British Society for the History of Mathematics 29, 83–97.

Butt, S.E., Cavalier, T.M., 1996. An efficient algorithm for facility location in the presence of forbidden regions. European Journal of Operational Research 90, 56–70.

Callaghan, B., Salhi, S., Nagy, G., 2017. Speeding up the optimal method of drezner for the p-centre problem in the plane. European Journal of Operational Research 257, 722–734.

Calvino, J.J., López-Haro, M., Muñoz-Ocaña, J.M., Puerto, J., Rodríguez-Chía, A.M., 2022. Segmentation of scanning-transmission electron microscopy images using the ordered median problem. European Journal of Operational Research .

Caragiannis, I., Kaklamanis, C., Kanellopoulos, P., Kyropoulou, M., 2009. On low-envy truthful allocations, in: International Conference on Algorithmic DecisionTheory, Springer. pp. 111–119.

Carrizosa, E., Conde, E., Fernandez, F., Puerto, J., 1992. A management tool for indicator-supported systems: A public health service application. European Journal of Operational Research 61, 204–214.

Carrizosa, E., Conde, E., Muñoz-Marquez, M., Puerto, J., 1995. The generalized weber problem with expected distances. RAIRO-Operations Research 29, 35–57.

Carrizosa, E., Muñoz-Márquez, M., Puerto, J., 1998. The weber problem with regional demand. European Journal of Operational Research 104, 358–365.

Chandrasekaran, R., Tamir, A., 1989. Open questions concerning weiszfeld's algorithm for the fermat-weber location problem. Mathematical Programming 44, 293–295.

Chanta, S., Mayorga, M.E., McLay, L.A., 2014. The minimum p-envy location problem with requirement on minimum survival rate. Computers & Industrial Engineering 74, 228–239. URL: https://www.sciencedirect.com/science/article/pii/S036083521400179X, doi:https://doi.org/10.1016/j.cie.2014.06.001.

Chen, G., Daskin, M., Shen, Z.J., Uryasev, S., 2006. The α-reliable mean-excess regret model for stochastic facility location modeling. Naval Research Logistics 53, 617–626.

Cherkesly, M., Landete, M., Laporte, G., 2019. Median and covering location problems with interconnected facilities. Computers & Operations Research 107, 1–18.

Chiclana, F., Herrera, F., Herrera-Viedma, E., Martınez, L., 2003. A note on the reciprocity in the aggregation of fuzzy preference relations using owa operators. Fuzzy Sets and Systems 137, 71–83.

Chung, C.H., 1986. Recent applications of the maximal covering location planning (mclp) model. Journal of the Operational Research Society 37, 735–746.

Church, R., ReVelle, C., 1974. The maximal covering location problem. Papers of the Regional Science Association 32, 101–118.

Church, R.L., 1984. The planar maximal covering location problem. Journal of Regional Science 24, 185–201.

Church, R.L., 1999. Location modelling and gis. Geographical information systems 1, 293–303.

Church, R.L., Murray, A., 2018a. Classic beginnings, in: Location Covering Models. Springer, pp. 23–47.

Church, R.L., Murray, A., 2018b. Location Covering Models. Springer.

Church, R.L., Stoms, D.M., Davis, F.W., 1996. Reserve selection as a maximal covering location problem. Biological conservation 76, 105–112.

Contreras, I., O'Kelly, M., 2019. Hub location problems, in: Location science. Springer, pp. 327–363.

Contreras, I., Ortiz-Astorquiza, C., 2019. Hierarchical facility location problems, in: Laporte, G., Nickel, S., Saldanha-da-Gama, F. (Eds.), Location Science. Springer International Publishing, 2nd edition. chapter 13, pp. 365–389.

Contreras, I., Tanash, M., Vidyarthi, N., 2017. Exact and heuristic approaches for the cycle hub location problem. Annals of Operations Research 258, 655–677.

Cooper, L., 1963. Location-allocation problems. Operations research 11, 331–343.

Cordeau, J.F., Furini, F., Ljubić, I., 2019. Benders decomposition for very large scale partial set covering and maximal covering location problems. European Journal of Operational Research 275, 882–896.

Correia, I., Lourenço, L.L., Saldanha-da Gama, F., 2012. Project scheduling with flexible resources: formulation and inequalities. OR spectrum 34, 635–663.

Craven, B.D., 2012. Mathematical programming and control theory. Springer Science & Business Media.

Current, J., O'Kelly, M., 1992. Locating emergency warning sirens. Decision Sciences 23, 221–234.

Current, J.R., Schilling, D.A., 1990. Analysis of errors due to demand data aggregation in the set covering and maximal covering location problems. Geographical Analysis 22, 116–126.

Current, J.R., Storbeck, J.E., 1988. Capacitated covering models. Environment and planning B: planning and Design 15, 153–163.

Curtin, K.M., Hayslett-McCall, K., Qiu, F., 2010. Determining optimal police patrol areas with maximal covering and backup covering location models. Networks and spatial economics 10, 125–145.

Dantzig, G.B., 1949. Programming in a linear structure. Econometrica 17, 73–74.

Dantzig, G.B., 2014. The nature of mathematical programming. URL: https://glossary.informs.org/second.php?page=nature.html.

Danzer, L., Grünbaum, B., Klee, V., 1963. Helly's theorem and it's relatives, in: Proceedings of the Symposium on Pure Mathemathics, American Mathematical Society. pp. 101–180.

Daskin, M., 1983. A maximum expected covering location model: formulation, properties and heuristic solution. Transportation science 17, 48–70.

Daskin, M., 1995. Network and discrete location: Models, algorithms and applications/mark s. daskin .

Daskin, M., Hesse, S., ReVelle, C., 1997. α-reliable p-minimax regret: a new model for strategic facility location modeling. Location Science 5, 227–246.

Daskin, M.S., Haghani, A.E., Khanal, M., Malandraki, C., 1989. Aggregation effects in maximum covering models. Annals of Operations Research 18, 113–139.

de Assis Corrêa, F., Lorena, L., Ribeiro, G., 2009. A decomposition approach for the probabilistic maximal covering location-allocation problem. Computers & Operations Research 36, 2729–2739.

Deleplanque, S., Labbé, M., Ponce, D., Puerto, J., 2020. A branch-price-and-cut procedure for the discrete ordered median problem. INFORMS Journal on Computing 32, 582–599.

Demaine, E.D., Hajiaghayi, M., Mahini, H., Sayedi-Roshkhar, A.S., Oveisgharan, S., Zadimoghaddam, M., 2009. Minimizing movement. ACM Transactions on Algorithms (TALG) 5, 1–30.

Desrosiers, J., Lübbecke, M.E., 2005. A primer in column generation, in: Column generation. Springer, pp. 1–32.

Dolu, N., Hastürk, U., Tural, M.K., 2020. Solution methods for a min–max facility location problem with regional customers considering closest euclidean distances. Computational Optimization and Applications 75, 537–560.

Domínguez-Marín, P., 2003. The Discrete Ordered Median Problem: Models and Solution Methods. Springer.

Drezner, T., Drezner, Z., 2014. The maximin gradual cover location problem. OR spectrum 36, 903–921.

Drezner, T., Goldstein, Z.D.Z., 2010. A stochastic gradual cover location problem. Naval Research Logistics (NRL) 57, 367–372.

Drezner, Z., 1984. The p-centre problem—heuristic and optimal algorithms. Journal of the Operational Research Society 35, 741–748.

Drezner, Z., 1985. A solution to the weber location problem on the sphere. Journal of the Operational Research Society 36, 333–334.

Drezner, Z., 1995. A note on accelerating the weiszfeld procedure. Location Science 3, 275–279.

Drezner, Z., Hamacher, H.W., 2004. Facility location: applications and theory. Springer Science & Business Media.

Drezner, Z., Nickel, S., 2009. Constructing a dc decomposition for ordered median problems. Journal of global optimization 45, 187–201.

Drezner, Z., Weslowsky, G., 1980. Optimal location of a facility relative to area demands. Naval Research Logistics Quarterly 27, 199–206.

Du Merle, O., Villeneuve, D., Desrosiers, J., Hansen, P., 1999. Stabilized column generation. Discrete Mathematics 194, 229–237.

Dwyer, F.R., Evans, J.R., 1981. A branch and bound algorithm for the list selection problem in direct mail advertising. Management Science 27, 658–667.

Ehrgott, M., Gandibleux, X., 2000. A survey and annotated bibliography of multiobjective combinatorial optimization. OR-Spektrum 22, 425–460.

Eilon, S., Watson-Gandy, C.D.T., Christofides, N., de Neufville, R., 1974. Distribution management-mathematical modelling and practical analysis. IEEE Transactions on Systems, Man, and Cybernetics , 589–589.

Elloumi, S., Labbé, M., Pochet, Y., 2004. A new formulation and resolution method for the p-center problem. INFORMS Journal on Computing 16, 84–94.

Elzinga, J., Hearn, D., 1972. Geometrical solutions for some minimax location problems. Transportation Science 6, 379–394.

Emir-Farinas, H., Francis, R.L., 2005. Demand point aggregation for planar covering location models. Annals of Operations Research 136, 175–192.

Erkut, E., 1993. Inequality measures for location problems. Computers & Operations Research .

Ernst, A.T., Krishnamoorthy, M., 1996. Efficient algorithms for the uncapacitated single allocation p-hub median problem. Location science 4, 139–154.

Espejo, I., Marin, A., Puerto, J., Rodríguez-Chía, A.M., 2009. A comparison of formulations and solution methods for the minimum-envy location problem. Computers & Operations Research 36, 1966–1981.

Espejo, I., Puerto, J., Rodríguez-Chía, A.M., 2021. A comparative study of different formulations for the capacitated discrete ordered median problem. Computers & Operations Research 125, 105067.

Espejo, L., Galvao, R., Boffey, B., 2003. Dual-based heuristics for a hierarchical covering location problem. Computers & Operations Research 30, 165–180.

Farahani, R.Z., Asgari, N., Heidari, N., Hosseininia, M., Goh, M., 2012. Covering problems in facility location: A review. Computers & Industrial Engineering 62, 368–407.

Fekete, S.P., Mitchell, J.S., Beurer, K., 2005. On the continuous fermat-weber problem. Operations Research 53, 61–76.

Fernández, E., Pozo, M.A., Puerto, J., 2014. Ordered weighted average combinatorial optimization: Formulations and their properties. Discrete Applied Mathematics 169, 97–118.

Fernández, E., Landete, M., 2019. Fixed-charge facility location problems, in: Laporte, G., Nickel, S., Saldanha-da-Gama, F. (Eds.), Location Science. Springer International Publishing, 2nd edition. chapter 4, pp. 67–98.

Filev, D., Yager, R.R., 1995. Analytic properties of maximum entropy owa operators. Information Sciences 85, 11–27.

Fourour, S., Lebbah, Y., 2020. Equitable optimization for multicast communication. International Journal of Decision Support System Technology (IJDSST) 12, 1–25.

Francis, R.L., Lowe, T.J., Tamir, A., 2000. Aggregation error bounds for a class of location models. Operations research 48, 294–307.

Frenk, J.B.G., Melo, M.T., Zhang, S., 1994. A weiszfeld method for a generalized lp distance minisum location model in continuous space. Computers & Operations Research .

Fullér, R., Majlender, P., 2001. An analytic approach for obtaining maximal entropy owa operator weights. Fuzzy sets and Systems 124, 53–57.

Fullér, R., Majlender, P., 2003. On obtaining minimal variability owa operator weights. Fuzzy sets and systems 136, 203–215.

Gamrath, G., Anderson, D., Bestuzheva, K., Chen, W.K., Eifler, L., Gasse, M., Gemander, P., Gleixner, A., Gottwald, L., Halbig, K., Hendel, G., Hojny, C., Koch, T., Le Bodic, P., Maher, S.J., Matter, F., Miltenberger, M., Mühmer, E., Müller, B., Pfetsch, M.E., Schlösser, F., Serrano, F., Shinano, Y., Tawfik, C., Vigerske, S., Wegscheider, F., Weninger, D., Witzig, J., 2020. The SCIP Optimization Suite 7.0. Technical Report. Optimization Online. URL: http://www.optimization-online.org/DB_HTML/2020/03/7705.html.

García, S., Labbé, M., Marín, A., 2011. Solving large p-median problems with a radius formulation. INFORMS Journal on Computing 23, 546–556.

Garcia-Quiles, S., Marín, A., 2019. Covering location problems, in: Laporte, G., Nickel, S., Saldanha-da-Gama, F. (Eds.), Location Science. Springer International Publishing, 2nd edition. chapter 5, pp. 93–113.

Gendreau, M., Laporte, G., Semet, F., 1997. The covering tour problem. Operations Research 45, 568–576.

Geoffrion, A.M., 1977. Objective function approximations in mathematical programming. Mathematical Programming 13, 23–37.

Gharaei, A., Hoseini Shekarabi, S.A., Karimi, M., 2020. Modelling and optimal lot-sizing of the replenishments in constrained, multi-product and bi-objective epq models with defective products: Generalised cross decomposition. International Journal of Systems Science: Operations & Logistics 7, 262–274.

Gleason, J.M., 1975. A set covering approach to bus stop location. Omega 3, 605–608.

Goldengorin, B., Krushinsky, D., 2011. Complexity evaluation of benchmark instances for the p-median problem. Mathematical and Computer Modelling 53, 1719–1736.

Goodchild, M.F., Lee, J., 1989. Coverage problems and visibility regions on topographic surfaces. Annals of Operations Research 18, 175–186.

Gorski, J., Pfeuffer, F., Klamroth, K., 2007. Biconvex sets and optimization with biconvex functions: a survey and extensions. Mathematical methods of operations research 66, 373–407.

Gross, C., 2008. A measure of fairness: An investigative framework to explore perceptions of fairness and justice in a real-life social conflict. Human Ecology Review , 130–140.

Gusfield, D., 1990. Very simple methods for all pairs network flow analysis. SIAM Journal on Computing 19, 143–155.

Hakimi, S.L., 1964. Optimum locations of switching centers and the absolute centers and medians of a graph. Operations research 12, 450–459.

Hakimi, S.L., 1965. Optimum distribution of switching centers in a communication network and some related graph theoretic problems. Operations research 13, 462–475.

Hamacher, H.W., Nickel, S., 1998. Classification of location models. Location Science 6, 229–242.

Hayden, H.P., 1981. Voice flow control in integrated packet networks. Ph.D. thesis. Massachusetts Institute of Technology.

Heckmann, I., Nickel, S., 2019. Location logistics in suppply chain management, in: Laporte, G., Nickel, S., Saldanha-da-Gama, F. (Eds.), Location Science. Springer International Publishing, 2nd edition. chapter 16, pp. 453–476.

Helly, E., 1923. Über mengen konvexer körper mit gemeinschaftlichen punkte (in German). Jahresbericht der Deutschen Mathematiker-Vereinigung 32, 175–176.

Helly, E., 1930. Über systeme von abgeschlossenen mengen mit gemeinschaftlichen punkten. Monatshefte für Mathematik und Physik 37, 281–302.

Herrera, F., Herrera-Viedma, E., Verdegay, J.L., 1996. Direct approach processes in group decision making using linguistic owa operators. Fuzzy Sets and systems 79, 175–190.

Heyns, A., van Vuuren, J., 2018. Multi-type, multi-zone facility location. Geographical Analysis 50, 3–31.

Huang, K., Rafiei, R., 2019. Equitable last mile distribution in emergency response. Computers & Industrial Engineering 127, 887–900. URL: https://www.sciencedirect.com/science/article/pii/S0360835218305655, doi:https://doi.org/10.1016/j.cie.2018.11.025.

Hurkała, J., Hurkała, A., 2013. Fair optimization with advanced aggregation operators in a multicriteria facility layout problem, in: 2013 Federated Conference on Computer Science and Information Systems, IEEE. pp. 355–362.

Irawan, C.A., 2016a. Aggregation and non aggregation techniques for large facility location problems-a survey. Yugoslav Journal of Operations Research 25.

Irawan, C.A., 2016b. Aggregation and non aggregation techniques for large facility location problems-a survey. Yugoslav Journal of Operations Research 25.

Jaffe, J., 1981. Bottleneck flow control. IEEE Transactions on Communications 29, 954–962.

Jain, R.K., Chiu, D.M.W., Hawe, W.R., et al., 1984. A quantitative measure of fairness and discrimination for resource allocation in shared computer system. Eastern Research Laboratory, Digital Equipment Corporation, Hudson, MA .

Jiang, Y., Wu, X., Chen, B., Hu, Q., 2021. Rawlsian fairness in push and pull supply chains. European Journal of Operational Research 291, 194–205.

Kalczynski, P., Brimberg, J., Drezner, Z., 2021. Less is more: discrete starting solutions in the planar p-median problem. TOP , 1–26.

Kalvenes, J., Kennington, J., Olinick, E., 2005. Hierarchical cellular network design with channel allocation. European Journal of Operational Research 160, 3–18.

Kantorovich, L.V., 1960. Mathematical methods of organizing and planning production. Management science 6, 366–422.

Karp, R.M., 1972. Reducibility among combinatorial problems, in: Complexity of computer computations. Springer, pp. 85–103.

Katz, I.N., 1974. Local convergence in fermat's problem. Mathematical Programming 6, 89–104.

Kelly, F.P., Maulloo, A.K., Tan, D.K.H., 1998. Rate control for communication networks: shadow prices, proportional fairness and stability. Journal of the Operational Research society 49, 237–252.

Klamroth, K., 2001. Planar weber location problems with line barriers. Optimization 49, 517–527.

Kleinberg, J., Rabani, Y., Tardos, E., 1999. Fairness in routing and load balancing, in: 40th Annual Symposium on Foundations of Computer Science (Cat. No.99CB37039), pp. 568–578. doi:10.1109/SFFCS.1999.814631.

Korani, E., Sahraeian, R., 2013. The hierarchical hub covering problem with innovative allocation procedure by covering radiuses. Scientia Iranica 20, 2138–2160.

Krau, S., 1997. Extensions du probleme de Weber. École Polytechnique de Montréal.

Kuhn, H.W., 1967. On a pair of dual nonlinear programs. Nonlinear programming 1, 38–45.

Kuhn, H.W., 1973. A note on fermat's problem. Mathematical programming 4, 98–107.

Kuhn, H.W., Kuenne, R.E., 1962. An efficient algorithm for the numerical solution of the generalized weber problem in spatial economics. Journal of Regional Science 4, 21–33.

Küçükaydın, H., Aras, N., 2020. Gradual covering location problem with multi-type facilities considering customer preferences. Computers & Industrial Engineering 147, 106577.

Labbé, M., Laporte, G., Martín, I.R., González, J.J.S., 2004. The ring star problem: Polyhedral analysis and exact algorithm. Networks: An International Journal 43, 177–189.

Labbé, M., Ponce, D., Puerto, J., 2017. A comparative study of formulations and solution methods for the discrete ordered p-median problem. Computers & Operations Research 78, 230–242.

Lan, T., Kao, D., Chiang, M., Sabharwal, A., 2010. An axiomatic theory of fairness in network resource allocation. IEEE.

Laporte, G., Nickel, S., Saldanha-da Gama, F., 2019a. Introduction to location science, in: Location science. Springer, pp. 1–21.

Laporte, G., Nickel, S., da Gama, F.S., 2019b. Location science. Springer.

Launhardt, C., 1900. The principles of location: the theory of the trace. part i: the commercial trace (bewley a, trans., 1900).

Leclerc, P.D., McLay, L.A., Mayorga, M.E., 2012. Modeling Equity for Allocating Public Resources. Springer New York, New York, NY. pp. 97–118. URL: https://doi.org/10.1007/978-1-4614-0806-2_4, doi:10.1007/978-1-4614-0806-2_4.

Li, C.F., Li, J.S., 2006. Fair allocation using a fast smart market auction. European journal of operational research 172, 352–365.

Li, X., Ramshani, M., Huang, Y., 2018. Cooperative maximal covering models for humanitarian relief chain management. Computers & industrial engineering 119, 301–308.

Lipton, R.J., Markakis, E., Mossel, E., Saberi, A., 2004. On approximately fair allocations of indivisible goods, in: Proceedings of the 5th ACM Conference on Electronic Commerce, pp. 125–131.

Liu, X., Chen, L., 2004. On the properties of parametric geometric owa operator. International Journal of Approximate Reasoning 35, 163–178.

Love, R.F., Morris, J.G., Wesolowsky, G.O., 1988. Facilities location .

Luss, H., 1999. On equitable resource allocation problems: A lexicographic minimax approach. Operations Research 47, 361–378.

Maldonado, S., Merigó, J., Miranda, J., 2018. Redefining support vector machines with the ordered weighted average. Knowledge-Based Systems 148, 41–46.

Mallozzi, L., Puerto, J., Rodríguez-Madrena, M., 2019. On location-allocation problems for dimensional facilities. Journal of Optimization Theory and Applications 182, 730–767.

Mandal, S., Patra, N., Pal, M., 2021. Covering problem on fuzzy graphs and its application in disaster management system. Soft Computing 25, 2545–2557.

Marín, A., Martínez-Merino, L.I., Puerto, J., Rodríguez-Chía, A.M., 2022. The soft-margin support vector machine with ordered weighted average. Knowledge-Based Systems 237, 107705.

Marín, A., Nickel, S., Puerto, J., Velten, S., 2009. A flexible model and efficient solution strategies for discrete location problems. Discrete Applied Mathematics 157, 1128–1145.

Marín, A., Nickel, S., Velten, S., 2010. An extended covering model for flexible discrete and equity location problems. Mathematical Methods of Operations Research 71, 125–163.

Marín, A., Ponce, D., Puerto, J., 2020. A fresh view on the discrete ordered median problem based on partial monotonicity. European Journal of Operational Research 286, 839–848.

Marsh, M.T., Schilling, D.A., 1994. Equity measurement in facility location analysis: A review and framework. European Journal of Operational Research 74, 1–17. URL: https://www.sciencedirect.com/science/article/pii/0377221794902003, doi:https://doi.org/10.1016/0377-2217(94)90200-3.

Marín, A., Martínez-Merino, L.I., Rodríguez-Chía, A.M., da Gama, F.S., 2018. Multi-period stochastic covering location problems: Modeling framework and solution approach. European Journal of Operational Research 268, 432–449.

Matisziw, T.C., Murray, A.T., 2009a. Area coverage maximization in service facility siting. Journal of geographical systems 11, 175–189.

Matisziw, T.C., Murray, A.T., 2009b. Siting a facility in continuous space to maximize coverage of a region. Socio-Economic Planning Sciences 43, 131–139.

Megiddo, N., 1974. Optimal flows in networks with multiple sources and sinks. Mathematical Programming 7, 97–107.

Megiddo, N., Supowit, K.J., 1984. On the complexity of some common geometric location problems. SIAM journal on computing 13, 182–196.

Megiddo, N., Zemel, E., Hakimi, S.L., 1983. The maximum coverage location problem. SIAM Journal on Algebraic Discrete Methods 4, 253–261.

Mehrez, A., 1983. A note on the linear integer formulation of the maximal covering location problem with facility placement on the entire plane. Journal of Regional Science 23, 553–555.

Mehrez, A., Stulman, A., 1982. The maximal covering location problem with facility placement on the entire plane. Journal of Regional Science 22, 361–365.

Mehrez, A., Stulman, A., 1984. An extended continuous maximal covering location problem with facility placement. Computers & Operations Research 11, 19–23.

du Merle, O., Vial, J., 2002. Proximal ACCPM, acutting plane method for column generation and Lagrangean relaxation: application to the p-median problem. Technical Report. HEC Genève, Université de Genève.

Mesa, J., 1991. Multiperiod medians on networks. RAIRO-Operations Research 25, 85–95.

Miehle, W., 1958. Link-length minimization in networks. Operations research 6, 232–243.

Miller, C.E., Tucker, A.W., Zemlin, R.A., 1960. Integer programming formulation of traveling salesman problems. Journal of the ACM (JACM) 7, 326–329.

Mirchandani, P.B., Francis, R.L., 1990. Discrete location theory.

Mo, J., Walrand, J., 2000. Fair end-to-end window-based congestion control. IEEE/ACM Transactions on networking 8, 556–567.

Moon, I.D., Chaudhry, S.S., 1984. An analysis of network location problems with distance constraints. Management Science 30, 290–307.

Moore, G., ReVelle, C., 1982. The hierarchical service location problem. Management science 28, 775–780.

Morris, J.G., Verdini, W.A., 1979. Minisum i p distance location problems solved via a perturbed problem and weiszfeld's algorithm. Operations Research 27, 1180–1188.

Muñoz-Ocaña, J.M., Bouziane, A., Sakina, F., Baker, R.T., Hungría, A.B., Calvino, J.J., Rodríguez-Chía, A.M., López-Haro, M., 2020. Optimization of stem-haadf electron tomography reconstructions by parameter selection in compressed sensing total variation minimization-based algorithms. Particle & Particle Systems Characterization 37, 2000070.

Murray, A.T., 2013. Optimising the spatial location of urban fire stations. Fire Safety Journal 62, 64–71.

Murray, A.T., 2016. Maximal coverage location problem: impacts, significance, and evolution. International Regional Science Review 39, 5–27.

Murray, A.T., Matisziw, T.C., Wei, H., Tong, D., 2008. A geocomputational heuristic for coverage maximization in service facility siting. Transactions in GIS 12, 757–773.

Murray, A.T., O'Kelly, M.E., 2002. Assessing representation error in point-based coverage modeling. Journal of Geographical Systems 4, 171–191.

Murray, A.T., Tong, D., 2007. Coverage optimization in continuous space facility siting. International Journal of Geographical Information Science 21, 757–776.

Nash, J., 1950. The bargaining problem. Econometrica 18, 155–162.

Nesterov, Y., Nemirovskii, A., 1994. Interior-point polynomial algorithms in convex programming. SIAM.

Netzer, A., Meisels, A., Zivan, R., 2016. Distributed envy minimization for resource allocation. Autonomous Agents and Multi-Agent Systems 30, 364–402.

Neumann, J.v., 1947. Discussion of a maximization problem. Institute for Advanced Study. Manuscript .

Nickel, S., Puerto, J., 2006. Location theory: a unified approach. Springer Science & Business Media.

Nickel, S., Puerto, J., Rodriguez-Chia, A.M., 2003. An approach to location models involving sets as existing facilities. Mathematics of Operations Research 28, 693–715.

Ogryczak, W., Luss, H., Pióro, M., Nace, D., Tomaszewski, A., 2014. Fair optimization and networks: A survey. Journal of Applied Mathematics 2014.

Ogryczak, W., Perny, P., Weng, P., 2011. On minimizing ordered weighted regrets in multiobjective markov decision processes, in: International Conference on Algorithmic Decision Theory, Springer. pp. 190–204.

Ogryczak, W., Tamir, A., 2003. Minimizing the sum of the k largest functions in linear time. Information Processing Letters 85, 117–122.

Ogryczak, W., Trzaskalik, T., 2006. Equity, fairness and multicriteria optimization. Multiple Criteria Decision Making'05 , 185–199.

O'Hagan, M., 1988. Aggregating template or rule antecedents in real-time expert systems with fuzzy set logic, in: Twenty-second Asilomar conference on signals, systems and computers, IEEE. pp. 681–689.

O'Kelly, M.E., 1992. A clustering approach to the planar hub location problem. Annals of Operations Research 40, 339–353.

Olender, P., Ogryczak, W., 2019. A revised variable neighborhood search for the discrete ordered median problem. European Journal of Operational Research 274, 445–465.

Orlandini, R., 2019. Residential School Locations Dataset (CSV Format). URL: https://doi.org/10.5683/SP2/RIYEMU, doi:10.5683/SP2/RIYEMU.

Padberg, M., Grötschel, M., 1985. Polyhedral computations, in: The Traveling Salesman Problem. A Guided Tour of Combinatorial Optimization. Wiley, pp. 307–360.

Padberg, M., Rinaldi, G., 1987. Optimization of a 532-city symmetric traveling salesman problem by branch and cut. Operations research letters 6, 1–7.

Pessoa, A., Uchoa, E., Aragão, M.P., Rodrigues, R., 2010. Exact Algorithm over an Arctime-Indexed Formulation for Parallel Machine Scheduling Problems. Mathematical Programming Computation 2, 259–290.

Peters, B., Herrmann, F.J., 2019. Generalized minkowski sets for the regularization of inverse problems. arXiv preprint arXiv:1903.03942 .

Pinto, J.V., 1977. Launhardt and location theory: Rediscovery of a neglected book. Journal of Regional science 17, 17–29.

Plastria, F., 2002. Continuous covering location problems. Facility location: applications and theory 1, 37–79.

Ponce, D., Puerto, J., Ricca, F., Scozzari, A., 2018. Mathematical programming formulations for the efficient solution of the k-sum approval voting problem. Computers & Operations Research 98, 127–136.

Preparata, F.P., 1977. The medial axis of a simple polygon, in: International Symposium on Mathematical Foundations of Computer Science, Springer. pp. 443–450.

Puerto, J., 2020. An exact completely positive programming formulation for the discrete ordered median problem: an extended version. Journal of Global Optimization 77, 341–359.

Puerto, J., Fernández, F., 1994. Multicriteria decisions in location. Studies in Locational Analysis 7, 185–199.

Puerto, J., Ramos, A., Rodríguez-Chía, A.M., 2011. Single-allocation ordered median hub location problems. Computers & Operations Research 38, 559–570.

Puerto, J., Ramos, A., Rodríguez-Chía, A.M., 2013. A specialized branch & bound & cut for single-allocation ordered median hub location problems. Discrete Applied Mathematics 161, 2624–2646.

Puerto, J., Ramos, A., Rodríguez-Chía, A.M., Sánchez-Gil, M.C., 2016. Ordered median hub location problems with capacity constraints. Transportation Research Part C: Emerging Technologies 70, 142–156.

Puerto, J., Rodríguez-Chía, A.M., 1999. Location of a moving service facility. Mathematical methods of operations research 49, 373–393.

Puerto, J., Rodríguez-Chía, A.M., 2011. On the structure of the solution set for the single facility location problem with average distances. Mathematical programming 128, 373–401.

Puerto, J., Rodríguez-Chía, A.M., 2019. Ordered median location problems, in: Laporte, G., Nickel, S., Saldanha-da-Gama, F. (Eds.), Location Science. Springer International Publishing, 2nd edition. chapter 10, pp. 261–302.

Puerto, J., Tamir, A., 2005. Locating tree-shaped facilities using the ordered median objective. Mathematical Programming 102, 313–338.

Rahmattalabi, A., Vayanos, P., Fulginiti, A., Rice, E., Wilder, B., Yadav, A., Tambe, M., 2020. Exploring algorithmic fairness in robust graph covering problems. arXiv preprint arXiv:2006.06865 .

Ratick, S., Osleeb, J., Hozumi, D., 2009. Application and extension of the moore and revelle hierarchical maximal covering model. Socio-Economic Planning Sciences 43, 92–101.

Reinelt, G., 1991. Tsplib—a traveling salesman problem library. ORSA journal on computing 3, 376–384.

Reinelt, G., 1992a. Fast heuristics for large geometric traveling salesman problems. ORSA Journal on computing 4, 206–217.

Reinelt, G., 1992b. Fast heuristics for large geometric traveling salesman problems. ORSA Journal on computing 4, 206–217.

ReVelle, C., 1989. Review, extension and prediction in emergency service siting models. European Journal of Operational Research 40, 58–69.

ReVelle, C., Hogan, K., 1989. The maximum availability location problem. Transportation science 23, 192–200.

Revelle, C., Hogan, K., 1989. The maximum reliability location problem and α-reliable p-center problem: derivatives of the probabilistic location set covering problem. Annals of Operations Research 18, 155–173.

ReVelle, C.S., Eiselt, H.A., 2005. Location analysis: A synthesis and survey. European journal of operational research 165, 1–19.

Righini, G., Zaniboni, L., 2007. A branch-and-price algorithm for the multi-source weber problem. International Journal of Operational Research 2, 188–207.

Rockafellar, R., Uryasev, S., 2000. Optimization of conditional value-at-risk. Journal of Risk 2, 21–41.

Rockafellar, R., Uryasev, S., 2002. Conditional value-at-risk for general loss functions. Journal of Banking & Finance 26, 1443–1471.

Romich, A., Lan, G., Smith, J.C., 2015. A robust sensor covering and communication problem. Naval Research Logistics (NRL) 62, 582–594.

Roth, R., 1969. Computer solutions to minimum-cover problems. Operations Research 17, 455–465.

Ryan, D.M., Foster, B.A., 1981. An integer programming approach to scheduling. Computer scheduling of public transport urban passenger vehicle and crew scheduling , 269–280.

Sakai, S., Togasaki, M., Yamazaki, K., 2003. A note on greedy algorithms for the maximum weighted independent set problem. Discrete Applied Mathematics 126, 313–322. doi:https://doi.org/10.1016/S0166-218X(02)00205-6.

Saydam, C., McKnew, M., 1985. Applications and implementation a separable programming approach to expected coverage: an application to ambulance location. Decision Sciences 16, 381–398.

Schilling, D.A., Revelle, C., Cohon, J., Elzinga, D.J., 1980. Some models for fire protection locational decisions. European Journal of Operational Research 5, 1–7.

Senne, E.L.F., Pereira, M.A., Lorena, L.A.N., 2010. A decomposition heuristic for the maximal covering location problem. Advances in Operations Research 2010.

Sherali, H.D., Nordai, F.L., 1988. Np-hard, capacitated, balanced p-median problems on a chain graph with a continuum of link demands. Mathematics of Operations Research 13, 32–49.

Smith, H.K., Laporte, G., Harper, P.R., 2009. Locational analysis: highlights of growth to maturity. Journal of the Operational Research Society 60, S140–S148.

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

Snyder, L.V., 2011. Covering problems, in: Foundations of location analysis. Springer, pp. 109–135.

Suzuki, A., Drezner, Z., 1996. The p-center location problem in an area. Location science 4, 69–82.

Suzuki, A., Okabe, A., 1995. Using voronoi diagrams. drezner, z.(ed.): Facility location: A survey of applications and methods.

Taillard, É.D., 2003. Heuristic methods for large centroid clustering problems. Journal of heuristics 9, 51–73.

Tamir, A., 2001. The k-centrum multi-facility location problem. Discrete Applied Mathematics 109, 293–307.

Toregas, C., Swain, R., ReVelle, C., Bergman, L., 1971. The location of emergency service facilities. Operations research 19, 1363–1373.

Torra, V., 2004. Owa operators in data modeling and reidentification. IEEE Transactions on Fuzzy Systems 12, 652–660.

Vajda, S., 2009. Mathematical programming. Courier Corporation.

Valero-Franco, C., Rodríguez-Chía, A.M., Espejo, I., 2013. Accelerating convergence in minisum location problem with ℓ_p norms. Computers & operations research 40, 2770–2785.

Venkateshan, P., Mathur, K., 2015. A heuristic for the multisource weber problem with service level constraints. Transportation Science 49, 472–483.

Vergin, R.C., Rogers, J.D., 1967. An algorithm and computational procedure for locating economic facilities. Management Science 13, B–240.

Vohra, R., Hall, N., 1993. A probabilistic analysis of the maximal covering location problem. Discrete Applied Mathematics 43, 175–183.

Walker, W., 1974. Using the set-covering problem to assign fire companies to fire houses. Operations Research 22, 275–277.

Ward, J.E., Wendell, R.E., 1985. Using block norms for location modeling. Operations Research 33, 1074–1090.

Weber, A., 1909. Über den standort der industrien, tübingen theory of the location of industries.

Wei, H., Murray, A.T., Xiao, N., 2006. Solving the continuous space p-centre problem: planning application issues. IMA Journal of Management Mathematics 17, 413–425.

Wei, R., Murray, A.T., 2015. Continuous space maximal coverage: Insights, advances and challenges. Computers & operations research 62, 325–336.

Weiszfeld, E., 1937. Sur le point pour lequel la somme des distances de n points donnés est minimum. Tohoku Mathematical Journal, First Series 43, 355–386.

Wesolowsky, G.O., 1993. The weber problem: History and perspectives. Computers & Operations Research .

White, J.A., Case, K.E., 1974. On covering problems and the central facilities location problem. Geographical Analysis 6, 281–294.

Wu, L.Y., Zhang, X.S., hang, J.L., 2006. Capacitated facility location problem with general setup cost. Computers & Operations Research 33.

Xia, L., Yin, W., Xu, W., Xie, M., Shao, J., 2009. Efficient optimization of maximal covering location problems using extreme value estimation, in: Winter Simulation Conference, pp. 2395–2404.

Xu, J., Chiu, S.Y., Glover, F., 1999. Optimizing a ring-based private line telecommunication network using tabu search. Management Science 45, 330–345.

Xu, Z., 2005. An overview of methods for determining owa weights. International journal of intelligent systems 20, 843–865.

Yager, R.R., 1988. On ordered weighted averaging aggregation operators in multicriteria decisionmaking. IEEE Transactions on systems, Man, and Cybernetics 18, 183–190.

Yager, R.R., 2003. Toward a language for specifying summarizing statistics. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 33, 177–187.

Yager, R.R., Kacprzyk, J., 1997. The ordered weighted averaging operators: theory and applications. Norwell, MA: Kluwer.

Yager, R.R., Kacprzyk, J., Beliakov, G., 2011. Recent developments in the ordered weighted averaging operators: theory and practice. volume 265. Springer.

Zhou, L., Geng, N., Jiang, Z., Wang, X., 2020. Public hospital inpatient room allocation and patient scheduling considering equity. IEEE Transactions on Automation Science and Engineering 17, 1124–1139. doi:10.1109/TASE.2019.2942990.