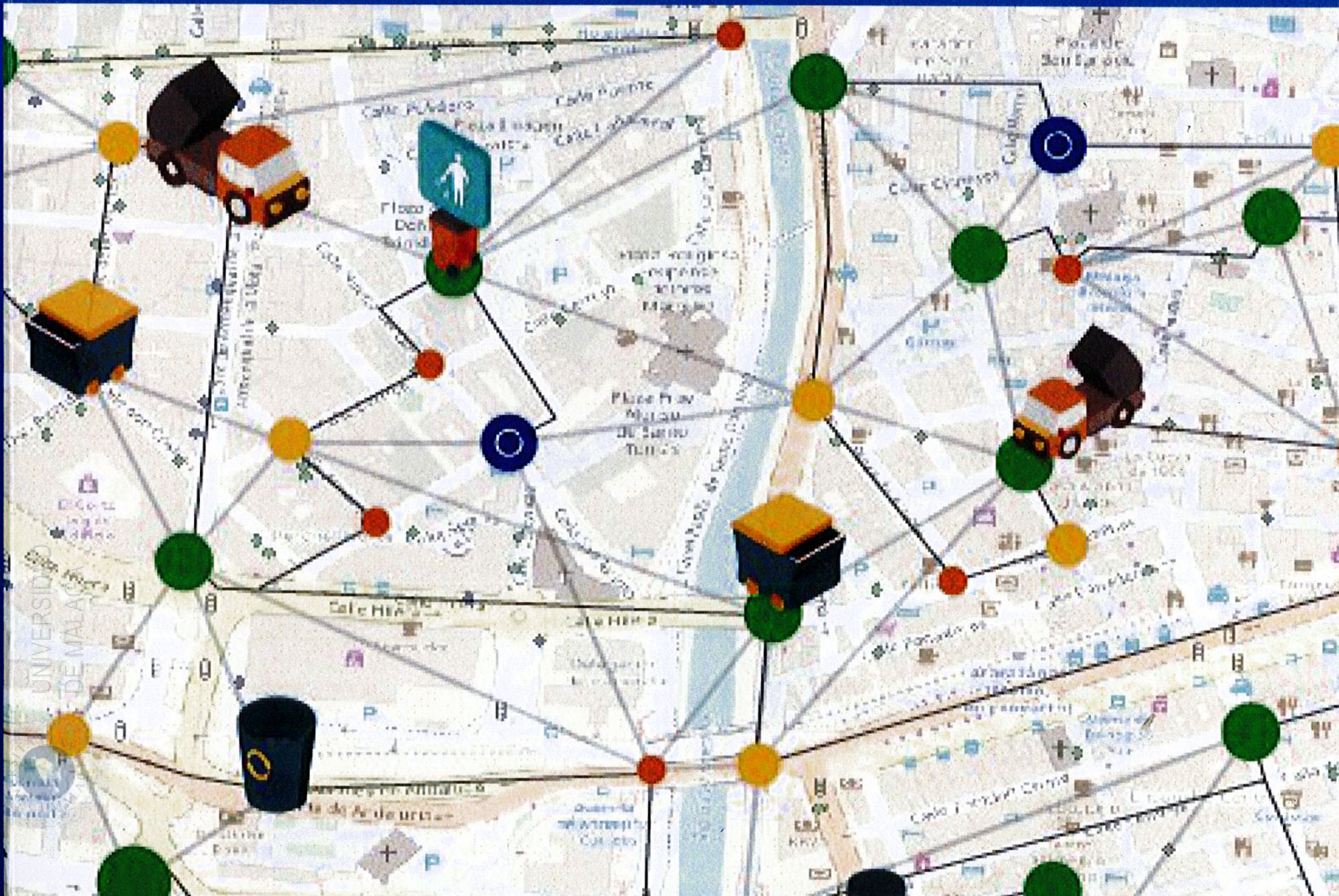
UNIVERSIDAD DE MÁLAGA

FACULTAD DE CIENCIAS ECONÓMICAS Y EMPRESARIALES

Departamento de Economía Aplicada(Matemáticas)

TESIS DOCTORAL | Laura Delgado Antequera | Málaga, Marzo 2018

Solving the Waste Collection Problem from a multiobjective perspective: new methodologies and case studies



Programa de Doctorado de Economía y Empresa

Facultad de Ciencias Económicas y Empresariales

UNIVERSIDAD
DE MÁLAGA

Solving the Waste Collection Problem from a Multiobjective Perspective: New Methodologies and Case Studies

Author

Laura Delgado Antequera

Tesis Doctoral, Marzo 2018

Supervisors

Dr. Rafael E. Caballero Fernández

Dr. Francisco Ruiz de la Rúa

AUTOR: Laura Delgado Antequera

http://orcid.org/0000-0003-3937-7422

EDITA: Publicaciones y Divulgación Científica. Universidad de Málaga

Acknowledgements

To my friends and family.

Thank you for being there.

For a better comprehension of the audience I am writing these lines to, let me express myself in Spanish.

GRACIAS

Parece sencillo dar las gracias, pero tengo tanto que agradecer que lo más difícil va a ser resumirlo de forma que llegue a todos.

En primer lugar quiero expresar mi eterno agradecimiento a mis **directores** de tesis, Dr. Rafael Enrique Caballero Fernández y Dr. Francisco Ruiz de la Rúa por el contínuo apoyo y haber creído en mí para realizar este proyecto. Gracias por la paciencia, la motivación y el inmenso conocimiento que me ha guiado a lo largo de esta grata experiencia como estudiante de doctorado. Gracias por todas esas conversaciones, casi paternales, que me han hecho crecer profesional y personalmente en estos últimos años.

Además, me gustaría agradecer a los miembros del tribunal de tesis: Dr. María Molinos, Dr. Fernando López, Dr. Mónica Hernádez, Dr. Ada Álvarez y Dr. Jesús Sánchez - Oro, sus comentarios, que han contribuido notablemente en la mejora de esta investigación y el manuscrito.

También, dar las gracias a mis compañeros del departamento de *Economía Aplicada (Matemáticas)* por la acogida, en especial a la Dra. Trinidad Gómez Nuñez y el Dr. Julián Molina Luque por estar siempre dispuestos a ofrecerme su ayuda y consejos. Gracias a Nancy Arellano, por nuestras eternas charlas de café y por enseñarme a escribir en LaTeX. Y finalmente, sin salir de la Universidad, a la Dra. Fátima Pérez García, por todas nuestras charlas y consejos que han facilitado los días más difíciles, tengo suerte de tenerte como compañera de despacho.

Le estoy también muy agradecida al Dr. Joaquín Pacheco y Dr. Alfredo García - Hernández Díaz por el apoyo, la disposición y la paciencia que han mostrado conmigo desde el principio de este proyecto. Además, mi más sincero agradecimiento al Dr. Manuel Laguna y a su familia, por acogerme en, lo que ya se denomina, la "Boulder Experience". Además del aprendizaje, sobre todo en el estudio de las metaheurísticas, mi estancia en Colorado me enriqueció como persona. Sumando a esta experiencia la suerte de compartirla con grandes personas e investigadores como Dr. Rafael Martí, Dra. Anna Martínez, Dra. Ana Dolores López y Dr. Jesús Sánchez - Oro.

Gracias al *Ministerio de Innovación y Ciencia* por aportar los fondos para el contrato predoctoral *(EEBB-I-16-11282)* asociado al proyecto *ECO2013-47129-C4-2-R* y a *Diputación Provincial de Málaga* por la colaboración en este trabajo.

A mi **familia deportiva**, gracias por alegrarme la vida con algo tan sano como el baloncesto y la buena compañía.

Gracias a mis **amigos**, sin nombrar a nadie. Aquellos con los que he compartido más de media vida, los de la carrera y erasmus y todos los que os habéis convertido en imprescindibles en estos últimos años, ya sea de estancia, de congresos o por casualidad. Quiero daros las gracias simplemente por estar ahí, que ya es mucho, en lo bueno y en lo malo e incluso en la distancia. Gracias por cada minuto que me regaláis, no hay nada tan importante para mí.

A toda mi **familia**, incluyendo los que ya no están con nosotros, y a Lourdes. Gracias por apoyarme y animarme a luchar por seguir.

Y, para terminar, gracias mamá, papá y ”Carlillos”. No existen palabras para describir lo afortunada que me siento de ser hija y hermana en esta familia. Gracias por hacer fácil lo difícil, por animarme en esta lucha continua contra mi curiosidad y mis ganas de hacerlo todo, gracias por la paciencia, gracias por la educación, la atención y el cariño, gracias por dejarme caer y ayudarme a levantarme mil veces,... gracias, gracias y GRACIAS.

Aquello que hace unos años empezó siendo todo un reto para mí hoy es una realidad. Y sin ninguna duda es gracias a vuestro apoyo y paciencia, por formar parte de mi vida y compartir conmigo lo mejor de vosotros.

GRACIAS.

RAFAEL CABALLERO FERNÁNDEZ, CATEDRÁTICO DE LA UNIVERSIDAD DE MÁLAGA (ESPAÑA) Y FRANCISCO RUIZ DE LA RUA, CATEDRÁTICO DE LA UNIVERSIDAD DE MÁLAGA (ESPAÑA).

INFORMAN:

Que Dª LAURA DELGADO ANTEQUERA, Lda. Matemáticas, ha realizado en la Universidad de Málaga y bajo nuestra dirección, el trabajo de investigación correspondiente a su Tesis Doctoral titulada:

"SOLVING THE WASTE COLLECTION PROBLEM FROM A MULTIOBJECTIVE PERSPECTIVE: NEW METHODOLOGIES"

Revisado el presente trabajo, estimamos que puede ser presentado al tribunal que ha de juzgarlo. Además, la contribución que avala la tesis, en cuanto a artículos: "An Interactive Biobjective Method for Solving a Waste Collection Problem", ha sido publicada en Mathematical Problems in Engineering (revista indexada en JCR), Volume 2016 (2016), Article ID 5278716, http://dx.doi.org/10.1155/2016/5278716.

La publicación aquí especificada no ha sido utilizada en tesis anteriores.

Y para que conste a efectos de lo establecido en lo regulado por la legislación vigente, autorizamos la presentación de esta Tesis codirigida en la Universidad de Málaga.

Málaga, 15 de diciembre de 2017

Fdo. RAFAEL CABALLERO FERNÁNDEZ

Fdo. FRANCISCO RUIZ DE LA RUA

CONTENTS

Resumen

En este trabajo se presenta una herramienta para analizar el Problema de la Recogida de Basura (WCP) en Málaga. El primer capítulo (Capítulo 1) desarrolla una breve introducción sobre el problema a tratar y define algunos conceptos metodológicos que serán utilizados en el resto del presente documento. A continuación, en el Capítulo 2, se hace un repaso de los trabajos previos que abordan el problema de la recogida de basura. Esta revisión incluye un análisis de las técnicas empleadas para resolver este tipo de problemas con uno y varios objetivos. En ella se aprecia un uso recurrente de Sistemas de Información Geográfica (GIS). En el Capítulo 3 se presenta la metodología desarrollada en este trabajo para abordar este tipo de problemas. Y, finalmente, la aplicación de la misma para analizar el problema real de Diputación de Málaga se presenta para concluir el documento.

En general, existe un interés creciente por el estudio de la gestión de residuos urbanos por parte de las administraciones locales en cualquier parte del mundo. Un manejo eficiente de la recogida y el transporte de los residuos conlleva una serie de beneficios tanto en el ámbito económico, como en el social y, también, en lo relacionado con el medio ambiente. Diferentes tareas se incluyen en la gestión de residuos. Entre ellas, se encuentra el estudio del tratamiento de los residuos sólidos, así como el diseño de distintas opciones para reutilizar los residuos reciclables. El tratamiento de residuos es un hecho que todo hogar y negocio necesita gestionar, para manejar el depósito de objectos y sustancias usadas de forma segura y eficiente.

Diseñar un sistema eficiente para gestionar los residuos no es tarea fácil. Se han de tener en cuenta diferentes factores que pueden intervenir, de forma más o menos relevante, en el proceso dependiendo del servicio a realizar. En un área, los residuos proceden de actividades industriales, viviendas o comercios. Por tanto, es necesario controlar diversos aspectos como la generación, el almacenamiento, la recolección, el transporte o transferencia, el procesamiento y el depósito de los residuos. A su vez, se han de respetar ciertos temas de la salud pública, económicos, estéticos, de ingeniería y otros en relación al medio ambiente.

En particular, en el marco del desarrollo sostenible, la gestión de residuos a nivel municipal adquiere cierta relevancia. En este ámbito, los gestores necesitan diseñar sistemas sostenibles que, a su vez, sean económicamente admisibles, socialmente aceptables y eficientes a nivel ambiental. Para obtener un sistema de tales características no existe un método único que asegure la mejor calidad, por lo que cada elemento del problema debe ser analizado cuidadosamente.

Hay trabajos que analizan distintos sistemas de gestión de residuos, otros estudian las razones del fracaso de estos sistemas y otros aportan unas pautas a seguir para diseñar sistemas eficientes. Sin embargo, este estudio se centra en el diseño del sistema de recogida de basura, distribuida en los distintos municipios de la provincia de Málaga a los que da servicio Diputación de Málaga.

A nivel económico, la recolección de basura y su transporte constituyen un alto porcentaje del coste de gestión de residuos. Los últimos estudios revelan que, en España, los gastos de recogida y tratamiento de los residuos superan el 40% de los ingresos que provienen de los impuestos o tasas asignadas a cubrir dicho servicio. Por tanto, el uso de un buen proceso de decisión conllevaría múltiples beneficios para las administraciones de dicho servicio. Este hecho, entre otros, ha impulsado el interés y los esfuerzos invertidos en el diseño de tal procedimiento.

Es objeto de este trabajo el desarrollar una metodología que ayude a encontrar un buen sistema de recogida para el WCP en Málaga. El continuo crecimiento de la población en este área, provoca un aumento en la cantidad de residuos sólidos generados. En consecuencia, resulta interesante conocer las distintas opciones, y analizar las modificaciones que habría que implementar para mejorar distintos aspectos de este servicio. Por tanto, se debe aplicar un modelo de criterios múltiples, que recoja todas las alternativas posibles de forma que los gestores obtengan una amplia visión de las posibilidades.

Al tratarse de un problema real, se ha analizado la solución que utiliza actualmente la entidad para la recogida de residuos. Para este problema, a nivel provincial, los datos y parámetros son aportados por la Diputación provincial de Málaga. Estos datos contienen la localización de los distintos contenedores, plantas de transformación y vertederos, la cual se contrasta utilizando un software de Sistema de Información Geográfica (SIG). Para ello, también es preciso cargar las distintas capas de carreteras (direcciones, sentido, límite de velocidad, giros, etc.) que nos sirven como base para construir la matriz de distancias y tiempos. Además, para el diseño de las rutas, se dispone de una estimación de la cantidad de residuos sólidos acumulada en cada punto mensualmente, lo que permite estimar una media de kilogramos de basura recogidos en cada contenedor visitado.

Habitualmente, resolver problemas reales implica considerar la optimización de más de un objetivo simultáneamente. Por ejemplo, es común encontrar trabajos donde el problema de la recogida de basura contempla la determinación de la periodicidad del servicio en cada punto y establecer las ventanas de tiempo para su recogida, de forma que pueden ser modelados como Problemas Periódicos de Rutas con Ventanas de Tiempo (PVRPTW). Sin embargo, la cantidad de residuos generados nos ha llevado a analizar un sistema de recogida diario, por lo que la periodicidad no está incluida en este estudio. Las administraciones locales tienen fijados unos horarios para ejecutar este servicio, enmarcado en unas ventanas de tiempo. Estas ventanas de tiempo vienen dadas por la duración de la jornada laboral, por lo que serán incorporadas en el modelo como una restricción en la duración de las rutas.

Normalmente, los problemas de recogida de basura (WCP) son abordados como problemas de optimización. En estos casos, una solución para un problema de optimización ha de especificar los valores de las variables de decisión, que a su vez determinan los valores de las funciones objetivo a optimizar. Por lo general, el conjunto de todas las decisiones posibles está limitado por una serie de restricciones. La solución será optima si produce el mejor valor de la función objetivo. Sin embargo, este concepto no se puede extrapolar a problemas con múltiples criterios, pues resulta poco probable encontrar una solución capaz de optimizar todos los objetivos simultáneamente, por lo que la resolución de un problema multiobjetivo no consiste en aportar una única solución, sino en construir un conjunto de soluciones eficientes, o de Pareto, denominado frontera de Pareto.

Así, para este problema, es preciso definir una estrategia para obtener dicho conjunto. Dada las dimensiones y la complejidad del problema, se presentan una serie de técnicas metaheurísticas que se apoyan en una adaptación para problemas multiobjetivo de GRASP (Resende and Ribeiro, 2016), en combinación con Path Relinking y Variable Neighborhood Search (VNS). El algoritmo GRASP se emplea con el fin de determinar una primera aproximación de la frontera eficiente.

La fase de construcción de este GRASP combina otras heurísticas de inserción para obtener una buena aproximación de la solución. A continuación, la solución obtenida es mejorada con un proceso de búsqueda local que utiliza los operadores comunes 2 - opt y los intercambios OR, propios de los problemas de optimización de rutas.

Tres alternativas diferentes se desarrollan en la Sección 3 de este trabajo para obtener dicha aproximación. Las dos primeras se basan en el trabajo de Martí et al. (2015). En ellas, se van almacenando las soluciones no dominadas obtenidas al alternar la construcción de soluciones enfocadas a optimizar uno u otro objetivo de forma ordenada (GRASP Puro Ordenado Multiobjetivo) o de forma aletoria

(GRASP Puro Aleatorio Multiobjetivo). La otra aproximación consiste en utilizar el metaheurístico GRASP definido para minimizar la función de logro definida por Wierzbicki (Wierzbicki, 1980) para un número de combinaciones de pesos.

A continuación, las aproximaciones de las fronteras son analizadas con la adaptación de Path Relinking a un problema multiobjetivo, en el que se busca enlazar dos soluciones no dominadas mediante la transformación de una otra en otra, tomando como guía uno de los objetivos implicados.

La otra alternativa almacena las soluciones potencialmente eficientes a lo largo de la primera etapa GRASP. A continuación, para cada una de ellas, se analiza entre qué dos pares de soluciones no dominadas se encuentra y se aplica la búsqueda VNS para minimizar la distancia de la solución a un punto de referencia definido por los mejores valores del par.

Combinando las distintas propuestas, se obtienen hasta 6 métodos distintos. Los resultados de estas técnicas son evaluados con instancias de la literatura. En el trabajo se incluyen tablas que muestran el éxito de la aproximación generada por la idea de Wierzbicki y Path Relinking. Una vez obtenido un conjunto de soluciones no dominadas, se hace necesario un método que ayude al decisor a decidir cuál de ellas es la más adecuada en base a sus preferencias. Esto nos lleva a requerir el desarrollo de una estrategia cuya solución, única, se adapte a las preferencias de los decisores. Para ello, existen una serie de metodologías que ayudan al decisor en la obtención de la solución más ajustada a sus necesidades dentro del conjunto de soluciones posibles. Estas técnicas son los denominados *métodos interactivos*, que destacan por su utilidad. Mientras otras técnicas multiobjetivo incorporan la información al principio o al final del proceso, en los métodos interactivos se repite un algoritmo iterativo en cada paso de forma que la información se va agregando a lo largo del proceso de resolución, guiando así al decisor hacia aquellas soluciones que satisfagan sus intereses. En este ámbito, se desarrolla un método de la familia de NAUTILUS. Se trata de métodos que no precisan de trade - offs y se apoyan en algunos comportamientos psicológicos del ser humano para diseñar el proceso interactivo que guiará la búsqueda.

Para evitar gastos de cómputo extensos, esta metodología se apoya en una pre - computación de los elementos de la frontera eficiente. Entonces, el método empieza en el peor escenario posible y va avanzando, en dirección a un punto de referencia, de forma que los valores de todos los objetivos mejoran continuamente. En el método propuesto, *R - NAUTILUS*, se integran distintas opciones: el decisor tiene información continua sobre el rango de valores que puede alcanzar cada objetivo, así como de la evolución del conjunto de soluciones alcanzables. Además, se muestra una gráfica del histórico de valores que han tomado las cotas de los valores de las funciones. En cualquier momento, el decisor puede parar el proceso y retroceder, definir un nuevo punto de referencia o limitar el valor superior o inferior de las funciones o dibujar la solución eficiente que se encuentra en la dirección de búsqueda.

Finalmente, el problema multiobjetivo de la recogida de basura en Málaga, descrito anteriormente, se subdivide en distintos subproblemas según la comarca y tipo de camión que proporciona el servicio. Además, para abarcar el mayor rango de opciones posibles, se han definido 4 objetivos: minimizar el coste, equilibrar las rutas, minimizar la diferencia entre la ruta más larga y más corta, en duración, y minimizar el número de rutas realizadas. Los datos proporcionados nos han permitido, con el software NEVA, generar la matriz de tiempos y distancias de un contenedor a otro, teniendo en cuenta el sistema de redes de carreteras en la provincia de Málaga.

La metodología anterior ha sido integrada en una interfaz gráfica de usuario que incorpora inicialmente la visualización de los contenedores en las distintas regiones y permite, al concluir, analizar el sistema de rutas obtenido en la solución elegida. Para ello, se ha utilizado lenguaje de programación Java 8, en un entorno Eclipse y la extensión de ArcGIS SDK Java for Developers.

El desarrollo del trabajo se ha apoyado en 3 pilares fundamentales, como

son la recogida y análisis de los datos, el estudio de la optimización de problemas de rutas y el diseño de un método interactivo para agilizar el proceso de decisión. Además, distintas contribuciones se obtienen de este estudio. Por un lado, se proponen distintas metologías para generar una frontera eficiente de problemas de rutas con múltiples objetivos, destacando los resultados de aquella que aplica la función de logro definida por Wierzbicki (Wierzbicki, 1980). Por otro lado, se presenta un método interactivo, enmarcado en la filosofía de NAUTILUS (Miettinen et al., 2010) para guiar al decisor a través de una aproximación de la frontera eficiente, hacia la solución más similar a sus preferencias. Finalmente, se lleva a cabo la implementación de la metodología propuesta en un entorno computacional que facilita la toma de decisiones a las entidades implicadas en el problema real. Para ello se diseña una interfaz gráfica que incorpora la visualización de los elementos del problema y la interpretación de las soluciones resultantes. De esta forma, la obtención de una solución que pueda ser llevada a la práctica con éxito supone el diseño de un nuevo sistema interactivo, junto con una estrategia metaheurística. Además, el análisis de los resultados en base a un software conectado a Sistemas de Información Geográfica (GIS), facilita la comunicación entre el analista y el gerente a lo largo del proceso de toma de decisiones.

SUMMARY

This work introduces a tool to analyze the *Waste Collection Problem (WCP)* in Málaga. In general, Waste Management is a critial issue to be studied by local administrations all over the world. Multiple factors need to be considered when dealing with this type of problems, so they have to be taken into consideration to model an efficient and effective Waste Management System.

The development of the present study is based on 3 essential pilars, such as the collection and data analysis, the study of multiobjective optimization *Vehicle Routing Problem* and the design of an interactive decision making process.

This document is structured in four chapters. First, a general introduction to the problem and some definitions are included in Chapter 1. Later, Chapter 2 describes previous works on Waste Management and, in particular, MultiObjective Waste Collection. Usually, Waste Collection problems are regarded as optimization problems, so that an optimal solution is to be found. However, in *MultiObjective Problems (MOP)* it is unusual to find a unique solution that optimizes all the objectives at the same time. Then, the aim of these problems, instead of finding an optimal solution, is to find a set of *Pareto solutions* or *efficient solutions* called *Pareto front, Pareto Set or Efficient Set*. In our problem, this set will contain the alternatives of route systems to run the waste collection service.

It is necessary to define an estrategy to solve this *MOP*. Since *Vehicle Routing Problems (VRPs)* are NP - hard, non exact methods are usually applied to obtain an approximation of the Pareto front, so different heuristics and metaheuristics are developed to generate a good approach of it. Therefore, three different multiobjective *GRASP* strategies are defined here in order to obtain a first approximation to the Pareto front and, then, *Path Relinking and Variable Neighborhood Search (VNS)* are applied to improve it. Two of these *GRASP* estrategies are based on the idea proposed in Martí et al. (2015) and the third one uses *GRASP* as a single - objective optimizer to minimize Wierzbicki's Achievement Scalarizing Function (Wierzbicki, 1980) for different weights combinations. The performances of these strategies are compared with problems from the literature and the best alternative turned to be the latter in combination with *Path Relinking*.

Given the approximation of the Pareto front, it is difficult to decide which one is the most appropriate according to the Decision Maker's (DM) preferences. Then, an interactive method is introduced next, in order to guide the decision making process. This method, called *R-NAUTILUS*, is based on *NAUTILUS* phylosophy and incorporates different features in order to facilitate the interaction.

Finally, the methodology developed, which is detailed in Chapter 3, is applied to the MultiObjective Waste Collection Problem in a southern city of Spain. To ease the process, the design of a Graphical User Interface is implemented using Java language programming and its package for ArcGIS. It includes the interactive method process, and it enables the DM to visualize the selected solution and analyze its different components.

Several contributions derive from this work such as methodologies to generate a Pareto efficient front, highlighting those results that apply an Achievement Scalarizing Function proposed by Wierzbicki, an interactive method which helps the DM in the analysis of the different alternatives available to manage the service, and the design of a Graphical User Interface that guides the decision making process towards the selection of the most preferred solution of a real Waste Collection Problem.

CHAPTER 1

INTRODUCTION

The concept of *waste management* encompasses the process of treating solid wastes and providing different solutions for recycling items to be reused. It also contemplates an analysis of how garbage can be used as a valuable resource. Waste management is something that each and every household and business owner in the world needs, to control the disposal of the products and substances that have been used, in a safe and efficient manner.

Designing an efficient system for waste management is not an easy task. It must take into consideration different factors that may become more or less relevant depending on the type of service provided. In particular, solid waste might be generated from industrial, residential and commercial activities in a given area. Its management covers several aspects that control the generation, storage, collection, transport or transfer, processing and disposal of solid waste materials in a way that best addresses the range of public health, conservation, economics, aesthetic, engineering and other environmental considerations.

Municipal solid waste management in the framework of sustainable development requires especial attention. In this situation, managers need to create sustainable systems that are economically affordable, socially acceptable, and environmentally effective. A unique method that provides the best solution for the municipal solid waste management does not generally exist, so each characteristic of the method must be carefully evaluated. Some works analyze different *Waste Management Systems*, including reasons of failure or determining the steps to follow in order to manage the waste collection. For instance, Pires et al. (2011) provide an analysis of the different waste collection system techniques applied within 15 European countries. This review classifies system analysis techniques into: *(i)* System engineering models, including cost - benefits, forecasting, optimization and integrated systems; and *(ii)* System assessment tools, such as management information, Decision Support Systems (DSS), scenario development, material flow, risk and environmental assessment, among others. This analysis states future lines of research due to the most recent legislation declared by the European Commision, which focuses on new definitions of waste that result in the study of selection of technologies in order to improve protection of human health and environment by promoting the reuse and recycling.

A simple summary of what should be achieved when planning a *Waste Management System* is given by Viotti et al. (2003):

> ”Successful planning and management of waste collection is primarly focused on reducing costs and environmental impacts related to such waste collection, along with improvement of user satisfaction, considering aesthetic and sanitary conditions.”

An analysis of the main causes for waste management failure is given in Guerrero et al. (2013). To support their argument, data from different locations are considered and different indicators that contemplate legislation, collection efficiency, sophistication of waste collection or environmental awareness. In particular, solid waste management is a critical issue of environmental hygiene and it must be incorporated into the environmental planning applied by local administrations. Emissions from the collection and transportation of solid materials, and the advanced treatments required, need longer distances, which implies an increment on the energy used to take materials to and from a facility. This implies that emissions of vehicles can cancel out the environmental gains obtained when applying different separation techniques like recycling. This is one of the facts that justify the design of efficient planning tools to control the transportation resulting from solid waste.

Economically speaking, waste collection and its transportation constitutes a large fraction of the total cost for the municipal solid waste management worldwide. Last studies reveal that the expenses required to cover the service of collecting and treating the solid waste generated in Spain are a 40% higher than the income from population's waste management related taxes.

Hence, local administrations could obtain multiple benefits from using a good decision making procedure that enabled them to improve this and other services. This fact, among other reasons, has encouraged an increasing interest and efforts invested to design such a procedure, that will provide a more efficient performance.

Usually, *Waste Collection Problems (WCP)* are treated as optimization problems. In general, a *solution* to an optimization problem specifies the values of the decision variables, $\mathbf{x}$, and therefore also the value of its objective function $f(\mathbf{x})$, that is, the function to be optimized. Usually, the set of all possible decisions $\mathbf{x}$ to be made is limited by a series of constraints. A solution that satisfies all constraints is said to be *feasible* and, furthermore, it is considered *optimal* if it gives the best objective function value.

In concrete, this work aims to provide a methodology to solve the *Waste Collection Problem (WCP)* in the region of Málaga (Spain). The increasing population in this area, and so the amount of waste generated, make it interesting to study the conditions that could be modified to improve the service if applying another waste collection system. Then, multiple criteria have been considered in order to cover the widest set of possibilities. Data was kindly provided by *Diputación de Málaga*, including the number of containers and their location, the number and type of vehicles available and the routing cost, facilities coordinates and the amount of waste collected per month and per municipality. Using GIS permits a visualization of the distribution of the containers into a map. *Diputación* *de Málaga* covers the service to more than 5,000 containers distributed through the region. Different issues arise when dealing with a problem of such a dimension.

Therefore, the **main objective of this work** is to analyze and solve the real problem of solid waste collection in the southern Spanish region of Málaga considering multiple objectives simultaneously. This task encompasses multiple items that can be subdivided into achieving the following **specific objectives**:

Obj. 1 *To analyze the current Waste Collection System in Málaga.*

Different techniques have been applied to analyze the dataset provided by *Diputación de Málaga*. First, the location of the containers provides a general vision of their distribution into four different areas, each of which has a depot associated. This data also contains additional information about the current waste collection system, such as the number of vehicles available or the total monthly amount of waste generated by municipality. It has allowed to perform several estimations to incorporate into the modelization process, such as the amount of waste expected to be collected at each location or the number of vehicles required to satisfy the population's request. This analysis is included in Section 4.1.

Obj. 2 *To define a realistic model for the Waste Collection Problem in Málaga.*

This model must include vehicle's characteristics, employees workload, employees shifts, road specifications such as speed, forbiden turns, etc. and, furthermore, satisfy citizen's needs. Besides, incorporating multiple objectives permits defining a more realistic model. Since no special requirements have been set by local administrations, some reasonable objectives have been considered instead, in order to provide the largest set of solutions that enables a complete exploration of the options. It is difficult to formulate a model with these characteristics, so based on the characteristics the general scheme of a *MultiObjective Capacitated Vehicle Routing Problem (MOCVRP)* has been considered as a reference, and other constraints have been incorporated. The description of this model is included in Section 4.1.

Obj. 3 *To study, design and implement an efficient, effective and fast method to solve the* MultiObjective Waste Collection Problem.

The methodology proposed must take into consideration multiple objectives at the same time and include labor, social and economical constraints derived from each particular problem. The dimension of the real problem, as well as the difficulty of modelling some of its constraints, make it unfeasible to use an exact method. Then, based on literature results, a new method should be designed, capable of generating good feasible solutions of the model at a reduced computational cost. To achieve that goal, a general methodology has been developed for the *MultiObjective Capacitated Vehicle Routing Problem (MOCVRP)*. Its description is detailed in Section 3.2. Additionally, Section 4.2 details the solutions obtained for the real *MOWCP*.

Obj. 4 *To define an appropriate interactive method which helps the waste manager in the decision making process*.

Literature review (Section 2) reveals a large number of Decision Support Systems (DSS) designed for the Waste Management Problem. In general, these studies focus on the location of containers and facilities and incorporate a Geographical Information System (GIS) to display the chosen solution or to modify it somehow. However, there is a lack of interactive methods that permits the DM to analyze the solutions available and guide the exploration to select the most preferred one, after an iterative process. Then, because of the simplicity in the information exchange, an interactive non - trade off method, which is inspired on the well known *NAUTILUS* family, has been developed in this work. Nevertheless, some modifications have been included into this method in order to deal with this kind of problem. Further details on the method developed can be found at Section 3.3.

Obj. 5 *To design and implement a decision interface to display the strengths and weaknesses of the proposed solutions*.

When dealing with real problems, it is important to transmit the results in a friendly - format. This is why it is common to use GIS in *VRP*. This kind of software permits an easier comprehension and a better interaction with the decision maker. Hence, in order to facilitate the decision making process, a friendly - environment should be designed integrating GIS and the decision making procedure. It must allow the DM to analyze, in depth, the options and be guided into the selection of the solution that best matches his / her preferences. An example of the application of the Graphical User Interface (GUI) designed in this work is included in Section 4.3.

In this case, the distribution and the information obtained from the dataset, allows to split the problem into subproblems, according to the subregion where the bins are allocated and the type of vehicle assigned to provide the service. Note that trucks are differenciated by the loading mechanism as well as the maximum capacity of waste they can store. Hence, because of the distribution of the street bins and the limitation on the capacity of the trucks, the modelization of this problem uses the *Capacitated Vehicle Routing Problem* model. Then, solutions are defined as a system of routes which are determined by the sequence of bins to serve.

Toth and Vigo (2014) provide an accurate verbal definition for the single - objective *Vehicle Routing problem (VRP)*:

> "Given a set of transportation requests and a fleet of vehicles, the objective is to determine a set of vehicle routes to perform all transportation requests with the given fleet at minimum cost; in particular, decide which vehicle handles which requests in which sequence so that all vehicle routes can be feasibly executed."

The given definition considers cost as the unique objective to be optimized. Nevertheless, many real - life optimization problems can hardly be considered as properly formulated without taking into account their multiple objective nature

(Stewart et al., 2008). In particular, multiple factors interfere in the design of a model that reflects better a real waste collection problem.

Frequently, solving real problems involves considering the optimization of more than one objectives at the same time. For instance, it is common to find studies where the waste collection problem includes taking care, also, of the determination of the periodicity of the service at each point and establishing the most appropiate time windows to run the service, so that they can be modelled as *Periodic Vehicle Routing Problems with Time Windows (PVRPTW)*. However, in our context, due to the amount of waste generated and population satisfaction, a daily service is studied so that the analysis of periodicity will not be included in this work. Local administration has a fixed schedule that determines the time windows, so it can be incorporated into the model as a constraint that limits route's duration to a specific length.

Therefore, the presence of multicriteria theory is also palpable when dealing with waste management, as one can read from 2.1.1. In order to provide a wide overview of the different options available for the current *WCP* in Málaga, saving costs and introducing some improvements on employees' conditions, as well as the quality of the service, are some of the aspects included in the modelization of the *WCP* in the region of Málaga. The distribution of the containers or bins and the assigment of vehicles to depot is already implemented by the administrations, in such a way that it satisfies the service within a specific area. Then, the aim is to reduce the routing cost, while improving employees' conditions and estimating the cost associated to the possibility of running a daily service.

As a parenthetical remark, some previous definitions and classical concepts on *Multicriteria theory* are included next. In general lines, multiobjective programming consists of the study of modelling and solving problems that integrate more than one objective. Let S be the feasible set for a *Multiobjective Optimization Problem (MOP)* where, without loss of generality, we wish to minimize simultaneously $k \geq 2$ objective functions, such as $f_i : S \to \mathbb{R}$. Then, any MOP can be formulated as follows:

$$\min \{f_1(\mathbf{x}), \cdots, f_k(\mathbf{x})\} \tag{1.1}$$

$$\text{subject to: } \mathbf{x} \in S \tag{1.2}$$

When optimizing more than one objective function at the same time, it is not likely to find a unique optimal solution, since these objectives are usually in conflict and it is not possible to improve one of their values without impairing, at least, one of the others. Then, *Pareto* or *efficient* solution arises as a generalization of the concept of optimal solution.

Definition: Given $\mathbf{z^1}$ and $\mathbf{z^2}$, solutions in the objective space, it is said that $\mathbf{z^1}$ *dominates* $\mathbf{z^2}$ if for each component i, $z_i^1 \leq z_i^2$ and $z_j^1 < z_j^2$ for some $j \in \{1, \ldots, k\}$. Otherwise, if $\mathbf{z^1}$ and $\mathbf{z^2}$ do not dominate each other, then it is said that $\mathbf{z^1}$ and $\mathbf{z^2}$ are *(mutually) nondominated*. Therefore, a decision vector $\mathbf{x}^* \in S$ is *Pareto Optimal* if there does not exist another decision vector $\mathbf{x} \in S$ such that its image, $f(\mathbf{x})$, dominates $f(\mathbf{x}^*)$.

The *Pareto front* is the image of all the Pareto solutions that belong to the *Pareto Set*. These solutions vary from the ideal objective vector to the nadir objective vector. The *ideal* objective vector is formed by the best values that each objective function can achieve in the Pareto Set, i.e. $z^* = (z_1, \cdots, z_k)^T$, where $z_i = \min_{x \in S} f_i(\mathbf{x})$; whereas the worst values achievable for each objective function in the Pareto Set are the components of the *nadir* objective vector.

In order to manage a whole set of efficient solutions instead of a single optimal solution, a decision making process will have to be applied to help the *Decision Maker (DM)* to find the solution which best fits his / her preferences. Different decision making procedures have been defined according to the generation of the Pareto Set (Miettinen, 2008).

Usually, *MOPs* are converted into a problem with a single objective function or a family of such problems, that can be solved using recognized single -

objective optimizers, by a simple procedure called *scalarization*. In particular, *a priori methods* take into account preferences given in advance and the method tries to find a Pareto solution as close as possible to them. *Value function method* (Keeney and Raiffa, 1993), *lexicographic ordering* (Fishburn, 1974) or *goal programming* (Romero, 1991), are examples of this kind of methods. On the other hand, the analyst provides a set of Pareto solutions to the DM, who will have to decide which of them is the most appropiate in *a posteriori methods*. In this context, two classical methods can be used: *weighting* method (Gass and Saaty, 1955; Zadeh, 1963) and ε - *constraint* method (Chankong and Haimes, 1983; Haimes et al., 1971). The weighting method re - formulates the problem into a single objective problem, defining the objective as a linear combination of the objective functions, where the scalar coefficients are the *weights*. These weights can be assigned following a specific pattern (Chankong and Haimes, 1983; Das and Dennis, 1997) or determined according to the preference information specified by the DM (Podinovskii, 1994; Roy and Mousseau, 1996). On the other hand, the ε - *constraint* method selects one objective to optimize and the others are introduced in the model as additional bounded constraints, so that several single objective problems have to be solved using an appropiate method. In *a priori* and *a posteriori* methods, the DM participates by expressing preference relations before or after the process.

Also, we can find another group of methods like *interactive methods* which allow the DM to update preferences during the process by an iterative algorithm, that is repeated until (s)he reaches a satisfactory solution. At each iteration, a solution is obtained and then, the DM specifies or adjusts his / her preferences information according to the result obtained. These methods will be described in more detail in Section 3.3.

Different classifications have been proposed to characterize Interactive methods. For instance, possible classifications are obtained when considering the information asked to the DM and the internal analysis in the solution process.

Classification depending on the information required. It is based on the questions formulated to the DM. To include the DM's preferences at each iteration of the decision process, the DM has some alternatives. Some approaches select the solution after a sequence of comparisons. Some of these techniques obtain the final solution by a process of pairwise comparisons or considering several objective vectors. One of the common approaches of this type is described in (Steuer and Choo, 1983), *Tchebycheff method*. Another group of interactive methods are the *interactive trade - off based methods*, where the *trade-off* defines the ratio that the DM is willing to risk to improve the value of one objective function when some other gets worse. Sometimes, the DM is provided with *objective trade - offs*, which indicate the real ratios when moving from one efficient point to another, and therefore (s)he needs to evaluate them and decide where to move in the next iteration. Other methods ask the DM to provide marginal rates of substitution between two objectives, such that they stay in the same indifference curve of the DM's utility function. Estrategies involving trade - offs are designed to solve convex and continuous problems, because their calculation implies regular conditions of derivative theory. However, trade - offs can be approximated using a finite quotient of increments.

The DM is asked to order his/her preferences or provide desired values in *especification levels methods*. For instance, in *Goal Programming interactive methods* (Romero, 1991), (s)he can define the goals of the problems in terms of the values and providing the aspiration levels. Another type of methods are based on the reference point scheme (Wierzbicki, 1982). Now, the DM starts by establishing a reference point or a vector of desired values. At each iteration, the DM is provided with the solution that is most similar to the reference point. This interactive perspective includes the normalization of the objectives and, to obtain more efficient points, the reference points can be perturbed at each iteration. More details about these methods can be found in Section 3.3 and in Miettinen (1999). Alternatively, in *classification* -

based methods the DM arranges the functions into classes, indicating what objectives should be improved, which ones already have an acceptable value and which ones are allowed to impair. Different methodologies correspond to this idea of classification, such as STOM (Nakayama, 1995) or NIMBUS (Miettinen and Mäkelä, 2000). Though maybe not explicitly, also these methods compute the values for the set of efficient solutions and so they involve the evaluation of trade - offs. Also, notice that *referent point - based* methods and *classification - based* methods require the definition of an achievement scalarizing function to obtain the next iteration point. The previous methods need of a continuous exchange of information between the analyst and the DM. On the other hand, *NAUTILUS* method (Miettinen et al., 2010) was proposed in such a way that it avoids to ask for trade - offs by starting from the worst possible scenario. This allows every objective function to improve at each iteration. This method established the standards for *interactive non trade - off methods*.

Classification by the internal optimization process It is based on how the information provided by the DM is used to determine the next iteration. Then, in terms of this information, we find *reduction of the feasible region* methods, *line search* methods, *reduction of the weighting space* methods, methods *based on multipliers* or *reference point or achievement function* methods, as the one introduced in Wierzbicki (1980). *Line search* methods are used on linear programming algorithms, whereas the *reduction of the weighting space* is applied in the Tchebycheff interactive procedure. After generating a filtered set of solutions, the DM selects one. Centered on the weights of this solution, the procedure continues by obtaining another filtered set and it repeats the previous steps until a satisfactory solution is found.

When the standards for the model of the *MOWCP* are defined, i.e. determine the objectives, incorporate the constraints and analyze the dataset, it is essential to design an appropriate methodology to solve the problem and, at the same time, to ease the flow of information implementing it into a friendly - environment so it can be used as a tool to guide the DM to his / her most preferred solution. Then, to implement the interactive - interface, there is a need of designing an algorithm able to generate the Pareto set. In this occasion we focus on the design of a methodology that solves *MultiObjective Vehicle Routing Problems (MOVRP)* with limited capacity, so that it can be applied to a wider range of problems. Nevertheless, in this work it is used to solve the *MOWCP*, as detailed in Section 4.

The family of *Vehicle Routing Problems* have been proved to be NP - hard (Lenstra and Rinnooy Kan, 1981), so the application of an exact method, if it does exist, might incur into a vast computational effort. This setback is solved using approximated methods or *metaheuristics*.

The term *heuristics* is used to define a technique, method or procedure that is intelligent enough to provide a solution to a task which does not derive from a formal analysis but from a wide knowledge on the subject. In particular, this term is used to define an efficient procedure that tries to provide solutions to a problem, in terms of the quality of the solution obtained and the required resources.

This concept of heuristic is generalized as *metaheuristic*, which comprehends those strategies designed to construct algorithms capable of escaping from local optima and perform a robust search of the solution space. To escape from local optima and access unexplored areas, metaheuristics allow moving to a worse solution or even to an unfeasible solution. Moreover, since these are approximated algorithms, there is no guarantee of finding the optimal solution, and so a stopping criterion must be defined.

As well as heuristics, metaheuristics are developped associated with the particular requirements of a problem, although a general scheme defines each of these techniques.

They can be classified depending on different characteristics such as: *Is it* *based on populations or a single - point?*, *Is the objective function dynamic or static?*, *Is the algorithm inspired in a natural process?*, *How many neighborhood structures does it use?* or *Does it use memory?*. For instance, *ant colony optimization* is a population based and bio - inpired algorithm. In general, a population based algorithm starts from an initial set of solutions (*population*) and, applying different operators, new populations are generated. *Genetic algorithms*, *memetic algorithms*, *scatter search* or *path relinking* are also population based methods. Also, some algorithms study the incorporation of memory such as *Tabu Search* (Glover, 1989, 1990), in order to avoid cyclics, whereas *Greedy Random Adaptive Search Procedure* (GRASP) is a memoryless metaheuristic which includes an explorative local search, as it occurs with *Variable Neighborhood Search* (Mladenovic and Hansen, 1997) or *Guided Local Search*.

Just a short information has been included here, but further information about metaheuristics and their applications are included in Glover and Kochenberger (2003) and Gendreau and Potvin (2010), among others multiple references.

The complexity and computational efforts required to solve single - objective *VRP* increase when considering multiple criteria, for which metaheuristics have also been widely studied to determine a good approximation of the Pareto front.

Then, a competitive metaheuristic needs to be designed in order to obtain a good approximation of the Pareto front for the *MOVRP*. Later, when a set of nondominated solutions is generated, they cannot be ordered unless some information is provided by a human Decision Maker (DM). Therefore, an interactive method will help with the information exchange between the analyst and the DM. Here, one supposes that the DM, in this context, waste managers, is fully aware of the necessities of the crew, their customers and service requirements.

Some advantages of using an interactive method include the active participation of the DM in the process, which enables him / her to control the search and encourage his / her confidence on the choice of the final solution. Another advantage is that the DM does not need to have a global preference structure, so (s)he can learn along the process and decide from a more realistic level. It facilites the interaction with the analyst and allows the incorporation, or modification, of the information obtained at each step. Most decision processes include two phases based on *learn* and *decide*.

In terms of quality and cost, the optimization of the Waste Management Service can only be achieved by using advanced decision support tools, which contemplate the different components of this kind of problems. Then, taking advantage of the natural way of introducing preferences into the process, we propose a methodology inspired on a well known non trading - off interactive method that belongs to the family of *NAUTILUS* (Miettinen, 2008; Miettinen et al., 2010).

Different properties, which will be discussed in Section 3.3, make *NAUTILUS* the ideal method to guide our decision problem.

Finally, this methodology is included in the design of a user - interface, so the procedure and the information obtained is simplified and displayed to facilitate the interpretation of the results. Therefore, the incorporation of a visual tool, to show the performance of the solution selected, is required. In this context, Geographical Information Systems (GIS) have become a relevant option in *VRP* and, in particular, for location strategies in Waste Management as indicated in Section 2.

The real problem, as here formulated, consists of providing an efficient design of feasible and optimum routes system to collect the generated solid waste in the region of Málaga. To obtain the desired solution, we develop a methodology that combines metaheuristic strategies and interactive methods. Metaheuristics permit to generate a set of approximated solutions in a short computational time; whereas interactive methods are included with the aim of guiding the decision maker to the best solution according to his/her preferences.

Along the process, the results will be displayed on a friendly environment based on a Geographical Information System (GIS). This additional tool will ease the interpretation and the decision making process.

The following chapters contain a more detailed information about this project. It begins with a wide analysis on waste management previous studies in Section 2, including single - objective and multi - objective perspectives and the description of the multiple Decision Support Systems (DSS) that have been proposed for this type of problem. Then, the methodology developed to be applied to our real problem is detailed in Section 3, including a description of the metaheuristics used to generate an approximation of the Pareto front, as well as the interactive method implemented for the decision making process. To conclude, the performance of this methodology is analyzed at Section 4 when applied to the real problem in Málaga.

Thus, the contributions of this work can be summed up into different items. On the one hand, the design of an efficient algorithm that generates a good approximation of the Pareto optimal set. In this case, the hybridization of GRASP and Path Relinking has been used to obtain an approximation of the Pareto optimal set applying two different schemes: one alternates the optimization of every objective and maintains the nondominated solutions visited, whereas the other one optimizes the resulting scalarizing achievement function proposed in Wierzbicki (1980). However, other metaheuristics can also be implemented within the same schemes.

Next, the DM is guided through this set using a non trade - off interactive method based on NAUTILUS phylosophy. The method here proposed incorporates specifications according to the waste collection problem, such as the management of a discrete Pareto front. These interactions between the analyst and the DM require the implementation of an interface displaying the information of interest for the DM which enables to assess the performance of each solution. All of these constitute a methodology able to find the most preferred solution for the multiobjective large dimension *Waste Collection Problem* in Málaga.

CHAPTER 2

STATE OF THE ART

The following chapter contains a summary of the multiple approaches proposed to deal with the *Waste Management Problem*. To set up the standards, a general definition of the *Waste Collection Problem (WCP)* is given in first place, including a large variety of examples based on the different constraints considered and methodologies developed when minimizing costs. Then, multiple criteria are considered in order to obtain more realistic models. As it happens in single - objective problems, metaheuristics, instead of exact methods, is the most common strategy applied in order to obtain "good" approximated solutions when dealing with this type of problems. Different metaheuristics based on generating some sort of population, such as *genetic algorithms* or *ant colony system*, are implemented including the corresponding modifications that enable the generation of good approximations of the Pareto front. To conclude this chapter, an analysis of *Decisions Support Systems (DSS)* designed to deal with Waste Management Problems is presented. It settles the corresponding steps for the implementation of *DSS* and highlights the lack of proper interactive multiobjective methods applied to solve this kind of problems.

All along these relevant works, it is important to highlight the role that the integration of Geographical Information Systems (GIS) plays in the resolution or in the decision process.

2.1 Waste Collection Problem

Nowadays, there exists a correlation between the growing population and the amount of waste generated, so *Solid Waste Management (SWM)* has become one of the most interesting themes for public decision makers. The increase in the generation of waste in modern economy is closely linked to the growth of production and consumption as well as to natural processes determining the rate at which the product lifespan goes into decline.

Two big perspectives are observed: *regional* and *municipal*. The main difference between them lies on the fact that *regional* Solid Waste Management is resposible for organizing the process from a macro - perspective, i.e. designing the network and location of facilites such as transformation plants or landfills; whereas the *municipal* Solid Waste Management is in charge of the transportation of the waste generated at each location to its corresponding depot.

More precisely, this work focuses on the *Waste Collection Problem (WCP)*, which consists of designing a system of vehicle routes to service a set of bins geographically distributed. Note that some of these bins might be concentrated on the same location. Every route must start and end at one depot, with the waste dumped at the treatement plant or landfill and every bin must be visited before it overflows, with a minimum frequency depending on the season and type of waste. Finally, hard constraints take into account that the amount of waste collected cannot exceed the capacity of the vehicle and the duration of each route must respect driver's shift length. However, in practice, other operational constraints arise such as recyclable materials management, dimension of vehicles to traverse certain streets, locations to be visited within an specific time windows, route balance, fleet size, lateral restrictions . . . Then, there is a natural subdivision when studying the *WCP*: *(i)* Determination of the frequency to visit each location and *(ii)* define the optimum set of routes to service all the corresponding locations everyday.

A formal definition (Toth and Vigo, 2014) states that:

> ”A *Waste Collection Vehicle Routing Problem* typically consists on a fleet of vehicles, stops, disposal facilities, a depot and a number of collection bins or collection points. A vehicle starts and ends at the same depot. Usually, the waste collection problem has been solved as an *Arc Routing Problem*, where the exact location of every customer is not needed.”

Based on the characteristics defined for each particular problem, *Node Routing* or *Arc Routing* will be appropiate to handle it. Usually, *Node Routing (VRP)* is applied when a large number of containers are located in a few different locations and *Arc Routing (ARP)* if waste is deposited in small containers distributed almost continuosly along the street. Due to the point of interest of this work, we will focus on *Node Routing*. For further information about the application of *Capacitated Arc Routing (CARP)* to solve *WCP*, see, for instance: Male and Liebman (1978), Hanafi et al. (1999),Constantino et al. (2015), Corberán and Laporte (2015) or Cortinhal et al. (2016). For the first time, Marks and Liebman (1970) highlighted some research lines that related those perspectives of the Solid Waste Management Problem that could be addressed in the field of Operational Research. This fact awared the sanitary departments of big cities like New York or Washington D.C., which started several studies focused on developing operational research strategies in order to improve their services. For instance, Beltrami and Bodin (1974) provided a methodology to solve the waste collection problem in New York City (U.S.A), taking into account the feasible combinations of days to service the set of containers at a predefined frequency. Two different approaches were proposed in order to minimize the overall routing cost, while visiting every container assigned each day. One of them is based on the idea of cluster first route second, and the other one optimizes the routes first and apply a giant tour technique afterwards. Since then, a large number of *Waste Collection Problems (WCP)* have been solved applying

Operational Research techniques, highlighting the application of metaheuristics, and including different realistic perspectives such as studying the frequency of the service, the type of vehicles, the location of additional containers or landfills, considering a large problem with multiple depots, etc.

Angelelli and Speranza (2002) deal with hygiene requirements by incorporating the study of frequency to the *WCP*. They considered a periodic node routing approach *(PVRP)*. First, close locations sharing the same service requirements are grouped into macro - points, to plan the schedule and then a *Tabu Search* technique, whose neighborhoods are defined by the shift operator, is implemented. Also, Baptista et al. (2002) developed a heuristic to maximize the benefits obtained from a periodical recycling paper collection.

An interesting study (Maniezzo and Roffilli, 2008) defines a methodology to transform *CARP* into *CVRP*. They minimize the overall distance travelled by a fleet of trucks, subject to time windows and allowing multiple trips. They also include a penalty function in the process if the demand exceeds truck's capacity. The transformation proposed consists of mapping each arc with a node, whose corresponding demand is the cummulated demand along the arc. Then, they use a multi - start heuristic combined with *Variable Neighborhood Search* to solve the resulting *CVRP*. The multi - start algorithm constructs an initial solution from scratch and improves it with *Tabu Search* technique. Then, different neighborhoods are explored using the following procedures:

- Shorten: It re - orders the sequence of points to be collected by the route.
- Add: Given a route, and a stop which is not visited by this route, it creates a new route including this stop.
- Drop: Given a route, it constructs a new route without one of the stops.
- Paste: Concatenate each route of the system in one route to be processed by shorten.

- Cut: Consider a route servicing several points, possibly exceeding the capacity constraint, then it determines the partition of the routes to obtain feasibility.

- Switch: If a vertex is visited more the than once in a route, it calculates the cost if all those subroutes having this vertex as endpoint are travelled in the opposite direction.

- Postop: It applies Paste, Switch, Cut and Shorten operators.

A different approach is given in Bautista et al. (2008), transforming an *ARP* into a *VRP* with a partition of its vertex into clusters [MCARPTC]. Then, it is modeled as a *General Vehicle Routing Problem (GVRP)* determining the vertex selection by solving a Location Routing Problem. The constructive algorithm combines nearest neighbor and nearest insertion, which is improved with substitutions, reinsertions and exchange operators. This method is combined with the *Ant Colony Optimization* technique to optimize a real *WCP*. A final analysis of the results shows a reduction on the night acoustic contamination, total time and cost to complete the service.

The study of complex waste management systems, in particular sitting waste management and disposal facilities and optimising waste collection and its transportation, has been a preferential field of *Geographical Information Systems (GIS)* applications (Chalkias and Lasaridi, 2009). Due to technology evolution and the complexity of spatial information to be handled, *GIS* modelling is becoming a strong support tool to manage the information interchanged with the decision maker in the process of decision making.

A location problem is faced in Ghose et al. (2006) incorporating their algorithm into the NETWORK package of ArcGIS. Here, three different types of bins and vehicles need to be located, so they develop a GIS user - interface implementing the algorithm described in Sharma (1974) to obtain the shortest path. A large - scale location problem is also solved in Li and He (2009), who integrate an intelligent *Ant Colony System* into GIS to solve a site selection problem. Multiple simulations are analyzed to test the performance of this strategy, and the final result is applied to solve a real problem in China.

Recently, Erfani et al. (2017) have proposed another model in order to improve bins distribution and vehicle routing for the *Municipal Solid Waste Problem*. This methodology is divided into three stages: *(i)* Collect the required information about the current status of Solid Waste Collection System, road maps, district population, etc; *(ii)* Process the data and import the results to spatial database and *(iii)* develop a network analysis model, using ESRI ArcGIS network analysis extension, in order to solve the location - allocation and *VRPs* and obtain optimal storage bin locations and tours.

However, GIS not only has been used for location allocation problems but also in routes optimization. In this context, Tavares et al. (2009) incorporate fuel consumption to the use of 3D route modelling within ArcGIS Network Analysis (NA) and Karadimas et al. (2007) add *Ant Colony System* metaheuristic to optimize routes and provide the most cost - effective itinerary to follow. Recently, Nguyen-Trong et al. (2017) have provided a solution to model Vietnam waste collection system with a succesfull cost reduction. In the process, data is uploaded into ArcMap and a simple heuristic, inspired on the Clarke and Wright saving algorithm (Clarke and Wright, 1964) in combination with an agent - based model, are integrated into a dynamic model.

As we can see, additional packages of the well known GIS software*ArcGIS*, have been used and, sometimes, even improved or modified. Further information is available on the latest reviews on Waste Collection and Management such as Belien et al. (2012); Marshall and Farahbakhsh (2013) and Bing et al. (2016).

2.1.1 Multiobjective Waste Collection Problem

In recent years, an increasing concern about the emissions to the environment due to the collection and transportation of waste can be appreciated. Then, a huge effort is invested on balancing the cost associated to the collection of recycling material and the vehicle's emissions during the service. This implies that efficient planning tools, and so multicriteria models, are needed to control the transportation resulting from separation and collection of waste.

Hence, to provide a more realistic approach to waste management, different multicriteria models have been proposed to obtain the best solution, according to the decision maker's requirements. Due to the dimension and complexity of real *WCP*, metaheuristics have become the main point of interest to find a good solution approach in the shortest possible time. In this context, Nuortio et al. (2006) successfully use Guided Variable Neighborhood Thresholding to solve a large *WCP* defining frequency and subject to time windows. Previously, a construction - improvement method was introduced in Tung and Pinnoi (2000) to improve the existing manual solution in Hanoi (Vietnam). The construction modifies *Salomon's I1 insertion heuristic* (Solomon, 1987) for the *VRPTW* to minimize the cost and the number of vehicles currently used. For each unrouted customer, the best position to be inserted is estimated by minimizing a linear combination of the objective functions that contemplates the distance and the delayed time for the next customer. To decide which customer will be included next, we must identify which is the one that maximizes the saving. This saving cost is calculated as the difference between the distance to the depot and the cost previously obtained. Finally, an improvement phase is launched using the inter - route move 2 -opt^* (Potvin and Rousseau, 1995), to reduce the number of vehicles used, and intra - route OR - opt (Or, 1976).

To minimize collection time, distance travelled and man - effort, a strategy is designed in Chalkias and Lasaridi (2009). It proposes the replacement of an existing large number of small size bins (120 and 240 L) with a reduced number of larger bins (1100 L). Hence, a model is developed into an extension of the GIS ArcGIS 9.2 (Network Analyst) to reallocate the waste bins. It consists of three steps: *(i)* Upload the spatial database of the study area; *(ii)* Use GIS spatial analysis functions to reallocate the bins to service and finally *(iii)* Construct routes to optimize time, distance, fuel consumption and gas emissions. At the last step, an alteration of the Djistra's algorithm is defined to optimize the path, in order to incorporate real problem restrictions such as oneway roads, prohibited turns, demand at intersections and along the roads or side - street constraints. The final output is an optimal solution in terms of distance or time criteria, instead of dealing with a multiobjective problem.

Different genetic algorithms have been developed to tackle *MOWCP*. For instance, Ombuki-berman et al. (2007) optimize the total distance and the number of vehicles by introducing a genetic algorithm based on Beasley's approach. Then each chromosome represents a network and is given by an array of integers, with no limitations that indicates the beginning or the end of each route. On the process, the fitness is evaluated via the weighting sum method and, also, using the Pareto Ranking procedure to select the elements that will generate the next population. Finally, after applying a crossover based on the best - cost route, the chromosome is splitted into capacitated clusters that will form the route system. Another metaheuristic, that belongs to the family of population metaheuristics, has recently been developed in Xue and Cao (2016). With the aim of minimizing the total cost, the accident cost, the accident risk and the exposure to the public, they define a multiobjective *Ant Colony Optimization* method coupled with min - max model and Djistra's algorithm. This methodology also takes advantage of ESRI ArcGIS tool to draw the resulting route system.

To minimize labor, operation and transport costs, Arribas et al. (2010) divide the resolution process of a *MultiObjective Waste Collection Problem (MOWCP)* into three phases, optimizing one objective at a time. Then, to minimize cost related to labor, operation and transportation, it first employs the regret function and local search to construct a cluster of the set of containers to be collected. This cluster is obtained attending the vehicle maximum capacity and the service schedule. Then, *VRP* is solved for each cluster using *Tabu Search* using ESRI ArcGIS and

ESRI Network Analysis. In fact, this GIS incorporates the urban road network characteristics to design the collection routes. Finally, an exact *Branch and Bound* algorithm is applied to minimize the number of vehicles, considering that each cluster should be served by only one vehicle.

An important example is the one given in Kim et al. (2006), because of its proximity to reality. Here, the *Vehicle Routing Problem with Time Windows (VRPTW)* is considered as the basic model. Then, other constraints are taken into account, like multiple disposal trips and driver's lunch break, with a unique depot. Four objectives are optimized in order to reduce costs and improve labor conditions: *(i)* number of vehicles, *(ii)* total travelling time, *(iii)* route compactness and *(iv)* workload balance.

- To minimize *(i)* and *(ii)*, a modification of Solomon's insertion algorithm is developed. To do so, the model includes two different capacities: one is given by the maximum capacity a vehicle can handle; and the other one is related with the especific characteristics of the route, such as the limited number of stops, lifts and volume or weight a driver can lift per day. This is implemented into GIS street network data and the shortest path is given by Djistra's algorithm.

- A capacitated clustering algorithm is designed to maximize *(iii)* and *(iv)* based on the k - means clustering method. They define centroids of centroids and use them to sort the stops by the distance to each of them, assigning in the first place the farthest one to its nearest cluster. After this assignation, the travel time is estimated by solving a Travel Salesman problem, and a simple improvement algorithm is launched if any overlap is found. However, due to the multiple constraints, there is no guarantee of finding a non - overlapping route system. Also, a metric is defined to quantify route's compactness.

This method is applied to solve a commercial WCP in North America. However, the main contribution of this research is the proposal of a new *Vehicle Routing Problem with Time Window (VRPTW)* benchmark.

As a consequence of the completeness of this model, the problem proposed has been taken as a reference to assess the quality of different methods such as Benjamin and Beasley (2010, 2013). Now, *Tabu Search* and *Variable Neighborhood Search (VNS)* are applied separately and in combination, using the *Tabu Search* within the neighbor's search. They also incorporate a reduction on the search space by determining a number of nearest - in time - nodes for each unrouted customer. In Benjamin and Beasley (2013), the metaheuristic proposed in Benjamin and Beasley (2010) is improved, comparing its solution with Hemmelmayr et al. (2009) taking into account the crew's rest period and time windows for waste collection. They reduce the computational cost by pre - evaluating facilities insertions in a disposal facility positioning procedure. Also, *Ant Colony Optimization* is implemented within a GIS to solve a *MOWCP* in Karadimas et al. (2007).

Applying *VNS* as well, Hemmelmayr et al. (2013) solve a separate *WCP* in Austria, with the aim of estimating frequencies and considering intermediate facilities. They provide a large analysis on *WCP* modelled *Periodic Vehicle Routing Problem (PVRP)*. Later, in Hemmelmayr et al. (2014), they improve their own method by providing a better design of a collection system taking into account vehicle routing and bin allocation, at the same time, and trying to balance the trade - off between the service frequency over a planning period and the number of bins that can be placed there. This method incorporates, into the *VNS*, dynamic programming to insert intermediate facilities and uses an acceptance criteria similar to the one used for *Simulated Annealing*. In addition to the route balance objective considered in this work, other aspects are contemplated in this work such as multiple waste type case, the number of bins allocated on a specific area and the capacity or volume of bins or the cost associated to a service. Route balance objective has also been considered in López-Sánchez et al. (2017) to solve a biobjective *WCP*.

Another definition of route balance is included in a biobjective *WCP* in Gómez et al. (2009), whose methodology was improved in Gómez et al. (2015). Both solve the problem defining a *Tabu Search* within *MultiObjective Adaptive Memory Procedure (MOAMP)* technique (Caballero et al., 2003). *MOAMP* is a metaheuristic algorithm designed to solve MultiObjective Combinatorial Optimization problems which is based on *Tabu Search*. It emphasizes neighborhood search over mechanisms for evolving a population of solutions. This methodology guides the user on what to do but not how to do it. Its phylosophy is based on two main ideas:

1. Efficient points are ”close” to each other in the solution space. This fact forces the application of local search around nondominated solutions.

2. Compromise points are ”close” to the ideal point in the objective function value space. It requires to define a function, g, that measures the distance of the objective function values of the current point, S, and the ideal objective function values given by $({f_1}^{min}, {f_2}^{min})$. Then, given a weighted factor $\lambda \in [0, 1]$, they optimize the function:

$$g(S) = \max \left\{ \lambda \cdot \frac{f_1(S) - {f_1}^{min}}{{f_1}^{max} - {f_1}^{min}}, (1 - \lambda) \cdot \frac{f_2(S) - {f_2}^{min}}{{f_2}^{max} - {f_2}^{min}} \right\} \tag{2.1}$$

In this work, a real - problem is solved considering two objectives: *(i)* to minimize the transportation cost and *(ii)* to improve the service level by minimizing the waste accumulated. The methodology is subdivided into three phases as detailed in Algorithm 1.

Algorithm 1 Summary of MOAMP adaptation to bi - objective WCP

PHASE 1:

Set $\wp = \emptyset$

S = *Quality generator function*

S = TabuSearch(f_1, S)

S = TabuSearch(f_2, S)

Update $\wp$ with nondominated solutions visited.

PHASE 2:

repeat

  Select λ using a uniform distribution, U[0,1]

  S = TabuSearch(g_λ, S)

  Update $\wp$ with non dominated solutions visited

until $\wp$ does not change for maxPhase iterations

PHASE 3:

repeat

  Explore the neighborhodd of each $S \in \wp$ that has not been explored.

  Update $\wp$ with non dominated solutions visited.

  Improve new non dominated solutions.

until $\wp$ does not change

Thus, an initial approach of the Pareto front, $\wp$, is obtained at PHASE 1. First, the best solution to minimize the waste accumulated is obtained with the *Quality generator function*. Then, connected Tabu Searches are launched towards each objective function, while updating the efficient frontier and obtaining the optimum value for each objective. The procedure continues applying Tabu Searches on PHASE 2, evaluating the merit of each solution $S \in \wp$ as indicated in (2.1). These searches last until no improvement has been found for a prefixed number of iterations, *maxPhase*. Finally, PHASE 3 focuses the search on the neighborhood of each solution of the approximated Pareto front set, $\wp$. Now, the goal is not to find a particular solution to make a move, but to explore the neighborhoods seeking new non dominated solutions.

Recently, Ferreira et al. (2015) proposes a methodology to solve a *MOWCP* considering to minimize total distance, to maximize the amount of waste collected, to maximize the amount of waste collected by kilometre, to maximize the number of ecopoints visited, to minimize the number of vehicles and to maximize the number of priority points collected. To achieve that goal, they propose a three - modules DSS for a real - world reclycling problem. In the first module they solve the routing optimization problem, modeling it as the *Team Orienteering Problem* with capacity constraints and time windows. This model is solved using a *genetic algorithm*. Then, a second module estimates the waste generated. In the last module, a set of indicators are defined to evaluate the performance of the solution at each objective. Then, a tool called *beSMART* (beS, 2017) is used to weight the importance of each performance indicator so that the most preferable solution is found. Several studies have designed *Decision Support Systems (DSS)* as a tool to ease the procedure of finding or selecting the most adeccuate solution for their particular preferences. These techniques, and designs, are discussed in the next section (Section 3.3).

2.1.2 Decision Support Systems for Multi-objective WCP

Different aspects related with cost, techology, environmental and health concern, limited landfill space or political and social aspects must be taken into consideration when defining a Waste Management System. The increasing interest of designing these systems efficiently have encouraged the development of **Decision Support Systems** (DSS) to help on the decision process. Adenso-Díaz et al. (2005) define a DSS as a guiding tool for the DM that is capable of providing the corresponding information, or suggestions, about what would happen if a series of decision are taken. The integration of GIS into DSS can assist in the analysis and comparison of different waste management and collection alternatives.

Some Multi Criteria Decision Making (MCDM) techniques have been applied within a DSS to guide the DM to the most preferred solution. A DSS is developed to provide a solution procedure to a case study from Finland (Hokkanen and Salminen, 1997). This problem considers different indicators related with environmental issues, characterized by the imprecision, so the authors employs the *ELECTRE III* decision - aid (Roy, 1991) because of its stability to variations in data and parameters. The objectives are derived from the set of indicators given by the group of decision makers. The weights assigned to each criteria were also obtained from different comitees of the municipality with the possibility of incorporating new criteria.

For the first time, MacDonald (1996) introduce a problem structuring tool for the planning of solid waste collection, recycling and incineration system to solve a real recycling problem in Philadelphia (U.S.A). To select the location of potential facilities and design the route followed by the fleet of trucks, the proposed method makes use of mathematical programming models to suggest scenarios, adding a *technology screening tool*, that contemplates the preferences or constraints; and a model base, that could be run to suggest a plan for the waste flow. In this case, GIS enables to specify distance or time variables. Then, in order to optimize the economic cost, the net energy used, water used and labor needs, seven steps define the process:

1. Determine the type of facilities that will not be considered.
2. Choose a technology within a given set of alternatives available. Here, the DSS deletes the policies that will not be considered, so that the set of techonologies is divided into *acceptable* and *non - acceptable*.
3. Characterization of the waste generation area with GIS.
4. Route planning is obtained, what will provide cost information and the required resources.
5. An scenario is suggested when the DM chooses the criteria of greatest concern for developing a plan.

6. Given a set of goals and constraints specified by the DM, a mathematical programming technique helps on the organization of the recycling and trash system.

Later, Chang et al. (1997) incorporate statistical and optimization analysis to a DSS. In order to decide the most appropiate allocation problem of waste stream for reclycling and incineration, they propose a multi - level interface. At each step, the DM selects a topic within the areas of interest: incinerator cost, manpower or equipment of collection team. This results on a map, supported by diagrams and numbers that sums up the corresponding information, which reveals real world issues of the selected alternative.

This work is improved in Chang and Wei (1999) with the incorporation of GIS, which calls an external multiobjective programming model base. Here, three objectives are taken into consideration: *(i)* to maximize the population served by recycling drop - off stations, *(ii)* to minimize the walking distance from household to recycling drop - off stations and *(iii)* to minimize the total driven distance during the vehicle routing. In the selection process, five scenarios are defined and five performance indices, such as service ration, utilization rate, average walking distance, recycling rate and routing ratio, are used to evaluate the scenario. In this case, a *genetic algorithm* is implemented as an external tool for the routing optimization.

Some works consider a set of indicators to evaluate and compare the different alternatives or scenarios and then apply *MultiCriteria Decision Analysis* techniques to guide the DM to the most prefered solution. For example, Stanisavljevic et al. (2015) define different scenarios using STAN (subSTance flow ANalysis), proposed in Cencic and Rechberger (2008). Then, different indicators are considered in order to evaluate these scenarios and compare them, taking into account different criteria that include to protect humans and environment, conserve resources and design a sustainable waste management. In this context, also Chifari et al. (2016) evaluate waste flow using indicators.

In addition to previous examples, more DSS have been developed as a tool for the decision making process when considering multiple criteria in waste management. Due to the changing policies related with the treatment of the different types of solid waste, the main area of application has been the allocation of facilities and the analysis of different aspects of the waste recycling management.

The main contribution of Simonetto and Borenstein (2007) is *SCOLDSS*, an operational management DSS that considers the solid waste processing capacity of sorting units. Four stages define this methodology designed to solve a multi - depot multi - trips waste management problem: *(1)* identify the problem and structure it, *(2)* develop a formal model to represent the problem, *(3)* implement an appropiate method to obtain the solution and *(4)* validate the method through different tests. The nature of the problem suggests the incoporation of, some sort of, simulation process to estimate the solid waste processing capacity at sorting units. Here, the software *ARENA* was integrated as a module in the DSS to obtain an estimation of the waste demand that each sorting unit is able to process at a certain day of work. *SCOLDSS* corresponds to the user interface block. In this case, a friendly environment interface is developed to upload the required data and run the different models in order to obtain the solid waste collection operational scenarios, including a map to show the computed vehicle routes. Note that this is not a formal multi - objective problem, so the interaction with the DM lies on selecting the date on which the separate collection planning will be made and the operating waste sorting units. Given the corresponding data, the simulation determines the amount of waste to be collected at each point. Finally, the user proceeds by executing the vehicle allocation, generating the optimal collection routes and, also, reports the results.

A friendly DSS is designed in Santos et al. (2008) for multiple - vehicle routing problems which are defined as *CVRP* to solve the *WCP* in Coimbra (Portugal). Another DSS is developed by Brebbia et al. (2000), including different factors such as waste generation forecasting, vehicle routing and economical analysis to assist the users during the planning phase of separate waste collection.

Khan and Samadder (2014) provide a rich review on DSS and available GIS softwares and give a new idea for the allocation of bins and landfills combining *MultiCriteria Decision Analysis (MCDA)* and GIS to minimize the total system cost. However, many authors support the idea of structuring a DSS on three linked blocks (Bani et al., 2009): database management, model base management and user interface. For instance, Haastrup et al. (1998) introduce three different models to deal with the *Model Management System* block. First, a model for the scenario construction that generates a given number (K) of alternatives is determined by combinations of locations of disposal and treatment facilities and the asignation of the users to each facility obtained after solving a single - objective knapsack problem. Then, four models to evaluate the scenarios are also implemented, including site risk, environmental impact, cost and transportation risk. And finally, a model for multicriteria analysis is proposed, NAIADE (Novel Approach to Imprecise Assessment and Decision Environments), to determine a ranking of the K alternatives. This analysis, based on aggregation of pairwise comparison, permits to rank the alternatives according to the set of evaluation criteria or according to decision maker's preferences and also provides information of the distances of the positions of the interest groups.

Gallardo et al. (2015) introduce the steps to follow to guide the companies in the design of an efficient waste collection plannning, depending on the available data. This method, supported by GIS, was successfully applied to real cases in Castellón (Spain). It first defines a number of waste fractions and then stablish an storage level: door - to - door, kerbside, drop - off sites, establishment, green point. Finally, GIS is used to locate the storage points.

In Xi et al. (2010) three different scenarios are analyzed to deal with the long - term planning of solid waste management in Beijing, China. A model is developed to minimize the system cost subject to a set of constraints which include: capacity balance, mass balance, waste residue, facilities expansion and the possitive sign of the decision variables. An interval mathematical programming is used for the model. Then, applying some transformation, the exact model is solved in LINGO to obtain the alternatives. Once the different schemes are obtained, MCDA techniques are applied in order to rank the alternatives or scenario. Two common multiple - attribute utility methods are applied: simple weight addition (2.2) and weighted product (2.3).

$$U_j = \sum_i w_j x_{ij} \forall j \in 1, ..k \tag{2.2}$$

$$U_j = \prod_i x_{ij}^{w_j} \forall j \in 1, ..k \tag{2.3}$$

where w_j is the importance of the i - th attribute and x_{ij} the normalized impact matrix. Also, a third method, *TOPSIS* is used. This method was introduced in Hwang and Yoon (1981) based on the relatively straight - forward assumption that each attribute takes a monotonically increasing utility. Then, preferences are ordered and selected by the alternatives with the minimum distance from the best solution and maximum distance to the worst.

A recent work involving multiple stakeholders is Soltani et al. (2017). Here, different uncertainty assessment methods are implemented to analyze a case study in Vancouver (BC, Canada). Due to the uncertainty on several aspects of the process, a fuzzy - AHP is introduced to be applied on the environmental and economic criteria. It is also combined with game theory, in order to mitigate the uncertainty derived from group decision making and allow to model interactions that helps stakeholders to, also, make decisions based on other's actions.

Hanine et al. (2017) propose another DSS to select the appropiate estrategy and reduce the impact when selecting the location for landfill or industrial waste in Casablanca (Morocco). The process applies previous techniques, creating an efficient combination between the spatial factors: OLAP / GIS and the DM.

1. Define the problem using the analytical tools of OLAP / GIS. This step contributes to generate the set of candidate locations for the landfill industrial waste (LIW).

2. Incorporating DM opinions determine the location selection criteria. Then, triangular fuzzy numbers are introduced to stablish the pairwise comparison into the tradicional AHP method, to avoid uncertainty in the calculation of attributes' weight.

3. To classify the different alternatives, TOPSIS is applied, which is considered one of the most efficient MCDM methods. TOPSIS includes the normalization of weights and the generation of a final ranking that considers a relative distance to the best and worst solutions.

Thus, to the best of our knowledge, there is a lack of research activities about the application of multiobjective interactive techniques to deal with the solid waste collection problem and, in particular, those that contemplate the optimization of route planning. Actually, in a previous work (Delgado-Antequera et al., 2016), an interactive methodology for a biobjective *WCP* that aims to optimize the overall distance and route balance is proposed. The approximated Pareto front is obtained by continuosly applying *GRASP* and the ε- constrained approach. In this case, the interactive process considers a reference point and going through a sequence of decisions, that narrow the decision space step by step, based on the objective values, until a reduced list of options is achieved and the most preferred solution is selected. Since then, this methodology has evolved, and a non trade - off interactive method is included in this work. Multiple objectives can also be handled with the new methodology. As it will be detailed in Section 3.3, this method permits a wide exploration of the possibilities provided by a hybrid metaheuristic. Additionally, this alternative might also be applied to guide the DM to his / her most preferred solution, in other multiobjective problems.

CHAPTER 3

METHODOLOGY

The *Waste Collection Problem (WCP)* has been widely studied from several points of view. As described in Section 2, multiple algorithms have provided, in combination with Geographical Information Systems (GIS), solutions to real problems. However, it is always a challenge to implement an algorithm to handle multiple objectives at the same time and guiding the decision making process to the most preferred solution.

In this work, different schemes are developed to obtain a good approximation of the Pareto front for any *MultiObjective Capacitated Vehicle Routing Problems (MOCVRPs)*. However, all of these alternatives can be divided in two big stages. First, an approximation of the Pareto front is generated using a multi - objective *Greedy Randomized Adaptive Search Procedure (GRASP)* heuristic and next improved, either by applying *Path Relinking* or *Variable Neighborhood Search*. This set of nondominated solutions contains the different alternatives to be considered for the interactive process taking place in the second stage. Note that, in spite of *GRASP* and *Path Relinking* or *VNS*, any other metaheuristic can be used to generate this approximation.

The present chapter contains, in Section 3.1, a summary of the applications of *GRASP* heuristic and its combination with *Path Relinking* to solve *Vehicle Routing Problems (VRPs)*, paying special attention to the resolution of those problems considering multiple criteria.

Next, based on these descriptions, different approaches are developed to obtain the best approximation of the Pareto front for the *MultiObjective Waste Collection Problem (MOWCP)*, which is modelled as a *Capacitated Vehicle Routing Problem (CVRP)*, so they are suitable for any *MultiObjective Capacitated Vehicle Routing Problem (MOCVRP)*.

These approaches, applying *GRASP* heuristic, make use of *VNS* in the improvement phase in two different perspectives, both considering a single objective. In addition to the usual *VNS* that minimizes one of the objective functions at a time, a multiobjective version is proposed to minimize the distance to a reference point. The last algorithm will be denoted as VNS_{ref}. Then, *Path Relinking* or VNS_{ref} will be defined to improve a first approach of the Pareto front obtained with *GRASP*. The different strategies derived from the combination of *GRASP* with *Path Relinking* or VNS_{ref} are explained in Section 3.2.

In order to test the performance of the algorithms proposed, a random sample of instances has been taken from the literature. In particular, a total of 25 examples have been considered from Christofides and Eilon (1969), Christofides et al. (1979) and Uchoa et al. (2017). These instances are usually considered as a reference to compare the results obtained by an algorithm when minimizing the total distance of a *VRP*. For the sake of simplicity, the performance of these strategies will be analyzed for a bi - objetive problem that contemplates the optimization of the total distance and the minimization of the longest route to control the routes balance. Including this second objective is important to solve our particular *MOWCP*, so a parenthesis is also included to sum up different works that have tackle the *MOCVRP* with route balance.

Finally, at the second stage of this work detailed at Section 3.3, an interactive strategy is developed and implemented into an interface that permits visual information exchange between the analysts and the decision maker. Interactive strategies are designed to guide the DM to select the most preferred solution, within a set of feasible options. Then, a friendly environment of information exchange has been designed into a Graphical User Interface (GUI), where GIS is used as part of the process in order to display the selected solutions, when required.

3.1 DESCRIPTION OF METAHEURISTICS TO BE USED

3.1.1 GRASP

Greedy Randomized Adaptive Search Procedure or GRASP, is a well known heuristic introduced in Feo and Resende (1989). In general lines, GRASP is a memory-less multi-start metaheuristic proposed to solve combinatorial optimization problems. The algorithm departs from a *seed* solution and builds a feasible solution by inserting one element at a time. A *Restricted Candidate List* or *RCL* is calculated at each step of the *construction phase*. In particular, RCL is the set of the candidates to be included in the partial solution ordered by the insertion cost given by a greedy function. The element to be included into the partial solution is randomly selected within this list, which is continuosly updated after each insertion. Solutions generated by GRASP construction are not necessarily optimal and a local search phase is required to improve them. Thus, the solution's neighborhood is deeply investigated during a second phase of *local search* until a local optimum is found. A pseudocode of GRASP heuristic is shown in Algorithm 12.

Algorithm 2 GRASP basics.

```
function UPDATEREFSET(α)
    initialize best solution S*.
    seed = generate a random seed solution.
    while Stopping condition is not achieved do
        S = GreedyRandomizedConstruction(α,seed);
        S = localSearch(S);
        if cost (S) is better than cost (S*) then
            S* = S;
        end if
    end while
    return S*
end function
```

Any GRASP is characterized by two parameters: one related to the stopping criterion, *number of iterations*, and another one to the quality of the elements in the restricted candidate list, $\alpha \in [0,1]$. This heuristic is considered adaptive due to cost variations when inserting a not - yet - chosen element. In order to obtain the Restricted Candidate List (RCL), determine the incremental cost of introducing element e into the solution under construction, $c(e)$, and c^{min} and c^{max}, the smallest and largest costs, respectively. The length of RCL can be limited by a fixed cardinality or by the value of a parameter α. In the latter case, RCL will contain all the "feasible" elements, e, to be inserted in the solution under construction, where their incremental cost ranges between $c(e) \in [c^{min}, c^{min} + \alpha(c^{max} - c^{min})]$. It can be proved that parameter α controls the amount of greediness and randomness in the algorithm, considering it *pure greedy* when $\alpha = 0$ and *pure random* when $\alpha = 1$ (see Resende and Ribeiro (2016)). In Prais and Ribeiro (2000) the role of parameter α is studied on four different combinatorial optimization problems including a matrix decomposition for traffic assignment in communication satellite, set covering, weighted MAX-SAT and graph planarization. This analysis is carried out in the following scenarios:

1. α is randomly chosen from a uniform discrete probabilistic distribution. Its performance presents a large number of best solutions found, proving the effectiveness of strategies that vary α parameter and how does it affect to the RCL.

2. α is randomly chosen from a decreasing non-uniform discrete probabilistic distribution.

3. α is fixed close to zero. While obtaining a fast algorithm, the least variability of the results implies finding the best solution in just a few cases.

4. The value of α is periodically modified according to the quality of the obtained solutions. In an attempt to introduce memory into GRASP, a set of m values is given for α. At the first iteration, all these values have the same probability to be selected as the RCL parameter, but these probabilities are periodically updated based on the quality of the solutions obtained. These modifications increase the probability associated with those values that provide higher quality results. Then, the robustness and solution quality improves when incorporating a learning mechanism into the GRASP construction, what has been called *reactive GRASP*. Here, parameter α is defined to determine the level of randomness used to set up RCL.

Its simplicity and easy implementation have leaded to successful applications on multiple combinatorial optimization problems, such as scheduling problems, quadratic assignment problems, satisfiability problems or graph planarization, among others.

In particular, some results have been obtained using GRASP on different variations of the vehicle routing problem. Due to the multiple benefits that it could bring to the companies in real world problems, the common goal in the methodologies developed has been the minimization of the total cost. For instance, Pacheco and Delgado (1999) describe a constructive GRASP for the *Vehicle Routing Problem (VRP)* with heterogeneous fleet. In tis case, if c_{ij} determines the cost of introducing element i into route j, at the position where the insertion cost is minimum, then $b_i = c_{ij} - c_{ij^*}$ represents the greedy function value of element i, where route j^* would be the best route and route j the second best route. Then,

$$RCL = \left\{ i / \frac{b_i}{b_{max}} > 1 - \alpha \right\}$$

and the one with highest value at the greedy function is selected. When obtaining a feasible solution, OR local search operator (Or, 1976) is applied. Later, Pacheco and Delgado (2000) introduce a combination with the heuristic of concentration (Rosing and ReVelle, 1997). Now, a list saves the elements that characterize the best solutions found after applying GRASP and then launch another heuristic or exact method to solve the original problem subject to select those elements in the list.

Also, Marinakis (2012) provide a method to solve the *Capacitated Vehicle Routing Problem (CVRP)* defining a cardinality - based RCL, using different constructions, including best known heuristics (Clarke and Wright (1964); Gillett and Miller (1974); Ryan et al. (1993)), at each iteration and incorporating Circle Restricted local search moves to the *Expanding Neighborhood Search* (Marinakis et al., 2005). These moves reduce the search to those edges with ending - nodes within a radius length larger or equal to the sum of the costs of the two candidates for deletion edges. Another approach to solve *CVRP* is given in Layeb et al. (2013). First, they re - define a density matrix, D, whose values are obtained using the following formulation:

$$D_{ij} = \frac{|Q - (q_i + q_j)|^p}{d_{ij}^k D_{i0} D_{0j}} \tag{3.1}$$

where Q is the maximum capacity and q_i the demand of customer i; d_{ij} is the distance between customers i and j, D_{0j} and D_{i0} represent the density value between customers i and the depot and so does D_{0j}, since the customer with index 0 represents the depot. The values for p and k are integers between 1 and 4. Hence, larger values of p introduce first customers with lower demand, whereas large values of k implies that the angular distance of the next customer to be introduced is small. Then, they construct a giant tour subject to the order given by the density matrix, i.e. the following customer to be introduced in the route will be the one with highest density. This giant tour is splitted based on the capacity constraint and each route is reordered using the nearest neighbor heuristic. In this study, *Simulated Annealing* metaheuristic is applied for the local search phase including a set of operators such as inter - route and intra - route swap and 2 - opt move.

In the last decades, an attempt to make logistic models closer to reality have leaded to an increasing number of researches that incorporate multiple objectives into the *Vehicle Routing Problem*. However, to the best of our knowledge, only two publications have been found that use *GRASP* metaheuristic to solve this kind of problems. Kontoravdis and Bard (1995) define a *GRASP* to minimize the total distance and the number of vehicles at the same time. Taking into account a hierarchical optimization idea, it uses the construction phase to optimize the distance and local search is not applied to each solution, but to the best solution after a fixed number of iterations, trying to reduce the number of routes used. The construction begins by the selection of seed customers, geographically dispersed and with a narrow time windows. Then, a penalty function is used to decide what other customer must become a seed-customer, based on the definition of opportunity cost. Another bi-objective *VRP* is tackled with *GRASP* metaheuristic in Oyola and Løkketangen (2014). Here, minimizing the difference between the individual routes length is incorporated, in addition to minimizing the overal distance, as the second objective function and no weights are assigned to any of them. It starts by finding the best solution for each objective applying Tabu Search. Then, based on the ruin and recreate strategy, it uses the common parts of the two solutions already found as the partial solution and *GRASP* is applied to complete it. The evaluation of each insertion customer, at the construction phase, uses *Pareto Rank* which was introduced in Mateo and Alberto (2012). During this procedure, non - dominated solutions are recorded and an additional set maintains a number of promising solutions that might become part of the approximation of the Pareto Set.

3.1.2 Path Relinking

In the need of creating a balance between search intensification and search diversification in combinatorial optimization problems, *Path Relinking* (Glover, 1997; Glover et al., 2000) was introduced as a natural extension of Scatter Search (Glover, 1977). In order to incorporate attributes of high quality solutions, it launches an exploration of trajectories connecting elite solutions. Normally, two solutions take part in this procedure: *initial solution* and *guiding solution*. Then, one or more paths connecting these solutions in the search space graph can be explored in the search for better solutions. This fact restricts the number of possible neighbors to the feasible solutions that are more similar to the *guiding solution*, which is one of the main advantages of this procedure. Usually, similarity is measured in terms of common attributes and during the algorithm, solutions are evaluated and, to improve efficiency of PR algorithms, some of them are selected to be improved by a local search algorithm.

In general, several items must be considered before designing a *Path Relinking* algorithm (Basseur et al., 2005):

- *Distance measure*. One of the key points of *Path Relinking* is to define the assessment of the distance between two solutions. It might be computed as the greatest shared substring, or the minimum number of permutations to join both solutions, or the number of attributes where they differ from each other.

- *Neighborhood structure*. It is important to decide the operator that will be considered to generate the path, with the intention to explore only those solutions that reduce the distance to the guiding solution. It is common to use swap or shift operators.

- *Selection criteria*. Two different strategies can be applied here: choose distant solutions to favor the exploration of the search space or select adjacent solutions to favor the intensification of the search around good solutions. Then, it will be necessary to establish a criterion to select the initial and the guiding solutions.

- *Path selection*. An easy, but very computationaly expensive, approach would be to generate all the possible paths and select the best one. Then, some approaches explore only the best solutions generated or a subset of the possible paths.

- *Improvement solutions*. Local search is applied in order to find any improvement of a new solution generated.

The selection of the initial and guiding solutions determines the character of *Path Relinking*. Suppose we have a minimization problem and two solutions: S^1 and S^2 such that $f(S^1) \leq f(S^2)$, where $f(S)$ denotes the value of solution S in the objective function f.

This process starts with one of the solutions from the *Elite Set*, S^1, and gradually transforms it into the other S^2 by swapping in elements from $S^2 \setminus S^1$ and swapping out elements from $S^1 \setminus S^2$. The total number of swaps made is $|S^2 \setminus S^1|$ known as the *symmetric difference* between S^1 and S^2. The choice of which swap to make in each stage is greedy, so usually, the most profitable move is performed. In addition to swapping, switch - moves are also studied.

Path Relinking follows a *backward* strategy if S^1 is chosen as the guiding solution and S^2 initial solution. Otherwise, it is known as *forward Path Relinking*. Another strategy is *mixed Path Relinking*, as detailed in Algorithm 3, where the connecting path is explored from extremities, i.e. at each iteration, the closest extremity to the current solution alternates between the initial and the guiding solution.

Algorithm 3 Mixed Path Relinking

function MIXED PATH RELINKING(S^i, S^g)
$S = S^i$
$S^* = S^g$
$f^* = f(S^*)$
  while $|N(S : S^g)| \geq 1$ **do**
    $S = argmin\{f(S') : S' \in N(S : S^g)\}$
    if $f(S) < f^*$ **then**
      $S^* = S$
      $f^* = f(S)$
    end if
    $S' = S$
    $S = S^g$
    $S^g = S'$
  end while
  Apply local search to improve S^*
  return $S^*, f(S^*)$
end function

It has been shown that exploring subpaths close to the extremities, often provides solutions that are as good as when exploring the path completely. Then, *Truncated Path Relinking* was designed to adapt *Path Relinking* so that only the restricted neighborhoods close to the extremities are explored. It can be applied to either backward, forward or mixed *Path Relinking*.

Besides, *restart strategies* empty the reference set, denoted as ε, so that they establish $\varepsilon = \emptyset$ if no improvement has been found for a fixed number of iterations.

When applying *Path Relinking* to solve *Vehicle Routing Problem (VRP)*, it is not immediately obvious what is meant by *moving along a path from a solution to the guiding solution*, so transformations, in this case, consist of moving nodes from one route to another or changing their positions within the same route. Ho and

Gendreau (2006) published an study that uses *Path Relinking* strategy to address a *VRP*. Here, five different strategies are defined to build the reference set, inspired by Ghamlouche et al. (2004), taking into account that the quality and the level of diversity of the solutions included in the Reference Set have a major impact on the quality of the generated solutions. It also adapts five criteria to choose the initial and guiding solutions. Finally, after running all possible combinations, the best results correspond to introducing in the reference set those solutions with a better objective value than those already in the set and the initial and guiding solutions are the most distant solutions within ε. The second best combination differs from the previous one in the selection criterion, where the guiding solution chosen corresponds to the best solution in ε, while the initial solution is the second best one. However, in case of large instances, another combination works better as a second alternative. In that case, it choses the initial and guiding solutions randomly from ε and the reference set is built with the best local minima obtained during the construction phase.

Moreover, when dealing with multiobjective problems, the philosophy of *Path Relinking* does not change, since the optimization process works guided by only one objective function. Basseur et al. (2005) describe *Path Relinking* in Pareto MultiObjective Genetic Algorithms, which is the largest family of metaheuristics used to solve multicriteria logistic problems. In particular, they design a method to minimize the makespan of a flow - shop scheduling problem. To increase the intensity of the search around solutions with similar quality on the different objectives, the authors propose connecting solutions that are close to each other in the objective space. Hence, they randomly choose among Pareto solutions obtained from the Genetic Algorithm. At each iteration, the neighborhood of the initial solution, generated by the shift operator, is evaluated and non - dominated solutions are saved. Also, to reduce the computing time, a random aggregation of the objectives is executed which enables to select only one solution in the set of elegible solutions. Finally, a Pareto Local Search (PLS) is launched after each *Path Relinking* generation. This local search maintains a pool of potentially efficient solutions and iteratively improves this set by including non - dominated solutions found when exploring its neighbourhood.

Taking into consideration these studies, a *Path Relinking* algorithm is developed in Section 3.2.3 to improve a given set of nondominated solution for the *MultiObjective Capacitated Vehicle Routing Problem (MOCVRP)*.

3.1.3 Hybridizing GRASP and Path Relinking

Usually, each search in *GRASP* uses no information obtained by any other previous search, so that, as explained in Section 3.1.1, *reactive GRASP* and other adaptive memory techniques were proposed to take advantage of previous iterations and diversify the search space. On the other hand, *Path Relinking* uses an *Elite or Reference Set* ε of size N_ε that contains a group of diverse high - quality solutions found at previous iterations. Then, given two solutions, their symmetric difference is determined by the set of elements that mark the difference between both solutions. The cardinality of this set of symmetric differences is used to obtain diversification in ε. Hence, the main reason for applying *Path Relinking* to *GRASP* heuristic is the incorporation of a long - term memory mechanism to *GRASP*. The hybridization of *GRASP and Path Relinking* was proposed, for the first time, to solve a 2 - layer straight line crossing minimization problem in Laguna and Martí (1999). Since then, it has been widely applied to solve different combinatorial optimization problems, such as job shop scheduling problem (Aiex et al., 2003), max - min diversity problem (Resende et al., 2010) or Capacitated Arc Routing Problem (CARP) (Reghioui et al., 2007), among others.

Each solution obtained by *GRASP* is relinked with one or more solutions from ε and the resulting solution, S, is considered to be included in the Elite Set, ε, taking into account the value of the symmetric difference. So, a solution S, paired with another solution S' in ε, will be added to the Elite set if $\Delta(S, S')$, which denotes the symmetric difference, is maximized. For a minimization problem, when inserting a new solution in ε, this set is updated as indicated by the following function (Algorithm 4):

Algorithm 4 Procedure to update the Reference Set.

function UPDATEREFSET(ε, S)
  if $|\varepsilon| < N_\varepsilon$ **then**
    if $\varepsilon = \emptyset$ **then**
      $\varepsilon = \varepsilon \cup S$
    else
      $\mathrm{k} = \min \left\{ |\Delta(S, S^{'})| : S^{'} \in \varepsilon \right\}$
      if $k > 0$ **then**
        $\varepsilon = \varepsilon \cup \{S\}$
      end if
    end if
  else
    $f^{max} = \max \{ f(S') : S' \in \varepsilon \}$;
    $\mathrm{k} = \min \left\{ |\Delta(S, S^{'})| : S^{'} \in \varepsilon \right\}$
    if $f^{max} > f(S)$ & $k > 0$ **then**
      $S^{min} = argmin \left\{ |\Delta(S, S^{'})| : S^{'} \in \varepsilon \right\}$
      such that $f(S^{'}) \geq f(S)$
      $\varepsilon = \varepsilon \cup S \setminus \left\{ S^{min} \right\}$
    end if
  end if
  return ε
end function

Let N_ε denote the maximum size allowed for ε. Solutions obtained from *GRASP* are introduced in ε while there are elements that differenciate both solutions, whose cardinal is given by k. When the size of ε is reached, to insert a new solution, S, into the set, another solution must be removed. To maintain quality and diversification, the worst value of the objective function evaluated on the current ε is computed and denoted by f^{max}. Also, one determines the minimum cardinality of symmetric differences between S and the current elements that belong to the elite set, k. Finally, S is added to ε if its value is better than the worst value, f^{max}, and k is not null. Also, to maintain the size of the elite set, the solution with the minimum number of dissimilar elements in ε, S^{min}, will be removed. Note that if one defines a minimum value that should be accomplished by k, k^{min}, the diversity of the Elite Set could increase. This is done by considering $k \leq k^{min}$ where k^{min} is greater than zero.

Different *Path Relinking* techniques have been introduced to speed up and improve the results. For instance, Resende and Werneck (2004) propose the *Evolutionary Path Relinking* as a post - optimization or periodical phase for *GRASP*. It consists of applying *Path Relinking* between each pair of solutions in ε. First, every multistart iteration is followed by an intensification step, in which the newly generated solution is combined with a solution selected from ε. They assign probabilities proportional to their symmetric difference with respect to the last solution obtained. Later, all solutions from ε are combined with each other.

As mentioned in Section 3.1.2, applying *Path Relinking* to *Vehicle Routing Problem (VRP)* is not trivial. To the best of our knowledge, only Sorensen and Schittekat (2013) describe a distance - based *Path Relinking*, which allows the definition of distance between two solutions as the minimal number of moves needed to transform one solution into another. To compute this calculation, the operator to be used in the transformation must be specified. Common operators from literature include *swap*, *relocate* or *2 - opt*, among others. For instance, Sorensen and Schittekat (2013) remove a customer from a route and re - insert it into another position in the same route or a different one. The Reference Set is constructed by launching a *GRASP* heuristic whose construction uses the classical insertion heuristic by Clarke and Wright (1964) and incorporates a Restricted Candidate List that provides randomness to the construction. *Path Relinking* is divided in two phases in this case. First of all, a squared distance matrix is computed containing the tours of the initial solution in rows and those from the guiding solution as columns, using the ”distance” definition previously stated.

Then, identifying this matrix as the cost matrix, the algorithm solves a minimum - cost assignment problem, which matches each route of the initial solution to another route of the guiding solution. Finally, a customer should be moved to the position that occupies in the guiding solution's relative route, so that the distance between both solutions decreases. A general pseudo - code is shown in Algorithm 5, where M represents the set of customers to be moved and F those to maintain fixed.

Algorithm 5 Path - relinking proposed in Sorensen and Schittekat (2013)

```
function PR(S^1, S^2)
    M = ∅
    F = ∅
    Calculate the distance between S^1 and S^2
    Find the elements of M and F by solving an minimum cost - assignment problem
    set u_0 = S^1
    for i = 1 to n-1 do
        Remove customer i from M
        Add customer i to F
        Create solution u_i: move customer i in u_{i-1} to the solution that it occupies in S^2,
relative to the customers in F.
    end for
    return ε
end function
```

If we extend the analysis to problems which consider more than one criteria, the hybridization of *GRASP* and *Path Relinking* has briefly been studied. One can find a rich summary of multiobjective *GRASP* applications in Martí et al. (2015), which involve metaheuristics designed to solve the multicriteria minimum spanning tree problem (Arroyo et al., 2008), the multiobjective quadratic assignment problem (Li and He, 2009) or path dissimilarity (Martí et al., 2009), among others. In particular, this work tackles the biobjective orienteering problem and the biobjective path dissimilarity problem, whose ideas might be implemented to solve *MultiObjective Vehicle Routing Problem (MOVRP)* as well. In this work, different proposals of multiobjective *GRASP* and *Path Relinking* are given and combined in order to obtain the best approximation of the Pareto Set for some optimization problems. A general idea might be subdivides the different proposals into a main scheme that follows three steps: *construction*, *local search* and *Path Pelinking*. Note that each construction is guided by a greedy function that measures the cost of inserting an element in a solution under construction, so for multiobjective *GRASP*, a set of greedy functions, $\{g_1, g_2 \ldots g_m\}$ must be defined in terms of the variation of each objective when inserting an element into a partial solution.

Hence, two different approaches hold for the construction phase: pure and combined.

- **Pure construction**. At each construction, a single objective is selected to be optimized. Then, only one greedy function will be considered at each construction. It allows the generation of the Restricted Candidates List (RCL) which will be used, and updated, at each step during the construction. However, the selection of this objective might correspond to an ordered fashion, which is called *pure - ordered* or *pure - random*, otherwise.

- **Combined construction**. Now, each construction is optimized guided by more than one greedy function at a time, so the selection of each candidate will be determined by a different greedy function g_i. Again, two alternatives arise, given by:

  - **Sequential - combined**. It incorporates one element at a time by generating the corresponding RCL based on a selected greedy function, g_i. Depending on the selection procedure, it might be subdivided into *pure - sequential* or *random - sequential*.

  - **Weighted - combined**. In order to evaluate the insertion cost of an element, a greedy function is defined by aggregating all the greedy functions considered. If w_j represents the weight associated to the j - th greedy function g_j, the aggregated greedy function is formulated as follows:

$$g(c) = \sum_{j=1}^{k} w_j g_j \tag{3.2}$$

In this case, the given set of weights can change at each construction step or maintain their values all along the process. Note that if there were objective functions with conflicting sign, they should be unified to avoid bias.

It is important to recall that, in multiobjective local search, every solution visited is a candidate to belong to the nondominated solution set, so it is necesary to check this relation along the process. Following the same scheme as in the construction phase, local search can be defined in two main strategies depending on the way of selecting the objective function. Hence, if the solution was generated by pure - construction, then pure - local search will attempt to improve the objective function under consideration after each construction, where the deterioration of any other objective function is permited if the one to optimize improves. However, if a combined constructive method has been applied, no deterioration is allowed. Moreover, sequential - combined local search consists of optimizing a different objective function when selecting the best solution in the neighborhood. And, finally, for the weighted - combined local search, the following aggregated objective function (Eq. (3.1.3)) guides the search:

$$f(c) = \sum_{j=1}^{k} w_j f_j \tag{3.3}$$

As usual, the design of *Path Relinking* needs to define how to measure the distance between the initial and guiding solutions and the operator that will take part in the transformation of one solution into another. Here, swap operator is considered to evaluate the behaviour of *Path Relinking* and the distance is given by the symmetric difference between both solutions. In order to evaluate the possible moves, a set maintains candidates to be swapped by evaluating the removing and insertion costs. The Elite Set is defined by nondominated solutions generated from *GRASP*. *Path Relinking* will be applied to all the possible pair - combinations, so that the selection of the initial and guiding solutions is not relevant. Again, three strategies are proposed to perform the *Path Relinking*: pure, sequential and weighted.

- **Pure - Path Relinking** selects the elements to be removed and re - inserted by evaluating all the possibilities, according to one objective function, and the best swap is performed. Only one of the objectives is considered at each application of the *Path Relinking* to select the intermediate solutions for the entire path.
- **Sequential Path Relinking** changes the objective function in an ordered fashion to evaluate the intermediate solutions found along the process.
- **Weighted Path Relinking** selects elements to swap according to the value of the aggregated function used for the construction and local search phase.

These alternatives are combined and tested for different combinatorial optimization problems and, surprisingly, a different alternative is more suitable for each group of instances. In particular, due to the possibility of translating this methodology to solve *MOVRP*, it is interesting to highlight that, for the biorienteering problem, the best results are obtained using the weighted variant.

3.2 Multiobjective algorithms proposed

In order to define models that best represent reality, many authors have incorporated multiple criteria to the different *Vehicle Routing Problems (VRPs)*. Literature reveals a considerable number of publications incorporating different combinations of criteria to *VRP*, specially to the *Vehicle Routing Problem with Time Windows (VRPTW)*. For instance, to minimize route's duration and customer's waiting time (Hong and Park, 1998), which are also used to solve multiobjective school bus routing problems such as Caballero et al. (2011) and Pacheco and Martí (2006); to minimize the number of vehicles used and the total distance (Ghoseiri and Ghannadpour, 2010; Rahoual et al., 2001) or to reduce total operational cost, labor infrautilization and vehicle maximum capacity (Calvete et al., 2007), among others.

In general, different metaheuristic approaches have been developed in order to determine the best approximation of the Pareto efficient set for *MultiObjective Vehicle Routing Problems (MOVRP)*. In this context, *Evolutionary algorithms* have been extensively developed. One of the best known is *NSGA-II* (Deb et al., 2002) which is flexible enough to be adapted to any variant of *VRP*. Also, genetic algorithms are implemented in combination with local search procedures. In such situation, we can find *Target Aiming Pareto Search* (Jozefowiez et al., 2007a), which is a genetic algorithm that improves solutions applying local search, or *Archive-Based hYbrid Scatter Search (AbYSS)* (Nebro et al., 2008), which consists of a *Scatter Search* that employs mutations and interchanges defined as genetic algorithms. Also, on non - linear multiobjective optimization problems, an hybridization of Scatter Search and Tabu Search, *SSPMO*, is introduced in Molina et al. (2007).

The set of algorithms directly designed for *MultiObjective Vehicle Routing Problem (MOVRP)* includes a multiobjective adaptation of *Ant Colony Optimization* (Baran and Schaerer, 2003) which provides the Pareto front for a *VRPTW* with three objectives: number of vehicles, total travelling time and total delivery time. Another example, *MOAMP*, is proposed in Caballero et al. (2007) to solve a location - routing problem using a metaheuristic based on *Tabu Search* and also used in the resolution process of a *Waste Collection Problem* with multiple criteria in Gómez et al. (2009).

Other objectives contemplate if the income of the driver depends on the travelled distance, so that including route balance in the problem could make it closer to reality. Different approaches of *evolutionary algorithms* which are combined with tabu search (Jozefowiez et al., 2007a, 2002) or other additional diversification strategies (Jozefowiez et al., 2009) have been proposed to provide solutions to this variant of *VRP* (see Jozefowiez et al. (2007b) for more details). Different factors highlight the importance of including the route balance objective in our research, that is why, as a parenthetical remark, it is convenient to provide some details about it in the following lines.

ROUTE BALANCING

To introduce fairness in the working journey, different studies try to balance the set of driven routes. Multiple alternatives have been used to model this objective and several researches propose strategies to deal with it. Route Balancing was incorporated to *VRP* for the first time in Jozefowiez et al. (2002), by defining the balance as the difference between the largest and shortest route (Eq.(3.5)). This " balance" - definition is also considered in Lacomme et al. (2015). They propose a solution process that starts from a seed solution where each customer defines a route. Then, these routes are merged by randomly selecting nodes and inserting them into the most promising position in the shortest route. Finally, routes are randomly concatenated and an improvement phase is launched. It, first, tries to remove all the customers from the shortest route and, then, apply the *2 - opt* operator, allowing only those moves that do not deteriorate the balance value. They use a *genetic algorithm* combined with *Path Relinking* to solve this problem, maintaining two different populations: one that contains promising nondominated solutions and the other one with the best approximation of the Pareto set. Here, *Path Relinking* converts each solution into a giant tour by concatenating the routes and then, it tries to transform one solution into another by nodes interchanges. Another approach introduces a memetic algorithm (Mandal et al., 2015).

In general, within the family of *Vehicle Routing Problems (VRPs)*, different approaches have been used to define route ”balance”.

> ”To define balancing objective, it is necessary to define the workload for a tour, which can be measured as the number of customers, the quantity of delivered goods, tour length or required time, among others.” (Jozefowiez et al., 2009)

Jozefowiez et al. (2007a) designed a local search within a *Genetic Algorithm* to solve a biobjective *VRP*. This methodology, based on *Target Aiming Pareto Search*, consists of an iterative process that combines cooperative local search and the use of a set of appropriate goals. Here, local search is only applied to those solutions that belong to the potential Pareto set. Moreover, it re - defines the direction of the search depending on the local search, l_i, and the goal, g_i, fixed for that l_i. Actually, the resultant objective function seeks the minimization of the distance to the goal, similar to Eq.(3.7) where the target (T) is given by g_i. Then, a set of nondominated solution is obtained for each i and the Pareto Set is formed by the union of all them.

Recently, Halvorsen-Weare and Savelsbergh (2016) provided an analysis on the results when applying different formulations for the ”balancing” for the Mixed Capacitated General Routing Problem. To solve the different bicriteria problems defined by the minimization of the total cost and route balance, this method incorporates the *box - method* into a lexicographic method combined with the well- known ε - *constraint* to densify the frontier. The definitions considered include:

$$\min \left\{ \sum_{i=1}^{R} (r_i - \mu)^2 \right\}, \tag{3.4}$$

where r_i represents the length of the i^{th} route and μ the mean of the routes length within the current solution.

$$\min\left\{r^L - r^S\right\}, \tag{3.5}$$

where r^L represents the longest route and r^S the shortest one.

$$\min\left\{\sum_{i=1}^{R}(r_i - r^S)\right\}, \tag{3.6}$$

where r^i represents the length of the i^{th} - route and r^S the length of the shortest one.

$$\min\left\{\sum_{i=1}^{R}(r_i - T)\right\}, \tag{3.7}$$

where r^i represents the length of the i^{th} - route and T is a given target value to achieve.

$$\min\left\{r^L\right\}, \tag{3.8}$$

$$\max\left\{r^S\right\}, \tag{3.9}$$

Based on the results obtained by a list of recognized instances, for the biobjective Mixed Capacitated General Routing Problem, a major number of nondominated solutions are obtained when defining route balance by Eq. (3.5).

Formulation (3.8) is used in Pacheco and Martí (2006) to minimize the number of vehicles and the travelling time for a school bus routing problem. Following the philosophy of $\varepsilon-$ *constraint method*, they solve a single objective problem for each possible value of the number of vehicles. Then, to minimize the longest travelling time at the bus, different heuristics are defined to construct an initial solution and it is lately improved with *Tabu Search*. When a first approximation of the Pareto front is generated, *Path Relinking* is applied to improve each of these results. Another application to the Bus Routing Problem is included in López-Sánchez et al. (2014). As the previously mentioned work, they also try to minimize the number of vehicles at the same time that they reduce the makespan, which is defined as the maximum time a customer spends at the vehicle. In this case, the methodology is developed for Open Vehicle Routing Problems. To generate an initial solution an insertion heuristic is applied, introducing unrouted customers into the route and position which minimizes the incurrent cost. Then, an improvement phase is applied to promising solutions.

A new alternative for the balance measurement has recently been defined in Zhou et al. (2013). It is formulated as the quotient of the difference between the longest and the shortest route and the mean of the total distance, i.e. the value is calculated using equation 3.2:

$$\frac{r^L - r^S}{\frac{1}{R} \cdot \sum_{i=1}^{R} r_i}, \tag{3.10}$$

where r^i represents the length of the i^{th} route, so that if $i = L$ corresponds to the length of the longest route and the shortest route if $i = S$, and R is the number of routes or vehicles. Then, the balance is obtained when minimizing equation (3.2). The methodology developed to solve this problem, where minimizing the total distance is also considered, applies Genetic Algorithm defining three operators: selection, crossover and mutation.

Recently, route balancing, has been incorporated into Waste Management. Among other objectives, Hemmelmayr et al. (2013, 2014) considers route balance and provide a complete study on *Waste Collection Problem*, modelled as a *Periodic Vehicle Routing Problem (PVRP)*. In this case, they solve bins allocation and routing design multicriteria problems in different scenario, with the aim of balancing the trade - off between the frequency of a given service and the number of bins that can be placed within that area. Their methodologies use *Variable Neighborhood Search* to obtain an efficient route design. The initial solution is obtained by the well - known *saving - heuristic*, whereas the shaking phase defines its neighborhoods based on different operators such as change of combination, move, cross or change of frequency. The solution obtained is improved with dynamic programming to insert intermediate facilities. Finally, the stopping criterion is formulated in the same terms of *Simulated Annealing*. Besides, other interesting aspects are contemplated in this work such as multiple waste type case, the number of bins allocated on a specific area, the capacity or volume of bins or the cost associated to a service. More recently, López-Sánchez et al. (2017) proposed a hybrid algorithm, which combines *GRASP* and *Variable Neighborhood Descend (VND)*, to minimize the overall distance and to minimize the longest route. Four different neighborhood structures are proposed, and adjusted for the balance objective. They also include *λ-interchange*, *exchange operator*, *relocate operator* or interchange two consecutive nodes.

After this revision of the different approaches considered for routing balance, this section continues by describing the different schemes developed in this work in order to obtain a good approximation of the Pareto Set for a *Multiobjective Capacitated Vehicle Routing Problem (MOCVRP)*. As seen at Section 3.1.1, just a few research works have considered applying this technique to solve *MOCVRP*. The approaches proposed in this work consists of two steps. First, to generate an approximation of the Pareto et using *GRASP* metaheuristic and a second step tries to improve it using another metaheuristic. Then, it is crucial to define the local search strategy to be implemented within the *GRASP* metaheuristic, as well as the *multiobjective GRASP* strategy itself. To obtain the set of nondominated solutions, *GRASP* and its combination with Path Relinking and Variable Neighborhood Search (VNS) are described in two different multiobjective metaheuristic approaches. This methodology, combined with the idea introduced in Martí et al. (2015) and the definition of an achievement function as detailed in Section 3.2.2, will provide different ways of obtaining an approximation of the Pareto Set which will be discussed in Section 3.2.4.

CONSTRUCTION PHASE

Given an initial partial solution *seed*, which consists of a single - customer route system, a random parameter $\beta \in [0, 1]$ is generated and the non-visited nodes are inserted in routes one at a time and marked as visited. In this case, to make it simple, the initial customer has randomly been chosen.

In order to evaluate the insertion cost, we combine two greedy functions: *Extramileage* and *Regret*.

- The first one is based on the classical heuristic introduced in Mole and Jameson (1976), whose insertion criterion is the evaluation of the extra distance, also known as *Extramileage*. Its value is obtained by evaluating the insertion cost of an unrouted customer k between two consecutive customers i and j in a particular route. Therefore, if the insertion is feasible, the extramileage value is given by $c(i, k, j) = c_{ik} + c_{kj} - c_{ij}$; otherwise, the extramileage value is set to infinity.
- On the other hand, the *Regret* value reflects the variation cost of inserting a node in the second best route instead of the best one. This is measured as the difference between the two minimum extramileage values of the node in both routes. This idea is taken from the economic concept of *opportunity cost* and was used for the first time in Christofides et al. (1981) as a second step of an insertion heuristic. Since then, it has been applied in other effective constructions, such as Fisher and Jaikumar (1981) and Pisinger and Ropke (2007).

These insertion costs are evaluated at each position of each route for each node to be inserted, and the minimum extramileage value is saved for each route, in order to facilitate the calculation of regret and update this data when a node is assigned to a route.

Parameter β indicates, at each construction, the number of nodes to be inserted maximizing the regret greedy function. Later, the next node to be inserted will correspond to the one that minimizes the cost given by the extramileage value. Given a seed solution s, we can summarize the construction as detailed in Algorithm 6.

Algorithm 6 Construction.

```
function CONSTRUCTION SCHEME(V, s, β)
    for n ∈ V \ {seed nodes} do
        CalculateExtramileage(n,s)
        CalculateRegret(n,s)
    end for
    while number of visited nodes < β do
        Find the node with the maximum regret value, maxRegretNode
        Insert maxRegretNode in the corresponding route, r, and position
        Mark maxRegretNode as visited
        for n ∈ V \ {visited nodes} do
            updateExtramileage(n,s,r)
            CalculateRegret(n,s)
        end for
    end while
    while there are non - assigned customers available do
        Find the node with minimum extramileage value, minExtramileageNode
        Insert minExtramileageNode in the corresponding route, r, and position
        Mark minExtramileageNode as visited
        for n ∈ V \ {visited nodes} do
            updateExtramileage(n,s,r)
        end for
    end while
    return s
end function
```

In order to save computational time, the function *CalculateExtramileage* determines the minimum extramileage cost for each node at each route and saves it in an array, so that when we call *updateExtramileage* with parameter r, only the extramileage asociated to route r needs to be re - computed. Besides, this scheme allows to calculate the regret value for each node as the difference between the two minimum values of the array.

This construction scheme is based on the construction algorithm proposed in Maniezzo and Roffilli (2008) and it does not depend on the objective considered, so the same scheme is applied independently of the function to optimize. However, a few modifications have to be made in order to evaluate the variation of each objective, so that the extramileage value for a node n in a route r , is given by the variation cost on the function to be optimized.

LOCAL SEARCH

When a feasible construction has been generated, local search is applied in order to improve its value for a given objective. A *Neighborhood* is defined as a set of mappings that associate each feasible solution S with a set of feasible solutions $N(S) = \{S_1, S_2, \ldots S_p\}$ that can be obtained by a simple modification of S. Then, each S_i is obtained from S by an operator called *move*. *Local search* consists of the evaluation of each element of $N(S)$ and executing the corresponding move if any improvement is found.

Different factors determine the effectiveness of a local search procedure, such as the neighborhood structure, the search technique, the speed required to evaluate the cost function of the neighbors and the starting solution itself. Two search strategies are defined: *best-improving* (Algorithm 8), where all neighbors are investigated and the current solution is replaced by the best neighbor, and *first-improving* (Algorithm 7), in which the current solution moves to the first neighbor that improves the objective value.

Algorithm 7 First improving scheme.

```
function FIRST IMPROVING SCHEME(S, N(S))
    improvement = TRUE
    while improvement = TRUE do
        improvement = FALSE
        for S' ∈ N(S) do
            while improvement = FALSE do
                if f(S') < f(S) then
                    S = S'
                    improvement = TRUE
                end if
            end while
        end for
    end while
    return S
end function
```

Algorithm 8 Best improving scheme.

```
function BEST IMPROVING SCHEME(S, N(S))
    improvement = TRUE
    while improvement = TRUE do
        improvement = FALSE
        f_best = ∞
        for S' ∈ N(S) do
            while improvement = FALSE do
                if f(S') < f_best then
                    S_best = S'
                    f_best = f(S')
                end if
            end while
        end for
        if f_best < f(S) then
            S = S_best
            improvement = TRUE
        end if
    end while
    return S
end function
```

There are a few techniques that help in the implementation of an efficient local search. Some examples, as the ones listed below, can be found at Resende and Ribeiro (2016):

- Commonly, the cost of each neighbor S' is computed by updating the cost of the current solution S instead of calculating it from scratch. Usually, this cost represents the variation on the objective value if the move were executed.
- Another technique consists on generating a *candidate list* of possible moves that restrict the size of the neighborhood or maintains additional

information from previous iterations. Best - improving strategy is better considered when applying this technique.

- However, if first - improving strategy is stablished, the *circular strategy* defines an order in the candidate list generated. Then, if p is the size of the candidate list, each of these moves is evaluated in ascending order until the first improvement is found. This idea lies on the fact that, if previous neighbors have been explored with no improvement found, then, it would be more interesting to continue moving forward and exploring the following neighbors in the candidate list instead of the already explored ones. Let J be the first - improving move found, meaning that for previous elements, S_i with $i < J$, and the considered operator, the search was not successful. When performing the J - move, the process continues evaluating S_i with $i > J$ instead of evaluating from $i = 1$, which only happens when $i - 1 = p$. This process goes on until a circle with no improvements is completed. This strategy is incorporated to well - known algorithms such as *Variable Neighborhood Descend (VND)* described at Algorithm 9 (Mladenovic and Hansen, 1997), which starts exploring neighborhoods whose elements can be quickly evaluated and progressively moves to more complex evaluations.

- Some metaheuristics, like *Iterated Local Search* or *Tabu Search*(Glover, 1989, 1990), use *ejection chains* (Glover, 1996) to diversify the search towards unexplored regions in the search space. This strategy incorporates compound moves that may vary between step and step. This strategy is computationally expensive, but really effective to introduce perturbation and diversification.

Algorithm 9 Variable Neighborhood Descend

```
function VND(S, {N_1(S), ··· , N_k(S)}, f, max_iter)
    index = 1
    while index ≤ k do
        S' = LocalSearch(S, N_index(S), f, max_iter)
        if f(S') < f(S) then
            S = S'
            index = 0
        else
            index = index + 1
        end if
    end while
    return S
end function
```

In the literature we can find a general classification of neighborhoods applied to *VRP* (Toth and Vigo, 2002) into *intra-route neighborhoods* (Golden and Assad, 1988), which operate on a single route at a time, or *inter - route neighborhoods* that consider moves between more than one route simultaneously.

Common neighborhoods used to solve combinatorial optimization problems such as *Travel Salesman Problem (TSP)* or *Vehicle Routing Problem (VRP)* include the well - known λ - *opt* (Lin, 1965) and *Or - exchanges* (Or, 1976). In the first case, λ edges, usually $\lambda = 2$ or $\lambda = 3$, are removed from the current solution and replaced by other λ edges; whereas the second strategy uses restricted neighborhoods characterized by the subset of moves associated with larger λ values. In general lines, local search is detailed as shown in Algorithm 10.

Algorithm 10 Local search

function LOCAL SEARCH SCHEME(S, $N(S)$, f, max_{iter})
  $iter = 0$
  while $iter < max_{iter}$ **do**
    $S^* = best - improving(S, N(S))$
    Check relation of dominance of S^*
    if S^* is mutually non - dominated by any other solution in the current Pareto Set
then Update Pareto Set by including S^*
    end if
    if $f(S^* < f(S)$ **then**
      $S = S^*$
      $iter = 0$
    else$iter = iter + 1$
    end if
  end while
  return S
end function

So, for a given solution, a neighborhood $N(S)$ is explored in order to find any improvement on the single- objective case or a nondominated solution in the multi - objective problem.

However, different strategies have been introduced in the last years. Here, we consider the application of a simple *Variable Neighborhood Search (VNS)* (Mladenovic and Hansen, 1997) detailed in Algorithm 11:

Algorithm 11 Variable Neighborhood Search

function VNS(S, $\{N_1(S), \cdots, N_k(S)\}$, f, max_{iter}, n_{max})
  $n = 1$
  while $n \leq n_{max}$ **do**
    $S' = shake(S, n)$
    $S' = VND(S', \{N_1(S), \cdots, N_k(S)\}, f, max_{iter})$
    if $f(S') < f(S)$ **then**
      $S = S'$
    end if
    $n = n + 1$
  end while
  return S
end function

To escape from local optima during the search, Variable Neighborhood search first applies a *shaking* method that moves a number n of randomly chosen nodes from their current route to another route for a given a solution S, providing a new solution. The resulting solution, S' will have a new value on the objective function which is allowed to deteriorate the original solution S. Then, a series of neighborhoods $\{N_1(S), \cdots, N_k(S)\}$ are explored in a given sequence so that if any improvement is found, the first neighborhood on the list is explored next, otherwise the sequence continues to explore the next neighborhood. This *cyclic* local search is known as *Variable Neighborhood Descend* (Algorithm 9) and it continues until no improvement is found for a prefixed number of iterations at the last neighborhood. Usually, the neighborhoods considered are ordered in terms of their sizes, so that the smallest ones come first.

In what follows the set of neighborhoods generated by the following moves will be considered in this work:

$N_1(S)$ As an example of inter - route operator, it moves, if feasible, a subchain of k nodes from one route to another.

$N_2(S)$ *2 - opt* operator is used to invert the order of a subchain in a given route.

$N_3(S)$ These neighbors are determined by moving a subchain of a route to another position, J, within the same route (see Subramanian et al. (2014)).

For the sake of simplicity, considering that this procedure will be applied to multiple objectives, these operators are defined as functions with a set of inputs that return a pre - computation of the cost if the move were executed. Then, for a given solution, S, $N_3(S)$ takes two different routes and the subroutes to be interchanged and it will compute the values, in terms of distance and time required, of these routes if the move were executed; whereas $N_2(S)$ will return the corresponding distance or time value if a given subchain reversed its order.

Therefore, the set $N(S) = \{N_1(S), N_2(S), N_3(S)\}$ will define a *Variable Neighborhood Descend (VND)* that will be used in any of the variants of *VNS* within this work, which are adjusted to deal with multicriteria problems as detailed in Section 3.2.3.

Thus, once the standards of the constructive and a local search scheme have been explained, the *GRASP* heuristic introduced in this work can be defined as indicated in Algorithm 12, where construction and local search improvement phase are applied according to the explanation above.

Algorithm 12 GRASP procedure.

```
function GRASP(V, numIter, f)
    Set s* as the best solution.
    for iter ∈ 1, 2, ··· numIters do
        Define a value for parameter β and generate the solution seed s.
        s = construction(V, s, β)
        s = localSearch(s, N(s))
        if f(s) < f(s*) then
            s* = s
        end if
    end for
    return s*
end function
```

This will be the scheme used in the following sections to solve the single - objective problems found.

3.2.1 MULTIOBJECTIVE GRASP: ALTERNATING OBJECTIVES

A first approach of the Pareto optimal set is inspired by the *MultiObjective GRASP (MOGRASP)* introduced in Martí et al. (2015), in particular, from their proposal of Pure approach of *MOGRASP*. As detailed at the end of Section 3.1.3, this method uses a single greedy function to construct each solution for every objective. However, the constructive algorithm developed in this work considers two greedy functions, as shown in Algorithm 6 which are: extramileage and regret. Notice that they only represent a concept of variation, so the computation of their values change depending on the objective to optimize.

Pure - ordered (Algorithm 13) and *pure - random*(Algorithm 14) alternatives of the original description are implemented, in order to generate an approximation of the Pareto front.

Algorithm 13 Pure Ordered Multiobjective GRASP.

function PURE - ORDERED MULTIOBJECTIVE GRASP(V, $numIter$,$f = (f_1, f_2, \cdots f_k)$)

  Initialize $\wp = \emptyset$ and the initial function to optimize, f_i.

  for $iter \in 1, 2, \cdots numIters$ **do**

    Define a value for parameter β and generate the solution seed s.

    s = construction(V, s, β)

    if s is nondominated in $\wp$ **then**

      $\wp = \wp \cup \{s\}$

      update $\wp$

    end if

    s = localSearch(s, **N**(s))

    if s is nondominated in $\wp$ **then**

      $\wp = \wp \cup \{s\}$

      update $\wp$

    end if

    if $i > k$ **then**

      Reset the index of function to optimize: $i = 1$.

      otherwise: $i = i + 1$

    end if

  end for

  return $\wp$

end function

Algorithm 14 Pure Random Multiobjective GRASP.

function PURE - ORDERED MULTIOBJECTIVE GRASP(V,$f = (f_1, f_2, \cdots f_k)$)
  Initialize $\wp = \emptyset$.
  for $i \in 1, 2, \cdots k$ **do**
    Determine a random function, $i < k$ to optimize, f_i.
    Define a value for parameter β and generate the solution seed s.
    s = construction(V, s, β)
    if s is nondominated in $\wp$ **then**
      $\wp = \wp \cup \{s\}$
      update $\wp$
    end if
    s = localSearch(s, **N**(s))
    if s is nondominated in $\wp$ **then**
      $\wp = \wp \cup \{s\}$
      update $\wp$
    end if
  end for
  return $\wp$
end function

Usually, as explained in Section 3.1.1, *GRASP* applies an improvement phase after each construction. In this case, in spite of considering multiple criteria, as the solution is generated based on a single objective, the local search will try to improve the same objective, verifying if a visited solution is candidate to be included into the nondominated solutions set.

These approaches take advantage of the randomness of *GRASP* to explore the function space, which allows to obtain a wide set of nondominated solutions, which may improve as the number of iterations increases. However, it does not guarantee a full exploration of the objective space, so additional techniques have to be implemented. This scheme was successfully applied in López-Sánchez et al. (2017) to solve a bi - objective waste collection problem.

3.2.2 Multiobjective optimization using an achievement scalarizing function

Considering an *achievement scalarizing function (ASF)* is one of the most widely used strategies to deal with multiple criteria problems. In particular, for the optimization in reference point based interactive methods. Miettinen (2008); Wierzbicki (1980) and Lewandowski and Wierzbicki (1989) confirm its ability to produce any (properly) Pareto optimal or weakly Pareto optimal solution. Given a weighting vector, λ, $\lambda_i > 0\ \forall i$, that determines the search direction, a partial solution x and a reference point $\mathbf{R}$, an *ASF* consists of an aggregation of terms of the form $\lambda_i \cdot (f_i(\mathbf{x}) - R_i)$. It aims to minimize the distance from R (specified by the Decision Maker (DM)) to the feasible region, if the reference point is unattainable, or minimizes the distance otherwise. Usually this distance is defined by an appropriate metric, such as L^∞, L^2 or L^1 in the objective space. A first approach of this *ASF*, introduced in Wierzbicki (1977), only ensured the optimal solution to be weakly efficient. In practive, *Wierzbicki's achievement function* modifies the metric L^∞ in order to ensure the generation of efficient solutions, so that if x is the solution under construction, $f_i(x)$ the values of this partial solution for every objective function $i \in 1, 2 \cdots k$, and R_i the *reference level* for each objective, then, considering the L^∞ metric, Wierzbicki's achievement function is formulated as:

$$\max\left\{\lambda_1 \cdot (f_1(\mathbf{x}) - R_1), \lambda_2 \cdot (f_2(\mathbf{x}) - R_2), \cdots, (1 - \sum_{i=1}^{k-1} \lambda_i) \cdot (f_k(\mathbf{x}) - R_k)\right\} + \rho \cdot \sum_{i=1}^{k} f_i(\mathbf{x}) \quad (3.11)$$

Not many authors have recently applied this approach to solve vehicle routing problem. Our goal is to minimize Wierzbicki's achievement function (Eq. (3.11)) for different values of λ, using the *GRASP* strategy as defined in Algorithm 12, where the construction and local search are detailed at the begining of this Section. Also note that the differences, in magnitudes, might cause a bias in the evaluation, so the values must be normalized. In this case, if the reference point is R_*, then the values are normalized into the interval [0,1], so that Wierzbicki's *ASF* is formulated in Eq. (3.12).

$$\max\left\{\lambda_1 \cdot \frac{f_1(\mathbf{x}) - R_1{}^*}{f_1^{max} - f_1^{min}}, \lambda_2 \frac{f_2(\mathbf{x}) - R_2{}^*}{f_2^{max} - f_2^{min}}, \cdots, (1 - \sum_{i=1}^{k-1} \lambda_i) \cdot \frac{f_k(\mathbf{x}) - R_k{}^*}{f_k^{max} - f_k^{min}}\right\} + \rho \cdot (\sum_{i=0}^{k} f_i(\mathbf{x})) \quad (3.12)$$

Theoretically, considering $f_{min} - \theta$ would define the *utopian point* as the reference point, where $\theta \in [0,1]$, which avoids generating weakly efficient solutions. However, in practice, discrete problems do not need to be aware of it. Hence, in order to obtain the best approximation of the Pareto front, we estimate the *ideal* (z^*) and *nadir* (z^{nad}) points by solving each of the single - objective problems derived from optimizing every objective function. Note that, in this context, we understand "nadir" as the worst possible scenario, and it is determined by the worst values for each function considered within the Pareto Set or its approximation. Again, to solve these single - objective problems, Algorithm 12 is applied to optimize the corresponding objective. Notice that these points are required to evaluate the scalarizing achievement function as formulated in (3.12).

Next, for a problem with k objectives, m convex combinations of λ values are randomly generated in the interval $[0,1]$, so that $\sum_{i=1}^{k} \lambda_i = 1$. λ will define the achievement scalarizing function (Eq. (3.12)), where $f_i^{max} = nadir_i$ and the best value $f_i^{min} = ideal_i$ for each $i = 1, 2, \cdots k$. These values are determined by optimizing each function individualy. Therefore, k single - objective problems will be solved, using Algorithm 12, in a first place in order to determine f^{min} and f^{max}.

For each λ combination, *GRASP* is applied in order to find the best solution that minimizes the *ASF*. During the procedure, a set of nondominated solutions is saved at variable $\wp_L$, which will be used to update the overall approximation of the Pareto front $\wp$.

Hence, this multiobjective algorithm is reduced to solve several single - objective problems, whose objective function is given by the *ASF* as detailed in Algorithm 15:

Algorithm 15 Wierzbicki achievement scalarizing function approach.

function ASF(V, $numIter$,$f = (f_1, f_2, \cdots f_k)$, m)
  Set $\wp = \emptyset$ as the approximation of the Pareto optimal set.
  Minimize each function f_i
$s_i = GRASP(V, numIter, f_i)$
  Define $z^* = f^{min}$ and $z^{nad} = f^{max}$.
  for $L \in 1, 2, \cdots m$ **do**
    Generate a random combination of weights: λ
    Use λ, f^{min} and f^{max} to define the ASF.
    Set $\wp_L = \emptyset$.
    $\wp_L = GRASP(V, numIter, f_i)$
    Set $\wp = \wp_L \cup \{\wp\}$ and update $\wp$.
  end for
  return $\wp$
end function

The larger the value of m, the more accurate the approximation is expected to be. However, the computational cost will increase, so the process is divided into two stages. In the first trial, a small value of m will compute a first approximation of the Pareto front $\wp$ and, next, this approach can be improved by applying any of the algorithms proposed in Section 3.2.3.

3.2.3 Resources to improve the approximation of the Pareto set

When a first approach of the Pareto front has been obtained applying any of the methods described in Section 3.2, an additional search for nondominated solutions is launched in order to improve this approximation. Two alternatives have been developed in this work, based on the well known metaheuristics: *Path Relinking* and *Variable Neighborhood Search (VNS)*.

Multiobjective Path Relinking

A first approach of the Pareto front, $\wp$, has been generated by *MultiObjective GRASP* detailed in the previous section (Section 3.2). The incorporation of a post - optimization Path Relinking attempts to find new elements of this set by evaluating a restricted space given by the feasible moves that transform one solution into another. It has been established in Section 3.1.2 how to proceed to define a *Path Relinking* and some references have been provided to apply it to *Vehicle Rouring Problems (VRP)*. Due to the multiple criteria considered in this work, the definition of a guiding objective function is required to proceed with this metaheuristic. Then, it is important to state the elements that characterize its implementation, such as the neighborhood operators and the definition of the reference set and the distance measure, as well as the selection criteria to determine the initial solution, S^i, and the guiding solution, S^G.

Reference Set The approximation of the Pareto front ($\wp$) is considered as the reference set. In this case, both of the solutions that take part in the *Path Relinking* procedure, belong to this set. The attributes that characterize the elements of this set are given by the relation of dominance. So that one will try to transform solution S^i into S^G for a given direction function, but the properties of S^i might contribute to other function which makes it better in comparison with S^G.

Selecting solutions A direction must be chosen in order to perform a *forward Path Relinking* strategy. When a guiding function is defined, f_G, for each pair of solutions from the reference set, the one with the best value on the guiding function will be S^G and so, the other one will be the initial solution. Taking into account the multiobjective character of the problem, the elements of the Reference Set are ordered in an increasing fashion by their value in f_G. It reduces the searching space when applying *Path Relinking* between each pair of consecutive solutions while no other nondominated solution has been found.

Distance measure The distance considered is the *symmetric difference*. This measure, denoted by $\Delta(S, S^G)$ and explained in 3.1.2, consists of determining the elements that are in S^G and not in S^i, taking into consideration their positions. It also indicates the number of moves required to transform S^i into S^G.

Neighborhood Operators Two operators are commonly used in *Vehicle Routing Problem*: *swap* and *shift*. Then, at each step, the neighborhoods generated by these two operators are evaluated and the best move is performed. Note that moves to solutions where the value function is worse are permited, in order to reduce the distance, in terms of the symmetric difference, to the guiding solution.

Let $\wp = \{z_1 \ldots z_p\}$ be the approximation of the Pareto front, ordered by one of the function's value. Without loss of generality, suppose $f_1(z_i) < f_1(z_{i+1})$. First we identify the initial solution and the guiding solution as: $S^i = z_{i+1}$ and $S^G = z_i$. The symmetric difference is computed next, for each pair of routes between both solutions. It can be represented by a matrix whose rows are given by routes that form S^i and columns by the routes from S^G. Two routes, one from each solution, with the largest number of elements in common, are selected (Algorithm 16). If this number coincides with the length of any of the routes, it means that both routes are equal alredy, so the next pair of routes with minimum symmetric difference is chosen.

Algorithm 16 Selection of routes at Path Relinking

function ROUTES SELECTION(S^G, S^i, f_G)

  Suppose $\wp$ is ordered by decreasing values of f_G.

  Define route to transform: R^T

  Define guiding route: R^G

  $k_{min} = \infty$

  for each Route $r_G \in S^G$ **do**

    for each Route $r_i \in S^i$ **do**

      k = min$\{|\Delta(r_i, r_G)|\}$

      if $k < k_{min}$ **then**

        $k_{min} = k$

        $R^G = r_G$

        $R^T = r_i$

      end if

    end for

  end for

  return R^G, R^T, k_{min}

end function

In any case, if both routes have the same length, the exploration of the path takes place within the neighborhood generated by the *shift operator*. Otherwise, the route that belongs to S^G determines if nodes must be inserted or removed from its paired route from S^i by the *swap operator*. These moves between routes is detailed in Algorithm 17.

Algorithm 17 Path Relinking between routes.

function PATH RELINKING BY ROUTES(R^G, R^T, k_{min})

if $R^G.length = R^T.length$ **then**

Explore the path between R^G and R^T checking if the neighbors are Pareto optimal, given $\wp$

Perform best - improvement using *shift - operator*

else

if $R^G.length > R^T.length$ **then**

Explore the path between R^G and R^T checking if the neighbors are Pareto optimal, given the Pareto Set $\wp$

Perform best - improvement using *swap - operator* by incorporating nodes from $\Delta(R^G, R^T)$ to R^T

else

Explore the path between R^G and R^T checking if the neighbors are Pareto optimal, given the Pareto Set $\wp$

Perform best - improvement using *swap - operator* by removing nodes given by $\Delta(R^G, R^T)$ from R^T

end if

end if

Update $\wp$

return $\wp$

end function

Then, this *Path Relinking* is launched using every function to guide the search, individually. For each pair of solutions from the reference set, $\wp$, this function will determine which solution is considered as the *guiding solution* (S^G). In any case this strategy uses the distance measure described in Sorensen and Schittekat (2013) to evaluate the progress of the process and two common operators in *VRP* to transform a nondominated solution into S^G, while the Reference Set keeps updated with all the nondominated solutions visited.

Multiobjective Variable Neighborhood Search

The concept of *Variable Neighborhood Search (VNS)* has been extrapolated to solve multiobjective combinatorial optimization problems (Duarte et al., 2015). Due to the multiobjective character of the algorithm, it is necessary to check if a visited solution is nondominated within the current approximation of the Pareto front, even when it does not generate any improvement on the objective function under consideration. Note that local search is applied for every objective function, one at a time.

In the present study, we implement *VNS* local search, as well as an additional approach which considers a reference point when dealing with more than one objective function. In what follows, the former will be denoted *VNS* and the latter VNS_{ref}. Nevertheless, when considering multiple criteria, a new approach of *VNS* is developed here to find nondominated solutions. Each solution obtained during the construction of the first approximation and within a given a fixed ratio, Δ, is checked for its inclusion into the nondominated solutions set or, otherwise, into the most promising solutions set (PS).

Definition: Given a multiobjective problem with $f = \{f_1, \cdots f_k\}$, one can define the *efficient region*, Ω, for a given pair of nondominated solutions S^A and S^B, as the set of points potentially efficient.

Definition: Given a multiobjective problem and a pair of nondominated solutions S^A and S^B defining an efficient region, Ω, we define the Δ - *efficient region* as the set of points $x \in S$ such that:

$$\Delta - region = \{x \in S : d(x, \omega) < \Delta\} \tag{3.13}$$

where d denotes the distance L_∞.

Definition: The solutions that belong to the Δ - efficient region are defined as *promising solutions*.

For the sake of simplicity, an illustrative example with 2 objectives, is detailed in Figures 3.1 and 3.2 to identify these regions.

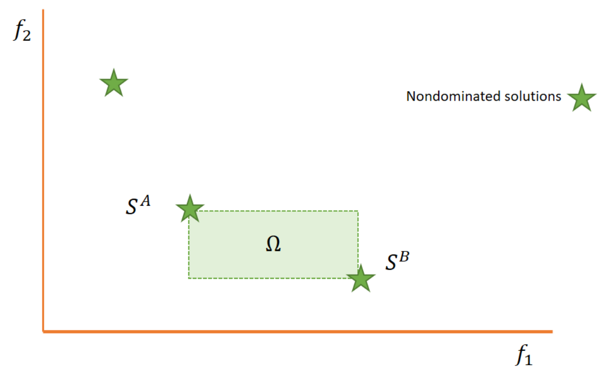


Figure 3.1: Identifying the efficient region, Ω

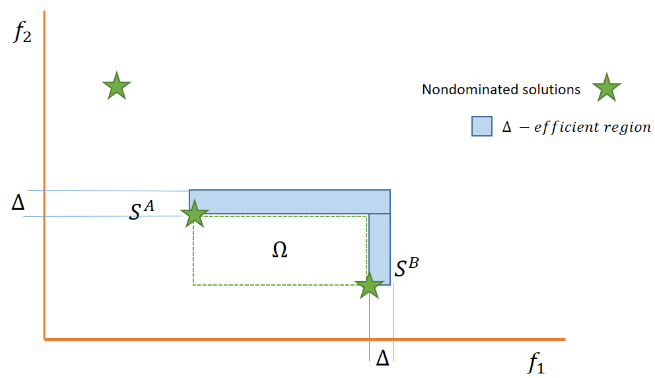


Figure 3.2: Identifying Δ - efficient region

Consider the set, $\wp$, of nondominated solutions and PS with the promising solutions visited. Next, for each pair of solutions, S^A and S^B in $\wp$, we define its *ideal* as the vector whose components are the best value for every objective, as detailed in Figure 3.3. This ideal corresponds to the reference point, Ref^{AB}.

Then, VNS_{ref} considers if a solution s in PS belongs to the Δ - efficient region defined by S^A and S^B. If so, then VNS_{ref} is applied in order to minimize the distance L^∞ between s and the reference point Ref^{AB}. Note that the objective values are normalized between 0 and 1 in the formulation of distance, in order to avoid any possible bias due to the magnitudes.

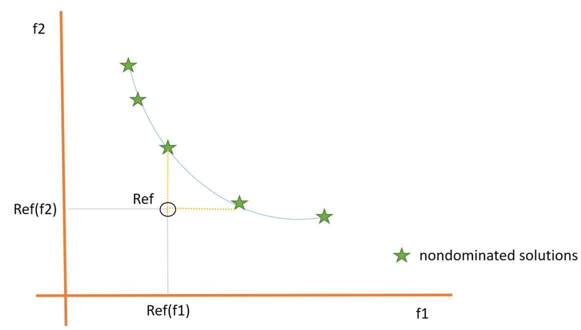


Figure 3.3: Defining the reference point in VNS_{ref}

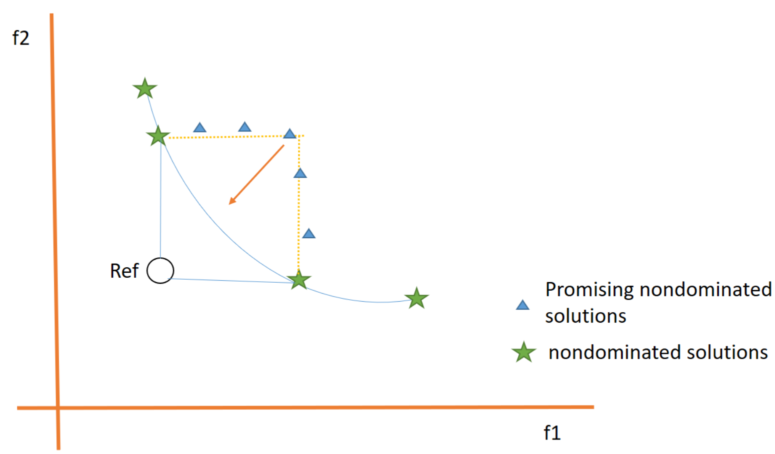


Figure 3.4: VNS_{ref}

Then, considering two nondominated solutions, S^A and S^B, the reference point Ref^{AB} and a promising solution s, the algorithm VNS_{ref} aims to minimize the distance between s and Ref^{AB}. It can be described as follows (Algorithm 18):

Algorithm 18 Variable Neighborhood Search with reference point

function $VNS_{ref}(S^A, S^B, \{N_1(S), \cdots, N_k(S)\}, d, max_{iter}, n_{max})$

$n = 1$

while $n \leq n_{max}$ **do**

$s' = shake(s, n)$

$s' = VNS(\text{s}, \{N_1(S), \cdots, N_k(S)\}, d, max_{iter})$

if $f(S') < f(S)$ **then**

$S = S'$

end if

$n = n + 1$

end while

return S

end function

Thus, two alternatives derive from applying the scheme of *VNS* as a local search procedure in order to solve a multiobjective combinatorial optimization problem. One of them just apply single - objective *VNS* focusing on one objective at a time, as detailed in Algorithm 11; while the other defines a reference point and the local search is run in order to minimize the distance to it. Both of this approaches maintain a set of nondominated solutions visited on the process.

3.2.4 Experimental Results

Christofides and Eilon (1969), Christofides et al. (1979) and Uchoa et al. (2017) have contributed with a wide set of instances to test heuristics for the VRP. In particular, a random selection of 25 of these instances have been used to test the performance of the mono - objective version of the GRASP metaheuristic proposed. Different sizes and customer distributions have been included in this sample set, as observed in Table 3.1, that reflects the characteristics of every instance considered. Note that n represents the number of nodes considered in the instance and v the number of vehicles available with a maximum capacity Q.

Also, the best known value (BKV) is included in this table for the minimum total distance (f_1).

Name	n	v	Q	BKV(f_1)
E-n101-k14	101	14	112	1082.65
E-n101-k8	101	8	200	826.14
E-n51-k5	51	5	160	524.94
E-n76-k10	76	10	140	837.36
E-n76-k14	76	14	100	1026.71
E-n76-k7	76	7	220	687.60
E-n76-k8	76	8	180	740.66
M-n200-k16	200	16	200	1294.67
M-n200-k17	200	17	200	1294.89
X-n1001-k43	1001	43	131	72742.00
X-n120-k6	120	6	21	13329.42
X-n143-k7	143	7	1190	15697.06
X-n162-k11	162	11	1174	14138.58
X-n186-k15	186	15	974	24154.29
X-n190-k8	190	8	138	16985.86
X-n233-k16	233	17	631	19239.22
X-n294-k50	294	51	285	47167.00
X-n303-k21	303	21	794	21744.00
X-n384-k52	384	53	564	66081.00
X-n449-k29	449	29	777	55358.00
X-n573-k30	573	30	210	50780.00
X-n655-k131	655	131	5	106780.00
X-n701-k44	701	44	87	82292.00
X-n733-k159	733	160	25	136366.00
X-n895-k37	895	38	1816	54172.00

Table 3.1: Instances sample description.

Note that our goal is to introduce a competitive technique which, applying a GRASP metaheuristic, generates a good approximation of the Pareto optimal set. Combining the previously described algorithms, up to 6 different approaches have been considered in order to find the best approximation:

1. GRASP Pure Ordered and Path Relinking.
2. GRASP Pure Random and Path Relinking.
3. GRASP Pure Ordered and VNS_{ref}.
4. GRASP Pure Random and VNS_{ref}.
5. GRASP to optimize ASF and Path Relinking.
6. GRASP to optimize ASF and VNS_{ref}.

The performance of all these methods is analyzed in this section, introducing a second objective, f_2, which is defined as the minimization of the longest route. To the best of our knowledge, there does not exist any set of instances that contemplates the biobjective perspective with these two criteria. Then, the given set of instances has been analyzed for the resulting biobjective problem, in order to test the performance of the methodologies designed and decide which would be the most convenient to apply in a multiobjective problem with more objectives, as the one introduced in Section 4. Also, a previous analysis solving the single - objective problem is studied first to test the quality of the *GRASP* proposed. One of the alternatives to evaluate the quality of a heuristic is to compare the results with the best known value or the optimum of the instance. To measure the deviation of the obtained value, f_h to this reference value, f_{opt}, the following metric is used (3.14):

$$gap(\%) = \frac{f_h - f_{opt}}{f_{opt}} \cdot 100 \tag{3.14}$$

To adjust the parameter α for the construction phase of *GRASP*, an experiment has been launched for 1000 iterations. Four scenarios are studied in this experiment. First α is randomly generated at each construction and the other three assign a fixed value for the parameter: $\alpha = 0.25$, $\alpha = 0.5$ and $\alpha = 0.75$. The resulting gaps, between the solutions obtained and the reference value, are displayed in Tables 3.2 and 3.3. Note that BKV(f_2) represents the ideal value of a completely balanced solution, though the existence of such a solution is not guaranteed.

Instance	BKV(f_1)	α Random	time(s)	$\alpha = 0.25$	time(s)	$\alpha = 0.5$	time(s)	$\alpha = 0.75$	time(s)
E-n101-k14	1082.65	7.06	0.92	12.52	0.833	9.92	0.833	6.51	0.966
E-n101-k8	826.14	4.84	0.866	7.55	0.91	7.19	0.91	7.44	0.945
E-n51-k5	524.94	5.87	0.239	5.71	0.213	4.22	0.213	4.23	0.221
E-n76-k10	837.36	7.49	0.575	10.3	0.449	10.56	0.449	7.13	0.481
E-n76-k14	1026.71	7.18	0.394	16.18	0.435	10.4	0.435	7.51	0.464
E-n76-k7	687.6	3.97	0.448	7.45	0.489	5.15	0.489	4.52	0.529
E-n76-k8	740.66	4.41	0.431	10.68	0.618	8.75	0.618	7.08	0.523
M-n200-k16	1294.67	17.74	3.956	28.51	4.335	21.65	4.335	21.33	5.522
M-n200-k17	1294.89	12.47	3.929	19.36	4.126	15.63	4.126	11.77	4.42
X-n1001-k43	72742	12.01	313.867	20.34	3199.487	16.72	3199.487	14.15	303.704
X-n120-k6	13329.42	4.01	1.456	11.13	1.527	5.92	1.527	4.41	1.568
X-n143-k7	15697.06	8.35	2.137	14.59	2.212	12.34	2.212	10.35	2.241
X-n162-k11	14138.58	7.58	2.453	12.94	2.601	8.57	2.601	9.64	2.66
X-n186-k15	24154.29	10.66	3.326	14.54	3.595	11.42	3.595	10.66	3.582
X-n190-k8	16985.86	5.55	4.77	11.54	4.565	8.54	4.565	5.48	4.658
X-n233-k16	19239.22	12.25	9.531	18.24	6.5	14.23	6.5	12.89	6.527
X-n294-k50	47167	10.03	11.063	26.76	15.354	18.23	15.354	13.08	57.644
X-n303-k21	21744	11.67	11.096	16.55	18.318	13.22	18.318	13.13	17.091
X-n384-k52	66081	16.24	33.653	24.94	34.324	21.45	34.324	14.72	38.232
X-n449-k29	55358	9.96	32.113	18.61	30.169	13.3	30.169	11.68	31.186
X-n573-k30	50780	6.94	60.145	9.19	67.487	7.34	67.487	7.13	66.873
X-n655-k131	106780	4.27	102.772	11.49	123.489	7.43	123.489	5.62	116.873
X-n701-k44	82292	8.44	715.037	15.14	111.721	11.92	111.721	10.05	109.799
X-n733-k159	136366	9.43	155.103	203.59	177.683	203.59	177.683	203.59	186.715
X-n895-k37	54172	12.96	226.933	15.51	227.069	16.78	227.069	13.8	220.791

Table 3.2: Constructions analysis to optimize f_1.

Instance	BKV(f_2)	α Random	time(s)	$\alpha = 0.25$	time(s)	$\alpha = 0.5$	time(s)	$\alpha = 0.75$	time(s)
E-n101-k14	77.33	68.53	0.644	39.91	0.559	39.91	0.524	39.91	0.521
E-n101-k8	103.27	21.31	0.589	20.16	0.545	20.16	0.551	20.16	0.552
E-n51-k5	104.99	10.31	0.143	15.49	0.126	15.49	0.124	15.49	0.122
E-n76-k10	83.74	22.64	0.324	18.48	0.297	18.48	0.282	18.48	0.281
E-n76-k14	73.34	28.45	0.241	29.06	0.26	29.06	0.295	29.06	0.269
E-n76-k7	98.23	13.66	0.285	12.88	0.284	12.88	0.302	12.88	0.295
E-n76-k8	92.58	11.32	0.271	16.36	0.277	16.36	0.296	16.36	0.287
M-n200-k16	80.92	60.95	2.527	61.09	2.579	61.09	2.673	61.09	2.652
M-n200-k17	76.17	97.36	2.525	39.91	2.641	39.91	2.782	39.91	2.668
X-n1001-k43	1691.67	117.11	180.04	57.13	215.604	57.13	203.399	57.13	197.182
X-n120-k6	2221.57	34.12	0.847	24.76	0.918	24.76	0.908	24.76	0.898
X-n143-k7	2242.44	51.16	1.279	27.71	1.398	27.71	1.407	27.71	1.352
X-n162-k11	1285.33	30	1.515	22.36	1.638	22.36	1.695	22.36	1.61
X-n186-k15	1610.29	96.27	2.143	49.69	2.302	49.69	2.383	49.69	2.33
X-n190-k8	2123.23	23.54	2.642	16.05	2.834	16.05	2.876	16.05	2.79
X-n233-k16	1131.72	38.45	3.707	36.95	3.933	36.95	3.989	36.95	3.971
X-n294-k50	924.84	56.48	6.386	61.14	7.241	61.14	7.845	61.14	7.106
X-n303-k21	1035.43	154.33	6.942	39.4	7.734	39.4	7.481	39.4	7.256
X-n384-k52	1246.81	73.87	17.419	70.48	21.947	70.48	20.868	70.48	22.893
X-n449-k29	1908.9	53.15	18.84	53.54	21.327	53.54	21.3	53.54	20.54
X-n573-k30	1692.67	73.35	35.408	65.86	39.02	65.86	41.885	65.86	38.892
X-n655-k131	815.11	98.33	49.885	116	56.077	116	59.508	116	54.941
X-n701-k44	1870.27	93.59	57.589	50.78	68.036	50.78	63.495	50.78	64.84
X-n733-k159	852.29	88.14	75.742	72.92	87.942	72.92	88.284	72.92	89.057
X-n895-k37	1425.58	160.67	130.757	60.78	153.779	60.78	149.087	60.78	147.322

Table 3.3: Constructions analysis to optimize f_2.

Results show no evidence of large differences between the GAP values obtained using random or a pre - defined value of alpha. Then, based on the running time, we will consider α randomly generated at each construction.

The order of application of the neighborhoods has also been tested, including a random order for every execution of the *Variable Neighborhood Search (VNS)* as proposed in Vidal et al. (2014). The performance of these combinations has guided us to apply them in a sequence such as it first optimizes each route, by inverting the order of a subchain of nodes (N_2) or moving them directly to another position within the same route(N_3), and finally it tries to move nodes from one route to another (N_1). In this fashion, *VNS* allows the improvement of a route length whenever a new node has been inserted in the current route. The GAP(%) to BKV(f_1) obtained for the sample of instances is summarized in Table 3.4.

Instance	GAP(%)	time(s)
E-n101-k14	4.18	1.699
E-n101-k8	2.11	3.295
E-n51-k5	1.84	0.615
E-n76-k10	4.15	0.936
E-n76-k14	4.12	0.762
E-n76-k7	2.8	1.772
E-n76-k8	3.69	1.212
M-n200-k16	11.46	10.13
M-n200-k17	7.88	9.892
X-n1001-k43	9.75	486.749
X-n120-k6	2.7	7.765
X-n143-k7	4.36	12.532
X-n162-k11	4.54	8.485
X-n186-k15	7.11	8.539
X-n190-k8	3.63	26.205
X-n233-k16	9.14	21.181
X-n294-k50	8.27	17.963
X-n303-k21	8.05	28.853
X-n384-k52	12.48	41.707
X-n449-k29	7.76	58.891
X-n573-k30	4.93	182.278
X-n655-k131	3.99	142.773
X-n701-k44	7.56	170.335
X-n733-k159	8.5	231.315
X-n895-k37	9.54	381.425

Table 3.4: Local search to optimize f_1.

Note that the computational time comprehends both, the construction and the local search phase. Since the value of BKV(f_2) is not representative, we do not consider it relevant to include another table. Here, the construction algorithm used is the same for the optimization of the *Achievement Scalarizing Function* given by Eq.(3.15) setting $\lambda = 1$ or directly optimizing the function f_1.

$$\max\{\lambda_1 \cdot (f_1(\mathbf{x}) - R_1), (1-\lambda_1)\cdot(f_2(\mathbf{x}) - R_2)\} + \rho \cdot \sum_{i=1}^{2} f_i(\mathbf{x}) \tag{3.15}$$

Once the best parameters have been discussed, as well as the neighborhood combination for the *VNS* algorithm, it is time to contrast the different algorithms designed to obtain the approximation of the Pareto front for the biobjective problem. Recall that these algorithms use *GRASP* and *Path Relinking* metaheuristics, including *VNS* in the local search phase for the single - objective and biobjective improvements.

Considering the same sample of instances analyzed before, the different approximations of *GRASP*, as described in Section 3.2, have been launched with the best parameters obtained from the single - objective analysis. Finally, the performance of the algorithms *GRASP Pure Ordered*, which is denoted *M1*, and *GRASP Pure Random*, as *M2*, as well as the approximation obtained when minimizing the *ASF*, for different combinations of λ and denoted by *M3*, is tested. Clearly, the number of nondominated solutions obtained with the last procedure is larger, because of the nature of the procedure. Those approaches, inspired by Martí et al. (2015), minimize one objective function at each time, so the extremities of the Pareto front approximation will be more populated and finding new nondominated solution will be more difficult, a priori. However, the variation of parameter λ, in the other approach, allows the algorithm to "sweep" the range of efficient front between the ideal and nadir point estimated, which determine the boundaries of the Pareto or efficient front.

Zitzler (1999) introduced the *coverage metric function* where given 2 approximation sets A and B, it returns the fraction of solutions in B that are weakly dominated by solutions in A. Its formulation can be described as follow:

$$C(A,B) = \frac{|\{b \in B : \exists a \in A, \text{such that} \preceq b\}|}{|B|} \tag{3.16}$$

where $a \preceq b$ denotes that "a dominates b".

This function is used in order to compare the quality of the different approaches obtained for the Pareto front. Its value indicates that the closer C(A,B) is to 1, the larger proportion of solutions in B will be dominated by solutions in A. Then, it could imply a better quality of the approximated front A against B.

Table 3.5 shows a comparison between the different approaches considered to generate a first approximation of the Pareto front.

Instance	M1	M2	M3	C(M1,M2)	C(M1,M3)	C(M2,M1)	C(M2,M3)	C(M3,M1)	C(M3,M2)
E-n101-k14	14	11	5	0	0	0	0	1	0
E-n101-k8	8	8	6	0	0	0	0	0	0
E-n51-k5	3	1	6	0	0	0	0	1	1
E-n76-k10	7	4	3	0	0	0	0	0	1
E-n76-k14	7	2	2	0	0	0	0	1	1
E-n76-k7	6	11	5	1	0	0	0	0	1
E-n76-k8	7	7	3	0	0	0	0	1	1
M-n200-k16	2	2	4	1	0	0	0	0	0
M-n200-k17	8	9	8	0	0	0	0	1	1
X-n1001-k43	22	29	12	0	0	0	0	1	0
X-n120-k6	16	24	12	0	0	0	0	0	0
X-n143-k7	19	23	7	0	0	0	0	0	0
X-n162-k11	5	8	5	0	0	0	0	0	0
X-n186-k15	14	14	8	0	0	0	0	1	0
X-n190-k8	28	34	10	0	0	0	0	0	0
X-n233-k16	17	16	4	0	0	0	0	0	0
X-n294-k50	9	5	11	0	0	0	0	1	1
X-n303-k21	11	15	8	0	0	0	0	0	0
X-n384-k52	3	2	4	0	0	1	0	1	1
X-n449-k29	8	13	13	0	0	0	0	0	0
X-n573-k30	21	17	14	0	0	0	0	1	0
X-n655-k131	6	5	4	0	0	0	0	1	1
X-n701-k44	15	7	11	0	0	0	0	1	1
X-n733-k159	4	3	5	0	0	0	0	1	1
X-n895-k37	17	13	7	0	0	1	0	0	0

Table 3.5: Coverage metric comparison between Multiobjective GRASP approaches.

This table also contains the number of nondominated solutions found by each method. One may conclude, from these values, that applying the technique derived from minimizing Wierzbicki's Achievement Scalarizing Function provides better results than any other, since the coverage values reach the maximum value at most of these instances when comparing this method to the other ones. However, when comparing the pure constructions, between each other, randomness seems to amplify the range of nondominated solutions obtained in the objective space. In these cases, the coverage function justifies the variation in the cardinality of the sets of nondominated solutions, obtained with the different multiobjective approaches.

An attempt to improve this first approximation is done by applying *Path Relinking* (denoted as "PR") and the other approach consists of using a reference point (VNS_ref) denoted as "VNS" local search procedure as detailed at Figure 3.4. The results of these improvement alternatives are displayed in Tables 3.6, 3.7 and 3.8 for each initial constructive GRASP. Notice that index i corresponds to an improvement of Mi.

Instance	PR_1	VNS_1	C(PR_1, M1)	C(VNS_1, M1)	C(PR_1,VNS_1)	C(VNS_1,PR_1)
E-n101-k14	12	14	0	0	0	0
E-n101-k8	8	9	0	0	0	0
E-n51-k5	3	3	0	0	0	0
E-n76-k10	7	7	0	0	0	0
E-n76-k14	7	4	0	0	0	0
E-n76-k7	6	6	0	0	0	0
E-n76-k8	7	7	0	0	0	0
M-n200-k16	2	2	0	0	0	0
M-n200-k17	11	8	0	0	0	1
X-n1001-k43	23	22	0	0	0	0
X-n120-k6	20	16	1	0	0	0
X-n143-k7	20	19	0	0	0	0
X-n162-k11	5	5	0	0	0	0
X-n186-k15	17	14	0	0	0	0
X-n190-k8	20	23	0	0	0	0
X-n233-k16	20	17	0	0	0	0
X-n294-k50	11	9	0	0	0	0
X-n303-k21	11	11	0	0	0	0
X-n384-k52	3	3	0	0	0	0
X-n449-k29	8	8	0	0	0	0
X-n573-k30	24	21	0	0	0	0
X-n655-k131	7	6	0	0	0	0
X-n701-k44	16	15	0	0	0	0
X-n733-k159	5	4	0	0	0	0
X-n895-k37	17	17	0	0	0	0

Table 3.6: Results and coverage for the improved biobjective GRASP Pure Ordered with biobjective Path Relinking and VNS_{ref}.

Instance	PR_2	VNS_2	C(PR_2, M2)	C(VNS_2, M2)	C(PR_2,VNS_2)	C(VNS_2,PR_2)
E-n101-k14	11	11	0	0	0	0
E-n101-k8	9	8	0	0	0	0
E-n51-k5	1	1	0	0	0	0
E-n76-k10	4	4	0	0	0	0
E-n76-k14	3	2	0	0	0	0
E-n76-k7	5	10	0	0	0	0
E-n76-k8	7	8	0	0	0	0
M-n200-k16	2	2	0	0	0	0
M-n200-k17	4	9	1	0	1	0
X-n1001-k43	29	29	0	0	0	0
X-n120-k6	27	24	0	0	0	0
X-n143-k7	26	23	0	0	0	0
X-n162-k11	8	8	0	0	0	0
X-n186-k15	15	14	0	0	0	0
X-n190-k8	38	34	0	0	0	0
X-n233-k16	15	16	0	0	0	0
X-n294-k50	6	5	0	0	0	0
X-n303-k21	16	15	0	0	0	0
X-n384-k52	2	2	0	0	0	0
X-n449-k29	12	13	0	0	0	0
X-n573-k30	20	17	0	0	0	0
X-n655-k131	6	5	0	0	0	0
X-n701-k44	7	7	0	0	0	0
X-n733-k159	3	3	0	0	0	0
X-n895-k37	14	13	0	0	0	0

Table 3.7: Results and coverage for the improved biobjective GRASP Pure Random with biobjective Path Relinking and VNS_{ref}.

These results (Tables 3.6 and 3.7) show how none of the improvement methodologies finds any new nondominated solution, based on the approximation of the Pareto front obtained with metaheuristic M1 or M2.

Instance	PR_3	VNS_3	C(PR_3, M3)	C(VNS_3, M3)	C(PR_3,VNS_3)	C(VNS_3,PR_3)
E-n101-k14	5	5	0	0	0	0
E-n101-k8	4	9	0	0	0	0
E-n51-k5	5	4	0	0	0	0
E-n76-k10	3	3	0	0	0	0
E-n76-k14	2	2	0	1	0	0
E-n76-k7	5	4	0	0	0	0
E-n76-k8	3	3	0	0	0	0
M-n200-k16	3	4	0	0	0	0
M-n200-k17	10	9	0	0	0	0
X-n1001-k43	15	12	0	0	0	0
X-n120-k6	11	13	0	0	0	0
X-n143-k7	9	10	0	0	0	0
X-n162-k11	4	5	0	0	0	0
X-n186-k15	10	8	0	0	0	0
X-n190-k8	14	13	0	0	0	0
X-n233-k16	3	5	0	0	0	0
X-n294-k50	8	12	0	0	0	0
X-n303-k21	5	11	0	0	0	0
X-n384-k52	1	4	0	0	0	0
X-n449-k29	6	15	0	0	0	0
X-n573-k30	19	25	0	0	0	0
X-n655-k131	5	4	0	0	0	0
X-n701-k44	10	11	0	0	0	0
X-n733-k159	6	5	0	0	0	0
X-n895-k37	17	7	0	0	0	0

Table 3.8: Results and coverage for the improved Wierzbicki's algorithm with biobjective Path Relinking and VNS_{ref}.

It can be observed how *Path Relinking* varely improve the results obtained from all the algorithms used to construct the first approximation. The number of nondominated solutions usually increases, however, in some cases the objective values are improved so that previous nondominated solutions become dominated, so this number degenerates or stays constant. Note that if only one nondominated solution has been found, the application of a biobjective *Path Relinking* does not make any sense.

Instance	C(PR_2,PR_1)	C(VNS_1,VNS_2)	C(VNS_3,PR_1)	C(VNS_3,PR_2)	C(VNS_3,VNS_1)	C(VNS_3,VNS_2)	C(PR_3,VNS_1)	C(PR_3,VNS_2)
E-n101-k14	0	0	1	0	1	0	1	0
E-n101-k8	0	0	0	0	0	0	0	1
E-n51-k5	0	0	1	1	1	1	1	1
E-n76-k10	0	0	0	1	0	1	0	1
E-n76-k14	0	0	1	1	1	1	1	1
E-n76-k7	0	1	0	1	0	1	1	1
E-n76-k8	0	0	1	1	1	1	1	1
M-n200-k16	0	1	0	0	0	0	0	0
M-n200-k17	1	0	1	0	1	1	1	1
X-n1001-k43	0	0	1	0	1	0	1	0
X-n120-k6	0	0	0	0	0	0	0	0
X-n143-k7	0	0	0	0	0	0	0	0
X-n162-k11	0	0	0	0	0	0	0	0
X-n186-k15	0	0	1	0	1	0	1	1
X-n190-k8	0	0	0	0	0	0	0	0
X-n233-k16	0	0	0	0	0	0	0	0
X-n294-k50	0	0	1	1	1	1	1	1
X-n303-k21	0	0	0	0	0	0	0	0
X-n384-k52	1	0	1	1	1	1	1	1
X-n449-k29	0	0	0	0	0	0	0	1
X-n573-k30	0	0	1	0	1	0	1	0
X-n655-k131	0	0	1	1	1	1	1	1
X-n701-k44	0	0	1	1	1	1	1	1
X-n733-k159	0	0	1	1	1	1	1	1
X-n895-k37	1	0	0	0	0	0	0	0

Table 3.9: Coverage metrics comparison between some Multiobjective approaches proposed.

Finally, Table 3.9 shows the comparison between the different methodologies. It reflects how VNS_3 obtains, in general, better results than any other approach. This methodology corresponds to a first approximation obtained with Wierzbicki's ASF improved with VNS_{ref}.

3.3 INTERACTIVE APPROACH

Interactive methods have been defined as a tool that provides alternatives, according to the preferences of the *Decision Maker (DM)*, among a set of feasible solutions. One of the main advantages of using interactive methods is that DM's preferences can be gradually incorporated, or modified, along the decision process. This favors a continuous and iterative interaction between the analyst and the DM, who becomes an active participant of the solution procedure.

Developping *Decision Support Systems* (DSS) has gained a major attention among the Waste Management community in the last decades (see Section 2.1.2). However, interactive methods bring the opportunity to explore different areas and appreciate real - limits of the current problem, which also improve the decision process. These procedures enable the DM to control the searching steps and, at the end, (s)he will feel more confident with the final decision.

An early approach to interactive methods in *Vehicle Routing Problems (VRPs)* was proposed in Wright (1994), named *Computer Aided System for Planning Efficient Routes (CASPER)*. Later, this method was applied to a snow removal problem (Wang and Wright, 1994), considering the optimization of travel time routes, non - service tour and road homogeneity. The interaction consists of modifying road conditions in terms of the forecast, and then applying *Tabu Search* in order to improve the result. A more sofisticated method is introduced in Iakovou (2001). To optimize the cost and the risk associated to the transportation of petroleum products, while improving the routing system. It first computes the basic nondominated solutions, i.e. the optimum for every objective function.

Just if the DM is not satisfied with any of them, the procedure continues using compromise programming until the DM reaches the most preferred solution. At each step, a linear combination of the objective functions is optimized and the DM compares the new solution with previously obtained nondominated solutions. This method allows moving backwards. Another example tackles a single - objective problem to find the minimal total cost by designing a graphical - user - interface to interact with the DM called *Computarized Routing Using Interactive Seed Entry (CRUISE)* (Baker and Carreto, 2003). At the beginning, all customer locations are displayed. Then, the user selects a single seed customer to indicate the region of operation of a vehicle. More customers can be sequentially selected in a given fashion, also provided by the DM, while the computer program checks that no constraint is being violated. The remaining customers are allocated using *GRASP* metaheuristic. Finally, the user can modify parameters, any of the routes obtained or stop the procedure if (s)he wants to make a manual alteration of the solution process. This Graphical User Interface (GUI) enables the user to incorporate local knowledge, such as constraints envolving route structure or any other hint that might be difficult to program within a heuristic, as well as to manage the results obtained.

The key factor to be considered when designing an interactive method lies on how the information is shared with the DM. According to the type of information asked, the way of incorporating this information and how a new solution is generated at each iteration, they might be classified into different groups (Osiadacz, 1986). Among the interactive procedures developed considering the former criteria, i.e. the type of information asked to the DM, we find *NAUTILUS* family (Miettinen et al., 2010) as a *non - tradeoff* method. An *Achivement Scalarizing Function (ASF)* and a vector, defined by the desired values that the DM would like to achieve for every objective, define Reference Points methods. Wierzbicki (1982) proposed an interactive approach of the reference - point scheme, where the objectives values are normalized by the range defined by the ideal and the nadir points. In order to obtain more efficient solutions, it perturbates the reference point and optimize the resulting ASF. In general lines, given a reference point, it is projected onto the Pareto front by minimizing an ASF which provides the corresponding solution. At each iteration, the DM is asked to provide a new reference point, **R** which will be **achievable** if $\exists x$, a feasible solution such that $f_i(x) \leq R_i$, where f_i are the objective functions $\forall i$. The usual line followed in this procedure can be described in five steps:

Step 1 Set $it = 1$, define a weighting vector **w** and generate an initial solution **(**x^1**,** f^1**)**.

Step 2 If the DM is satisfied with the solution, the procedure *STOPS*; otherwise, $it = it + 1$ and move to *Step 3*.

Step 3 The DM provides a reference point, R^{it}.

Step 4 Generate a set of efficient solutions by perturbating the reference point when optimizing *ASF*.

Step 5 Show the solutions to the DM and go to *Step 2*.

As mentioned before, in Wierzbicki's scheme, weights (λ) have just a normalizing role. Nevertheless, the convergence of a reference point based iterative method might be accelerated if preferential weights are used. These weights can be obtained based on previous iterations or if the DM stablishes a preferential ranking or a relation of preference. Different interpretations have been assigned to the weighting vector, w. Ruiz et al. (2009) provide a wide analysis of the weighting scheme, which establishes the differences between various reference point based methods. Considering that weights represent the relative importance of achieving each given reference value, up to nine different weighted schemes are analyzed in this work to justify their double role in ASFs: as a normalization factor, but also as the relative importance assigned to each objective.

The history (Wierzbicki, 1977, 1979, 1980) and the range of applications of these methods holds the idea of implementing an interface, or *DSS*, that simplifies the information provided during decision making process. First, the DM provides the reference points and the DSS computes a neutral solution. Then, based on the preference information obtained from the reference points, the *DSS* minimizes an achievement function determined by the position occupied by them within an approximation of the ranges of objective functions. The DM is allowed to change the reference point at any time, what permits an exploration of the most interesting part of the Pareto Set. Different alternatives have been proposed to guide the DM through the Pareto optimal set, such as the *Tchebycheff* method (Steuer and Choo, 1983), a visual method called *Pareto Race* (Korhonen and Laakso, 1986), *REF - LEX* for nonlinear problems (Miettinen and Kirilov, 2005), the satisficing trade - off method introduced in Nakayama and Sawaragi (1984) which later on inspired *NIMBUS* (Miettinen, 1999; Miettinen and Mäkelä, 2000; Miettinen et al., 2006) or the *light beam search* (Jaszkiewicz and Slowinski, 1999), among others.

In general, these Interactive methods, when applied to real computationally expensive problems, need metaheuristics in order to generate each new solution. Jaszkiewicz and Branke (2008) analyze some of these approaches focusing on evolutionary algorithms (EA) and highlighting that these strategies will only provide an approximation of the Pareto solution, which is not necessary optimal. Traditional approaches solve every single - optimization problem and show the resulting set of solutions to the DM, who will decide according to his / her satisfaction degree. Additionally, other approaches, known as *Semi - a - posteriori*, generate a set of approximated solutions and, making use of statistical pre - analysis, guarantee a guided control to the most preferred solution. Also, *interactive Multiobjective Metaheuristics* consist of introducing some modifications into the process that allow the DM to interact during the metaheuristic compilation. This is, for example, the aim of the algorithm presented in Molina et al. (2009), *g - dominance*. Given a reference point, they alter the concept of Pareto dominance. First, the objective space is divided according to the location of the solution in relation to the reference point, R. To determine this subdivision, a flag is associated to any solution generated, s, as formulated in Eq. 3.17.

$$Flag_R(s) = \begin{cases} 1 & \text{if } s_i \leq R_i \forall i \\ 1 & \text{if } s_i \geq R_i \forall i \\ 0 & \text{otherwise} \end{cases} \tag{3.17}$$

The relation of dominance defined is given by:

Definition: Given two solution s^1, s^2 and a reference or guiding point g , where $s^1, s^2, g \in \mathbb{R}^p$, it is said that s^1 is **g - dominated** by s^2 if

$$Flag_g(s^2) > Flag_g(s^1)$$

or

$$Flag_g(s^1) = Flag_g(s^2) \text{ and } {s^2}_i \leq {s^1}_i \ \forall i = 1, \cdots p$$

Algorithm 19 g - Dominance interactive procedure

procedure G - DOMINANCE (R, f, s, it)

Evaluate $f(s) = (f_1(s), \cdots, fk(s))$.

Determine $Flag_R(f)$.

if $Flag_R(f) = 0$ **then**

$f_i(s) = f_i(s) + M \ \forall i = 1, 2, \cdots k$.

end if

Include preferences providing a reference point or a choosing a reference solution, s^R.

Compute the new reference point, R^{it+1}, for the next iteration:

$$R^{it+1} = (1 - \theta) \cdot R^{it} + \theta \cdot s^R$$

Use a clustering procedure to select the approximated Pareto solutions to display.

end procedure

where M is a large penalty assigned to those solutions with null flag, so they become dominated. Following this procedure (briefly detailed in Algorithm 19), one can obtain an approximation of the Pareto front around the projection of the reference point, without varying or setting any parameter in the multiobjective solver. The most interesting characteristic of this approach is that it can be easily implemented into any metaheuristic strategy, it just needs to re - define the dominance based on a reference point.

Some techniques like *Pareto Iterated Local Search*, an interactive approach of *multiobjective Simulated Annealing* or reference point methods have been proposed, combined with other techniques that help managing the preferences or dealing with the comparisons between nondominated solutions (Barbosa and Barreto, 2001; Phelps and Köksalan, 2003). For instance, *Pareto Race* (Korhonen and Wallenius, 1988) is a learning - oriented procedure where the DM can freely move around in the Pareto optimal set in order to identify the trade - off that best fits his / her preferences. A visual environment displays the set of available solutions in the given direction, so that the DM is aware in real - time of the continuous changes. This idea is improved in Eskelinen et al. (2010), in the interactive method called *Pareto Navigator*. To begin with, a discrete representation of the Pareto set is given to the DM. The best and worst values for each objective are defined either by the DM or by the ideal and nadir point, respectively. To reduce computational cost, a first approximation of the Pareto optimal set is obtained and the direction of search is set. The DM is continuously informed of the objective values, which are displayed on a bar chart, so (s)he has the opportunity of stopping the process and provide a new reference point by indicating the desired values or choosing a solution within the set available.

Some reference point methods require that the preference information is introduced in terms of a feasible point. However, also modifying the concept of dominance and using Chebyshev preference relation, López-Jaimes and Coello (2014) introduce two alternatives to incorporate this information using either feasible or unfeasible reference points into an achievement scalarizing function.

The large variety of interactive methods makes it difficult to select one for its application to a given problem. With the aim of incorporating some psychological aspects into the interactive process, Miettinen et al. (2010) introduced *NAUTILUS*. The reason is derived from the fact that the final decision might become biased by previous experiences, since the human being does not react equally to gains or losses. Then, a reference - point with no trade - off method is defined. The algorithm begins at the worst scenario, which is either the nadir point or given by the DM, and continuosly approaching the Pareto front in such a way that each iteration dominates the previous one. Along the process, an interface shows the progress, where one is able to observe a constant improvement in every objective function at the same time, so that no trade - off is required. Miettinen and Ruiz (2016) describe *NAUTILUS Framework* which encompasses previous works such as Miettinen et al. (2010, 2015) and Ruiz et al. (2015). In general, the following steps describe how this procedure works:

Step 1. Compute the ideal, z^*, and nadir, z^{nad}, point of the problem. Also, ask the DM for the number of iterations to be carried out, itn. Then, set the current iteration h, and the value function boundaries $f^{1,up}$ and $f^{1,lo}$ to:

$$h = 1; z^0 = f^{1,up} = z^{nad}; f^{1,lo} = z^*; it^1 = itn$$

Step 2. The DM is asked to give preference information, which will be used to determine the weights, λ_i, of the *Achievement Scalarizing Function (ASF)*. Note that λ_i represents the preference information including a normalization, in order to avoid any possible bias effect due to the different magnitudes of the objectives.

Step 3. Define $q = z^{h-1}$, $\lambda_i = \lambda_i^h$ and x^h which is the optimum of the single - objective problem:

$$\min \max \{\lambda_i \cdot (f_i(x) - q_i)\} + \rho \cdot \sum_{i=1}^{k} \frac{f_i(x) - q_i}{z_i^{nad} - z_i^*} \tag{3.18}$$

$$\text{subject to: } \mathbf{x} \in \wp^h \tag{3.19}$$

Also, determine $f^h = f(x^h)$.

Step 4. Compute the new iteration point:

$$z^h = \frac{it^h - 1}{it^h} \cdot z^{h-1} + \frac{1}{it^h} \cdot f^h$$

Step 5. Calculate the bounds for the next iteration by applying an ε - *constraint method*, so that k different problems, defined as Eq.(3.20), must be solved.

$$(P_r) \begin{cases} \text{minimize} & f_r(\mathbf{x}) \\ \text{subject to} & f_j(\mathbf{x}) \leq z_j,\ j = 1, \ldots, k, j \neq r, \\ & \mathbf{x} \in \wp^h, \end{cases} \tag{3.20}$$

Determine, also, the distance from the current point to the Pareto front:

$$d^h = 100 \cdot \frac{\| z^h - z^{nad} \|_2}{\| f^h - z^{nad} \|_2}$$

Step 6. Show the current values to the DM: z_i^h, $[f_i^{h+1,lo}, f_i^{h+1,up}]$ and d^h.

Step 7. Ask if the DM would like to modify the number of iterations and set it^h.

Step 8. Ask if the DM would like to go a step backwards, in that case, go to *Step 10*.

Step 9. If $it^h = 1$, then *stop*. Otherwise, $it^{h+1} = it^h - 1$, $h = h - 1$. If the DM wants to change the preference information, then go to *Step 2*; otherwise, move forward, set $f^h = f^{h-1}$ and go to *Step 4*.

Step 10. Ask if the DM would like to provide new preference information, starting from z^{h-1}. If so, move to *Step 1*. Otherwise, DM can take a shorter step with the same preference information just setting $z^h = 0.5 \cdot z^h + 0.5 \cdot z^{h-1}$ and go back to *Step 4*.

Afterwards, additional features have been incorporated which incur on different *NAUTILUS* approaches. For instance, *E - NAUTILUS* pre - computes a good approximation of the Pareto front instead of generating new solutions at each iteration, to avoid large computational costs during the decision process. Essentially, these differences come from the two modules that link *NAUTILUS*: *Preference elicitation module* and *solver module*.

Preference elicitation module It consists of the phase where the DM is asked for information. Two main approaches have been used until now:

- Choosing one reference solution among a given set of achievable alternatives from z^{h-1}. This set is defined by a fixed number of options that are selected using clustering techniques, as detailed in Ruiz et al. (2015).
- The DM defines a direction of improvement, δ^h. It can be done by directly providing the corresponding specifications, by pairwise comparison or by importance based options (Luque et al., 2009), such as ranking the objectives or assigning different percentages or ratio of improvement to every objective function. Later, δ^h is used to define the weights since it verifies: $\delta^h = \dfrac{1}{\lambda_i^h}$.

Solver module As one may observe, *NAUTILUS* involves solving many single - objective optimization problems. For instance, it needs to optimize Eq.(3.18) several times to generate new Pareto solutions and the problems derived from it, *using ε - constraint method*, to determine the bounds from $z^{h,lo}$. Hence, an efficient and speedy single - objective solver is required to reduce computational cost. Again, two are the options:

1. *Optimization option* consists of optimizing single - objective problems by incorporating exact solvers within the method.
2. The option called *A posteriori*, generates a representative set of solutions from the Pareto Set, or an approximation of it, in a pre - processing step.

Combining these modules, different *NAUTILUS* variants can be developed. Originally, *NAUTILUS* (Miettinen et al., 2010, 2015) introduces preference elicitation by asking the DM for a direction of improvement and an optimization solver is applied in order to direct the search. On the contrary, *E - NAUTILUS* (Ruiz et al., 2015) follows three stages: pre - processing, interactive decision making and post - processing. First, it pre - generates a collection of nondominated solutions as an approximation of the Pareto Set using *evolutionary algorithms*. This gives the opportunity to show more than one alternative in the objective space, as many as the DM is able to handle, so the DM can choose one of them for the next iteration. Then, each iteration reduces the range of possibilities and the interaction finishes, whenever the DM stops and chooses a final solution or the collection is reduced to one item.

Landing on the *MultiObjective Waste Collection Problem*, though it could be generalized to any *MultiObjective Vehicle Routing Problem (MOVRP)*, a new variant of *NAUTILUS* is introduced in this work. Given a general description of *NAUTILUS*, provided above, it is implemented within an interface in order to display the required information to guide the decision making process. The algorithm follows *NAUTILUS* scheme, but it incorporates additional features inspired on *Pareto Navigator* (Eskelinen et al., 2010) which allows a further exploration of the Pareto front, in those directions where the DM feels the most promising solutions could be located.

Step 1. This method starts by generating a discrete representation of the Pareto front ($\wp$), where the ideal and nadir points are defined by the DM as the best and worst scenario considered.

Step 2. The DM gives a reference point, in terms of values or chosing one of the given solutions, which will be projected in $\wp$. This projection determines the direction of movement.

Step 3. In real time, the method generates solutions that sequentialy improve the previous ones while slowly moving along the given direction towards the approximation of the Pareto front, displaying into a bar chart the objectives values. Note that the speed and the direction can be adjusted at any time of the process.

Step 4. Once the objective values of the solution are available, if the DM is satisfied with the solution found, then it is projected to the actual Pareto front. Otherwise, (s)he is asked to provide a new direction or change preferences.

The generation of aproximated Pareto solutions involves solving multiple parametric problems using an *ASF*, which is determined by a reference point and a given direction, so that it is important to find a fast optimizer. The methodology developed in the previous subsections of is now used to optimize *Wierzbicki's ASF*.

To deal with a *Vehicle Routing Problem* other factors must be taken into consideration when designing the GUI. For instance, in the initial step, the approximation of the Pareto front, $\wp$ will have a finite number of solutions. Then, one can detail the performance of each solution if they are displayed into a map using GIS. Notice that it can be time consuming to generate all these images if the size of $\wp$ is too large, so a map with the final solution is generated within the GUI. Also, since an approximated method has been applied to generate the Pareto front, at the last stage, the DM has the possibility to ask for one more exploration. This will start considering the final solution, S, as the reference point and, to minimize the *ASF* in the problem (3.18) for a reduced set of weights, the algorithm will try to find any nondominated solution which dominates S.

As it happens in *E-NAUTILUS*, the proposed variant of *NAUTILUS* which will be denoted *R-NAUTILUS*, will have 3 stages: pre - processing, decision making and post - processing. Let us detail each of them in the following lines.

Pre - processing stage Due to the computational efforts derived from solving a *MOWCP* or *MOVRP* of the given characteristics, a good estimation of the Pareto front is generated by applying one of the algorithms described in Section 3.2. The DM is asked if (s)he wants to be awared when there are 2 or 3 solutions left in the set of reachable solutions.

Decision making stage The method must take into consideration the discrete, and non necessarily convex, character of the problem to tackle. As the procedure advances, the range of reachable solutions in $\wp^h$ from the current point z^h shrinks. Notice that it may happen that the projection of the reference point in $\wp$ does not correspond to any feasible solution, so the evaluation of each solution in $\wp^h$ is computed in order to find the one that minimizes the *Achievement Scalarizing Function (ASF)* as described by Eq. (3.12). This procedure is implemented within a Graphical User Interface (GUI) that permits, at any time, to inform the DM about the ranges that every objective may reach in $\wp^h$ and the evolution of the reachable solutions set. Small steps are taken so that the approximation to $\wp$ looks continuous in the GUI. At any time, the DM is allowed to stop the process, set some limits to the value functions (f^{up} and f^{lo}) or visualize the projection of the current solution in $\wp$. Also, the reference point can be modified and restarting the process from a desired previous iteration point is also permited. Finally, if the DM chooses to be awared if a reduced number of solutions, as many as (s)he is able to handle, are left in $\wp_h$ then, they can be displayed one by one allowing a wider analysis of them.

Post processing stage Once the DM has selected the most preferred solution, it is displayed on a map so that (s)he can evaluate the real performance of the service. At this point, the DM has two options:

- If the DM is satisfied with the chosen solution, the interaction ends.
- If the DM is curious about a possible improvement of the chosen solution, (s)he has the opportunity to ask for a last exploration. In this case, the last solution defines the reference point and, for a set of λ values, the multiobjective algorithm is launched in order to find nondominated solutions that improve the values of the current solution in the ASF (Eq. (3.18)).

In any of these situations, an approximation method has been used to generate $\wp$, so the final solution reached will be nondominated in $\wp$ but its Pareto optimality is not assured. This is why it would be convenient to give the opportunity of a last search, once the DM is satisfied with the chosen solution. However, due to the discrete character of vehicle routing problems, it is not possible to guarantee the Pareto optimality of a given solution by projecting it onto the Pareto optimal front.

Then, one might observe how *R -NAUTILUS* shares some similarities with *E - NAUTILUS* and *Pareto Navigator*. However, some other points differentiate these methods. On the one hand, though *R -NAUTILUS* considers the same pre - processing stage of *E - NAUTILUS*, the former method generates the approximation of the Pareto front using the algorithm developed in this work, while *E - NAUTILUS* applies evolutionary algorithms. Another difference between these methods lies on the number of solutions managed by the DM. In the case of *R - NAUTILUS* is set to one, whereas it is chosen by the DM in *E - NAUTILUS*. On the other hand, *Pareto Navigator* pre - computes a diversified set of approximated Pareto solutions, which is used to guide the search into the most promising area, unlike *R-NAUTILUS* which pre - computes the best possible approximation of the Pareto front. The new approach saves computational effort during the iterative procedure, since there is no need of generating new solutions.

Thus, including the stages explained above and the particularities of *VRP*, *R - NAUTILUS* procedure could be sumarized as follows:

Step 0. Determine an approximation of the Pareto front, $\wp$, using any of the algorithms proposed at Section 3.2. This approximation will contain the ideal, z^*, and nadir, z^{nad}, points of the problem. Where the components of z^{nad} are determined by the worst values obtained for each objective within the elements of $\wp$.

Step 1. Ask the DM for the values desired for each objective, that will define the reference point, q^1. Also, (s)he is asked for a progression speed $s^1 \in \{1,2,3,4,5\}$ and a maximum number of solutions that (s)he is able to handle, nS, in case (s)he wants to be awared when a few points are left in the reachable set $\wp_{max}$. Then, set the curent iteration, $h = 1$, and estimate $f^{1,up}$ and $f^{1,lo}$, of every objective function. These values can also be provided by the DM. By default these parameters are initialized as follows:

$$z^0 = f^{1,up} = z^{nad}; f^{1,lo} = z^*; it^1 = it_{max}; \wp^0 = \wp; nS = 1$$

Step 2. Once the process starts, the DM is able to stop it and update the preference information q^h, display the solution projected in the given direction of improvement or move backwards to a previous iteration which enables to analyze the situation and set new parameters.

Step 3. If $h = 1$ or the DM has changed the reference point to q^h, then compute x^h which is the optimal solution of the single - objective problem formulated in equation (3.18). Otherwise set $q^h = q^{h-1}$ and $x^h = x^{h-1}$. Also, determine $f^h = f(x^h)$.

Step 4. Compute the new iteration point:

$$z^h = \frac{it^{max} - h - 1}{it^{max} - h} \cdot z^{h-1} + \frac{1}{it^{max} - h} \cdot f^h$$

Step 5. Determine the bounds for the next iteration $f^{h,lo}$ and $f^{h,up}$. Also, estimate the progress at the current iteration h as:

$$d^h = 100 \cdot \frac{\| z^h - z^{nad} \|_2}{\| f^h - z^{nad} \|_2}$$

Step 6. Update the subset, $\wp^h$, of reachable solutions from z^h. This set will contain all those feasible solutions $x \in \wp^{h-1}$ that satisfy $f_i^{h,lo} \leq f_i(x) \leq f_i^{h,up} \forall i \in \{1,2,\cdots k\}$.

Step 7. When $it^h = 1$, then the process stops, the final solution is given by f^h and the post - processing stage is applied. Later, different scenarios may arise at this point of the iteration:

- In case the DM would like to go a step backwards, go to *Step 2* using the new information given.
- If the DM wants to change the preference information, then reset q^{h+1} and go to *Step 3*.
- Also, the DM can establish a lower or upper bound for any of the objectives considered. Is this is the case, the current set of reachable solutions is updated, so that only the solutions satisfying the bound are left.
- Otherwise continue the movement towards the Pareto optimal front setting $it^{h+1} = it^{max} - (h+1)$, $h = h+1$, $f^h = f^{h-1}$ and go to *Step 5*.

To sum up, this chapter is based on the description of the methodology developed to face a *MultiObjective Waste Collection Problem (MOWCP)*. This proposal aims to facilitate the decision making for any *MultiObjective Capacitated Vehicle Routing Problem (MOCVRP)*. Solving multiobjective real problems is a challenge for the current society. The interactive method defined uses a pre - computed approximation of the Pareto front, in order to reduce computational costs, so a competitive algorithm is required. Here, different approaches combining *GRASP* and *Path Relinking* metaheuristics have been compared and the best one will be used to solve the real problem at Section 4.

Two different schemes have been used here to obtain the first approximation of the Pareto front. One is based on the structured idea of multiobjective Pure - ordered and Pure - Random MultiObjective *GRASP* proposed in Martí et al. (2015) which consists of generating the approximation by optimizing one objective at a time. The other one, optimizes *Wierzbicki's achievement scalarizing function*, defined in terms of the objectives considered, for a number of weighting combinations. To improve these sets of nondominated solutions, also two alternatives are given here. On the one hand, *Path Relinking* is adapted to face a *MOCVRP*, guiding the search direction to one of the objectives at a time. On the other hand, during the first stage, a set of promising solutions is maintained and a MultiObjective variant of *Variable Neighborhood Search* is designed, in order to reduce their distance to the ideal point defined for each pair of nondominated solutions. As we can see from the results, the combination that makes use of *Wierzbicki's ASF*, and specially combined with VNS_{ref}, provideS the best approximation of the Pareto front.

Finally, based on the family of *NAUTILUS* methods and *Pareto Navigator*, a new interactive method has been designed and incorporated within an interface to simplify the decision making process. Notice that the main advantage of the methodologies proposed in this work is its defined structure, what enables its applicability to any other multiobjective combinatorial optimization problem.

To conclude, Chapter 4 details the implementation and design of a Graphical User Interface (GUI) to solve the *MultiObjective Waste Collection Problem* in the southern spanish region of Málaga. This GUI uses the methodology designed in this work,including the optimization of the problems and *R - NAUTILUS* and adjusting the corresponding parameters of the algorithms to the set of criteria proposed.

CHAPTER 4

APPLICATION TO SOLVE A REAL PROBLEM

The present Chapter contains the description of the real Waste Collection Problem. It takes place in a southern region of Spain, and *Diputación de Málaga* manages the service. Here, the reader will find a description of this problem, followed by the results obtained when applying the methodology developed in Section 3. This methodology has been implemented within a Graphical User Inserface (GUI) in order to facilitate the information exchange with the Decision Maker. The WCP of Málaga has been subdivided into subproblems according to the closest depot and type of truck used. The GUI, which is described in this Section, permits the selection of what is the problem to be analyzed and, then, upload the approximated Pareto Set previously generated to proceed with the interactive phase that will guide the DM to select the most preferred solution.

Section 4.3 contains figures that represent the process and results for each of the given subproblems.

4.1 DESCRIPTION OF THE PROBLEM

Located at the southern region of Andalusia (Spain), the province of Málaga hosts a growing population of more than 1.6 million people, distributed through an extension of 7,276 km^2. In the last years, the population in Málaga has raised in almost a 25%. Besides, a large number of tourists visit this city every year which, added to the current number of inhabitants, generate an amount of waste difficult to handle. Hence, municipal and rural administrations, awared of the situation and its possible evolution, showed their interest on the study of their current waste collection system, which is analyzed in this document.

From the very beginning, the company in charge of managing the waste collection within the region of Málaga, provided the real data related to their problem, including containers and landfill location in different coordinate systems, the number of vehicles available at each depot, as well as their current routing cost and the total amount of waste collected per month by each route. An estimation of the total amount of waste collected per month enables the analyst to determine the average quantity of waste accumulated at each point. This value is translated as the *demand* and, since all these street bins must be serviced, our *Waste Collection Problem* can be modelled as a *Capacitated Vehicle Routing Problem (CVRP)* .

In summary, a fleet of 51 vehicles are distributed into four depots, according to the subdivision in areas of service: Antequera, Axarquía, Guadalhorce and Ronda. For an overview, a simple map in Figure 4.1 shows how the province is divided into different areas.

Figure 4.1: Subdivision of Málaga.

The assignment of vehicles to each of these regions is given a priori, based on the number and type of containers to be visited. Some particularities of the roads force the use of a specific type of trucks (Figure 4.2) which can be *rear loading* trucks or *side loading* truck, each of which corresponds to a bin size and has its own limited capacity.

(a) Side loading truck (b) Rear loading truck

Figure 4.2: Types of trucks

In particular, rear trucks have a limited capacity of 12,200 kg and it will be 16,800 kg for side loading trucks. Therefore, vehicle's distribution, as shown in Table 4.1 has been established, by *Diputación de Málaga*, in order to satisfy the demand within the area under consideration.

AREA	Rear trucks	containers (RT)	Side trucks	containers (ST)
Axarquia	5	1456	2	251
Ronda	3	616	0	0
Antequera	7	991	3	1130
Guadalhorce	4	646	3	361

Table 4.1: Vehicle distribution within the region of Málaga.

A large number of containers of two different dimensions are strategically allocated in the region to store the total waste generated by the current population, which exceeds 500,000 *kg* per day.

The number of containers placed at each municipality or town is proportional to the number of inhabitants within an specific ratio in the corresponding area. Figure 4.3 shows the location of all the street bins or containers, to be serviced by *Diputación de Málaga*. At first sight, one may observe large empty areas, most of them concentrated on the coast. This is due to the fact that some municipalities hire private companies for the waste management, so these services are excluded from this study.

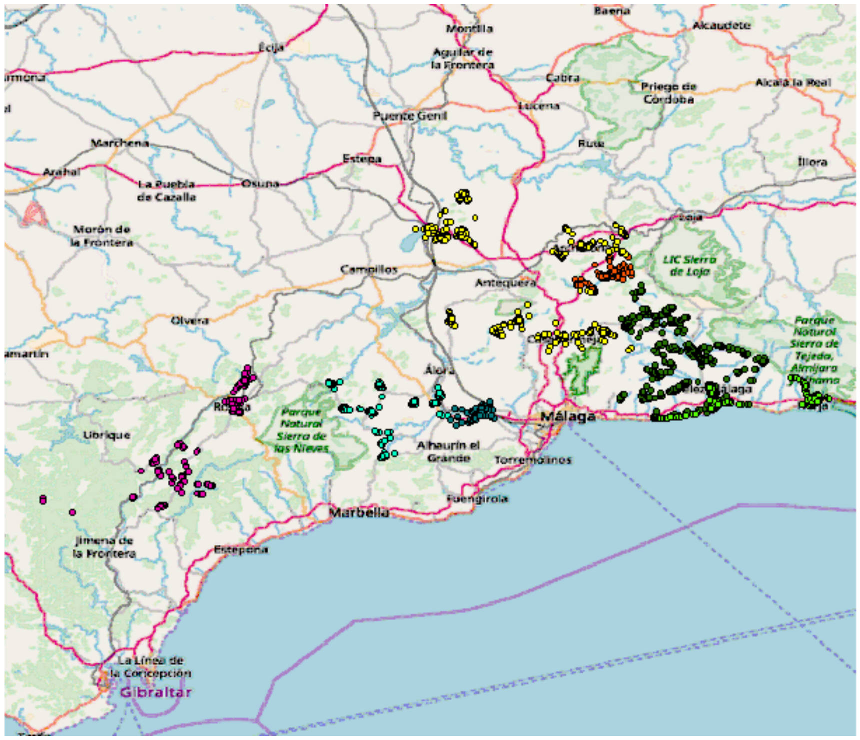


Figure 4.3: Containers distribution.

Currently, *Diputación de Málaga* provides the waste collection service, regarding a total of 4,130 street bins; then transfer them to the corresponding landfill and the final treatment of solid waste generated. Note that the location of these bins also contemplates the characteristics of the road that should be driven from the depot to the collecting point, so that the vehicles available can perform their route.

In real *Waste Collection Problems (WCP)*, containers are located according to the trucks and the population concentrated within an area. Since rear containers or bins have a smaller capacity, they are usually placed at the nucleus of the municipality, in order to allow the corresponding vehicle to perform the service and satisfy the population requests. This fact arises when the dimensions required by a side truck to traverse an specific street is larger than the road's width can handle.

Note that a small number of side containers is appreciated, and usually placed at road crossings or at the suburbs of urban areas, where side trucks have complete access to manage the waste loading process. In addition to this, the capacity that a side truck can host is larger, as well as the corresponding bin's size. That is why the required number of this type of vehicles and containers is lower. Based on this, *Diputación de Málaga* has considered to use, for instance, 4 rear loading vehicles to service 646 containers in Guadalhorce; whereas, Axarquía utilizes only 2 side trucks to satisfy the demand of 1,456 street bins.

As previously stated, this problem can be interpreted as a *Capacitated Vehicle Routing Problem (CVRP)*, so a solution is defined as a set of routes followed by a fleet of trucks with maximum capacity. Besides, an additional time constraint is added, in order to satisfy worker's shifts, limited to seven and a half hours. In order to provide the opportunity to explore a wide set of alternatives, to analyze in depth the *WCP* in Málaga, in addition to the usual objective of cost minimization, some other improvements are incorporated to the current service, as well as an estimatation of the cost associated to the possibility of running a daily service.

To attain these requirements, we formulate the problem incorporating multiple objectives. Then, in terms of the modelization, the following objectives are formulated:

- f_1 To minimize total distance. This value is given by the sum of the overall route distances.
- f_2 To balance the route system. Different approaches, as explained in Section 3.2.1, have been studied. Here, the optimization of route balancing will be determined by the minimization of the longest route.
- f_3 To minimize the difference between the duration of the longest and shortest routes in terms of time.
- f_4 To minimize the number of routes.

Some of the common issues to deal with in *Waste Management Problems (WMPs)* are already solved, as local administrations have already set the location of the bins, the assignation of vehicles to depot and the subdivision in areas to deal with the multi - depot planning. Also, the distance and time matrix have been obtained with a GIS called *NEVA* (Pacheco, 2015). The performance of the current system can be analyzed from Table 4.2, where f_1 denotes the total distance, f_2 the length of the longest route in the system and f_4 the number of routes running. Unfortunately, there are not reliable reference data to compare f_3.

Real Problem	total distance (km)	longest route (km)	numRoutes
Guadalhorce_ST	150.252	108.009	2
Guadalhorce_RT	326.402	122.025	3
Ronda_ST	406.086	210.427	2
Antequera_ST	346.853	242.360	3
Antequera_RT	847.304	237.543	5
Axarquia_ST	322.131	322.131	1
Axarquia_RT	1008.001	424.000	3

Table 4.2: Current performance.

These data are not comparable with the result of our proposal, since *Diputación de Málaga* considers a periodical service and thus, they do not collect every container in the same day. Then, two different concepts of ”route” are used here: from the database, we understand that ”route” corresponds to the tour performed by a vehicle on a day. However, they use this term to define a group of tours that cover the service of some specific municipalities, but not necessarily visiting all the containers everytime. So, it happens that a group of bins are daily collected while others are being visited periodicaly. The distribution of these routes can be observed in Table 4.3 and a screenshot of one example of the data provided is displayed in Figure 4.4.

Route	n	length
Guadalhorce R-1 Est. Cartama (side)	200	108.009
Guadalhorce R-2 Cartama P. (Sun-Tue-Thu)	98	107.291
Guadalhorce R-2 Cartama P. Trasera (Mon-Wed-Fri-Sat)	65	55.751
Guadalhorce R-2 Cartama Pueblo (side)	76	42.243
Guadalhorce Ruta Nº 4 Casarab-Monda-Guaro 1 (Sun-Mon)	162	82.044
Guadalhorce Ruta Nº 4 Casarab-Monda-Guaro 2 - (Sun-Mon)	132	80.366
Guadalhorce Ruta Nº4 Casarab-Monda-Guaro 1 - (Tu to Fri)	222	97.086
Guadalhorce Rº 3 Yunq-Bur-Aloz (Mon to Sat)	112	55.397
Guadalhorce Rº 3 Yunq-Bur-Aloz (Tues to Sat)	123	63.215
Guadalhorce Rº 3 Yunq-Bur-Aloz-(Mon)	103	66.628

Figure 4.4: Guadalhorce routing distribution.

Here n indicates the number of containers serviced by the given route and the right colum ”length” denotes its distance, in km. From this Figure, one may appreciate how they enumerate up to 4 routes but subdivided into different tours every day. In this work, f_4 is considered as an additional objective, but it is introduced in the model as a parameter, so that multiple multiobjective problems, considering f_1, f_2 and f_3, are solved for a range of values of f_4. Then, f_4 is considered as a parameter that provides new opportunities. Besides, regarding the information on route balance, it may seem that f_3 and f_4 are equivalent objectives. However, f_4 contemplates the stopping time required to load the bins along the route, so that the difference between the longest and shortest routes, in terms of time, will implicitly balance the number of stops for every route as well.

Region	loading type	number of tours	number of routes defined	number of vehicles	number of containers
Antequera	rear	15	5	6	991
Antequera	side	11	3	4	1130
Axarquía	rear	11	3	5	1456
Axarquía	side	2	1	2	251
Guadalhorce	rear	8	3	4	646
Guadalhorce	side	2	2	3	361
Ronda	rear	5	2	3	616

Table 4.3: The use of available vehicles in *Diputación de Málaga.*

Then, since the study of a daily service is being considered in this work, we will refer indistinctly to the number of routes and tours. Regarding the objective denoted f_4, as a starting point for the construction of solutions, we consider less tours than what is currently used, so that one can analyze if there exists any feasible solution with a reduced or increasing number of tours used. Later, when the final solution is found, some adjustments can be performed with these tours in order to take advantage of the differences between the employees' shifts and the tours' duration. A smart combination of these tours among the number of vehicles available, subject to time windows determined by shift's duration, could bring multiple benefits to the company. This fact is an immediate consequence of the analysis carried out in this work.

The following section contrasts the results obtained for the *MOWCP* in Málaga. To simplify this problem, in addition to the subdivision into areas, we have split the problem according to the type of truck that collects the bins, i.e. side loading and rear loading trucks. This was not a difficult task, since the dataset provided by *Diputación de Málaga* tags each container according to its type. Then, a total of seven problems derived from the original, each of which is detailed in the next lines.

4.2 Results and discussion

The *MultiObjective Waste Collection Problem (MOWCP)* of *Diputación de Málaga* can be reduced to solve seven different, and with a smaller dimension, *MOWCPs*. As previously mentioned, these problems have been determined in terms of the type of truck and the area where the street bins are allocated, which are described as follows:

Ronda

Ronda is an area located at the north - west part of the region of Málaga, in the middle of a mountain range. More than 51,816 people live in this area that extends over 1,253 km^2. It is important to highlight the dominance of the city of *Ronda*, which hosts more than the half of this population. However, other municipalities such as *Cortes de la frontera*, *Benaoján* or *Montejaque*, also belong to this area.

Its complex geography does not allow to design the most efficient route system. In addition to this, the municipalities of this area are characterized by the narrowness of their streets and other difficulties arise in some areas due to the pavement conditions. Therefore, these issues make it not suitable, for the moment, to use side containers in this area. Hence, this part of the region counts with rear containers only, so a single *WCP* is solved in Ronda.

The number of containers assigned to each town or municipality is shown in Table 4.4 and its distribution can be seen in Figure 4.5.

RONDA	num. Containers	number of persons	Total Kg/Year
ALGATOCIN	24	866	279,860
ALPANDEIRE		271	147,455
ARRIATE	145	4,075	2,155,276
ATAJATE	5	142	45,889
BENALADID	7	258	83,376
BENALAURIA	20	492	158,996
BENAOJAN		1,531	833,041
BENARRABA	16	544	287,723
CARTAJIMA		250	136,029
CORTES DE LA FRONTERA	120	3,461	1,118,469
FARAJAN		246	133,852
GAUCIN	36	1,647	871,102
GENALGUACIL	12	499	161,259
IGUALEJA		823	447,807
JIMERA DE LIBAR	12	461	148,978
JUBRIQUE	31	712	230,092
JUZCAR		239	130,044
MONTEJAQUE		1,010	549,557
PARAUTA		241	131,132
PUJERRA		314	170,852
RONDA	182	36,665	19,949,992
TOTAL	610	54,747	28,170,781

Table 4.4: Distribution of containers by population in Ronda.

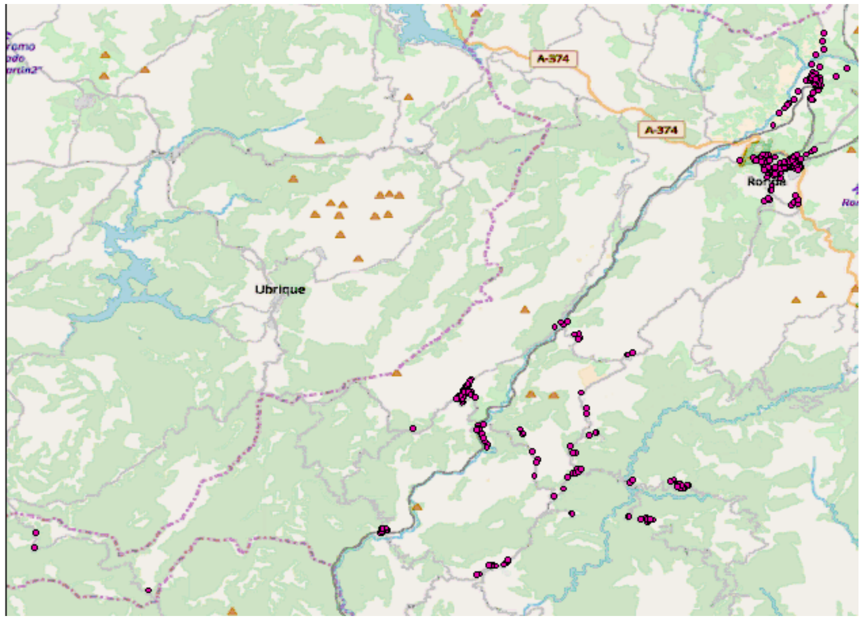


Figure 4.5: Distribution of rear bins in Ronda

Note that nowadays this area deals with a total of 616 bins to service using 3 routes. We try to improve this service by providing multiple solutions obtained when solving the *MultiObjective Waste Collection Problem* using a discrete variation in the number of routes between 3 and 7, which leaves the Decion Maker (DM) a wide set of alternatives to select his / her most preferred route system.

Antequera

Located at the north of the province of Málaga, the subregion of **Antequera** also shares boundaries with the Andalusian province of Córdoba. It constitutes an important needle of the transportation network in Andalucía because of its extension along the plain of the Guadalhorce river. Mainly dedicated to the agriculture, this region hosts a population of over 126,000 inhabitants, spread in 24 different municipalities.

Figures 4.6 and 4.7 shows the location of rear and side street bins in the area of Antequera.

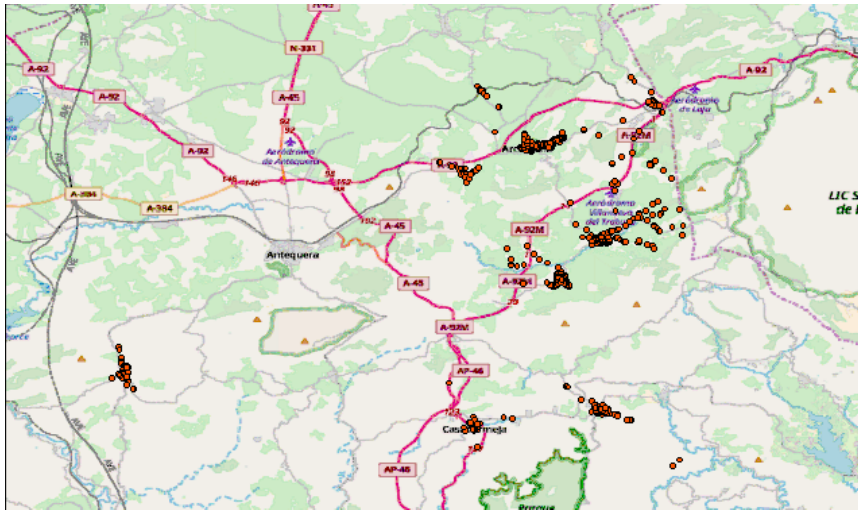


Figure 4.6: Distribution of rear bins in Antequera

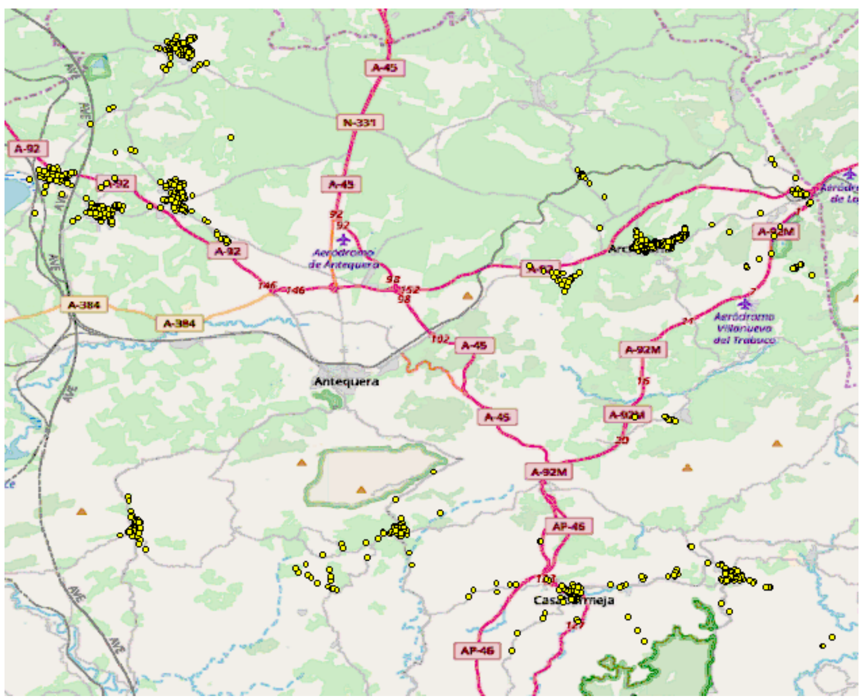


Figure 4.7: Distribution of side bins in Antequera

In this area, 3 vehicles are used to service 1,130 side containers placed on the roads and other accesible points; whereas 7 are the rear loading vehicles in charge of collecting the waste generated at 991 different points within this area. However, the number of routes vary, since the current system covers the service with a periodical routing system that counts up to 11 and 15 for the side and rear collection problems, respectively, as previously mentioned.

Table 4.5 summarizes the number of containers placed at each of its towns.

Municipalities	num. Containers	number of persons	Total Kg/Year
ALAMEDA	100	5,455	2,168,289
ALMARGEN		2,045	864,300
ANTEQUERA	13	41,620	18,157,347
ARCHIDONA	316	8,705	3,464,903
ARDALES		2,588	1,093,793
CAÑETE LA REAL		1,812	765,824
CAMPILLOS		8,677	3,667,251
CARRATRACA		816	344,875
CASABERMEJA	115	3,651	2,132,136
COLMENAR	148	3,583	915,671
CUEVAS BAJAS		1,494	594,666
CUEVAS DE SAN MARCOS		4,029	1,457,050
CUEVAS DEL BECERRO		1,704	720,179
FUENTE PIEDRA	72	2,733	1,086,331
HUMILLADERO	70	3,430	1,439,469
MOLLINA	108	5,185	2,175,991
SIERRA DE YEGUAS		3,488	1,474,170
TEBA		4,044	1,709,158
VALLE DE ABDALAJIS	112	2,712	997,340
VILLANUEVA DE ALGAIDAS		4,471	1,561,940
VVA. DE LA CONCEPCION	76	3,460	2,020,594
VVA. DE TAPIA		1,603	638,052
VVA. DEL ROSARIO	132	3,588	1,428,153
VVA. DEL TRABUCO	231	5,444	2,166,907
TOTAL	1,493	126,337	53,044,388

Table 4.5: Distribution of containers by population in Antequera.

Axarquía

The eastern zone of the province of Málaga corresponds to the area of Axarquía, which is extended through the inner and coastal border limit with Granada. More than 205 thousand people live in the different municipalities of this area. The number of inhabitants at each town, and the containers placed there, can be checked at Table 4.6.

AXARQUIA	num. Containers	number of persons	Total Kg/Year
ALCAUCIN		2,832	1,187,820
ALFARNATE	38	1,240	370,442
ALGARROBO		6,601	3,301,080
ALMACHAR	76	1,915	614,710
ARCHEZ	26	487	154,800
ARENAS	38	1,397	444,058
ARFARNATEJO	29	515	153,853
ARROYO DE BENAMARGOSA	57	1,613	481,873
BENAMARGOSA	57	1,613	517,769
BENAMOCARRA	102	3,084	989,956
BORGE (EL)	52	984	315,862
CANILLAS DE ACEITUNO	92	1,851	588,369
CANILLAS DE ALBAIDA	30	979	311,190
COMARES		1,583	485,720
COMPETA	125	3,885	791,305
CUTAR	35	661	78,613
FRIGILIANA	89	3,395	1,718,345
IZNATE	42	943	302,701
LOS ROMANES		414	123,680
MACHARAVIAYA	34	500	160,499
MOCLINEJO	51	1,283	411,840
NERJA	1	22,918	13,440,410
PERIANA	127	3,542	421,253
RINCON DE LA VICTORIA		41,827	18,258,626
RIOGORDO	99	3,083	921,025
SALARES	11	229	72,791
SAYALONGA	40	1,568	498,413
SEDELLA	23	715	227,274
TORROX		18,514	8,869,920
TOTALAN	23	736	236,254
VELEZ MALAGA	4	76,911	37,177,695
VIÑUELA	138	2,073	658,935
TOTAL	1,439	209,891	94,287,079

Table 4.6: Distribution of containers by population in Axarquia.

Despite the fact that this might be one of the largest area, most of its municipalities manage their own waste collection system, so only a few towns appear in the previous table. Even those municipalities with a large population, show a small number of containers as can be the case of Rincón de la Victoria, in the former case. This might be occasioned by some arragements between the town hall and *Diputación de Málaga* to collect a larger amount at a reduced number of points where they have placed side containers, as it occurs in Vélez - Málaga.

The highest point in the province is located in this region, which is part of a mountainous zone where one can also find streams and reservoirs. These geographical characteristics determine the location of rear and side street bins, depending on the properties of the road. In this case, the depot is located in *Vélez Málaga*, one of the largest town in the area, in terms of extension. Then, in order to satisfy the requirements of such an amount of people, two different systems run the collection service, one for the side bins and the other one for the rear containers.

From Figure 4.8 one may observe how side containers are located at the edges of the roads and, usually, on main roads in order to facilitate the loading of the waste on the side truck.

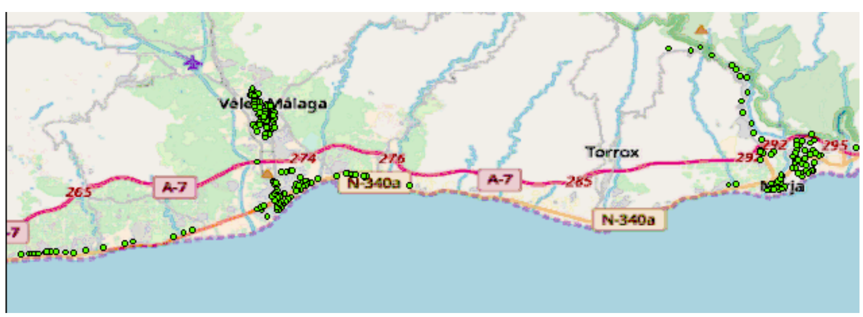


Figure 4.8: Distribution of side bins in Axarquia

The rest of containers placed within the area (Figure 4.9) are collected using rear trucks.

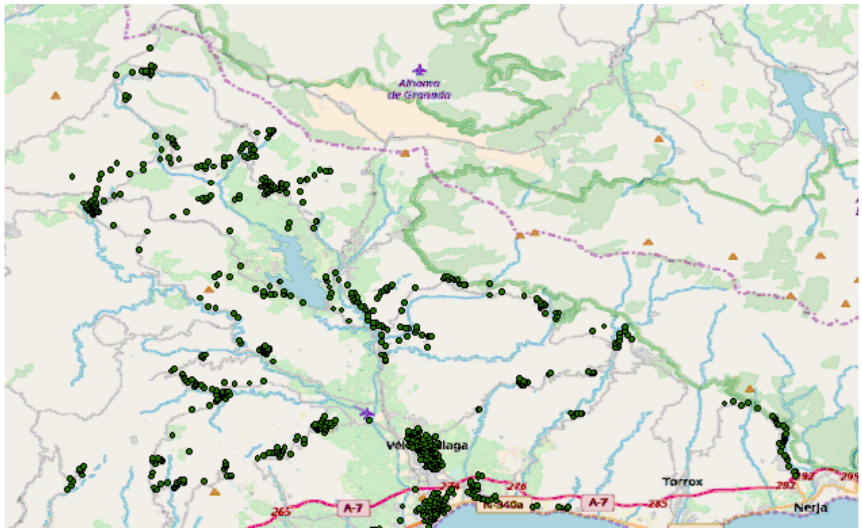


Figure 4.9: Distribution of rear bins in Axarquia

In this occasion, more than 1,400 bins are placed in Axarquía. Then, 5 rear trucks are available to run a total of 11 tours divided into 3 routes. The service performed for the side loading containers is split into 2 tours run by 2 vehicles. However, only one large route is designed to collect these street bins.

GUADALHORCE

The Guadalhorce region is located close to the city centre of Málaga. It is named after the river that runs over the large valley where it lays out and, taking advantage of its position, it features a good road network. In particular, it defines the link between the inner province of Málaga and the coast.

Up to 38,794 inhabitants make *Alhaurín de la Torre* the largest town in this region. Smaller municipalities, such as *Alhaurín el Grande*, *Almogía*, *Álora*, *Cártama*, *Coín*, *Pizarra and Valle de Abdalajís*, are also located in this valley.
An outline of the distribution of containers, in terms of the population of each municipality, can be found in Table 4.7. It shows the distribution of containers to be collected and transported to the depot by *Diputación de Málaga* in Guadalhorce. In this case, the depot is located at Cártama, which is a strategic point in the area

because of its road connectivity.

GUADALHORCE	num. Containers	number of persons	Total Kg/Year
ALH. DE LA TORRE		38,067	13,576,387
ALH. EL GRANDE		24,249	8,648,273
ALORA		13,342	4,758,351
ALOZAINA	65	2,206	794,276
BURGO, EL	75	1,947	701,022
CARTAMA	351	24,242	8,645,777
CASARABONELA	70	2,690	1,072,395
COIN		22,536	8,037,341
GUARO	75	2,284	910,539
MONDA	79	2,383	950,006
PIZARRA	65	9,298	3,316,081
TOLOX		2,295	899,220
YUNQUERA	116	3,091	1,112,922
TOTAL	896	148,630	53,422,590

Table 4.7: Distribution of containers by population in Guadalhorce.

In this occasion, the waste management in this sector of the province also counts with two different types of trucks, each of which requires a specific sort of container. Figure 4.10 shows the distribution of rear containers along the municipalities of the valley.

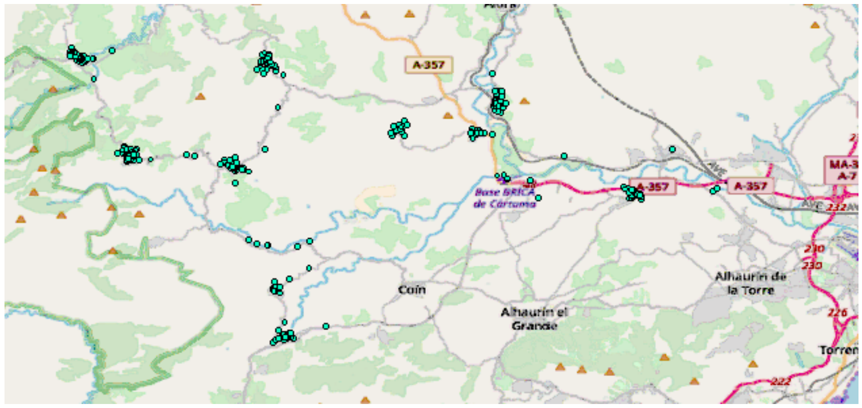


Figure 4.10: Distribution of rear bins in Guadalhorce

Currently, the service is run with a fleet of 4 rear trucks that perform a total of 8 different routes to visit all the 646 containers periodicaly.

Furthermore, the location of the set of side bins to be collected by the administrations is displayed in Figure 4.11. Here, one may observe that most of these containers are distributed along the roads, instead of being concentrated in urban areas. However, the conditions of their location in the municipality of Cártama, make it feasible to set side loading street bins. The main advantage derived from this distribution is the reduction of resources needed to collect the street bins, since a smaller number of containers imply to cut off the number of routes, and so the number of vehicles running the service, what permits a cost reduction to the administration.

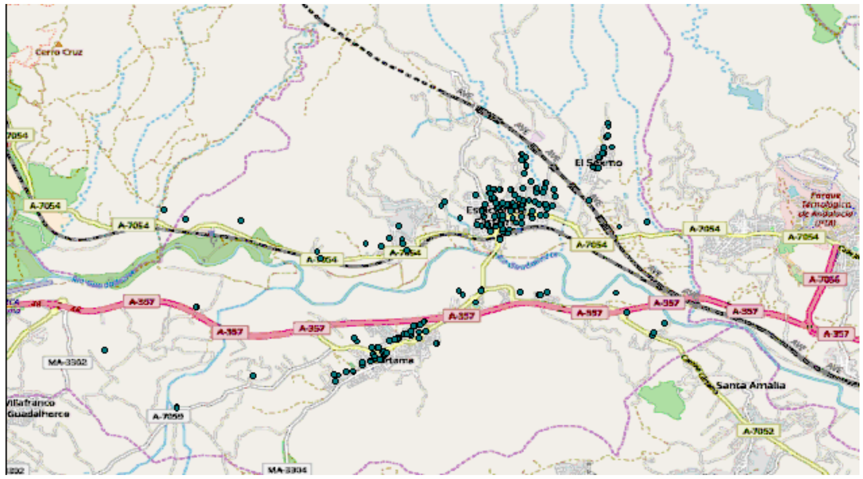


Figure 4.11: Distribution of side bins in Guadalhorce

To satisfy the population requests, 2 different routes are performed to complete the service that counts with around 360 containers.

Hence, the main problem has been subdivided into the seven scenarios detailed above, which could be listed as:

1. Ronda rear loading problem.
2. Antequera rear loading problem.
3. Antequera side loading problem.
4. Axarquía rear loading problem.
5. Axarquía side loading problem.
6. Guadalhorce rear loading problem.
7. Guadalhorce side loading problem.

Each of these problems has been tackled as a *CVRP*, including a time constraint in the duration of each tour, which is limited by the shift length to 7h30min. Then, to solve these problems, Wierzbicki's Achievement Scalarizing Function is minimized multiple times using *GRASP* and improved with *Path*

Relinking. This algorithm has been proved to be the one that provides the best approach, within the set of algorithms developed in this work. To generate an approximation of the Pareto front, as explained in Section 3.2, including the optimization of the number of routes, the method has been run several times, each of them incrementing the number of routes used.

Note that, when applying *Path Relinking* to this particular problem, some factors must be taken into consideration. On the one hand, if f_1 or f_3 is the guiding objective function, *Path Relinking* studies the possible connections between two solutions from the approximated Pareto front, z_i and z_{i+1}. Each of them are defined with a tag. Therefore, the guiding solution, S^G, will be the one with the best value of the objective function under consideration and the other one will be the initial solution in the *Path Relinking* process. To reach the solution guiding the search from the initial solution, S^i, it is neccessary to transform each route from S^i into its match in S^G. This is done by avoiding the evaluation of the common elements and evaluating the restricted list of moves given by the symmetric difference set. Finally, at each step, the best - improvement strategy is considered to select the move to be performed.

On the other hand, if f_2 is the guiding objective function, then the initial solution (S^i) and the one that guides the search (S^G) are chosen in terms of the value f_2, so that the solution with better value will be chosen as S^G. Now, there is no need to compute the symmetric difference for each pair of routes, but only between routes from S^i and the longest route from S^G. When two routes are paired, the subsequent steps are equal to the previous case.

Visual representations of the approximations of the Pareto fronts obtained are presented in the following Figures (Figures 4.12,4.13, 4.14, 4.15, 4.16, 4.17, 4.18). Note that the number of routes utilized are represented in different colours in order to state the different options.

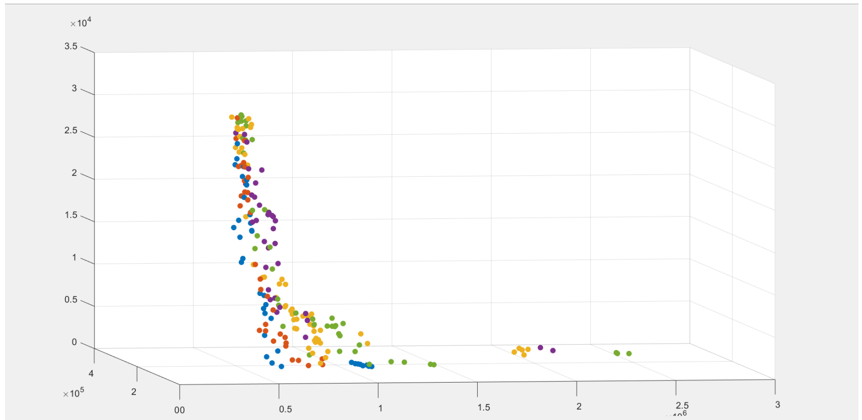


Figure 4.12: Approximated Pareto front Antequera rear loading.

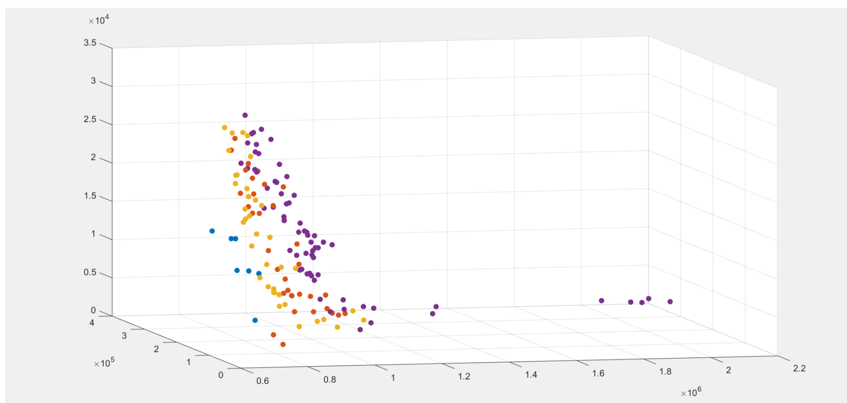


Figure 4.13: Approximated Pareto front Antequera side loading.

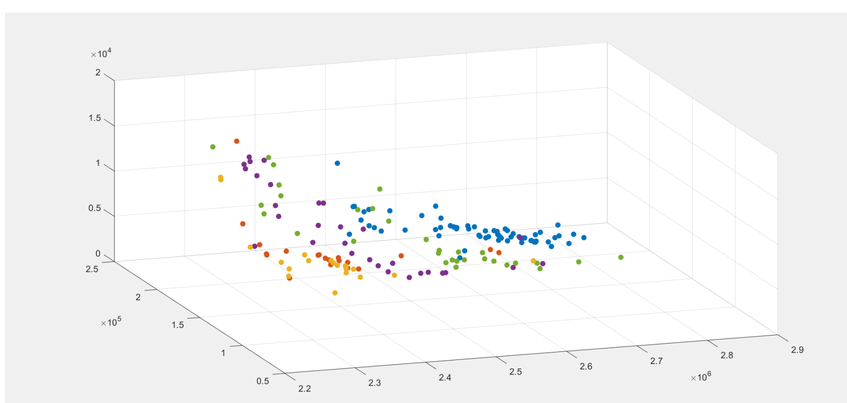


Figure 4.14: Approximated Pareto front Axarquía rear loading.

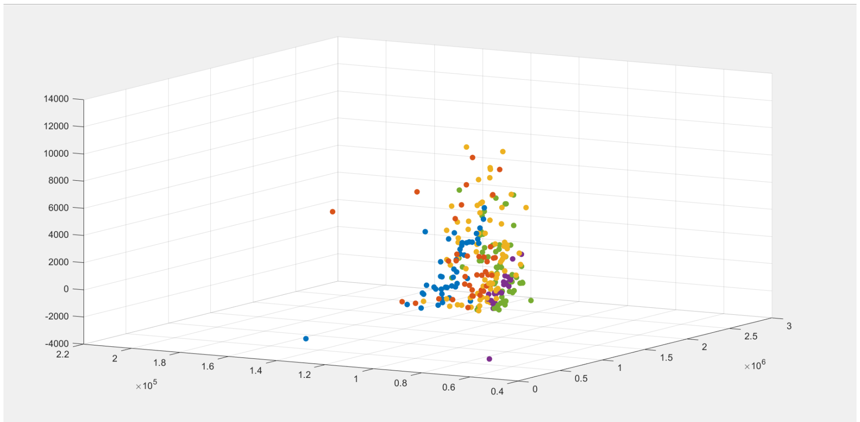


Figure 4.15: Approximated Pareto front Axarquía side loading.

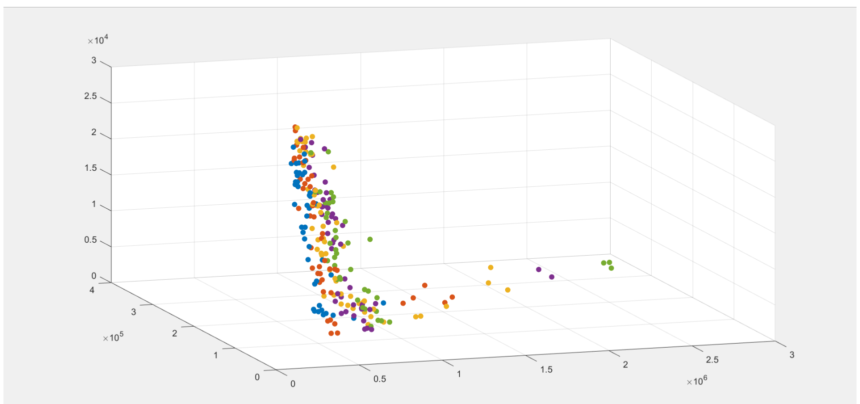


Figure 4.16: Approximated Pareto front Guadalhorce rear loading.

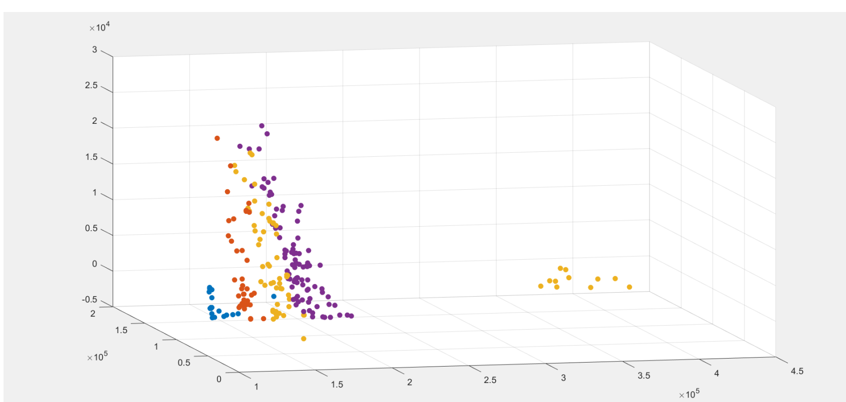


Figure 4.17: Approximated Pareto front Guadalhorce side loading.

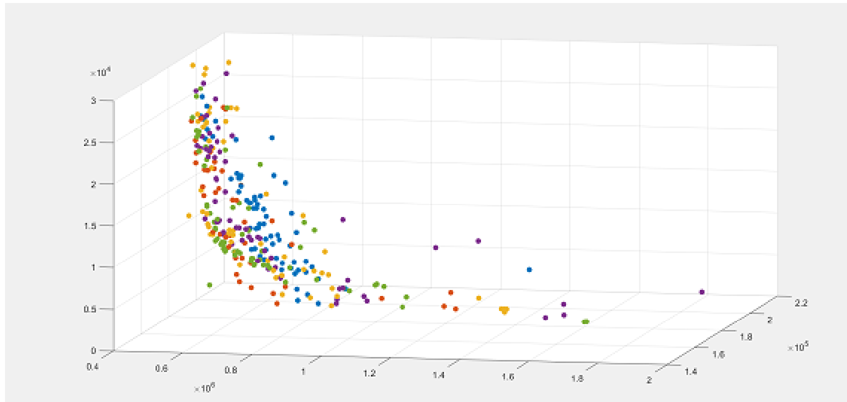


Figure 4.18: Approximated Pareto front Ronda

To interpret these graphics, remember that the XY axis, which defines the bottom surface, is determined by the values corresponding to f_1 and f_2 in meters, whereas the Z axis represents the values of f_3 in seconds. Each of these graphics represents the union of the results obtained for each of these values, where solutions with the same number of routes are plotted using the same color and thus, the value achieved by objective f_4 is defined in terms of colours. For a given problem, up to 5 different number of routes have been studied.

In general, results show a concentration of nondominated solutions in the proximities of the area of what would be defined as the ideal point, given by the algorithm used. In the resolution process, the Pareto front found in Axarquía rear loading problem (see Figure 4.14), shows a reduced number of nondominated solutions. Increasing the number of routes reduces the difference between the duration of the longest and the shortest routes in the collection system, what improves the value of f_3. However, this variation implies an incremental cost into the total distance, in meters, driven by the employees.

Road conditions make it difficult to achieve an appropriate balance of the routing system in Ronda, as it can be appreciated from its Pareto front (Figure 4.18). This might correspond to the driving conditions, which enlarge the time required to move from one point to another in some particular cases. However, for this problem, good values have been achieved for the other objectives, improving the current system in most of the solutions provided.

The Pareto front obtained for Guadalhorce rear loading problem, reveals a large number of nondominated solutions concentrated within a range of values for f_2. There are some isolated nondominated solutions at the tail of the graph. The limitation on the duration and the distance matrix might be the main cause for this spread distribution of efficient solutions.

There are multiple factors that prevent from obtaining a more populated Pareto front when dealing with real problems. In this case, road conditions, bins distributions and the number of routes applied could be the main reasons. However, as detailed in Section 3.2, literature reveals the influence of the definition considered for the route balance. Two of these formulations have been considered in this problem: the minimization of the longest route and the minimization of the difference between the longest and shortest duration of the routes. The former consists of optimizing a min - max problem, which, in multiobjective problems, is not easy to handle. However, the last definition, as deduced in Halvorsen-Weare and Savelsbergh (2016), provides a greater number of nondominated solutions.

A lack of efficient solutions can be observed in some regions of the Pareto front (see Figures 4.17 and 4.15) for the side loading problems in Guadalhorce and Antequera. Here, using a larger set of routes leads to better values on the largest route. However, the empty areas in the Pareto front are due to the real distance between some of the bins to collect, which makes it impossible to interchange bins in the routing sequence to reduce their length.

Nevertheless, to handle this amount of alternatives, a Graphical User Interface (GUI) has been implemented in order to help the Decision Maker (DM) in the decision process. To generate the approximation of each Pareto front, this interface includes the methodology developed in this work, as well as the interactive method (*R-NAUTILUS*) that permits an exploration of these solutions to analyze the different alternatives. An example is detailed in the following section (Section 4.3).

4.3 GUI TO SOLVE THE REAL PROBLEM

Once we have obtained the set of nondominated solutions for each *MultiObjective Waste Collection Problem (MOWCP)* of *Diputación de Málaga*, as detailed in a previous section (Section 4.2), it is convenient to translate this information to the Decision Maker (DM) using a Graphical User Interface (GUI).

This GUI has been designed, specifically for the *MultiObjective Waste Collection Problem (MOWCP)* of *Diputación de Málaga*. The code is implemented using language *Java 8.1*, within *Eclipse Oxygen 1.0*. In order to display the location and routing, the use of different extensions of *ArcGIS 10* are required. In this case, the *MOWCP* of *Rear Loading in Ronda* has been chosen, as an example, to detail the performance of this interface. The decission procedure follows 4 different phases:

Phase 1. The application is run and the main window shows how to proceed. Press the " START " button as shown in Figure 4.19.

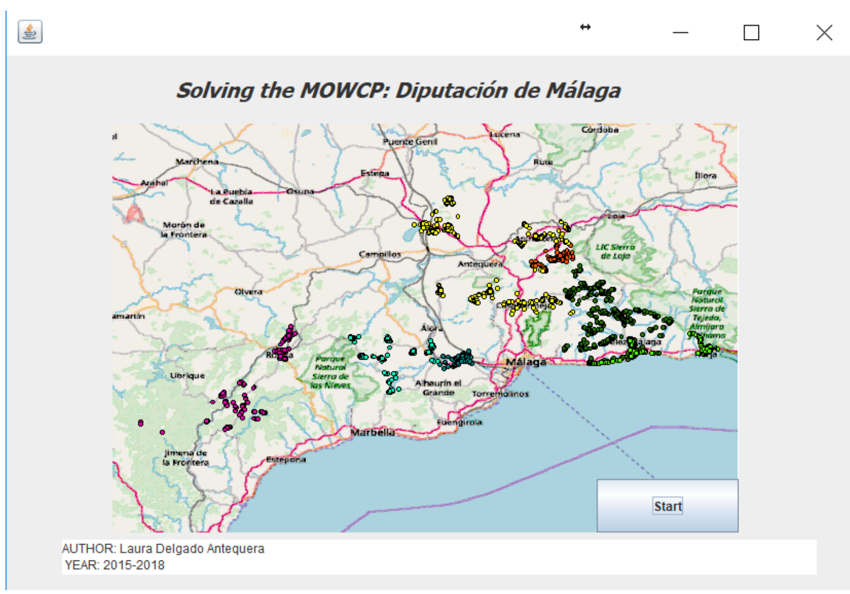


Figure 4.19: Main window of the GUI designed.

Phase 2. Then, as shown in Figure 4.20, select the problem to analyze. Clicking on the name, the DM might observe the distribution of the containers of the selected problem through the region. Then, press ″ Proceed to analysis ″ to start the decision process.

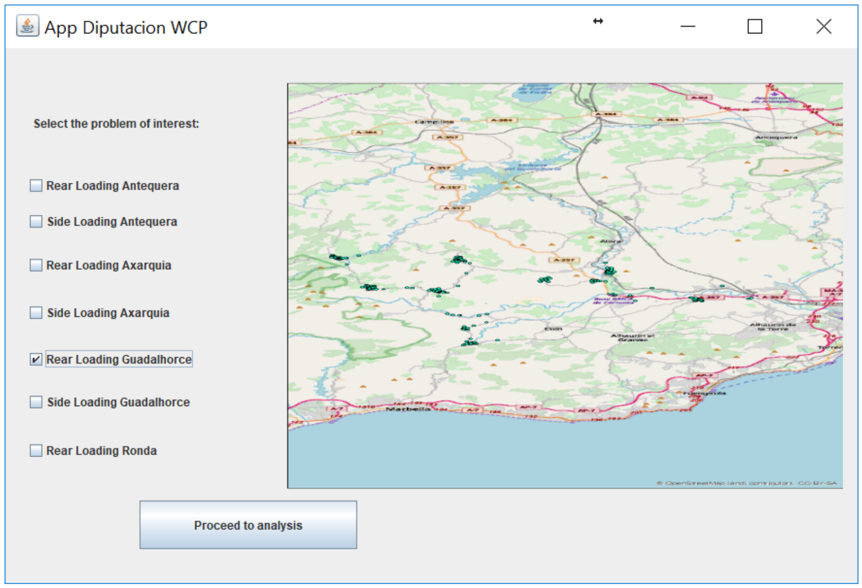


Figure 4.20: GUI is used to select the problem.

Phase 3. At this point, the set of nondominated solutions of the chosen problem is being loaded, so please wait. Next, *R - NAUTILUS* as detailed in Section 3.3 is used for selecting the most preferred solution. Some boxes are provided on the left to define the initial reference point. Also, on the right hand, the evolution of the achievable region can be observed, as well as the variation range of each objective function (See Figure 4.21). Notice that, at any time, the DM might press ″ STOP ″ to re - define the navigation or to go to a previous point, as desired.

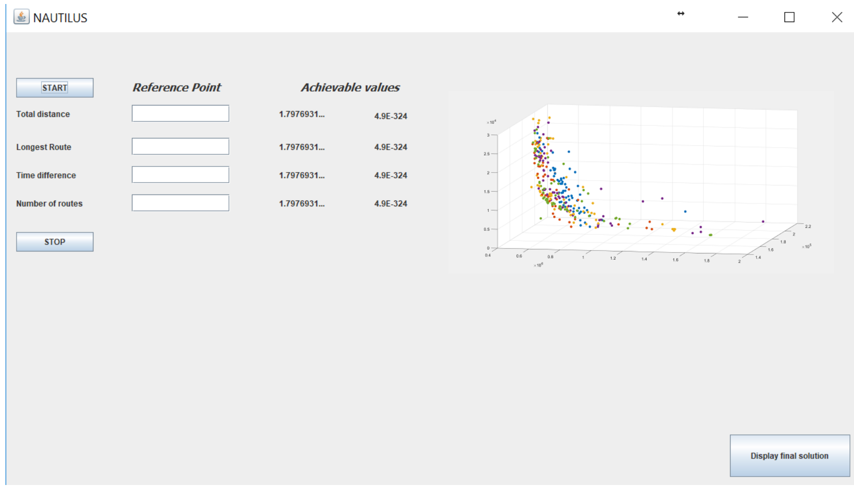


Figure 4.21: *R - NAUTILUS* is implemented within the GUI.

Phase 4. When the most preferred solution is selected, a wider analysis can be obtained if the routing system is drawn. This is dispayed on the last screen, after clicking on ”Display final solution”.

Phase 3 lasts until the DM is satisfied with the solution found, being aware of the limitations on the values that the functions may achieve. Different steps are followed in this part of the process.

Step 1. First, set the reference point. For each objective, a desired value is defined by the DM. These values must be introduced into the white boxes on the left of the screen.

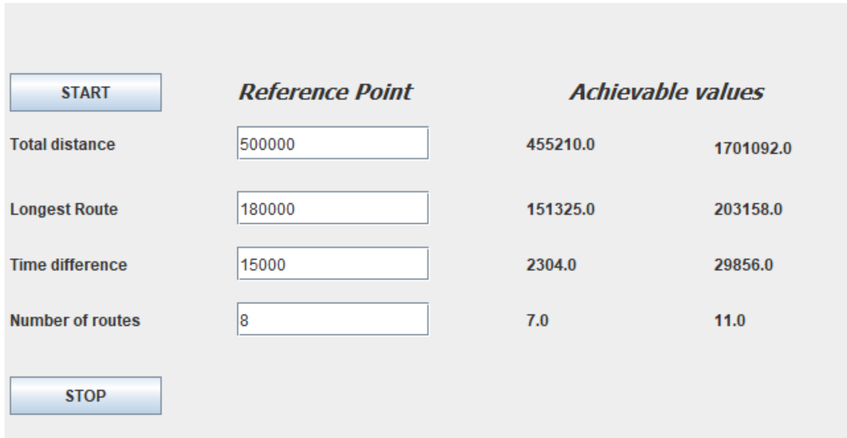


Figure 4.22: Set the reference point.

Step 2. Then press the ”START” button. At this point, the range of values varies dinamically, as well as the achievable set of points, while the procedure advances as one can observe in Figures 4.23, 4.24 and 4.25.

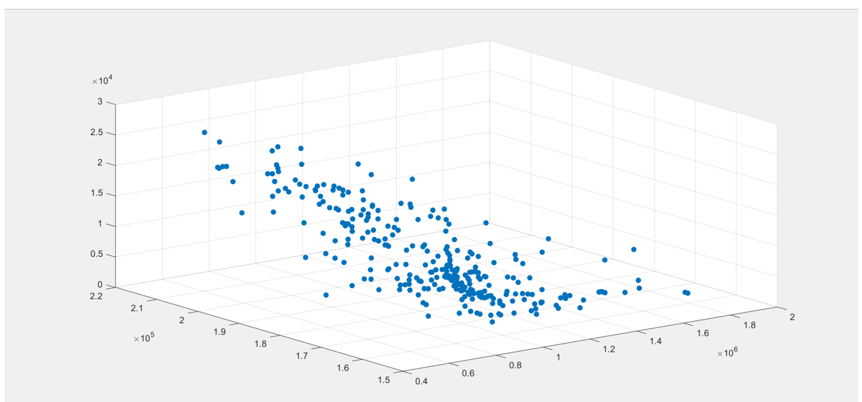


Figure 4.23: First achievable set.

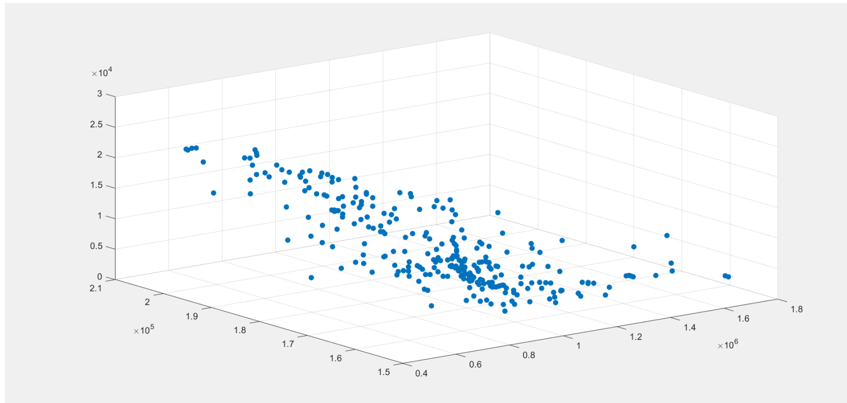


Figure 4.24: First iteration achievable set.

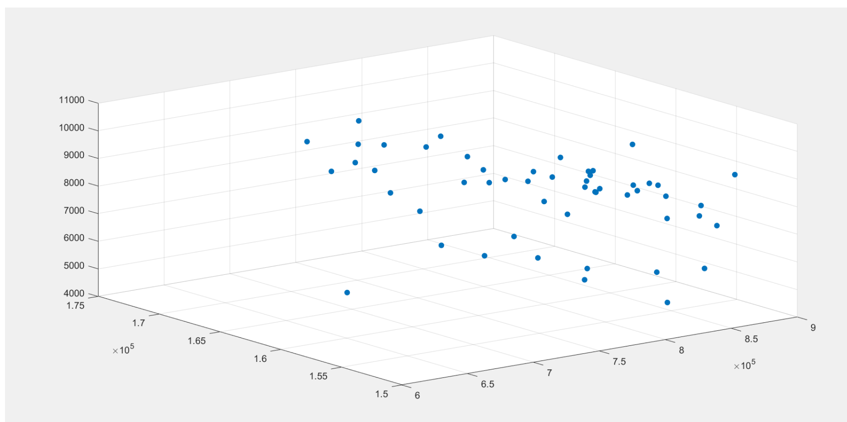


Figure 4.25: n - th iteration achievable set.

.

Step 3. At any time, the DM can press the ”STOP” button and decide.

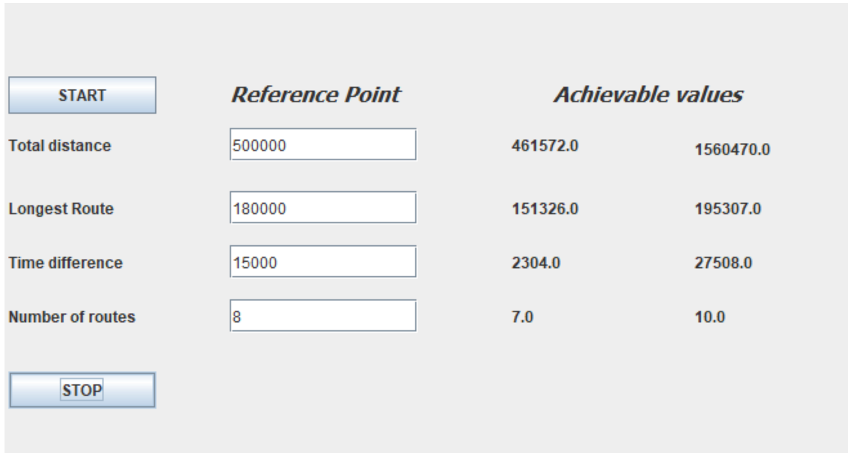


Figure 4.26: Screen shot when ”STOP”.

Different options arise at this moment:

- The DM may ask to see the current most preferable solution. In this case, (s)he should press the button ”DISPLAY FINAL SOLUTION”, which leads the process to Stage 4.
- A new reference point can be defined. This option would guide the process to *Step 1*.
- The DM might be interested in considering a perspective observed from any of the previous iterations. In this case, the DM should click on the area and the procedure would restart from that point.
- Clicking on the ”PROCEED” button, the process continues normally.

Note that, if the DM does not press the ”STOP” button, the process continues until it reaches the most preferred solution, in the direction established.

The last screen shows a description of the solution selected, including some details that cannot be noticed from the simple values, such as the distance driven from the depot to the first container visited, or from the last one to the depot, the average speed of each route, the time invested to service a number of containers or the time required to complete each route. The latter value will allow the user to design different strategies combining different tours that will be serviced by a single vehicle, obeying the time constraint. An example of the solution is shown in Figure 4.27. Here, some of the routes are overlapped, so an individual performance is detailed in Figures,4.28,4.29,4.30 and 4.31.

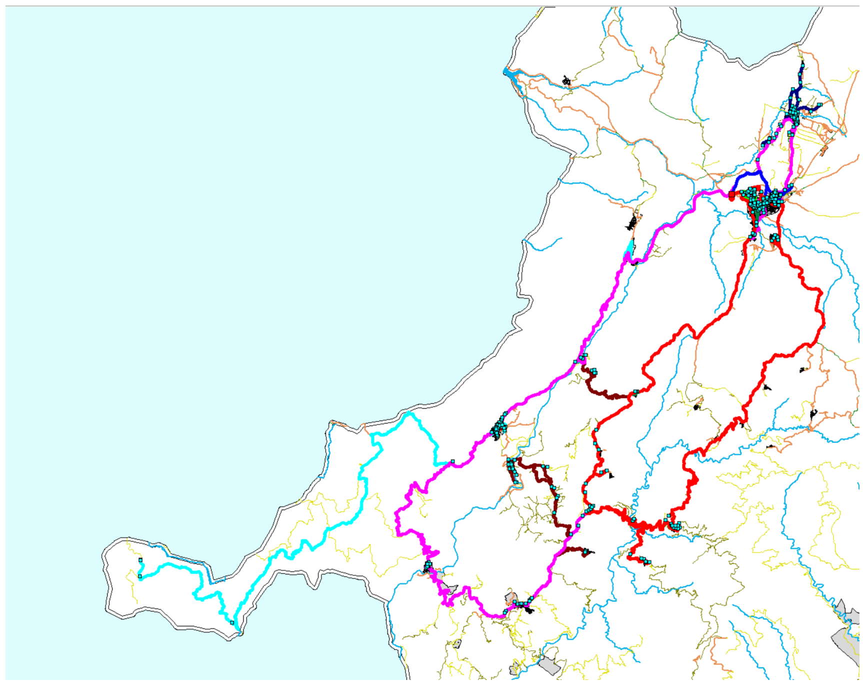
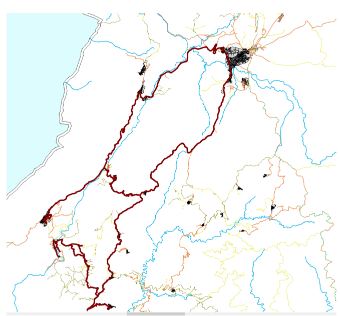
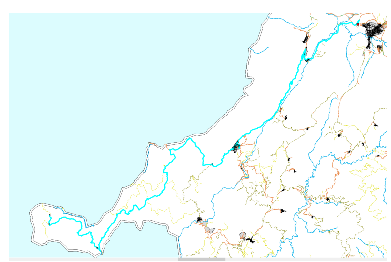


Figure 4.27: Solution performance.

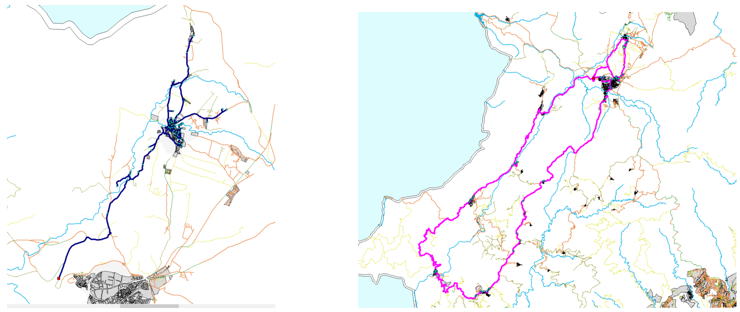


(a) Route 0



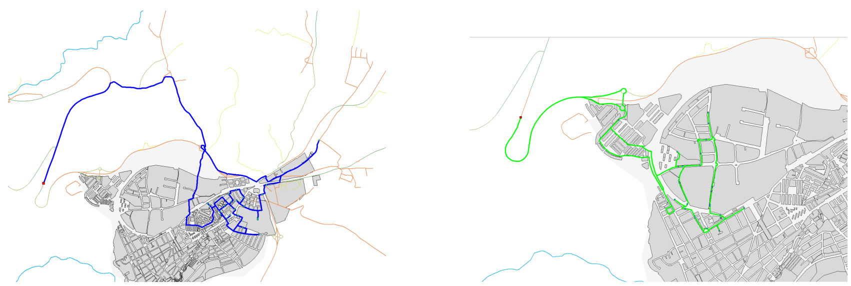
(b) Route 1

Figure 4.28: Routes solution S.



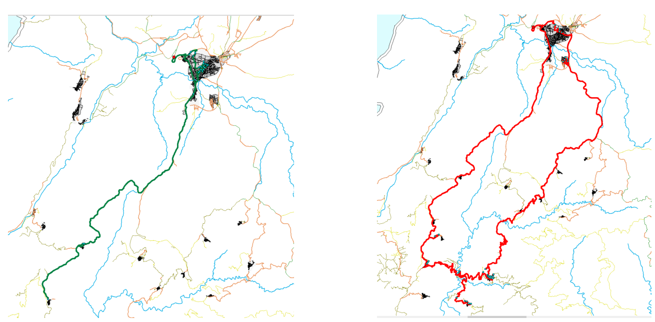
(a) Route 2 (b) Route 3

Figure 4.29: Routes solution S.



(a) Route 4 (b) Route 5

Figure 4.30: Routes solution S.



(a) Route 6 (b) Route 7

Figure 4.31: Routes solution S.

This GUI has been designed for this particular problem, but the idea can be extrapolated to solve any other *MultiObjective Capacitated Vehicle Routing Problem (MOCVRP)*.

CHAPTER 5

CONCLUSIONS AND FUTURE LINES OF RESEARCH

Waste collection problems have been a social challenge for many years. The particular features of each scenario of application make it unfeasible to find the appropriate methodology that encompasses all of them. In addition to this, decision making approaches can be considered. For instance, the interest might lay on how to structure waste management. In this context, some authors have defined Decision Support Systems (DSS) to define the best management policy in terms of a given set of attributes. A different perspective arises when multiple costs are to be optimized, so determining the best solution involves a decision making process and, likely, the design of an efficient engine to find the most preferable solution.

This document introduces a methodology to analyze the real *MultiObjective Waste Collection Problem (MOWCP)* in Málaga. Dealing with such a problem requires to solve different issues that determine the main contribution of this work. Generally speaking, the goal is to provide a useful tool that permits an easy information exchange between the waste managers and the analyst. This procedure implies getting over several aspects, corresponding to the objectives of this research as introduced in Chapter 1:

Obj.1 To analyze the current Waste Collection System in Málaga.

Obj.2 To define a realistic model for the Waste Collection Problem in Málaga.

Obj.3 To study, design and implement an efficient, effective and fast method to solve the MultiObjective Waste Collection Problem.

Obj.4 To define an appropriate interactive method which helps the waste manager on the decision making process.

Obj.5 To design and implement a decision interface to display the strengths and weaknesses of the proposed solutions.

This document began with a brief literature review on Waste Management (Chapter 2), focused on those works that consider routing and a modelization based on nodes instead of arcs. It revealed the lack of an efficient multiobjective method which allows the DM to learn and guide the search of the most preferred solution during the decision making process.

Hence, a methodology capable of generating a good approximation of the Pareto Set is introduced in Chapter 3, reaching *Objective 3*. *Vehicle Routing Problems*, and so optimizing the routing collection system in *WCP*, are hard to solve using exact methods. Therefore, metaheuristic strategies are developed to generate the best possible approximation of the Pareto front. Different approaches are proposed in this work based on *GRASP* improved with *Variable Neighborhood Search* and *Path Relinking* strategies. Previous researches successfully apply the hybridization of GRASP and either Path Relinking or VNS to optimize single - objective problems. Here, an extrapolation to the multicriteria perspective is applied. To obtain a first approach of the Pareto front, *GRASP pure - ordered* and *GRASP pure - random* approaches pursue the optimization of a different objective at each construction in an ordered or random fashion, respectively. This idea was introduced in Martí et al. (2015), improved with *Path Relinking*. Moreover, another alternative is proposed in this study. It formulates Wierzbicki's Achievement Scalarizing Function (Wierzbicki, 1980) which is optimized for different weight combinations, providing nondominated solutions. The optimization process uses, in both cases, *GRASP* metaheuristic. During the construction phase of the *GRASP* proposed, two different greedy functions are utilized to construct the whole system of routes at the same time.

The performance of these approaches have been tested with common instances from the literature. Note that no instances have been found for a biobjective *VRP* which contemplates route balance, so the common set of instances for the single - objective VRP has been used to evaluate the performance of the algorithms proposed. Then, the sets of nondominated solutions obtained have been compared with each other. An analysis of the results details that the approximation of the Pareto fronts obtained using *GRASP pure - ordered* or *GRASP pure - random* approaches are usually dominated by the ones obtained if *Achievement Scalarizing Function (ASF)* is applied, leading to the conclusion that the last method generates the best approximation. Varying the weights allows a wider exploration of the range of options available in the solution space. To the best of our knowledge, this technique had not been used within *VRP* to generate the Pareto front, although some studies on other fields define an ASF after determining the weights, so that only one efficient solution will be found.

An improvement methodology is applied next, in order to improve, if possible, the approximation of the Pareto front ($\wp$) obtained. Now, two alternatives are included: *Path Relinking* and *Variable Neighborhood Search (VNS)*. The former selects a pair of nondominated solutions from $\wp$ and tries to transform one into another, with a better value on one of the objectives, by moving nodes in order to reduce the differences between these solutions. On the other hand, a multiobjective variation of the metaheuristic *VNS* is defined, denoted by VNS_{ref}. In this case, a set of potential efficient solutions is maintained during the previous phase. Later, for each pair of nondominated solutions in $\wp$, a reference point is determined by their best values on each objective and *VNS* is applied to minimize the distance between the potential efficient solution to this point. Results show the effectiveness of both alternatives. On the one hand, due to the good performance of the *ASF* method in the construction phase, the improvement phase does not contribute with a large number of new nondominated solutions, if so *Path Relinking* is able to find them in a shorter fraction of time. On the other hand, *Path Relinking* and VNS_{ref} improve the results obtained for the other constructions. In particular, VNS_{ref} performs better. Based on its own definition, it pursues to fill the gap looking for a new nondominated solution, if possible, between two distant nondominated solutions.

Once the approximation of the Pareto Set has been generated, a decision process is needed to manage the amount of solutions generated. Note that, to save computational cost, this approximation is generated in advance, so that the dimensions of the problem do not obstruct the decision making process. In this context, according to *Objective 4*, *NAUTILUS* has been the selected philosophy to guide the decision making to the most preferred solution. Among other reasons, this method gives the opportunity to explore the area according to the given preferences. However, different features have been incorporated to this method in order to simplify the interaction with the DM. For example, it allows to go backward and modify the reference point at any time, as the decision maker wills. In spite of asking for a initial reference point, this method starts from the worst scenario which can be determined by the nadir point or provided by the DM. Information about the range of achievable values is continuosly visible, which awares the DM of his / her possibilities. These are the main reasons why this interactive procedure is appropriate to face a MOWCP, or any MultiObjective Capacitated Vehicle Routing Problem. It permits a deep exploration of the objective space, and not only based on the value of the objectives, but also incorporating further information about the routes of the chosen solution, such as the number of containers used, the duration of the route or the time, among others.

To tackle a real problem, it is important to determine, first of all, what is the aim of it, in order to define the objectives and set some constraints. In this work, Section 4.1 deals with *Objective 1*, including a descriptive analysis of the real problem in a region in south Spain to provide a wide overview of the achievable possibilities. Then, due to the complexity of formulating a mathematical model of certain characteristics of the problem, it is defined by considering four different objectives for the real *WCP* in Málaga. Economical and labor factors are included into the model when defining the objective functions and the constraints. Then, cost savings and route balance, as well as the number of routes, determine the objectives, subject to truck's capacity and the limitation on the working shifts. This contributes to *Objective 2*, which consists of defining the Waste Collection model in Málaga.

When dealing with real logistic problems, it is interesting to display the different locations or routing that form solutions for *VRP* in general, and *WCP* in particular. This is the objective achieved (corresponding to *Objective 5*) in Section 4, where it is described how this methodology is applied to a real *MultiObjective Waste Collection Problem (MOWCP)* using Geographical Information Systems (GIS). In general, GIS like *ArcGIS* and its extension to *Network Analysis*, are a useful tool to draw the solution obtained in terms of the routes followed, as well as the location of the points to serve, in order to provide a better idea of the performance of the solution.

The difficulties arised when dealing with problems that need to provide a service to over 4,000 bins. This fact has led us into the subdivision of the real problem into 7 smaller problems. The distribution of the containers into subregions and the type of vehicles in charge of providing the service have been decisive elements to define these subproblems.

An analysis of each individual problem reveals the strengths and weaknesses of the current solid waste management. So, in order to provide a wide vision of the alternatives available to run the service, four different objectives are considered:

Route's lengths are included in the dataset provided by *Diputación de Málaga* for every problem, one might consider it as a reference point to compare our solutions. Note that better values of objective f_1 would imply saving costs on the routing system. However, the waste manager must incorporate his / her preferences into the decision process in order to determine which are the best options overall.

A Graphical User Interface (GUI) has been implemented in Java 8.1 programming language, using Eclipse Oxygen and some extensions. The proposed methodology is applied to solve the *MOWCP* of *Diputación de Málaga*. It contains the complete interactive process, from the location of the containers to the decision making process, including the selection of the problem to be analyzed and *R-NAUTILUS*, as well as a final visualization of the selected solution performance. Here, the DM is provided with an interactive method that allows an exploration of the alternatives defined by the set of nondominated solutions previously generated.

Finally, the GUI constitutes a useful tool which translates the numerical results obtained into visual information. Its management helps the DM in the decision making process and at the same time, it allows us to analyze the range of solutions available, which permits the DM to learn about the different options to handle the Waste Collection System. Therefore, the objectives marked in Section 1 have been accomplished.

To sum up, a general scheme can be deduced from this methodology, which enables its application into any other *MOVRP*. As observed, three main stages define this methodology:

1. Generate a good approximation of the Pareto front for the MultiObjective problem.

2. Apply an interactive method to deal with the decision making process.

3. Design a GUI which permits an easy interpretation of the results obtained.

To conclude this document, some proposals on future lines of research are listed in the following paragraphs.

Future lines of research

Future lines of research derived from this study might be focused on different directions.

For instance, the development of new algorithms to improve the approximation of the Pareto front, either efficiently or computationally. Currently, genetic algorithms, evolutionary algorithms and other metaheuristics based on population are gaining popularity within the MultiObjective community. Note that, besides the difficulties derived from dealing with multiple objectives, one must consider the complexity due to the large scale of this kind of problems if applied to real world challenges.

Also, designing an improved Graphical User Interface or implementing this methodology within the options of a Geographic Information System (GIS) would be a good opportunity to share this methodology with the scientific community.

Finally, based on improving the Waste Collection System for the real problem, it would be interesting to incorporate some other challenges, such as:

- Considering the multi - depot problem, where the exchange of trucks among them is allowed, in order to reduce the fleet size and, therefore, the environmental cost.
- Introducing more objectives into the model such as periodicity or some environmental perspectives.
- Contemplating the real trace of the routes, in order to improve the compactness of the collection system.

Bibliography

(2017). *beSMART: a software tool to support the selection of decision software*.

Adenso-Díaz, B., Tuya, J., and Goitia, M. (2005). EDSS for the evaluation of alternatives in waste water collecting systems design. *Environmental Modelling & Software*, 20(5):639 – 649.

Aiex, R., Binato, S., and Resende, M. (2003). Parallel GRASP with Path-Relinking for job shop scheduling. *Parallel computing*, 29(4):393 – 430.

Angelelli, E. and Speranza, M. (2002). The Application of a Vehicle Routing Model to a Waste-Collection Problem: Two Case Studies. *The Journal of the Operational Research Society*, 53(9):944–952.

Arribas, C., Andrea Blazquez, C., and Alejandra Lamas, A. (2010). Urban solid waste collection system using mathematical modelling and tools of geographic information systems. *Waste Management and Research*, 28:355–363.

Arroyo, J. E. C., Sampaio Vieira, P., and Soares Vianna, D. (2008). A GRASP algorithm for the multi-criteria minimum spanning tree problem. *Annals of Operation Research*, 159:125–133.

Baker, B. M. and Carreto, C. A. (2003). A visual interactive approach to vehicle routing. *Computers and Operation Research*, 30:321–337.

Bani, M., Rashid, Z., Hamid, K., Harbawi, M., Alias, A., and Aris, M. J. (2009). The development of decision support system for waste management; a review.

International Journal of Chemical, Molecular, Nuclear, Materials and Metallurgical Engineering, 3(1).

Baptista, S., Oliveira, R. C., and Zúquete, E. (2002). A period vehicle routing case study. *European Journal of Operational Research*, 139(2):220 – 229.

Baran, B. and Schaerer, M. (2003). A Multiobjective Ant Colony System for Vehicle Routing Problem with Time Windows. In *The 21st IASTED International Multi-Conference on Applied Informatics (AI 2003), February 10-13, 2003, Innsbruck, Austria*.

Barbosa, H. J. C. and Barreto, A. M. S. (2001). An Interactive Genetic Algorithm with Co-evolution of Weights for Multiobjective Problems. In *Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation*, GECCO'01, pages 203–210, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Basseur, M., Seynhaeve, F., and Talbi, E.-G. (2005). Path Relinking in Pareto Multi-objective Genetic Algorithms. In Coello Coello, C. A., Hernández Aguirre, A., and Zitzler, E., editors, *Proceedings of the Third International Conference, EMO*, pages 120–134, Berlin, Heidelberg. Springer Berlin Heidelberg.

Bautista, J., Fernández, E., and Pereira, J. (2008). Solving an Urban Waste Collection Problem Using Ants Heuristics. *Computer Operations Research*, 35(9):3020–3033.

Belien, J., Boeck, L., and De Ackere, J. V. (2012). Municipal Solid Waste Collection and Management Problems : A Literature Review. *Transportation Science*, 48(1).

Beltrami, E. and Bodin, L. (1974). Networks and Vehicle Routing for municipal Waste Collection. *Network*, 4(1):65–79.

Benjamin, A. and Beasley, J. (2010). Metaheuristics for the Waste Collection Vehicle Routing Problem with Time Windows, driver rest period and multiple disposal facilities. *Computers and Operations Research*, 37:2270–2280.

Benjamin, A. and Beasley, J. (2013). Metaheuristics with disposal facility positioning for the waste collection VRP with Time Windows. *Optimization Letters*, 7(7):1433–1449.

Bing, X., Bloemhof, J. M., Ramos, T. R. P., Barbosa-Povoa, A. P., Wong, C. Y., and van der Vorst, J. G. (2016). Research challenges in municipal solid waste logistics management. *Waste Management*, 48(Supplement C):584 – 592.

Brebbia, C. A., Ferrante, A., Rodriguez, M., and Terra, B. (2000). *The Sustainable City: Urban Regeneration and Sustainability (Advances in Architecture)*. WIT Press / Computational Mechanics.

Caballero, R., González, M., Guerrero, F., Molina, J., and Paralera, C. (2007). Solving a multiobjective location routing problem with a metaheuristic based on Tabu Search. Application to a real case in Andalusia. *European Journal of Operational Research*, 177(3):1751 – 1763.

Caballero, R., Laguna, M., Martí, R., and Molina, J. (2011). Scatter Tabu Search for multiobjective clustering problems. *Journal of the Operational Research Society*, 62:2034–2046.

Caballero, R., Molina, J., and Rodriguez, M. (2003). MOAMP: Programación multiobjetivo mediante un procedimiento de búsqueda tabú. In *Actas del II Congreso Español sobre Metaheurísticas, Algoritmos Evolutivos y Bioinspirados*, pages 153 –159.

Calvete, H., Galé, C., Oliveros, M.-J., and Sánchez-Valverde, B. (2007). A Goal Programming approach to vehicle routing problems with soft Time Windows. *European Journal of Operational Research*, 177(3).

Cencic, O. and Rechberger, H. (2008). Material flow analysis with software STAN. *Journal of Environmental Engineering and Management*, 18:3–7.

Chalkias, C. and Lasaridi, K. (2009). A GIS based model for the optimisation of municipal solid waste collection : the case study of Nikea , Athens , Greece. *WSEAS Transactions on environment and development*, 5(10):640–650.

Chang, N.-B., Lu, Y., and Wei, Y. L. (1997). GIS Technology for Vehicle Routing and Scheduling in Solid Waste Collection Systems. *Journal of Environmental Engineering*, 123.

Chang, N.-B. and Wei, Y. (1999). Strategic planning of recycling drop-off stations and collection network by multiobjective programming. *Environmental Management*, 24(2):247–263.

Chankong, V. and Haimes, Y. (1983). *Multiobjective decision making: theory and methodology*. North-Holland series in system science and engineering. North Holland.

Chifari, R., Piano, S. L., Bukkens, S. G., and Giampietro, M. (2016). A holistic framework for the integrated assessment of urban waste management systems. *Ecological Indicators*, 86.

Christofides, N. and Eilon, S. (1969). An Algorithm for the Vehicle-dispatching Problem. *Journal of the Operational Research Society*, 20(3):309–318.

Christofides, N., Mingozzi, A., and Toth, P. (1979). The Vehicle Routing Problem. In Christofides, N., Mingozzi, A., Toth, P., and Sandi, C., editors, *Combinatorial Optimization*, pages 315–338. Wiley, Chichester, New York, NY.

Christofides, N., Mingozzi, A., and Toth, P. (1981). Exact algorithms for the vehicle routing problem, based on spanning tree and shortest path relaxations. *Mathematical Programming*, 20(1):255–282.

Clarke, G. and Wright, J. (1964). Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, 12(4):568–581.

Constantino, M., Gouveia, L., ao, M. M., and Nunes, A. (2015). The mixed capacitated arc routing problem with non-overlapping routes. *European journal of Operational Research*, 244(2):445–456.

Corberán, A. and Laporte, G. (2015). Routing in waste collection. In *Arc Routing : Problems, methods and applications*, pages 351–371. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA.

Cortinhal, M. J., Mourao, M. C., and Nunes, A. C. (2016). Local search heuristics for sectoring routing in a household waste collection context. *European Journal of Operational Research*, 255:68–79.

Das, I. and Dennis, J. E. (1997). A closer look at drawbacks of minimizing weighted sums of objectives for Pareto set generation in multicriteria optimization problems. *Structural optimization*, 14(1):63–69.

Deb, K., Pratap, A., Agarwal, S., and Meyarivan, T. (2002). A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Transactions On Evolutionary COmputation*, 6(2):182–197.

Delgado-Antequera, L., Pérez, F., Hernández-Díaz, A. G., and López-Sánchez, A. D. (2016). An Interactive Biobjective Method for Solving a Waste Collection Problem. *Mathematical Problems in Engineering*, 2016.

Duarte, A., Pantrigo, J. J., Pardo, E. G., and Mladenovic, N. (2015). Multi-objective Variable Neighborhood Search: an application to combinatorial optimization problems. *Journal of Global Optimization*, 63:515–536.

Erfani, S. M. H., Danesh, S., Karrabi, S. M., and Shad, R. (2017). A novel approach to find and optimize bin locations and collection routes using a geographic information system. *Waste Management and Research*, 35(7):776–785.

Eskelinen, P., Miettinen, K., Klamroth, K., and Hakanen, J. (2010). Pareto Navigator for interactive nonlinear multiobjective optimization. *OR Spectrum*, 32(1):211–227.

Feo, T. A. and Resende, M. G. (1989). A probabilistic heuristic for a computationally difficult Set Covering Problem. *Operational Research letters*, 8:67–71.

Ferreira, J. A., Costa, M., Tereso, A., and Oliveira, J. A. (2015). A Multi-Criteria Decision Support System for a Routing Problem in Waste Collection. In Gaspar-Cunha A., Henggeler Antunes C., C. C., editor, *Evolutionary Multi-Criterion Optimization: 8th International Conference, EMO 2015, Guimarães, Portugal, March 29 –April 1, 2015. Proceedings, Part II*, volume 9019, pages 388–402. Springer, Cham, Cham.

Fishburn, P. C. (1974). Lexicographic Orders, Utilities and Decision Rules: A Survey. *Management Science*, 20(11):1442–1471.

Fisher, M. L. and Jaikumar, R. (1981). A generalized assignment heuristic for vehicle routing. *Networks*, 11(2):109–124.

Gallardo, A., Carlos, M., Peris, M., and Colomer, F. (2015). Methodology to design a municipal solid waste pre - collection system: A case study. *Waste Management*, 36:1–11.

Gass, S. and Saaty, T. (1955). The computational algorithm for the parametric objective function. *Naval Research Logistics Quarterly*, 2(1-2):39–45.

Gendreau, M. and Potvin, J.-Y. (2010). *Handbook of Metaheuristics*, volume 146. Springer US, 2nd edition.

Ghamlouche, I., Crainic, T., and Gendreau, M. (2004). Path Relinking, Cycle-Based Neighbourhoods and Capacitated Multicommodity Network Design. *Annals of Operations Research*, 131(1-4):109–133.

Ghose, M. K., Dikshit, K., and Sharma, S. (2006). A GIS based transportation model for solid waste disposal–a case study on Asansol municipality. *Waste management*, 26(11):1287–1293.

Ghoseiri, K. and Ghannadpour, S. (2010). Multi-objective Vehicle Routing Problem with Time Windows using Goal Programming and Genetic Algorithm. *Applied Soft Computing*, 10(4):1096–1107.

Gillett, B. E. and Miller, L. R. (1974). A Heuristic Algorithm for the Vehicle-Dispatch Problem. *Operations Research*, 22(2):340–349.

Glover, F. (1977). Heuristics for integer programming using surrogate constraints. *Decision Sciences*, 8(1):156–166.

Glover, F. (1989). Tabu Search — Part I. *ORSA Journal on Computing*, 1(3).

Glover, F. (1990). Tabu Search — Part II. *ORSA Journal on Computing*, 2(1).

Glover, F. (1996). Ejection chains, reference structures and alternating path methods for traveling salesman problems. *Discrete Applied Mathematics*, 65(1):223 – 253.

Glover, F. (1997). Tabu Search and Adaptive Memory Programming: Advances, Applications and Challenges. In Barr, R. S., Helgason, R. V., and Kennington, J. L., editors, *Interfaces in Computer Science and Operations Research: Advances in Metaheuristics, Optimization, and Stochastic Modeling Technologies*, pages 1–75. Springer US, Boston, MA.

Glover, F. and Kochenberger, G. (2003). *Handbook of Metaheuristics*. International Series in Operations Research & Management Science. Springer US.

Glover, F., Laguna, M., and Martí, R. (2000). Fundamentals of scatter search and path relinking. *Control and Cybernetics*, 29(3):653–684.

Golden, B. L. and Assad, A. A. (1988). *Vehicle routing: Methods and studies*, volume 16. Amsterdam ; New York : North-Holland ; New York, N.Y., U.S.A. : Sole distributors for the U.S.A. and Canada, Elsevier Science Pub. Co.

Gómez, J. R., Pacheco, J., Caballero, R., and Molina, J. (2009). Metaheurística MOAMP para un problema de recogida de basuras en ’areas rurales. *Rect@*, 17(1):203.

Gómez, J. R., Pacheco, J., and Gonzalo-Orden, H. (2015). A Tabu Search Method for a Bi-Objective Urban Waste Collection Problem. *Computer-Aided Civil and Infrastructure Engineering*, 30(1):36–53.

Guerrero, L. A., Maas, G., and Hogland, W. (2013). Solid waste management challenges for cities in developing countries. *Waste Management*, 33(1):220 – 232.

Haastrup, P., Maniezzo, V., Mattarelli, M., Rinaldi, F. M., Mendes, I., and Paruccini, M. (1998). A Decision Support System for urban waste management. *European Journal of Operational Research*, 109(2):330 – 341.

Haimes, Y., Lasdon, L., and Wismer, D. (1971). On a Bicriterion Formulation of the Problems of Integrated System Identification and System Optimization. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-1(3):296–297.

Halvorsen-Weare, E. and Savelsbergh, M. (2016). The bi-objective mixed capacitated general routing problem with different route balance criteria. *European Journal of Operational Research*, 251:451–465.

Hanafi, S., Freville, A., and Vaca, P. (1999). Municipal solid waste collection: An effective data structure for solving the sectorization problem with local search methods. *INFOR: Information Systems and Operational Research*, 37(3):236–254.

Hanine, M., Boutkhoum, O., Tikniouine, A., and Agouti, T. (2017). An application of olap/gis-fuzzy ahp-topsis methodology for decision making: Location selection for landfill of industrial wastes as a case study. *KSCE Journal of Civil Engineering*, 21(6):2074–2084.

Hemmelmayr, V., Doerner, K., Hartl, R., and Rath, S. (2009). Metaheuristics for a real world solid waste collection problem. Technical report, Department of Business Administration, University of Vienna.

Hemmelmayr, V., Doerner, K. F., Hartl, R. F., and Rath, S. (2013). A heuristic solution method for node routing based solid waste collection problems. *Journal of Heuristics*, 19:129–156.

Hemmelmayr, V., Doerner, K. F., Hartl, R. F., and Vigo, D. (2014). Models and Algorithms for the Integrated Planning of Bin Allocation and Vehicle Routing in Solid Waste Management. *Transportation Science*, 48(1):103–120.

Ho, S. and Gendreau, M. (2006). Path Relinking for the Vehicle Routing Problem. *Journal of Heuristics*, 12(1):55–72.

Hokkanen, J. and Salminen, P. (1997). Choosing a solid waste management system using multicriteria decision analysis. *European Journal of Operational Research*, 98(1):19 – 36.

Hong, S.-C. and Park, Y.-B. (1998). A heuristic for bi-objective vehicle routing with time window constraints. *International Journal of Production Economics*, 62:249–258.

Hwang, C.-L. and Yoon, K. (1981). Methods for Multiple Attribute Decision Making. In *Multiple Attribute Decision Making: Methods and Applications A State-of-the-Art Survey*, pages 58–191. Springer Berlin Heidelberg, Berlin, Heidelberg.

Iakovou, E. T. (2001). An interactive multiobjective model for the strategic maritime transportation of petroleum products: risk analysis and routing. *Safety Science*, 39(1):19 – 29.

Jaszkiewicz, A. and Branke, J. (2008). Interactive Multiobjective Evolutionary Algorithms. In Branke, J., Deb, K., Miettinen, K., and Słowiński, R., editors, *Multiobjective Optimization: Interactive and Evolutionary Approaches*, pages 179–193. Springer Berlin Heidelberg, Berlin, Heidelberg.

Jaszkiewicz, A. and Slowinski, R. (1999). The Light Beam Search approach: an overview of methodology applications. *European Journal of Operational Research*, 113(2):300 – 314.

Jozefowiez, N., Semet, F., and Talbi, E. (2007a). Target aiming pareto search and its application to the vehicle routing problem with route balancing. *Journal of Heuristics*, 13(5):455.

Jozefowiez, N., Semet, F., and Talbi, E.-G. (2002). Parallel and Hybrid Models for Multi-objective Optimization: Application to the Vehicle Routing Problem. In Guervós, J. J. M., Adamidis, P., Beyer, H.-G., Schwefel, H.-P., and Fernández-Villacañas, J.-L., editors, *Parallel Problem Solving from Nature — PPSN VII: 7th International Conference Granada, Spain, September 7–11, 2002 Proceedings*, pages 271–280. Springer Berlin Heidelberg, Berlin, Heidelberg.

Jozefowiez, N., Semet, F., and Talbi, E. G. (2007b). Multi - objective Vehicle Routing Problems. *European Journal of Operational Research*, 189(2):293–309.

Jozefowiez, N., Semet, F., and Talbi, E.-G. (2009). An evolutionary algorithm for the Vehicle Routing Problem with route balancing. *European Journal of Operational Research*, 195:761–769.

Karadimas, N. V., Papatzelou, K., and Loumos, V. G. (2007). Optimal solid waste collection routes identified by the Ant Colony System algorithm. *Waste Management & Research*, 25(2):139–147.

Keeney, R. and Raiffa, H. (1993). *Decisions with Multiple Objectives: Preferences and Value Trade-Offs*. Wiley series in probability and mathematical statistics. Applied probability and statistics. Cambridge University Press.

Khan, D. and Samadder, S. R. (2014). Municipal solid waste management using Geographical Information System aided methods: A mini review. *Waste Management and Research*, 32(11):1049–1062.

Kim, B.-I., Kim, S., and Sahoo, S. (2006). Waste Collection Vehicle routing problem with time windows. *Computers & Operations Research*, 33:3624–3642.

Kontoravdis, G. and Bard, J. (1995). A GRASP for the Vehicle Routing Problem with Time Windows. *ORSA Journal on Computing*, 7(1):10–23.

Korhonen, P. and Wallenius, J. (1988). A Pareto race. *Naval Research Logistics (NRL)*, 35(6):615–623.

Korhonen, P. J. and Laakso, J. (1986). A visual interactive method for solving the multiple criteria problem. *European Journal of Operational Research*, 24(2):277 – 287.

Lacomme, P., Prins, C., Prodhon, C., and Ren, L. (2015). A Multi-Start based Path Relinking (MSSPR) approach for the vehicle routing problem with route balancing. *Engineering Applications of Artificial Intelligence*, 38:237–251.

Laguna, M. and Martí, R. (1999). GRASP and Path Relinking for 2-layer straight line crossing minimization. *INFORMS Journal of computing*, 11:44–52.

Layeb, A., Ammi, M., and Chikhi, S. (2013). A GRASP algorithm based on new randomized heuristic for Vehicle Routing Problem. *Journal of Computing and Information Technology*, 21(1):35–46.

Lenstra, J. K. and Rinnooy Kan, A. (1981). Complexity of vehicle routing and scheduling problems. *Networks*, 11:221–227.

Lewandowski, A. and Wierzbicki, A. (1989). *Aspiration Based Decision Support Systems: Theory, Software and Applications*. Lecture Notes in Economics and Mathematical Systems. Springer.

Li, X. and He, J. (2009). Intelligent GIS for solving high dimensional site selection problems using Ant Colony System technology. *International Journal of Geographical Information Science*, 23(4):399–416.

Lin, S. (1965). Computer Solutions of the Traveling Salesman Problem. *Bell System Technical Journal*, 44(10):2245–2269.

López-Jaimes, A. and Coello, C. A. C. (2014). Including preferences into a multiobjective evolutionary algorithm to deal with many-objective engineering optimization problems. *Information Sciences*, 277(Supplement C):1 – 20.

López-Sánchez, A., Hernández-Díaz, A., Gorázar, F., and Hinojosa, M. (2017). A multiobjective GRASP - VND algorithm to solve the waste collection problem. *International Transactions in Operational Research*, pages 545 – 567.

López-Sánchez, A., Hernández-Díaz, A., Vigo, D., Caballero, R., and Molina, J. (2014). A multi-start algorithm for a balanced real-world Open Vehicle Routing Problem. *European Journal of Operational Research*, 238(1):104 – 113.

Luque, M., Miettinen, K., Eskelinen, P., and Ruiz, F. (2009). Incorporating preference information in interactive reference point methods for multiobjective optimization. *Omega*, 37(2):450 – 462.

MacDonald, M. L. (1996). A multi-attribute spatial decision support system for solid waste planning. *Computers, Environment and Urban Systems*, 20(1):1 – 17.

Male, J. and Liebman, J. (1978). Districting and routing for solid waste collection. *Journal of the Environmental Engineering Division- ASCE*, 104.

Mandal, S. K., Pacciarelli, D., Løkketangen, A., and Hasle, G. (2015). A memetic NSGA-II for the bi-objective mixed capacitated general routing problem. *Journal of Heuristics*, 21(3):359–390.

Maniezzo, V. and Roffilli, M. (2008). Algorithms for Large Directed Capacitated Arc Routing Problem Instances. In Cotta, C. and van Hemert, J., editors, *Recent Advances in Evolutionary Computation for Combinatorial Optimization*, pages 259–274. Springer Berlin Heidelberg, Berlin, Heidelberg.

Marinakis, Y. (2012). Multiple Phase Neighborhood Search-GRASP for the Capacitated Vehicle Routing Problem. *Expert Systems with Applications*, 39(8):6807 – 6815.

Marinakis, Y., Migdalas, A., and Pardalos, P. M. (2005). Expanding Neighborhood GRASP for the Traveling Salesman Problem. *Computational Optimization and Applications*, 32(3):231–257.

Marks, D. and Liebman, J. (1970). *Mathematical Analysis of Solid Waste Collection: Final Report*. Public Health Service publication. U.S. Government Printing Office.

Marshall, R. E. and Farahbakhsh, K. (2013). Systems approaches to integrated solid waste management in developing countries. *Waste Management*, 33(4):988 – 1003.

Martí, R., Campos, V., Resende, M. G., and Duarte, A. (2015). Multiobjective GRASP with Path Relinking. *European Journal of Operational Research*, 240:54–71.

Martí, R., Velarde, J. L. G., and Duarte, A. (2009). Heuristics for the bi-objective path dissimilarity problem. *Computers & Operations Research*, 36(11):2905 – 2912.

Mateo, P. and Alberto, I. (2012). A mutation operator based on a Pareto anking for multi-objective evolutionary algorithms. *Journal of Heuristics*, 18:53–89.

Miettinen, K. (1999). *Nonlinear multiobjective optimization*. Kluwer Academic Publishers, Boston.

Miettinen, K. (2008). Introduction to Multiobjective Optimization: Noninteractive Approaches. In Branke, J., Deb, K., Miettinen, K., and Słowiński, R., editors, *Multiobjective Optimization: Interactive and Evolutionary Approaches*, pages 1–26. Springer Berlin Heidelberg, Berlin, Heidelberg.

Miettinen, K., Eskelinen, P., Ruiz, F., and Luque, M. (2010). NAUTILUS method: An interactive technique in multiobjective optimization based on the nadir point. *European Journal of Operational Research*, 206(2):426 – 434.

Miettinen, K. and Kirilov, L. (2005). Interactive reference direction approach using implicit parametrization for nonlinear multiobjective optimization. *Journal of Multi-Criteria Decision Analysis*, 13(2-3):115–123.

Miettinen, K. and Mäkelä, M. M. (2000). Interactive multiobjective optimization system WWW-NIMBUS on the Internet. *Computers & Operations Research*, 27(7):709 – 723.

Miettinen, K., M.Mäkelä, M., and Kaario, K. (2006). Experiments with classification-based scalarizing functions in interactive multiobjective optimization. *European Journal of Operational Research*, 175(2):931 – 947.

Miettinen, K., Podkopaev, D., Ruiz, F., and Luque, M. (2015). A new preference handling technique for interactive multiobjective optimization without trading-off. *Journal of Global Optimization*, 63(4):633–652.

Miettinen, K. and Ruiz, F. (2016). NAUTILUS framework: towards trade-off-free interaction in multiobjective optimization. *Journal of Business Economics*, 86(1):5–21.

Mladenovic, N. and Hansen, E. (1997). Variable Neighborhood Search. *Computers and Operations Research*, 24(11):1097–1100.

Mole, R. and Jameson, S. (1976). A sequential route-building algorithm employing a generalized saving criterion. *Operational Research Quarterly*, 27(2):503–511.

Molina, J., Laguna, M., Martí, R., and Caballero, R. (2007). SSPMO: A Scatter Tabu Search Procedure for Non-Linear Multiobjective Optimization. *INFORMS Journal on Computing*, 19(1):91–100.

Molina, J., Santana, L. V., Hernández-Díaz, A. G., Coello, C. A. C., and Caballero, R. (2009). g-dominance: Reference point based dominance for multiobjective metaheuristics. *European Journal of Operational Research*, 197(2):685 – 692.

Nakayama, H. (1995). Aspiration Level Approach to Interactive Multi-Objective Programming and Its Applications. In Pardalos, P. M., Siskos, Y., and Zopounidis, C., editors, *Advances in Multicriteria Analysis*, pages 147–174. Springer US, Boston, MA.

Nakayama, H. and Sawaragi, Y. (1984). Satisficing Trade-off Method for Multiobjective Programming. In Grauer, M. and Wierzbicki, A. P., editors, *Interactive Decision Analysis: Proceedings of an International Workshop on*

Interactive Decision Analysis and Interpretative Computer Intelligence Held at the International Institute for Applied Systems Analysis (IIASA), Laxenburg, Austria September 20–23, 1983, pages 113–122. Springer Berlin Heidelberg, Berlin, Heidelberg.

Nebro, A. J., Luna, F., Alba, E., Dorronsoro, B., Durillo, J. J., and Beham, A. (2008). AbYSS: Adapting Scatter Search to Multiobjective Optimization. *IEEE Transactions on Evolutionary Computation*, 12(4):439–457.

Nguyen-Trong, K., Nguyen-Thi-Ngoc, Nguyen-Ngoc, D., and Dinh-Thi-Hai, V. (2017). Optimization of municipal solid waste transportation by integrating GIS analysis, equiation-based and agent-based model. *Waste Management*, 59:14–22.

Nuortio, T., Kytöjoki, J., Niska, H., and Bräysy, O. (2006). Improved route planning and scheduling of waste collection and transport. *Expert Systems with Applications*, 30(2):223 – 232.

Ombuki-berman, B. M., Andrew, R., and Hanshar, F. (2007). Waste Collection Vehicle Routing Problem with time windows using multi-objective genetic algorithms. Technical report, Brock University, Canada.

Or, I. (1976). *Traveling Salesman Type Combinatorial Problems and their relations to the logistics of blood banking*. PhD thesis, Dpt. of Industrial Engineering and Management Sciences, Northwestwern University.

Osiadacz, A. J. (1986). Multiple criteria optimization: theory, computation, and applications. *Optimal Control Applications and Methods*, 10(1):89–90.

Oyola, J. and Løkketangen, A. (2014). GRASP-ASP: An algorithm for the CVRP with route balancing. *Journal of Heuristics*, 20(4):361–382.

Pacheco, J. (2015). Neva: Sistema para la generación de instancias y representación de soluciones en modelos de rutas, transporte y logística. Registro de la Propiedad Intelectual a nombre de la Universidad de Burgos.

Pacheco, J. and Martí, R. (2006). Tabu Search for a multi-objective routing problem. *Journal of the Operational Research Society*, 57(1):29–37.

Pacheco, J. A. and Delgado, C. R. (1999). Diseño de metaheurísticas para problemas de rutas con flota heterogénea: GRASP. *Cuadernos de Estudios Empresariales*, (9):173–192.

Pacheco, J. A. and Delgado, C. R. (2000). Diseño de metaheuristicos para problemas de rutas con flota heterogenea: concentracion heuristica. *Estudios de economia aplicada*, (14):137–151.

Phelps, S. P. and Köksalan, M. (2003). An Interactive Evolutionary Metaheuristic for Multiobjective Combinatorial Optimization. *Management Science*, 49(12):1726–1738.

Pires, A., Martinho, G., and Chang, N. B. (2011). Solid waste management in European countries : A review of system analysis techniques. *Journal of Environmental Management*, 92:1033–1050.

Pisinger, D. and Ropke, S. (2007). A general heuristic for vehicle routing problems. *Computers & Operational Research*, 34(8):2418–2429.

Podinovskii, V. V. (1994). Criteria importance theory. *Mathematical Social Sciences*, 27(3):237 – 252.

Potvin, J.-Y. and Rousseau, J.-M. (1995). An Exchange Heuristic for Routeing Problems with Time Windows. *The Journal of the Operational Research Society*, 46(12).

Prais, M. and Ribeiro, C. (2000). Reactive GRASP: An application to a matrix decomposition problem in TDMA traffic assignment. *INFORMS Journal of computing*, 12:164–176.

Rahoual, M., Kitoun, B., Mabed, M., Bachelet, V., and Benameur, F. (2001). Multicriteria genetic algorithms for the vehicle routing problem with time windows. In *Metaheuristics International Conference*, pages 527–532.

Reghioui, M., Prins, C., and Labadi, N. (2007). Grasp with path relinking for the capacitated arc routing problem with time windows. In Giacobini, M., editor, *Applications of Evolutionary Computing: EvoWorkshops 2007: EvoCoMnet, EvoFIN, EvoIASP,EvoINTERACTION, EvoMUSART, EvoSTOC and EvoTransLog. Proceedings*, pages 722–731. Springer Berlin Heidelberg, Berlin, Heidelberg.

Resende, M., Martí, R., Gallego, M., and Duarte, A. (2010). GRASP and path relinking for the max-min diversity problem. *Computers and Operations Research*, 37:498–508.

Resende, M. and Werneck, R. (2004). A hybrid heuristic for the p-median problem. *Journal of Heuristics*, 10:59–88.

Resende, M. G. and Ribeiro, C. C. (2016). *Optimization by GRASP*. Springer-Verlag New York.

Romero, C. (1991). *Handbook of critical issues in Goal Programming*. Pergamon Press Oxford.

Rosing, K. and ReVelle, C. (1997). Heuristic concentration: Two stage solution construction. *European Journal of Operational Research*, 97(1):75 – 86.

Roy, B. (1991). The outranking approach and the foundations of electre methods. *Theory and Decision*, 31(1):49–73.

Roy, B. and Mousseau, V. (1996). A Theoretical Framework for Analysing the Notion of Relative Importance of Criteria. *Journal of Multi-Criteria Decision Analysis*, 5(2):145–159.

Ruiz, A. B., Sindhya, K., Miettinen, K., Ruiz, F., and Luque, M. (2015). E-NAUTILUS: A decision support system for complex multiobjective optimization problems based on the NAUTILUS method. *European Journal of Operational Research*, 246(1):218 – 231.

Ruiz, F., Luque, M., and Cabello, J. (2009). A classification of the weighting schemes in reference point procedures for multiobjective programming. *Journal of the Operational Research Society*, 60:544–553.

Ryan, D., Hjorring, C., and Glover, F. (1993). Extensions of the petal method for vehicle routing. *Journal of the Operational Research Society*, 44:289–296.

Santos, L., Coutinho-Rodrigues, J., and Current, J. R. (2008). Implementing a multi-vehicle multi-route spatial decision support system for efficient trash collection in Portugal. *Transportation Research Part A: Policy and Practice*, 42(6):922 – 934.

Sharma, S. (1974). *Operation research*. Meerut.

Simonetto, E. and Borenstein, D. (2007). A Decision Support System for the operational planning of solid waste collection. *Waste Management*, 27(10):1286–1297.

Solomon, M. (1987). Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints. *Operations Research*, 35:254–265.

Soltani, A., Sadiq, R., and Hewage, K. (2017). The impacts of decision uncertainty on municipal solid waste management. *Journal of Environmental Management*, 197(Supplement C):305 – 315.

Sorensen, K. and Schittekat, P. (2013). Statistical analysis of distance-based path relinking for the capacitated vehicle routing problem. *Computers and Operations Research*, 40:3197–3205.

Stanisavljevic, N., Vujovic, S., Zivancev, M., Batinic, B., Tot, B., and Ubavin, D. (2015). Application of MFA as a decision support tool for waste management in small municipalities. Case study of Serbia. *Waste Management & Research*, 33(6):550–560. PMID: 26060233.

Steuer, R. E. and Choo, E.-U. (1983). An interactive weighted Tchebycheff procedure for multiple objective programming. *Mathematical Programming*, 26(3):326–344.

Stewart, T., Bandte, O., Braun, H., Chakraborti, N., Ehrgott, M., Göbelt, M., Jin, Y., Nakayama, H., Poles, S., and Di Stefano, D. (2008). Real-World Applications of Multiobjective Optimization. In Branke, J., Deb, K., Miettinen, K., and Słowiński, R., editors, *Multiobjective Optimization: Interactive and Evolutionary Approaches*, pages 285–327. Springer Berlin Heidelberg, Berlin, Heidelberg.

Subramanian, A., Battarra, M., and Potts, C. N. (2014). An Iterated Local Search heuristic for the single machine total weighted tardiness scheduling problem with sequence-dependent setup times. *International Journal of Production Research*, 52(9):2729–2742.

Tavares, G., Zsigraiova, Z., Semiao, V., and Carvalho, M. (2009). Optimisation of WSW collection routes for minimun fuel consumption using 3D GIS modeling. *Waste Management*, 29:1176–1185.

Toth, P. and Vigo, D. (2002). *The Vehicle Routing Problem*. Monographs on Discrete Mathematics and Applications. Society for Industrial and Applied Mathematics.

Toth, P. and Vigo, D. (2014). *Vehicle Routing: Problems, Methods, and Applications*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2nd edition.

Tung, D. V. and Pinnoi, A. (2000). Vehicle routing - scheduling for waste collection in Hanoi. *European Journal of Operational Research*, 125(3):449 – 468.

Uchoa, E., Pecin, D., Pessoa, A., Poggi, M., Vidal, T., and Subramanian, A. (2017). New benchmark instances for the Capacitated Vehicle Routing Problem. *European Journal of Operational Research*, 257(3):845 – 858.

Vidal, T., Maculan, N., Ochi, L. S., and Penna, P. H. V. (2014). Large neighborhoods with implicit customer selection for vehicle routing problems with profits. Working paper, LIDS (Massachusetts Institute of Technology), POM (University of Vienna), Universidade Federal do Rio de Janeiro, Instituto de Computaçao (Universidade Federal Fluminense).

Viotti, P., Pelettini, A., Pomi, R., and Innocetti, C. (2003). Genetic algorithms as a promising tool for optimisation of the MSW collection routes. *Waste Management and Research*, 21:292–298.

Wang, J. Y. and Wright, J. R. (1994). Interactive Design of Service Routes. *Journal of Transportation Engineering*, 120(6):897–913.

Wierzbicki, A. P. (1977). Basic properties of scalarizmg functionals for multiobjective optimization. *Mathematische Operationsforschung und Statistik. Series Optimization*, 8(1):55–60.

Wierzbicki, A. P. (1979). The use of reference objectives in multiobjective optimization - theoretical implications and practical experience. Iiasa working paper, International Institute for Applied Systems AnalysisLaxenburgAustria, IIASA, Laxenburg, Austria.

Wierzbicki, A. P. (1980). The Use of Reference Objectives in Multiobjective Optimization. In Fandel, G. and Gal, T., editors, *Multiple Criteria Decision Making Theory and Application*, pages 468–486. Springer Berlin Heidelberg, Berlin, Heidelberg.

Wierzbicki, A. P. (1982). A mathematical basis for satisficing decision making. *Mathematical Modelling*, 3(5):391 – 405. Special IIASA Issue.

Wright, J. (1994). *Computer-aided System for Planning Efficient Routes*. PhD thesis.

Xi, B., Su, J., Huang, G., Qin, X., Jiang, Y., Huo, S., Ji, D., and Yao, B. (2010). An integrated optimization approach and multi-criteria decision analysis for supporting the waste-management system of the city of Beijing, China. *Engineering Applications of Artificial Intelligence*, 23(4):620 – 631.

Xue, W. and Cao, K. (2016). Optimal routing for waste collection : a case study in Singapore. *International Journal of Geographical Information Science*, 30(3):554–572.

Zadeh, L. (1963). Optimality and non-scalar-valued performance criteria. *IEEE Transactions on Automatic Control*, 8(1):59–60.

Zhou, W., Song, T., He, F., and Liu, X. (2013). Multiobjective Vehicle Routing Problem with Route Balance Based on Genetic Algorithm. *Discrete Dynamics in Nature and Society*, 2013(2013):9.

Zitzler, E. (1999). *Evolutionary Algorithms for Multiobjective Optimization: Methods and Applications*. PhD thesis.