UNIVERSITAT OBERTA DE CATALUNYA

FACULTY OF COMPUTER SCIENCE, MULTIMEDIA AND TELECOMMUNICATIONS

UNIVERSIDAD DE LA SABANA

SCHOOL OF ENGINEERING

Optimizing Transportation Systems and Logistics Network Configurations: from Biased-Randomized Algorithms to Fuzzy Simheuristics

A thesis submitted in fulfillment of the requirements for the degrees of

Doctor of Philosophy in Network & Information Technologies

from the UNIVERSITAT OBERTA DE CATALUNYA and

Doctor of Philosophy in Logistics & Supply Chain Management

from the UNIVERSIDAD DE LA SABANA

Author:

Rafael David Tordecilla-Madera

UOC Committee Members:

Prof. Dr. Javier Panadero

Prof. Dr. Angel A. Juan

Unisabana Committee Members:

Prof. Dr. Carlos L. Quintero-Araujo

Prof. Dr. Jairo R. Montoya-Torres

Barcelona, Spain

February 23, 2022

"There is no gene for the human spirit."

Gattaca - Andrew Niccol

"A lesson without pain is meaningless. That's because you cannot gain anything without sacrificing something else in return. But once you have endured it and overcome it, you shall obtain a powerful, unmatched heart. A fullmetal heart."

Fullmetal Alchemist - Hiromu Arakawa

Acknowledgements

Although it is only my name which appears as the author of this thesis, I really would not be at this point if many people who have been by my side had not given me their support, both personally and academically. Of course, the most important ones, the ones that have accompanied me since I was born, are my parents, Mrs. Leila and Mr. Rafa. Without you none of this would have been possible. Your unconditional support for years and years is invaluable. My lifetime will not be enough to thank you for so much support, both material and immaterial. This is yours!

To my supervisors: Ángel Juan, director of the ICSO group, who gave me the opportunity to belong to this wonderful group in which, motivated by him and by his passion for whatever he does, we not only learn about heuristics or about publishing scientific articles, but also a lot about teamwork, where the knowledge of each person complements the one of the colleagues. Jairo Montoya, director of my doctorate at Unisabana, who was the first contact I had in this adventure, who motivated me to apply for the scholarship offered by the UOC, and who was a great support with my academic and administrative doubts and requirements regarding the co-supervision between both universities. Javier Panadero, who was the most closest supervisor when developing some projects and writing articles. And Carlos Quintero, who was always there to give me his technical and academic support. Thank you all for guiding me and sharing your knowledge with me.

To my sister Carolina in Montería, for always being there, looking after me and being attentive that everything goes well with my life. To my family in Girona: my aunt Edna and Pepe, for being a strong support in this walk through European lands since day 1. To my other family in Barcelona: Andrés, Rosa and Fabián, for welcoming me into their home and making me feel like one more member of their family all these years. I will never forget the shabbats. To all my friends and colleagues from the ICSO group and the UOC, especially Leandro, Juliana, Sara, Pedro, María, Cristina, Santiago: thank you for your support and the happy moments shared. To the Colombian combo that accompanied me in my first months in Barcelona: Julio, Carolina and Claudia. And to all the people who cared about me in these 3 years, who always had a greeting or a call for me: my grandmother Enadys, my aunts, my friends in Colombia.

To the Unity, the Energy, the Force, that invisible entity that some people call God and that represents that being with whom you talk when you are alone, the one that moves you forward. I needed much of this, and there he/she/it was, making me grow not only as an academic but, and more importantly, as a person. And finally, to you, Julia, because wherever you are in the multiverse, I know that you would be happy to see me reach this point, to see me in Europe studying, learning, traveling, knowing, fulfilling dreams. Your body left very soon, but your energy will live for many more years, because whoever is remembered does not die, and even less someone with too much desire to live like you.

Agradecimientos

A pesar de que solo sea mi nombre el que aparezca como autor de esta tesis, realmente no estaría en este punto si no fuera por el apoyo de muchas personas que han estado a mi lado, tanto en el ámbito personal como en el académico. Por supuesto, las más importantes, las que me han acompañado desde que nací, son mis padres, doña Leila y don Rafa. Sin ustedes nada de esto habría sido posible. Su apoyo incondicional por años y años es algo invaluable. No me alcanzará la existencia para agradecerles por tanto apoyo, material e inmaterial. ¡Esto es de ustedes!

A mis directores: Ángel Juan, director del grupo ICSO, quien me dio la oportunidad de pertenecer a este maravilloso grupo en el cual, motivado por él y por su pasión por lo que hace, no solo se aprende sobre heurísticas o a publicar artículos científicos, sino también a trabajar mucho en equipo, en donde los conocimientos de cada persona complementan los de los otros compañeros. Jairo Montoya, director de mi doctorado en Unisabana, quien fue el primer contacto que tuve en esta aventura, quien me motivó a presentarme a la beca ofrecida por la UOC, y quien fue un gran apoyo en mis dudas y requerimientos académicos y administrativos relacionados con la cotutela entre ambas universidades. Javier Panadero, quien de mis 4 directores fue con quien más cerca trabajé en el desarrollo de algunos proyectos y escritura de artículos. Y Carlos Quintero, quien siempre estuvo ahí para brindarme su apoyo técnico y académico. A todos gracias por su guía y compartir conmigo sus conocimientos.

A mi hermana Carolina en Montería, por siempre estar ahí atenta y pendiente de mí, de que todo marchara bien con mi vida. A mi familia en Girona: mi tía Edna y Pepe, por ser un fuerte soporte en este andar por tierras europeas desde el día 1. A mi otra familia en Barcelona: Andrés, Rosa y Fabián, por acogerme en su casa y hacerme sentir como un miembro más de su familia todos estos años. Nunca olvidaré los shabbats. A todos mis amigos y compañeros del grupo ICSO y de la UOC, especialmente Leandro, Juliana, Sara, Pedro, María, Cristina, Santiago: gracias por el apoyo y los alegres momentos compartidos. Al combo colombiano que me acompañó en mis primeros meses en Barcelona: Julio, Carolina y Claudia. Y a todas las personas que se preocuparon por mí en estos 3 años, que siempre tenían para mí un saludo o una llamada: mi abuela Enadys, mis tías, mis amigos y amigas en Colombia.

A la Unidad, la Energía, la Fuerza, esa entidad invisible que algunos llaman Dios y que representa ese ser con quien hablas cuando estás solo, el que te mueve hacia adelante. Mucho de esto necesité y ahí estuvo, haciéndome crecer no solo como académico sino, y más importante aún, como persona. Y finalmente, a ti, Julia, porque en cualquier parte del multiverso en la que te encuentres, sé que te alegrarías de verme llegar hasta aquí, de verme en Europa estudiando, aprendiendo, viajando, conociendo, cumpliendo sueños. Tu cuerpo se fue muy pronto, pero tu energía vivirá por muchos años más, pues no muere quien es recordado, y menos alguien con tantas ganas de vivir como tú.

Contents

List of Figures

List of Tables

List of Algorithms

Nomenclature

ACO Ant Colony Optimization

AFRVRP Agri-Food Rich Vehicle Routing Problem

APL Automated Parcel Locker

ARP Arc Routing Problem

BH Backhaul

BKS Best-Known Solution

BR Biased-Randomized

BRKGA Biased Random-Key Genetic Algorithm

COP Combinatorial Optimization Problem

COVID-19 Coronavirus Disease 2019

CWS Clarke & Write Savings

DES Discrete-Event Simulation

DRSP Dynamic Ride-Sharing Problem

EV Electric Vehicle

FLP Facility Location Problem

FLPSD Facility Location Problem with Stochastic Demands

GA Genetic Algorithm

GRASP Greedy Randomized Adaptive Search Procedure

ILS Iterated Local Search

ISI Institute for Scientific Information

JCR Journal Citation Reports

KPI Key Performance Indicator

LB Lower Bound

LH Linehaul

LRPFS Location Routing Problem with Facility-Sizing decisions

LRPFS-S/F-D Location Routing Problem with Facility-Sizing decisions and Stochastic and Fuzzy Demands

LRPFSSD Location Routing Problem with Facility-Sizing decisions and Stochastic Demands

LS Local Search

MATL Maximum Allowed Tour Length

MCS Monte Carlo Simulation

MDVRP Multi-Depot Vehicle Routing Problem
MILP Mixed-Integer Linear Programming
MTL Maximum Tour Length
NSGA Non-dominated Sorting Genetic Algorithm
OP Orienteering Problem
OVRP Open Vehicle Routing Problem
OVRPSSTT Open Vehicle Routing Problem with Stochastic Service and Travel Times
PPE Personal Protective Equipment
PSO Particle Swarm Optimization
R-BR Reactive Biased-Randomized
RSP Ride-Sharing Problem
RTI Returnable Transport Item
RTOP Rich Team Orienteering Problem
RVRP Rich Vehicle Routing Problem
SA Simulated Annealing
SARS-CoV-2 Severe Acute Respiratory Syndrome Coronavirus 2
SCN Supply Chain Network
SCND Supply Chain Network Design
SL Savings List
SO Simulation-Optimization
STOP Stochastic Team Orienteering Problem
T&L Transportation and Logistics
TOP Team Orienteering Problem
TOP-S/F-TT Team Orienteering Problem with Stochastic and Fuzzy Travel Times
TOPSSTT Team Orienteering Problem with Stochastic Service and Travel Times
TS Tabu Search
UB Upper Bound
UL Urban Logistics
VNS Variable Neighborhood Search
VRP Vehicle Routing Problem
VRP-S/F-D Vehicle Routing Problem with Stochastic and Fuzzy Demands
VRPB Vehicle Routing Problem with Backhauls
VRPOB Vehicle Routing Problem with Optional Backhauls
WOS Web of Science

Abstract

Transportation and logistics (T&L) are currently highly relevant functions in any competitive industry. Locating facilities or distributing goods to hundreds or thousands of customers are activities with a high degree of complexity, regardless of whether facilities and customers are placed all over the globe or in the same city. A countless number of alternative strategic, tactical, and operational decisions can be made in T&L systems; hence, reaching an optimal solution –e.g., a solution with the minimum cost or the maximum profit– is a really difficult challenge, even by the most powerful existing computers. Approximate methods, such as heuristics, metaheuristics, and simheuristics, are then proposed to solve T&L problems. They do not guarantee optimal results, but they yield good solutions in short computational times. These characteristics become even more important when considering uncertainty conditions, since they increase T&L problems' complexity. Modeling uncertainty implies to introduce complex mathematical formulas and procedures, however, the model realism increases and, therefore, also its reliability to represent real world situations. Stochastic approaches, which require the use of probability distributions, are one of the most employed approaches to model uncertain parameters. Alternatively, if the real world does not provide enough information to reliably estimate a probability distribution, then fuzzy logic approaches become an alternative to model uncertainty. Hence, the main objective of this thesis is to design hybrid algorithms that combine fuzzy and stochastic simulation with approximate and exact methods to solve T&L problems considering operational, tactical, and strategic decision levels. This thesis is organized following a layered structure, in which each introduced layer enriches the previous one. Therefore, biased-randomized heuristics and metaheuristics are firstly explained to solve T&L problems that only include deterministic parameters. Later, Monte Carlo simulation is introduced to these approaches to deal with stochastic parameters. Finally, fuzzy simheuristics are employed to address simultaneously fuzzy and stochastic uncertainty. A series of numerical experiments are designed to test the proposed algorithms, using real-world, newly-created, and benchmark instances. Obtained results demonstrate both the cost- and time-efficiency of the designed algorithms, as well as their reliability to solve realistic problems including uncertainty and multiple constraints and conditions that enrich all the addressed problems.

Resumen

El transporte y la logística (T&L) son actualmente funciones de gran relevancia en cualquier industria competitiva. La localización de instalaciones o la distribución de mercancías a cientos o miles de clientes son actividades con un alto grado de complejidad, independientemente de si las instalaciones y los clientes se encuentran en todo el mundo o en la misma ciudad. En los sistemas de T&L se pueden tomar un sinnúmero de decisiones alternativas estratégicas, tácticas y operativas; por lo tanto, llegar a una solución óptima –por ejemplo, una solución con el mínimo costo o la máxima utilidad– es un desafío realmente difícil, incluso para las computadoras más potentes que existen hoy en día. Así pues, métodos aproximados, tales como heurísticas, metaheurísticas y simheurísticas, son propuestos para resolver problemas de T&L. Estos métodos no garantizan resultados óptimos, pero ofrecen buenas soluciones en tiempos computacionales cortos. Estas características se vuelven aún más importantes cuando se consideran condiciones de incertidumbre, ya que estas aumentan la complejidad de los problemas de T&L. Modelar la incertidumbre implica introducir fórmulas y procedimientos matemáticos complejos, sin embargo, el realismo del modelo aumenta y, por lo tanto, también su confiabilidad para representar situaciones del mundo real. Los enfoques estocásticos, que requieren el uso de distribuciones de probabilidad, son uno de los enfoques más empleados para modelar parámetros inciertos. Alternativamente, si el mundo real no proporciona suficiente información para estimar de manera confiable una distribución de probabilidad, los enfoques que hacen uso de lógica difusa se convierten en una alternativa para modelar la incertidumbre. Así pues, el objetivo principal de esta tesis es diseñar algoritmos híbridos que combinen simulación difusa y estocástica con métodos aproximados y exactos para resolver problemas de T&L considerando niveles de decisión operativos, tácticos y estratégicos. Esta tesis se organiza siguiendo una estructura por capas, en la que cada capa introducida enriquece a la anterior. Por lo tanto, en primer lugar se exponen heurísticas y metaheurísticas sesgadas-aleatorizadas para resolver problemas de T&L que solo incluyen parámetros determinísticos. Posteriormente, la simulación Monte Carlo se agrega a estos enfoques para modelar parámetros estocásticos. Por último, se emplean simheurísticas difusas para abordar simultáneamente la incertidumbre difusa y estocástica. Una serie de experimentos numéricos es diseñada para probar los algoritmos propuestos, utilizando instancias de referencia, instancias nuevas e instancias del mundo real. Los resultados obtenidos demuestran la eficiencia de los algoritmos diseñados, tanto en costo como en tiempo, así como su confiabilidad para resolver problemas realistas que incluyen incertidumbre y múltiples restricciones y condiciones que enriquecen todos los problemas abordados.

Resum

El transport i la logística (T&L) són actualment funcions de gran rellevància a qualsevol indústria competitiva. La localització d'instal·lacions o la distribució de mercaderies a centenars o milers de clients són activitats amb un alt grau de complexitat, independentment de si les instal·lacions i els clients es troben a tot el món oa la mateixa ciutat. En els sistemes de T&L es poden prendre un gran nombre de decisions alternatives estratègiques, tàctiques i operatives; per tant, arribar a una solució òptima –per exemple, una solució amb el mínim cost o la màxima utilitat– és un desafiament realment difícil, fins i tot per als ordinadors més potents que hi ha avui dia. Així doncs, mètodes aproximats, tals com a heurístiques, metaheurístiques i simheurístiques, són proposats per resoldre problemes de T&L. Aquests mètodes no garanteixen resultats òptims, però ofereixen bones solucions en temps computacionals curts. Aquestes característiques esdevenen encara més importants quan es consideren condicions d'incertesa, ja que augmenten la complexitat dels problemes de T&L. Modelar la incertesa implica introduir fórmules i procediments matemàtics complexos, però el realisme del model augmenta i, per tant, també la seva confiabilitat per representar situacions del món real. Els enfocaments estocàstics, que requereixen l'ús de distribucions de probabilitat, són un dels enfocaments més emprats per modelar paràmetres incerts. Alternativament, si el món real no proporciona prou informació per estimar de manera fiable una distribució de probabilitat, els enfocaments que fan ús de lògica difusa es converteixen en una alternativa per modelar la incertesa. Així doncs, l'objectiu principal d'aquesta tesi és dissenyar algorismes híbrids que combinin simulació difusa i estocàstica amb mètodes aproximats i exactes per resoldre problemes de T&L considerant nivells de decisió operatius, tàctics i estratègics. Aquesta tesi s'organitza seguint una estructura per capes, on cada capa introduïda enriqueix l'anterior. Per tant, en primer lloc s'exposen heurístiques i metaheurístiques esbiaixades-aleatoritzades per resoldre problemes de T&L que només inclouen paràmetres determinístics. Posteriorment, la simulació Monte Carlo s'afegeix a aquests enfocaments per modelar paràmetres estocàstics. Finalment, es fan servir simheurístiques difuses per abordar simultàniament la incertesa difusa i estocàstica. Una sèrie d'experiments numèrics és dissenyada per provar els algorismes proposats, utilitzant instàncies de referència, instàncies noves i instàncies del món real. Els resultats obtinguts demostren l'eficiència dels algorismes dissenyats, tant en cost com en temps, així com la seva confiabilitat per resoldre problemes realistes que inclouen incertesa i múltiples restriccions i condicions que enriqueixen tots els problemes abordats.

Chapter 1

Introduction

1.1 General Overview

Transportation and logistics (T&L) activities are core functions in any modern industry that seeks for competitiveness in a globalized world, where competitors can be located anywhere around the planet. These functions cover all decision levels, i.e., they can be strategic, tactical, or operational, corresponding to decisions made in a long-, medium-, and short-term basis, respectively. Regardless of this level, the scientific literature has demonstrated that realistic T&L problems are NP-hard (Nagy and Salhi, 2007), i.e., they are computationally complex and, therefore, the number of alternative solutions grows exponentially when the instance size grows linearly. This fact makes highly difficult to find an optimal solution –e.g., a solution that yields the minimum cost, or the maximum profit– in short computing times when the instance size is medium or large, even by the most powerful existing computers. Nevertheless, many real-world problems require agile solutions, i.e., solutions that can be obtained in a matter of minutes, seconds, or even in real time. Smart cities (Faulin et al., 2018), internet of things (IoT) (Lin et al., 2017), autonomous vehicles (Bagloee et al., 2016), supply chain disruptions (Snyder et al., 2016), humanitarian logistics (Holguín-Veras et al., 2012), or healthcare (Hiermann et al., 2015) are only a few examples of applications requiring fast solutions. In this context, decision-makers usually prefer a non-optimal but good solution that can be attained quickly. Hence, approximate methods, such as heuristics, metaheuristics, or simheuristics, are frequently employed since they have been proved to be very efficient to obtain high-quality solutions to a myriad of problems, including those related to T&L (Juan et al., 2015a).

Different constraints and conditions make realistic T&L problems even more complex and challenging. Examples of these constraints are: multiple decision levels –i.e., strategic, tactical, and operational– considered at the same time (Prodhon and Prins, 2014), multi-compartment vehicles to transport different types of products that cannot be mixed (Derigs et al., 2011), multiple interrelated periods (Nickel et al., 2012), inventory decisions (Coelho et al., 2013), non-smooth objective functions (Bagirov and Yearwood, 2006), dynamic conditions (Fikar et al., 2016), product deliveries and pickups (Parragh et al., 2008), facility sizing decisions (Correia et al., 2010), limited driving range (Panadero et al., 2020b), resilience (Carvalho et al., 2012), or uncertainty conditions (Govindan et al., 2017). Accordingly, new agile and flexible solving methods are necessary to provide solutions to real-world problems with these characteristics. Hence, this thesis proposes multiple methods to solve different problems including every aforementioned constraint, as well as additional restrictions that will be explained in every chapter. Most proposed methods are approximate. Moreover, a few mixed-integer linear programs (MILP) are proposed to either have some references of optimal solutions for comparison purposes, or show the potential of simulation to deal with uncertainty in a simulation-optimization approach.

Uncertainty is a pervasive characteristic of all real-world T&L activities. However, considering uncertainty when modeling this type of problems increases their complexity (Simangunsong et al., 2012). Hence, uncertainty is frequently ignored to design more tractable mathematical models and solution approaches. For instance, the most basic versions of heuristic and metaheuristic algorithms do not include uncertain parameters. Probability distributions have been one of the most used approaches to model uncertain parameters. This case is called by the literature as a stochastic approach. However, this approach is helpful as long as we have enough information to reliably estimate all parameters of any probability distribution. This estimation is made performing a thorough input analysis procedure (Law, 2013), which requires sufficient input data to yield statistically significant conclusions. Conversely, if the real world provides only a low quantity of data, or the available information has poor quality (Corlu et al., 2020), then fuzzy approaches become an alternative to model uncertain parameters. Instead of calculating the probability of occurrence of an event, e.g., that a customer has a certain level of demand, fuzzy logic allows this demand to be true or false with a given degree of membership by employing some preset linguistic rules.

The different uncertainty approaches and the different decision levels in T&L are the main criteria to classify the problems addressed in this thesis. Figure 1.1 displays a schema where this classification is illustrated. Decision levels are displayed in the horizontal axis and solution approaches are depicted in the vertical axis. The use of each solution approach depends on the considered uncertainty level. Hence, biased-randomized (BR) heuristics – which are algorithms that introduce a skewed random behavior to select the next movement from a list of candidates when constructing a solution (Grasas et al., 2017)– and metaheuristics do not include uncertain parameters, i.e., all input parameters values are known in advance. The difference between both approaches is that the employed metaheuristics embed the BR heuristics into a more complex process that includes additional procedures, e.g., destruction-reconstruction strategies. The next level is the inclusion of Monte Carlo simulation (MCS) to address uncertainty. Initially, a simulation-optimization (SO) approach that combines a MILP model –i.e., an exact algorithm– with MCS is proposed. This approach serves as a transition to propose more complex procedures in what is called a simheuristic, which is an algorithm that combines metaheuristics with MCS to efficiently address NP-hard problems that include stochastic parameters. Finally, the use of fuzzy simheuristics is proposed to deal with problems including both stochastic and fuzzy parameters. The acronyms employed in Figure 1.1 are listed below. The main characteristics of each problem and each proposed solution approach are thoroughly explained in Chapter 2.

- RVRP: rich vehicle routing problem (VRP).

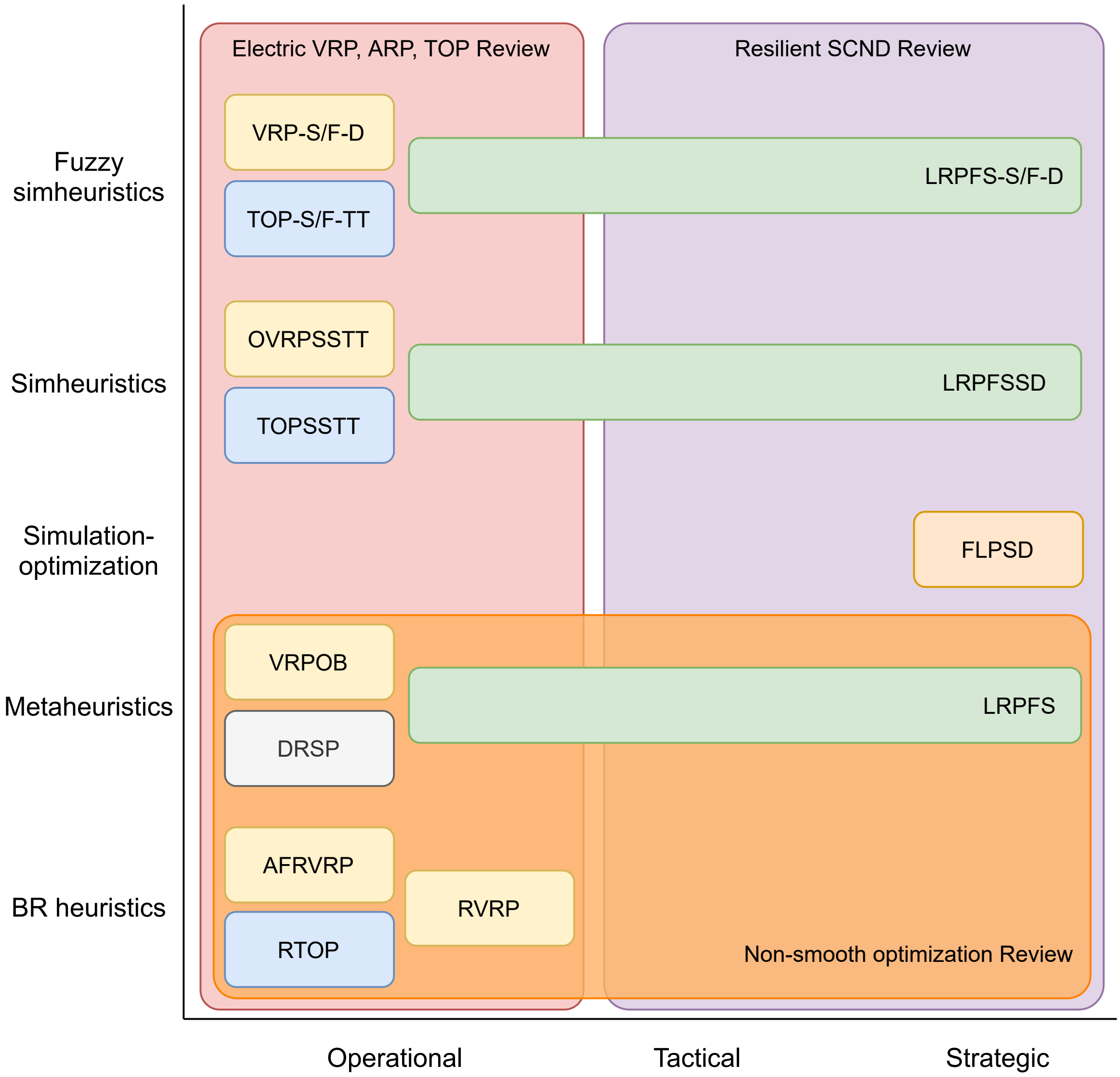


Figure 1.1: Classification schema of the problems addressed in this thesis.

- AFRVRP: agri-food rich VRP.
- RTOP: rich team orienteering problem (TOP).
- VRPOB: VRP with optional backhauls.
- DRSP: dynamic ride-sharing problem.
- LRPFS: location routing problem (LRP) with facility-sizing decisions.
- FLPSD: facility location problem (FLP) with stochastic demands.
- OVRPSSTT: open VRP with stochastic service and travel times.
- TOPSSTT: TOP with stochastic service and travel times.
- LRPFSSD: LRP with facility-sizing decisions and stochastic demands.

- VRP-S/F-D: VRP with stochastic and fuzzy demands.
- TOP-S/F-TT: TOP with stochastic and fuzzy travel times.
- LRPFS-S/F-D: LRP with facility-sizing decisions and stochastic and fuzzy demands.
- ARP: arc routing problem.
- SCND: supply chain network design.

Problems with the same color in Figure 1.1 belong to the same family, e.g., TOPs are blue. Most addressed problems are operational, i.e., they deal with short-term decisions that must be made in a daily basis. This type of problems are part of what is called "transportation systems". Conversely, only the FLPSD is a pure strategic problem, and it is part of what is called "logistics network configurations", since the main objective of this problem is to find a strategic configuration that minimizes the design costs of the logistics network. Finally, one of the most complex challenges in this thesis is the LRPFS, since strategic, tactical, and operational decisions must be made, i.e., it is a combination of the FLP and the VRP and, therefore, it includes both transportation problems and logistics network configurations. Additionally, facility-sizing decisions are considered, i.e., the facilities' capacity level is an additional variable to model. Finally, three review processes that produced three articles were developed, which provide a framework to all other problems:

- A review considering the use of electric vehicles in VRPs, ARPs, and TOPs. This review includes pure operational decisions and all types of solution approaches.
- A review studying SO methods to design resilient supply chain networks. This review covers strategic-tactical decisions and all types of solution approaches.
- A review addressing the use of BR heuristics and metaheuristics to solve non-smooth optimization problems. This review covers all decision levels.

Most solution approaches in this thesis are embedded into a bigger and more complete procedure, forming a layered structure, as Figure 1.2 shows. BR heuristics are the "most basic" solution method and are employed to solve three real-world rich problems. Then, BR heuristics are always included as an important part of the rest of the proposed algorithms, since they lead to a broader exploration of the solution space and, therefore, they improve the performance of these algorithms. For instance, a BR discrete-event driven metaheuristic is used to solve a DRSP, while a VRPOB and an LRPFS are solved by a BR iterated local search (ILS) metaheuristic. Subsequently, parameters such as customers' demands, service times, or travel times, which were assumed to be deterministic when using the BR heuristic and metaheuristic algorithms, are considered as stochastic in the LRPFSSD, the OVRPSSTT, and the TOPSSTT. In this case, a simheuristic approach is proposed. Finally, the last layer of this thesis proposal corresponds to the inclusion of fuzzy parameters. The proposed solution approach is called "fuzzy simheuristics". It is an extension of simheuristics since the addressed problems consider that a single parameter can be either stochastic or fuzzy depending on the studied element. For instance, a subset of customers in a VRP has a stochastic demand, while other subset has a fuzzy demand. Finally, it is important to clarify that the FLPSD is not included in Figure 1.2 since the SO approach employed to solve it is a hybridization between a MILP and MCS, i.e., no approximate methods were used. Hence, it cannot be part of this layered schema, but it shows how simulation can be included into an optimization approach to solve a stochastic logistics problem. Additionally, addressing an FLP allows to have a better understanding of the LRP, since the latter is a combination between the FLP and the VRP.

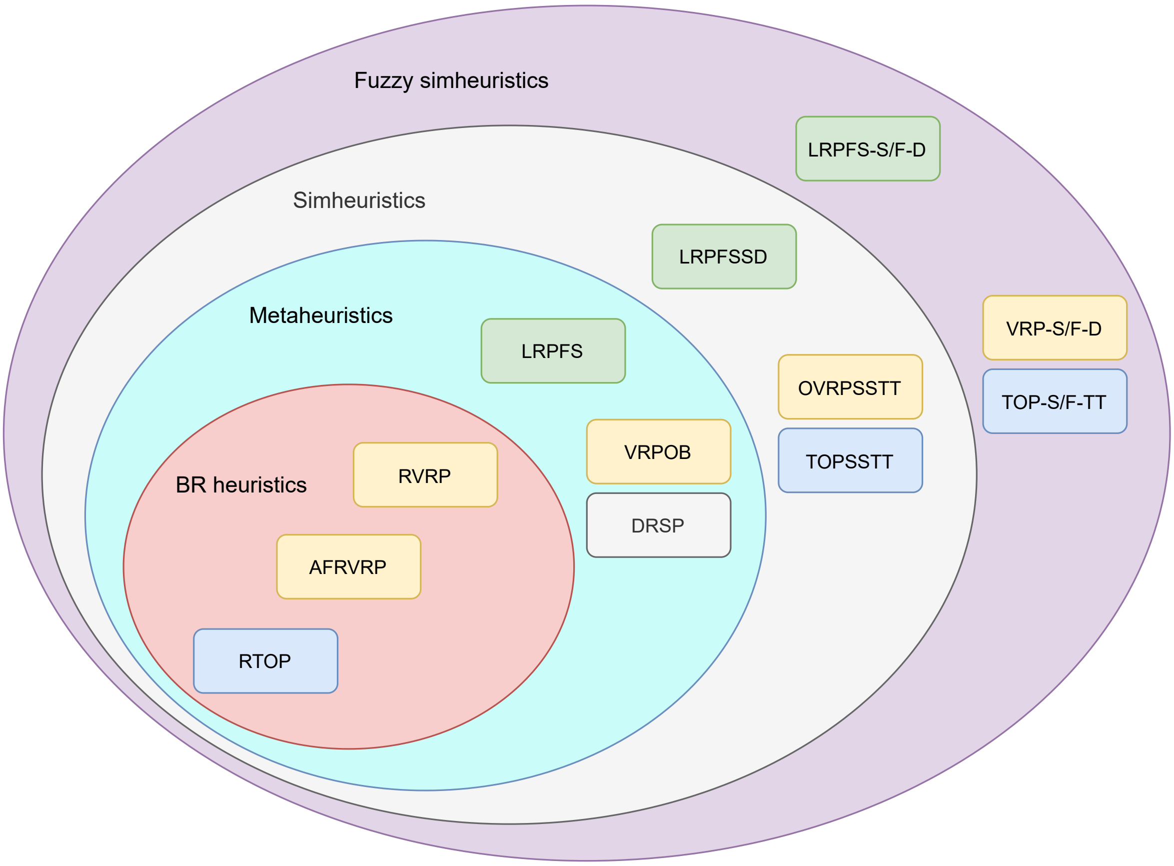


Figure 1.2: Relation between the approximate methods proposed in this thesis.

1.2 Objectives and Original Contribution

The main objective of this thesis is to design hybrid algorithms that combine fuzzy and stochastic simulation with approximate and exact methods to solve T&L problems considering operational, tactical, and strategic decision levels. Hence, the main contribution of this thesis is the proposal of a generic fuzzy simheuristic approach to solve problems regarding transportation systems and logistics network configurations, considering simultaneously all decision levels. This approach is generic because: *(i)* it includes pure deterministic, pure stochastic, and pure fuzzy approaches as particular cases; and *(ii)* it is tested in an LRPFS, which is a generic problem that not only includes the VRP and the FLP as particular cases, but also considers facility-sizing decisions, i.e., the traditional LRP is also a particular case of our addressed problem. Multiple specific objectives contribute to achieve this main objective:

1. To develop BR heuristics and metaheuristics to solve deterministic rich T&L problems.
2. To design a hybrid approach that combines MILP with MCS to solve an FLPSD.
3. To extend the developed heuristics and metaheuristics into simheuristic algorithms, by including stochastic parameters and MCS.
4. To propose a procedure to deal simultaneously with fuzzy and stochastic uncertainty in T&L problems.
5. To design computational and numerical experiments to test the performance of the aforementioned algorithms, employing real-world, newly-created, or benchmark instances.

1.3 Outline

All results shown in this thesis are part of multiple research outcomes that apply different solution approaches to T&L problems in all decision levels. Most outcomes have been already published at the time of writing this thesis. Section 8.3 summarizes the complete scientific production. Moreover, Appendix B.2 shows the cover page of each outcome. All these articles and conference papers contribute to Chapter 2, which shows a thorough literature review about the T&L problems addressed in this thesis, as well as about the proposed solution approaches. The subsequent chapters follow the structure depicted in the vertical axis of Figure 1.1 from bottom to top. Initially, Chapter 3 presents two real-world applications of BR heuristics to solve an RVRP, an RTOP, and an AFRVRP. Chapter 4 shows one application of a discrete-event driven metaheuristic to solve a DRSP, and two applications of an ILS metaheuristic to solve both a VRPOB and a LRPFS. Chapter 5 presents an FLPSD that is solved by using a hybrid SO approach combining a MILP and MCS. Chapter 6 shows two applications of BR simheuristics to solve both an OVRPSSTT and a TOPSSTT. Additionally, this chapter extends the LRPFS addressed in Chapter 4 to include stochastic demands. An ILS-based simheuristic is proposed to solve this problem. Chapter 7 presents two applications of fuzzy simheuristics to solve both a VRP-S/F-D and a TOP-S/F-TT. Furthermore, this chapter extends the LRPFSSD addressed in Chapter 6 to include fuzzy demands. An ILS-based fuzzy simheuristic is proposed to solve this problem. Finally, Chapter 8 outlines the general concluding remarks of this thesis, as well as some future research lines and the research outcomes.

Chapter 2

Literature Review

Transportation and logistics network configuration problems are part of the broad field of logistics. On the one hand, decisions related to transportation are usually short-term, i.e., they are made in a daily or even in an hourly basis. Hence, these are considered operational decisions and are reviewed in Section 2.1. On the other hand, decisions related to logistics network configuration are usually medium- and/or long-term, i.e., they are made monthly or yearly. Therefore, these are considered tactic and strategic decisions, and are reviewed in Section 2.2. This chapter and the whole document preserve this general division to provide a more organized classification of our research. Additionally, solution approaches of these types of problems, which go from biased-randomized (BR) heuristics to fuzzy simheuristics, are reviewed as well. Findings are shown in Section 2.3. All our published and under-review articles contribute to this chapter. Finally, it is worth to clarify that our three review articles are explicitly cited in footnotes, since they are not mentioned in the rest of the chapters of this thesis.

2.1 Transportation Problems

This section reviews the published literature about a group of subproblems included in the general topics of the vehicle routing problem (VRP), the team orienteering problem (TOP), and the rid-sharing problem (RSP). Characteristics such as deterministic or stochastic parameters (Panadero et al., 2020b), agri-food issues (Tordecilla-Madera et al., 2017), or richness (Caceres-Cruz et al., 2014) differentiate these subproblems. Additionally, given the importance of sustainability in transportation problems, a review about the electric VRP, arc routing problem (ARP), and TOP has been developed as well.

2.1.1 The Vehicle Routing Problem

The first academic mention of a VRP was made by Dantzig and Ramser (1959), as a generalization of the traveling salesman problem. More than 60 years have passed since then, in which a large set of variants has been identified. Each variant of this problem has shown its relevance both academically –given the algorithmic challenges that they pose– and economically –given their high applicability in real-world problems. A few of these variants are: heterogeneous fleet of vehicles (Eskandarpour et al., 2019; Penna et al., 2019), time-windows (Yu et al., 2019; Marinakis et al., 2019), multiple depots (Calvet et al., 2019; Li et al., 2019), multiple delivery levels (Martins et al., 2021a; Bayliss et al., 2020c), simultaneous pick-up and deliveries (Koç et al., 2020; Hornstra et al., 2020), or combinations of the former (Belgin et al., 2018; Rezaei et al., 2019; Zhou et al., 2018). A thorough study of these variants, solving methods, and applications is carried out by Toth and Vigo (2014). More recently, a concise research is showed by Sharma et al. (2018). Also, a review centered on VRP-related themes instead of traditional variants is made by Vidal et al. (2019).

The VRP is a classical NP-hard problem (Lenstra and Kan, 1981) with a vast number of applications in the transportation sector (Braekers et al., 2016). In its basic version, a set of customers has a known demand that must be satisfied by a single depot. This depot has a virtually unlimited capacity, and a fleet of homogeneous capacitated vehicles are employed to deliver the demanded units of a single product. Each customer must be visited only once. Given the limited capacity of the vehicles, several routes must be designed to meet all customers demands. Finally, once the vehicle has delivered all its assigned load, it must return to the depot. The objective is to minimize transportation costs, usually measured in terms of total traveled distance or time (Pisinger and Ropke, 2007). Figure 2.1 depicts the network topology of a basic VRP, where a fleet of three vehicles is employed to serve 12 customers from a central depot. The loaded vehicles depart from the depot, and the demanded product is delivered at each customer location.

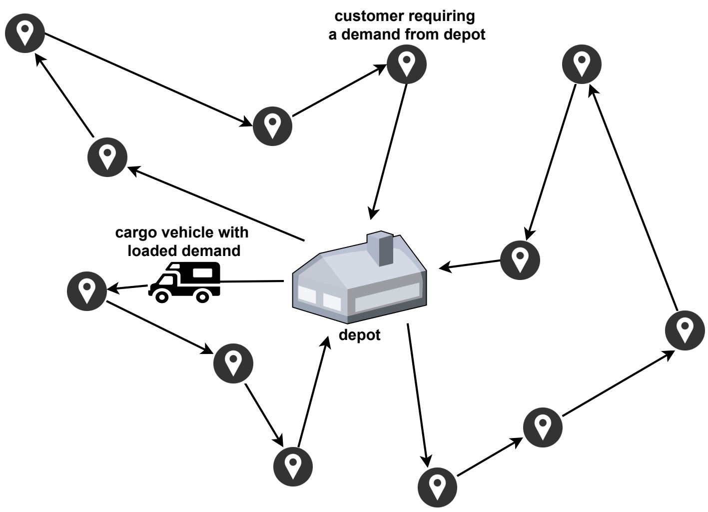


Figure 2.1: Network topology of a basic VRP.

A commonly addressed variant in the VRP literature is the VRP with backhauls (VRPB), which integrates forward logistics with reverse logistics. A highly cited survey in the topic is carried out by Parragh et al. (2008), who classify this problem in four groups. Traditionally, the main goal of the VRPB has been to minimize the total distribution and collection cost by taking advantage of the non-used capacity of the vehicles in the return trip. Some initial approaches for the VRPB are presented by Deif and Bodin (1984), Goetschalckx and Jacobs-Blecha (1989), and Jacobs-Blecha and Goetschalckx (1992). These papers present heuristic algorithms to solve the problem efficiently. In addition, the VRPB is also modeled as a mixed-integer linear problem (MILP) by Toth and Vigo (1997). They solve it through a branch-and-cut algorithm for instances between 25 and 68 customers. As pointed out by Koç and Laporte (2018) and Belloso et al. (2019), the VRPB is still offering challenges that need to be solved, such as including stochastic parameters or using hybrid solution methods. Wassan (2007) proposes a heuristic that uses reactive tabu search (TS) and adaptive memory programming for solving the VRPB. Zachariadis and Kiranoudis (2012) propose the static move descriptor strategy, which is intended to reduce the computational complexity required to examine neighborhoods with very large solutions. Brandão (2016) solves a VRPB through a deterministic version of the iterated local search (ILS) metaheuristic. Dominguez et al. (2016a) consider two-dimensional loads in a VRPB. Here, a hybrid approach that combines biased randomization with a large neighborhood search metaheuristic is proposed. In general, better solutions –in terms of cost and computational times– were found in comparison with current state-of-the-art heuristics. Belloso et al. (2019) use an iterative method based on local search (LS) and a BR process to solve the heterogeneous-fleet VRPB, obtaining 20 new best-known solutions in a set of 36 instances.

All the aforementioned papers assume that visiting all backhaul (BH) customers is mandatory. Nevertheless, selection of customers to service is also an alternative, i.e., only some BH customers are visited. In the context of the general VRP (not the VRPB), this problem is called the VRP with deliveries and selective pickups (Gribkovskaia et al., 2008). It can be classified into two groups: *(i)* deliveries and pickups are carried out alternately (Assis et al., 2013; Ting et al., 2017; Al Chami et al., 2018); and *(ii)* pickups must be carried out only after all deliveries are done, i.e., this problem shows characteristics of a VRPB. To the best of our knowledge, in this group only García-Nájera et al. (2015) address BHs as a central problem. Bruck and Iori (2017) and Gutiérrez-Jarpa et al. (2010) also tackle BHs, but only for comparative purposes with other problems.

The possibility that vehicles start and end their routes in different nodes creates a different VRP variant, which is called in the literature the *open* VRP (OVRP). From the earlier definition stated by Schrage (1981), the OVRP has been studied and enriched by many other constraints (Braekers et al., 2016). Among them, we can highlight the rich variants with multiple depots (Lahyani et al., 2019; Brandão, 2020), heterogeneous fleet of vehicles (Yousefikhoshbakht and Dolatnejad, 2017) or even a conjunction of them (Tavakkoli-Moghaddam et al., 2019; Husakou et al., 2020). Although most of these works address a single objective, studies considering multiple objectives have aimed, for instance, to reduce the total number of routes, the total travel cost, and the longest route altogether (Sánchez-Oro et al., 2020). According to Li et al. (2007), the range of applications that ends up in OVRPs is commonly found in contexts where contractors –who are not employees of the delivery company– use their vehicles and do not return to the depot, such as home delivery of packages and newspapers.

Rich VRPs have been increasingly addressed by the academic community, since they incorporate highly realistic constraints, especially when these are considered simultaneously (Azadeh and Farrokhi-Asl, 2019). Characteristics regarding input data, decision management components, types of vehicles, time constraints, among others, turns a classical VRP into a rich VRP (Lahyani et al., 2015b). For instance, Alemany et al. (2016) combine the well-known Clarke & Wright savings (CWS) heuristic (Clarke and Wright, 1964) with Monte

Carlo simulation (MCS) to solve a heterogeneous-fleet, multi-depot, multi-compartment, multi-product, and multi-trip VRP. In general, vehicles can be classified according to their physical characteristics, e.g., they can be homogeneous or heterogeneous, or compartmentalized or not. The relevance of considering compartmentalized vehicles emerges whenever different types of products are demanded and they are incompatible, i.e., products must be carried separately into the same vehicle and not be mixed. Despite the practical applications of this strategy for addressing real-world problems, the multi-compartment VRP has been scarcely studied (Derigs et al., 2011). Both theoretical and real-world cases can be found in the multi-compartment VRP literature. Silvestrin and Ritt (2017) and Muyldermans and Pang (2010) show examples of the former. These works propose metaheuristic approaches given the combinatorial nature of this problem. Regarding real-world cases, products as diverse as apparel, fuel, food, and waste require the use of compartmentalized vehicles for performing an appropriate transport (Wang et al., 2014; Reed et al., 2014; Vidović et al., 2014; Coelho and Laporte, 2015).

Agri-food supply chains represent also a field where the multi-compartment VRP has been addressed. These chains have special characteristics that should be taken into account in its modeling, such as products perishability (Tordecilla-Madera et al., 2018) or supply and demand seasonality (Vlajic et al., 2012). For instance, Lahyani et al. (2015a) propose a branch-and-cut algorithm to solve a multi-period and multi-compartment VRP with heterogeneous vehicles. A real case from the olive-oil collection process in Tunisia is considered, where compartments cleaning activities are considered. Oppen et al. (2010) address also cleaning activities in a multi-compartment VRP where inventory constraints are considered. Different types of animals are transported in this case, as well as a heterogeneous fleet and multiple trips. An exact method based on column generation is used as solving approach. Alternatively, employing approximate methods is a usual approach in agri-food multi-compartment VRPs. For instance, Caramia and Guerriero (2010) propose a hybrid approach combining mathematical programming and LS techniques to solve a real-life case regarding the collection of different types of milk in Italy. Finally, the number and capacity of compartments can also be a variable to consider, i.e., compartments are flexible. For instance, a large neighborhood search algorithm is proposed by Hübner and Ostermeier (2019) to solve this variant of the multi-compartment VRP. A relevant contribution of this paper is the consideration of loading and unloading costs, which are a function of the number of compartments.

Considering uncertain parameters in the VRP allows to solve more realistic and, therefore, more complex cases. Highly cited articles reviewing the stochastic VRP can be found in the literature (Ritzinger et al., 2016; Gendreau et al., 2016; Gendreau et al., 1996). Early works authored by Stewart Jr and Golden (1983) and Dror and Trudeau (1986) consider stochastic customer demands, i.e., its actual value is not known with certainty until the vehicle arrives to the customer location. Both mathematical models and heuristics are proposed to address the problem. More recent articles consider stochastic demands as well, such as the ones published by Jaillet et al. (2016), Marinakis et al. (2013), Juan et al. (2013b) or Juan et al. (2011). Additional studied stochastic parameters have been travel times (Guimarans et al., 2018; Taş et al., 2013), service times (Errico et al., 2016), a combination of service and travel times (Li et al., 2010; Kenyon and Morton, 2003; Laporte, 1992), vehicle speeds considering environmental issues (Çimen and Soysal, 2017), or perishability (Rahbari et al., 2019; Chen et al., 2009). The consideration of time windows is also a usual addressed characteristic (Errico et al., 2016; Li et al., 2010; Bent and Van Hentenryck, 2004). Finally, a growing approach consists in considering fuzzy parameters –instead of stochastic ones– as an alternative to model uncertainty. Articles by Zhang et al. (2020b), Shi et al. (2017), Ghannadpour et al. (2014), Zheng and Liu (2006), or Teodorović and Pavković (1996) demonstrate the suitability of this approach.

So far[1], the vast majority of cited papers in this section assume that VRP constraints are hard, i.e., they must be met mandatorily. Nevertheless, some realistic cases consider that constraints are soft, i.e., they can be violated to some extent by incurring a penalty cost. This cost is usually non-linear or non-continuous, which turns a traditional smooth objective function into a non-smooth one (Bagirov and Yearwood, 2006). Considering these characteristics increases the natural computational complexity for solving the VRP. In this context, soft constraints can include time windows or capacity constraints that are allowed to be violated to some extent (Hashimoto et al., 2006). These soft constraints also allow decision-makers to consider more realistic models that take into account different management strategies and policies. For instance, customers would accept a delayed delivery if the supplier offers a discount. Likewise, a percentage of the deliveries can be outsourced if in-house capacity is exceeded. For instance, Juan et al. (2013a) consider a capacitated version of the problem with a non-smooth and non-convex objective function and soft constraints. These authors propose a BR approach called *MIRHA*. This is a multi-start procedure consisting of two phases: a first phase in which a BR version of a constructive heuristic is designed according to a geometric probability distribution; and a second (improvement) phase in which an adaptive LS procedure is implemented. Several instances from the literature are used to test the proposed algorithm and compare it with a traditional greedy randomized adaptive search procedure (GRASP).

2.1.2 The Team Orienteering Problem

The TOP can be considered an extension of the traditional VRP, where visiting all customers is not possible due to the limitations on the fleet size and on the maximum driving range of each vehicle. Hence, customers have to be prioritized based on their location and also in the reward obtained by visiting them. Therefore, the TOP aims at maximizing the collected reward by visiting a subset of nodes, subject to constraints such as a maximum tour length, a given vehicle capacity, or a limited driving range. Figure 2.2 displays an example of a basic TOP, where black nodes represent the visited customers, and gray nodes represent the non-serviced ones. The TOP is an extension of the orienteering problem (OP), which has

[1]This paragraph is based on the review article:
Juan, A.A., Corlu, C.G., **Tordecilla, R.D.**, de la Torre, R., & Ferrer, A. (2020). On the use of biased-randomized algorithms for solving non-smooth optimization problems. *Algorithms*, 13(1), 8.

been proved to be NP-hard (Golden et al., 1987). The OP is a problem with the same characteristics as the TOP, except that the former employs a single vehicle to serve the customers, whereas the latter uses a fleet of vehicles. Hence, the TOP is a more challenging problem than the OP. Since the TOP is an NP-hard problem, the use of metaheuristics is required in order to obtain high-quality solutions in short computing times, especially when dealing with large-sized instances (Bayliss et al., 2020b). The TOP was first introduced in the literature by Chao et al. (1996). Since then, multiple variants of this problem have been studied, such as the ones with time windows (Labadie et al., 2012; Lin and Vincent, 2012; Vansteenwegen et al., 2009), soft constraints (Estrada-Moreno et al., 2020), multiple periods (Tricoire et al., 2010), precedence constraints (Hanafi et al., 2020), dynamic rewards (Reyes-Rubiano et al., 2020a), rescue operations (Saeedvand et al., 2020), use of electric vehicles (EV) (Xu et al., 2020), or use of drones in the context of smart cities (Juan et al., 2020a).

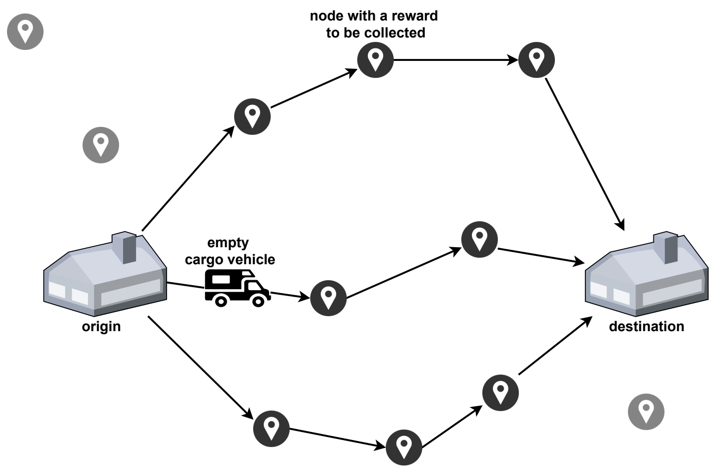


Figure 2.2: Network topology of a basic TOP.

The stochastic TOP (STOP) is another variant in which different parameters are modeled randomly. For instance, Panadero et al. (2020b) study a STOP with stochastic travel times. A simheuristic is proposed to cope with this problem. It combines a BR multi-start metaheuristic approach with MCS. According to the authors, their simheuristic approach can generate solutions that combine efficiently expected costs and variability under a stochastic environment. In a similar way, Bayliss et al. (2020b) address a STOP with dynamic rewards. In this case, the stochastic component is related to travel times, while the reward values for customers are composed of both a static and a dynamic component. The dynamic component accounts for bonuses when customers are visited earlier during a route, and penalties in case these nodes are visited at the end of the corresponding route. A simheuristic-learnheuristic approach is proposed to solve this problem, in which dynamic values are learned from simulation experiments during the search process. Mei and Zhang (2018) consider the STOP with time windows (STOPTW) in the context of tourist trip design (TTD). A set of points of interest (POI) must be selected to be visited. The visit duration of a POI is modeled as a random variable, which means that some pre-planned trips might become infeasible in practice. A genetic programming hyper-heuristic is proposed to solve this problem. The results obtained by this solution approach outperform the manually designed policies, achieving, for some cases, an average total score more than twice the total score obtained by the manually designed policies. Likewise, Karunakaran et al. (2019) address the STOPTW in TTD with a stochastic visiting time of POIs. An evolutionary multi-tasking optimization genetic programming approach is proposed. This methodology is based on island models, in which knowledge is shared through multi-tasking, thus exploiting the implicit parallelism of population-based search algorithms to simultaneously tackle multiple distinct optimization tasks.

Fuzzy techniques have been barely used in the OP and the TOP. Verma and Shukla (2015) and Ni et al. (2018) consider OPs in which both the collected rewards and the travel times are fuzzy. The former authors propose a parallel algorithm as a solving approach, whereas the latter employ a genetic algorithm (GA). Regarding the TOP, Brito et al. (2016) propose a GRASP to solve this problem considering fuzzy rewards and fuzzy travel times. A fuzzy linear program is formulated to model the addressed problem, with the objective of maximizing the total collected reward. Finally, Oliva et al. (2020) propose a fuzzy simheuristic approach to solve a TOP in which rewards offered by half of the customers are stochastic, whereas rewards of the other half are fuzzy.

2.1.3 The Ride-Sharing Problem

The RSP is a topic whose core idea is to foster that personal private vehicles are shared by a group of people, instead of being used only by the driver or car owner. Nowadays, the massive use of apps and interconnected smartphones facilitates the immediate contact between drivers and users for sharing trips. Furthermore, ride-sharing activities provide multiple benefits for drivers, users, and the entire community (Bistaffa et al., 2019; Stiglic et al., 2015), such as the reduction in costs, pollution, and traffic congestion. The basic version of the RSP (Martins et al., 2021c) consists of a finite set of capacitated homogeneous or heterogeneous vehicles, each one driven by an individual owner, who offers empty seats to users with similar itineraries. Each user requests a service, providing their current location, and drivers pick them up in these locations. This means that drivers have some kind of flexibility to adapt their routes so they visit the pickup point. Moreover, the vehicle capacity allows for more than one user to be transported. Hence, the route performed by each vehicle consists of an origin point (each driver's home), a set of locations where the driver picks up the users, and an arrival point. It is usual to assume that a driver can pick up each user only if the destination of all of them is the same. However, the destination points can be the same or different for each vehicle. Since the vehicle capacity and the number of vehicles are limited, not all users requesting a service can be picked up. Hence, the challenge is not only to design the routes, but also to select the users that will be picked up. This selection process is carried out based on both the distance between the user location and the driver's origin and destination points, as well as the fee that is paid by the user to the driver for being transported. The objective of the RSP is then to maximize the total collected fee. Figure 2.3 displays an example of a complete solution for the basic version of this problem. Connected houses represent the users who are served by the vehicles, whereas non-connected houses represent the non-served users.

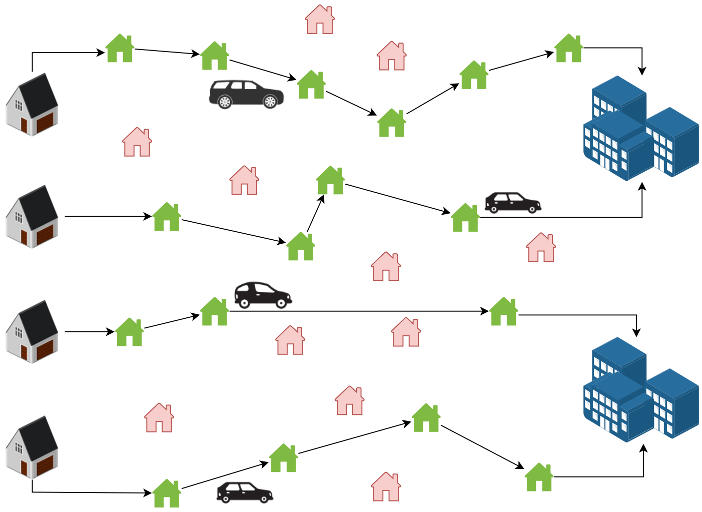


Figure 2.3: Network topology of a basic RSP.

Both Mourad et al. (2019) and Machado et al. (2018) have authored recent articles reviewing the published literature about the shared mobility topic. Whereas the former focuses on quantitative approaches, i.e., both optimization and solution methods are reviewed, the latter provides conceptual approaches, such as the contribution of this topic to alleviate current transportation problems in large cities, or an analysis of the different modalities of shared mobility. Alonso-Mora et al. (2017) propose a reactive anytime optimal algorithm to design ride-sharing services. Instances taken from the New York city taxicab public dataset are employed to test their approach. Vehicles with capacity of 1, 2, 4, and 10 passengers are tested. Service rate, waiting time, and traveled distance are considered as performance indicators. Fagnant and Kockelman (2018) address a dynamic RSP in which autonomous (fully-automated) vehicles are considered. A series of agent-based simulation runs are performed, employing a real-world network from Austin, USA. Agatz et al. (2011) solve optimally a dynamic RSP with the objective of minimizing the total traveled distance of all participants. Real-world demand data from the Atlanta Regional Commission, USA is employed. The obtained results are compared with those attained by means of a greedy algorithm. Mahmoudi and Zhou (2016) address the RSP as a VRP with pickup and delivery and time windows, which is formulated through a MILP model. A Lagrangian relaxation approach is proposed to find optimal solutions that minimize the total routing cost. Both benchmark and real-world instances –based on Chicago and Phoenix, USA networks– are employed to test their approach. Hosni et al. (2014) also propose a Lagrangian decomposition approach to solve a MILP model of the so-called shared-taxi problem, with the objective of maximizing the total profit. Furthermore, authors propose a novel heuristic based on incremental costs. A set of benchmark instances are used to compare the results obtained by each proposed algorithm.

2.1.4 Electric VRP, ARP, and TOP[2]

With the goal of promoting sustainability, many cities in the world are observing an increasing use of EVs, both for citizens' mobility (Ruggieri et al., 2021) and for last-mile logistics (Patella et al., 2021). The use of zero-emission technologies is supported by governmental plans in regions such as Europe (Hooftman et al., 2020), North America (Greene et al., 2014), and Asia (Masiero et al., 2016). According to Kapustin and Grushevenko (2020), EVs will account for a noticeable share (between 11% and 28%) of the road transportation fleet by 2040. Still, many authors point out batteries' driving range anxiety, high recharging times, scarcity of recharging stations, and lack of effective financial incentives that compensate for the higher cost of most EV models as some of the main barriers for the generalization of EVs in our cities (Juan et al., 2016; Mukherjee and Ryan, 2020; O'Neill et al., 2019).

In urban, peri-urban, and metropolitan areas, many activities related to freight transportation and citizens' mobility are carried out by fleets of vehicles (Beneicke et al., 2019). The efficient coordination of these fleets becomes necessary in order to reduce monetary costs, operation times, energy consumption, and environmental/social impacts on the city. However, this coordination constitutes a relevant challenge that is typically modeled as a mathematical optimization problem. Depending on the specific characteristics of the transportation activity, different families of problems can be found in the scientific literature. Among the most popular ones, we can include VRPs, ARPs, and TOPs. These problems, which can model scenarios involving both road and aerial EVs, are NP-hard even in their simplest versions. Thus, the use of heuristic-based algorithms (Maier et al., 2019) and simulation-based approaches (Chica et al., 2020) becomes a first-resource tool when solving rich and real-life instances, which usually contain a large number of nodes to be visited. From an operational perspective, the inclusion of driving range constraints and long recharging times constitute additional challenges that must be properly addressed when providing near-optimal transportation plans in any of the aforementioned routing problems (Figure 2.4).

2.2 Logistics Network Configurations

This section reviews the published literature about quantitative approaches regarding strategic, and strategic-tactic-operational decisions in logistics networks. Specifically, subproblems included in the general topics of the facility location problem (FLP) and the location routing problem (LRP) are reviewed. Furthermore, given the current relevance that the resilience concept has gained in the management of supply chains, a review about quantitative methods for designing supply chain networks (SCN) under uncertainty scenarios has been developed as well.

[2]This subsection adapts the Introduction of the following review article:
Martins, L.C., **Tordecilla, R.D.**, Castañeda, J., Juan, A.A., & Faulin, J. (2021). Electric vehicle routing, arc routing, and team orienteering problems in sustainable transportation. *Energies*, 14(16), 5131.

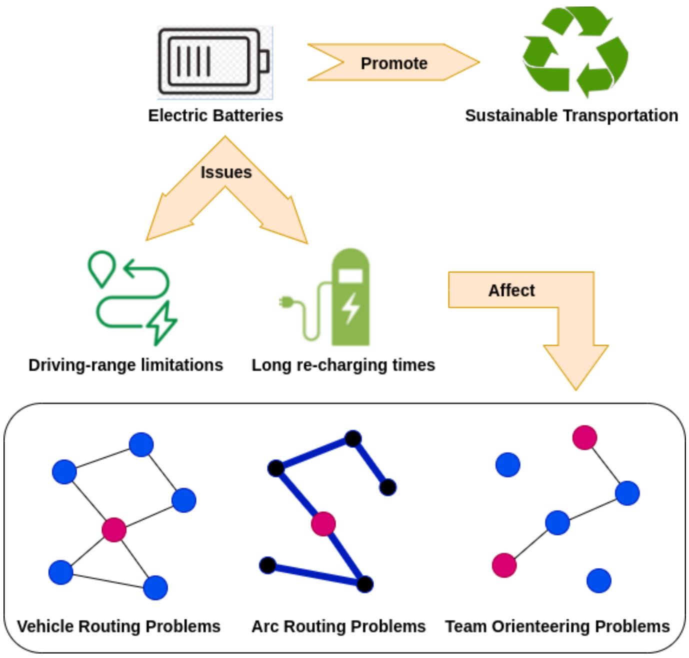


Figure 2.4: Issues of electric batteries in transportation problems.

2.2.1 The Facility Location Problem

The FLP is a classic highly studied problem consisting in locating a set of facilities –which can be production plants, warehouses, depots, etc.– with the objective of minimizing both the setup costs of these facilities and the cost incurred when serving a set of customers from there. In general, the location process can be carried out considering either a continuous space or a discrete set of potential locations (Klose and Drexl, 2005). The abbreviation FLP will be used henceforth only for the discrete version. Figure 2.5 displays an example of a final solution for a basic FLP. Customers are represented by green houses, red warehouses represent the open facilities, and black and white warehouses represent the non-open facilities. Additionally, solid lines are active connections, i.e., they indicate which facility must serve each customer. Conversely, dashed lines are inactive connections, i.e., possible connections that were not selected for the final solution. As Figure 2.5 shows, the constraints of a basic FLP are: *(i)* each customer must be served by only one facility, and *(ii)* a non-open facility must not serve any customer.

One of the most cited review articles in the FLP was published by Melo et al. (2009). This problem is studied in the contexts of supply chain management and supply chain network design (SCND). Variables such as capacity, inventory, procurement, production, routing, and transportation modes are analyzed. Taxonomies regarding the number of layers and periods, performance measures, solution approaches, and the application to real-world cases are also analyzed. A review performed by Owen and Daskin (1998) formulate a set of FLP mathematical models. Static, dynamic, deterministic, and stochastic models are shown. Klose and Drexl (2005) present also a series of mathematical models, including continuous location, multi-stage, and multi-product models, among others. Review articles considering particular FLP subproblems and characteristics have been published as well, such as

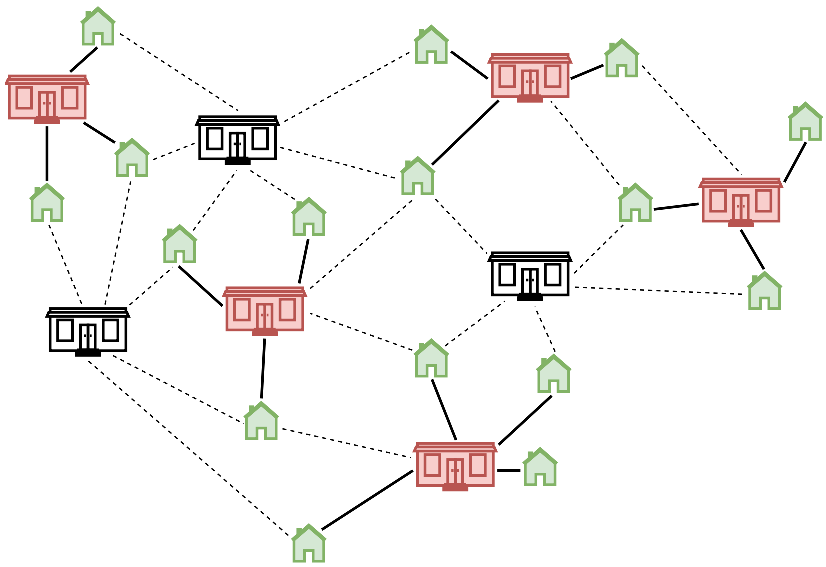


Figure 2.5: Network topology of a basic FLP.

multiple levels (Ortiz-Astorquiza et al., 2018), humanitarian logistics (Boonmee et al., 2017), healthcare (Ahmadi-Javid et al., 2017), hub location (Farahani et al., 2013; Alumur and Kara, 2008), covering problems (Farahani et al., 2012), multiple criteria (Farahani et al., 2010), or uncertainty conditions (Snyder, 2006).

Perhaps the main taxonomy in the FLP classifies the studied problems into capacitated (CFLP) and uncapacitated (UFLP). Whereas the former considers that facilities to locate have a limited capacity, the latter assumes that this capacity is virtually infinite. For instance, Pagès-Bernaus et al. (2019) address a CFLP with stochastic demands for e-commerce activities. By employing benchmark instances, authors compare the results obtained by means of a two-stage stochastic programming approach and a simheuristic. Estrada-Moreno et al. (2019a) propose a BR-ILS metaheuristic to solve a CFLP with soft constraints, and a non-smooth and non-convex objective function. A novel MILP model is formulated. The capacity of each facility may be exceeded by considering such soft constraints. In real world cases, decision-makers manage this situation by using strategies such as storing safety stocks, performing emergency deliveries, and outsourcing part of the customers' service. These strategies tend to generate additional costs that need to be considered as well during the optimization process. Regarding the UFLP, Martins et al. (2021b) propose an agile optimization algorithm based on a BR heuristic. Dynamic scenarios regarding internet of vehicles networks are considered. De Armas et al. (2017) propose both a metaheuristic and a simheuristic to solve, respectively, a deterministic and a stochastic version of the UFLP.

Snyder and Daskin (2005) consider that facilities may fail, i.e., customers demands must be met from facilities different to those initially planned. Furthermore, multiple objectives are considered: minimizing the operating cost, and minimizing the expected failure cost as a measure of reliability. A series of MILP models are formulated. A Lagrangian relaxation algorithm is proposed to solve this problem. Shen et al. (2011) address failures and reliability in a UFLP as well. A two-stage stochastic program is proposed. Multiple objectives are also considered by Amin and Zhang (2013) and Amin and Baki (2017). Both address the CFLP in the context of a closed-loop supply chain network. However, whereas the former address uncertain demands and returns by considering probabilistic scenarios, the latter considers fuzzy demands.

Growing trends in the FLP are related to the location of charging facilities for EVs and the study of humanitarian logistics. Regarding the former, for instance, Shavarani et al. (2018) focus their work in charging stations for drones. In this case, a distance capacity of each drone is also taken into account given the limited driving range provided by the electric battery. A real-world case of Amazon in San Francisco, USA is studied. Other works considering EVs in general can be found in the literature (Chen et al., 2020; Liu and Wang, 2017; Riemann et al., 2015; Jung et al., 2014). Regarding humanitarian logistics, Döyen et al. (2012) and Balcik and Beamon (2008) address a CFLP for humanitarian relief in which inventory decisions are considered, and disaster scenarios are modeled probabilistically. Additionally, Döyen et al. (2012) formulate a two-stage stochastic programming model, while Balcik and Beamon (2008) propose a MILP model with budget constraints. Additional highly recent works considering humanitarian logistics can be found in the literature (Erbeyoğlu and Bilge, 2020; Zhong et al., 2020; Oksuz and Satoglu, 2020).

2.2.2 The Location Routing Problem

The traditional LRP consists in opening one or more facilities and designing for each open facility a number of routes that meet the customers demands. Each route must start and finish at the same facility. The set of routes must serve all customers and minimize a total cost comprising the fixed and variable costs of opening facilities, the fixed costs of the used vehicles, and the transportation costs incurred when performing the routes. Figure 2.6 depicts an example of a complete LRP solution, where green houses represent the customers, red warehouses symbolize the open facilities, black and white warehouses represent the non-open facilities, and arrows symbolize the designed routes. For each open facility a set of routes starting and finishing at the corresponding facility location is designed to serve all customers demands. Main decisions in a traditional LRP are: *(i)* the number and location of facilities to open; *(ii)* the allocation of customers to open facilities; and *(iii)* the design of routes to serve customers from each facility using a fleet of vehicles. This means that the LRP considers jointly the FLP and the VRP, which increases its computational complexity.

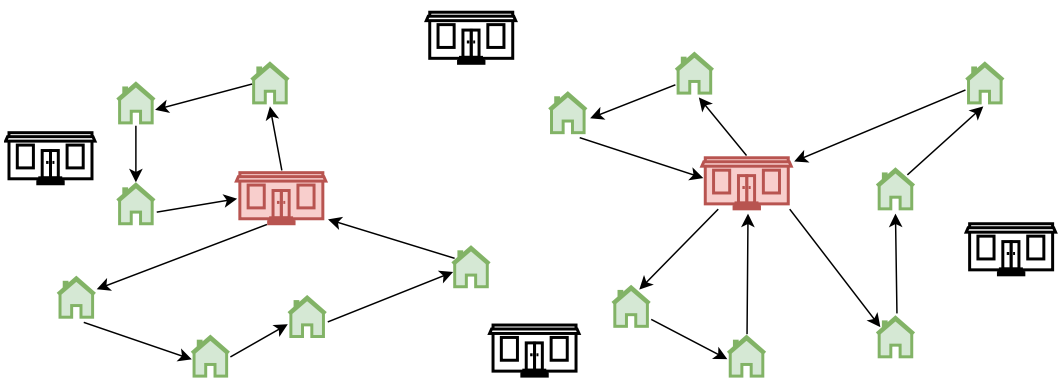


Figure 2.6: Network topology of a basic LRP.

The LRP has been broadly studied, especially in the last few decades. Literature reviews by Nagy and Salhi (2007), and more recently by Prodhon and Prins (2014) show the rise of the LRP. Traditional taxonomies, such as capacitated or uncapacitated vehicles, capacitated or uncapacitated depots, single or multiple periods, among others are tackled by these authors. A broad taxonomy is also presented by Lopes et al. (2013). Schneider and Drexl (2017) provide a review focused on solving approaches for the standard LRP, such as exact methods, matheuristics or metaheuristics. Albareda-Sambola and Rodríguez-Pereira (2019) review different mathematical formulations for the LRP, as well as heuristic algorithms and location-arc routing problems. Non-standard LRP approaches are addressed by Drexl and Schneider (2015). Papers regarding the use of fuzzy data, continuous locations, split deliveries, among other variants are reviewed.

Maranzana (1964) is perhaps the first author who combines location decisions with transportation costs. Multiple highly-cited papers were published some years later. For instance, Jacobsen and Madsen (1980) and Madsen (1983) assess three heuristics to solve an LRP for distributing newspapers. Perl and Daskin (1985) present a mixed integer program to solve a warehouse LRP. The authors propose a heuristic that decomposes the problem into three interdependent subproblems. They consider that both depots and vehicles are capacitated. The model is applied to a real distribution system in an area including Missouri, Oklahoma and Western Kansas. Theoretical problems are also addressed in this period, as well as the use of exact algorithms to solve them. For instance, Laporte et al. (1986) propose an integer linear program to solve a capacitated LRP. The capacitated part of the problem refers only to the vehicle capacity, i.e., open depots are uncapacitated. An exact algorithm applied after a constraint relaxation method is employed to solve the problem optimally. Laporte et al. (1988) study a cost-constrained LRP, where the cost of each designed route cannot exceed a known limit. Capacity-constrained and cost-constrained multi-depot VRPs are also analyzed. Finally, Laporte et al. (1989) are perhaps the first authors addressing a stochastic LRP, in which customers demands are random. A chance constraint model and a bounded penalty model are proposed and solved optimally.

Aykin (1995) addresses a hub location routing problem where hubs can interact each other. An integer program is formulated and an iterative heuristic is proposed to solve the problem. Tuzun and Burke (1999) also shows a mixed integer program, based on the work by Perl and Daskin (1985). Unlike these authors, Tuzun and Burke (1999) do not consider depots capacity. Additionally, they propose a two-phase TS algorithm as solution approach. Wu et al. (2002) consider a multi-depot LRP where vehicles are heterogeneous and the fleet of each type of vehicle is limited. A heuristic decomposition method is proposed, where the problem is divided into two subproblems. Then, each subproblem is solved through an embedded simulated annealing algorithm (SA). Prins et al. (2006) hybridize GRASP with a learning process and a path relinking to solve a capacitated LRP. A randomized version of the CWS heuristic is employed, as well as several LS procedures. Prins et al. (2007) propose a metaheuristic that decomposes the LRP into two phases: the first one solves the FLP through a Lagrangean relaxation, and the second phase employs a granular TS to solve the routing part.

Hazardous waste management has been addressed by Alumur and Kara (2007), who propose a multi-objective LRP. They formulate a mixed integer programming model to minimize both total costs and transportation risk. A real-world problem in the Central Anatolian region of Turkey is considered and solved using an exact algorithm. This problem has also been tackled by Samanlioglu (2013). In this paper, three objective functions are intended to be minimized: total costs, transportation risk, and treatment and disposal centers risk. A mixed integer programming model is formulated and solved employing an exact algorithm. A real-world problem in the Marmara region of Turkey is considered. Yu et al. (2010) employs an SA to solve a capacitated LRP. Different sets of benchmark instances are used to test the proposed heuristic. Sustainability and food perishability are addressed by Govindan et al. (2014) in a two-echelon LRP with time windows. The authors propose a metaheuristic that hybridizes a multi-objective particle swarm optimization (PSO) with an adapted multi-objective variable neighborhood search (VNS). Other fields such as supply chain network design (Lashine et al., 2006), horizontal cooperation (Quintero-Araujo et al., 2019a), city logistics (Nataraj et al., 2019), and humanitarian logistics (Ukkusuri and Yushimito, 2008) also show the application of the LRP in a deterministic context.

The LRP has been studied as well considering stochastic parameters. For instance, Quintero-Araujo et al. (2019b) propose a simheuristic algorithm to deal with demand uncertainty for the LRP. Rabbani et al. (2019) propose also a simheuristic approach to solve an LRP in the context of the hazardous waste management industry. Both generated waste and number of people at risk are stochastic. Customers demand is one of the most addressed stochastic parameters. For instance, Sun et al. (2019) address a real-world case from an express delivery company in Shanghai. Authors tackle an LRP in which demand can be split for self-pickup. Then, a simulation-based optimization model is proposed and two heuristics' results are compared. The emergence of new technologies introduces new challenges. This is the case of the work by Zhang et al. (2019b), who address the problem of locating battery swap stations and routing EVs with stochastic demands. This problem is solved employing a hybrid approach combining a VNS with a binary PSO algorithm. The problem's complexity increases when considering the low autonomy of this type of vehicles, since route failures can frequently be present when demands are not known in an accurate manner. Other parameters are also considered as uncertain. For instance, Herazo-Padilla et al. (2015) hybridize an ant colony optimization (ACO) metaheuristic with discrete-event simulation (DES) to solve an LRP in which both transportation costs and vehicle travel speeds are considered stochastic. Authors demonstrate that their proposed approach is not only efficient, but is able to find statistical interactions among the different parameters. Zhang et al. (2018) present an approach that hybridizes a GA with simulation to solve a sustainable multi-objective LRP in the context of emergency logistics. The authors consider the travel distance, the demand, and the cost of opening a depot as uncertain variables.

An alternative to study uncertainty in the LRP has been the consideration of fuzzy parameters. For instance, Zhang et al. (2020a) propose a hybrid PSO algorithm to solve a capacitated LRP with fuzzy triangular demands (CLRP-FD). The hybrid PSO algorithm is composed of three phases including an LS method and stochastic simulation. In addition, the authors propose a chance-constrained programming model for the CLRP-FD. Zarandi et al. (2011) consider a multi-depot LRP with fixed depot capacity and fuzzy travel times. Mehrjerdi and Nadizadeh (2013) present a fuzzy chance constrained programming model where demands are modeled as fuzzy numbers. A four-phase method called "greedy clustering" is proposed, in which both an ACO metaheuristic and stochastic simulation are included. Fazayeli et al. (2018) propose an LRP with time windows and fuzzy demands as the delivery part of a multimodal transport network. The mixed integer mathematical fuzzy model is coded and solved using GAMS and compared to the results provided by a GA. Nadizadeh and Kafash (2019) analyze an LRP with simultaneous pick-up and delivery in the context of reverse logistics. Both pick-up and delivery demands are fuzzy variables. A fuzzy chance constrained programming model is proposed to represent the problem, and a greedy clustering method is used to solve it.

2.2.2.1 Facility Sizing Decisions in the LRP

Allowing facility sizing decisions is a form of soft constraint (Juan et al., 2020b). The traditional LRP considers a rigid value for the maximum capacity of a facility, however, this constraint can be violated by providing multiple size alternatives and, therefore, either incurring an additional opening cost for a bigger size or saving money for a smaller size. Some real-world problems show the relevance of considering a set of available sizes to select those that fit better. Cases from different industries that employ either LRP or non-LRP approaches have considered this set. An example of the latter is shown by Tordecilla-Madera et al. (2017), who address the problem of locating a set of milk refrigeration tanks for a dairy cooperative in Colombia. Several tank sizes are found in the market, i.e., the considered problem must determine both the number and size of tanks that should be bought and their location, among other decisions. Correia and Melo (2016) state that, in applied problems, the capacity is often acquired in the market from a set of discrete sizes. Furthermore, economies of scale can be incurred when the facility size is an additional variable to model. The different available sizes are usually associated with investment activities, such as building facilities (Zhou et al., 2019), qualifying workforce (Correia and Melo, 2016), or purchasing equipment (Tordecilla-Madera et al., 2017). This means that considering facility sizing decisions is a strategy for decreasing the invested capital, if necessary, or even for reducing the operational costs by increasing the investment level.

To the best of our knowledge, only three papers consider facility sizing decisions in an LRP context: two works considering only deterministic parameters, and one work addressing uncertainty. These papers are referenced below. Hemmelmayr et al. (2017) consider a deterministic periodic LRP for collaborative recycling in hunger relief agencies. Possible depot locations belong to the same set as the customers, i.e., some customers are chosen to locate the depots there. A mixed-integer programming model is proposed, which is solved through CPLEX for small instances. Then, an adaptive large neighbourhood search heuristic is proposed to solve small and large instances. High cost savings for the agencies are attained through this approach. A variant in the problem considers that customers and depots belong to different sets of nodes. For instance, Tunalıoğlu et al. (2016) consider a multi-period LRP for collecting olive oil mill wastewater. A mixed-integer non-linear model is proposed. Then, the problem is solved through a metaheuristic named *Multiperiodic-Adaptive Large Neighbourhood Search (MP-ALNS)*. A case study in Turkey is considered. A sensitivity analysis is carried out and numerical results are showed as well as some managerial insights. Zhou et al. (2019) also consider different sets for depots and customers. They propose a hybrid approach combining a GA and an SA approach to solve a bilevel multi-sized terminal LRP with simultaneous home delivery and customer's pickup services. A real-world case in an e-commerce company in China is considered. Parcels' deliveries can be carried out between a distribution center (DC) and intermediate terminals, and between the same DC and the customers. The customers have the option of either to receive the deliveries at their homes or to pick up the parcels in a terminal. Hence, each customer's demand is computed considering the probability of selecting each alternative. This probability depends on the distance of the customer to its closest terminal. Once the demand has been calculated, this parameter is considered as deterministic.

2.2.3 SO Methods for Designing Resilient SCN under Uncertainty Scenarios[3]

An SCN is a typical example of a complex and large-scale system. Bidhandi et al. (2009) define it as a network of suppliers, manufacturing plants, warehouses, and distribution channels organized to acquire raw materials, convert these raw materials into finished products, and distribute these products among customers. Many decisions must be made in such a complex system in order to guarantee a good performance. However, the more complex a system is, the more imprecise or inexact is the information available to characterize it and, therefore, the greater the uncertainty level (Booker and Ross, 2011).

Supply chain network design is a concept broadly studied during the last decades, both from a qualitative and a quantitative perspective. Authors have referred to it by using the terms *supply chain design* and *supply chain network design*. Carvalho et al. (2012) state that a SCND problem "comprises the decisions regarding the number and location of production facilities, the amount of capacity at each facility, the assignment of each market region to one or more locations, and supplier selection for sub-assemblies, components and materials". These decisions are related to a strategic level, and must be optimized considering a long-term (usually several years) efficient operation of the supply chain as a whole (Altiparmak et al., 2006). One of the more challenging responsibilities in SCND is addressing uncertainty. Anticipating the future is crucial in planning and designing processes. However, the future conditions of the business environment is generally difficult to predict. Blackhurst et al. (2004) state that one of the causes of SCNs complexity is their dynamic nature and the

[3] This subsection adapts the Introduction of the following review article:
Tordecilla, R.D., Juan, A.A., Montoya-Torres, J., Quintero-Araujo, C., & Panadero, J. (2021). Simulation-optimization methods for designing and assessing resilient supply chain networks under uncertainty scenarios: A review. *Simulation Modelling Practice and Theory*, 106, 102166.

uncertainty in variables such as demand, capacities, transportation times, or manufacturing times.

In recent years, a trend in the literature has been the consideration of resilience for designing and assessing SCNs in order to face uncertainty. Christopher and Peck (2004) define *resilience* as "the ability of a system to return to its original state or move to a new, more desirable state after being disturbed". Similar definitions can be found in fields different to SCND, such as ecology, psychology and economy (Ponomarov and Holcomb, 2009), or natural disasters risks mitigation and adaptation in urban systems (Harrison and Williams, 2016). For instance, a concept from earthquake studies is given by Bruneau et al. (2003), who state that "seismic resilience is the ability of both physical and social systems to withstand earthquake-generated forces and demands and to cope with earthquake impacts through situation assessment, rapid response, and effective recovery strategies."

Resilient SCND has been a topic able to attract the attention of researchers, especially when trends such as leanness and globalization have increased the risks that supply chains must face. Regarding leanness, it makes SCNs more vulnerable due to the reduction or even removal of redundancies (Behzadi et al., 2017). Regarding globalization, the increasing complexity of SCNs in a globalized world causes higher uncertainty (Hohenstein et al., 2015). Moreover, globalization increases supply chain vulnerabilities (Dixit et al., 2016). Expanding globally a supply chain raises the likelihood of facing new risks that might not exist in a local range. For instance, a natural disaster such as the 2011 earthquake in Japan, which triggered a tsunami and a nuclear crisis, affected many global companies like those in the silicon wafers industry. Since 60% of silicon wafers world demand were supplied by Japan (Pariazar and Sir, 2018), this product availability decreased considerably. The same disaster affected also all Toyota factories. Although most of them were not directly affected, a two-week shutdown was caused by disruptions in the components supply, given the Toyota's lean production planning (Goldbeck et al., 2020). Human-induced disasters are also a source of disturbances for supply chains, either they are deliberate –e.g., terrorist attacks– or caused by involuntary mistakes or negligence –e.g., the 2010 oil spill in the Gulf of Mexico–, as described in Ramezankhani et al. (2018). These examples show the relevance of considering resilience aspects when designing and assessing supply chains, since they need to recover successfully after the occurrence of such disruptive events.

The terms *risk* and *vulnerability* are closely related to resilience. Carvalho et al. (2012) relate supply chain vulnerability to the incapacity of a SCN to react to disturbances. More exactly, Heckmann et al. (2015) define *supply chain vulnerability* as "the extent to which a supply chain is susceptible to a specific or unspecific risk event". Here, the *disturbance* concept is similar to the *risk* concept, being this a primary term previous to vulnerability. Peck (2006) defines *supply chain risk* as "anything that disrupts or impedes the information, material or product flows from original suppliers to the delivery of the final product to the ultimate end-user". Therefore, the more resilient a SCN, the lower its vulnerability to risks (Rajagopal et al., 2017). A review about the use of quantitative approaches in supply chain risk management is carried out by Oliveira et al. (2019). They perform a systematic literature review to analyze and synthesize the contribution of simulation and optimization methods in this field. Moreover, when risks cause a disruption in a few nodes, their effects can easily spread to other parts of the supply chain. This phenomenon is known as *the ripple effect* (Li and Zobel, 2020). According to Dolgui et al. (2018), the ripple effect causes lower revenues, delivery delays, loss of market share and reputation, as well as stock return decreases, hence affecting the global performance of the supply chain.

Epidemic outbreaks are a very special case of SCN risks characterized by a long-term disruption, disruption propagation –i.e., the ripple effect–, and high uncertainty due to simultaneous disruptions in supply, demand, and logistics infrastructure (Ivanov, 2020). Particularly in 2020, the global pandemic caused by the COVID-19 disease has largely affected all areas of the economy and society worldwide. Some supply chains have experienced an increase of demand that they are not able to satisfy (facial masks, hand sanitizer, ventilators, etc.), while others are suffering long-time production stops like the ones of non-essential products. These companies are in danger of bankruptcies and needing help from governments. As pointed out by Ivanov and Dolgui (2020), supply availability in global supply chains has been largely decreased and imbalanced with the demands. Thus, this pandemic is an unprecedented and extraordinary situation that clearly shows the need for advancing in research and practices of SCN resilience. In addition, new concepts related to resilience, such as supply chain survivability, are emerging in the literature.

In logistics and supply chain management, quantitative approaches are mainly classified into two groups: optimization and simulation, which are mostly used independently to address uncertainty, e.g., see Govindan et al. (2017) and Stefanovic et al. (2009) for each group, respectively. However, given the growth in computational power, the use of hybrid SO methods has increased in recent years (Juan et al., 2018) in order to combine the most important advantages of both worlds, mainly because of its suitability to address uncertainty (Chiadamrong and Piyathanavong, 2017). Nevertheless, in the more specific topic of SCND, applications of hybrid simulation-optimization (SO) methods are still scarce and, to the best of our knowledge, it is almost nonexistent in SCND resilience. In regard to existing review articles about this topic, most of them still address conceptual papers, which shows the relevance of carrying out a review analyzing papers following a quantitative approach.

2.3 Solution Approaches

Even the most basic versions of the aforementioned problems have been proved to be NP-hard (Nagy and Salhi, 2007). Despite the fact that some of the reviewed works propose mathematical models and solve them optimally by employing exact methods, current trends such as the Internet of things (IoT) (Lin et al., 2017), the smart cities (Faulin et al., 2018), or the use of autonomous vehicles (Bagloee et al., 2016) require to find solutions in an agile fashion, or even in real time. Although exact methods are capable of finding optimal solutions, several hours or even days can be taken to achieve them. Conversely, heuristic approaches are capable of finding very good or even near-optimal solutions in really short computing times. Hence, this section addresses different levels of heuristic approaches used to solve transportation and logistics network configuration problems. Additionally, we perform a brief review about SO approaches to provide a context to those heuristics that include uncertain parameters, such as the simheuristics and the fuzzy simheuristics.

2.3.1 Biased-Randomized Heuristics and Metaheuristics

Pure greedy constructive heuristics are algorithms that iteratively build a solution by selecting the next movement from a list of candidates. Such candidates have been sorted previously according to some criteria, such as costs, savings, profits, etc. These heuristics typically select the "most promising" candidate from the list. Since they follow a constructive logic, a good final solution is expected by the end of the procedure. Nevertheless, these algorithms are deterministic, i.e., the solution is always the same every time the heuristic is executed. This means that the exploration process is poor, which prevents the algorithm from finding better solutions unless more complex searching structures –i.e., LS and perturbation movements– are considered by investing more computing time. Examples of such heuristics are the well-known CWS heuristic for the VRP (Clarke and Wright, 1964), the nearest neighbor criterion for the traveling salesman problem (Bellmore and Nemhauser, 1968), or the shortest processing time dispatching rule for some scheduling problems (Panwalkar and Iskander, 1977).

As described by Grasas et al. (2017) and Juan et al. (2013a), using a skewed (non-uniform) probability distribution to introduce a biased-randomization behavior into the process that selects the candidates from the sorted list is an efficient way of generating better solutions. The idea is to assign a different probability to each candidate in the list, such that the more promising candidates –those at the top of the list– receive a higher probability of being selected than those below them. This randomization process leads to the generation of slightly different solutions every time the algorithm is executed if, for instance, BR heuristics are embedded into a multi-start framework (Martí et al., 2013). Hence, multiple executions of a BR heuristic –either completed in a sequential or in a parallel mode– will yield a set of alternative solutions, all of them based on the logic behind the heuristic. Since many biased-random variations of the constructive procedure defined by the heuristic are executed, chances are that some of these "near-greedy" heuristics lead to solutions that outperform the one generated by the greedy heuristic (Ferone et al., 2019).

The proposed methodology can be seen as a natural extension of the basic GRASP (Resende and Ribeiro, 2010), as analyzed by Ferone et al. (2019). Instead of employing empirical probability distributions, which require time-consuming parameter fine-tuning and thus might slow down computations, a theoretical probability distribution such as the geometric distribution or the decreasing triangular distribution can be used. Random variates from these theoretical distributions can be quickly generated by employing analytical expressions. Moreover, they tend to have less parameters, and these are typically easy to set. Application fields such as food logistics (Estrada-Moreno et al., 2019b), flow-shop scheduling (Ferone et al., 2020), or mobile cloud computing (Mazza et al., 2018) have successfully utilized geometric distributions to introduce BR processes during the selection of the candidates that are employed to construct a feasible solution. Figure 2.7 illustrates how geometric probability distributions with four different parameter values ($p \in \{0.1, 0.3, 0.6, 0.9\}$) will have a different behavior while assigning probabilities of being selected to the elements of the sorted list, during the iterative construction of a BR solution. Thus, while for $p = 0.1$ the distribution is closer to a uniform one –i.e., the probabilities are distributed among a relatively large number of top positions in the sorted list–, for $p = 0.9$ the behavior is closer to the greedy one that characterizes a classical heuristic, with the top element in the sorted list accumulating most of the chances of being the next selected element. Both extremes ($p \rightarrow 0$ and $p \rightarrow 1$) represent diversification and greediness, respectively. Usually, parameter values in the middle of both extremes are able to provide a better trade-off between these two cases, thus promoting some degree of diversification without losing the rational (domain-specific) criterion employed to sort the list.

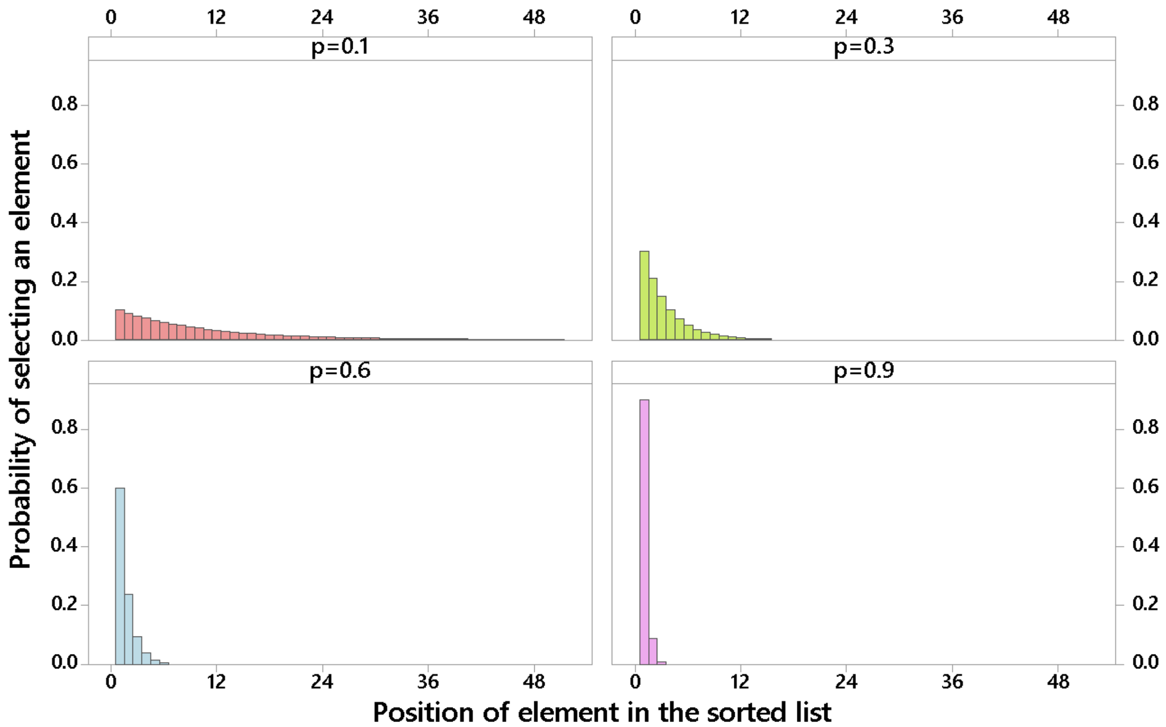


Figure 2.7: Biased-random sampling of elements from a list using a Geometric distribution.

BR heuristics have been successfully used during the last years to solve NP-hard problems from different fields, such as the VRP (Reyes-Rubiano et al., 2020b; Dominguez et al., 2016b), the LRP (Quintero-Araujo et al., 2017), the FLP (Estrada-Moreno et al., 2019a), the ARP (Gonzalez-Martin et al., 2012), permutation flow-shop scheduling (Juan et al., 2014), waste collection (Gruler et al., 2015), horizontal cooperation (Quintero-Araujo et al., 2019a), portfolio optimization (Kizys et al., 2019), marketing (Marmol et al., 2021; Marmol et al., 2020), and catastrophe insurance (Bayliss et al., 2020a). A different class of BR heuristics was introduced by Gonçalves and Resende (2011) for solving combinatorial optimization problems. Since its core is a GA, the biased random-key genetic algorithm (BRKGA) aim to bias the selection of parents for generating new solutions. Recently, this solving methodology has been developed for solving an OVRP with capacity and distance constraints (Ruiz et al., 2019). In the literature, the BRKGA has been also largely applied for solving different scheduling problems (Brandão et al., 2015; Brandão et al., 2017; Andrade et al., 2019; Homayouni et al., 2020).

2.3.2 Simulation-Optimization Approaches

Simulation and optimization are two traditional approaches of Operations Research, which have been employed extensively in an independent fashion to solve complex problems (Figueira and Almada-Lobo, 2014). Nevertheless, the great increase in computational power in the last two decades have fostered the development of hybrid methods that take the best characteristics of each approach (Juan et al., 2018). For instance, whereas simulation is a highly suitable technique for dealing with uncertainty, it is not very useful when seeking for optimal solutions. This flaw can be overcome if simulation is combined with exact or heuristic algorithms. Hybrid SO approaches have been employed to solve problems belonging to different fields, such as transportation and logistics (Juan et al., 2015a), project portfolio management (Fu et al., 2005), or healthcare (Xu et al., 2015).

Multiple review articles addressing SO approaches can be found in the literature. An early highly-cited work was published by Fu (1994), who provides a classification of SO approaches according to solved problems and solution methodologies. Both discrete and continuous state space problems are analyzed. This classification is also made by Tekin and Sabuncuoglu (2004), who additionally classify the studied articles as a part of local or global optimization. More recently, Figueira and Almada-Lobo (2014) propose a taxonomy to classify SO methods. Amaran et al. (2014) perform a review with a strong focus in solution algorithms. Methods such as ranking and selection, gradient-based procedures, sample path optimization, or metaheuristics are analyzed by Xu et al. (2015) and Fu et al. (2005), as well as applications in multiple real-world fields. Finally, a few review articles study SO methods considering special characteristics, such as multiple objectives (Yoon and Bekker, 2020), or behavioral factors (Gruler et al., 2019).

The field of SCND has also employed SO approaches as a decision-making tool. Both exact and heuristic or metaheuristic methods are employed as optimization approach. The latter are preferred in those cases in which fast solutions are required and near-optimal solutions are enough to most decision makers. In this case, GAs are usually considered (Costa-Salas et al., 2017; Ding et al., 2009). For instance, Ko et al. (2006) design and assess a distribution network through a hybrid SO model. A strategic-tactical MILP model is proposed and a GA is used as a solving approach combined with DES. Other metaheuristics found in the SCND literature are TS (Correll et al., 2014) and PSO (Salem and Haouari, 2017). Regarding exact methods, MILP is a highly implemented approach (Zhang et al., 2019a; Gumus et al., 2009). For instance, Chiadamrong and Piyathanavong (2017) propose a hybrid model that, in early stages, solve independently both the deterministic and the stochastic models for designing a SCN. Then, authors combine these models and compare results with the analytical model and a simulation-based optimization model. The solving time required by this hybrid approach is shorter than the one employed by traditional simulation-based optimization models.

Multiple uncertainty approaches are considered when using SO approaches for designing SCN, such as probability distributions (Mattos et al., 2019; Keizer et al., 2015; Yoo et al., 2010), scenarios with probabilities (Kim et al., 2011), or fuzzy sets (Ji et al., 2007). An example of the use of probability distributions is provided by Martins et al. (2017), who propose an SO approach to redesign a pharmaceutical wholesaler SCN. They affirm that literature addressing supply chain network redesign is very scarce, and that a redesign process should be carried out carefully because it is different to a design process. In the former, the company has already a market share that can be severely affected if the redesign process is not performed well. A MILP model is used for strategic and tactical decisions of redesign, and a DES model is also employed for operational decisions related to the evaluation of the impact of redesign in daily activities. These authors consider that customers demand is uncertain. Other authors who consider it the same are González-Hernández et al. (2019), Salehi et al. (2019), and Saif and Elhedhli (2016). Additional considered uncertain parameters are costs (Mattos et al., 2019; Guerrero et al., 2018), supply (Kristianto and Zhu, 2017; Ekşioğlu et al., 2013), or selling prices (Leonzio et al., 2019; Koo et al., 2008).

2.3.3 Simheuristics and Fuzzy Simheuristics

Simheuristic algorithms are a special case of SO approaches that combine metaheuristics with simulation, and has been proved to be a successful approach when dealing with combinatorial optimization problems (COP) involving probabilistic uncertainty. They can be considered a very efficient approach to deal with stochastic COPs. Such efficiency is measured both in terms of computing times and solution quality, i.e.: *(i)* simheuristics consume relatively low computing times given the inclusion of fast metaheuristics in the solution search procedure. Furthermore, promising initial solutions are evaluated by an initial simulation using a small number of runs (i.e., more intensive simulations are reserved only for a small group of elite solutions); and *(ii)* the inclusion of metaheuristics also has an impact on enhancing the solution quality. Moreover, since considering stochastic inputs implies that outputs are also stochastic, simheuristics not only assess the quality of the solution in terms of traditional indicators, such as costs or profits, but also in terms of risk and reliability values (Hatami et al., 2018).

In general, a simheuristic algorithm works as follows: *(i)* given a stochastic problem, the random variables are transformed into their deterministic counterpart by using expected values; *(ii)* an approximated framework (heuristic or metaheuristic) is used to generate high-quality solutions for the transformed deterministic instance that can also be "promising" solutions for the stochastic version of the problem; *(iii)* these promising solutions are sent to a simulation engine in order to estimate its quality in a stochastic environment. The simulation engine, in addition, provides feedback to better guide the search used by the approximated procedure; and *(iv)* an improved estimation of the quality of the solutions is obtained for a subset of "elite" solutions using a longer simulation process.

Simheuristics have been successfully employed to solve problems related to different application fields, such as flow shop scheduling (Negri et al., 2021; Villarinho et al., 2021), job shop scheduling (Caldeira and Gnanavelbabu, 2021), waste collection (Yazdani et al., 2021; Gruler et al., 2020b; Gruler et al., 2017b), hazardous waste management (Rabbani et al., 2019), facility location (Pagès-Bernaus et al., 2019), military applications (Lam et al., 2019), healthcare (Dehghanimohammadabadi et al., 2017), finance (Panadero et al., 2020a), telecommunication networks (Alvarez Fernandez et al., 2021), or disaster management (Yazdani et al., 2020). Nevertheless, simheuristics have been mainly applied to the optimization of transportation systems. Different variants can be found in the literature. For instance, Latorre-Biel et al. (2021) combine simheuristics with machine learning and Petri nets to solve a single-depot VRP with stochastic and correlated demands. The proposed algorithm is capable of forecasting both customer demands and their correlations. Stochastic demands have also been considered in the VRP with multiple depots (Calvet et al., 2019). Travel times have also been considered stochastic in the literature about VRPs. Different types of problems address this parameter, e.g., VRP in the context of the so-called omnichannel retailing mode with pick up and delivery (Martins et al., 2020), two-dimensional VRP (Guimarans et al., 2018), or EVs routing (Reyes-Rubiano et al., 2019).

A natural and realistic extension of the VRP is achieved by including inventory management in transportation decisions. For instance, Gruler et al. (2018) uses a simheuristic to solve a single-period inventory routing problem with stochastic demands. Stochastic demands have also been considered in the context of agri-food supply chains. In this case, the proposed approaches are tested by addressing a real-world case (Raba et al., 2020) or by using benchmark instances (Onggo et al., 2019). The latter work also includes perishable products. Finally, simheuristics have been used less frequently in other transportation and routing problems, such as the ARP (Keenan et al., 2021), the LRP (Quintero-Araujo et al., 2019b), or the TOP (Panadero et al., 2020b). Finally, Oliva et al. (2020) introduce the concept of "fuzzy simheuristics" to deal with the general case where both stochastic and fuzzy uncertainty is present, e.g., when the parameter(s) related to a subset of customers are stochastic, whereas the parameter(s) related to another subset of customers are fuzzy. Hence, considering all parameters of the problem as stochastic, fuzzy or deterministic are particular cases. A TOP in which customers rewards are uncertain is considered to test their proposed approach.

2.4 Conclusions

This section has performed a broad literature review about the different topics addressed in this thesis. The main works about transportation and logistics networks configurations problems, as well as solution approaches to solve these problems have been reviewed. Multiple research challenges have been identified, namely: *(i)* including rich characteristics in the VRP to represent real-world problems more accurately. Recently studied problems as the TOP and the RSP are examples of topics that enrich the classic VRP; *(ii)* solving optimization problems in real time given the rise of new technologies. For instance, the RSP is efficiently possible given the immediate connection between drivers and users driven by smartphones apps. Concepts such as internet of things, cloud computing, and smart cities are considered when studying these problems; *(iii)* employing heuristic algorithms as an efficient method for finding high-quality solutions in an agile fashion, or even in real time; *(iv)* considering sustainability, humanitarian, and resilience aspects jointly with the aforementioned challenges, given the growing concern about the transportation and logistics networks performance when assessed in terms of environmental and social impacts, as well as when natural or human-made disasters occur, e.g., earthquakes, terrorist attacks, or pandemics such as the recent COVID-19; and *(v)* considering simultaneously stochastic and fuzzy scenarios in NP-hard transportation or location problems.

Chapter 3

Applications of Biased-Randomized Heuristics

BR heuristics[1] are the "most basic" algorithms employed in this thesis, since the algorithms described in the rest of the chapters use BR as a part of a more complex structure. As described in Section 2.3.1, BR techniques allow to introduce a random behavior when constructing solutions for combinatorial optimization problems, such as the ones addressed in this work. Furthermore, if this procedure is embedded into a multi-start approach, a great number of different solutions can be generated in short computing times, which allows to broadly diversify the search for solutions. Hence, this chapter presents two real-world applications of BR heuristics. Both cases are characterized for being rich transportation problems, i.e., they have multiple realistic features that require the use of tailor-made heuristics to efficiently find good-quality solutions. The first case is an application of the rich vehicle routing problem (RVRP) and the rich team orienteering problem (RTOP) for collecting 3D-printed elements for supporting hospital logistics during the COVID-19 crisis in Barcelona. The second case is an application of the RVRP in the agri-food sector for feeding pigs in Catalonia.

3.1 The Rich VRP and the Rich TOP

The COVID-19 pandemic crisis is one of the toughest global challenges we have faced in decades. The exponential growth of cases that needed medical attention led to a sudden shortage of protective material, so that the medical and support staff were subject to higher risk to also become infected, endangering the needed level of attention in hospitals and also a faster spread of COVID-19. By March 2020, the pandemic had a strong impact in countries

[1] The contents of this chapter are based on the following works:

- **Tordecilla, R.D.**, Martins, L.C., Saiz, M., Copado-Mendez, P.J., Panadero, J., & Juan, A.A. (2021). Agile computational intelligence for supporting hospital logistics during the COVID-19 crisis. In: *Computational Management*. Springer, pp. 383–407.
- **Tordecilla, R.D.**, Copado, P., Panadero, J., Martins, L., & Juan, A.A. (2021). An agile and reactive biased-randomized heuristic for an agri-food rich vehicle routing problem. *Transportation Research Procedia*, 58, 385-392.
- Raba, D., **Tordecilla, R.D.**, Copado, P., Juan, A., & Mount, D. (2021). A digital twin for decision making on livestock feeding. *INFORMS Journal on Applied Analytics (Interfaces)*, 0, 1-16.

like Italy and Spain. As it happened in other regions, in the metropolitan area of Barcelona a community of volunteers, the so-called "Coronavirus Makers" community, arose with the aim to supply protective material to the staff working in the hospitals, nursing homes, and emergency medical attention. The main tool used were home 3D-printers, which helped to iterate the design very fast in order to reach, within a few days, the design level that was considered acceptable by the staff in charge of guaranteeing safety and quality of the produced items. It soon became noticeable that the bottleneck was the logistic side of this endeavor, due to the fact that the lockdown situation meant that each 3D-printer was located at each individual home, and route planning needed technological support, in order not to expose the drivers to more risk than strictly necessary.

This section shows the experience of matching the needs of a hospital logistics real-world case with the academic knowledge regarding the rich versions of the VRP and the TOP. In this case, it was needed to find a fast way of applying the knowledge gathered within years' of research to an urgent need, where every day counts. The target was to support the Makers community (with each maker located in his/her individual home) on their voluntary initiative to supply the sanitary staff with as much protective material as possible, and with a limited time to avoid unnecessary exposure for volunteer drivers. The main contributions of this section are: *(i)* to describe a real-life case in which computational intelligence was used to support hospital logistics during the COVID-19 pandemic crisis; *(ii)* to illustrate how real-life logistics might be rich in the sense that they combine multiple routing problems with dynamic characteristics and constrains, which might vary even from day to day; *(iii)* to provide an example of how 'agile' optimization can be applied –in combination with other technologies– to support decision making in scenarios under stress; and *(iv)* to discuss how to develop new agile-optimization tools that can efficiently cope with the aforementioned scenarios.

3.1.1 Problem Definition

The Makers community was born to contribute with creative capacity to offer a service to the healthcare system, the geriatric staff, and home-support personnel. This was a 100% altruistic and non-profitable initiative. The aim was to alleviate the need for additional protective material in hospitals and health centers derived from the scarcity of resources due to the unprecedented level of demand worldwide generated by the COVID-19 outbreak. The initiative was conceived in less than 48 hours between March 12th and 14th, and grew at an average rate of $1,000$ new volunteers per day during the first two weeks. The Makers community from Barcelona and the surrounding provinces adhered soon to this initiative, and the community was already handing out material to the hospitals on March 16th, 2020. Figure 3.1 provides an example of the problem magnitude in the area of Barcelona. As we can notice, several pickup and delivery points are geographically distributed, being the coordination of both loading and unloading activities the next challenge to face. This was the first sign that the logistics were becoming a bottleneck and further help was needed.

The items to be collected are generated by a group of Makers located at their respective homes. These homes are connected by edges, which represent streets in cities. We consider

Figure 3.1: Pickup locations in the area of Barcelona.

two types of items: face shields and ear savers. The location of each Maker home is known and identified by a coordinated pair *(latitude, longitude)*. Items should be picked up by a set of vehicles, which are driven by a group of volunteers. Each volunteer driver departs from a common origin point, collects the items according to the planned route, and brings them to a given hospital or healthcare center. The coordinates of both the origin and the destination locations are also known. Each house can only be visited by just one vehicle. The number of vehicles is given in advance. These vehicles are considered as virtually unlimited in capacity, since the size of the items to be transported is small. At the end of each day, Makers inform about the exact quantity that each of them offers for being picked up on the next morning. This imposes a hard constraint on the computational time that can be employed by the algorithm to solve a new instance of the problem every day, since drivers must have their routing plans available at first time in the morning.

In general, this hospital logistics problem shows characteristics of several rich variants of the VRP. Nevertheless, we address all daily challenges as variants of two big groups: *(i)* a rich VRP, in which all customers (Makers' houses) are visited and the objective is to minimize the total time requested in completing the routes; and *(ii)* a rich TOP, in which a time limit must be met and, hence, not all customers can be visited, since the number of drivers and pick-up vehicles is also limited. Therefore, some customers are not visited, seeking a maximum reward while satisfying the constraints. Notice that, since the problem characteristics change every day, the challenge we face was typically not a pure VRP or a pure TOP, but an RVRP and an RTOP. A maximum time to complete each single route is considered in the RTOP version. The idea is that each driver should not be working for more than a certain number of hours per day, in order to reduce the risk of exposure to the virus and also to avoid legal issues during the lockdown. Figure 3.2 displays a simple example of a complete solution for our problem in its RTOP version, where some collection points are skipped.

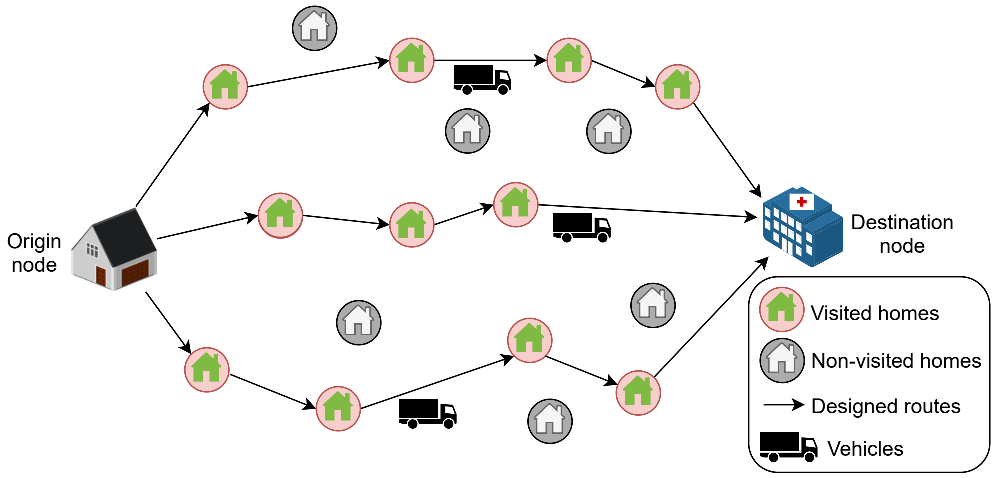


Figure 3.2: Representation of a complete TOP solution.

Our addressed problems are notably dynamic since they must be frequently modified, even on a daily basis. These daily challenges are determined by limited resources and a variety of operational decisions, such as:

1. *A maximum tour length for drivers:* as they are volunteers (and also to avoid excessive exposition to risk), the total time that drivers can dedicate to pick up elements is limited. Routes must be as balanced as possible, so that the work time of each driver is similar. Additionally, this total length must include a service time per visited node, which is assumed to be constant.

2. *A limited number of vehicles:* the number of volunteers are variable each day, which limits the quantity of routes that can be designed. This condition, jointly with the limit in the drivers' work time, makes that a few nodes must be skipped some days. These instances are then solved preferentially using a TOP-like algorithm.

3. *Origin and arrival nodes are the same or different:* most instances require an OVRP-like solution, in which drivers depart from a point that is different to the final destination. However, sometimes this constraint can be relaxed.

4. *Mandatory nodes to visit:* the three previous constraints rely on a TOP-like algorithm, resulting in locations that are not visited. However, there are some cases in which specific nodes must be mandatorily visited. These mandatory nodes to visit are more likely to be the preparation nodes, the medical centers, or some makers who offer a large number of medical supplies.

5. *Segmentation of nodes:* drivers are more willing to visit the makers and medical centers depending on the geographical zone where they are located. Hence, a segmentation process is required, in which nodes are grouped in clusters. Besides, some instances include multiple origin or arrival depots, and each cluster must contain only one pair origin-arrival. Whenever these conditions show, the solving process is semi-automatic in order to create the clusters properly.

6. *Precedence constraints:* sometimes it is mandatory to pass through a specific node to pick up supplies before making any delivery at the medical centers, e.g., to visit a preparation node before a hospital. Hence, this situation imposes a mandatory precedence in a specific group of nodes.

7. *Pickups and deliveries:* despite most nodes are pickup points, the loaded freight must be unloaded in somewhere, either into an intermediate location during the routing operation, or, more commonly, at the end of this process. Therefore, routes are frequently characterized by both operations.

As we can notice, all this dynamism, in terms of daily-based problem conceptualization, enlarges the problems' complexity. Therefore, our solving methodologies must be flexible enough to deal with them smartly, quickly, and efficiently.

3.1.2 Solution Approach

Our multi-start approach relies on multiple executions of a BR heuristic designed to solve the RVRP and the RTOP. The basic heuristic for the former is shown in Algorithm 1. It works as follows: firstly, a dummy solution is generated (line 1), which is composed of one route per location (house). For each route, a vehicle departs from the origin depot, visits the location, and then travels to the destination depot. The second stage regards the computation of savings that are associated with each edge (i, j) connecting two different locations (line 2). These savings are computed following the Equation (3.1), where t_{ij} is the time required to travel between the collection points i and j, and 0 and n are the origin and destination nodes, respectively. Later, the savings list (SL) is sorted in descending order of savings value. Next, based on the sorted SL, a route-merging process starts. The edge with the highest savings, i.e., that one at the top of the sorted list, is selected in each iteration (line 4). By using the selected edge, its two corresponding routes are merged into a new one (line 7). A few conditions (Clarke and Wright, 1964) must be validated (line 8) and, if they are met, the solution is updated (line 9). The selected edge is later removed from the SL, and this process is repeated either until the list is empty or until the number of routes in the solution equals the number n of available vehicles.

$$s_{ij} = t_{in} + t_{0j} - t_{ij} \tag{3.1}$$

This heuristic is later extended into a probabilistic algorithm by introducing a BR behavior, which smooths the original greedy performance of the heuristic. Biased-randomization techniques employ skewed probability distributions to induce an oriented (non-uniform) random behavior into deterministic procedures, consequently transforming them into randomized algorithms while preserving the logic behind the original greedy heuristics. For doing so, we employ a geometric probability distribution with a single parameter β ($0 < \beta < 1$), which controls the relative level of greediness present in the randomized behavior of the algorithm. This strategy replaces the greedy selection of the next element from the SL, thus facilitating the generation of multiple alternative solutions. Therefore, our BR heuristic

Algorithm 1 Example of our heuristic approach for the RVRP case

```
sol ← generateDummySolution(Inputs)
savingsList ← computeSortedSavingsList(Inputs)
while (savingsList is not empty or the number of routes in sol is greater than n) do
    edge ← selectNextEdge(savingsList, β)
    iRoute ← getStartingRoute(edge)
    jRoute ← getClosingRoute(edge)
    newRoute ← mergeRoutes(iRoute, jRoute)
    if (isMergeValid) then
        sol ← updateSolution(newRoute, iRoute, jRoute, sol)
    end if
    deleteEdgeFromSavingsList(edge)
end while
return sol
```

is embedded into a multi-start framework (Martí et al., 2013), which computes several solutions until a maximum number of iterations or execution time is achieved. The best solution is returned at the end of the process. After a fine-tuning process, $\beta = 0.3$ has presented a good performance, being this value selected to be used in our computations.

Algorithm 2 shows the heuristic employed for solving the RTOP case. It is similar to Algorithm 1, except for the following steps. Firstly, when generating the dummy solution, in case that any route cannot be performed within the maximum driving time, its respective location is discarded from the problem, since visiting this location is not possible. Secondly, savings are calculated following the Equation (3.2) (Panadero et al., 2020b), where u_i and u_j represent the rewards obtained for visiting the collection points i and j, respectively. By integrating the travel time and the reward in Equation (3.2), the savings are able to reflect not only the objective of minimizing travel times, but also the aim for increasing the number of collected goods. Thirdly, the merging process is carried out as far as the new route does not violate the driving-range constraint (line 9). The final list of generated routes is sorted according to the total collected reward (line 15). Finally, the first n routes of this list are selected (line 16).

$$s_{ij} = \alpha(t_{in} + t_{0j} - t_{ij}) + (1 - \alpha)(u_i + u_j) \tag{3.2}$$

3.1.3 Computational Experiments and Results

Taking into account that there were not service during most weekends –specially as the urgency for the new material was lower after the first weeks–, a total of 29 instances (days) were solved during this COVID-19 crisis. Table 3.1 displays both the known characteristics of each instance (input columns) and obtained results (output columns). Additionally, the instance name is shown, which corresponds to the date for which the instance was solved. Notice that the service time decreases to 4 minutes from the instance *apr-04*. Initially, coordinators estimated a constant service time per node of 7 minutes, however, drivers suggested a shorter time given the experience acquired in previous days. In general, the origin node

Algorithm 2 Example of our heuristic approach for the RTOP case

```
sol ← generateDummySolution(Inputs)
savingsList ← computeSortedSavingsList(Inputs)
while (savingsList is not empty or the number of routes in sol is greater than n) do
    edge ← selectNextEdge(savingsList, β)
    iRoute ← getStartingRoute(edge)
    jRoute ← getClosingRoute(edge)
    newRoute ← mergeRoutes(iRoute, jRoute)
    timeNewRoute ← calcRouteTravelTime(newRoute)
    isMergeValid ← validateMergeDrivingConsts(timeNewRoute, drivingRange)
    if (isMergeValid) then
        sol ← updateSolution(newRoute, iRoute, jRoute, sol)
    end if
    deleteEdgeFromSavingsList(edge)
end while
sortRoutesByReward(sol)
deleteRoutesByReward(sol, maxVehicles)
return sol
```

is not the same as the arrival node. Nevertheless, some instances allow to relax this constraint. They are marked with an asterisk in Table 3.1. Strictly speaking, these instances correspond to an OVRP. However, since the arrival node must also be visited before starting the route, i.e., there are *mandatory nodes* and *precedence constraints*, it was possible to adjust these instances in a pre-processing step, so that a VRP-like problem was solved.

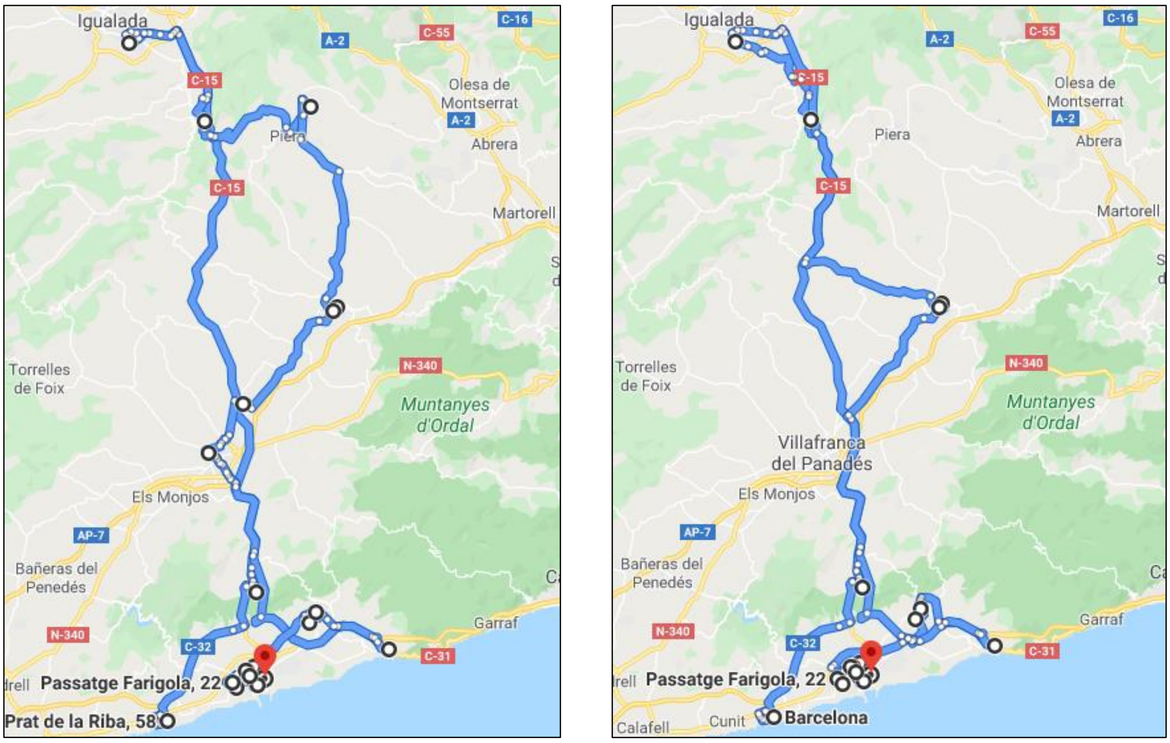
The output columns in Table 3.1 show the maximum tour length (MTL) and the number of visited nodes according to the results obtained by each algorithm. As the VRP algorithm is designed to visit always all points, the number of nodes in the corresponding column represents the total input nodes in the instance. Conversely, the number yielded by the TOP algorithm is less than or equal to the total nodes. Skipping nodes is necessary in some instances given the limitations in both tour lengths and available vehicles. For example, the instance *apr-13b* imposes that the single available vehicle must not take more than 6 hours in completing its tour. The VRP algorithm yields a total travel time of 6 hours and 34 minutes to visit 24 nodes, which violates such constraint. Alternatively, the TOP algorithm designs a 21-node route that takes 5 hours and 54 minutes. Therefore, the TOP algorithm is the best strategy to solve this instance.

The hardness of the travel time constraint depends on the problem instance, since drivers are not the same every day. Hence, the travel time is a soft constraint in instances *apr-04* and *apr-17*. Anyway, the TOP algorithm is the best strategy in these 2 instances since the time yielded by the VRP algorithm is prohibitively high. The rest of the instances have the total travel time as a hard constraint. For most of them the VRP algorithm is the best strategy, because it yields a shorter time than the TOP algorithm, guaranteeing complete routes visiting all nodes. Figure 3.3 shows an example of the best routes obtained by VRP and TOP algorithms for the instance *apr-08*. Three nodes are skipped in the second case to meet the time constraint of 5 hours, which generates savings of 35 minutes with respect to the first case. This example shows the advantages of using a TOP algorithm when the time is limited, since it finds a good balance between the reward offered by each node and the

Table 3.1: Instances' inputs and outputs.

Instance	Input							Output				
	Number of clusters	Maximum tour length (h)	Service time (min)	Number of vehicles	Mandatory nodes	Precedence constraints	Same origin-arrival node	VRP algorithm		TOP algorithm		Best strategy
								MTL(hh:mm)	Visited nodes	MTL(hh:mm)	Visited nodes	
mar-25	1	5	7	6				3:50	95	3:52	95	VRP
mar-26	1	5	7	4				4:38	77	4:59	77	VRP
mar-27	2	5	7	6				3:36	62	3:43	62	VRP
mar-28	1	5	7	2				4:59	48	5:09	48	VRP
mar-30	1	5	7	2	✓			4:01	32	4:15	32	VRP
mar-31	2	7	7	2				6:32	53	6:36	53	VRP
apr-01	2	7	7	2	✓	✓		6:39	31	6:41	31	VRP
apr-02	2	5	7	2				4:58	29	5:02	29	VRP
apr-03	1	5	7	2				4:05	29	4:14	29	VRP
apr-04	1	4	4	1	✓			5:56	22	4:07	14	TOP
apr-06a	2	6	4	2				5:58	46	6:07	46	VRP
apr-06b	1	5	4	1	✓			3:12	15	3:17	15	VRP
apr-07	1	5	4	1	✓			4:35	17	4:36	17	VRP
apr-08	1	5	4	1	✓			5:28	19	4:53	16	TOP
apr-09a	2	6	4	2				6:15	51	5:54	49	TOP
apr-09b	1	6	4	1	✓	✓	*	5:43	22	5:48	22	VRP
apr-10	1	6	4	1	✓	✓	*	5:21	25	5:31	25	VRP
apr-13a	2	5	4	2				4:59	39	5:05	39	VRP
apr-13b	1	6	4	1				6:34	24	5:54	21	TOP
apr-14	1	6	4	1	✓	✓	*	5:30	21	5:33	21	VRP
apr-15	1	7	4	1	✓	✓	*	8:38	39	6:32	28	TOP
apr-17	1	6	4	1	✓	✓	*	6:53	33	6:22	29	TOP
apr-18	2	5	4	2				4:55	38	4:54	38	TOP
apr-20	1	5	4	1	✓	✓	*	5:32	21	4:48	17	TOP
apr-22	1	5	4	1	✓	✓	*	6:33	21	4:46	16	TOP
apr-25	1	5	4	1	✓	✓	*	5:37	23	4:59	20	TOP
apr-28	1	5	4	1	✓	✓	*	6:06	23	4:54	19	TOP
apr-30	1	6	4	1	✓	✓	*	5:34	14	5:40	14	VRP
may-02	1	5	4	1				3:30	13	3:36	13	VRP

cost of service.



(a) (b)

Figure 3.3: Routes generated for the instance *apr-08* by VRP(a) and TOP(b) algorithms.

3.2 The Agri-Food Rich VRP

Feeding pigs in the pork production industry is a highly relevant activity to achieve successfully the supply chain goals (Rodríguez et al., 2014). Such activity requires a precise logistics from the production plant to the farms where the pigs are raised. Hence, this work consists in designing a set of vehicle routes that meet the feed demand of a set of pig farms, considering the real case of a pork production company in Spain. From an academic point of view, the analyzed problem can be considered as an RVRP (Caceres-Cruz et al., 2014), since: *(i)* vehicles are heterogeneous and have multiple compartments to separate different types of incompatible products that must be distributed to a set of farms; *(ii)* each farm may require multiple products; *(iii)* some farms admit only that a small-medium vehicle deliver the feed; *(iv)* a visit priority must be met, which indicates that some farms must be visited as soon as possible, whereas other farms must be the last to be served; and *(v)* the cost function considers a set of flat tariffs, which depend on both the location of the farm and the number of farms visited in the same route. A flexible and enriched heuristic is then proposed to address this problem. Apart from the multi-product and multi-compartment RVRP, this heuristic must be able to deal with an objective function that relies on a flat-rate policy instead of the traditional distance-based minimization. Then, this enriched savings-based heuristic is extended into a BR algorithm, which is able to provide multiple solution configurations in short computational times. As described in Grasas et al. (2017), BR techniques are based on the introduction of an oriented (non-uniform) randomization process inside the constructive stage of a given heuristic. By doing so, a deterministic heuristic is transformed into a randomized algorithm that can be run multiple times (either in sequential or in parallel) without losing the logic behind the heuristic. Hence, the main contributions of this section are: *(i)* the consideration of a flat-rate cost function, together with multi-product and multi-compartment characteristics; *(ii)* the design of a flexible and agile heuristic, which enriches the traditional savings heuristic, to solve a rich and real-life problem in the agri-food distribution industry; *(iii)* the extension of the former heuristic into a BR algorithm capable of providing, in short computational times, a set of alternative solution configurations to the problem, each of these including different dimensions; and *(iv)* the introduction of a reactive (automatic) fine-tuning process for the main parameter of the biased-randomization process.

3.2.1 Problem Definition

The part of the supply chain addressed in this section is that in charge of distributing the animal food from central depots to the farms, as displayed in Figure 3.4. We consider each day as an independent instance, where the subset of farms requiring service can be different. Each farm generates an order and each order may be composed of different types of feed, e.g., Figure 3.4 displays circles, hexagons and triangles representing three different products. In general, products can be classified in medicated and non-medicated. Also, the characteristics of each type of product depend on the growth stage of each herd, i.e., the required diet mix is different according to the age (in weeks) of each individual. The demand of each product in each farm is deterministic. The feed distribution is carried out from a depot through a set of compartmentalized heterogeneous vehicles. For instance, Figure 3.4 shows two types of vehicles with three and four compartments, respectively. Compartments are also heterogeneous, i.e., each compartment has a different known capacity. The demanded quantity per product and farm is at most the capacity of a vehicle. Hence, each vehicle can visit multiple farms in the same route, as long as the aggregate demand does not exceed the vehicle's capacity. Split deliveries are not allowed, i.e., a single farm must be served by a single vehicle. The objective of using compartmentalized vehicles is to separate each type of feed, since they cannot be mixed during a trip. In addition, if the demand of a product is higher than the capacity of a single compartment, it can be split into two or more compartments in the same vehicle. Nevertheless, in general, medicated feed cannot be transported in the same route as non-medicated feed. Not all types of vehicles can visit all customers, since some farms have access constraints. That is, a subset of farms can be served by all types of vehicles, whereas another subset cannot be served by large vehicles. An additional constraint assigns a sanitary priority indicator, which determines a specific order in which a subset of farms must be visited in case they are in the same route. The company classifies the farms into 3 types according to this sanitary priority: *(i)* a subset of farms with an assigned priority according to a consecutive natural number. These farms must always be served in ascending order whenever they are in the same route, e.g., a farm with a priority of 2 must always be visited before a farm with a priority of 5; *(ii)* a subset of farms with no priority; and *(iii)* a subset of farms with a "negative" priority, which indicates that they must be the last to be served in any route.

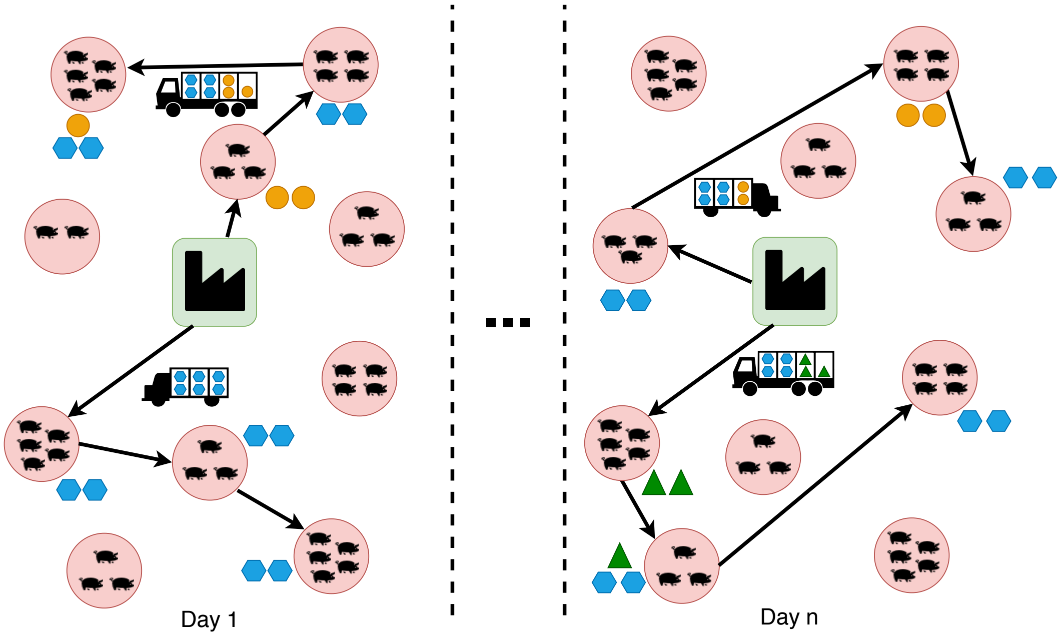


Figure 3.4: Representation of our real-life problem.

Our main objective is to minimize the total distribution cost. As the company outsources the feed transportation, the distribution cost calculation has been settled in a distribution agreement. This cost is computed as the product of the delivered quantity and a pre-established tariff. The whole distribution region is clustered in zones, so that the tariff $c(n,z)$ depends on both the zone z where the customer is located and the number of farms n visited in the same route. Each customer has three different tariffs according to n (Equation 3.3), where $c_1(z) < c_2(z) < c_3(z)$.

$$c(n,z) = \begin{cases} c_1(z), & \text{if } n = 1 \\ c_2(z), & \text{if } n = 2 \\ c_3(z), & \text{if } n \geq 3 \end{cases} \tag{3.3}$$

Figure 3.5 displays a few examples of tariffs (expressed in €/t) employed by the company. Figure 3.5*a* shows the case in which each farm is the only one visited in its route. Hence, the tariff of all customers in the Zone 1 is $c_1(1) = 7.74$ and the tariff of the customer 4, located in the Zone 2, is $c_1(2) = 8.98$. Figure 3.5*b* shows the case in which all customers in the Zone 1 form a single route, therefore, the employed tariff is $c_3(1) = 8.76$. The customer 4's tariff remains the same as in the former case. Finally, Figure 3.5*c* shows the case in which customers of different zones form a unique route. Under these circumstances, the distribution agreement indicates that the employed tariff must be the greatest one. Hence, as $c_3(1) = 8.76$ and $c_3(2) = 9.24$, the final distribution tariff for the route in this instance is 9.24 €/t. Since the total satisfied demand is the same in the 3 cases of Figure 3.5, and the total variable cost depends on the supplied food-load in tonnes, the case in Figure 3.5*b* incurs a higher variable cost than the instance in Figure 3.5*a*, and the case in Figure 3.5*c* incurs the highest variable cost in the example. This means that merging routes increases the variable cost in our problem, which is the opposite of merging routes in traditional routing problems. This behavior is caused by the flat tariffs indicated in the distribution agreement.

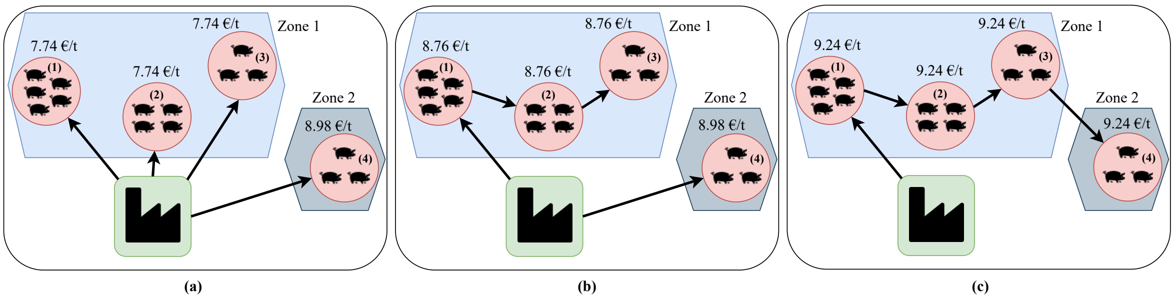


Figure 3.5: Examples of tariffs used by the company.

The considered problem requires that the total delivery cost is not the only key performance indicator (KPI), i.e., the approach used to solve this problem must show enough flexibility to consider additional KPIs, such as the number of designed routes and the total traveled distance. Despite its non-typical objective function and unique constraints, the problem can be classified as a rich variant of a multi-product and multi-compartment open VRP. Hence, it is an NP-hard problem and, as such, the use of heuristic-based approaches (Londoño et al., 2020) is justified whenever the size of the problem goes beyond a certain level.

3.2.2 Solution Approach

This section shows our approach for dealing with the described AFRVRP. This approach is based on both multi-start (Martí et al., 2013) and BR algorithms (Grasas et al., 2017). Algorithm 3 provides a general view of the proposed heuristic to solve the AFRVRP. The core of our approach is a flexible and fast two-stage heuristic, which includes all problem characteristics considering multiple KPIs. In the stage 1, a first initial solution is generated, in which each customer is assigned to a vehicle in a single round-trip, meeting all the considered constraints. Once this initial solution is generated, the algorithm merges routes in stage 2 as much as possible, reducing the number of used vehicles. Algorithm 4 outlines the stage 2, which consists of the following steps: firstly, it computes the *savings* associated with potential route merges. These savings are computed for every edge and are based on both the distance between farms and the tariff per zone. Then, a list of edges associated with the savings values is created and sorted in decreasing order. The main loop iterates on the sorted SL, where each edge is selected to be part of the solution only if it meets the following merging conditions: *(i)* both customers in the origin and the end of the edge belong to different routes; and *(ii)* these customers are adjacent to the depot. Unlike the traditional savings method, we do not consider the total vehicle capacity. Instead, it is evaluated whether the demand of each product fits in the available compartments, considering both their capacity and a feasible layout. When a feasible assignment is found, the algorithm merges the routes and updates the solution; otherwise, the current edge is rejected and the algorithm proceeds to the next iteration with a new alternative. The current solution is updated by removing the routes at both extremes of the selected edge and adding the resulting new merged route. All KPIs are then updated, including the cost, which considers the flat-rate delivery tariffs (Figure 3.5). Again, notice that this approach is different to the distance-based cost computation employed in most articles on the VRP, which do not consider a flat-rate tariff. Finally, the current edge is removed from the list, and the whole process is repeated until the SL is empty, returning a complete new solution *sol*.

Algorithm 3 Multi-Start R-BR

$sol \leftarrow$ Stage1($inputParameters$)
$\beta_1, \beta_2 \leftarrow T(0, 0.5, 1)$
$newsol_1 \leftarrow$ Stage2(sol, β_1)
$newsol_2 \leftarrow$ Stage2(sol, β_2)
$sol, m^* \leftarrow$ best($newsol_1, newsol_2$), best(β_1, β_2)
while time **not** reaches the limit **do**
    $\beta_s \leftarrow T(0, m^*, 1)$
    $newsol \leftarrow$ Stage1($inputParameters$)
    $newsol \leftarrow$ Stage2($newsol, \beta_s$)
    $sol, m^* \leftarrow$ best($sol, newsol$), best(m^*, β_s)
    if $sol \notin S^*$ **then**
        $S^* \leftarrow$ add(S^*, sol)
    end if
end while
return S^*

Algorithm 4 Stage2

$savings \leftarrow$ computeSavingsSorted(sol)
while $savings \neq \emptyset$ **do**
    $edge \leftarrow$ selectNextArc($savings, \beta$)
    $savings \leftarrow$ remove($savings, edge$)
    if isMergePossible($edge$) **then**
        $sol \leftarrow$ updateSolution($sol, edge$)
    end if
end while
return sol

The previous heuristic is extended into a reactive biased-randomized (R-BR) algorithm. This procedure allows not only to diversify the search for good solutions, but also to generate alternative solutions assessed in terms of multiple KPIs. Our proposed methodology in Algorithm 3 uses both stages 1 and 2 (Algorithm 4) as the base for the R-BR algorithm. Previously-described steps are followed the same, except for the selection of the next edge in the SL. This selection is now performed by considering a skewed probability distribution, which introduces a sort of randomness into this process. In our case, the selection of the next element is performed according to a geometric distribution with parameter $0 < \beta < 1$. Employing this distribution introduces diversification to explore other regions of the solution space, preserving at the same time the savings heuristic original purpose. Unlike previous works, our algorithm is *reactive*, since the parameter β is automatically fine-tuned. The R-BR implementation procedure is described next: firstly, initialize parameters β_1 and β_2 using a symmetric Triangular probability distribution with mode $m = 0.5$. Secondly, generate two complete solutions using β_1 and β_2, respectively. Then, compare the yielded costs (or any other KPI) to obtain the best-found mode m^* and the best-found solution *sol* so far. Then, the algorithm iterates while the time limit is not reached. For each iteration, a new β_s is computed using a Triangular distribution with mode equal to m^*. Later, generate a new complete solution *newsol* using β_s. Again, obtain the best-found mode m^* and solution *sol*. Finally, introduce the new solution *sol* in the pool of solutions S^*.

3.2.3 Computational Experiments and Results

Real-world instances representing multiple products demands from 44 workdays have been provided by the company. They represent daily deliveries made to 214 farms. Currently, the company performs a delivery only when the customer generates an order. Hence, only a subset of farms is served each day. Furthermore, the delivered product mix also changes every day, and each customer may require multiple types of food at the same day. The feed shelf life is greater than one day, therefore, perishability is not included in our case study. The number of vehicle types are 3: a vehicle type with 6 compartments and a total capacity of 26 t, a vehicle type with 6 compartments and a total capacity of 21 t, and a vehicle type with 5 compartments and a total capacity of 21 t. A single product demand can vary between 1 t and 26 t. Our approach yields 4 KPIs: *(i)* total distance, computed as an approximation by employing the Euclidean distance between two farms, considering their real Cartesian coordinates; *(ii)* total cost, computed employing the flat tariffs described in Section 3.2.1; *(iii)* total number of routes; and *(iv)* average utilization of vehicles, computed considering the utilization percentage of every vehicle used in every route of a complete solution. The algorithm is implemented in Python 3 and executed in a personal computer with 16 GB RAM and a 2.8 GHz Intel Core *i*7-1165*G*7 processor.

Our algorithm yields 4 solutions per run, where each solution is the best one according to each KPI. Table 3.2 shows the average results obtained by our greedy heuristic in real time, i.e., one single non-random run is performed in this case. Each number is calculated as an average of the values yielded by our 44 instances. Firstly, the real-life results obtained by the real company are displayed. Then we show the 4 solutions yielded by our heuristic, assessed in terms of each KPI. For instance, the *Best-distance* solution is the one that achieves the minimum distance. Hence, the reached value of the KPI *Distance* is underlined for this solution. The reasoning in this example can be extended for the rest of the KPIs. The KPIs distance, cost, and the number of routes are better when they are smaller, and the utilization is better when it is greater. Additionally, the columns of gaps show the average percentage difference between our solution and the company's. All gaps are better when they are lower. A negative gap indicates that we outperformed the company's results, i.e., our agile greedy heuristic can find both smaller distance and costs than the company. Regardless of the type of solution, the cost gap is always negative. This difference is not less than -1% given the use of flat-rate tariffs. Hence, the cost improves when the other KPIs get worse, and vice versa, which indicates that the solution to be selected by the company for their daily routes depends on the KPI to optimize. Finally, the average number of routes is the same for both *Best-#routes* and *Best-utilization* solutions. However, the distance and utilization KPIs are worse in the *Best-#routes* solution, and the improvement in cost is very slight. Therefore, we can assume that the *Best-utilization* solution always outperforms the *Best-#routes* solution.

Table 3.2: Average results considering different KPIs obtained in real time.

Type of solution	KPI				Gap			
	Distance	Cost	#Routes	Utilization	Distance	Cost	#Routes	Utilization
Real company	1153.6	5555.5	23.9	95.8%				
Best-distance	1124.7	5544.0	25.0	91.4%	-2.6%	-0.2%	4.9%	4.4%
Best-cost	1153.3	5512.4	25.4	90.3%	-0.1%	-0.8%	6.6%	5.5%
Best-#routes	1207.9	5531.4	24.8	92.4%	4.6%	-0.4%	3.9%	3.4%
Best-utilization	1186.2	5534.5	24.8	92.6%	2.5%	-0.4%	3.9%	3.1%

Table 3.3 shows the average results after running our BR algorithm for the same 44 instances after 60 seconds of run time. This table compares the results obtained when considering a non-reactive and a reactive BR heuristic. The latter refers to the procedure described in Section 3.2.2. The former refers to the case already described in the literature, in which the parameter β of the geometric probability distribution must be fine-tuned by hand. In our experiments, our manual fine-tuning process found the best results when β follows a uniform probability distribution between 0.01 and 0.40. Table 3.3 also shows the results obtained by the company in its real daily operations. Obviously, these results are the same as in Table 3.2 and independent of our both BR procedures. In general, our results in Table 3.3 outperform those in Table 3.2, which is more evident if we observe the underlined gaps. Values obtained by the non-reactive BR are only slightly better than the ones yielded by the reactive BR, i.e., differences are minimal. Nevertheless, the non-reactive BR requires a few work hours for performing the fine-tuning process, whereas the reactive BR is automatic and does not require any fine-tuning.

Table 3.3: Average results considering different KPIs after 60 seconds of run time.

Type of solution	Non-reactive BR				Reactive BR			
	KPI				KPI			
	Distance	Cost	#Routes	Utilization	Distance	Cost	#Routes	Utilization
Real company	1153.6	5555.5	23.9	95.8%	1153.6	5555.5	23.9	95.8%
Best-distance	1104.0	5541.7	24.7	92.5%	1106.6	5540.6	24.8	92.3%
Best-cost	1201.3	5495.7	26.7	86.2%	1196.9	5497.5	26.8	86.1%
Best-#routes	1178.8	5544.3	24.2	94.1%	1173.3	5542.4	24.3	93.8%
Best-utilization	1168.5	5549.6	24.2	94.8%	1174.7	5548.6	24.3	94.6%
	Gap				Gap			
Best-distance	-4.4%	-0.2%	3.5%	3.3%	-4.1%	-0.3%	3.7%	3.5%
Best-cost	4.3%	-1.1%	12.3%	9.6%	4.0%	-1.1%	12.6%	9.6%
Best-#routes	2.0%	-0.2%	1.4%	1.7%	1.5%	-0.2%	1.7%	2.0%
Best-utilization	1.1%	-0.1%	1.4%	1.0%	1.7%	-0.1%	1.7%	1.2%

The average percentage difference between our solution and the company solution is shown in the columns *Gap* of Table 3.3. This indicator is computed considering the gap between each KPI obtained for each instance. A negative gap indicates that our solution outperforms the company's. If the gap is positive, then the smaller the gap, the better. Hence, a few results can be highlighted. Firstly, our heuristic always reaches a smaller cost than the company, regardless of the type of solution. Secondly, savings in distance provided by our heuristic are high when considering the *Best-distance* solution. Thirdly, the company slightly outperforms our algorithm when considering the number of routes and the vehicle utilization. Finally, the cost is a KPI whose behavior is opposite to the rest of the indicators', i.e., when the cost improves, the other KPIs worsen. This behavior is a result of considering the flat tariffs explained in Section 3.2.1.

The best-found distance and best-found cost gaps between our solution and the company solution for the 44 instances are displayed in Figure 3.6. This figure also shows a comparison between our both tested heuristics, i.e., the non-reactive BR (NR-BR) and the reactive BR (R-BR). Regarding the distance, only a few instances exceed the 0% limit, i.e., our agile approach is able to outperform the company's distance results for the vast majority of instances. Furthermore, our approach always reaches a negative gap in costs, which is a great result considering the tough restriction imposed by the flat tariffs. Finally, Figure 3.6 also shows that our reactive BR is able to yield solutions highly similar to the ones achieved by the non-reactive BR.

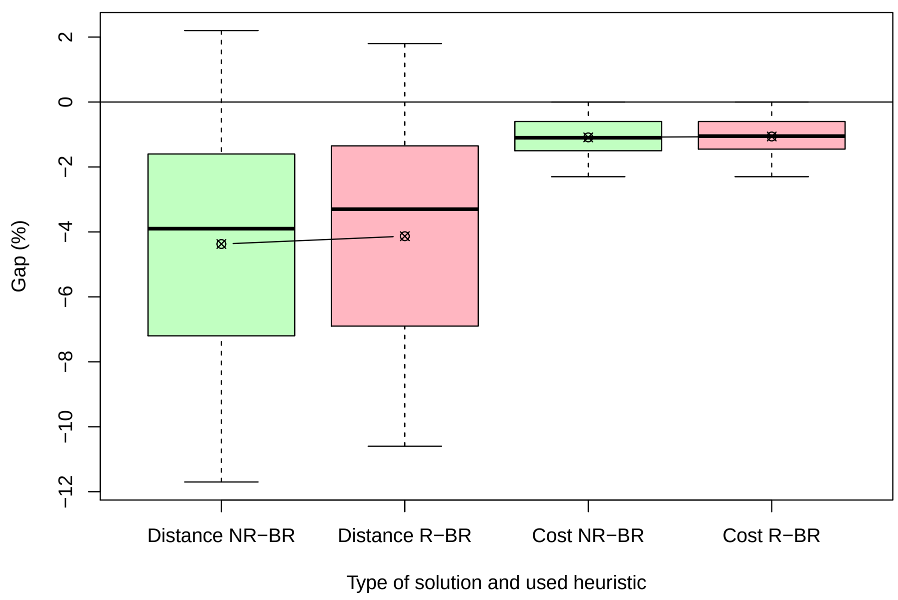


Figure 3.6: Distance and cost gaps of our best-found solutions with respect to the company's.

3.3 Conclusions

This chapter has presented two real-world cases where BR heuristics have been proved to be an efficient approach to find good-quality solutions in short computing times. Both cases show a series of rich characteristics that make already complex transportation problems even more challenging. The described problems require fundamentally flexible and fast solution approaches, since they must easily adapt to changing constraints and yield solutions that are satisfactory for decision makers. Our developed heuristics have met successfully all these conditions. In the first case, we describe a case study that deals with complex logistics challenges during the COVID-19 pandemic. It is developed in the metropolitan area of Barcelona (Spain) during March, April, and May 2020, where the pandemic were at his high and hospitals did not have enough sanitary material as to protect their nurses, doctors, and other staff. Under those critical circumstances, a self-organized community of "Makers" was able to use 3D printers at their homes to generate thousands of face shields, ear savers, and similar sanitary items. The challenge of collecting these items from hundreds of individual houses and using a limited fleet of vehicles and a threshold time per service was huge, not only due to the size of the collection-routing problem but also mainly to the fact that the problem was evolving day after day. Thus, while some days the problem was more similar to an RVRP, other days it needed to be modeled as an RTOP.

In order to cope with this optimization challenge, which basically consisted in a different problem every day that needed to be solved in a few minutes, some of the VRP and the TOP heuristic algorithms were adapted. The adaptation consisted in transforming 'heavy' algorithms into flexible and agile ones capable to adapt themselves –with little or no extra effort on our side– to the new characteristics of the problem, which were changing every day. The experience has shown us that in crisis scenarios like the one described in this chapter, a more 'agile' optimization paradigm is requested, in contrast to the use of complex algorithms that focus on the solving of a single optimization problem, more flexible and fast algorithms are needed.

In the second case shown in this chapter, a reactive BR heuristic to solve a real-world AFRVRP for distributing animal food is proposed. A set of complex constraints have been considered, such as multi-compartment heterogeneous vehicles, flat tariffs, visit priorities, among others. Four KPIs have been proposed to assess the solutions quality. Advantages of employing our agile approach are mainly twofold. Firstly, our yielded results outperform the real company's outcomes in terms of traveled distance and distribution cost. These results are obtained in only a few seconds, whereas designing these routes by the company takes a few work hours. Secondly, results yielded by our reactive BR algorithm are highly competitive when compared with a non-reactive one. However, the latter requires a time-costly fine-tuning process, whereas our proposed heuristic is parameter-less and, therefore, it does not require to perform this procedure.

Chapter 4

Applications of Metaheuristics

This chapter[1] extends the concept of BR heuristics by including more refined procedures to both diversify and intensify the search for high-quality solutions in transportation and logistics problems. Although a BR heuristic embedded into a multi-start approach can be understood as a metaheuristic by itself, this chapter shows applications where BR heuristics are employed as a part of a procedure that is more complex than the ones explained in Chapter 3. Two metaheuristics are employed in this chapter: the novel discrete-event driven metaheuristic –which has been used only by one previous work (Fikar et al., 2016)– is applied to a dynamic ride-sharing problem (DRSP) in Section 4.1; and the iterated local search (ILS) (Lourenço et al., 2019), which is applied to a VRP with optional backhauls (VRPOB) in Section 4.2, and to an LRP with facility sizing decisions in Section 4.3.

4.1 The Dynamic Ride-Sharing Problem

In today's modern society, urban centers are facing the so-called booming of information. Due to the population growth in many countries around the globe, and recent innovations in information and telecommunication technologies, several activities and related challenges have jointly arisen. People are increasingly consuming more information through their mobile devices, vehicles are equipped with different intelligent systems, devices are distributed around the cities for gathering and generating information, and urban areas are continuously taking advantage of these information technologies and big data. Consequently, the so-called smart cities have emerged, whose scope combines sustainable development with the intelligent management of gathered data in order to enhance the operation of different services within urban areas, such as waste collection management (Gruler et al., 2017b), car-sharing/ride-sharing activities (Martins et al., 2021c), optimal location of recharging stations for electric vehicles (Frade et al., 2011), among others. In this matter, during the past

[1]The contents of this chapter are based on the following works:

- Peyman, M., Copado, P.J., **Tordecilla, R.D.**, Martins, L.C., Xhafa, F., & Juan, A.A. (2021). Edge computing and IoT analytics for agile optimization in intelligent transportation systems. *Energies*, 14(19), 6309.
- Londoño, J.C., **Tordecilla, R.D.**, Martins, L.C., & Juan, A.A. (2020). A biased-randomized iterated local search for the vehicle routing problem with optional backhauls. *TOP*, 29, 387-416.
- **Tordecilla, R.D.**, Montoya-Torres, J.R., Quintero-Araujo, C.L., Panadero, J., & Juan, A.A. (2022). The location routing problem with facility sizing decisions. *International Transactions in Operational Research*, 0, 1-31.

few years, the Internet of things (IoT) has become a popular term that plays a significant role to expand and produce a lot of data through sensors and allows citizens and things to be connected in any situation or with anyone (Beneicke et al., 2019). Also, fog and cloud computing come to support IoT to manage the large amount of generated data (Aazam and Huh, 2014).

Integrating IoT and open data initiatives in smart cities allows governments, public, and private sectors to develop new services and applications by ensuring the effective handling and managing of data that are constantly shared among individual citizens and different industries (Ahlgren et al., 2016). For instance, sensing real-time traffic flow and mobility tracking data, such as vehicle states (e.g., location, speed, etc.), intersection information (e.g., the length of the queue waiting at the intersection, etc.), and the situation of the road (e.g., under repair, traffic accidents, etc.) can be analyzed (Tang et al., 2019; Aazam and Huh, 2015) to explain the dynamics of urban vehicles as micro and macroscopic simulations, traffic flow, and travel time estimations (Jiménez-Meza et al., 2013).

In this context, mobile internet technology is one of the actors which enables dynamic and on-demand sharing activities. In ride-sharing systems, for example, people are allowed to offer trips for riders by using their own private vehicles, i.e., its core idea is to foster that personal private vehicles are shared by a group of people, instead of being used only by the car owner. Nowadays, the massive use of apps and interconnected smartphones facilitates the immediate contact between drivers and users for sharing trips. Furthermore, ride-sharing activities provide multiple benefits for drivers, users, and the entire community (Bistaffa et al., 2019), such as reduction in costs, pollution, and traffic congestion. Hence, utilizing the cloud and edge computing helps to handle terabytes of data extracted from IoT devices, including information about vehicles mobility and traffic conditions. Also, by analyzing these data and combining it with the concept of ride-sharing, some mentioned urban problems can be reduced or even solved. In this context, optimization techniques, such as approximate methods –i.e., heuristics and metaheuristics– have been proved to be both efficient and capable of generating high-quality solutions for large-scale and complex real-world problems (Grasas et al., 2017). This means that heuristics have a high potential to provide agility and real-time responses, which are necessary issues for a good performance of intelligent transportation systems and, in general, of this type of systems. Nevertheless, after reviewing some related work, very few articles combining heuristics with IoT analytics by utilizing the cloud and edge computing have been found.

Hence, in order to fill this gap, a DRSP is addressed in this section, where dynamic conditions usually encountered in modern urban centers affect decision-making processes. In other words, the DRSP considers dynamic traffic conditions that might lead to several changes on the initially designed routes due to the incorporation of updated information, such as traffic conditions and vehicles states. In this problem, a set of routes must be designed so that the total reward collected by picking up passengers is maximized. A discrete-event driven metaheuristic is proposed to solve this problem. This solution method is enhanced with BR techniques to provide an efficient exploration of the solution space. Hence, our main goal is to propose a methodology for solving the DRSP.

4.1.1 Problem Definition

The DRSP is an enriched version of the static ride-sharing problem (RSP), which allows to update the routes as new events such as traffic conditions, new service requests, or service cancellations can change the originally designed routes after the vehicles have already departed from their origins. Hence, the static version of the RSP (Martins et al., 2021c) consists of a finite set of capacitated heterogeneous vehicles, each one driven by an individual owner, who offers empty seats to users with similar itineraries. Each user requests a service, providing their current location, and drivers pick them up in these locations. This means that drivers have some kind of flexibility to adapt their routes so they visit the pickup point. Moreover, the vehicle capacity allows for more than one user to be transported. Hence, the route performed by each vehicle consists of an origin point –e.g.,each driver's home–, a set of locations where the driver picks the users up, and an arrival point. We assume that a driver can pick up each user only if the destination of all of them is the same. However, the destination points can be the same or different for each vehicle. Since the vehicle capacity and the number of vehicles are limited, not all users requesting a service can be picked up. Hence, the challenge is not only to design the routes, but also to select the users that will be picked up. This selection process is carried out based on both the distance between the user location and the driver's origin and destination points, as well as the fee that is paid by the user to the driver for being transported. The objective of the RSP is then to maximize the total collected fee. Figure 2.3 displays an example of a complete solution for the static version of the addressed problem. Connected houses represent the users who are served by the vehicles, whereas non-connected houses represent the non-served users.

The static version of the RSP assumes that, once all routes have been designed, they cannot be further modified. Nevertheless, real-world events, such as traffic conditions, new orders, or cancellations, may lead to make changes in the original route plans. Since these events occur frequently when vehicles are already in route, a DRSP allows for the design of new routes, which include only those users who have not yet been picked up. Such redesign process is performed in discrete time intervals. Formally speaking, the DRSP can be defined on a directed graph $G(N, E)$, where N is the set of nodes, and E is the set of edges linking these nodes, i.e., $E \subseteq N \times N = \{(i, j) \mid i \in N, j \in N, i \neq j\}$. Three subsets of nodes are considered, such that $N = I \cup O \cup A$. I is the subset of nodes where the users are located, O is the subset of drivers'/vehicles' origin nodes, and A is the subset of final destination nodes. Each pickup point $i \in I$ has a known fee f_i, which is paid by each user for being transported. Traversing each edge $(i, j) \in E$ has a deterministic cost c_{ij}. Routes are performed by a set K of vehicles. The capacity b_k of each vehicle $k \in K$ is known as well. Each pickup point $i \in I$ must be visited only once, and each vehicle $k \in K$ is assigned to only one route. Only a subset of nodes $J \subset I$ can be visited. Each route starts in an origin node $o \in O$, traverses a subset of nodes $H \subset J$, and finishes in a destination node $a \in A$. If d_h is the demand of each node $h \in H$, then $\sum_{h \in H} d_h \leq b_k$. If t is the time interval set to recalculate the routes, then this process is always performed in the time $\tau = nt$, where $n \in \mathbb{N}$. Such recalculation is performed iteratively until all vehicles have arrived to their respective destinations. Furthermore, if L_n is the subset of non-visited nodes in the period

n, such that $L_n \subseteq J$, then the routes recalculation is performed only for the nodes in both L_n and A. This assumption is helpful to respect the commitment of serving the originally selected customers, taken in the period $n = 0$. We assume that routes are only affected by traffic conditions, i.e., new orders and cancellations are not allowed in our addressed version of the problem. Hence, our problem consists in designing a set of $|K|$ routes that meet the aforementioned constraints, such that the total collected fee is maximized.

4.1.2 Solution Approach

In this section, we describe the proposed methodology for solving the DRSP. This methodology is based on a discrete-event driven metaheuristic (Fikar et al., 2016), which generates promising solutions according to events that occur over time. These events are related to circumstances in which the system has to react appropriately, i.e., changes in traffic conditions involve re-planning routes. Discrete-event constructive heuristics are based on the use of discrete-event (time-based) simulation (DES) to handle time dependencies that arise as the solutions are constructed. In our case, the basic idea is to complete a DES of arrival, departures, and re-planning of routes, so that vehicles can re-plan routes according to the traffic conditions provided by the open data server to minimize the travel time towards the final destination. This re-planning procedure is done at each time interval t. Hence, any event can belong to one of the following types: vehicle delivery, vehicle arrival, and traffic update. Each event is associated with a vehicle and a trigger time. Likewise, the vehicle is assigned to a current trip (between two customers) and the current route that is covered.

The flowchart of our solving approach is presented in Figure 4.1. At the beginning (period $n = 0$), the algorithm produces an initial static planning without considering any traffic conditions, i.e., all the information employed during this stage is not modified. In addition, the list of events is initialized adding one departure event for each available vehicle, and a traffic update event. Departure events are programmed to occur at the period $n = 0$, while the traffic data event arises in the period $n = 1$. Each period n lasts t time units, which is set as an input parameter. The main loop iterates over a list of events that occur during the execution of the routes until this list is empty. At each iteration, the algorithm takes the first event and proceeds according to the event type. In the case of a departure event, the vehicle located at the customer departs to the next destination of the current trip, hence, a new arrival event is scheduled, considering the travel time and the current traffic congestion. If the event is an arrival event, two possibilities could be given: the vehicle arrives either to a customer or to the final destination. In the case of the former, the vehicle must stop for performing a pick-up action. Then, the vehicle capacity availability is updated with the passenger demand. In addition, the algorithm creates a new vehicle departure event from this customer. In the case of the latter, the vehicle arrives to the final destination and, thus, no further action is required. Finally, for the case of the traffic update event, which occurs in each period n, the algorithm re-plans the vehicles' routes, considering the current traffic data, from the next stop to the final destination.

Algorithm 5 outlines the approach for solving the DRSP in a given period n. This approach is a two-stage heuristic algorithm capable of providing a good trade-off between

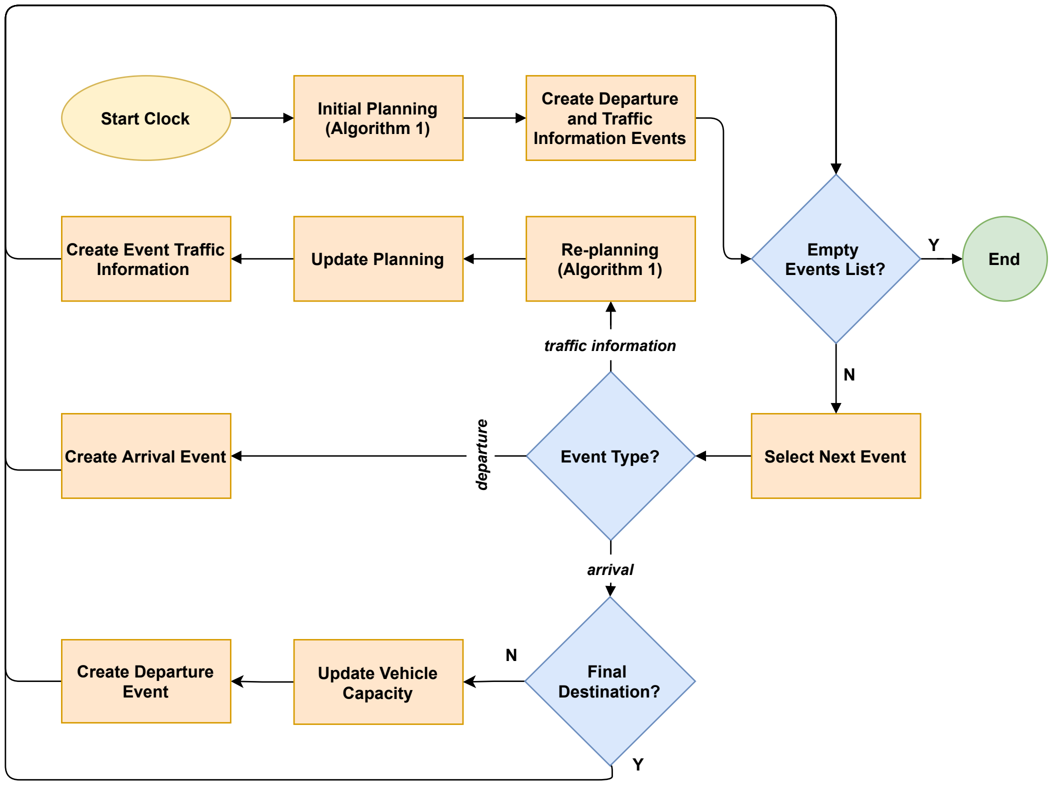


Figure 4.1: Discrete-event algorithm flowchart.

solution quality and required computational effort. In the first stage, the original problem is divided into small sub-problems (clusters) according to origin points and destination points. Notice that different sub-problems might share some of the pick-up locations. In the second stage, each cluster is solved by applying a savings-based heuristic proposed by Panadero et al. (2020b). This heuristic involves the following steps: *(i)* generation of a dummy solution where each pick-up point (node) is connected by one route (vehicle) with the origin and the final destination, and *(ii)* construction of the enriched savings list (SL) of edges, where each savings value is related to an edge that links whichever location such as origin, destination, and pick-up point. This enriched savings value is computed according to Equation 4.1. The input parameter α is set within $(0, 1)$, c_{ij} denotes the traveling time between i and j. Likewise, 0 and m are the origin and destination nodes, respectively. Finally, μ_i and μ_j are the assigned fees at each node. These savings values consider both the traveling time and the aggregated fee collected by visiting both locations i and j. The main loop iterates while the list is not empty. For each iteration, the edge at the top of the list is chosen, then, the associated routes are merged if and only if the resulting route does not exceed the vehicle capacity, otherwise, the edge is rejected.

$$s_{ij} = \alpha(c_{im} + c_{0j} - c_{ij}) + (1 - \alpha)(\mu_i + \mu_j) \tag{4.1}$$

This heuristic is deterministic because the merging process always selects the first element of the SL. We extend this heuristic introducing a BR process in order to produce a variety of solutions without losing the logic behind the original heuristic. The geometric

Algorithm 5 Two-stage approach algorithm for solving a static RSP

```
clusters ← computeClusters(customers, vehicles)
for cluster_k in clusters do
    sol_k ← dummySolution(customers(cluster_k))
    savings ← genSavingsList(customers(cluster_k),α)
    while savings ≠ ∅ do
        edge_ij ← pick(savings, β)
        r_i, r_j ← getRoutes(edge_ij)
        if checkMerging(r_i, r_j,vehicle(cluster_k)) then
            route_i ← merge(route_i,route_j)
            sol_k ← replace(sol_k,route_i)
        end if
        savings ← remove(savings, edge_ij)
    end while
    sol ← add(sol, sol_k)
end for
return sol
```

probability distribution, Geom(β) with $\beta \in (0, 1)$, is employed to introduce a BR behavior. In Algorithm 6, the BR heuristic is executed for a maximum number of iterations or computational time, resulting in a multi-start approach (Martí et al., 2013). Therefore, several feasible and promising solutions are generated, and the one with the highest collected fee is returned.

Algorithm 6 Multi-Start approach for solving the RSP

```
bestSol ← Heuristic(customers, vehicles)
while end not reached do
    sol ← BR(customers, vehicles, β, α)
    if fee(sol) > fee(bestSol) then
        bestSol ← sol
    end if
end while
return bestSol
```

4.1.3 Computational Experiments and Results

A series of numerical experiments have been designed to test our approach. 27 instances with different characteristics have been tested. The instance name in Table 4.1 sets both the number of customers requesting for a service and the number of available vehicles. For example, *drsp63x6-1* is the first instance in the list considering 63 potential customers and 6 vehicles. Hence, three groups of instances with different sizes are tested. Each potential customer's demand and location have been generated randomly. Available vehicles are heterogeneous in each instance, with capacities varying between 4 and 8 users. The aggregated capacity of all vehicles is proportional to the total demand. Only for experimental purposes, the traffic conditions have also been generated randomly for each edge $(i, j) \in E$ in each period n. These conditions are represented by a coefficient w^n_{ij}, which was generated according to a uniform probability distribution, such that $w^n_{ij} \sim U(0, 1)$. For real-world cases, w^n_{ij} can be computed after retrieving the corresponding traffic information from open data repositories. w_{ij}^n affects the cost c_{ij} incurred when traversing this edge. For instance, if c_{ij}^0 is the time spent for going from customer i to customer j in the period 0, i.e., before the vehicles' departure, then the simulated real time for traversing the edge (i, j) in the period n, affected by the traffic conditions, is computed according to Equation (4.2). Finally, the time interval that triggers the re-computation of routing plans is set in $t = 10$ time units. Notice that the number of times that routes are re-computed is instance-dependent, since these new computations are performed iteratively until all vehicles have arrived to their corresponding final destinations. The algorithm has been implemented in Python 3.8. The experiments were carried out on an i7-8750 CPU at 2.20 GHz with 16 GB of RAM memory installed. The time limit for the biased-randomization process was set in 1 second (in order to keep the real-time condition).

$$c_{ij}^n = c_{ij}^0(1 + w_{ij}^n) \tag{4.2}$$

Table 4.1: Obtained results by our algorithm for a DRSP.

Instance	Served customers	Total collected fee	OBS cost	OBD cost	Gap
drsp43x4-1	17	237	595.80	384.14	-35.53%
drsp43x4-2	18	295	587.88	492.58	-16.21%
drsp43x4-3	20	373	512.51	398.23	-22.30%
drsp43x4-4	22	394	502.89	402.97	-19.87%
drsp43x4-5	15	223	582.33	374.95	-35.61%
drsp43x4-6	22	347	528.00	528.70	0.13%
drsp43x4-7	16	261	497.84	367.13	-26.26%
drsp43x4-8	23	468	528.30	455.39	-13.80%
drsp43x4-9	18	313	650.22	418.83	-35.59%
Average	19.00	323.44	553.97	424.77	-22.78%
drsp63x6-1	33	676	685.87	635.04	-7.41%
drsp63x6-2	34	678	777.73	735.61	-5.42%
drsp63x6-3	31	494	811.37	726.05	-10.51%
drsp63x6-4	35	700	786.22	721.99	-8.17%
drsp63x6-5	30	515	714.26	665.46	-6.83%
drsp63x6-6	33	560	827.48	847.60	2.43%
drsp63x6-7	30	506	799.50	654.05	-18.19%
drsp63x6-8	28	383	873.49	791.88	-9.34%
drsp63x6-9	34	608	782.46	650.52	-16.86%
Average	32.00	568.89	784.26	714.24	-8.92%
drsp83x8-1	40	582	1114.75	1020.69	-8.44%
drsp83x8-2	38	722	1411.50	1040.39	-26.29%
drsp83x8-3	39	660	1310.40	1173.44	-10.45%
drsp83x8-4	43	740	1050.79	1051.51	0.07%
drsp83x8-5	34	516	1155.35	882.54	-23.61%
drsp83x8-6	39	661	1092.90	1019.06	-6.76%
drsp83x8-7	41	730	1124.13	1019.06	-9.35%
drsp83x8-8	40	726	1315.08	1340.66	1.95%
drsp83x8-9	39	668	1212.02	1052.88	-13.13%
Average	39.22	667.22	1198.55	1066.69	-10.67%

Table 4.1 shows the results obtained after running our approach for the considered instances. The limited number of vehicles and their limited capacities make infeasible to visit all the potential customers. Hence, the maximum number of served customers is 23, 35, and 43 for each group of instances, respectively. The total collected fee for serving these

customers is also shown. Then, we employ our algorithm to generate two different solutions: our best static solution (OBS) and our best dynamic solution (OBD). The OBS is a solution generated in the period 0 (an initial solution) that cannot be further modified, i.e., the vehicles perform the same static routes regardless of the traffic conditions. Alternatively, the OBD is a solution that is recomputed each 10 time units and, therefore, it adapts to the changing traffic conditions. This recalculation is performed in the same way as the initial solution, but considering only the remaining customers to serve and the cost per edge given by Equation (4.2). The total collected fee is not different for the OBS and the OBD since the customers selected to be served by the initial solution are respected in all the recalculations done to obtain the OBD. Only the routes can change in this case. Finally, gaps in Table 4.1 represent the percentage difference between the costs attained by the OBD and the ones attained by the OBS. A negative gap means that the OBD obtains a lower cost than the OBS. Average gaps are always negative regardless of the instance size. Only 4 out of 27 instances reach a positive gap, with a maximum value of only 2.43% for the instance *drsp63x6-6*. Conversely, the most negative gap reaches a value of -35.61% for the instance *drsp43x4-5*. These results indicate that, in terms of costs, our dynamic approach always outperforms the scenario in which the solution is not adaptable to external changing conditions.

Figure 4.2 shows a series of box plots displaying the attained costs by the OBS (pink) and the OBD (green). Each box plot depicts the group of instances classified according to their size. Additionally, a crossed circle indicates the mean of each group of data. This figure shows the natural cost increase when the instance size grows. Nevertheless, this increase is made up for the proportional rise in the total collected fee, as Table 4.1 shows. Furthermore, Figure 4.2 indicates graphically the cost savings attained after solving this problem with our dynamic approach, instead of employing a static metaheuristic.

Figure 4.3 depicts an example of the execution of our algorithm in two consecutive periods for the instance *drsp63x6-2*. Big black nodes indicate the origin points, big red nodes indicate the destination points, medium-sized numbered nodes represent the served customers, and small unnumbered gray points represent the non-served customers. Routes are depicted by lines of different colors and styles. Figure 4.3a shows the initial best-found solution (BFS), generated in the period $n = 0$. If this set of routes is not further modified, then we obtain the OBS, at a cost of 777.73 (Table 4.1). Alternatively, Figure 4.3b shows the BFS in the period $n = 1$ considering dynamic conditions and, therefore, the originally designed routes have changed to eventually obtain the OBD. At this moment, each vehicle has already arrived to the location of its respective first customer. Our algorithm takes these locations as new origin points, hence, the former origins are represented by green nodes in Figure 4.3b. Notice that routes after the black nodes in this figure are different from those of Figure 4.3a. For instance, the black dashed route in Figure 4.3a follows the sequence *4-65-24-12-47-40-69*. In the period $n = 1$, the corresponding vehicle has already traveled from node *4* to node *65*. Nevertheless, the dynamic traffic conditions makes that the original planned route changes and the vehicle follows the new sequence *4-65-40-32-69*. The total customers selected in the period $n = 0$ remain the same, i.e., the commitment of serving them is respected, although routes are different now. Given the limited space, the new re-computed routes for the rest of

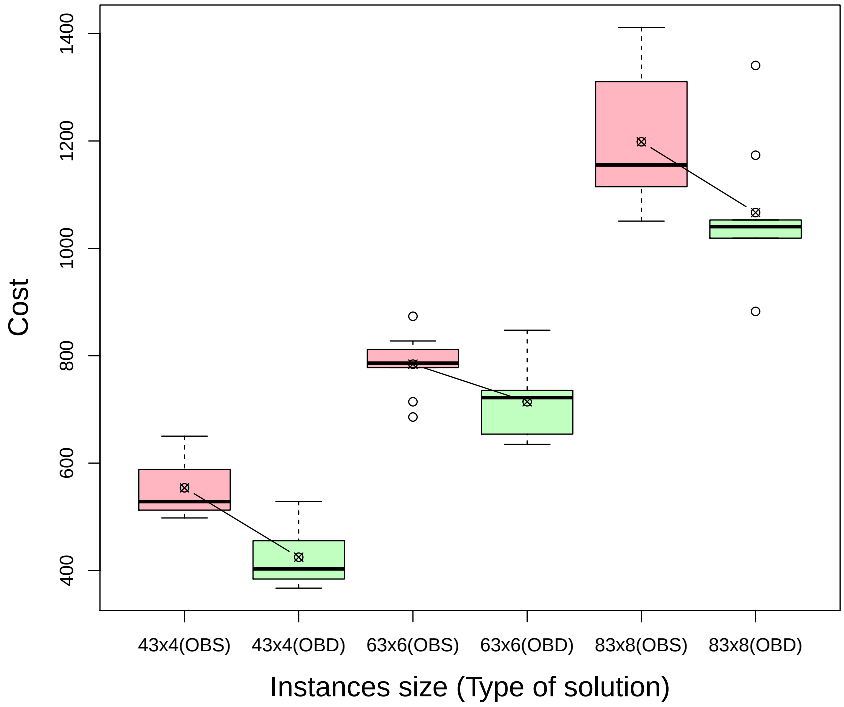


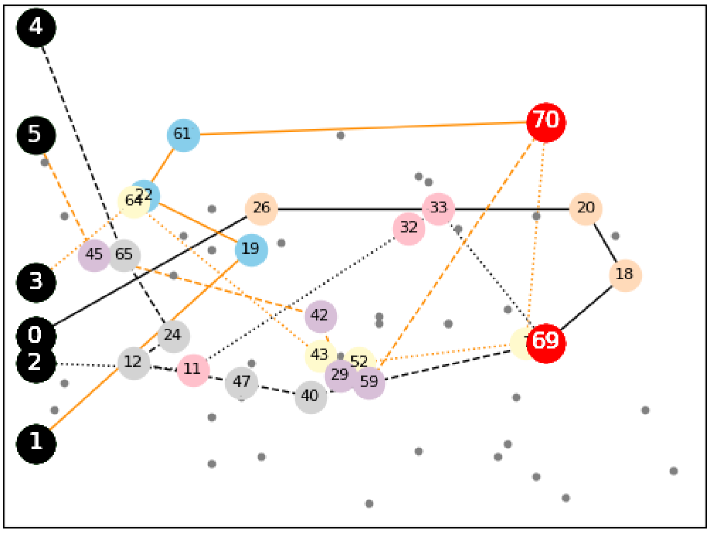
Figure 4.2: Costs obtained by each type of solution for different instance sizes.

the periods are not displayed, however, this process is repeated until every vehicle has arrived to the nodes *69* and *70*, respectively. All these successive recalculations lead to obtain the OBD, at a cost of 735.61 (Table 4.1).

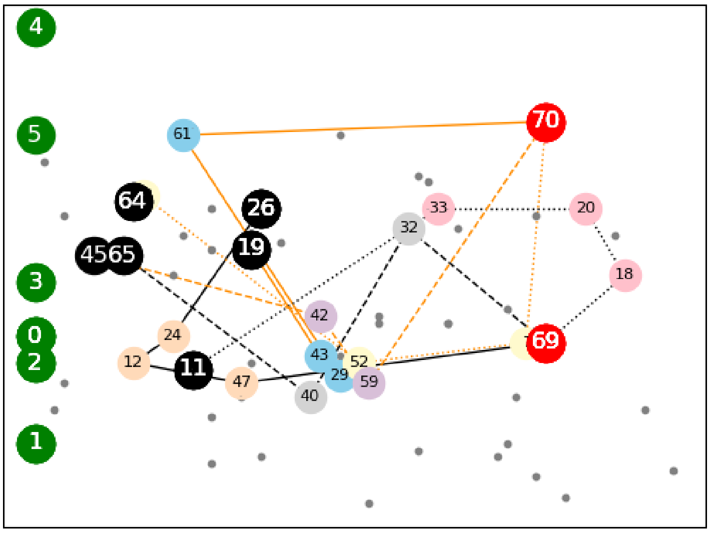
4.2 The VRP with Optional Backhauls

Reverse logistics and closed-loop supply chains have been increasingly studied in recent years. Both concepts are related to the return of products or materials from the point of consumption in order to recover value. In particular, Govindan and Soleimani (2017) found that most addressed issues are re-manufacturing and waste management, while the topic of package recovery is barely tackled despite its environmental impact (Kroon and Vrijens, 1995). Given these considerations, disposable packages have been replaced by returnable transport items (RTI), e.g., reusable pallets, trays, boxes, or any other mean to assemble goods (ISO, 2016). Still, environmental issues are not the only concern regarding RTIs management. According to Glock (2017), RTIs are an important asset for many industries, since they can decrease the selling cost for customers.

Examples from different industries highlight the importance of RTIs in real-world transportation practices. For instance, in agri-food supply chains it is usual that products are harvested and transported in boxes or baskets to preserve their quality (Tordecilla-Madera et al., 2018). Once these products have been delivered to customers, they are unpacked and RTIs are prepared to be returned to the supplier (Kim et al., 2014). The drink industry



(a) Period $n = 0$.



(b) Period $n = 1$.

Figure 4.3: Example of the BFS for the instance *drsp63x6-2* in the periods $n = 0$ and $n = 1$.

also uses RTIs in the distribution process to decrease costs or loss rate. This is the case, for instance, when transporting beer (Fan et al., 2019) or soft drinks (Soysal, 2016; Koç and Laporte, 2018). Finally, Mason et al. (2012) provide an example from the gas industry. These authors focus on tracking the cylinders, since these RTIs are highly likely to be lost or stolen. They present an inventory management system based on the use of radio frequency identification (RFID) technology as a more sophisticated identification technique (Ilic et al., 2009). The problem discussed in this section is motivated by a real-world case from the agri-food industry: a Colombian company that produces packed bread and cereal, distributing these products in RTIs. A complete review including several real-life cases on the use of RTIs is provided by Mahmoudi and Parviziomran (2020).

A good strategy for RTIs management is the interchange between suppliers and customers (Elia and Gnoni, 2015), i.e., suppliers deliver products in RTIs and customers return them empty. However, such interchange cannot be done simultaneously because pickups are only possible when deliveries have already been made and the vehicle is empty (Koç and Laporte, 2018). Therefore, the return of RTIs must be performed in two alternative ways: *(i)* by using dedicated collection vehicles; or *(ii)* by using vehicles that make deliveries firstly and then collections. The latter case is known as the vehicle routing problem with backhauls (VRPB) (Berbeglia et al., 2007; Belloso et al., 2017). Being a rich extension of the well-known VRP (Caceres-Cruz et al., 2014), the VRPB is also an NP-hard problem. Studies on the VRPB typically assume that all customers must be visited. However, in our case we explore the scenario in which visiting backhaul (BH) customers is optional, i.e., as illustrated in Figure 4.4, the carrier might decide not to visit some BH nodes, thus incurring in a penalty cost. This cost is associated with the fact that some customers will need to temporarily store the empty RTIs until a new visit is scheduled.

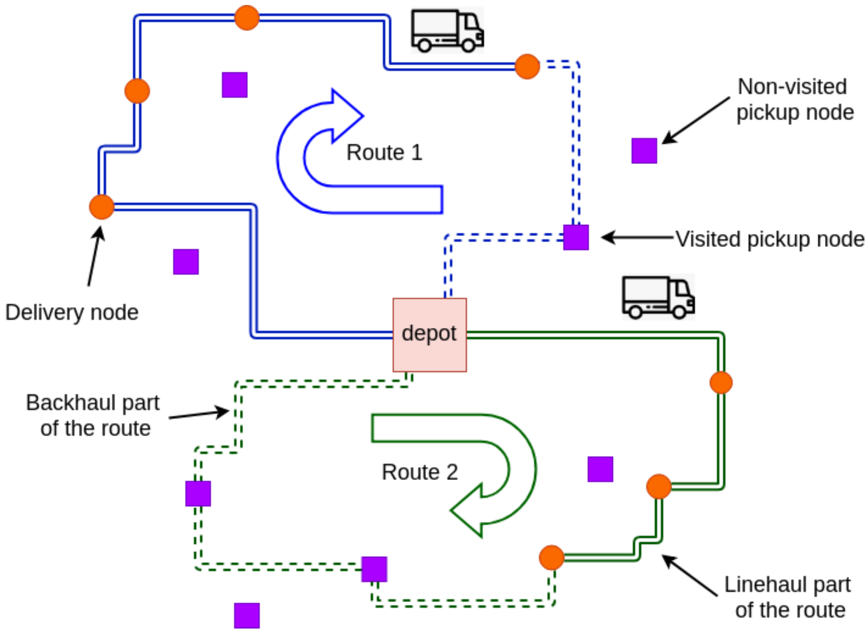


Figure 4.4: Illustrating the vehicle routing problem with optional backhauls.

Thus, for example, if the quantity of RTIs available in a supplier's facility is enough to ensure future deliveries, collection of RTIs in the current period can be reduced to diminish routing cost or speed up the routing process. This makes sense whenever the cost of holding the additional inventory at the customers facilities –i.e., the cost of holding the RTIs in stock– is lower than the marginal routing cost associated with their collection. In practice, the cost of holding RTIs in stock at the customers facilities might vary from one period to another, depending on factors such as how many RTIs are needed for the next period distribution or how long have been the RTIs staying at the customers inventories. Accordingly, the main contribution of this section can be summarized as follows: *(i)* we propose and analyze the VRPOB, which has been scarcely considered in the literature; *(ii)* we provide a mathematical model of the problem, which is then solved using exact methods for small-scale instances; *(iii)* we propose a BR ILS to solve larger instances of the problem; and *(iv)* we analyze how routing solutions evolve as we consider different levels of penalty cost. BR techniques allow for extending traditional metaheuristic frameworks in order to enhance their performance (Gonzalez-Neira et al., 2017; Ferone et al., 2019) and facilitate agile optimization (Martins et al., 2021a).

4.2.1 Problem Definition

The VRPOB consists in a set of linehaul (LH) and BH customers whose demands must be satisfied –at least the LH ones– using a fleet of homogeneous vehicles initially located at a depot. This depot has enough capacity and vehicles to cover the aggregated customers' demand. Therefore, all LH customers will be serviced. The supplier uses RTIs for transporting product units. Hence, after delivering all units, empty RTIs from previous deliveries should be collected at some BH customers and returned to the central depot for future deliveries. Collecting RTIs from BH customers is optional, and not doing it might generate savings in transportation costs, but it will also raise some penalty costs associated with the lack of service, which implies that some customers will need to keep the RTIs in stock until the next visit. Thus, in the VRPOB the following decisions must be made in order to minimize the total cost (routing cost plus penalty cost): *(i)* to determine which BH customers will be visited; *(ii)* to assign customers to a predefined number of routes (and vehicles); and *(iii)* to establish the sequence in which customers should be visited.

Formally speaking, consider a non-directed graph $G = (V, A)$, with $V = \{0\} \cup L \cup B$ representing the set of nodes, being node 0 the depot, $L = \{1, 2, \ldots, n\}$ the set of n LH customers, and $B = \{n+1, n+2, \ldots, n+m\}$ the set of m BH customers. Likewise, $A = \{(i, j) : i, j \in V, i < j\}$ is the set of edges linking each pair of nodes. For each $i \in L$, there is a positive demand $d_i > 0$ representing units of product to be delivered. Similarly, for each $j \in B$, there is a negative demand $d_j < 0$ representing items to be collected. Also, consider a set K of homogeneous vehicles available at the depot, each of them with a capacity $q >> \max\{|d_i| : i \in V\}$. Travelling an edge from node i to node j has a cost $c_{ij} = c_{ji} > 0$. The following constraints need to be considered:

- Each route begins and ends at the depot.
- Each route must have at least one LH customer, i.e., routes formed just by BH customers are not allowed.
- In any route, LH customers are serviced before BH customers.
- Each LH customer must be mandatorily serviced, but visiting BH customers is optional.
- For each route, the quantity of product to deliver and to collect must not exceed the vehicle's capacity.
- Whenever a LH customer is visited in a route, all its demand is serviced; similarly, whenever a BH customer is visited in a route, all its items are collected.

The VRPOB can be formulated as a mixed-integer linear program (MILP), which is shown in Appendix A.1.1.

4.2.2 Solution Approach

Being an extension of the VRP, the VRPOB is also an NP-hard problem. Therefore, a BR version of a metaheuristic algorithm is proposed to solve large-size instances of the VRPOB. This algorithm employs the ILS (Lourenço et al., 2019) as the base metaheuristic. Hence, our algorithm relies on a perturbation stage (a destruction-reconstruction one), which significantly modifies a base solution, followed by an LS that tries to improve the modified solution. The algorithm also uses an acceptance criterion to update the base solution, whenever the criterion is satisfied, even if the new solution does not improve it. This whole process is done iteratively. Algorithm 7 presents the general idea of our approach. Firstly, an initial solution is obtained through a savings-based heuristic (line 1), which incorporates a penalization strategy to delay merging between LH and BH nodes. This initial solution is improved through a fast LS procedure (line 2) composed of: *(i)* node-insertion and node-swap operators; and *(ii)* a cache (hash map) data structure. Then, the iterated search starts from this initial solution, henceforth called as *base* solution. Here, r% of the routes from the base solution are destroyed and reconstructed (line 5). It implies solving a smaller problem using a BR version of the heuristic (line 6), which is later incorporated into the base solution (line 7). This new solution is later improved by applying an LS procedure (line 8). Finally, an acceptance criterion –based on the cost of the solutions– is employed for selecting and evaluating the best-found and base solutions (line 9). This process is repeated until an ending criterion is met (line 10). Finally, the best-found solution is returned (line 11). These steps are detailed below.

Algorithm 7 Procedure ILS

1: $initialSolution \leftarrow$ penalizedBRSavingsHeuristic($inputParameters, \alpha, \beta, \lambda$)
2: $initialSolution \leftarrow$ fastLocalSearch($inputParameters, initialSolution$)
3: $baseSolution \leftarrow initialSolution$
4: **repeat**
5: $\quad pendingNodes \leftarrow$ destroyRoutes($r, baseSolution$)
6: $\quad subSolution \leftarrow$ penalizedBRSavingsHeuristic($inputParameters, \alpha, \beta, pendingNodes$)
7: $\quad newSolution \leftarrow baseSolution \cup subSolution$
8: $\quad newSolution \leftarrow$ fastLocalSearch($inputParameters, newSolution$)
9: $\quad bestSolution \leftarrow$ acceptanceCriterion($baseSolution, newSolution$)
10: **until** time reaches the limit
11: **return** $bestSolution$

4.2.2.1 Generating an Initial Solution

In order to generate an initial solution, a savings-based heuristic is enhanced and extended by including the following three strategies: *(i)* introducing a penalization cost in the savings calculation, which refers to the possibility of considering optional backhauls; *(ii)* penalizing any merging process between LH and BH nodes to delay their selection –this strategy guarantees that all deliveries are done before pickups; and *(iii)* including a BR procedure during the search stage. The first novelty of our approach refers to an alternative savings value for merging routes. Since each BH customer has a penalty cost for not being served, the traditional way to calculate such savings ($s_{ij} = c_{i0} + c_{0j} - c_{ij}$) is extended according to the expression in Equation (4.3), proposed by Panadero et al. (2020b) for the team orienteering problem, where h_i and h_j are the unitary penalty costs for not visiting the BH customers i and j, respectively. Notice that this extended savings applies just to edges whose both nodes are BH ones.

$$s'_{ij} = \alpha s_{ij} + (1-\alpha)(h_i + h_j), \ \forall i \in B, \forall j \in B \tag{4.3}$$

In Equation (4.3), $0 \leq \alpha \leq 1$. The idea is that this α tries to balance both penalty costs and traditional savings in distances or times. The case in which $\alpha = 1$ corresponds to the traditional case in which penalty costs are not considered. In the case in which $\alpha = 0$, savings in transport cost are not considered, and only penalty costs are relevant. Intermediate values of α assign more or less weight to transport and penalty costs. As a second extension, saving values are updated in order to address properly the BH nodes and, consequently, provide a feasible merging between LH and BH customers. This procedure is described in Algorithm 8. The idea is to delay the selection of interface edges, i.e., those ones linking a LH node with a BH node, so that both LH and BH routes are "complete" before being merged. This delay is done by subtracting p from the previously computed savings (line 7). The value of p is computed as $p = max_s * \lambda$ (line 6), in which max_s represents the maximum savings that can be attained (line 2), while λ is a penalty coefficient ranging between 0 and 1. In our case, λ is uniformly chosen at random between 0.05 and 0.20 (line 5).

Algorithm 8 Procedure penalizeSavingsList

1: $savingsList \leftarrow$ createSavingsList($inputParameters, \alpha$)
2: $max_s \leftarrow$ obtainMaximumSavings($savingsList$)
3: **for** each $edge$ in $savingsList$ **do**
4:     **if** $edge$ is interface **then**
5:         Randomly select $\lambda \in \{0.05, 0.20\}$
6:         $p \leftarrow max_s * \lambda$
7:         updateSavings($edge, p$)
8:     **end if**
9: **end for**
10: **return** $savingsList$

The first stage of constructing an initial solution refers to the creation of a dummy solution, which is composed of a set of single-node routes. Therefore, the last extension in this stage is the use of BR techniques (Grasas et al., 2017) to guide the selection of an element in the penalized SL. BR assigns a probability of being chosen to each edge in this list. Therefore, the selection is smoothed by replacing the original greedy behavior with a probabilistic one. To achieve this purpose, the geometric probability distribution is employed in a Monte Carlo simulation, in which only one parameter (β) must be fine-tuned. After sorting the SL in descending order, β can be interpreted as the probability of choosing the edge with the highest savings. As a result of performing some preliminary experiments, we observed that good results are obtained when β is selected uniformly randomly between 0.01 and 0.50. The selection within this interval provides the algorithm with the right balance between exploration and exploitation of the solution space, since $\beta = 0$ refers to a uniform random selection, and $\beta = 1$ represents a greedy strategy.

Once a promising merging edge is selected, a set of pre-defined merging conditions are checked. The two corresponding routes of an edge e can be merged if: *(i)* its nodes are adjacent to the depot; *(ii)* its nodes belong to different routes; and *(iii)* the vehicle capacity constraint is met. If both routes belong to the same cluster, i.e., to either the LH group or the BH group, the merge is accepted. Otherwise, the optional pickups for that merging are analyzed: this union is allowed only if the total transportation cost is less than total penalty cost for not picking up. Thus, if the total transportation cost for BH nodes is high, the algorithm is more likely to choose not to collect RTIs in that route. In this case, the union is discarded, and the BH route is penalized. At the end of each iterative step, the selected edge is removed from the SL, and the process is repeated until no more edges are available.

By not taking into account the number of vehicles at this stage, this procedure for creating a solution might lead to infeasible solutions regarding this hard constraint. Therefore, a recursive corrective operator is executed at the end of the procedure to attain it (Belloso et al., 2017). Broadly speaking, this procedure relies on selecting potential routes to be kept in the solution –according to its current status– and reconstructing new routes in order to meet the number of routes constraint. When the current solution contains fewer routes than required (the available number of vehicles), large-sized demand routes are more likely to be chosen to be incorporated into the new solution, then allowing the construction of more routes. Alternatively, when the solution is composed of more routes than the number of available vehicles, small-sized demand routes have more probability of being selected. The remaining nodes –from non-selected routes– are now part of a subproblem, which is solved by the same mechanism, and then incorporated into the final solution. This procedure is repeated until a feasible solution –i.e., a solution which is composed of $|K|$ routes– is found.

4.2.2.2 Local Search Procedures

The procedure in Section 4.2.2.1 yields an initial solution that can be improved. Taking into account the problem structure, an LS procedure based on the use of an inter-routes node-insertion operator, an inter-routes nodes-swap operator, and a cache (memory-based or hash map) data structure is implemented. The cache procedure relies on storing in memory the best-found route for a given group of nodes. Whenever a lower-cost route formed by the same set of nodes is found, it is returned by the method. Otherwise, in case of the current route has a better cost, it replaces the existing one in memory. Finally, if the route does not exist, it is inserted into the cache. This procedure is applied whenever a new solution is constructed (Algorithm 7, lines 2 and 8).

4.2.2.3 Perturbation Stage

The improved initial solution is used as a base solution by the algorithm. At each iteration, this base solution is perturbed. The perturbation process consists in destroying at least two routes (Algorithm 7, line 5) and reconstructing them. Such destruction implies that, temporarily, m nodes do not belong to any route. Observe that $m < n$, where n is the total number of nodes in the instance. This implies that a subproblem must be solved and, therefore, a smaller solution will be attained. In this way, a destruction ratio r is defined in order to establish a maximum number of routes to be destroyed (DR). Following Belloso et al. (2017), r is generated as a random number between 0.10 and 0.50. According to these authors, a general value was adopted for this parameter in order to avoid the need for performing complex and time-expensive fine-tuning processes. On the one hand, a destruction ratio larger than 0.50 would destroy most of the constructed routes and the process would lose time-efficiency. On the other hand, a destruction ratio lower than 0.10 would destroy very few routes, and the effect of the perturbation would be almost negligible. The number of routes to be destroyed is computed as the maximum between 2 routes and an r% of the total routes in the solution:

$$DR = \max\{2,\ r * \#Routes\} \tag{4.4}$$

Once the destruction process is finished, the reconstruction is made from scratch for the m nodes, by using again the procedure described in Section 4.2.2.1. After a feasible solution is generated, the LS mechanism described in Section 4.2.2.2 is applied (Algorithm 7, line 8), and the acceptance criterion is tested (Algorithm 7, line 9). This process (Algorithm 7, lines 4-10) is repeated until a maximum time limit is reached.

4.2.2.4 Acceptance Criterion

An acceptance criterion is incorporated into our approach in order to accept promising solutions and avoid local minima. This mechanism relies on the possibility of updating the base solution by another one of lower quality in order to provide the exploration of new regions in the solution space.

4.2.3 Computational Experiments and Results

A set of 33 instances introduced by Toth and Vigo (1997) (*TV* instances) has been solved in order to evaluate the performance of our proposed methodology. These instances are different in the number of LH and BH nodes, vehicle capacities, and demands. In total, 3 proportions of LH customers are used: 50%, 66%, and 80%. Thus, for instance, in the latter case (80% or 4 out of 5) customers 1, 2, 3, and 4 are LH, while the 5th one is BH, and so on. Here, we can assume that RTIs to be collected were used in previous periods to deliver products to the associated customer. Small instances can be solved through an exact method using the model introduced in Appendix A.1.1. Larger instances need a heuristic or metaheuristic approach such as the one introduced in Section 4.2.2.

4.2.3.1 Solving Small-Scale Instances with Exact Methods

The model in Appendix A.1.1 was implemented in IBM CPLEX, which was used to solve the instances *eil_22*, *eil_23*, *eil_30*, *eil_33*, and *eil_51* as proposed by Christofides and Eilon (1969). Such instances were adapted for the VRPB by Toth and Vigo (1997). Four scenarios are considered for the unitary penalty cost, h_i, namely: high, medium-high, medium-low, and low. In the *eil_51* instances, the unitary penalty cost $h_i \in \{12.40, 6.20, 3.10, 1.24\}$ for the respective scenario. In contrast, for smaller instances this value is $h_i \in \{0.66, 0.33, 0.16, 0.06\}$. These h_i values were established after some preliminary tests and they differ in magnitude due to the demand size of the instances. Table 4.2 shows the results for instances with 22, 23, 30, 33, and 51 nodes. For instances with 22 or 33 nodes, all BH customers are visited regardless of the proportion of LH and BH customers. Some BH customers are not visited in other instances, although just when the unitary penalty cost is not high (this is the case of instances with 23 nodes), or when it is medium-low or low (this is the case of instances with 30 or 51 nodes). These results confirm that the value of the unitary penalty cost affects the number of non-serviced BH (NSBH) customers. As expected, the lower the h_i, the higher the number of NSBH customers. Notice that, for all considered instances, all BH customers are serviced when h_i is high.

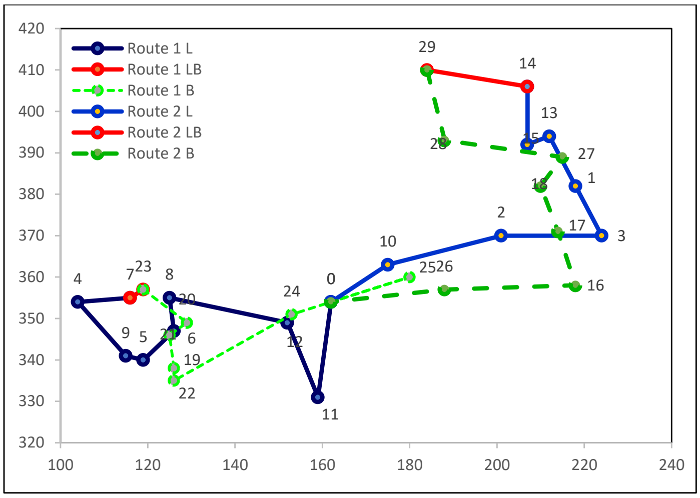
In general, not collecting RTIs might yield some savings in total cost when the unitary penalty cost is not high. For example, in instance *eil23_66*, when using $h_i = 0.66$, the total cost of the solution is 649.0, which corresponds to visiting all customers –i.e., no penalty cost is added in this case. However, when $h_i = 0.33$, the associated penalty cost is 127.1 and the total cost is 601.1 –i.e., equivalent to a reduction of 7.4% in total cost. The performance is similar for medium-low and low penalty costs in instances with 23, 30, and 51 nodes. These results show that, if the unitary penalty cost is low, it might pay off not to visit all BH customers. Figure 4.5 illustrates a solution for instance *eil30_50* with $h_i = 0.66$ (high) and $h_i = 0.06$ (low). All customers are visited in the first case *(a)*, while 3 BH customers are not serviced in the second case *(b)*. The depot is depicted as node 0. In this example, non-visited customers are those that are more distant from the depot and have the lowest demands (in absolute value). Finally, notice that changes in the number of visited BH customers might also affect the LH part of a route.

4.2.3.2 Solving Large-Scale Instances with Our Metaheuristic Approach

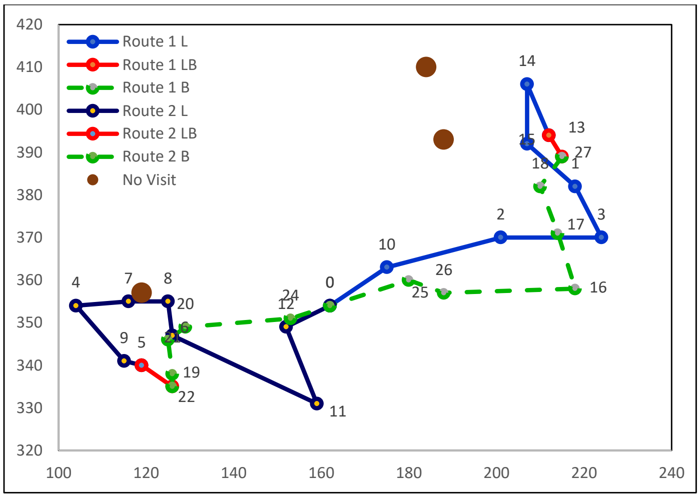
As displayed in Table 4.2, computational times increase dramatically with the size of each instance. For example, instance *eil51_80* employs more than 6.5 hours in finding the optimal solution. Realistic instances might have even a greater number of nodes, thus limiting the possibility of using exact methods to solve them. Hence, for solving these large-sized instances, the metaheuristic algorithm proposed in Section 4.2.2 is employed. The proposed approach has been implemented in Java, and a standard PC with an Intel Core *i7* CPU at 2.7 GHz and 16 GB RAM has been employed to run all tests. Initially, we will assume that visiting all customers is mandatory (i.e., for any customer, the penalty cost associated with non-visiting it is extremely large). This will allow us to compare the performance of our algorithm with the results provided by Toth and Vigo (1997) and Belloso et al. (2017). Table 4.3 shows the comparison between our best solutions (OBS) and the best-known solutions (BKS) in the literature. Notice that, our method is able to match the BKS for 91% of the tested instances, also achieving similar results to the ones provided by Belloso et al. (2017) and outperforming the results by Toth and Vigo (1997). Therefore, it is possible to conclude

Table 4.2: Solutions of the exact algorithm for different levels of unitary penalty cost.

Instance	h_i	Total BH demand	LH	BH	Total penalty cost	Total cost	Gap (cost reduction)	Number of non-collected RTIs	% of non-collected RTIs	NSBH	Time (s)
eil22_50	0.66	12800	11	10	0.0	371.0	0.0%	0	0.0%	0	9.2
	0.33				0.0	371.0	0.0%	0	0.0%	0	5.7
	0.16				0.0	371.0	0.0%	0	0.0%	0	5.4
	0.06				0.0	371.0	0.0%	0	0.0%	0	7.6
eil22_66	0.66	5500	14	7	0.0	366.0	0.0%	0	0.0%	0	3.3
	0.33				0.0	366.0	0.0%	0	0.0%	0	2.2
	0.16				0.0	366.0	0.0%	0	0.0%	0	6.2
	0.06				0.0	366.0	0.0%	0	0.0%	0	2.6
eil22_80	0.66	5400	17	4	0.0	375.0	0.0%	0	0.0%	0	11.2
	0.33				0.0	375.0	0.0%	0	0.0%	0	40.8
	0.16				0.0	375.0	0.0%	0	0.0%	0	36.8
	0.06				0.0	375.0	0.0%	0	0.0%	0	10.1
eil23_50	0.66	6604	11	11	0.0	682.0	0.0%	0	0.0%	0	2.2
	0.33				24.8	657.8	-3.6%	175	2.6%	1	1.9
	0.16				12.0	645.0	-5.4%	760	11.5%	5	1.5
	0.06				93.2	567.2	-16.8%	1360	20.6%	6	0.7
eil23_66	0.66	2080	15	7	0.0	649.0	0.0%	0	0.0%	0	2.9
	0.33				127.1	601.1	-7.4%	385	18.5%	3	2.2
	0.16				89.6	533.6	-17.8%	560	26.9%	4	2.3
	0.06				33.6	477.6	-26.4%	560	26.9%	4	3.0
eil23_80	0.66	5050	18	4	0.0	623.0	0.0%	0	0.0%	0	2.3
	0.33				49.5	608.5	-2.3%	150	3.0%	1	1.6
	0.16				24.0	583.0	-6.4%	150	3.0%	1	1.8
	0.06				57.0	542.0	-13.0%	950	18.8%	3	1.9
eil30_50	0.66	5825	15	14	0.0	501.0	0.0%	0	0.0%	0	6.4
	0.33				0.0	501.0	0.0%	0	0.0%	0	12.8
	0.16				40.0	481.0	-4.0%	250	4.3%	2	3.6
	0.06				36.0	431.0	-14.0%	600	10.3%	3	2.7
eil30_66	0.66	3500	20	9	0.0	537.0	0.0%	0	0.0%	0	34.9
	0.33				0.0	537.0	0.0%	0	0.0%	0	53.5
	0.16				16.0	523.0	-2.6%	100	2.9%	1	45.1
	0.06				6.0	513.0	-4.5%	100	2.9%	1	40.0
eil30_80	0.66	1950	24	5	0.0	514.0	0.0%	0	0.0%	0	45.0
	0.33				0.0	514.0	0.0%	0	0.0%	0	6.6
	0.16				0.0	514.0	0.0%	0	0.0%	0	19.5
	0.06				12.0	503.0	-2.1%	200	10.3%	1	25.8
eil33_50	0.66	11890	16	16	0.0	738.0	0.0%	0	0.0%	0	22.5
	0.33				0.0	738.0	0.0%	0	0.0%	0	41.0
	0.16				0.0	738.0	0.0%	0	0.0%	0	39.6
	0.06				0.0	738.0	0.0%	0	0.0%	0	17.1
eil33_66	0.66	6480	22	10	0.0	750.0	0.0%	0	0.0%	0	14.4
	0.33				0.0	750.0	0.0%	0	0.0%	0	21.9
	0.16				0.0	750.0	0.0%	0	0.0%	0	33.3
	0.06				0.0	750.0	0.0%	0	0.0%	0	31.3
eil33_80	0.66	5540	26	6	0.0	736.0	0.0%	0	0.0%	0	62.2
	0.33				0.0	736.0	0.0%	0	0.0%	0	199.2
	0.16				0.0	736.0	0.0%	0	0.0%	0	188.1
	0.06				0.0	736.0	0.0%	0	0.0%	0	112.4
eil51_50	12.40	401	22	10	0.0	559.0	0.0%	0	0.0%	0	115.0
	6.20				0.0	559.0	0.0%	0	0.0%	0	92.1
	3.10				0.0	559.0	0.0%	0	0.0%	0	134.9
	1.24				43.4	538.4	-3.7%	35	8.7%	5	58.9
eil51_66	12.40	257	22	10	0.0	548.0	0.0%	0	0.0%	0	322.0
	6.20				0.0	548.0	0.0%	0	0.0%	0	765.3
	3.10				0.0	548.0	0.0%	0	0.0%	0	382.9
	1.24				17.4	542.4	-1.0%	14	5.4%	2	1 190.4
eil51_80	12.40	155	22	10	0.0	553.0	0.0%	0	0.0%	0	3 864.8
	6.20				0.0	553.0	0.0%	0	0.0%	0	10 616.3
	3.10				0.0	553.0	0.0%	0	0.0%	0	15 812.8
	1.24				26.0	539.0	-2.5%	21	13.5%	2	23 646.0



(a)



(b)

Figure 4.5: Optimal solutions for instance *eil30_50* and unitary penalty costs of 0.66 *(a)* and 0.06 *(b)*.

that our algorithm obtains competitive solutions when compared with state-of-the-art approaches in the VRPB with mandatory backhauls. The next step is to use our algorithm to solve the VRPOB, that is, the version of the problem in which visiting BH customers is optional and, therefore, can be skipped if savings in routing cost are higher than the associated penalty cost.

Accordingly, new tests were carried out in order to measure the performance when parameters h_i and α, from Equation (4.3), are incorporated in the solution cost. We set $\alpha \in \{0.2, 0.5, 0.8, 1.0\}$. Regarding the unitary penalty costs, we set $h_i \in \{0.66, 0.33, 0.16, 0.06\}$. Specifically, for h_i we set this parameter from higher to lower values in order to measure the impact on the cost when collecting and not collecting the RTIs, respectively. A higher value of h_i ensures not to fail collecting any RTI, while a lower h_i might generate a better solution cost by not collecting all of them. For calibrating the parameter α, we have followed the methodology proposed by Calvet et al. (2016), who provided a general procedure, based on statistical learning. This procedure does the following: *(i)* it chooses a subset of benchmark

Table 4.3: Comparison between our algorithm and previous works.

Instance	LH	BH	Q	K	Toth and Vigo (1997) (BKS in 1997)	Belloso et al. (2017) (BKS in 2017)	OBS ($h_i = \infty$)	Gap OBS-Belloso
eil22_50	11	10	6000	3	371.0	371.0	371.0	0.0%
eil22_66	14	7	6000	3	366.0	366.0	366.0	0.0%
eil22_80	17	4	6000	3	375.0	375.0	375.0	0.0%
eil23_50	11	11	4500	2	682.0	682.0	682.0	0.0%
eil23_66	15	7	4500	2	649.0	649.0	649.0	0.0%
eil23_80	18	4	4500	2	623.0	623.0	623.0	0.0%
eil30_50	15	14	4500	2	501.0	501.0	501.0	0.0%
eil30_66	20	9	4500	3	537.0	537.0	537.0	0.0%
eil30_80	24	5	4500	3	514.0	514.0	514.0	0.0%
eil33_50	16	16	8000	3	738.0	738.0	738.0	0.0%
eil33_66	22	10	8000	3	750.0	750.0	750.0	0.0%
eil33_80	26	6	8000	3	736.0	736.0	736.0	0.0%
eil51_50	25	25	160	3	559.0	559.0	559.0	0.0%
eil51_66	34	16	160	4	548.0	548.0	548.0	0.0%
eil51_80	40	10	160	4	565.0	565.0	565.0	0.0%
eilA76_50	37	38	140	6	739.0	739.0	739.0	0.0%
eilA76_66	50	25	140	7	768.0	768.0	768.0	0.0%
eilA76_80	60	15	140	8	781.0	781.0	781.0	0.0%
eilB76_50	37	38	100	8	801.0	801.0	801.0	0.0%
eilB76_66	50	25	100	10	873.0	873.0	873.0	0.0%
eilB76_80	60	15	100	12	919.0	919.0	919.0	0.0%
eilC76_50	37	38	180	5	713.0	713.0	713.0	0.0%
eilC76_66	50	25	180	6	734.0	734.0	734.0	0.0%
eilC76_80	60	15	180	7	733.0	733.0	733.0	0.0%
eilD76_50	37	38	220	4	690.0	690.0	690.0	0.0%
eilD76_66	50	25	220	5	715.0	715.0	715.0	0.0%
eilD76_80	60	15	220	6	703.0	694.0	695.0	0.1%
eilA101_50	50	50	200	4	843.0	831.0	831.0	0.0%
eilA101_66	67	33	200	6	846.0	846.0	846.0	0.0%
eilA101_80	80	20	200	6	916.0	856.0	856.0	0.0%
eilB101_50	50	50	112	7		923.0	925.0	0.2%
eilB101_66	67	33	112	9		982.0	987.0	0.5%
eilB101_80	80	20	112	11		1008.0	1008.0	0.0%
Average								0.0%

instances at random; *(ii)* it selects the range over which each parameter will be varied; *(iii)* it applies an experimental design to explore promising regions; and *(iv)* it obtains a set of parameter values by intensifying the search. Following this procedure, only small instances with known optimal solutions were considered. Thus, for each combination of instances, unitary penalty cost h_i, and α, a total of 30 runs were performed (each run using a different seed for the pseudo-random number generator). The performance of the metaheuristic was measured as the percentage gap between OBS and the BKS –which is also the optimal solution in this case. After these experiments, a value of $\alpha = 0.8$ is set for testing our approach on large-sized instances.

A total of 30 runs were performed for each of the 33 instances and h_i value. In the case of large instances (those ones composed of 55 nodes or more), a new fine-tuning process was carried out to establish their h_i values, namely: high ($h_i = 12.40$), medium-high ($h_i = 6.20$), medium-low ($h_i = 3.10$), and low ($h_i = 1.24$). Re-adjusting the values of h_i for these instances is necessary since they have different levels of demand as compared to the smaller ones. The stopping criterion was fixed to 25, 75, and 300 seconds for instances up to 50 nodes, up to 100 nodes, and with over 100 nodes, respectively. Tables 4.4 and 4.5 show the results obtained by our algorithm. For each instance and h_i level, the following data is provided: the BKS, OBS, average cost (AVG) and standard deviation (SD), the number of non-serviced BH customers, CPU time (in seconds), and the percentage gap. Results obtained for small instances (22, 23, 30 and 33 nodes) are the same as the optimal ones (Table 4.2) for 46 out of 48 instances. Besides, the times employed by our approach are much smaller than the ones employed by the exact method. More relevant results are obtained when the number of non-serviced BH customers is greater than zero. Here, 45 solutions show this effect. In 43 out of these (93%), the gap is negative, meaning that, by allowing collections to be optional, it is possible to generate solutions with a lower total cost than when all BH customers need to be visited. Regarding the variance of the results, the columns *AVG (SD)* show a relatively small dispersion around the average cost, which allows us to illustrate the robustness of our methodology.

Based on Tables 4.4 and 4.5, Figures 4.6 and 4.7 present a boxplot of the percentage gaps between OBS and the BKS, as well as of the number of non-collected RTIs for each penalization level. Notice that, for smaller penalization values, the solutions present a significant reduction in transportation costs, achieving a reduction of about 25%. Analyzing Figure 4.7, one can observe the following: as the penalization value decreases, the number of uncollected RTIs of the solution increases. Therefore, relaxing the constraint of having to visit all customers can bring significant savings in the routing cost and, consequently, provide better overall distribution plans for decision-makers.

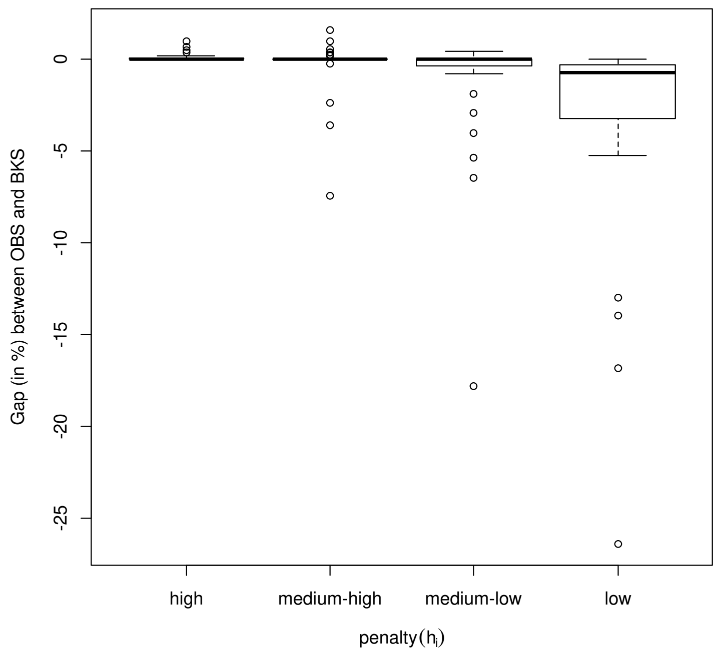


Figure 4.6: Gap between our OBS and the BKS for each penalization scenario (h_i).

There is not a general relation between the non-serviced BH customers and the gap. However, in those 5 out of 44 cases in which a gap is positive, there are only 1 or 2 non-serviced BH customers. That is, a higher quantity of non-serviced BH customers usually

Table 4.4: Found solutions for instances with small demands and penalty costs.

Instance	*BKS*	$h_i = 0.66$ (*high*)					$h_i = 0.33$ (*medium-high*)					$h_i = 0.16$ (*medium-low*)					$h_i = 0.06$ (*low*)				
		OBS	**AVG (SD)**	**NSBH**	**Time (s)**	**Gap**	**OBS**	**AVG (SD)**	**NSBH**	**Time (s)**	**Gap**	**OBS**	**AVG (SD)**	**NSBH**	**Time (s)**	**Gap**	**OBS**	**AVG (SD)**	**NSBH**	**Time (s)**	**Gap**
eil22_50	371.0	371.0	371.0 (0.0)	0	0.0	0.0%	371.0	372.3 (7.3)	0	0.0	0.0%	371.0	371.0 (0.0)	0	0.0	0.0%	371.0	371.0 (0.0)	0	0.0	0.0%
eil22_66	366.0	366.0	369.7 (6.3)	0	0.0	0.0%	366.0	368.3 (5.3)	0	0.0	0.0%	366.0	366.5 (2.6)	0	0.0	0.0%	366.0	366.0 (0.0)	0	0.0	0.0%
eil22_80	375.0	375.0	376.7 (2.9)	0	0.1	0.0%	375.0	376.7 (3.5)	0	0.1	0.0%	375.0	376.8 (3.3)	0	0.0	0.0%	375.0	375.3 (1.3)	0	0.0	0.0%
eil23_50	682.0	682.0	682.0 (0.0)	0	0.0	0.0%	657.8	658.4 (2.4)	1	0.0	-3.6%	645.4	661.1 (8.1)	1	8.5	-5.4%	567.2	567.2 (0.0)	7	0.1	-16.8%
eil23_66	649.0	649.0	651.7 (3.4)	0	0.0	0.0%	601.1	611.7 (6.8)	3	2.8	-7.4%	533.6	537.0 (1.0)	4	0.7	-17.8%	477.6	482.0 (8.0)	4	0.0	-26.4%
eil23_80	623.0	623.0	623.2 (1.3)	0	0.0	0.0%	608.5	608.5 (0.0)	1	0.0	-2.3%	583.0	583.6 (0.3)	1	0.0	-6.4%	542.0	542.0 (0.0)	3	0.1	-13.0%
eil30_50	501.0	501.0	501.0 (0.2)	0	0.0	0.0%	501.0	501.0 (0.0)	0	0.0	0.0%	481.0	482.0 (0.5)	2	0.4	-4.0%	431.0	431.0 (0.0)	3	0.0	-14.0%
eil30_66	537.0	537.0	537.8 (2.3)	0	0.0	0.0%	537.0	537.6 (1.7)	0	0.0	0.0%	527.0	527.6 (0.9)	1	0.0	-1.9%	517.0	517.0 (0.0)	1	0.0	-3.7%
eil30_80	514.0	514.0	514.2 (0.8)	0	0.1	0.0%	514.0	514.6 (1.8)	0	0.0	0.0%	514.0	514.2 (1.3)	0	0.0	0.0%	503.0	503.0 (0.0)	1	0.0	-2.1%
eil33_50	738.0	738.0	739.0 (4.8)	0	0.1	0.0%	738.0	738.1 (0.3)	0	0.0	0.0%	738.0	738.7 (1.2)	0	0.0	0.0%	738.0	738.1 (0.4)	0	0.0	0.0%
eil33_66	750.0	750.0	752.1 (5.2)	0	0.0	0.0%	750.0	750.1 (0.4)	0	0.0	0.0%	750.0	750.2 (1.1)	0	0.0	0.0%	750.0	750.0 (0.0)	0	0.0	0.0%
eil33_80	736.0	736.0	745.3 (5.8)	0	7.9	0.0%	736.0	744.2 (7.3)	0	0.7	0.0%	736.0	740.9 (4.2)	0	0.2	0.0%	736.0	737.8 (3.6)	0	0.0	0.0%
Average				**0**	**0.7**	**0.0%**			**0**	**0.3**	**-1.1%**			**1**	**0.8**	**-3.0%**			**2**	**0.0**	**-6.3%**

Table 4.5: Found solutions for instances with large demands and penalty costs.

Instance	*BKS*	$h_i = 12.40$ (***high***)					$h_i = 6.20$ (***medium-high***)					$h_i = 3.10$ (***medium-low***)					$h_i = 1.24$ (***low***)				
		OBS	**AVG (SD)**	**NSBH**	**Time (s)**	**Gap**	**OBS**	**AVG (SD)**	**NSBH**	**Time (s)**	**Gap**	**OBS**	**AVG (SD)**	**NSBH**	**Time (s)**	**Gap**	**OBS**	**AVG (SD)**	**NSBH**	**Time (s)**	**Gap**
eil51_50	559.0	559.0	560.0 (3.0)	0	0.2	0.0%	559.0	559.4 (2.2)	0	0.1	0.0%	559.0	560.1 (2.8)	0	0.2	0.0%	541.2	549.3 (4.5)	4	72.3	-3.2%
eil51_66	548.0	548.0	557.1 (22.1)	0	1.2	0.0%	548.0	550.6 (3.5)	0	0.7	0.0%	548.0	548.3 (0.9)	0	0.0	0.0%	542.4	542.6 (1.1)	2	0.1	-1.0%
eil51_80	565.0	565.0	578.9 (19.2)	0	2.4	0.0%	565.0	575.2 (5.6)	0	0.5	0.0%	548.7	557.4 (4.1)	1	0.2	-2.9%	535.7	545.0 (3.6)	1	1.0	-5.2%
eilA76_50	739.0	739.0	744.4 (6.4)	0	2.0	0.0%	739.0	745.6 (15.2)	0	0.6	0.0%	736.1	742.1 (5.7)	1	0.7	-0.4%	734.2	737.7 (3.6)	1	0.2	-0.6%
eilA76_66	768.0	771.0	785.2 (13.7)	0	14.4	0.4%	771.0	783.6 (9.9)	0	9.8	0.4%	767.1	776.6 (10.0)	1	4.1	-0.1%	765.2	770.1 (5.0)	1	4.6	-0.4%
eilA76_80	781.0	789.0	812.3 (13.8)	0	41.7	1.0%	781.0	807.1 (15.5)	0	17.3	0.0%	781.0	801.3 (20.7)	0	19.1	0.0%	775.7	796.0 (20.1)	1	17.6	-0.7%
eilB76_50	801.0	801.0	820.6 (10.7)	0	11.8	0.0%	801.0	817.1 (11.2)	0	2.1	0.0%	798.1	824.1 (30.0)	1	1.8	-0.4%	796.2	811.4 (8.3)	1	2.1	-0.6%
eilB76_66	873.0	873.0	896.3 (16.2)	0	56.6	0.0%	873.0	889.2 (11.6)	0	21.4	0.0%	872.1	889.2 (14.1)	1	4.2	-0.1%	870.2	877.5 (10.7)	1	0.8	-0.3%
eilB76_80	919.0	920.0	938.3 (12.1)	0	60.2	0.1%	919.0	932.7 (7.6)	0	7.0	0.0%	919.0	931.9 (8.2)	0	5.4	0.0%	913.7	929.9 (6.8)	1	19.7	-0.6%
eilC76_50	713.0	713.0	715.9 (4.8)	0	3.6	0.0%	713.0	714.8 (2.8)	0	0.5	0.0%	713.0	715.5 (3.7)	0	3.3	0.0%	711.2	713.3 (2.8)	1	0.3	-0.3%
eilC76_66	734.0	735.0	754.8 (14.2)	0	7.9	0.1%	737.0	749.1 (8.0)	0	74.3	0.4%	733.1	744.2 (6.1)	1	33.0	-0.1%	731.2	737.8 (5.1)	1	0.4	-0.4%
eilC76_80	733.0	738.0	749.4 (9.0)	0	32.8	0.7%	736.0	744.2 (6.3)	0	2.5	0.4%	736.0	743.5 (4.8)	0	2.3	0.4%	729.1	737.0 (4.6)	2	16.3	-0.5%
eilD76_50	690.0	690.0	699.9 (6.8)	0	2.7	0.0%	690.0	697.6 (7.0)	0	1.2	0.0%	690.0	695.9 (6.9)	0	0.4	0.0%	689.2	697.3 (7.2)	1	0.3	-0.1%
eilD76_66	715.0	715.0	725.1 (7.9)	0	23.2	0.0%	715.0	720.9 (8.9)	0	5.1	0.0%	715.0	721.6 (11.9)	0	0.4	0.0%	715.0	719.0 (5.8)	0	0.6	0.0%
eilD76_80	694.0	695.0	709.6 (8.1)	0	29.3	0.1%	698.0	703.6 (3.2)	0	0.9	0.6%	695.0	704.3 (5.1)	0	13.5	0.1%	689.2	695.8 (2.6)	3	63.9	-0.7%
eilA101_50	831.0	831.0	847.8 (8.1)	0	83.2	0.0%	829.2	842.5 (4.5)	1	147.9	-0.2%	824.4	844.3 (8.1)	2	87.3	-0.8%	790.9	808.4 (12.6)	10	140.7	-4.8%
eilA101_66	846.0	848.0	871.4 (12.2)	0	246.6	0.2%	846.0	856.9 (13.9)	0	10.4	0.0%	846.0	849.0 (4.3)	0	0.5	0.0%	830.0	839.8 (10.4)	6	32.7	-1.9%
eilA101_80	856.0	856.0	888.0 (9.4)	0	242.4	0.0%	870.0	883.4 (9.5)	0	4.8	1.6%	860.0	876.8 (7.8)	0	222.4	0.5%	838.7	847.6 (6.8)	7	79.2	-2.0%
eilB101_50	923.0	925.0	936.6 (8.3)	0	108.3	0.2%	925.0	941.8 (14.1)	0	90.9	0.2%	921.1	948.0 (32.5)	1	20.6	-0.2%	879.8	895.4 (10.8)	8	20.4	-4.7%
eilB101_66	982.0	987.0	1025.0 (18.9)	0	286.0	0.5%	992.0	1015.3 (15.0)	0	119.4	1.0%	984.3	1008.2 (16.1)	1	138.7	0.2%	963.8	985.0 (15.6)	4	256.7	-1.9%
eilB101_80	1008.0	1008.0	1043.4 (16.1)	0	268.0	0.0%	1011.0	1048.0 (22.4)	0	238.6	0.3%	1012.8	1030.4 (10.6)	2	64.5	0.5%	986.0	997.1 (8.3)	6	278.2	-2.2%
Average				**0**	**72.6**	**0.2%**			**0**	**36.0**	**0.2%**			**1**	**29.6**	**-0.2%**			**3**	**48.0**	**-1.5%**

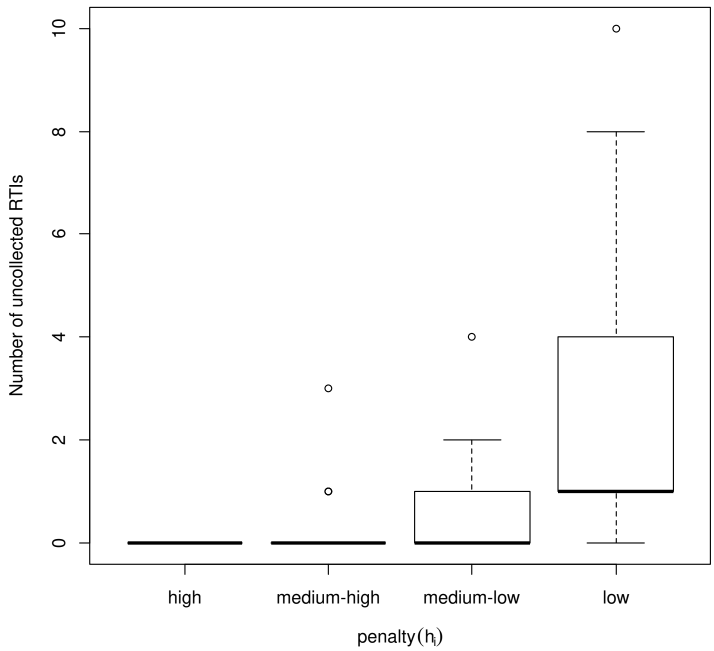


Figure 4.7: Number of non-collected RTIs of our OBS for each penalization scenario (h_i).

yields savings in total cost. This does not mean that the RTIs will never be collected, but just that they can be picked up during the next time period. Likewise, BH customers that are visited during the current period might be skipped in the next one. The unitary penalty cost is also a parameter that has an influence on cost savings. On the one hand, high values of h_i lead to the collection of most RTIs. In this case, we obtain an average gap of 0.0%, which highlights the efficiency of the proposed solving approach. On the other hand, only in a few cases a very low h_i does not yield savings, which shows the advantage of using our approach for such values of h_i.

4.2.3.3 Solving Small-Scale Instances with Heterogeneous Unitary Penalty Cost

So far, we considered the unitary penalty cost (h_i) as homogeneous for all customers in a single run. This approach allowed us to analyze appropriately the influence of h_i in both the number of non-visited customers and total costs in a single time period. However, in some real-life scenarios the unitary penalty cost might be heterogeneous, i.e.: each customer has a different h_i in the same period. This can be caused by temporal or geographic conditions –e.g., it is expected that the greater the number of days that a customer is not served, the greater h_i. Therefore, we have run some additional experiments considering this case. In these experiments, we consider that the high, medium-high, medium-low, and low levels of h_i are not scalar but equal-length intervals. Specifically, for each of the aforementioned scenarios, we set $h_i \in \{(0.4975, 0.6600), (0.3350, 0.4975), (0.1725, 0.3350), (0.0100, 0.1725)\}$. Instances *eil23*, *eil30*, and *eil33* were used to carry out the experiments. Each BH customer was assigned a random h_i uniformly distributed in the considered interval. The exact approach was selected to guarantee an optimal solution. Table 4.6 shows the obtained results.

Table 4.6: Found solutions for heterogeneous penalty levels.

Instance	h_i	Total penalty cost	Total cost	Number of non-collected RTIs	Number of non-served customers
eil 23_50	High	0.0	682.0	0	0
	Medium-high	0.0	682.0	0	0
	Medium-low	21.0	654.0	75	1
	Low	119.2	610.2	1379	6
eil 23_66	High	0.0	649.0	0	0
	Medium-high	153.2	627.2	385	3
	Medium-low	90.6	564.6	385	3
	Low	35.2	479.2	560	4
eil 23_80	High	0.0	623.0	0	0
	Medium-high	0.0	623.0	0	0
	Medium-low	48.3	607.3	150	1
	Low	24.0	583.0	150	1
eil 30_50	High	0.0	501.0	0	0
	Medium-high	0.0	501.0	0	0
	Medium-low	0.0	501.0	0	0
	Low	22.4	450.4	400	3
eil 30_66	High	0.0	537.0	0	0
	Medium-high	0.0	537.0	0	0
	Medium-low	26.2	533.2	100	1
	Low	22.0	515.0	350	3
eil 30_80	High	0.0	514.0	0	0
	Medium-high	0.0	514.0	0	0
	Medium-low	0.0	514.0	0	0
	Low	21.8	512.8	200	1
eil 33_50	High	0.0	738.0	0	0
	Medium-high	0.0	738.0	0	0
	Medium-low	0.0	738.0	0	0
	Low	5.3	715.3	500	1
eil 33_66	High	0.0	750.0	0	0
	Medium-high	0.0	750.0	0	0
	Medium-low	0.0	750.0	0	0
	Low	12.2	689.2	450	1
eil 33_80	High	0.0	736.0	0	0
	Medium-high	0.0	736.0	0	0
	Medium-low	0.0	736.0	0	0
	Low	0.0	736.0	0	0

These results are coherent with those obtained when h_i is homogeneous, i.e.: high penalties yield a null number of non-served customers (NSBH), while decreasing values of h_i generate an increasing number of non-collected RTIs. Nevertheless, a few particular outputs are not the same as those in Section 4.2.3.1. For example, instance *eil23_80* yields an NSBH of 1 when h_i is low. In contrast, this same scenario yields an NSBH of 3 when $h_i = 0.06$ (Table 4.2). These results demonstrate the high sensitivity of visiting decisions when the unitary penalty cost changes.

4.3 The LRP with Facility Sizing Decisions

The LRP is a traditional strategic-tactical-operational problem that considers a set of potential facilities and a set of customers with a known demand, whose main decisions are: *(i)* the number and location of facilities to open; *(ii)* the allocation of customers to open facilities; and *(iii)* the design of routes to serve customers from each facility using a fleet of vehicles. This means that the LRP considers jointly the FLP and the VRP. As both problems are NP-hard in nature, the LRP maintains this characteristic (Nagy and Salhi, 2007). Hence, its inherent complexity makes necessary the use of approximate solution approaches, such as heuristic or metaheuristic algorithms, to solve it efficiently, especially when dealing with large-sized instances. Therefore, the research about this problem has increased mainly during the last decade, given the recent advances in computing power.

Different versions of the LRP have been considered in the scientific literature depending on the analyzed constraints. Among them, we can find: *(i)* the capacitated version in which only the vehicle capacity is limited; and *(ii)* the capacitated version that establishes capacity constraints for both depots and vehicles. The latter variant assumes that the following parameters are known in advance: the facility opening cost, the traveling cost between two nodes, the demand of each customer, the capacity of each vehicle, and the capacity (size) of each open facility. Nevertheless, this latter is traditionally assumed as a fixed parameter, i.e., once a facility is open, a rigid known size is assigned. However, some real-world problems show the relevance of considering a set of available sizes to select those that fit better. Cases from different industries that employ either LRP or non-LRP approaches have considered this set. An example of the latter is shown by Tordecilla-Madera et al. (2017), who address the problem of locating a set of milk refrigeration tanks for a dairy cooperative in Colombia. Several tank sizes are found in the market, i.e., the considered problem must determine both the number and size of tanks that should be bought and their location, among other decisions. Correia and Melo (2016) state that, in applied problems, the capacity is often acquired in the market from a set of discrete sizes. Furthermore, economies of scale can be incurred when the facility size is an additional variable to model. The different available sizes are usually associated with investment activities, such as building facilities (Zhou et al., 2019), qualifying workforce (Correia and Melo, 2016), or purchasing equipment (Tordecilla-Madera et al., 2017). This means that considering facility sizing decisions is a strategy for decreasing the invested capital, if necessary, or even for reducing the operational costs by increasing the investment level, as we demonstrate in this work.

Allowing facility sizing decisions is a form of soft constraint (Juan et al., 2020b). The traditional LRP considers a rigid value for the maximum capacity of a facility, however, this constraint can be "violated" by providing multiple size alternatives and incurring an additional opening cost for a bigger size. This approach is quite common in real-life cases. Nevertheless, our approach not only allows bigger sizes but also smaller ones in order to diminish costs. Besides, considering sizing decisions increases the hardness of the problem. Therefore, we propose an approach formed by a BR version of a savings-based constructive heuristic (Grasas et al., 2017) and the ILS metaheuristic (Lourenço et al., 2019) to solve a deterministic version of the LRP with facility sizing decisions. Hence, the contributions of this section are fourfold: *(i)* to analyze a more realistic version of the capacitated LRP in which different sizes for each depot location are considered; *(ii)* to extend classical medium- and large-sized benchmark instances of the LRP in order to adapt them to the variant under study; *(iii)* to propose a competitive metaheuristic algorithm based on biased randomization techniques to deal with the LRP with facility sizing decisions; and *(iv)* to provide a numerical analysis of the results obtained by employing alternative MILP models, in terms of costs and computing times.

4.3.1 Problem Definition

The LRP can be defined on a complete, weighted, and undirected graph $G(V, A, C)$, in which V is the set of nodes (comprising the subset J of potential depot locations and subset I of customers), A is the set of arcs, and C is the cost matrix of traversing each arc. A set of unlimited homogeneous vehicles with capacity constraints (K) is available to perform the routes. Moreover, it is assumed that all vehicles are shared by all depots (i.e., no depot has a specific fleet) and each arc $a \in A$ satisfies the triangle inequality. Customer demands are deterministic and known in advance. Each customer must be serviced from the depot to which it has been allocated by a single vehicle. Figure 2.6 depicts an example of a complete LRP solution, where green houses represent the customers, red warehouses symbolize the open facilities, black and white warehouses represent the non-open facilities, and arrows symbolize the designed routes. The version studied in this section considers that the capacity of each depot is not known in advance, instead it is a decision to be made. Hence, a discrete set L of available sizes is known, from which the best alternative for each depot is selected. The following constraints must be satisfied: *(i)* the total demand of customers assigned to one depot must not exceed its capacity; *(ii)* each route begins and ends at the same depot; *(iii)* each vehicle performs at most one trip; *(iv)* each customer is served by one single vehicle (split deliveries are not allowed); and *(v)* the total demand of customers visited by one vehicle fits its capacity. The location routing problem with facility sizing decisions can be formulated as a mathematical programming model, whose sets, parameters, and variables are shown in Table 4.7.

Table 4.7: Sets, parameters, and variables of a 3-index model for the LRP with facility sizing decisions.

Sets
V = Set of nodes
K = Set of vehicles
L = Set of available sizes
I = Set of customers, $I \subset V$
J = Set of depots, $J \subset V$
A = Set of arcs, $A = V \times V = \{(m,n) : m \in V, n \in V \wedge m \neq n\}$
$\delta^+(S)$ = Set of arcs leaving S, $S \subset V$, $\delta^+(S) \subset A$
$\delta^-(S)$ = Set of arcs entering S, $S \subset V$, $\delta^-(S) \subset A$
Parameters
s_{jl} = Available size of type $l \in L$ for the depot $j \in J$
d_i = Demand of customer $i \in I$
f_j = Fixed opening cost of depot $j \in J$
o_{jl} = Variable opening cost of depot $j \in J$ with size of type $l \in L$
c_a = Cost of traversing arc $a \in A$
v = Fixed cost for using a vehicle
q = Capacity of each vehicle
M = A very large number when compared to the magnitude of the rest of the parameters
Variables
y_{jl} = Binary variable equal to 1 if depot $j \in J$ is open with size of type $l \in L$, 0 otherwise
x_{ij} = Binary variable equal to 1 if customer $i \in I$ is assigned to depot $j \in J$, 0 otherwise
w_{ak} = Binary variable equal to 1 if arc $a \in A$ is used in the route performed by vehicle $k \in K$, 0 otherwise
u_{ik} = Accumulated deliveries by vehicle $k \in K$ until customer $i \in I$

$$Minimize \sum_{j \in J} \sum_{l \in L} (f_j + o_{jl}) y_{jl} + \sum_{a \in A} \sum_{k \in K} c_a w_{ak} + \sum_{a \in \delta^+(J)} \sum_{k \in K} v w_{ak} \tag{4.5}$$

s.t.

$$\sum_{k \in K} \sum_{a \in \delta^-(i)} w_{ak} = 1, \quad \forall i \in I \tag{4.6}$$

$$\sum_{i \in I} \sum_{a \in \delta^-(i)} d_i w_{ak} \leq q, \quad \forall k \in K \tag{4.7}$$

$$\sum_{a \in \delta^+(n)} w_{ak} = \sum_{a \in \delta^-(n)} w_{ak}, \quad \forall k \in K, \forall n \in V \tag{4.8}$$

$$\sum_{a \in \delta^+(J)} w_{ak} \leq 1, \quad \forall k \in K \tag{4.9}$$

$$u_{ik} + d_h \leq u_{hk} + M(1 - w_{ak}), \quad \forall a \in \delta^+(i \in I) \cap \delta^-(h \in I), \forall k \in K \tag{4.10}$$

$$\sum_{a \in \delta^+(j)} w_{ak} + \sum_{a \in \delta^-(i)} w_{ak} \leq 1 + x_{ij}, \quad \forall i \in I, \forall j \in J, \forall k \in K \tag{4.11}$$

$$\sum_{j \in J} x_{ij} = 1, \quad \forall i \in I \tag{4.12}$$

$$\sum_{i \in I} d_i x_{ij} \leq \sum_{l \in L} s_{jl} y_{jl}, \quad \forall j \in J \tag{4.13}$$

$$\sum_{l \in L} y_{jl} \leq 1, \quad \forall j \in J \tag{4.14}$$

$$\forall\ y_{jl}, x_{ij}, w_{ak} \in \{0, 1\} \tag{4.15}$$

$$\forall\ u_{ik} \geq 0 \tag{4.16}$$

The objective function (4.5) minimizes the total costs. These are comprised by the opening cost, which can be understood as the investment capital, and the operational cost, which includes the distance-based cost and the cost of the usage of vehicles. The constraints of the model are explained next. Constraints (4.6) guarantee that each customer is served by a single route. Constraints (4.7) are associated to vehicle capacity. Constraints (4.8) and (4.9) guarantee the continuity of each route and the return of a route to the depot from which it has started. Constraints (4.10) are devoted to eliminate sub-tours. Constraints (4.11) guarantee that a customer is only assigned to a depot if there are routes serving that depot. Constraints (4.12) guarantee that a customer is assigned to only one depot. Constraints (4.13) ensure that the total demand of the customers allocated to a single depot does not exceed its assigned size. Constraints (4.14) guarantee that a single size is assigned to an open depot. Constraints (4.15) and (4.16) define the values of decision variables. This is the model employed to obtain our first set of results shown in Section 4.3.3.1. Nevertheless, different models can be formulated to represent our addressed problem. Appendix A.1.2 shows two additional models, which are thoroughly compared with the aforementioned one.

4.3.2 Solution Approach

The problem described in Section 4.3.1 is NP-hard, since it contains as special cases the VRP (single depot case), the multi-depot VRP (MDVRP) (case without location decisions), and the FLP, all of them known to be computationally hard. Hence, the LRP solution space is even much larger than the one of each individual problem, which makes prohibitive the use of exact methods to solve medium- and large-scale instances. Therefore, a metaheuristic approach is proposed. The implemented method is based on the work by Quintero-Araujo et al. (2017), who solve the LRP using a BR ILS metaheuristic. As discussed in Gruler et al. (2017a) and Gonzalez-Martin et al. (2018), these frameworks are efficient, relatively easy-to-implement, do not contain a large number of parameters (therefore avoiding time-consuming setting processes), and offer an excellent trade-off between simplicity and performance. Thus, they have also been successfully employed in solving other combinatorial optimization problems (Londoño et al., 2020; Muñoz-Villamizar et al., 2019; Guimarans et al., 2018; Ferrer et al., 2016). The work by Quintero-Araujo et al. (2017) has a fixed input parameter for the depots size, which is the traditional approach for the LRP. Our approach extends this previous work considering that this parameter is not fixed, i.e., several known sizes are provided and our approach selects those that minimize the total routing and opening costs. Figure 4.8 depicts the flowchart of our approach, which is composed of two phases. The Phase 1 (blue) selects quickly some top complete solutions and the Phase 2 (pink) intensifies the search starting from these solutions as a base.

Firstly, the Phase 1 calculates a minimum (lb) and a maximum quantity (ub) of required depots, by dividing the total demand by the maximum and the minimum available size, respectively. This estimation conforms a set of necessary depots. Then, for each number of depots in this set $(b \in \{lb, lb+1, ..., ub-1, ub\})$, the algorithm selects randomly which depots must be open. Later, the algorithm chooses randomly and feasibly the size to assign to each open depot (s_l), considering only the available discrete sizes. Since this is a random procedure, a known number of iterations is carried out. Hence, each iteration generates an MDVRP instance to be solved, after the depots number, location and size have been selected. Two main decisions must be made in this problem: *(i)* how to allocate customers to open depots, and *(ii)* how to design the routes to serve all customers. The allocation problem is solved through a BR savings heuristic, where the greedy behavior of the heuristic is relaxed (Dominguez et al., 2016b). BR techniques induce a non-uniform random behavior by using skewed probability distributions. Through this process, a deterministic heuristic is transformed into a randomized algorithm whilst preserving the logic behind the original greedy heuristic. The geometric or the triangular probability distributions are useful to guarantee this behaviour. In our algorithm we use the geometric probability distribution, which has only one parameter (β), such that $0 < \beta < 1$. This parameter controls the relative level of greediness present in the randomized behavior of our algorithm, and consequently, introduces the biased randomization process. Notice that biased randomization prevents the

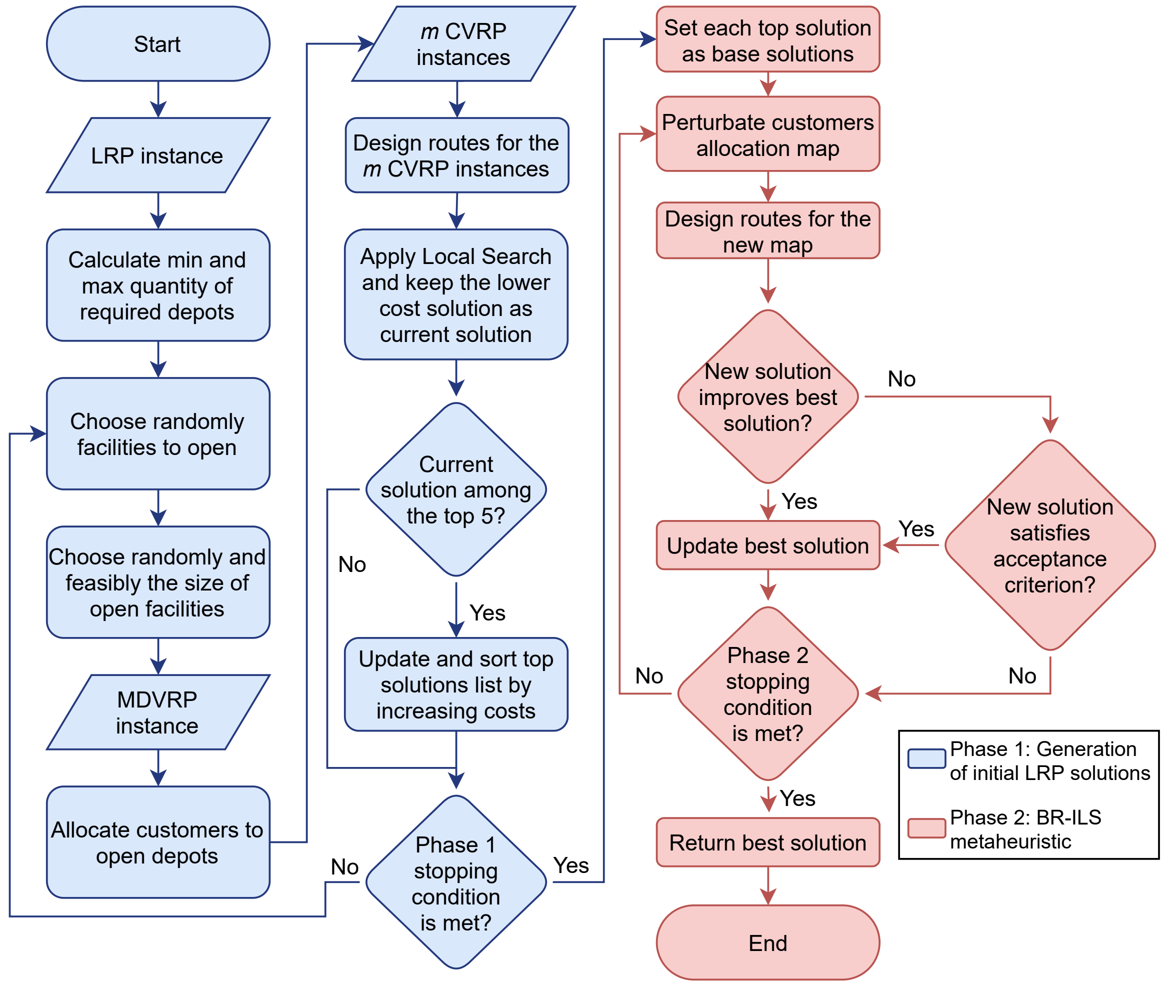


Figure 4.8: Our biased-randomized metaheuristic.

same solution from being obtained at every iteration. At the same time, using this BR procedure ensures that the perturbed solution is not far from the original solution. The savings for the allocation process are calculated following these steps:

1. Calculate the cost (Euclidean distance) c_{ij} between each customer $i \in I$ and each depot $j \in J$.
2. Find the cost (distance) c^*_{ij} between $i \in I$ and $j^* \in J$, where j^* is the depot alternative to j closer to i.
3. Calculate h_{ij}, the marginal savings of allocating the customer $i \in I$ to the depot $j \in J$, instead of the best alternative $j^* \in J$: $h_{ij} = c^*_{ij} - c_{ij}$.

This procedure generates a list of customers and savings for each depot. Positive savings mean that the customer i is closer to the depot j than to any other depot. Hence, if a depot j has several customers with positive savings, the customer with the highest savings is a priority for that depot, given their relative proximity. Then, the list of customers of each depot is sorted in descending order according to the savings. Later, each depot j is selected iteratively to perform a single customer allocation per iteration. Our algorithm selects the next depot according to a purely random policy, as long as its remaining capacity meets the demand of the next customer to allocate. Once the depot j is selected, the customer i is chosen from its SL. This selection is performed randomly, according to a biased probability distribution, i.e., the first customer in the SL has the highest probability to be selected, the second element has the second highest probability, and so on. The underlying idea is to preserve the savings criterion as a good heuristic to intensify the search process, and at the same time to provide diversification by enabling the selection of other alternatives.

This procedure generates m submaps, where each submap is formed by one depot and a subset of customers, i.e., m independent VRP instances must be solved. Then, the final decision to obtain a complete LRP solution consists in designing the delivery routes to serve all customers. Several routes can be designed for each submap, depending on the vehicle capacity. We use a BR version of a savings-based heuristic. The procedure is similar to that used for the allocation decisions. In this case, a savings value is calculated for each edge in the submap, forming a list that is sorted in descending order. Then, each route is iteratively constructed by selecting an element of the list. This selection is carried out randomly by using a biased probability distribution, in the same fashion that in the allocation procedure. Then, an LS procedure is applied to each complete solution. Four LS operators are implemented: *(i)* a customer swap inter-route operator, where two customers of different routes and allocated to the same depot are swapped; *(ii)* an inter-depot node exchange, where two customers allocated to different depots are swapped, *(iii)* a two-opt inter-route operator, where two chains of customers are interchanged between different depots; and *(iv)* a cross-exchange operator, where three non-consecutive customers from different depots are exchanged. Also, a hash table is used to evaluate each new found VRP solution. Finally, the Phase 1 is embedded into a multi-start approach (Martí et al., 2013), which means that it is repeated until the stopping condition is met, saving in memory the top solutions, i.e., those with the minimum cost.

The top solutions are the inputs of the Phase 2. This phase employs the BR ILS metaheuristic to search for better solutions by performing successive construction and reconstruction processes. Each of the top solutions starts as a base solution whose allocation map is perturbed, i.e., the open depots and their assigned sizes are not modified further. The perturbation procedure consists in selecting a set of customers and trying to reallocate them to another depot, as long as its capacity is not violated. Then, the new allocation map contains a set of m VRPs to solve through a more intensive BR savings. As well as in Phase 1, each new solution is both enhanced through the four LS operators and evaluated through a hash table. Whenever a new solution improves the best solution in terms of cost, the latter is updated. Nevertheless, if the best solution is not improved, the new solution is assessed through a Demon-like (Talbi, 2009) acceptance criterion to escape from local optima. Finally, our approach returns the best solution after the stopping condition for the Phase 2 is met.

4.3.3 Computational Experiments and Results

Both new and benchmark LRP instances have been used to test our approach. Ten small-scale instances were created: half has 8 customers and 2 alternative depots and the other half has 10 customers and 3 alternative depots. These instances were solved through an exact method using the MILP model described in Section 4.3.1. It yields optimal results useful to compare our algorithm's performance. However, benchmark instances cannot be run efficiently using this model due to their larger size. Three well-known sets of benchmark instances were considered: Akca's (Akca et al., 2009), Barreto's (Barreto et al., 2007) and Prodhon's (Belenguer et al., 2011). Each benchmark instance was slightly modified by introducing 5 known available sizes for facilities, hence, our algorithm selects a size for each open depot. All experiments were run in a PC with an Intel Core i7 processor and 16 GB RAM, and using Windows 10 as operating system.

4.3.3.1 Solving Newly Created Small-Sized Instances

LRP benchmark instances are usually medium- and large-scale and they are not useful to test our MILP model. Therefore, we created 10 small-scale instances. Most parameters (Table 4.7) were generated randomly and others were assigned deliberately:

- $I = \{1, 2, 3, ..., 8\}$ and $J = \{1, 2\}$ for 5 instances.
- $I = \{1, 2, 3, ..., 10\}$ and $J = \{1, 2, 3\}$ for 5 instances.
- $L = \{1, 2, 3, 4, 5\}$.
- $d_i \sim U(50, 150), \ \forall i \in I$.
- $f_j \sim U(30, 40), \ \forall j \in J$.
- $v \sim U(20, 30)$.
- c_a is established as the Euclidean distance between nodes whose coordinates are $cx_h \sim U(0, 200)$ and $cy_h \sim U(0, 200); \forall h \in I \cup J$.
- $q \sim U\left(0.5 \sum_i d_i, 0.7 \sum_i d_i\right)$.
- $s_{jl} \in \{500, 750, 1000, 1250, 1500\} \ \forall j \in J$. Given the importance of facility sizing in our work, these alternatives were fixed. These values also avoid infeasibilities regarding d_i.
- $o_{jl} = \frac{s_{jl}}{2s_{j3}} \cdot \frac{\sum_j f_j}{|J|}, \ \forall j \in J, \forall l \in L$. This definition keeps o_{jl} in the same order than f_j and proportional to s_{jl}.
- $M = 999999$. This number is large enough when compared to the magnitude of the rest of the parameters.

Generated instances were solved through both CPLEX and our approach. Given the random nature of our algorithm, 10 random seeds and the following parameters were used to test it:

- Iterations for MDVRP instances generation = 5000.

- Iterations for map perturbations = 350.
- Iterations for BR savings heuristic = 150.
- Iterations for splitting = 150.
- Geometric distribution parameter for biased allocation map in splitting process $(\beta_1) = 0.05 \leq \beta_1 \leq 0.80$.
- Geometric distribution parameter for BR savings heuristic $(\beta_2) = 0.07 \leq \beta_2 \leq 0.23$.

As is shown in Table 4.8, our approach reaches the optimal solution for all the tested instances, i.e., the gap between both algorithms is 0.0%. Regarding the computational time, our approach outperforms the exact algorithm for 9 out of 10 instances. Moreover, in average, our approach invests about 99% less computational time to reach the optimal solution, which shows its efficiency.

Table 4.8: Results comparison between the exact algorithm and our approach.

Instance	Total demand	Exact algorithm		Our approach			Quantity of used vehicles	Quantity of open depots	Selected sizes
		Optimal cost	CPU time (s)	Best found	CPU time (s)	Gap			
tor08x2a	767	751.23	0.88	751.23	1.25	0.0%	2	2	{500, 500}
tor08x2b	913	747.05	1.54	747.05	0.86	0.0%	3	2	{500, 1000}
tor08x2c	703	664.56	13.67	664.56	1.08	0.0%	2	2	{500, 500}
tor08x2d	764	606.30	2.96	606.30	0.12	0.0%	2	1	{1000}
tor08x2e	853	815.57	3.70	815.57	0.84	0.0%	2	2	{500, 500}
tor10x3a	1185	878.93	83.60	878.93	1.81	0.0%	2	1	{1250}
tor10x3b	1063	652.50	188.16	652.50	2.72	0.0%	2	2	{500, 750}
tor10x3c	1007	948.99	1028.08	948.99	1.88	0.0%	2	1	{1250}
tor10x3d	976	742.36	19.70	742.36	1.26	0.0%	2	2	{500, 750}
tor10x3e	1125	788.30	31.25	788.30	1.91	0.0%	2	1	{1250}
Average			137.35		1.37	0.0%			

Results regarding location-routing characteristics show the flexibility provided by our approach. For both 8- and 10-customer instances the algorithm opens 1 or 2 depots, depending on what is less costly. For example, one single depot of size 1000 is open in the instance *tor08x2d*, instead of 2 depots of size 500 as is happening in *tor08x2a*, *tor08x2c* and *tor08x2e*. The total size of both cases is the same, but cost parameters, and customers and potential depots locations determine the quantity of depots to open. For instance, in the case of Figure 4.9b, potential depots and most customers are close to each other. Optimal cost is obtained when the depot 2 (D2) is not used, and the depot 1 (D1) is open with a size of 1000 units. Using both depots would increase the opening costs and routing costs savings would be low. In all figures below, a black triangle represents an open depot, and a gray square represents a non-open depot.

An opposite case is showed in Figure 4.9a. Potential depots are far from each other and clusters of customers can be identified easily. Both depots are open with a size of 500 each. The additional cost incurred in opening a second depot is made up for routing costs savings. That is, if only one depot were open in this instance, at least one route would be very long. Results displayed in Table 4.9 support this idea. This table shows a comparison between the optimal case and a slightly modified case in which the model is forced to open a different

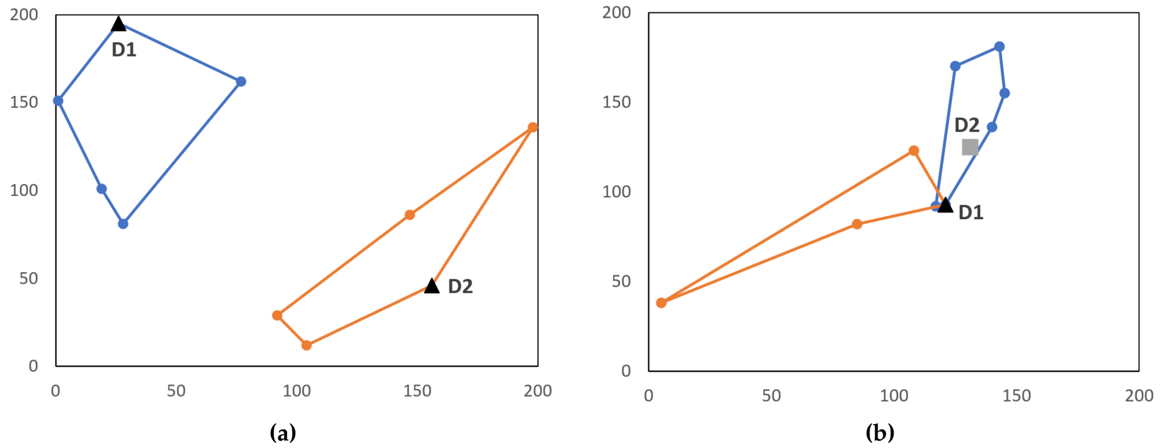


Figure 4.9: Optimal location-routing for instances *tor08x2a* (a) and *tor08x2d* (b).

number of depots, e.g., if two depots are used to obtain the optimal cost, the modified case shows the situation in which only one depot is open. Instances that reduce the quantity of open depots show higher total cost increases, given that: *(i)* routing costs always grow in this situation, and *(ii)* routing costs are always much greater than opening and vehicle costs. The highest total cost difference between modified and optimal case is 14.5% for the instance *tor08x2a*. Table 4.9 also shows these differences for each cost component. For both *tor08x2a* and *tor08x2e* instances, selected sizes of 500 and 500 are replaced by an equivalent size of 1000. However, the selected size is 750 for the instance *tor08x2c* in the modified case. Obviously, a 750-size depot is enough to meet a demand of 703, but the use of two depots of size 500 each yields cost savings by generating shorter routes. All these considerations show the flexibility of our approach. For example, consider a traditional LRP with fixed sizes of 1000. Costs would be always higher for the instance *tor08x2c*, regardless of the number of open depots.

A particular case is identified for the instance *tor08x2b*. Figure 4.10 shows the optimal location-routing plan. Total served demand is: *blue route = 189*, *orange route = 644*, and *red route = 80*. That is, D1 meets a total demand of 833, and D2, a total demand of 80. Minimal available depot sizes that satisfy feasibly such demands are 1000 and 500, respectively. Therefore, the total demand is 913 and the total capacity is 1500, which exceeds demand in about 65%. That is, one single depot of size 1000, two depots of size 500, or even two depots of sizes 500 and 750 each would be theoretically enough, although routing costs would increase. For instance, the modified case in Table 4.9 shows that routing costs are 27.7% higher when one single 1000-size depot is open. If the instance *tor08x2b* were a real-world case, a decision-maker may formulate the question if opening D2 is worth, since its used capacity is only 16% and D1 can meet the whole demand. In terms of total costs, it is really worth since the modified case in Table 4.9 shows a total cost that is 12.5% higher. Opening and vehicle costs decrease but routing costs increase. Besides, when mid- and long-term planning is considered, demand can change over time and D2 may become necessary. A final test with the MILP model was done. It shows that parameters f_j and o_{jl} must be at least 3.1 times the

Table 4.9: Comparison between opening either 1 or 2 depots.

	Output	Instance				
		tor08x2a	**tor08x2b**	**tor08x2c**	**tor08x2d**	**tor08x2e**
Optimal case	Quantity of Open Depots	2	2	2	1	2
	Selected Sizes	{500, 500}	{500, 1000}	{500, 500}	{1000}	{500, 500}
	Opening cost	91.25	104.50	81.25	56.75	78.75
	Routing cost	613.98	579.55	541.31	491.55	688.82
	Vehicle cost	46.00	63.00	42.00	58.00	48.00
	Total cost	751.23	747.05	664.56	606.30	815.57
Modified case	Quantity of Open Depots	1	1	1	2	1
	Selected Sizes	{1000}	{1000}	{750}	{500, 500}	{1000}
	Opening cost	56.25	58.00	45.19	93.75	47.75
	Difference	-38.4%	-44.5%	-44.4%	65.2%	-39.4%
	Routing cost	757.80	740.18	608.38	432.69	807.69
	Difference	23.4%	27.7%	12.4%	-12.0%	17.3%
	Vehicle cost	46.00	42.00	42.00	87.00	48.00
	Difference	0.0%	-33.3%	0.0%	50.0%	0.0%
	Total cost	860.05	840.18	695.57	613.44	903.44
	Difference	14.5%	12.5%	4.7%	1.2%	10.8%

original values to open only one depot in the optimal case. That is, f_j and o_{jl} must be at least 3.1 times higher to consider that opening a second depot is not worth.

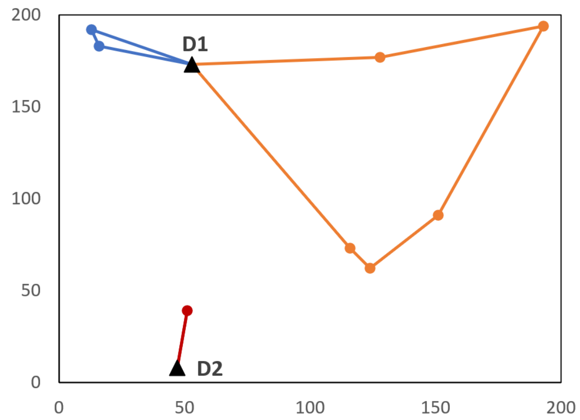


Figure 4.10: Optimal location-routing for the instance *tor08x2b*.

4.3.3.2 Solving Medium- and Large-Sized Benchmark Instances

Known LRP benchmark instances have been used to test our approach. Nevertheless, traditional algorithms using them assume that the depot size is fixed, i.e., algorithms choose if a depot is open or not, and if it is, only one alternative size is available to assign. This fact increases costs and decreases flexibility in decision making, as we will demonstrate below. Since benchmark instances have a single value for the size per potential depot, they were slightly modified to introduce new alternative sizes. 5 alternatives were considered: the original size parameter in the instance, 2 sizes smaller than the original, and 2 sizes greater than the original. If s_j is the original size for each potential depot, each available size is given by the elements in the set: $s_{jl} \in \{(1-2r)s_j, (1-r)s_j, s_j, (1+r)s_j, (1+2r)s_j\}$, where r is the variability range between available sizes, and $0.0 < r < 0.5$. For these initial experiments, $r = 0.25$. Other values of r are considered in Section 4.3.3.3.

The variable *Opening cost* (o_{jl}) is another non-considered parameter in benchmark instances. We calculate this parameter according to Equation (4.17). This equation keeps o_{jl} in the same order than f_j and allows to assign negative variable opening costs for sizes smaller than the original, positive costs for sizes greater than the original, and zero variable cost for the original size. The goal of this definition is to compare properly our results with those obtained when using the traditional benchmark instances in previous LRP papers. Finally, the number of iterations and geometric distribution's parameters are the same as those in Section 4.3.3.1.

$$o_{jl} = \frac{s_{jl} - s_j}{2s_j} \cdot \frac{\sum_j f_j}{|J|}, \quad \forall j \in J, \forall l \in L \tag{4.17}$$

Our approach results were compared with those obtained by Quintero-Araujo et al. (2019a) in the so-called *Fully cooperative scenario* (a traditional LRP). This paper was chosen since it does not show only a total cost per instance but also details about cost components. Table 4.10 shows this comparison for Akca's and Barreto's instances, and Table 4.11 shows it for Prodhon's instances. A total of 59 instances were tested. In terms of *Total costs*, our results always outperform Quintero-Araujo's, except for the Barreto's instance *Gas-32x5b*, in which we attain a slight positive gap of 0.09%. The rest of the instances show a negative gap, i.e., we obtain smaller costs by allowing the selection of size for each facility. The last two columns of Tables 4.10 and 4.11 show that our results also outperform the best-known solution (BKS) for most original instances. Small positive gaps were obtained for only 6 out of 59 instances. Hence, for Barreto's and Prodhon's instances the average gap between our results and Quintero-Araujo's are −4.32% and −7.54%, respectively, which are greater than our average gap in regard to the BKS (−3.90% and −7.12%, respectively). The average gap is the same for Akca's instances (−3.12%).

Cost components are also shown, namely: *Opening*, *Routing* and *Vehicle costs*. However, since the input parameter v is not equal to zero only in Prodhon's instances, the *Vehicle cost* is not included in Table 4.10. Our approach decreases the *Opening cost* in 44 out of 59 instances, which is a direct consequence of offering several alternative sizes. For example, the BKS for the Akca's instance *Cr30x5b-2* opens two depots of size 1000. Our approach finds that opening one depot of size 750 and one depot of size 1000 is enough, generating savings of 6.25% in the opening cost. In fact, more than half of the instances attains opening cost savings of at least 18%, with a maximum of 46.38% on the Prodhon's instance *Coord20-5-1b*.

This instance is very useful to illustrate what is happening. The total demand is 308 in this case, and the only available size is originally 300. Therefore, the traditional LRP needs to open at least 2 depots to meet such demand, with a total size of 600. Our flexible approach only requires to open a single bigger depot. Since differences between available sizes are 25% ($r = 0.25$) for the experiments in this section, the chosen size by our approach is 375, which is the minimum available size to meet a demand of 308. Notice in Table 4.11 that our approach increases the routing cost in 14.95% for this instance, although the total

Table 4.10: Results on Akca's and Barreto's instances.

Instance	Quintero-Araujo et al. (2019a)			Our approach					Gap			BKS	
	Opening Cost	Routing Cost	Total Cost	Opening Cost	Routing Cost	Total Cost	Selected sizes	CPU Time (s)	Opening Cost	Routing Cost	Total Cost	Total Cost	Gap
Cr30x5a-1	200.0	619.51	819.51	200.00	575.14	775.14	{500, 1500}	25.87	0.00%	-7.16%	-5.41%	819.51	-5.41%
Cr30x5a-2	200.0	621.45	821.45	200.00	607.28	807.28	{750, 1250}	26.30	0.00%	-2.28%	-1.73%	821.45	-1.73%
Cr30x5a-3	200.0	502.29	702.29	187.50	507.92	695.42	{750, 1000}	21.56	-6.25%	1.12%	-0.98%	702.29	-0.98%
Cr30x5b-1	200.0	680.03	880.03	225.00	623.22	848.22	{500, 500, 500}	20.71	12.50%	-8.35%	-3.61%	880.02	-3.61%
Cr30x5b-2	200.0	625.32	825.32	187.50	625.32	812.82	{750, 1000}	18.86	-6.25%	0.00%	-1.51%	825.32	-1.51%
Cr30x5b-3	200.0	684.58	884.58	187.50	684.58	872.08	{750, 1000}	19.04	-6.25%	0.00%	-1.41%	884.58	-1.41%
Cr40x5a-1	200.0	728.10	928.10	162.50	731.84	894.34	{875, 1312}	43.19	-18.75%	0.51%	-3.64%	928.10	-3.64%
Cr40x5a-2	200.0	688.42	888.42	225.00	637.26	862.26	{875, 875, 875}	39.46	12.50%	-7.43%	-2.95%	888.42	-2.94%
Cr40x5a-3	200.0	747.26	947.26	162.50	752.88	915.38	{875, 1312}	31.92	-18.75%	0.75%	-3.37%	947.26	-3.37%
Cr40x5b-1	200.0	852.04	1052.04	162.50	852.04	1014.54	{875, 1312}	38.06	-18.75%	0.00%	-3.56%	1052.04	-3.56%
Cr40x5b-2	200.0	781.54	981.54	225.00	690.57	915.57	{875, 875, 875}	34.18	12.50%	-11.64%	-6.72%	981.54	-6.72%
Cr40x5b-3	200.0	764.33	964.33	175.00	764.33	939.33	{875, 1750}	34.16	-12.50%	0.00%	-2.59%	964.33	-2.59%
							Average Akca's	**29.44**	**-4.17%**	**-2.87%**	**-3.12%**		**-3.12%**
Perl-12x2	100.0	103.98	203.98	100.00	103.98	203.98	{280}	0.21	0.00%	0.00%	0.00%	203.98	0.00%
Gas-21x5	100.0	324.90	424.90	93.75	324.90	418.65	{112, 150}	10.22	-6.25%	0.00%	-1.47%	424.90	-1.47%
Gas-22x5	50.0	535.11	585.11	43.75	535.11	578.86	{11250}	3.52	-12.50%	0.00%	-1.07%	585.10	-1.07%
Min-27x5	544.0	2518.02	3062.02	442.00	2518.02	2960.02	{450, 675}	12.87	-18.75%	0.00%	-3.33%	3062.00	-3.33%
Gas-29x5	100.0	412.10	512.10	81.25	412.10	493.35	{7500, 11250}	14.89	-18.75%	0.00%	-3.66%	512.10	-3.66%
Gas-32x5	50.0	512.22	562.22	75.00	476.20	551.20	{17500, 17500}	22.24	50.00%	-7.03%	-1.96%	562.20	-1.96%
Gas-32x5b	50.0	454.33	504.33	50.00	454.77	504.77	{35000}	6.92	0.00%	0.10%	0.09%	504.33	0.09%
Gas-36x5	50.0	425.99	475.99	37.50	420.83	458.33	{300}	27.27	-25.00%	-1.21%	-3.71%	460.37	-0.44%
Chr-50x5ba	80.0	485.62	565.62	60.00	489.59	549.59	{5000, 5000}	51.51	-25.00%	0.82%	-2.83%	565.62	-2.83%
Chr-50x5be	80.0	485.60	565.60	60.00	485.60	545.60	{5000, 5000}	47.95	-25.00%	0.00%	-3.54%	565.60	-3.54%
Perl-55x15	720.0	392.38	1112.38	510.00	417.90	927.90	{550, 687}	78.01	-29.17%	6.50%	-16.58%	1112.06	-16.56%
Chr-75x10ba	120.0	730.13	850.13	90.00	705.87	795.87	{5000, 5000, 5000}	114.21	-25.00%	-3.32%	-6.38%	844.40	-5.75%
Perl-85x7	1116.0	509.08	1625.08	790.5	538.39	1328.89	{850, 1062}	200.83	-29.17%	5.76%	-18.23%	1622.50	-18.10%
Das-88x8	83.6	272.68	356.28	75.88	260.84	336.72	{12500000, 18750000, 18750000}	312.85	-9.23%	-4.34%	-5.49%	355.78	-5.36%
Chr-100x10	80.0	760.52	840.52	60.00	759.31	819.31	{750, 750}	466.58	-25.00%	-0.16%	-2.52%	833.40	-1.69%
Min-134x8	804.0	4993.53	5797.53	804.00	4989.06	5793.06	{3000, 3000, 3000}	852.19	0.00%	-0.09%	-0.08%	5709.00	1.47%
Das-150x10	15000.0	29190.59	44190.59	11875.00	31145.92	43020.92	{37500000, 45000000}	1918.82	-20.83%	6.70%	-2.65%	43919.90	-2.05%
							Average Barreto's	**243.59**	**-12.92%**	**0.22%**	**-4.32%**		**-3.90%**

Table 4.11: Results on Prodhon's instances.

Instance	Quintero-Araujo et al. (2019a)				Our approach						Gap				BKS	
	Opening Cost	Routing Cost	Vehicle Cost	Total Cost	Opening Cost	Routing Cost	Vehicle Cost	Total Cost	Selected sizes	CPU Time (s)	Opening Cost	Routing Cost	Vehicle Cost	Total Cost	Total Cost	Gap
Coord20-5-1	25549	24472	5000	55021	20250	25929	5000	51179	{140, 210}	7.06	-20.74%	5.95%	0.00%	-6.98%	54793	-6.60%
Coord20-5-1b	15497	20607	3000	39104	8310	23688	3000	34998	{375}	8.10	-46.38%	14.95%	0.00%	-10.50%	39104	-10.50%
Coord20-5-2	24196	19712	5000	48908	16940	21499	5000	43439	{140, 210}	7.21	-29.99%	9.07%	0.00%	-11.18%	48908	-11.18%
Coord20-5-2b	13911	20631	3000	37542	9322	21081	3000	33403	{375}	5.43	-32.99%	2.18%	0.00%	-11.02%	37542	-11.02%
Coord50-5-1	25442	52822	12000	90264	21780	52669	12000	86449	{262, 262, 315}	45.30	-14.39%	-0.29%	0.00%	-4.23%	90111	-4.06%
Coord50-5-1b	15385	41908	6000	63293	15385	41180	6000	62565	{262, 525}	55.64	0.00%	-1.74%	0.00%	-1.15%	63242	-1.07%
Coord50-5-2	29319	46979	12000	88298	19420	48690	12000	80110	{262, 525}	34.44	-33.76%	3.64%	0.00%	-9.27%	88298	-9.27%
Coord50-5-2b	29319	32314	6000	67633	19420	31494	6000	56914	{262, 525}	36.04	-33.76%	-2.54%	0.00%	-15.85%	67308	-15.44%
Coord50-5-2BIS	19785	52871	12000	84656	15964	50982	12000	78946	{437, 437}	34.02	-19.31%	-3.57%	0.00%	-6.75%	84055	-6.08%
Coord50-5-2bBIS	18763	27120	6000	51883	14929	22437	6000	43366	{450, 450}	30.33	-20.44%	-17.27%	0.00%	-16.42%	51822	-16.32%
Coord50-5-3	18961	55307	12000	86268	18961	55114	12000	86075	{315, 525}	59.69	0.00%	-0.35%	0.00%	-0.22%	86203	-0.15%
Coord50-5-3b	18961	37021	6000	61982	10711	44273	6000	60984	{210, 630}	60.78	-43.51%	19.59%	0.00%	-1.61%	61830	-1.37%
Coord100-5-1	132890	120083	24000	276973	96994	128451	24000	249445	{770, 875}	361.22	-27.01%	6.97%	0.00%	-9.94%	274814	-9.23%
Coord100-5-1b	132890	71008	12000	215898	96994	74036	11000	182030	{770, 875}	526.38	-27.01%	4.26%	-8.33%	-15.69%	213615	-14.79%
Coord100-5-2	102246	70248	24000	196494	102246	68384	24000	194630	{770, 840}	239.07	0.00%	-2.65%	0.00%	-0.95%	193671	0.50%
Coord100-5-2b	102246	44427	11000	157673	102246	43864	11000	157110	{770, 840}	272.53	0.00%	-1.27%	0.00%	-0.36%	157095	0.01%
Coord100-5-3	88287	89260	24000	201547	88287	89277	23000	200564	{770, 840}	378.62	0.00%	0.02%	-4.17%	-0.49%	200079	0.24%
Coord100-5-3b	88287	54002	11000	153289	88287	53205	11000	152492	{770, 840}	450.39	0.00%	-1.48%	0.00%	-0.52%	152441	0.03%
Coord100-10-1	165068	102669	26000	293737	133684	114980	24000	272664	{840, 840}	360.97	-19.01%	11.99%	-7.69%	-7.17%	287695	-5.22%
Coord100-10-1b	154942	69510	12000	236452	133684	66873	12000	212557	{840, 840}	381.92	-13.72%	-3.79%	0.00%	-10.11%	230989	-7.98%
Coord100-10-2	149586	66641	23000	239227	120190	72683	23000	215873	{735, 840}	196.20	-19.65%	9.07%	0.00%	-9.76%	243590	-11.38%
Coord100-10-2b	149586	42456	13000	205042	120190	44604	11000	175794	{735, 840}	236.33	-19.65%	5.06%	-15.38%	-14.26%	203988	-13.82%
Coord100-10-3	136123	92082	25000	253205	112079	100924	23000	236003	{735, 840}	338.44	-17.66%	9.60%	-8.00%	-6.79%	250882	-5.93%
Coord100-10-3b	136123	52008	11000	199131	112079	58267	11000	181346	{735, 840}	301.97	-17.66%	12.03%	0.00%	-8.93%	204317	-11.24%
Coord200-10-1	266151	164764	47000	477915	241539	182115	47000	470654	{455, 892, 1785}	2577.42	-9.25%	10.53%	0.00%	-1.52%	475294	-0.98%
Coord200-10-1b	253840	104306	22000	380146	196905	133469	21000	351374	{1365, 1785}	4654.19	-22.43%	27.96%	-4.55%	-7.57%	377043	-6.81%
Coord200-10-2	280370	123205	48000	451575	208501	162640	47000	418141	{1225, 1890}	717.70	-25.63%	32.01%	-2.08%	-7.40%	449006	-6.87%
Coord200-10-2b	280370	72608	23000	375978	208501	89469	22000	319970	{1225, 1890}	1146.87	-25.63%	23.22%	-4.35%	-14.90%	374280	-14.51%
Coord200-10-3	272528	156128	46000	474656	202277	202273	46000	450550	{1470, 1680}	3367.43	-25.78%	29.56%	0.00%	-5.08%	469433	-4.02%
Coord200-10-3b	234660	109891	22000	366551	193088	116413	22000	331501	{1400, 1680}	611.54	-17.72%	5.93%	0.00%	-9.56%	362653	-8.59%
									Average Prodhon's	**583.44**	**-19.44%**	**6.96%**	**-1.82%**	**-7.54%**		**-7.12%**

cost remains lower than Quintero-Araujo's in 10.50%. The explanation is the same as in Section 4.3.3.1: if one depot is open instead of two, more and longer routes must be designed, increasing routing costs and generating savings in opening costs. These savings are greater than the increase in routing costs.

Total costs decrease is not only a consequence of the reduction in opening costs. 4 out of 59 instances show an increase in these costs because of selecting bigger facilities, which results in a drop in routing costs. Moreover, 8 instances show 0.00% in opening costs savings but still routing costs decrease. The Akca's instance *Cr30x5a-1* is an example of this situation. The total demand to meet is 1662. The original non-flexible best solution is 819.51 ($Opening\ cost = 200.00$ and $Routing\ cost = 619.51$), by opening 2 depots with a size of 1000 each. Designed routes are shown in Figure 4.11a. Our approach attains the same opening cost by opening the same depots than the original LRP but assigning them sizes of 500 and 1500 for D4 and D2, respectively. Given our formula for costs calculation in Equation (4.17), a total size of $1000 + 1000$ costs the same as a total size of $500 + 1500$, but conditions may be different in real-world problems, depending on the cost structure of each company or supply chain. Regardless of this situation, assigning different depot sizes leads to design better routes, as can be seen in Figure 4.11b. Our routing cost is 575.14, since the depot D2 has now more capacity to serve some customers that are closer to it than to the depot D4. This shows the flexibility and cost-efficiency of our approach.

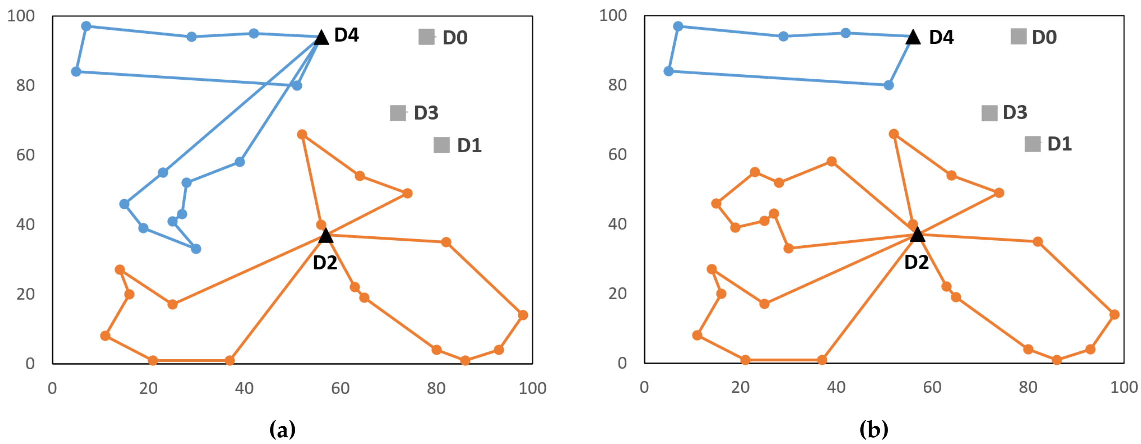


Figure 4.11: Best found solution by the non-flexible LRP (a) and our approach (b) for the Akca's instance *Cr30x5a-1*.

Prodhon's instances consider also a vehicle cost. Such consideration leads to reduce total costs not only by decreasing opening costs or traveled distance, but also by reducing the number of routes. 8 out of 30 instances show this performance, which is a direct consequence of the flexibility in facility sizing. For example, our approach creates one route less than Quintero-Araujo et al. (2019a) for the instance *Coord100-5-1b*. Our algorithm opens 2 depots with capacities of 770 and 875, respectively, whereas the non-flexible approach opens 3 depots with capacities of 700, 770, and 770, respectively. As our open facilities are bigger, only 2 depots are necessary and, therefore, the algorithm finds more flexibility to distribute the customers differently by using less vehicles. Obviously, this is also subject to the capacity of vehicles. The non-flexible approach yields an average vehicle utilization of 87.9%, which means a margin for improvement. The bigger facilities in our approach leads to a reorganization of the routes and an average vehicle utilization of 95.9%.

4.3.3.3 Sensitivity Analysis Regarding Available Sizes

So far, the variability range between available sizes remained fixed in 25%, i.e., $r = 0.25$. This section's objective is to analyze the effect that other values of r have in the obtained results. Hence, 7 values of r are considered, namely: $r \in \{0.00, 0.01, 0.05, 0.10, 0.15, 0.20, 0.25\}$. The case in which $r = 0.00$ corresponds to the non-flexible case shown by Quintero-Araujo et al. (2019a). A total of 20 Akca's, Barreto's and Prodhon's instances were selected to carry out our experiments. These instances show different number of customers and depots from each other. Regardless of the instance, the highest total cost is always obtained when $r = 0.00$. In average, the total costs show a decreasing trend when increasing r, as Figure 4.12 displays. As costs in all instances have very different scales, they were normalized to create this chart. These results indicate that providing sizes with broader variability has a positive impact in total costs, i.e., the greater the differences among input sizes, the smaller the average costs. Figure 4.12 demonstrates the advantages of considering our flexible approach, since even a range as small as 1% in available sizes is enough to yield cost savings.

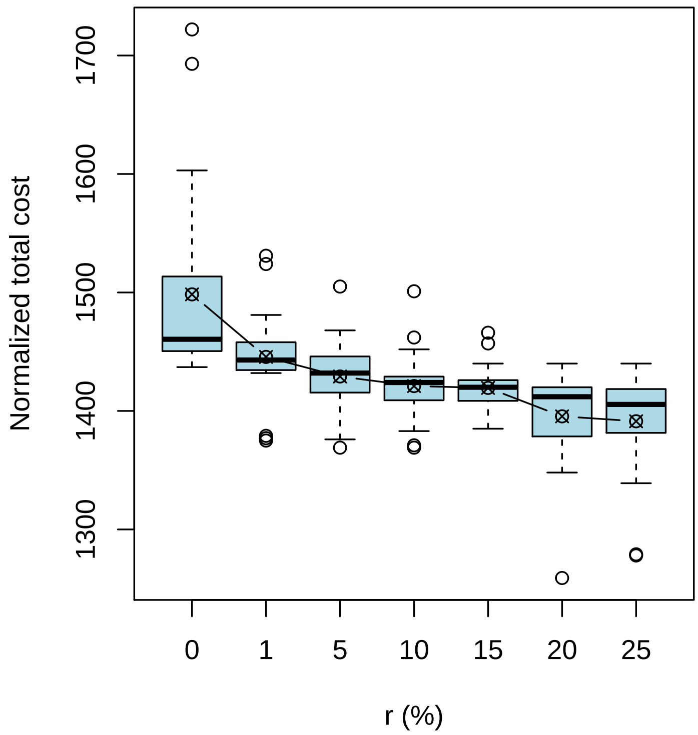


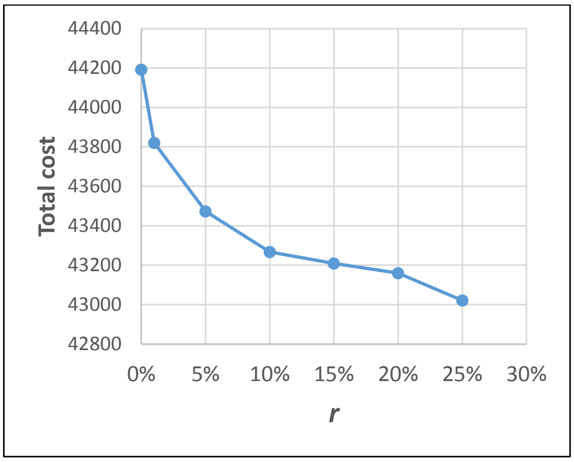
Figure 4.12: Variation in total costs in relation to r.

It is important to highlight that this is an average performance, which means that most instances show a total cost decrease when increasing r. However, some individual instances do not perform in this way. Whenever considering $r > 0$, three types of results are identified: *(i)* 12 instances show a steady decrease in total cost; *(ii)* 7 instances show a fluctuating performance; and *(iii)* 1 instance's total cost increases steadily with r. Figure 4.13 shows an example of each case. The chart (a) represents the general trend in which providing sizes with a broader variability yields smaller total costs, the chart (b) shows a performance with no clear trend, and the chart (c) shows the only instance with an opposite performance. The relation between the total demand and available sizes is the cause of such behavior. For example, the total demand in the instance *Coord20-5-2b* is 302. The available original size is 300, i.e., at least 2 depots are necessary to meet the demand when $r = 0$. However, if a size 1% bigger is available, only one depot is enough. In this case, providing still bigger sizes is redundant and, therefore, opening costs increase with r. The underlying idea is that the algorithm searches for a total capacity as close as possible to the total demand in order to minimize the total cost. Nevertheless, such as real-world cases show, s_{jl} is not usually a continuous parameter. Hence, our metaheuristic tries to find the less-costly combination of available sizes so that total demand is met. Most times, providing sizes with a bigger range helps to attain this objective, but sometimes, they cannot be combined so that the total capacity is closer to the total demand. This also explains the fluctuating performance of the instance *Gas-21x5*, as observed in Figure 4.13b.

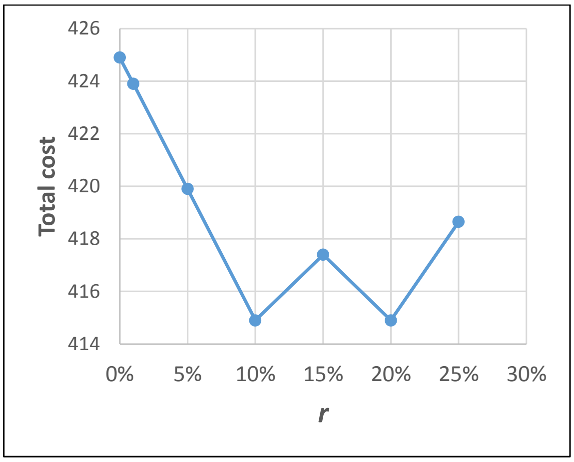
4.4 Conclusions

This chapter has presented three applications of metaheuristics in problems that have been barely studied in the literature: the DRSP, the VRPOB, and the LRP with facility sizing decisions. In all cases the proposed metaheuristics have been proved to provide high-quality solutions in comparison to: *(i)* a static version of the DRSP; and *(ii)* benchmark instances for the cases of the VRPOB and the LRPFS. Moreover, the proposed metaheuristics have been adapted to increase flexibility in the studied problems. Firstly, the DRSP is a flexible version of the static RSP since routes are dynamically adapted to changing events, such as traffic conditions. Secondly, the VRPOB allows that some BH customers are not visited, which is a flexible version of the VRPB. Finally, The LRPFS is a flexible version of the LRP since the latter considers that an unchanging size is available for each open depot. Conversely, a set of available sizes to select the one that fits better is considered in our approach. Allowing this flexibility has been proved to generate cost savings in all our addressed problems. Furthermore, these problems are more general, i.e., the traditional RSP, VRPB, and LRP are particular cases of them.

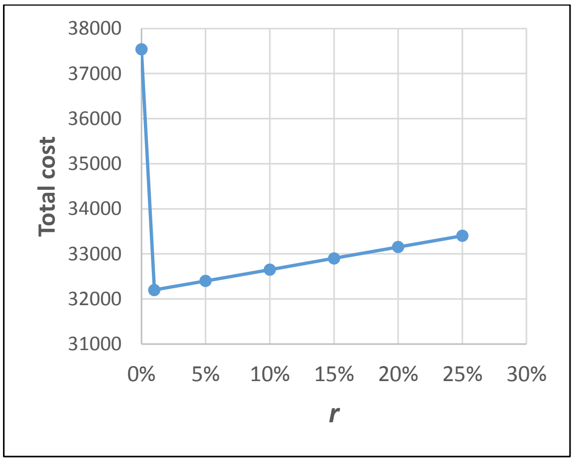
Particularly, in the case of the DRSP we propose a numerical case study in which a more traditional static scenario is compared against a dynamic one. In the former, the transportation system is optimized just at the beginning, while in the latter new data are employed to periodically re-optimize the system. The computational experiments show the benefits of our proposed dynamic approach, which clearly outperforms the traditional one in terms of costs. Furthermore, our experiments include instances with multiple sizes; hence, our algorithm has been proved to be flexible and scalable, i.e., it can be easily adapted to cope



(a)



(b)



(c)

Figure 4.13: Variation of the total cost with *r* for instances *Das-150x10* (a), *Gas-21x5* (b), and *Coord20-5-2b* (c).

with even bigger instances. Nevertheless, it is worth mentioning that we have not included the occurrence of events such as new requests, cancellations, or destination modifications. These events are present in real-life situations and, therefore, they can be included in future work.

Regarding the VRPOB, we demonstrate that savings in transportation costs are achieved, although penalty costs are incurred. These penalties can be understood as additional inventory costs for holding RTIs, which are located at the customer facilities. Two approaches are proposed to deal with this NP-hard combinatorial optimization problem: *(i)* a MILP model, which is solved for a set of traditional instances. Here, instances up to 51 nodes were solved using exact methods; and *(ii)* a BR ILS for solving larger instances. This algorithm is adapted to consider that RTIs collection is optional, subject to a penalty (inventory holding) cost. Consequently, a new parameter α is introduced, which indicates the weight given to the transport cost versus the penalty cost. After calibrating α, our analysis shows that it is possible to obtain lower aggregated (routing plus inventory) costs in scenarios where collection is not mandatory. We demonstrate that the collection decision is sensitive to the unit penalty cost h_i, i.e., our tests show that the lower h_i, the greater the number of not collected RTIs. This action yields substantial savings in total costs up to 26%. Nevertheless, some benchmark instances require visiting all customers regardless of the value of h_i. In other words, cost savings are instance-dependent, hence, decision makers in real-world supply chains should estimate h_i as accurately as possible to decrease the total aggregated cost successfully.

Finally, regarding the LRPFS, the obtained results show the great advantages of considering facility sizing decisions instead of having a fixed value as traditional approaches do. Noticeable cost savings are obtained with our approach due to: *(i)* the possibility of designing customized facilities that adjust to the current and forecasted demand in each region; and *(ii)* reallocating customers and redesigning routes by locating either larger- or smaller-size facilities. Both alternatives have been proved to decrease total costs, which are formed by opening costs (investment capital), and operational costs (routing and vehicle costs). The former alternative allows to save routing costs, although the opening cost can grow. The latter alternative may increase routing costs, but the initial investment is lower. Regardless of the size of the instance, our approach has been proved to yield very competitive results in terms of total costs. Small-, medium-, and large-sized instances have been used in our experiments. Initially, three MILP models are proposed and tested by solving optimally a few newly created small-sized instances, as well as benchmark instances whose number of nodes is smaller than 30. The same instances were solved using our BR ILS metaheuristic. Then, this approach is employed to solve both medium- and large-sized benchmark instances. They were slightly modified to consider facility sizing decisions, by providing both a set of alternative sizes and a variable cost according to each size. The experiments' results show not only that cost savings are attained after considering flexibility in facility sizes, but also that our metaheuristic is both time- and cost-efficient. Additionally, all proposed MILP models have been proved to be quite inefficient when compared with our metaheuristic approach. Finally, a sensitivity analysis is carried out, in which we study the effect of different sets of sizes in the cost. Average results show that those sets with a bigger range of difference between sizes yield smaller total costs. Nevertheless, a few instances do not follow this trend, which indicates that total costs in an LRPFS depend on the relation between demand and available sizes for each instance or real-world case.

Chapter 5

Applications of Simulation-Optimization Approaches

All parameters and variables considered in the problems addressed in Chapters 3 and 4 are deterministic. Nevertheless, uncertainty is a highly relevant feature in a myriad of real-world T&L problems. If a case study has uncertain variables, but these are not considered in the modeling phase, costs are very likely to increase (Birge and Louveaux, 2011; Panadero et al., 2020b). Hence, from this chapter on, uncertainty is considered in the studied problems. Particularly, this chapter[1] addresses an FLP where customers demands are stochastic. A hybridization between an exact MILP model and Monte Carlo simulation (MCS) is proposed to solve an urban logistics problem based on a real-world case in Dortmund, Germany.

5.1 The FLP with Stochastic Demands

Researchers have been employing simulation-optimization (SO) techniques for solving complex T&L problems for many years (Figueira and Almada-Lobo, 2014). Both exploring the behavior of logistics systems and estimating their response to various changes in their environment is the primary purpose behind the use of simulation (Crainic et al., 2018). In logistics systems, SO enables to represent and estimate different scenarios for policy changes and environmental regulations, leading to better accommodation of logistics schemes. In this context, we focus on SO models in urban logistics (UL) systems. Urban logistics has been a subject of interest for researchers during the last decades. UL is defined by Gonzalez-Feliu et al. (2014) as “the multi-disciplinary field that aims to understand, study and analyze the different organizations, logistics schemes, stakeholders and planning actions related to the improvement of the different goods transport systems in an urban zone and link them in

[1]The contents of this chapter are based on the following works:

- Rabe, M., Gonzalez-Feliu, J., Chicaiza-Vaca, J., & **Tordecilla, R.D.** (2021). Simulation-optimization approach for multi-period facility location problems with forecasted and random demands in a last-mile logistics application. *Algorithms*, 14(2), 41.
- Rabe, M., Chicaiza-Vaca, J., **Tordecilla, R.D.**, & Juan, A.A. (2020). A simulation-optimization approach for locating automated parcel lockers in urban logistics operations. *2020 Winter Simulation Conference (WSC)*, pp. 1230-1241.

a synergic way to decrease the main nuisances related to it”. Hence, UL includes different stakeholders who are seen in urban logistics, as well as a wide variety of aims, which imposes hard challenges to decision makers.

This section focuses on the usage of automated parcel locker (APL) systems, such as pack stations or locker boxes, as one of the most promising initiatives to improve the UL activities. The APL is a group of electronic lockers with variable opening codes, so that it can be used by different customers whenever it is convenient for them. APLs are located in apartment blocks, workplaces, railway stations, or near to consumers’ homes, to which parcels are delivered. The costs of APL deliveries are lower than those of home deliveries, and the risk of missed deliveries is reduced. Some studies confirm that online shoppers will use APLs more frequently in the future (Moroz and Polkowski, 2016). Despite there are limitations to the concept, many third-party logistics providers, such as DHL, InPost, Norway Post, PostDanmark, UPS, or Amazon, continue to invest in APLs to gain a competitive advantage (Moroz and Polkowski, 2016). As remarked by Verlinde et al. (2018), an APL has multiple benefits in comparison to home deliveries: economic benefits, less traffic in city centers, no double parking in front of customers’ homes, no failed home deliveries, fewer traveled kilometers and stops, off-hour deliveries, and cost reduction for e-retailers and delivery operators. Besides, the use of APLs offers environmental benefits as well, e.g., less pollutant emissions (Faulin et al., 2018). Moreover, there are also social benefits, as improved quality of life and less noise. E-customers are free to choose the pick-up time of their parcels (24/7). Also, the APL can be a focal point for the local community. However, APLs have at the same time some disadvantages as difficulties with the APL interfaces, limited payment flexibility *in situ*, limited storage possibilities, and sensitivity to crime or vandalism (Vakulenko et al., 2018).

Locating APLs is one of the critical issues related to the users’ expectations. These facilities should be located close to customers’ homes, on their way to work, or in places with a high availability of parking spaces (Iwan et al., 2016). Furthermore, Guerrero and Díaz-Ramírez (2017) point out that the APL strategy has barely discussed in the scientific literature, but is observed in practice. For example, many studies do not look at the installation costs of the APLs, their suitable locations, as well as the required capacity for seasonal peaks in e-commerce. Hence, this section addresses a case in the city of Dortmund, Germany. Its population, of about $600,000$ people, makes it the seventh largest city in Germany and the 34^{th} largest in the European Union. The considered problem is modeled as a multi-period capacitated FLP. While considering users demand that must be satisfied, our objective is to find the minimum-cost number of APLs that should be installed in every considered period inside the time horizon, as well as their locations. Multiple scenarios considering different estimates for the demands in future periods are considered and solved. Then, the performance of the associated solutions in a stochastic environment is assessed by using MCS.

5.1.1 Problem Definition

The FLP is a well-known optimization challenge where the typical goal is to find the minimum costs and location of facilities that must be open to meet customer requirements, either deterministically (Melo et al., 2009) or stochastically (De Armas et al., 2017; Pagès-Bernaus et al., 2019). In our work, facilities to open are the potential APLs in Dortmund. In general, the FLP is classified either as capacitated or uncapacitated. The former refers to the case where the facilities have a known limit to the demand they can meet. The latter is the case where the service capacity of each facility exceeds the total customers demand. A simple example of an FLP final solution is shown in Figure 2.5, where each customer (green houses) is assigned to the nearest open facility (red warehouses) via an active connection (solid lines). Additionally, black and white warehouses represent the non-open facilities, and dashed lines are inactive connections.

A multi-period capacitated FLP is considered in our work. Decisions made in a given period affect future periods over a time horizon T. In particular, as demand is expected to increase in future periods, we assume that whenever an APL is opened within a period $t \in T$, it must remain open until the end of the time horizon, i.e., for all $t' \in T : t' > t$. Similarly, third-party logistics providers indicate that a minimum percentage of $m \in (0, 1)$ of total installed capacity must be used. Therefore, with the set I of nodes representing all districts in the city, each district $i \in I$ could contain no, one, or more APLs, each with a known capacity $a_i > 0$. Similarly, each district $j \in I$ has an aggregated demand in the period $t \in T$, $d_{jt} > 0$. For two districts $i, j \in I$, the unit costs of assigning an APL located in the district i to a customer located in the district j is $c_{ij} > 0$. Similarly, the costs of opening an APL in district $i \in I$ during the period $t \in T$ is indicated as $f_{it} > 0$. In this context, the binary variable x_{ijt} takes the value 1 if customers in the district $j \in I$ are assigned to an APL in the district $i \in I$ during the period $t \in T$; otherwise, the value is 0. Similarly, the integer variable y_{it} represents the number of APLs that are open in the district $i \in I$ in the period $t \in T$. Then, our multi-period FLP can be formulated as follows.

$$\text{Minimize} \quad \sum_{i\in I}\sum_{j\in I}\sum_{t\in T} c_{ij}d_{jt}x_{ijt} + \sum_{i\in I}\sum_{t\in T} f_{it}(y_{it} - y_{it-1}) \tag{5.1}$$

s.t.

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

$$y_{it} \geq y_{it-1} \quad \forall i \in I, \forall t \in T \tag{5.3}$$

$$\sum_{j\in I} d_{jt}x_{ijt} \leq a_i y_{it} \quad \forall i \in I, \forall t \in T \tag{5.4}$$

$$\sum_{j\in I} d_{jt} \geq m \sum_{i\in I} a_i y_{it} \quad \forall t \in T \tag{5.5}$$

$$x_{ijt} \in \{0, 1\} \quad \forall i \in I, \forall j \in I, \forall t \in T \tag{5.6}$$

$$y_{it} \in \mathbb{Z}^+ \quad \forall i \in I, \forall t \in T \tag{5.7}$$

The expression (5.1) indicates the objective function that minimizes the total costs: the first term indicates the service costs of APLs, while the second represents the fixed costs of opening new APLs in the time horizon. Constraints (5.2) ensure that for each period $t \in T$ and each district $j \in I$ exactly one APL is assigned. Restrictions (5.3) ensure that once an APL is opened, it remains open until the end of the time horizon. Constraints (5.4) ensure that for the open APLs in district $i \in I$ and time period $t \in T$, the demand served by them does not exceed their capacity. Constraints (5.5) guarantee a minimum utilization percentage of the total installed capacity of APLs for each period $t \in T$. Finally, constraints (5.6) and (5.7) specify the ranges of the decision variables.

5.1.2 Solution Approach

Hybrid models play an important role in most real-world systems, since they can bring more comprehensive and efficient estimations of a reality by enhancing the synergies among different methods and giving the suitable output for decision-makers (Palacios-Argüello et al., 2018). One of the main goals of SO methods is to efficiently address both optimization and uncertainty. The possibilities of combining SO are vast and the appropriate design depends highly on the problem characteristics. Figueira and Almada-Lobo (2014) describe in detail the main classification of different SO combinations. According to their classification, we consider an analytical model enhancement approach by using simulation to improve the model results, either by refining its parameters or by extending them, e.g., considering different scenarios. Hence, initially we propose a MILP model that provides an optimal location for the APLs considering expectations on users demands. This model is solved using the CPLEX solver. Nevertheless, in real-life, the demand of each district during each period is subject to uncertainty, so it is usually modeled as a random variable. Particularly, our approach consists of the following stages (Figure 5.1): *(i)* for different scenarios, with each scenario defined by a different level of demand (e.g., lower than expected, as expected, or higher than expected), solve the associated FLP model; and *(ii)* use a MCS to evaluate the solutions obtained in the previous step when they are employed in a stochastic environment.

5.1.3 Computational Experiments and Results

We consider a case in the city of Dortmund, which is divided into 62 districts. Three demand scenarios $s \in S$ are used to feed our MILP model, where $S = \{S1, S2, S3\}$, corresponding to a low-, medium-, and high-level demand, respectively. The specific values were obtained by a system dynamics simulation model, as can be consulted in Rabe et al. (2021) and Rabe et al. (2020a). We evaluate ten APL network configurations ($k \in K$, where $K = \{1, 2, ..., 10\}$) with the demand increasing proportionally to k, based on the scenario S2. Each configuration is obtained by optimally solving the FLP model using the procedure described below. Additionally, a planning horizon of 36 months is considered.

1. Consider a uniformly distributed random demand D_{jtk} per district $j \in J$ during the period $t \in T$ for generating the configurations.

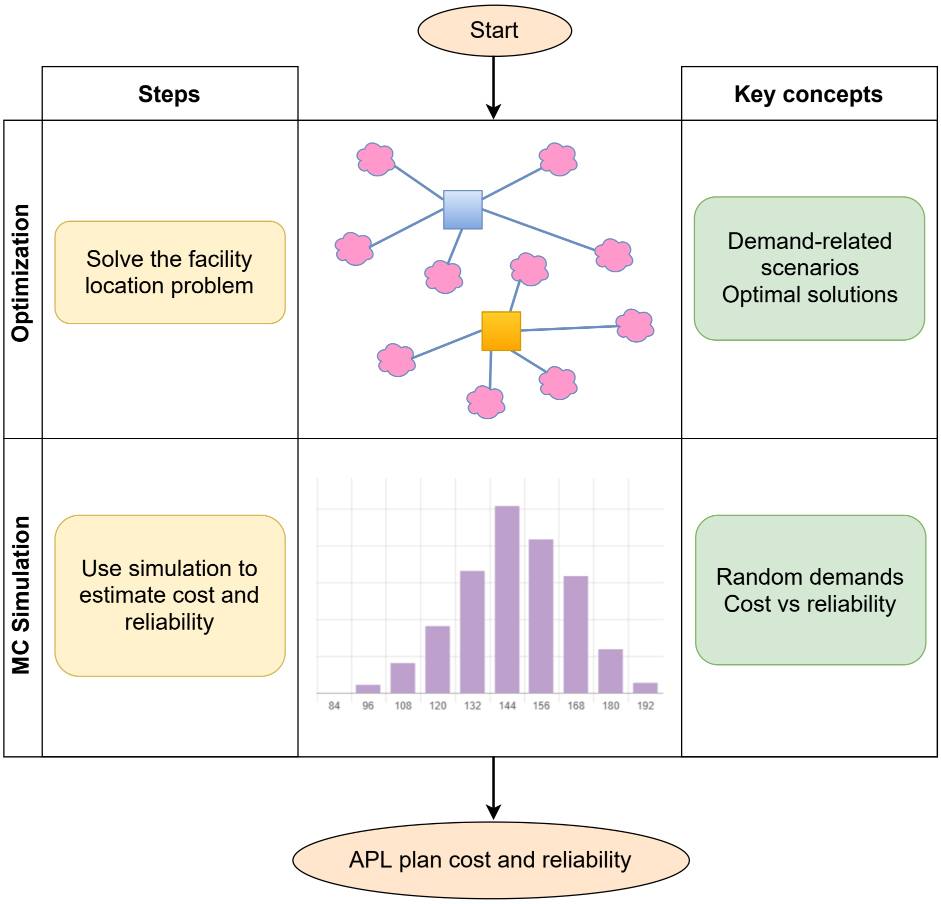


Figure 5.1: Schema of the integrated simulation-optimization approach.

2. Define $\mu_{jt} = E[D_{jtk}]$ and assume that μ_{jt} is the medium demand corresponding to the scenario $S2$.

3. Define a factor $\delta = 0.01$ to increase the size of the uniform interval as we move forward into future periods.

4. Generate the random demand using Equation (5.8). The expression $1 + \frac{k-1}{|K|-1}$ is useful to increase μ_{jt} proportionally to the value of k. In this way, we guarantee that generated configurations differ in size.

$$D_{jtk} \sim U\left(\left[1 + \frac{k-1}{|K|-1}\right](1-\delta t)\mu_{jt}, \left[1 + \frac{k-1}{|K|-1}\right](1+\delta t)\mu_{jt}\right) \quad \forall j \in I, \forall t \in T, \forall k \in K \tag{5.8}$$

The variable costs c_{ij} are proportional to the distance between each pair of districts. They were estimated using a web mapping service. The fixed costs are $f_{it} = 5500$ € for the first year and each district, and increase according to an average inflation rate of 2% per year. The capacity of each APL in a district $i \in I$ is $a_i = 6000$ units per month, and the minimum utilization percentage is $m = 40\%$. Then, our MILP model is solved with CPLEX for all ten configurations. The number of resulting open APLs per month is shown in Figure 5.2 for three out of these configurations. The lowest and highest lines represent solutions for the lowest and highest demand, respectively. The rest of the solutions are in between. As the demand μ_{jt} increases over time, the number of open APLs will behave the same regardless of the configuration. However, this consistent behavior does not extend beyond the year 1 for $k = 10$ and beyond the year 2 for $k = 1$ and $k = 5$, when the total installed APLs are sufficient to cover the total demand by the end of the planning horizon. Furthermore, there is a sharp increase in open APLs from months 11 to 12. This behavior is caused by two parameters: the annual growth of the fixed costs f_{it} drives the APLs that are open when they are less expensive, but always limited by the minimum utilization percentage m. Finally, the total number of APLs installed varies significantly from one scenario to another, for example, while 165 APLs are required for $k = 10$, only 99 APLs are installed in the configuration $k = 1$ at the end of the planning horizon.

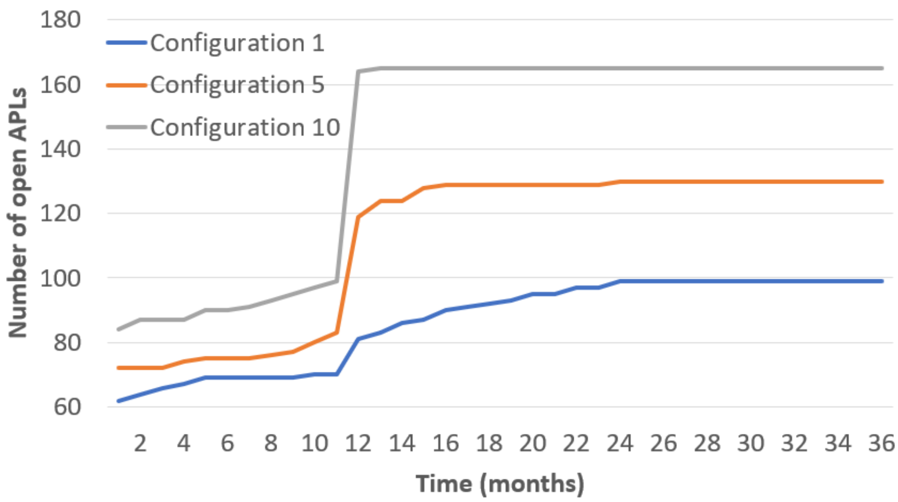


Figure 5.2: Number of total open APLs along the planning horizon for three configurations ($k = 1, 5, 10$).

Once all configurations have been generated, they are tested in a stochastic environment, assuming that the demand per district is uncertain and follows a known probability distribution. Consider a random demand D_{jts} whose mean and standard deviation are μ_{jts} and σ_{jts}, respectively, per district $j \in J$ during the period $t \in T$ for the scenario $s \in S$. Since the goal is to evaluate the performance of every configuration, they must be tested under the same demand conditions; therefore, the demand does not depend further on the configuration. Then, D_{jts} is simulated and each configuration is evaluated in terms of total costs (Equation (5.1)) and reliability. Studies on reliability in supply chains can be found in Adenso-Diaz et al. (2012) and Peng et al. (2011). We define the reliability R_{ks} of the configuration $k \in K$ for the scenario $s \in S$ as the probability that the stochastic demand of all districts in the city can be successfully satisfied, i.e.,

$$R_{ks} = \left(1 - \frac{b_{ks}}{n}\right) \cdot 100\% \quad \forall k \in K, \forall s \in S \tag{5.9}$$

where b_{ks} is the total number of simulation runs where the configuration does not cover all district demands, and n is the total number of runs. In other words, if at least one APL in a configuration is not able to cover all assigned needs, this configuration will fail. In our experiments, a total of $n = 5000$ runs are performed for each combination of scenario s and configuration k. Without losing generality, we assume that demand is independent of the customers' district, but our methodology can easily be adapted to take into account correlated demands. For the realization of the demand, three probability distributions have been tested:

1. A uniform distribution, according to Equation (5.10). In this case, $\sigma_{jts} = \frac{\sqrt{3}}{3}\delta t \mu_{jts}$.

$$D_{jts} \sim U\left([1-\delta t]\,\mu_{jts}, [1+\delta t]\,\mu_{jts}\right) \tag{5.10}$$

2. A symmetric triangular distribution, according to Equation (5.11), i.e., the mode equals μ_{jts}. To obtain conditions similar to the point 1, the lower and upper limits of this distribution are calculated assuming that the standard deviation is equal.

$$D_{jts} \sim T\left(\left[1-\sqrt{2}\delta t\right]\mu_{jts}, \mu_{jts}, \left[1+\sqrt{2}\delta t\right]\mu_{jts}\right) \tag{5.11}$$

3. A log-normal distribution, according to Equation (5.12). Again, the standard deviation is the same as in the point 1 to preserve similar conditions.

$$D_{jts} \sim Lognormal\left(\mu_{jts}, \sigma_{jts}\right) \tag{5.12}$$

Figure 5.3 shows the main results of the simulation process for each configuration. Blue, orange, and green lines represent the results from the demand for uniform, triangular, and log-normal distribution, respectively. In addition, dotted, solid, and dashed lines represent the results for the scenarios $S1$ (low demand), $S2$ (medium demand), and $S3$ (high demand), respectively. Each dot on each line represents a single configuration. In general, more expensive configurations result in higher reliability, because they include a larger number of installed APLs. When the demand follows either a uniform or a triangular distribution, the most expensive half of the configurations always achieve a 100% reliability level, regardless of the scenario. In other words, the configuration $k = 6$, with total costs of $748,660$ €, already locates a suitable number of APLs and eliminates the need to consider more expensive configurations. However, if the budget is lower, our approach offers other good alternatives for decision makers.

In general, configurations are less reliable when demand scenarios are increased. For example, configuration $k = 4$, with total costs of $661,100$ €, only achieves a reliability level of 14% under the high demand scenario and a log-normal distribution. Conversely, this configuration achieves a reliability level of 98.8% under the low demand scenario. Furthermore, the reliability is very sensitive to the probability distribution. Broadly speaking, a configuration fails if the demand is too high. Hence, configurations simulating a log-normal demand, which has no upper limit, are less reliable than those where the probability distribution is

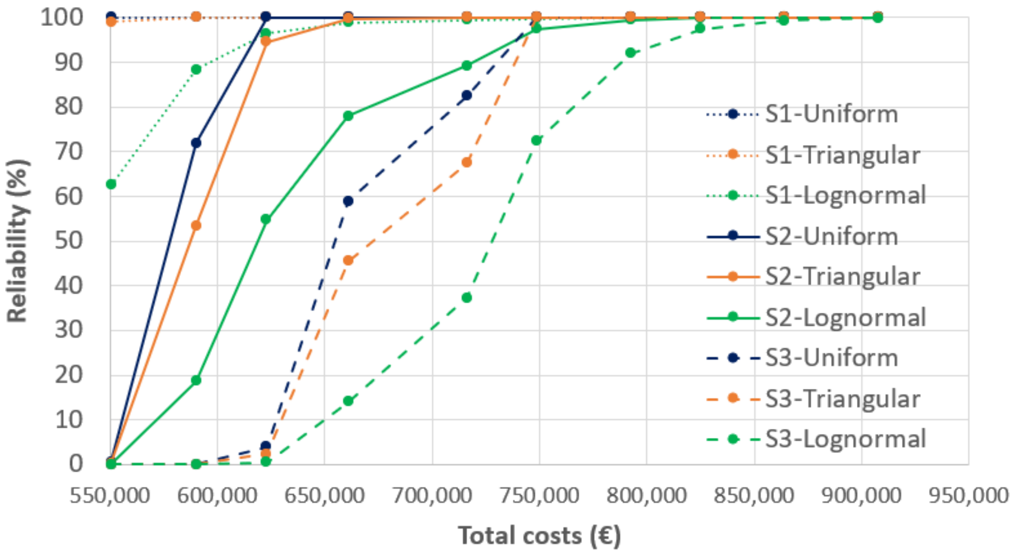


Figure 5.3: Optimal solutions evaluated in terms of costs and reliability.

either uniform or triangular (Equations (5.10) and (5.11)). This fact shows the relevance of integrating a study to determine the demand behavior in the real case.

5.2 Conclusions

With the goal of determining the optimal number and location of APL systems in a multi-period time horizon, this chapter has proposed the use of an integrated SO approach combining exact optimization and MCS. We propose this integrated model as a decision support tool for future APL implementations as a last-mile distribution scheme. The analysis is based on a real-world case study where demands are considered as random variables that evolve over time. Firstly, a multi-period facility location MILP model that provides the optimal number of APLs is formulated for a 36-month planning horizon. Different scenarios are considered and solved with exact methods to deal with the demand uncertainty. Then, the solutions associated with each scenario are sent to a MCS to estimate both their costs and reliability level.

The model provides an optimal number of APLs, taking into account the expectations of user demands. We have considered three scenarios $S1$, $S2$, and $S3$, corresponding to a low-, medium-, and high-level demand, respectively. The results for the number of deliveries (units) after 36 months show a wide range of shipments from about $277,000$ in $S1$ to nearly $400,000$ in $S3$. We used our MILP to evaluate 10 APL network configurations $(k = 1, ..., 10)$ with increasing demand in relation to each scenario. Obviously, there is a strong impact on the number of APLs that the city needs. After 36 months, the number of APLs increases from 99 in the case of the lowest demand to 165 at maximum demand. Interestingly, the number of APLs stabilizes from month 24 in all configurations. Thus, we can conclude that the effect on APLs appears linear in relation to the potential users of APL with no obvious scale effects. From a stochastic environment, we assumed that the demand per district is uncertain and follows a known probability distribution. Whenever the demand follows either a uniform or a triangular distribution, the most expensive configurations always reach a reliability level of 100.0% regardless of the scenario. The configuration $k = 6$, with total costs of $748,660$ €, already locates a suitable number of APLs. However, if the budget is lower, our approach offers other alternatives for decision makers. All in all, the work illustrates the potential of combining different simulation and optimization techniques to correctly address complex optimization problems in real urban logistics, where uncertainties must also be taken into account.

Chapter 6

Applications of Simheuristics

The previous chapter addressed a problem that was solved by a hybridization of exact methods and MCS. Nevertheless, realistic problems can become highly complex and, therefore, finding optimal solutions in short computational times can be very unlikely. Hence, this chapter shows three cases in which exact methods are not further employed. Instead, we hybridize heuristic methods with MCS. Since the basic versions of BR, multi-start, and metaheuristic approaches (Chapters 3 and 4) are not suitable to deal with stochastic variables, MCS provides the tools to do it properly. This hybridization is called "simheuristics". Hence, this chapter[1] shows three applications of simheuristics. Initially, two cases based on the COVID-19 pandemic crisis are described. Biased-randomization, multi-start approaches, and MCS are hybridized to solve an OVRP and a STOP. Finally, the LRP with facility sizing decisions shown in Section 4.3 is extended by including stochastic demands. In this case, the ILS metaheuristic is considered as well in the SO hybridization.

6.1 The Open VRP with Stochastic Service and Travel Times

The outbreak of the COVID-19 pandemic not only has caused a significant global social and economic crisis, but also has had dramatic effects on the environment. Governments and health officials around the globe have introduced mandatory policies including lockdowns, quarantines, and border closures to fight the spread of the COVID-19. While these measures have positive impacts on the environment due to the reductions in air pollution, they are most likely temporary, as pollution levels may rise again when the world recovers from the pandemic. However, consumption of personal protective equipment (PPE), such as masks and gloves, during the pandemic has already generated billions of contaminated waste. To date, COVID-19 continues to be a challenge to global public health. Saberian et al. (2021)

[1]The contents of this chapter are based on the following works:

- Peyman, M., Li, Y., **Tordecilla, R.D.**, Copado, P., Juan, A.A., & Xhafa, F. (2021). Waste collection of medical items under uncertainty using Internet of things and city open data repositories: a simheuristic approach. *2021 Winter Simulation Conference (WSC)*. Accepted conference article.
- Rabe, M., **Tordecilla, R.D.**, Martins, L.C., Chicaiza-Vaca, J., & Juan, A.A. (2021). Supporting hospital logistics during the first months of the COVID-19 crisis: a simheuristic for the stochastic team orienteering problem. *2021 Winter Simulation Conference (WSC)*. Accepted conference article.
- **Tordecilla, R.D.**, Panadero, J., Quintero-Araújo, C.L., Montoya-Torres, J.R., & Juan, A.A. (2020). A simheuristic algorithm for the location routing problem with facility sizing decisions and stochastic demands. *2020 Winter Simulation Conference (WSC)*, pp. 1265-1275.

estimate that 6.88 billion –approximately 206,470 tons– face masks are generated around the world each day. In many cities, the daily face mask usage (in terms of quantity of pieces) can be roughly estimated by simply multiplying the city population size by the acceptance rate of masks (Nzediegwu and Chang, 2020). Many of these masks are made of petroleum-based non-renewable polymers that are non-biodegradable (Dharmaraj et al., 2021), which means that they take hundreds or even thousands of years to break down in the environment. Most of the used PPE is ultimately relieved to the landfill, as well as to the oceans, contaminating the environment and affecting the fauna and flora population. Therefore, proper treatment of the PPE and, in general, of sanitary and medical waste, is an urgent need. Neglecting the seriousness of this issue may cause significant environmental and health problems.

One of the feasible approaches to tackle this problem is to assure that both the PPE and medical waste are forwarded for special processing to prevent creating more contaminating waste. Therefore, considering the case of the COVID-19 pandemic in Barcelona, specialized medical waste containers can be deployed in the most populated areas and sanitary/medical centers of the city, and small sensor devices can be installed in every container to measure its saturation level (Figure 6.1). Data regarding the waste levels of each container are sent to the open data center of the city. These data are retrieved periodically by the competent authorities, and the cargo vehicles are assigned to visit and empty the containers. By employing small sensors in the container, visiting containers that still have enough available capacity is avoided, thus reducing the overall transportation cost. Using specialized PPE waste containers brings at least two benefits:

1. Studies have already shown that the SARS-CoV-2 can stay on hard surfaces for long periods of time (Choi et al., 2021). Therefore, inappropriate management of PPE and medical waste may increase the chances of COVID-19 spread in the environment, and may lead to infection among waste workers. Using specialized waste containers can reduce the risk of exposure to the SARS-CoV-2 as PPE and medical waste will be stored and handled differently than other general waste.

2. The collected PPE and medical waste could be recycled for use in other applications, e.g., recycled concrete aggregate for pavement constructions (Saberian et al., 2021). The activity of reusing and recycling PPE and medical waste requires extensive processes of sorting out the materials, whose processing time can be saved by using specialized waste containers.

In this section, the waste collection problem is modeled as an open vehicle routing problem (OVRP). The OVRP differs from the classical VRP by considering different origin and end depots (Li et al., 2007). We employ a BR heuristic algorithm to determine the optimal route with the shortest travel time, as described by Belloso et al. (2019). This problem has been traditionally studied under deterministic conditions, where travel and service times are known and fixed. However, related activities are naturally stochastic in real-world problems. Therefore, we integrate a simulation component into the BR framework to assess the solution robustness under stochasticity, where both travel and service times are stochastic. Hence, the data about the waste levels can be consulted in the cloud, and used to feed a

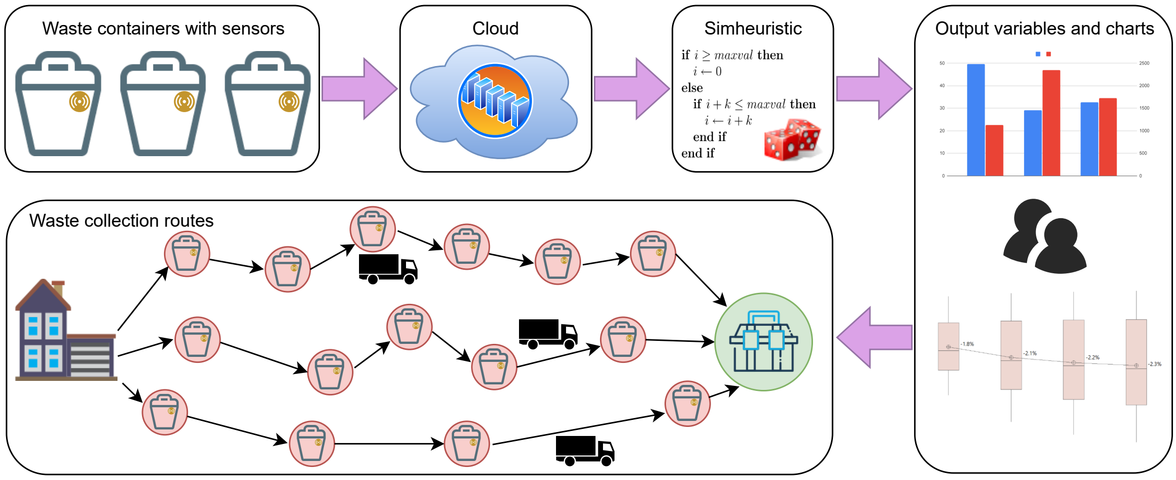


Figure 6.1: Schema of the information flow in our medical waste collection problem.

simheuristic algorithm. The simheuristic yields a set of quality solutions that are assessed by decision makers. Key performance indicators, such as cost or reliability levels can be used, as well as descriptive charts. Finally, a single routing plan is selected and employed to perform the waste collection tasks.

6.1.1 Problem Definition

The medical waste collection OVRP consists in designing a set of open routes intended to pick up medical waste. This waste has been disposed in multiple collection points across the city that must be visited. These points are connected by edges, which represent streets in cities. Each collection point has a given demand, as well as coordinates, e.g., latitude and longitude. A single vehicle is assigned to each route and visits each collection point once. We assume also that the set of vehicles is homogeneous, and that the fleet size is constant. In addition, the vehicles capacity is assumed to be unlimited because the considered waste (e.g., surgical masks, syringes, hypodermic needles, scalpel blades, etc.) requires virtually negligible space. Since loading capacity is not a constraint, each vehicle has a maximum amount of time to complete its route i.e., a time-dependent capacity arises. Service times refer to the time required to pick up the medical waste at each collection point, while travel times refer to the time invested in traversing each edge. Both travel and service times are considered to be stochastic, since they might depend upon multiple random factors (e.g., traffic conditions, weather conditions, road disruptions caused by car accidents, etc.). We assume that travel times are independent from each other. The vehicle's origin and destination points are not the same, i.e., the designed routes are open (Figure 6.1). For instance, each vehicle may depart from the firm's headquarters, visit its assigned collection points, and finish the route in a treatment facility. The Government of Catalonia has established a procedure to perform these activities in its entire territory, including Barcelona (*Health Care Waste* 2021; *Gestió Extracentre dels Residus Sanitaris* 2019). In practice, the headquarters and the treatment facility might have the same location or not, so we consider the more general scenario in which they can be located at different parts of the metropolitan area. The goal is to minimize the total time invested by the fleet of vehicles to complete the collection task. Notice, however, that whenever a route exceeds its maximum time, a mandatory stop for resting has to be performed before resuming the collection plan. Finally, although our addressed problem focuses on medical waste collection, it is worth mentioning that the collection of waste products whose size is much smaller than the vehicle capacity, e.g., batteries or cell phones, can be considered as well.

Formally speaking, the problem can be defined on a directed graph $G(N, E)$, in which N is the set of nodes, and E is the set of edges linking these nodes, such that $E \subseteq N \times N = \{(i, j) \mid i \in N, j \in N, i \neq j\}$. The set N is formed by a set I of collection points, a singleton set O representing the origin facility, and a singleton set F representing the end facility, such that $N = I \cup O \cup F$. Each collection point $i \in I$ has a service time S_i, as well as a deterministic and known demand d_i. This demand refers to the quantity of medical items to be collected. Each edge $(i, j) \in E$ is traversed in a time T_{ij}. Both S_i and T_{ij} are random variables and follow known probability distributions. These distributions are assumed to be based on historical data. A set V of uncapacitated vehicles is available to perform the routes. Each collection point must be visited once by only one vehicle. Each vehicle is assigned to only one route. Each route must start in the origin facility and finish in the end facility. The total time of each route must not exceed a given time limit t_{max}. Hence, the problem consists in designing a set of $|V|$ routes that meet the aforementioned constraints, such that the expected total time of performing all routes is minimized. Notice that no loading capacity constraints are considered since, as explained before, it is assumed that the medical items to be collected are of small size.

6.1.2 Solution Approach

Initially, a BR heuristic is proposed to tackle the deterministic version of the waste collection OVRP. This BR heuristic is later embedded into a multi-start metaheuristic framework (Martí et al., 2013). This multi-start BR heuristic is able to find feasible and promising solutions in short computing times. However, this metaheuristic is designed to solve the problem when data inputs are deterministic. In real-life applications, uncertainty is a crucial part of the decision-making process. Therefore, the multi-start BR heuristic is extended into a simheuristic algorithm to deal with a more realistic scenario under uncertainty. Simheuristics have been recently employed in solving complex stochastic optimization problems, such as stochastic inventory routing problems (Gruler et al., 2018; Gruler et al., 2020a) or stochastic facility location problems (Pagès-Bernaus et al., 2019). These algorithms combine the use of heuristics/metaheuristics with simulation in order to deal with uncertainty (Juan et al., 2018). Consequently, solutions that offer a good trade-off between the expected total time and solution reliability can be generated. In the context of this section, reliability refers to the probability that a routing plan can be implemented without failures, i.e., without routes exceeding the maximum time allowed to complete the waste collection process. In our case, we combine the proposed multi-start BR heuristic with MCS in order to address the case where both the service time of each collection point and the travel time between each pair of nodes are random variables following a given probability distribution which we assume has been obtained by fitting historical data for each collection point and each traveling edge in the network representing the city. Hence, the reliability rate determines the robustness of the deterministic solution under stochastic scenarios, i.e., whether the total travel time of the route is still acceptable/feasible under such scenarios or not, considering the maximum route travel time.

Our simheuristic is integrated into the multi-start BR heuristic, which is depicted in Figure 6.2. This algorithm works as follows. Firstly, an initial solution S^* is generated by using the BR heuristic after setting β to 1. Then, from solution S^*, travel times and service times are replaced by their stochastic counterparts, and a short simulation (small number of replications) is performed to estimate the average stochastic time and its reliability rate. The solution S^* is set as the best deterministic and stochastic solutions so far. During these iterations, the best solution S^* is kept, assessed in terms of total deterministic and stochastic times. Following Rabe et al. (2020b), in each iteration a new solution S' is generated by the BR heuristic. If the solution S' is promising, i.e., if the deterministic time of S' is less than the deterministic time of S^*, then the stochastic time of S' is computed by means of a short simulation. Later, if the stochastic time of S' outperforms the stochastic time of S^*, then S^* is replaced by S', otherwise, S' is rejected. This procedure is repeated until a stopping criteria is met. Finally, the best solution is tested in a long simulation process (large number of replications) to increase both the accuracy of the calculated stochastic time and reliability level.

6.1.3 Computational Experiments and Results

The data used to test our approach is based on a deterministic real-world case arisen during the first months of the COVID-19 pandemic in Barcelona. These data are assumed to be provided daily by sensors installed in 96 medical waste containers in the metropolitan area of the city. The origin depot and the destination treatment facility are indicated as red cylinders in Figure 6.3, and the rest of collecting locations are shown in blue points.

This case study has a four-hour time limit (t_{max}) for each route. This time limit was imposed by mobility restrictions during the pandemic lockdown. Additionally, six vehicles are available to perform the routes, and all of them must be used. A few tests considering different fleet sizes were performed. A fleet size of five vehicles or less leads to infeasible solutions, given the time limit t_{max}. To solve the problem, the original deterministic instance is transformed into a stochastic one by considering both a random travel time T_{ij} and a random service time S_i, as detailed next:

- T_{ij} follows a log-normal distribution with a minimum time limit t_{ij}^{min} along the edge (i, j). This limit represents the deterministic time in optimal travel conditions. That is, if the pure random part of T_{ij} is represented by Θ_{ij}, and $\Theta_{ij} \sim logN(\lambda, \gamma)$, where $logN(\lambda, \gamma)$ is a log-normal distribution with location parameter λ and scale parameter γ, i.e.:

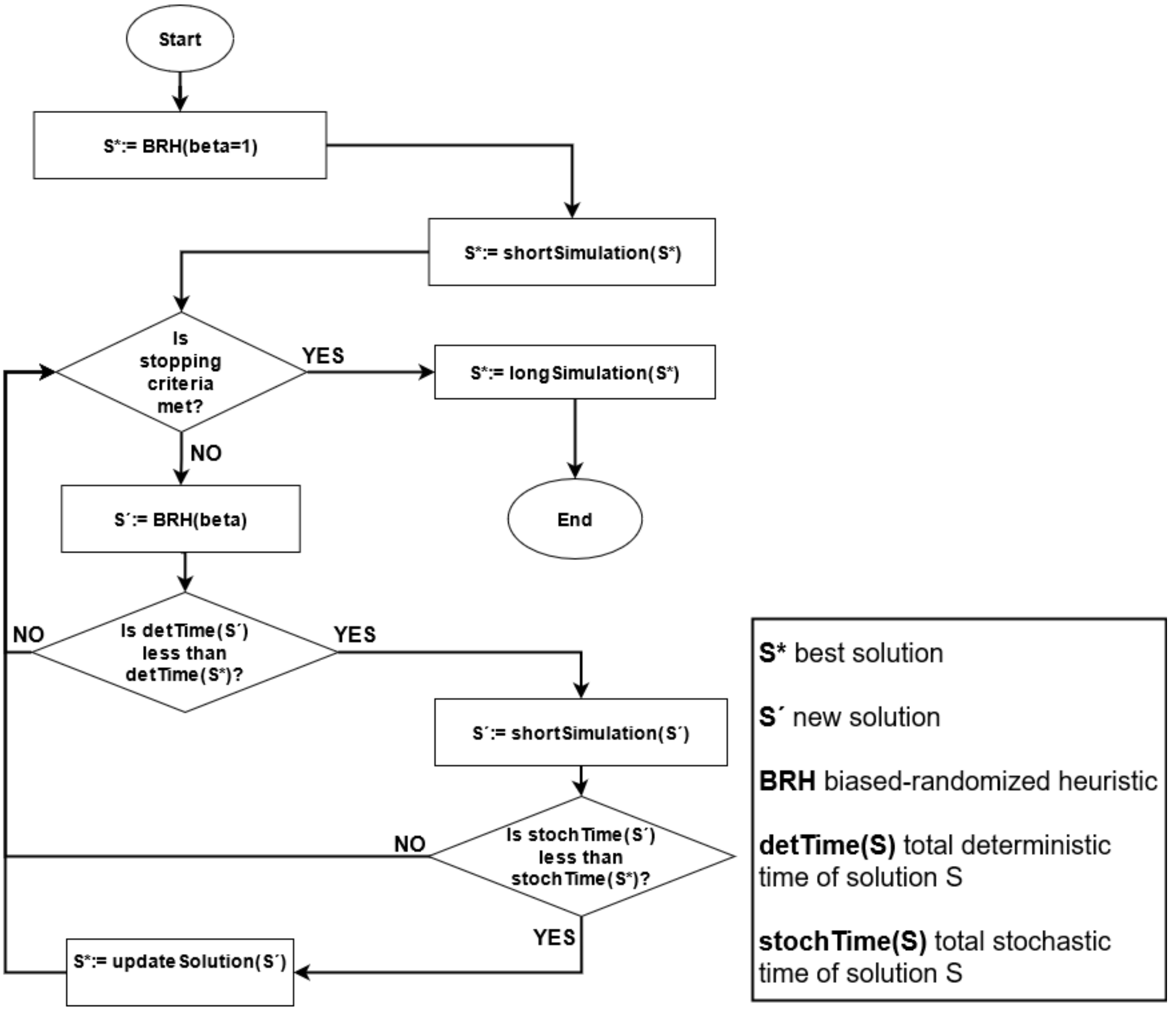


Figure 6.2: Flowchart of the simheuristic algorithm for solving the stochastic OVRP.

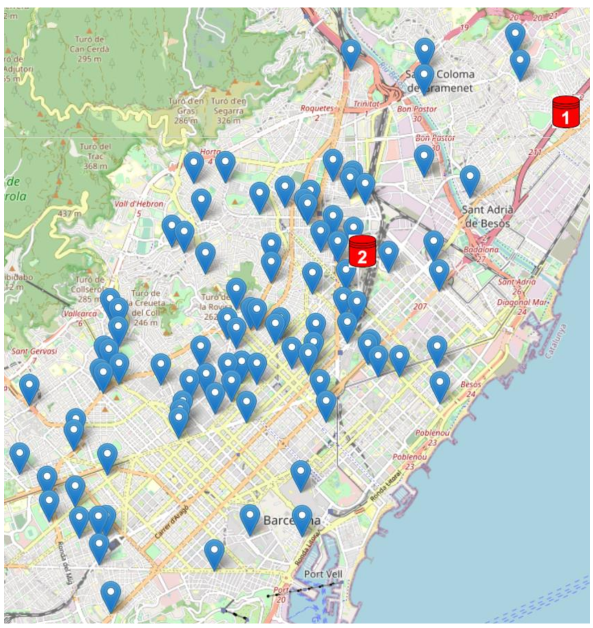


Figure 6.3: Map of the collection points and depots used for our experimental instance.

$$t_{ij} = t_{ij}^{min} + k\theta_{ij},$$

where k is a variability level, and t_{ij} and θ_{ij} are realizations of T_{ij} and Θ_{ij}, respectively. In our experiments, we consider $\lambda = 0$, $\gamma = 1$, and $k \in \{5, 6, 7, 8, 9, 10\}$. These values were selected so that diverse variability levels are tested. Hence, the experiment results lead to a more comprehensive analysis. Additionally, t_{ij}^{min} is estimated for each edge using a web mapping service.

- $S_i \sim logN(\lambda, \gamma)$. Inspired by the real-life data, in our experiments we compute λ and γ such that $E[S_i] = 420$ seconds, and $Var[S_i] = 5$, where $E[S_i]$ and $Var[S_i]$ are the expected value and the variance, respectively.

The variability level k leads to six different scenarios: very-low, low, medium-low, medium-high, high, and very-high. Regarding the algorithm parameters, we have set the β parameter used for the BR heuristic as a random value in the interval $(0.05, 0.25)$ (i.e., in each new iteration, the β value is randomly selected inside this interval, which according to our initial experiments seems to provide reasonably good solutions). The number of short simulation runs is set to 100, while the number of runs for the long simulation is set to $1,000$. A more thorough study about the number of appropriate simulation runs can be found in Rabe et al. (2020b). Finally, in case that the route stochastic time exceeds t_{max}, it is penalized with an extra time of $1,500$ seconds. This time is justified by the need of adding a mandatory stop every time a driver reaches the maximum driving time allowed.

The algorithm is implemented in Python 3 and executed in a personal computer with 16 GB of RAM and an Intel Core *i*7-8750H at 2.2 GHz. For each variability level, the maximum computational time employed is set to 60 seconds. For each uncertainty scenario, Table 6.1 shows the comparison between our best deterministic solution (OBD, which is the best solution we have been able to find for the deterministic version of the problem) and our best stochastic solution (OBS, which refers to the best solution we have found under uncertainty conditions). Each time value in this table is shown in the format *hours:minutes:seconds*, and includes both the time spent traveling throughout the route and the service time. When comparing the total time of all routes, the longest-route time (LRT) represents the maximum found time for a single route. The results show that, in the deterministic scenario, the maximum time of four hours is never exceeded, and all the six available vehicles are used. The deterministic total time is the time obtained after adding the total time of all routes in the solution when considering scenarios under absolute certainty. Obviously, this time is the same regardless of the variability level.

Once the deterministic solution is obtained, it is simulated in a stochastic environment to get the expected total time of this plan when implemented in a scenario under uncertainty conditions. The associated reliability value is also estimated, which is the estimated probability that the collection plan can be completed without any route exceeding its maximum allowed time. The OBS columns show the same key performance indicators as the OBD columns. However, these indicators are obtained when running our simheuristic algorithm.

Table 6.1: Results for a case study considering different levels of uncertainty.

Variability level	Our best deterministic solution (OBD)				Our best stochastic solution (OBS)		
	Longest-route time	Deterministic total time	Stochastic total time	Reliability	Longest-route time	Stochastic total time	Reliability
Very-low	3:58:37	22:26:05	23:03:32	15.63%	3:55:26	22:44:20	97.50%
Low	3:58:37	22:26:05	23:13:00	4.75%	3:55:26	22:48:41	91.33%
Medium-low	3:58:37	22:26:05	23:22:45	1.40%	3:55:26	22:54:05	81.04%
Medium-high	3:58:37	22:26:05	23:31:43	0.31%	3:55:26	23:01:15	64.54%
High	3:58:37	22:26:05	23:40:00	0.07%	3:55:26	23:08:31	48.70%
Very-high	3:58:37	22:26:05	23:48:40	0.02%	3:55:26	23:16:33	32.53%
Average	3:58:37	22:26:05	23:26:37	3.70%	3:55:26	22:58:54	69.27%

The results show a clear superiority of the simheuristic algorithm, since the stochastic total time is lower than the OBD's for all instances under all variability levels. Moreover, the reliability of the OBS is higher than the OBD's, i.e., our simheuristic is able to guarantee a higher probability of not exceeding the time limit of each route. Furthermore, since the time limit of four hours is never exceeded, our solution never incurs a penalty time. Notice also that the LRT of the OBS is smaller than the OBD's. Since the OBD does not consider stochastic conditions, routes' times can be really close to the four-hour limit. Nonetheless, the closer the LRT to the time limit, the greater the probability to violate this constraint in a stochastic environment. Hence, the LRT of the OBS is smaller. Finally, our results show a sharp decrease in the quality of the deterministic collection plan when increasing the variability level. Figure 6.4 displays the distribution of the stochastic total time results when running long simulations. Shown plots correspond to the very-low (VL) and the very-high (VH) variability levels, as well as to the OBD (pink charts) and the OBS (green charts). The cross circle indicates the mean value of each sample. Under stochastic scenarios, our results show an evident total time decrease when modeling our problem through a simheuristic approach, instead of considering a deterministic solution. The higher the considered variability is, the sharper the total time decrease. Furthermore, our simheuristic reduces the results variability, which can be noticed by comparing the range between the extreme points in each box plot.

6.2 The TOP with Stochastic Service and Travel Times

In Section 3.1 we showed an application of a BR heuristic to a real-world case in the context of the COVID-19 pandemic crisis in Barcelona. The necessity of immediate solutions required a quick development of solving algorithms, which means that they should be as realistic as possible, but without incurring in unnecessary complexity. Hence, due to the atypical situation derived from the pandemic crisis in March 2020, all parameters are considered deterministic in the case explained in Section 3.1, since the empty roads facilitated the motion of vehicles. However, considering that routes must be carried out mainly in urban areas, now we extend the problem addressed in that section by considering stochastic travel and service times, which is more realistic. Addressing a problem with these characteristics means that a traditional VRP (Faulin et al., 2008; Juan et al., 2009) is not appropriate to provide a feasible solution. Instead, a stochastic team orienteering problem (STOP) has proved to be a more suitable approach (Keshtkaran et al., 2016). Based on the works by

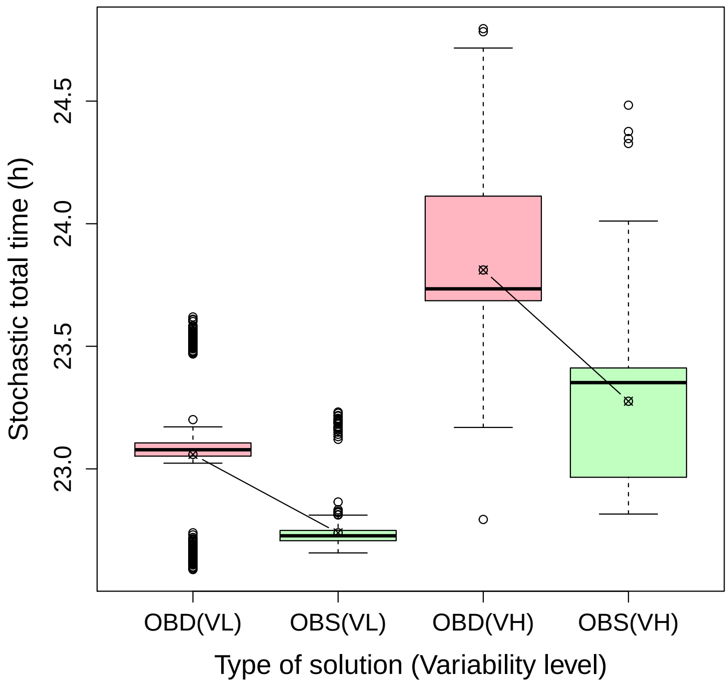


Figure 6.4: Total time results of the long simulations under different variability levels.

Panadero et al. (2017) and Chao et al. (1996), we define the STOP as a variant of the VRP with the following main characteristics: *(i)* the fleet size is limited; *(ii)* each available vehicle must meet a maximum tour length (MTL); *(iii)* characteristics *(i)* and *(ii)* imply that only a subset of the customers in the network can be visited; *(iv)* the objective is to maximize the total collected reward obtained after visiting this subset of customers; and *(v)* one or several input parameters are stochastic. Therefore, the development and application of agile algorithms became really helpful to guarantee the agility and efficiency that the route design and implementation processes required, considering as well that this is a typical NP-hard problem (Panadero et al., 2020b).

Hence, the main contributions of this section are: *(i)* to describe a real-world and deterministic case where BR algorithms were developed to support hospital logistics during the first months of the COVID-19 crisis; *(ii)* to extend the previous application into a stochastic variant, in which travel and service times are modeled as random variables; *(iii)* to propose a simheuristic approach to properly cope with the stochastic problem; and *(iv)* to provide examples of the application of agile optimization, in which algorithms are used to provide fast solutions to challenging stochastic optimization problems in the area of logistics and transportation.

6.2.1 Problem Definition

The hospital logistics case addressed in this work is modeled as a STOP. We assume that both service and travel times are stochastic, since they might depend on multiple random factors, e.g., traffic and weather conditions, unexpected delays, etc. The service time is the time spent by the driver in performing a pickup activity at each collection point. The traveling time is the time spent by a vehicle in moving from one node to another during its route. The simultaneous consideration of an MTL and stochastic service and travel times can easily lead to design infeasible routes. Allowing that some collection points are not visited is a manner of avoiding a potential infeasibility. Therefore, the STOP becomes a suitable approach to address our studied problem. Figure 3.2 displays an example of a complete solution for our problem, where some collection points are skipped. Hence, our main objective is to maximize the total reward collected by the set of vehicles, fulfilling the maximum allowed tour length (MATL).

Formally speaking, the problem can be defined on a directed graph $G(N, E)$, where N represents the set of nodes, and E represents the set of edges that link these nodes, i.e., $E \subseteq N \times N = \{(i, j) \mid i \in N, j \in N, i \neq j\}$. The set N is formed by three subsets, such that $N = I \cup O \cup F$: a set I of collection points, and the singleton sets O and F, which represent the origin and destination depots, respectively. Each collection point $i \in I$ has a deterministic reward u_i, and a stochastic service time S_i that follows a known probability distribution. The time T_{ij} spent to traverse each edge $(i, j) \in E$ is considered stochastic as well. Routes are performed by a set K of uncapacitated vehicles. Each collection point $i \in I$ must be visited just once, and each vehicle $k \in K$ is assigned to only one route. Each route starts in the origin node in O, and finishes in the destination node in F. The expected total time of each route must not exceed a given time limit t_{max}. Hence, our addressed problem consists in designing a set of $|K|$ routes that meet the aforementioned constraints, such that the total collected reward, $\sum_{i \in I} u_i x_i$, is maximized, where x_i is a binary variable that takes the value 1 if collection point $i \in I$ is visited by a vehicle $k \in K$, and it takes the value 0 otherwise.

6.2.2 Solution Approach

For solving the described STOP in the context of healthcare logistics, we propose a simheuristic approach that combines a BR multi-start metaheuristic (Belloso et al., 2019) with MCS. Our multi-start approach relies on multiple executions of a BR heuristic designed to solve the deterministic version of the TOP under the described application context. Despite offering the capability of exploring different regions of the solution space, BR algorithms are not able to consider uncertainty scenarios, such as those that can be found in real-life applications, e.g., random travel times, random processing and service times, etc. Therefore, we have extended our BR heuristic into a simheuristic algorithm to better deal with the stochastic variant of the TOP considered in this section, as Figure 6.5 depicts. Our approach starts by generating a feasible solution for the stochastic problem variant. While the deterministic solutions are generated by the BR heuristic, the solutions for the stochastic problem are achieved by replacing the deterministic travel times of the deterministic solution with the stochastic ones. These stochastic values are computed by employing a probability distribution. Hence, for each edge of the solution, multiple random observations are generated using MCS, i.e., the final stochastic value for each edge is given by the average of the multiple simulations runs. The use of multiple simulation runs for generating stochastic travel times allows us to measure the solution reliability, which refers to the solution feasibility under uncertainty conditions.

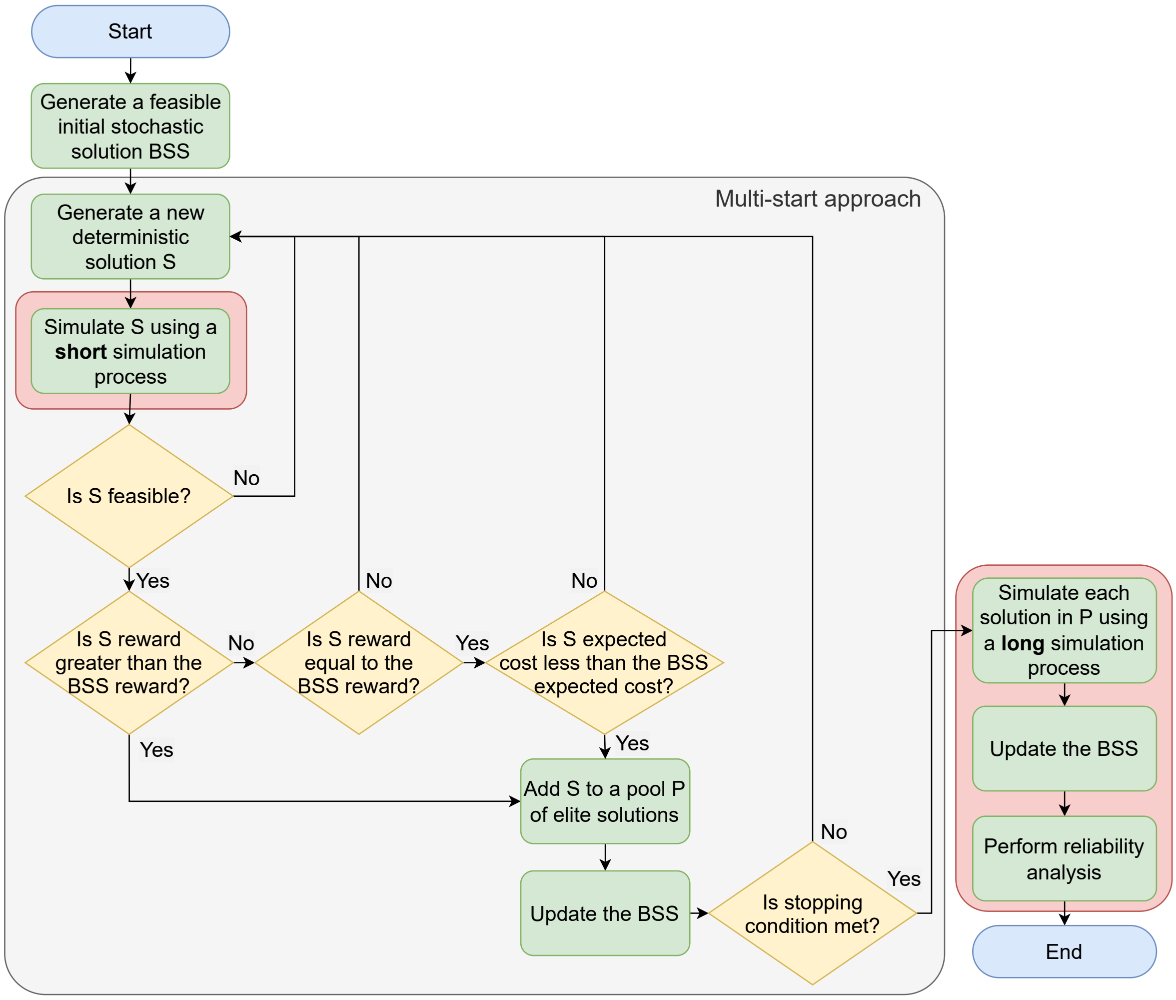


Figure 6.5: Flowchart of the simheuristic algorithm for solving the STOP.

Then, a multi-start framework starts. In this case, a solution is generated for the deterministic problem. This current solution is submitted to a simulation process only if its reward is greater or equal to the best-found solution. At this stage, a reduced number of simulation runs (the short simulation process) is performed, since this simulation stage can be time-consuming. The simulation stages are depicted inside red squares in Figure 6.5. The current solution replaces the best-found stochastic solution and becomes part of a pool of elite solutions only if: *(i)* its reward is greater than the best-found stochastic solution reward; or *(ii)* its reward is equal to the best-found stochastic solution reward, and its expected cost (e.g., travel time) is smaller. This process is repeated until a stop criterion is met. Later, a larger number of simulation runs (the long simulation process) is performed for each solution in the pool, in order to collect more reliable statistic information regarding the solutions' performance. Finally, these best-found solutions are returned.

6.2.3 Computational Experiments and Results

Based on the real-world instances presented in Section 3.1.3, we illustrate the performance of our approach using three of these instances. They correspond to real cases from March 25th, March 26th, and April 4th, 2020, respectively. They have been selected given the differences in their inputs, as shown in Table 6.2. Initially, the number of collection points represents how many volunteers are offering 3D elements that day. Next, the MATL indicates how much time is available to perform a single route. A service time is also considered. The parameter shown in Table 6.2 corresponds to the mean of a log-normal probability distribution. Hence, if S_i is a random variable representing the service time, $S_i \sim logN(\mu_i, \sigma_i^2)$, where μ_i and $\sigma_i^2 = 0.05\mu$ are the expected value and variance or the servicing time at collection point i, respectively. In the real-world case, μ_i decreased from 7 minutes in the instance *mar-25* to 4 minutes in the instance *apr-04*, which was due to the experience acquired by the drivers between both days. Finally, the number of vehicles represents the number of volunteer drivers available to collect the elements that day.

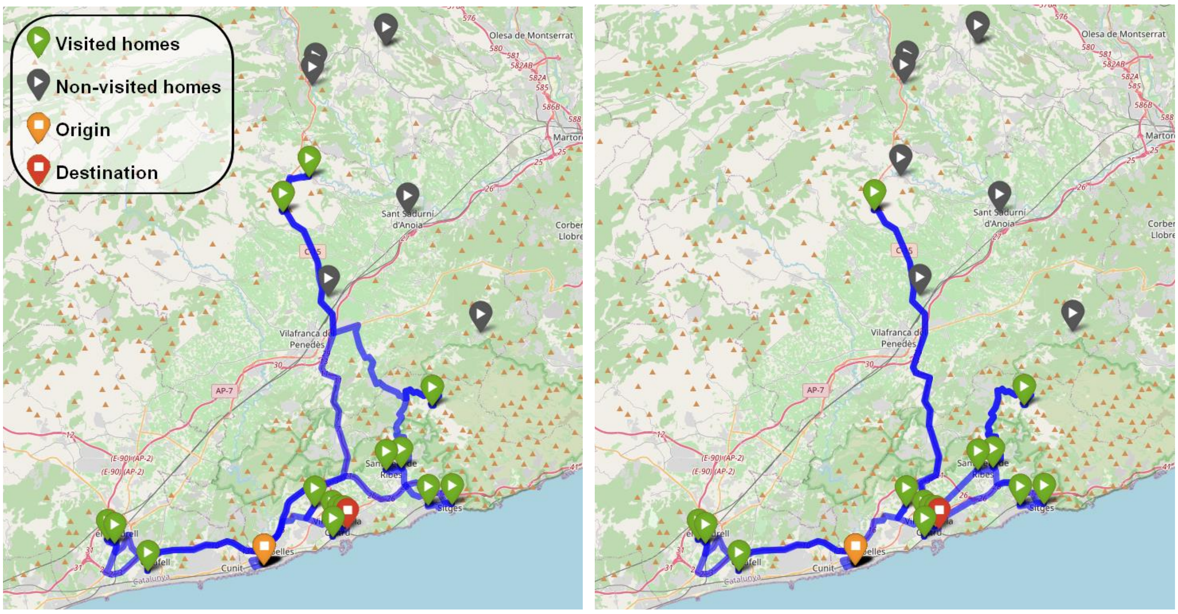
Table 6.2: Inputs and results for three real instances.

	Instance								
Input	**mar-25**			**mar-26**			**apr-04**		
Number of collection points	95			77			22		
MATL (min)	300			300			250		
Mean service time (min)	7			7			4		
Number of vehicles	6			4			1		
Output	**Greedy**	**BDS**	**BSS**	**Greedy**	**BDS**	**BSS**	**Greedy**	**BDS**	**BSS**
Deterministic total time (min)	1417.16	1394.92	-	1137.06	1123.56	-	246.39	248.35	-
Stochastic total time (min)	1454.81	1431.32	1428.69	1167.11	1152.71	1169.04	255.94	257.69	247.96
Deterministic MTL (min)	299.89	294.27	-	298.96	298.24	-	246.39	248.35	-
Stochastic MTL (min)	306.51	302.33	299.28	306.96	306.07	297.20	255.94	257.69	247.96
Visited collection points	95	95	95	77	77	77	14	15	14
Total collected demand	847	847	847	805	805	805	218	236	218
Reward	8104	8104	8104	7666	7666	7666	51800	51980	51800
Reliability	0%	0%	96%	0%	0%	100%	0%	0%	100%

We also consider a random travel time T_{ij} between two nodes i and j, where $T_{ij} = t_{ij}^{min} + W_{ij}$, with t_{ij}^{min} being the minimum time requested to travel from i to j assuming 'perfect' travel conditions, and W_{ij} is a random variable such that $W_{ij} \sim logN(\mu_{ij}, \sigma_{ij}^2)$. Sampled observations from T_{ij} and W_{ij} are denoted by t_{ij} and w_{ij}, respectively. Notice that W_{ij} represents a random delay caused by uncertainty conditions, such as traffic or weather. A web mapping service is employed to estimate t_{ij}^{min}. In our numerical experiments, we will assume that $\mu_{ij} = \sigma_{ij}^2 = 0.05t_{ij}^{min}, \forall (i,j) \in E$. It is worth clarifying that, despite the deterministic part of the problem is totally based on real data, the probability distributions and their parameters for both S_i and W_{ij} are experimental values selected to test our simheuristic approach. The parameter α employed to compute the savings is set to 0.6. This value was selected after performing a series of fine-tuning experiments. Also, based on previous studies (Ferone et al., 2019), we set the β parameter of the geometric probability distribution as a random value in the interval $(0.05, 0.25)$. In each iteration of our approach, β is randomly chosen inside this interval. The number of runs for the short simulation process and the long simulation process is set to 100 and 1000, respectively. A more thorough study for setting a suitable number of simulation runs is proposed by Rabe et al. (2020b). The maximum computational time employed by our approach is set to 60 seconds. The algorithm is implemented in Python 3 and executed in a personal computer with 16 GB RAM and an Intel Core *i*7-8750H processor at 2.2 GHz.

The rows titled as *Output* in Table 6.2 show our main obtained results. Each tested instance yields three types of solutions. First, our best deterministic solution (BDS), which is the best-found solution assuming that all parameters' values are perfectly known. Second, our best stochastic solution (BSS), which is the best solution found by our simheuristic, considering random service and traveling times. Moreover, the results obtained by the greedy deterministic heuristic are introduced. This greedy heuristic does not consider the biased selection of candidates, but a procedure corresponding to a 'manual planning', since the best current candidate is always selected when constructing the solution. It is worth saying that the BDS solutions are equivalent to those generated under the deterministic environment addressed in March 2020. Both the deterministic and the stochastic total times are the sum of the times of each designed route. The former represents the time yielded considering unrealistic perfectly known conditions. Hence, this parameter does not make sense for the BSS. Conversely, the stochastic total time is the time obtained after the BDS and the BSS have been simulated. Table 6.2 also shows the deterministic and the stochastic MTL. The MTL represents the time spent by the longest route in the solution. Notice that, in all instances, the BDS stochastic MTL exceeds the corresponding MATL. Alternatively, the BSS fulfills this time.

Since the considered problem is solved as a TOP, some collection points can be skipped. This is the case of the instance *apr-04*. Originally, 22 volunteers should be visited that day, however, since the MATL must be fulfilled, having only one vehicle is not enough to visit them all. Hence, the BDS achieves a maximum reward when visiting 15 collection points, while the greedy solution and the BSS achieve it by visiting 14. This fact implies that the BSS reaches a total collected demand and a reward smaller than the BDS. Nevertheless, since the BDS stochastic MTL exceeds the MATL, this solution has been proved to be infeasible. Therefore, although the BSS is a lower-quality solution, it is the best-found feasible solution when considering a realistic stochastic environment. Considering the greedy solution, it shows a similar behavior: despite collecting a reward smaller than the BDS, and the same amount as the BSS, its reliability is null. Figure 6.6 displays the best-found routes by the BDS (6.6a) and the BSS (6.6b). Orange and red markers represent the origin and destination points, respectively. Green markers are the visited collection points, and gray markers represent the non-visited ones. Figure 6.6b shows that an additional collection point is skipped by the route designed in the BSS, in order to meet the MATL constraint. Notice also that the routes designed by each solution type are not the same. For instance, those edges that are not traversed in the BSS are those that may show more variability due to the uncertainty derived from the real world, such as traffic congestion or weather conditions.



(a) Best deterministic solution. **(b)** Best stochastic solution.

Figure 6.6: Designed routes by the BDS and the BSS for the instance *apr-04*.

Finally, Table 6.2 shows a reliability indicator as an additional measure of the solution quality. We define the reliability of a solution as the probability that it does not fail. All routes within a solution are assumed to be independent. We consider that a route fails when the total time spent to traverse it is greater than the MATL. Hence, if K is the set of routes in a solution, a_k is the total number of simulation runs in which the route $k \in K$ fails, and n is the total number of performed simulation runs, the reliability R is computed according to Equation 6.1. Therefore, the reliability results for all instances show that both the greedy solution and BDS fail completely in guaranteeing the MTL, while the BSS is able to achieve a high reliability regardless of the stochastic environment.

$$R = \prod_{k \in K} \left(1 - \frac{a_k}{n}\right) \cdot 100\% \tag{6.1}$$

6.3 The LRP with Facility Sizing Decisions and Stochastic Demands

The LRP is one of the most complete problems in logistics optimization, since it includes all decision levels, i.e., strategic, tactical, and operational. From an Operational Research perspective, it can be seen as the combination of the FLP and the VRP, which are both NP-hard problems. Hence, the LRP is also NP-hard, and heuristic approaches are required for solving medium- and large-sized instances. Due to its complexity, the first reported studies on the LRP tackled it by separating the corresponding sub-problems (Salhi and Rand, 1989; Nagy and Salhi, 2007). As expected, this approach led to sub-optimal solutions. More recently, given the increase in computational power and the development of non-exact approaches, such as heuristic and metaheuristic algorithms, the LRP has been studied in an integrated way, which has clearly improved the obtained results (Prodhon and Prins, 2014). The LRP has been used to support decision-making processes related to supply chain network design (Lashine et al., 2006), humanitarian logistics (Ukkusuri and Yushimito, 2008), horizontal cooperation (Quintero-Araujo et al., 2019a), and city logistics (Nataraj et al., 2019), among others. One of the most studied versions of the LRP is the capacitated LRP, in which both depot and vehicle capacity constraints must be satisfied (the acronym LRP will henceforth refer to this problem). However, all previous works consider the depot capacity as a fixed value. This could not be a suitable approach when dealing with realistic problems, since it is usual that decision-makers can select the size of a facility from a discrete set of known available sizes, or even freely. For real-world problems, this set is usually associated with investment activities, such as building facilities (Zhou et al., 2019), purchasing equipment (Tordecilla-Madera et al., 2017), or qualifying workforce (Correia and Melo, 2016).

From an academic point of view, the consideration of flexible sizes in the facilities has been rarely addressed in the literature. Nevertheless, real-life examples from both LRP (Zhou et al., 2019; Hemmelmayr et al., 2017; Tunalıoğlu et al., 2016) and non-LRP (Tordecilla-Madera et al., 2017; Correia and Melo, 2016) contexts show the relevance of considering a variety of facility sizes to select from, instead of just a unique size alternative, as it is the case in most LRP studies. These problems consider that parameters are deterministic, i.e., they assume that inputs are known in advance. This assumption is far from reality in many applications, such as waste collection or humanitarian logistics. The LRP literature addressing stochastic parameters is still scarce. Most found works hybridize simulation with some heuristic or metaheuristic to tackle efficiently both uncertainty and NP-hardness. For example, Quintero-Araujo et al. (2019b) propose a simheuristic to solve an LRP with stochastic demands. They hybridize MCS with an ILS metaheuristic. A set of benchmark instances are used to test the proposed approach. Rabbani et al. (2019) also propose a simheuristic approach that combines a non-dominated sorting genetic algorithm-II (NSGA-II) and MCS. They address a multi-objective multi-period LRP in the context of the hazardous waste management industry. Both generated waste and number of people at risk are stochastic. Inventory decisions are also taken into account.

6.3.1 Problem Definition

The LRP with facility sizing decisions and stochastic demands consists in: *(i)* opening one or more depots (facilities) of different sizes; and *(ii)* designing, for each open depot, a number of routes whose aggregated customers' demand does not exceed its capacity. Each route must start and finish at the same depot. As demands are stochastic, a percentage of the vehicles' capacity is reserved, as a safety stock (*SS*), in case the demand is higher than expected. Therefore, the main decision variables in this problem are related to the number of facilities to open, where to locate them, what size must be installed for each facility, which customers must be allocated to each open depot, how many vehicles must be used, and how to design the associated routes. This problem is NP-hard since it contains, as special cases, the VRP, the MDVRP, and the FLP, all of them known to be computationally hard.

Formally speaking, the LRP can be defined on a complete, weighted, and undirected graph $G(V, E, C)$, in which V is the set of nodes (comprising the subset J of potential depot locations and subset I of customers), E is the set of edges, and C is the cost matrix of traversing each edge. A set K of unlimited homogeneous vehicles with capacity constraints is available to perform the routes. Moreover, it is assumed that all vehicles are shared by all depots (i.e., no depot has a specific fleet) and each edge $e \in E$ satisfies the triangle inequality. The customers' demands are stochastic and follow a known probability distribution. The version studied in this section considers that the size of the facilities to open is a decision to be made. To achieve this, a set of alternative sizes for each depot and associated opening costs are provided as inputs. Depots might have equal or different capacities among them. Each customer must be serviced from the depot to which it has been allocated by a single vehicle, i.e., split deliveries are not allowed.

The objective is to minimize the expected total cost (TC), which includes the cost of opening depots (OC), the routing cost (RC), and the stochastic cost (SC), i.e., $TC = OC + RC + SC$. The latter cost is incurred each time that a route fails, i.e., each time that the realized demand of a route is greater than the vehicle capacity. In this case, two different costs are calculated, depending on the corrective action considered: *(i)* a reactive strategy with a cost c_{reac}, in which a vehicle must perform a round-trip to the depot for a replenishment in the occurrence of a route failure; and *(ii)* a preventive strategy with a cost c_{prev}, in which a vehicle performs a detour to the depot before visiting the next customer, if the expected current non-served demand is higher than the vehicle's current load. Then, the expected stochastic cost is computed as $SC = \min\{c_{reac}, c_{prev}\}$.

6.3.2 Solution Approach

We propose a simheuristic approach (Juan et al., 2018) for minimizing the expected total cost. Simheuristics have been recently used to solve optimization problems with stochastic components, such as arc routing problems with stochastic demands (Gonzalez-Martin et al., 2018) or stochastic waste collection problems (Gruler et al., 2017a). In particular, our methodology combines an ILS metaheuristic with MCS to deal with the stochastic nature of the problem. As discussed by Grasas et al. (2016) and Ferone et al. (2019), the ILS and GRASP metaheuristic frameworks offer a well-balanced combination of efficiency and relative simplicity, and can be easily extended to a simheuristic. Figure 6.7 depicts the main characteristics of our approach, composed of three stages. During the first stage, a set of promising facility-location maps are generated using a constructive heuristic, which employs BR techniques. The main input parameters of our model are: the number of customers and potential depots, their locations in Cartesian coordinates, the vehicle capacity, the available sizes for each potential depot, the customers' demands with known mean and standard deviation, and the fixed and variable costs of opening a depot. The algorithm starts computing the minimum and maximum number of required depots, based on both the total demand and the maximum and minimum available sizes. Next, it randomly selects both the number of depots to be opened –which is a value between the range of values previously computed– and the size of each depot. This size is selected from a discrete set of known available sizes.

Subsequently, the customers are assigned to a specific depot. This allocation is performed using a marginal-savings criterion proposed by Juan et al. (2015b). Broadly speaking, this criterion computes the savings of assigning a customer $i \in I$ to an open facility $j \in J$ with respect to assigning i to the best alternative facility $j^* \in J$. Finally, when all customers have been allocated to a given facility, a modified version of the savings heuristic described by Belloso et al. (2019) is applied to generate an initial routing plan. This heuristic is based on BR techniques. This process permits transforming a deterministic heuristic into a probabilistic algorithm, while still preserving the logic behind the heuristic. This complete procedure is repeated until the maximum computation time for this phase is reached. Finally, all maps are evaluated according to the total deterministic cost, formed by the opening cost and the routing cost. Then, the most "promising" maps obtained in this first stage are sent to the second one. The number of most "promising" maps depends on how much time is available to perform a broader exploration. Our algorithm stores the 2 maps with the lowest deterministic cost.

During the second stage, an initial short number of MCS runs is carried out to estimate a safety stock. This decision variable refers to a percentage of the vehicle capacity, which is reserved to respond more effectively to the random demand. Then, the ILS metaheuristic improves the set of "promising" maps by iteratively exploring the search space and conducting a second process of simulation runs. This procedure is based on: *(i)* perturbing the current solution to obtain a new starting point; and *(ii)* exploring the neighborhood of this new solution using a local search (LS) procedure. As perturbation methods, we have used two different strategies. In the first one, the algorithm randomly selects a set of customers and tries to reassign them in a random way to another facility without violating its capacity. Regarding the second strategy, the algorithm randomly exchanges the allocation of a percentage of customers among facilities. This percentage starts in 0.5, and it is successively increased in each new iteration of the algorithm to explore different neighborhood sizes. The strategy to be used in each iteration of the algorithm is randomly selected. Following this perturbation process, the algorithm starts an LS around the newly generated solution in order to improve it. As LS we have used a *two-opt inter-route* operator, which interchanges two chains of randomly selected customers between different facilities. This operator is applied until it cannot be further improved. After this LS, we quickly assess the obtained solution under stochastic conditions by employing a short number of MCS runs. This allows us to generate rough estimates of the solution performance under stochastic conditions, which also enables for identifying a pool of 'elite' solutions. Whenever a new solution outperforms the current base solution of the ILS, the latter is updated with the former and added to the pool of elite solutions. Notice that the MCS does not only provide estimates to the expected cost associated with the solutions generated by our approach, but also it reports feedback to the metaheuristic search process. In order to further diversify the search, the algorithm might occasionally accept non-improving solutions following an acceptance criterion. The process is repeated until the stopping criterion of this stage is met. Finally, in the third stage a refinement procedure using a larger number of simulation runs is applied to the elite solutions. This enables to obtain a more accurate estimation of the expected total cost as well as

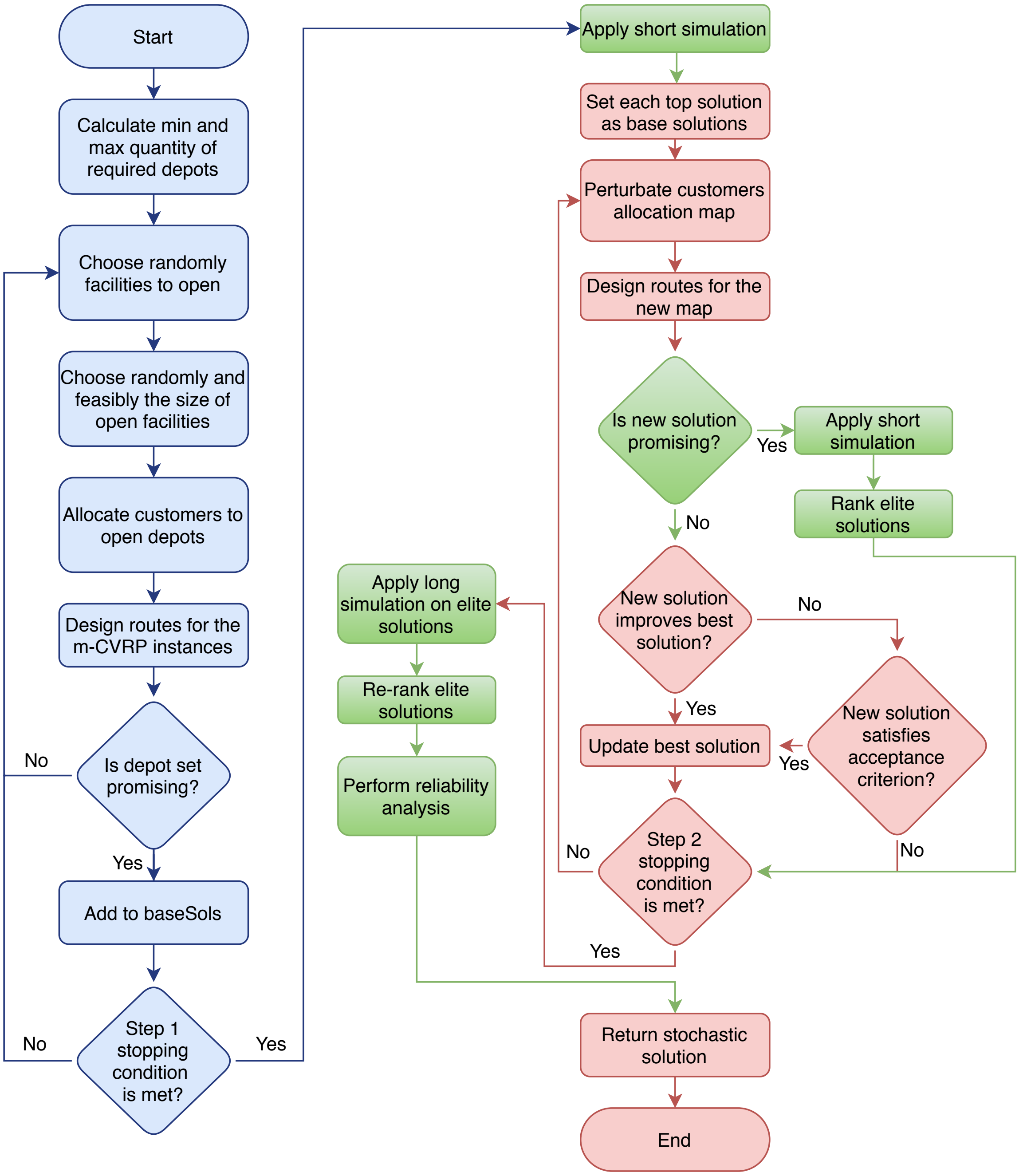


Figure 6.7: Flowchart of our simheuristic approach.

other statistics, e.g., solution reliability, variance, etc. Particularly, the main output variables in our experiments are: the total opening and routing costs, the total stochastic cost incurred whenever a route fails, the solution reliability and the safety stock.

6.3.3 Computational Experiments and Results

All our experiments were carried out using Akca's instances (Akca et al., 2009). Such instances are designed for a deterministic and non-flexible LRP. Therefore, they were adapted to our stochastic and flexible case. Three main modifications were made:

1. We assume that the demand proposed in these instances is the mean of a log-normal probability distribution. That is, if D_i is the random variable representing the demand of the customer $i \in I$, and d_i is the deterministic demand in Akca's set, then $E[D_i] = d_i$. Also, three different values of variance are considered: low, medium, and high, i.e., for $\lambda \in \{0.05, 0.10, 0.20\}$, $Var[D_i] = \lambda \cdot d_i$. It is worth to notice that our approach allows the use of any other probability distributions.

2. As the original Akca's instances consider that a fixed size is available to assign to open depots, we provide a total of five alternative sizes, so that our algorithm selects one of them for each open depot. If s_j is the size proposed by the original instance for each potential depot $j \in J$, and L is the set of available sizes, our approach' alternative sizes are $s_{jl} \in \{(1-2r)s_j, (1-r)s_j, s_j, (1+r)s_j, (1+2r)s_j\}$, where $l \in L$ and $0.0 < r < 0.5$. This parameter represents the difference between available sizes. When $r = 0$, the case is the same as Akca's. Our experiments consider a fixed value of $r = 0.25$.

3. Akca's instances consider a fixed cost (f_j) incurred whenever a depot $j \in J$ is open. Our experiments also consider this. Additionally, we introduce a variable cost (o_{jl}) depending on f_j and s_{jl}, namely: $o_{jl} = \frac{(s_{jl} - s_j)}{2s_j} \frac{\sum_j f_j}{|J|}$. This formula preserves o_{jl} in the same order as f_j for each depot $j \in J$. Besides, it yields negative costs whenever $s_{jl} < s_j$, positive costs whenever $s_{jl} > s_j$, and a null cost when $s_{jl} = s_j$. Thus, our results can be compared with those found in the LRP literature.

Our algorithm uses the following parameters to run the experiments: *(i)* a total of 350 iterations for map perturbations; *(ii)* a total of 150 iterations for the BR savings heuristic; *(iii)* a total of 150 iterations for splitting; *(iv)* a random value between 0.05 and 0.80 for β_1, the parameter of the geometric distribution associated with the BR selection during the allocation map process; *(v)* a random value between 0.07 and 0.23 for β_2, the parameter of the geometric distribution associated with the BR heuristic for routing; *(vi)* a total of $n = 100$ runs for the initial simulation stage; and *(vii)* a total of $N = 5,000$ runs for the intensive simulation stage. The safety stock is estimated through a fast simulation process of 100 iterations, testing only discrete values of SS between 0% and 10%. Our proposed algorithm was coded as a Java application. All experiments were executed on a standard Windows PC with a Core $i5$ processor and 6 GB RAM. A total of ten different random seeds were used for each instance. Our results are compared in terms of both costs and reliability (R) with those obtained by Quintero-Araujo et al. (2019b), who do not consider flexibility in facility sizes. If M is the set of routes in a solution, the reliability of a single route $m \in M$ is defined as the probability that it does not fail, i.e.:

$$R_m = \left(1 - \frac{b_m}{N}\right) \cdot 100\% \tag{6.2}$$

where b_m is the total number of simulation runs in which the route $m \in M$ fails. Routes within a solution are considered independent components in a series system. Thus, the estimated reliability of a solution with $|M|$ routes is computed as:

$$R = \prod_{m=1}^{M} R_m \tag{6.3}$$

Tables 6.3, 6.4, and 6.5 show our best-found stochastic solution (OBS) considering low-, medium-, and high-variance scenarios, respectively. Regardless of the variance level, and due to the flexibility in the selection of the facility size, our approach is able to outperform the results provided by Quintero-Araujo et al. (2019b) in terms of expected total costs (TC) for 11 out of 12 instances. Hence, the flexibility in the facility size enables to reach savings up to 6.72% in a single instance. On the average, our solutions are also better when comparing just opening (OC) or just routing cost (RC), which is a direct consequence of increasing the flexibility in depot-sizing decisions. Most instances show a reduction in opening costs, although routing costs increase. Except for instance *Cr30x5a-3*, savings in opening costs are always higher than increments in routing costs.

Table 6.3: Results with a low-variance level.

Instance	Quintero-Araujo et al. (2019b)					Our Best Stochastic						TC	Reliability
	OC	*RC*	*SC*	*TC*	*R*	*OC*	*RC*	*SC*	*TC*	*R*	*SS*	Gap	difference
Cr30x5a-1	200.00	619.51	2.10	821.61	94%	200.00	575.14	2.37	777.51	94%	0%	-5.37%	0%
Cr30x5a-2	200.00	621.45	0.00	821.45	100%	200.00	607.28	0.04	807.32	100%	3%	-1.72%	0%
Cr30x5a-3	200.00	502.29	4.53	706.82	85%	187.50	509.25	10.99	707.74	83%	3%	0.13%	-2%
Cr30x5b-1	200.00	680.03	1.11	881.14	98%	225.00	623.22	9.37	857.59	85%	0%	-2.67%	-13%
Cr30x5b-2	200.00	625.32	0.00	825.32	100%	187.50	625.32	0.00	812.82	100%	2%	-1.51%	0%
Cr30x5b-3	200.00	684.58	1.74	886.32	96%	187.50	684.58	2.25	874.33	97%	1%	-1.35%	1%
Cr40x5a-1	200.00	729.12	0.04	929.16	100%	162.50	731.84	0.03	894.37	100%	1%	-3.74%	0%
Cr40x5a-2	200.00	688.80	0.00	888.80	100%	225.00	639.02	0.10	864.12	100%	0%	-2.78%	0%
Cr40x5a-3	200.00	748.64	4.67	953.31	93%	162.50	752.88	0.97	916.35	98%	0%	-3.88%	5%
Cr40x5b-1	200.00	852.05	5.24	1057.29	86%	162.50	852.04	6.90	1021.45	87%	1%	-3.39%	1%
Cr40x5b-2	200.00	781.54	0.05	981.59	100%	225.00	690.57	0.08	915.65	100%	1%	-6.72%	0%
Cr40x5b-3	200.00	769.76	2.34	972.10	92%	175.00	772.87	0.07	947.93	100%	2%	-2.49%	8%
Average	200.00	691.92	1.82	893.74	95%	191.67	672.00	2.76	866.43	95%	1%	-2.96%	0.00%

Table 6.4: Results with a medium-variance level.

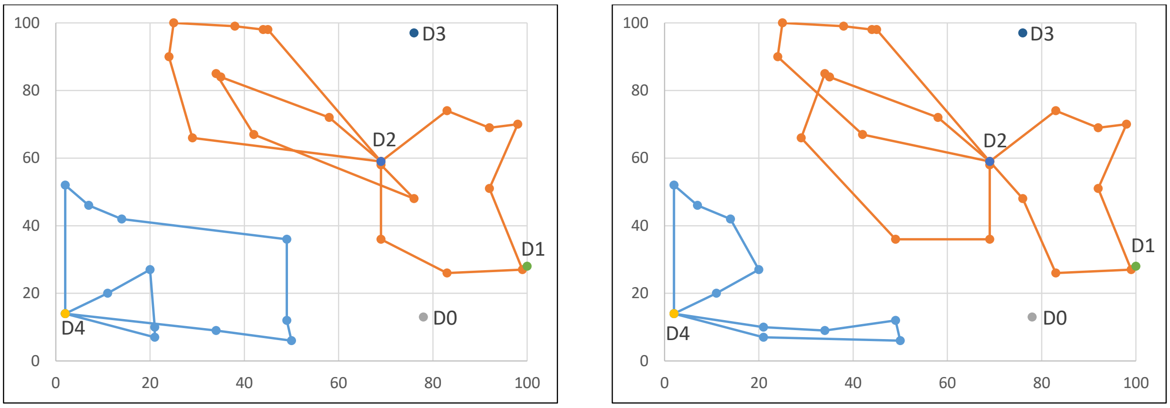
Instance	Quintero-Araujo et al. (2019b)					Our Best Stochastic						TC	Reliability
	OC	*RC*	*SC*	*TC*	*R*	*OC*	*RC*	*SC*	*TC*	*R*	*SS*	Gap	difference
Cr30x5a-1	200.00	619.51	6.41	825.92	84%	200.00	575.14	7.63	782.77	83%	2%	-5.22%	-1%
Cr30x5a-2	200.00	621.46	0.27	821.73	100%	200.00	607.28	0.46	807.74	99%	3%	-1.70%	-1%
Cr30x5a-3	200.00	502.29	7.44	709.73	76%	187.50	509.25	18.50	715.25	72%	3%	0.78%	-4%
Cr30x5b-1	200.00	682.97	2.61	885.58	95%	225.00	623.22	14.63	862.85	77%	0%	-2.57%	-18%
Cr30x5b-2	200.00	625.32	0.00	825.32	95%	187.50	625.32	0.00	812.82	100%	2%	-1.51%	5%
Cr30x5b-3	200.00	684.58	6.49	891.07	86%	187.50	684.58	10.21	882.28	86%	0%	-0.99%	0%
Cr40x5a-1	200.00	731.84	0.70	932.54	98%	162.50	739.24	0.01	901.75	100%	3%	-3.30%	2%
Cr40x5a-2	200.00	688.81	0.01	888.82	100%	225.00	643.52	3.07	871.59	93%	1%	-1.94%	-7%
Cr40x5a-3	200.00	752.89	4.25	957.14	93%	162.50	752.88	4.46	919.85	93%	1%	-3.90%	0%
Cr40x5b-1	200.00	862.57	1.53	1064.10	98%	162.50	858.58	4.54	1025.62	91%	2%	-3.62%	-7%
Cr40x5b-2	200.00	781.54	2.06	983.60	98%	225.00	690.57	2.06	917.63	98%	1%	-6.71%	0%
Cr40x5b-3	200.00	769.76	5.18	974.94	82%	175.00	772.87	1.42	949.29	96%	2%	-2.63%	14%
Average	200.00	693.63	3.08	896.71	92%	191.67	673.54	5.58	870.79	91%	2%	-2.78%	-1.42%

In contrast, less costly routes can be designed when either increasing or keeping the same total available capacity. The latter case yields the same opening cost but it re-configures the open facilities size. Figure 6.8 shows an example of this situation for the instance *Cr30x5a-2* in the high-variance scenario. Figure 6.8(a) displays the routes obtained by Quintero-Araujo et al. (2019b) while Figure 6.8(b) shows our best stochastic solution. Both solutions open depots *D2* and *D4*. However, while the approach by Quintero-Araujo et al. (2019b) opens

Table 6.5: Results with a high-variance level.

Instance	Quintero-Araujo et al. (2019b)					Our Best Stochastic						TC Gap	Reliability difference
	OC	*RC*	*SC*	*TC*	*R*	*OC*	*RC*	*SC*	*TC*	*R*	*SS*		
Cr30x5a-1	200.00	619.51	14.99	834.50	67%	200.00	575.14	19.66	794.80	66%	2%	-4.76%	-1%
Cr30x5a-2	200.00	621.46	3.34	824.80	96%	200.00	607.74	0.02	807.76	100%	5%	-2.07%	4%
Cr30x5a-3	200.00	502.29	11.31	713.60	66%	187.50	509.25	27.86	724.61	57%	2%	1.54%	-9%
Cr30x5b-1	200.00	682.97	8.00	890.97	86%	225.00	623.22	19.99	868.21	68%	10%	-2.55%	-18%
Cr30x5b-2	200.00	625.32	0.04	825.36	100%	187.50	625.32	0.10	812.92	100%	3%	-1.51%	0%
Cr30x5b-3	200.00	684.57	16.64	901.21	68%	187.50	684.58	24.93	897.00	68%	1%	-0.47%	0%
Cr40x5a-1	200.00	729.13	4.75	933.88	87%	162.50	737.20	2.85	902.55	92%	2%	-3.35%	5%
Cr40x5a-2	200.00	688.80	0.80	889.60	99%	225.00	642.02	1.79	868.82	95%	3%	-2.34%	-4%
Cr40x5a-3	200.00	760.10	2.53	962.63	92%	162.50	763.69	5.78	931.97	87%	2%	-3.19%	-5%
Cr40x5b-1	200.00	863.91	7.98	1071.89	89%	237.50	786.00	4.65	1028.14	91%	3%	-4.08%	2%
Cr40x5b-2	200.00	781.54	9.55	991.09	88%	225.00	690.57	9.35	924.91	89%	2%	-6.68%	1%
Cr40x5b-3	200.00	769.76	10.97	980.73	69%	175.00	780.62	4.14	959.76	92%	3%	-2.14%	23%
Average	200.00	694.11	7.58	901.69	84%	197.92	668.78	10.09	876.79	84%	3%	-2.63%	-0.17%

two depots with size 1,000 each, we open one depot with size 750 (*D4*) and one depot with size 1,250 (*D2*). This slight change induced by the flexibility in sizing decisions allows for reassigning one customer and to reduce routing costs. A similar analysis can be carried out for the remaining instances.



(a) (b)

Figure 6.8: Best-found solution by the non-flexible LRP (a) and our approach (b) for the instance *Cr30x5a-2*.

Our best-found deterministic solution (OBD) is also tested in a stochastic environment, using 0% of safety stock protection against uncertainty. Table 6.6 compares this solution's results with our best stochastic solution (OBS) in terms of cost and reliability. Comparisons are drawn in terms of both cost gaps and reliability differences. On average, results show a slight improvement in the obtained costs, yielding a decrease up to 0.47% in the high-variance scenario. Nevertheless, the real contribution of using a stochastic approach in our problem is the increase in reliability, which reaches a maximum of 48% in a single instance, and an average of 12.5% in the high-variance scenario.

6.4 Conclusions

This chapter has presented three applications of simheuristic algorithms. The first two sections show cases based on the pandemic crisis posed by the COVID-19 in Barcelona in

Table 6.6: Best deterministic vs. best stochastic solutions under stochastic scenarios.

Instance	OBD		OBS		TC gap	Reliability difference
	TC	*R*	*TC*	*R*		
Low Variance						
Cr30x5a-1	778.25	94%	777.51	94%	-0.10%	0%
Cr30x5a-2	807.34	100%	807.32	100%	0.00%	0%
Cr30x5a-3	707.44	84%	707.74	83%	0.04%	-1%
Cr30x5b-1	855.49	86%	857.59	85%	0.25%	-1%
Cr30x5b-2	812.82	100%	812.82	100%	0.00%	0%
Cr30x5b-3	874.61	96%	874.33	97%	-0.03%	1%
Cr40x5a-1	894.40	100%	894.37	100%	0.00%	0%
Cr40x5a-2	873.29	82%	864.12	100%	-1.05%	18%
Cr40x5a-3	916.47	98%	916.35	98%	-0.01%	0%
Cr40x5b-1	1022.24	86%	1021.45	87%	-0.08%	1%
Cr40x5b-2	915.68	100%	915.65	100%	0.00%	0%
Cr40x5b-3	953.44	69%	947.93	100%	-0.58%	31%
Average	867.62	91%	866.43	95%	-0.13%	4.08%
Medium Variance						
Cr30x5a-1	785.26	84%	782.77	83%	-0.32%	-1%
Cr30x5a-2	808.15	99%	807.74	99%	-0.05%	0%
Cr30x5a-3	716.16	71%	715.25	72%	-0.13%	1%
Cr30x5b-1	860.37	76%	862.85	77%	0.29%	1%
Cr30x5b-2	812.82	100%	812.82	100%	0.00%	0%
Cr30x5b-3	882.73	85%	882.28	86%	-0.05%	1%
Cr40x5a-1	895.16	98%	901.75	100%	0.74%	2%
Cr40x5a-2	881.88	68%	871.59	93%	-1.17%	25%
Cr40x5a-3	919.86	92%	919.85	93%	0.00%	1%
Cr40x5b-1	1034.90	66%	1025.62	91%	-0.90%	25%
Cr40x5b-2	917.95	98%	917.63	98%	-0.03%	0%
Cr40x5b-3	959.70	56%	949.29	96%	-1.08%	40%
Average	872.91	83%	870.79	91%	-0.23%	7.92%
High Variance						
Cr30x5a-1	799.56	67%	794.80	66%	-0.60%	-1%
Cr30x5a-2	811.14	95%	807.76	100%	-0.42%	5%
Cr30x5a-3	726.17	57%	724.61	57%	-0.21%	0%
Cr30x5b-1	864.96	67%	868.21	68%	0.38%	1%
Cr30x5b-2	812.94	100%	812.92	100%	0.00%	0%
Cr30x5b-3	897.49	67%	897.00	68%	-0.05%	1%
Cr40x5a-1	898.12	90%	902.55	92%	0.49%	2%
Cr40x5a-2	892.21	53%	868.82	95%	-2.62%	42%
Cr40x5a-3	926.90	82%	931.97	87%	0.55%	5%
Cr40x5b-1	1052.68	45%	1028.14	91%	-2.33%	46%
Cr40x5b-2	925.67	88%	924.91	89%	-0.08%	1%
Cr40x5b-3	967.39	44%	959.76	92%	-0.79%	48%
Average	881.27	71%	876.79	84%	-0.47%	12.50%

2020. Real-world data have been employed to model customers' locations and demands in an OVRP and a STOP, respectively. Travel and service times are considered stochastic. Conversely, demands are considered stochastic in the LRP with facility sizing decisions addressed in the last section. The results obtained in the three applications show the advantages of using a simheuristic algorithm for solving stochastic transportation and logistics problems, since it outperforms the best deterministic solution in terms of costs and reliability.

Particularly, in the case of the OVRP we study a medical waste collection problem. Typical loading constraints are not considered here due to the small size of most medical items. However, maximum driving time per route must be taken into account. Also, the introduction of random travel and pick-up times makes the optimization problem more challenging.

In order to deal with this complexity, we first propose a BR heuristic, which is later extended to a full simheuristic combining an optimization module with a simulation one. This allows us to obtain better solutions with higher reliability levels. Our experiments also illustrate how a good collection plan under deterministic conditions can become a sub-optimal plan as uncertainty is introduced in the scenario.

Regarding the STOP, collection activities of 3D-printed elements must be performed; however, since the drivers are volunteers, both the number of available vehicles and the time to traverse a route are limited. The obtained results show the suitability of using a STOP for this type of problems, since it allows for skipping some collection points with low reward in order to fulfill the time limit. Furthermore, our approach considers not only these rewards to construct the routes, but also the travel times between each pair of collection points. This double criterion enables to design good-quality solutions. Multiple indicators were used to evaluate this quality, namely: the total travel time, the MTL, the visited collection points, the reward, and the reliability. Two types of solutions are generated: the best deterministic solution (BDS) and the best stochastic solution (BSS). The BDS is the best-found solution when perfectly known conditions are considered, although these are unrealistic. Moreover, the simulation of the BDS shows the infeasibility of this solution under stochastic conditions. As an alternative, our simheuristic provides the BSS, which achieves outstanding values for the considered indicators, preserves the feasibility of the solution, and far outperforms the BDS in terms of reliability.

Finally, regarding the LRPFS, we consider the case in which the size of the facilities to open is an additional variable to model. This size is usually selected from a discrete set related to diverse investment activities. However, the literature addressing this variable is still scarce, despite the relevance of the problem in real-world applications. Moreover, we have considered that customers' demands are stochastic. To the best of our knowledge, it is the first time that the stochastic LRP with facility sizing decisions has been studied. As this problem is NP-hard, we have proposed a simheuristic algorithm as solving approach. Medium-sized benchmark instances with different variability levels were used. On the one hand, the obtained results show that cost savings are attained due to the considered flexibility in facility sizing. These savings may be yielded by: *(i)* a reduction in opening costs, given the installation of smaller size facilities; or *(ii)* a reduction in routing costs, given the installation of higher size facilities, which enables to reconfigure the designed routes to make them shorter. On the other hand, our approach increases the reliability of solutions when compared against the best deterministic solution tested in a stochastic setting. All in all, these results illustrate the benefits of using a simheuristic approach.

Chapter 7

Applications of Fuzzy Simheuristics

The applications of simheuristics described in the previous chapter assume that problem's parameters are either deterministic or stochastic, i.e., they follow a known probability distribution. Nevertheless, this might not be the case in real-world problems where there is not enough available information to even reliably estimate a probability distribution (Corlu et al., 2020). In these cases, an alternative to address uncertainty is the use of fuzzy systems, which make use of fuzzy logic to estimate the value of any parameter. Hence, fuzzy systems employ both linguistic rules and different degrees of membership for different groups. That is, fuzzy logic allows any value to be true or false with a given degree of membership for each of these categories. In order to address combinatorial optimization problems (COP) that include this type of uncertainty, Oliva et al. (2020) introduce the concept of "fuzzy simheuristics", and propose an algorithm to solve a TOP where both battery levels and weather conditions are fuzzy. In this chapter[1] we extend the fuzzy simheuristics methodology to address other logistics and transportation problems in which customers demands are fuzzy. Concretely, Section 7.1 considers a VRP with fuzzy demands and a TOP with fuzzy travel times between nodes. In turn, Section 7.2 addresses an LRP with facility sizing decisions and fuzzy demands. Hence, the main contribution of this chapter is to propose a methodology that solve efficiently combinatorial T&L problems where demands of a subset of customers are stochastic, while demands of the complementary subset are fuzzy.

7.1 The VRP and the TOP with Stochastic and Fuzzy Parameters

Managers tend to rely on analytical methods that allow them to make informed decisions. This explains why optimization models play a key role in many industries and businesses, including the logistics and transportation sector. Whenever accurate information on the inputs and constraints of the optimization problem is available, the resulting deterministic models can be solved by using well-known methods, either of exact or approximate nature.

[1]The contents of this chapter are based on the following works:

- **Tordecilla, R.D.**, Martins, L.C., Panadero, J., Copado, P.J., Perez-Bernabeu, E., & Juan, A.A. (2021). Fuzzy simheuristics for optimizing transportation systems: dealing with stochastic and fuzzy uncertainty. *Applied Sciences*, 11(17), 7950.
- **Tordecilla, R.D.**, Copado-Mendez, P.J., Panadero, J., Quintero-Araujo, C.L., Montoya-Torres, J.R., & Juan, A.A. (2021). Combining heuristics with simulation and fuzzy logic to solve a flexible-size location routing problem under uncertainty. *Algorithms*, 14(2), 45.

Many optimization problems in real-life transportation involve taking into account a large number of variables and rich constraints, which often makes them to be NP-hard (Fausto et al., 2020). When this is the case, the computational complexity makes it difficult to obtain optimal solutions in a short computational time. At this point, heuristic approaches can provide near-optimal solutions that, in turn, cover all the requirements of the problem (Schneider and Kirkpatrick, 2007). When dealing with challenging optimization problems, there is a tendency to divide them into sub-problems, which simplifies the difficulty but might also lead to sub-optimal solutions (Salhi and Rand, 1989; Nagy and Salhi, 2007). Given the increase in computational power experienced during the last decade, and also the development of advanced metaheuristic algorithms, it is possible nowadays to solve rich and large-scale problems that were intractable in the past (Prodhon and Prins, 2014). In the scientific literature on COPs, it is often assumed that the input values are constant and known. However, in a real-world scenario this is rarely the case, since uncertainty is often present and affects these inputs.

In the context of T&L, some examples of these inputs are: travel times, customer demands, service times, battery durability, etc. Whenever these inputs can be modeled by random variables, simheuristic algorithms –which combine heuristics with simulation– become a useful tool to address the associated optimization problem (Chica et al., 2020). Nevertheless, simheuristics are designed to handle situations where uncertainty can be modeled by random variables, each of which follows a well-known probability distribution. When dealing with non-probabilistic uncertainty, fuzzy techniques might be a good choice. Therefore, fuzzy techniques can be particularly interesting for modeling uncertainty whenever it cannot be represented by random variables, for example: if not enough data are available, if the data cannot be fitted to a probability distribution, or if qualitative expert opinions must also be considered. A fuzzy system is based on fuzzy logic. Inputs enter the system, which computes fuzzy outputs on the basis of a set of rules established by a human expert (Teodorović and Pavković, 1996). In order to obtain solutions that mix information from different sources, the output of the fuzzy system includes different degrees of membership for different groups. This means that a fuzzy system can handle decisions in a non-binary logic scenario, since the outputs have a partial degree of being "true" or "false". Therefore, the main contribution of this section is to provide both conceptual and practical insights on how fuzzy simheuristics can be applied in the optimization of different transportation systems, which include the well-known VRP under uncertainty conditions, as well as the TOP under uncertainty conditions. A comprehensive introduction to both problems can be found in Toth and Vigo (2014) and Chao et al. (1996), respectively. Therefore, we address and discuss the novel concept of fuzzy simheuristics, which has hardly been addressed in the literature. Accordingly, this new class of solution methodology is designed to solve the aforementioned transportation problems, whose performance and prospects have been duly analyzed and presented.

7.1.1 Problem Definition

Both the VRP and the TOP are addressed in this section. Hence, we provide below an overview about both transportation problems.

7.1.1.1 The Vehicle Routing Problem

The VRP is a well-known COP with a vast number of applications in the transportation sector (Braekers et al., 2016). Solving the VRP aims to design cargo vehicle routes with minimum transportation costs to distribute goods between depots and a set of consumers. Since the capacity of the cargo vehicles is usually taken into account, the VRP is often referred to as capacitated VRP. In its basic version, the distribution network of the VRP can be defined as a directed graph $G = (N, E)$, where: *(i)* $N = \{0, 1, \ldots, |C|\}$ is the set of vertices, with node 0 being the central depot and C being the set of customers; and *(ii)* $E = \{(i, j) \mid i, j \in N, i < j\}$ is the set of edges connecting pairs of nodes. Each customer $i \in C$ requires a demand $d_i > 0$, which affects the of the vehicle . The objective, in solving this problem, is to minimize the total cost of serving all customers, subject to: *(i)* each route starts and ends at the central depot; *(ii)* each customer is visited only once and by exactly one vehicle; and *(iii)* the total demand required by the costumers on a route does not exceed the vehicle capacity. Apart from this basic version, multiple extensions of the problem can be found in the literature (Section 2.1.1). Many real-life versions of this problem combine some of the aforementioned constraints and others, making them difficult to solve (Simeonova et al., 2020).

7.1.1.2 The Team Orienteering Problem

One of the main differences between the TOP and the VRP is that in the former visiting all customers is not mandatory. In other words, some nodes can be omitted during the generation of the routing plan. This is due to restrictions on the fleet size and on the maximum length that can be covered by any route. In a typical TOP, rewards are collected when a node is visited and, therefore, the objective is to maximize the total reward collected by a fixed fleet of vehicles. Vehicles depart from an origin node and have to reach a destination node. Cargo constraints are not usually considered in the basic version of the TOP. However, as in the case of the VRP, many variants can be found in the literature (Section 2.1.2). Typically, non-visited nodes are customers located far away from each other or offering low rewards. As in the VRP case, the TOP network is composed of an origin depot, a destination depot, and a set of customers. It can be defined as a directed graph $G = (N, E)$, where *(i)* $N = \{0, 1, \ldots, |C|, n\}$ is the set of nodes, including the origin depot (node 0) and the destination depot (node n); and *(ii)* $E = \{\{(i, j) \mid i, j \in N, i < j\}$ represents the set of edges connecting the nodes. Each customer $i \in C$ offers a reward $u_i > 0$ when it is visited. A fleet of vehicles is available at the origin depot. Each vehicle visits a selected subset of nodes in C to collect the associated rewards, and moves to the destination depot.

7.1.2 Solution Approach

Both the VRP and the TOP are NP-hard problems (Lenstra and Kan, 1981; Golden et al., 1987). Therefore, due to the combinatorial nature of these problems, the use of exact solving approaches is often limited by the size of the problem instances. When dealing with real-life instances, which are typically large-scale instances, the use of heuristics and metaheuristics has been proved to be a good alternative (Elshaer and Awad, 2020). Although metaheuristics are capable of finding near-optimal (or even optimal) solutions to many different COPs in reasonable computing times, these approaches have been mainly designed to solve deterministic versions. Consequently, metaheuristics are not able to cope adequately with stochastic components, being their application constrained against uncertain scenarios as the ones proposed in this section. In order to tackle uncertainty, metaheuristics have been combined with simulation methods in recent years. The resulting simheuristic approaches, apart from finding cost-effective solutions for the deterministic problem –through the optimization component– are also able to provide efficient solutions for the stochastic scenario (Chica et al., 2020). In many real-life situations, large-scale and complex optimization problems assume different degrees of uncertainty, not only of a stochastic but also of a fuzzy (non-stochastic) nature. The latter might occur, for instance, when the volume of observations is low or the available data have insufficient quality. In this case, the accurate modeling of the uncertainty sources simply does not follow the natural pattern of modeling them only as stochastic variables following a probability distribution. Instead, fuzzy inference systems (FIS) are also considered to achieve this goal.

In this section, we propose a fuzzy simheuristic approach to solve both the VRP and the TOP under general uncertainty scenarios (i.e., those including both probabilistic as well as non-probabilistic uncertainty). This hybrid solution approach combines a multi-start metaheuristic with MCS and FIS to deal with stochastic and fuzzy variables, respectively. Specifically, this solution method is composed of three stages. The first stage refers to the construction of an initial feasible solution through a savings-based constructive heuristic, which is designed considering the characteristics of each problem. The second stage consists of enriching this heuristic with BR decisions (Quintero-Araujo et al., 2017), which are then incorporated into a multi-start framework, in order to generate multiple solutions. This stage, in addition to exploring different regions of the solution space, conducts a short number of simulations on a set of promising solutions in order to evaluate their efficiency under stochastic conditions. Finally, the third stage performs a refinement procedure, in which a larger number of simulation runs are applied to a set of elite solutions. This procedure allows to obtain a more accurate estimation of the different solution properties.

Figure 7.1 depicts a high-level description of the proposed methodology. As explained, this process starts from solving the deterministic problem, whose corresponding solution is submitted to a short simulation procedure, i.e., the exploratory stage. Consequently, new solutions are generated for both the stochastic and the fuzzy environment. These steps are repeated until a stopping criterion is met. Finally, the best-found solutions (or a set of elite solutions) are submitted to a large number of simulation runs –the intensive stage– in order to obtain a more accurate summary of output variables, such as total cost/reward and risk/reliability values.

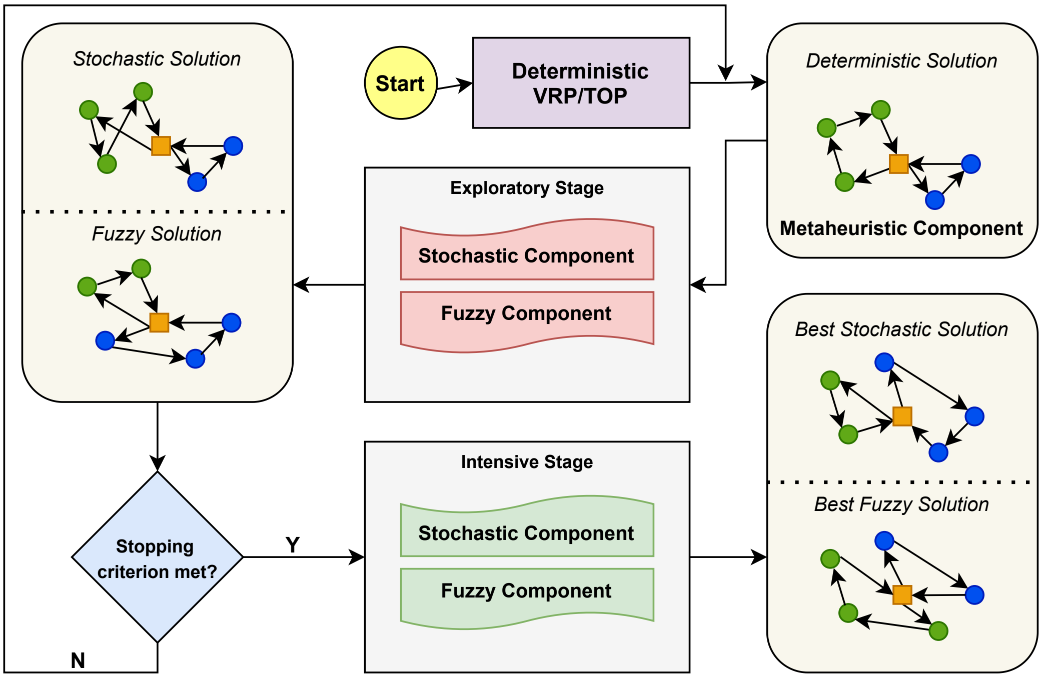


Figure 7.1: High-level flowchart of the proposed solution method.

In order to facilitate reproducibility, the low-level details of each of the stages of the described methodology are provided below:

1. The constructive heuristics for solving the VRP and TOP are based on the savings concept (Clarke and Wright, 1964). Despite being structurally similar for both problems, their particularities are introduced to adquately cope with each respective case, as follows:

    - Firstly, a dummy solution is constructed. This hypothetical solution is composed of a set of routes, each of them being designed to serve one customer. The vehicle departs from the origin depot, visits the customer and continues the trip towards the destination depot. In the case of the TOP, this stage takes into account the maximum tour length when designing these dummy routes. That is, those dummy routes whose total travel time is greater than this limit are automatically discarded. Similarly, in the case of the VRP, this stage takes into account the maximum loading capacity of each vehicle (i.e., if the demand of a customer is higher than this capacity, this customer is discarded).
    - Secondly, a savings list (SL) is created, which includes all the edges connecting two different locations. For each edge $(i, j) \in SL$, a savings value is computed according to Equation (7.1) for the VRP, and Equation (7.2) for the TOP. In both cases, t_{ij} represents the time- or distance-based cost associated with traveling from node i to node j, 0 is the origin depot. In the case of the TOP, n represents the destination depot, while u_i and u_j represent the rewards obtained when

customers i and j are visited for the first time. In the case of the TOP, considering a linear combination of both travel time and reward allows us to define an 'enriched savings' concept that reflects not only the desire of maximizing the total collected reward, but also takes into account how far a customer is from the rest of nodes on the emerging route (Panadero et al., 2020b). Once computed, the SL is sorted in descending order of savings value, which implies that edges with the highest savings are placed at the top of the list.

$$s_{ij} = t_{0i} + t_{j0} - t_{ij} \tag{7.1}$$

$$s_{ij} = \alpha(t_{in} + t_{0j} - t_{ij}) + (1 - \alpha)(u_i + u_j) \tag{7.2}$$

- The sorted SL reflects the most promising movements to reduce the corresponding costs. In this way, the edge at the top of the SL is selected to perform the merging of the associated routes. This procedure is performed only if a feasible combined route is generated. For the VRP, two routes can only be merged if the vehicle capacity is not exceeded. Alternatively, for the TOP, two routes can only be merged if the total travel time does not exceed the maximum tour length. Once the selected savings edge is checked, it is deleted from the SL. Then, the new edge at the top is selected to continue this procedure, which is repeated until the SL is empty. At the end of this process, a feasible solution is generated.

2. The described heuristics are deterministic, which implies that the same decisions are made whenever they start from the same configuration. To change this behavior, these deterministic heuristics are transformed into a probabilistic algorithms by "smoothing" the selection of candidates from the SL using a probability distribution. This concept is called biased-randomization, and is described by Dominguez et al. (2014). In our case, the geometric probability distribution was adopted, as suggested by Ferone et al. (2019). Introducing BR decisions in our heuristics requires dealing with additional parameters, such as the $\beta \in (0, 1)$, which defines the geometric distribution. The value of this parameter was set after a quick tuning process over a random sample, establishing a good performance for both algorithms whenever β falls in the interval $(0.3, 0.4)$. Algorithm 9 describes the heuristic operational structure. Notice that the difference between the two algorithms, designed to solve their respective problems, consists in how the SL is constructed (line 2). Finally, the resulting BR algorithms are embedded into a multi-start framework in order to generate many alternative solutions. Then, the best-found solution is updated and returned at the end of this procedure.

3. The last stage refers to the incorporation of both simulation and fuzzy components into the BR framework, so that promising solutions are processed to estimate their expected costs (Algorithm 10). For the VRP and the TOP, the uncertain variables are the customer demands and the travel times, respectively.

Algorithm 9 A biased-randomized algorithm

```
sol ← createDummySolution(V)
SL ← createSavingsList(sol)
SL ← sort(SL)
while there are edges in SL do
    e ← selectEdgeFromList(β, SL)
    i ← getOrigin(e)
    j ← getEnd(e)
    iRoute ← getEvolvingRouteOfNode(i)
    jRoute ← getEvolvingRouteOfNode(j)
    if all route-merging conditions are satisfied then
        sol ← mergeRoutesUsingEdge(e, iRoute, jRoute, sol)
    end if
    SL ← deleteEdgeFromList(e, SL)
end while
return sol
```

- For stochastic variables, a new value is assigned to each random element based on its probability distribution. For stochastic variables, the MCS is used to estimate them.
- For fuzzy variables, the new value of each element is based on its fuzzy function. Accordingly, fuzzy variables are estimated through the FIS. This procedure is explained more thoroughly in Sections 7.1.3 and 7.2.3.

Algorithm 10 A fuzzy simheuristic

```
initSol ← BiasedRandAlgorithm(V, β)
simulation(initSol, q_short)
bestSol ← initSol
n_iter ← 0
while n_iter < max_iter do
    sol ← BiasedRandAlgorithm(V, β)
    if detCost(sol) < detCost(bestSol) then
        simulation(sol, q_short)
        if expCost(sol) < expCost(bestSol) then
            bestSol ← sol
            Elite ← Elite ∪ {sol}
        end if
    end if
    n_iter ← n_iter + 1
end while
for sol ∈ Elite do
    simulation(sol, q_long)
end for
Elite ← sort(Elite)
bestStochSols ← selectTopSols(Elite)
return bestStochSols
```

Once the deterministic version of each problem is solved, their respective solutions are submitted to a exploratory stage, in which only a low number of simulation (q_{short}) runs are performed to avoid jeopardizing the time of the metaheuristic component (Rabe et al., 2020b). These short simulation runs are applied only to solutions that meet an acceptance criterion (line 8). Altering these stochastic and fuzzy values involves a re-evaluation of both the objective function and the constraints, so that the expected cost/reward of each promising solution can be computed. These short simulation runs allow multiple elite solutions to be found (line 11). In this way, once the BR main loop is completed, a larger number of simulation (q_{long}) runs are executed for each elite solution (line 17). Consequently, the algorithm is able to obtain more accurate values of the output variables. Finally, a reduced set of best-found solutions is returned. From this set, managers can assess not only the expected costs/rewards but also the risk or reliability values associated with each solution, as described by Chica et al. (2020).

7.1.3 Computational Experiments and Results

The proposed fuzzy simheuristic has been implemented using Python 3.8 and tested on a common PC with a multi-core processor Intel *i*7 and using 8 GB of RAM. The algorithm was executed five times with different seeds for a maximum time of 100 seconds for each instance. To the best of our knowledge, there are no instances in the literature for the stochastic-and-fuzzy problems described above. Accordingly, we extend the well-known deterministic benchmarks proposed by Augerat et al. (1995) and Chao et al. (1996) for the VRP and the TOP problems, respectively. The following subsections describe in detail the process used to transform these deterministic benchmarks into stochastic-fuzzy ones.

7.1.3.1 A Fuzzy Approach for the VRP

In order to check the performance of our algorithm, we compare it with some benchmark instances that can be found in the literature. From Augerat et al. (1995), we have chosen 29 classical instances that can be suitable for our study. The nomenclature of the instances is as follows: $'L - nXX - kY'$ where $L \in \{A, B, E\}$ is the set identification, XX denotes the number of customers and Y establishes the number of vehicles. For carrying out the experiments, we assume that the demand d_i of each customer i is uncertain and, therefore, we model it as either a stochastic or a fuzzy variable.

Regarding the stochastic scenario, the instances are extended by considering that the stochastic demand D_i follows a log-normal probability distribution. The parameters of this distribution are adjusted according to the mean $E[D_i] = d_i \ \forall i \in N$, where d_i is the deterministic demand, and the variance $Var[D_i] = c \cdot d_i$. The parameter c is a design parameter that allows us to set up the level of uncertainty. It is expected that, as c converges to zero, the results of the stochastic version will converge to those obtained in the deterministic scenario. In our experiments, we employ the value $c = 0.25$, which introduces a medium level of uncertainty. Regarding the fuzzy scenario, we consider the demand D_i for each customer i as a fuzzy variable. This demand can be estimated as low, medium, or high (DL, DM, DH, respectively). Likewise, we assume that the vehicle available capacity after serving the customer i, C_i, is an input variable of the fuzzy system. Besides, each of the aforementioned demand levels is defined by a triangular fuzzy number $D_i = (d_{1i}, d_{2i}, d_{3i})$. Figure 7.2 shows the membership functions of these fuzzy sets. Similarly, the available vehicle capacity C_i after serving the customer i is represented by a triangular fuzzy number $C_i = (c_{1i}, c_{2i}, c_{3i})$, which takes the values low, medium, or high (CL, CM, CH, respectively). Figure 7.3 displays the membership function of the capacity fuzzy sets. Note that both the demands and the available capacities are expressed as a percentage of the total vehicle capacity, i.e., $0 \leq D_i \leq 1$ and $0 \leq C_i \leq 1$.

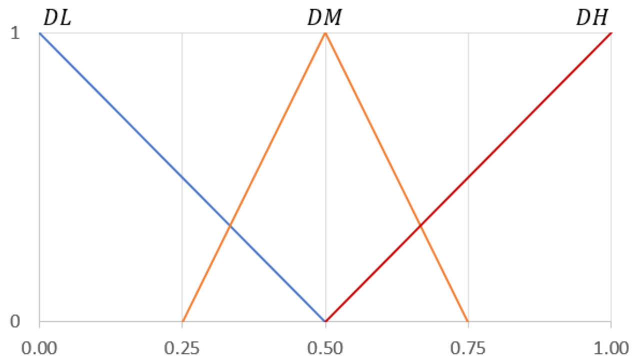


Figure 7.2: Fuzzy sets for the customer i demand.

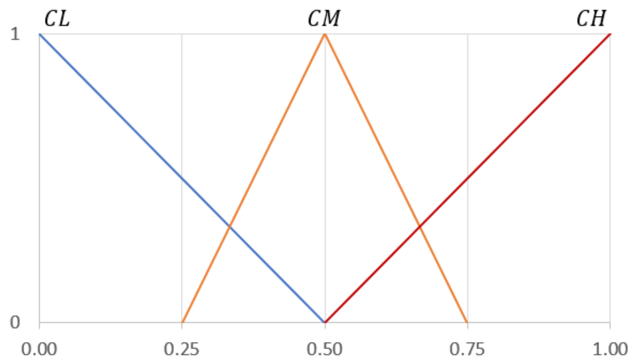


Figure 7.3: Fuzzy sets for the available capacity after visiting customer i.

For each node i, we define a preference index, p_i, as the output of the fuzzy system, such that $0 \leq p_i \leq 1$. When this index takes the maximum value ($p_i = 1$) then the next node of a route will be visited for sure as the available capacity C_i of the vehicle can meet the demand D_{i+1}. Moreover, if $p_i = 0$, then we are sure that $D_{i+1} > C_i$ and, consequently, the vehicle needs a replenishment at the depot. The preference index is classified into very low (PVL), low (PL), medium (PM), high (PH) and very high (PVH) levels. The membership function related to each of these categories can be seen in Figure 7.4. The reasoning rules that determine the preference to travel to the next node –depending on the levels of the demand and the available capacity– are featured in Table 7.1. After performing a set of fine-tuning experiments, we set the threshold value to visit the next node to $p^* = 0.25$. This means that whenever the calculated p_i is greater than 0.25, the next node will be visited; otherwise, the vehicle will return to the depot for a replenishment. The calculation of a specific value for p_i requires converting the input variables into a crisp value. Hence, the estimated crisp values of the demand and the available capacity, the membership functions and the reasoning rules are employed in a fuzzification-defuzzification process to obtain the preference index. In our case, the defuzzification method applied was the well-known center of gravity method to obtain the output crisp value.

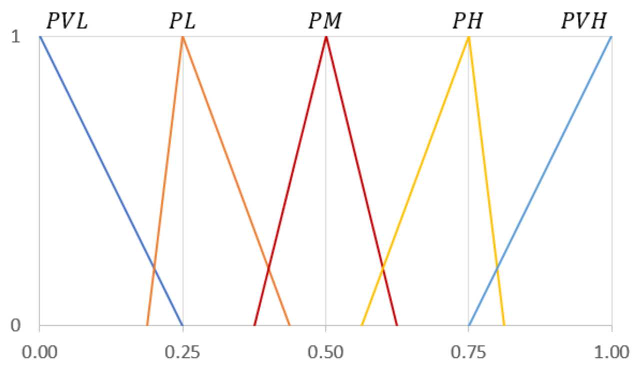


Figure 7.4: Fuzzy sets for the preference strength to travel to customer i.

Table 7.1: Reasoning rules determining the visit preference strength for the VRP.

Demand	Available capacity		
	CL	CM	CH
DL	PM	PH	PVH
DM	PL	PM	PH
DH	PVL	PL	PM

7.1.3.2 A Fuzzy Approach for the TOP

The employed deterministic benchmark contains a total of 320 instances that are distributed in 7 subsets. The instances are identified following the nomenclature $'pa.b.c'$, where a represents the subset, b defines the number of available vehicles, and c identifies the specific instance under study. For experimentation purposes, we consider that the uncertainty is located in the travel times between two pairs of nodes. To extend the instances to be employed in the stochastic scenarios, we assume that travel times, T_{ij}, follow a log-normal probability distribution. In setting up the stochastic instances, we assume that $E[T_{ij}] = t_{ij}$, $\forall (i, j) \in N$, where t_{ij} is the travel time for the corresponding deterministic instance. We set the variability in the travel times with reference to the deterministic travel time such that $Var[T_{ij}] = c \cdot t_{ij}$, and $c \geq 0$. As in the VRP problem, we employ the value $c = 0.25$ to induce a medium level of uncertainty in the travel times. With the aim of extending the instances to be used also in fuzzy scenarios, we consider the travel times for each pair of nodes, t_{ij}, as a fuzzy variable. This variable is modeled using a fuzzy inference system. We assume the case of electric vehicles and use their battery levels, as well as the reward of each node, as the input variables of the fuzzy system. The battery level (Q) of each vehicle can be estimated as low (QL), medium (QM), or high (QH). The low and high levels are represented by a triangular fuzzy number $Q = (q_1, q_2, q_3)$, while the medium level follows a trapezoidal fuzzy number $Q = (q_1, q_2, q_3, q_4)$. All battery values are expressed as a proportion of the total battery level, i.e., $0 \leq Q \leq 1$. The membership function of this fuzzy set is displayed in Figure 7.5. Similarly to the battery level, the reward of each node has been categorized using three fuzzy sets: low (RL), medium (RM), or high (RH), where each of them follows a triangular distribution. The reward values have been represented as a proportion of the maximum reward that can be collected at any node of all the possible nodes to be visited.

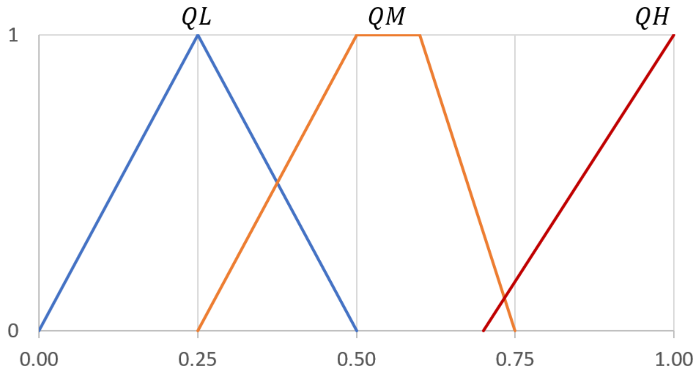


Figure 7.5: Fuzzy sets for the battery of each vehicle.

Finally, the output of the fuzzy system gives a preference index, p_i, which indicates the inclination to visit the next node in the route. This index depends on both the reward of the next node and the remaining battery of the vehicle. This preference index has been defined between 0 and 1, i.e., $0 \leq p_i \leq 1$. When $p_i = 1$, the vehicle will definitely visit the next node in the route, since the vehicle will reach the node. On the contrary, when $p_i = 0$, we are sure that the vehicle will not reach the next node, and the vehicle will remain in the current node. In this case, the route will present a failure, because the vehicle fails to reach the final depot and, consequently, the total reward of the route has been only partially collected. We have classified the preference as: very low (PVL), low (PL), medium (PM), high (PH), or very high (PVH). Each of these categories is represented by a fuzzy set. Finally, we establish a set of reasoning rules (Table 7.2), which describe the knowledge needed to determine the preference to visit the next node. After a quick fine-tuning process, we set the threshold value for visiting the next node to $p^* = 0.45$. Note that this is a sensitive value, as a larger value could lead to generating overly conservative routes, while a value close to 0 could lead to risky decisions. In order to transform the input variables into a crisp value, the contribution of each membership function is combined on the inference, while a union operator is used to determine the output distribution. Subsequently, the center of gravity method is applied in order to obtain a crisp output value corresponding to the preference value.

Table 7.2: Reasoning rules determining the visit preference strength for the TOP.

Battery	**Reward**		
	RL	RM	RH
QL	PVL	PL	PM
QM	PL	PM	PH
QH	PM	PH	PVH

7.1.3.3 Results and Discussion

Tables 7.3 and 7.4 display the results of selected instances with different characteristics for the VRP and the TOP problems, respectively. In the case of the VRP and the TOP, the results –with the exception of the gap column– are measured in terms of distance and reward units, respectively. The first column of the tables identifies the instances. We have divided the remaining columns into three different parts. Firstly, our best-found deterministic solutions (OBD) are presented. These solutions do not consider stochastic or fuzzy variables, instead, they refer to the deterministic version of the problem. We compare the gap of our solutions (column 2) with respect to the best-known solutions (column 1). In the second part of the table, we present the obtained solutions for the stochastic scenario. Column 3 displays the expected cost when the OBD is evaluated under a stochastic scenario, with the corresponding level of uncertainty. To compute the expected cost, a simulation process is applied to the OBD solution. Similarly, the next column shows the expected cost obtained using our simheuristic approach for the stochastic version of the problem. The last part of the table reports the results obtained considering fuzzy scenarios. Thus, column 5 reports the best hybrid fuzzy-stochastic solutions. To compute these solutions, we assume that half of the nodes follow a log-normal distribution, and the remaining half are considered to be fuzzy. In the case of the TOP, where the uncertainty is related to the edges, we have considered the origin node to evaluate the type of uncertainty. Finally, the last column of the table reports the solutions obtained in a scenario with a high level of uncertainty, where all the uncertain variables are considered as fuzzy. Notice that, although the goal of this section is not to solve the deterministic version of the problem, the results show that our approaches are highly competitive for the deterministic version of both problems. For the VRP problem, we obtain an average gap of 0.39%, with a maximum gap of 1.27%. Furthermore, the obtained gap is 0.0% for the TOP problem. These results highlight the quality of our base algorithms, which constitutes the optimization component in our fuzzy simheuristic, validating their potential to be used in uncertainty scenarios.

Regarding the uncertainty scenarios, which represent the main contribution of this section, Figures 7.6 depict an overview of Tables 7.3 and 7.4, respectively. In these box plots, the vertical axis represents the gap that was obtained in the stochastic and fuzzy scenarios with respect to the OBD solution, which is used here as a reference value. The latter can be considered as an ideal scenario with perfect information, which is not the case in scenarios with stochastic or fuzzy components. Concerning the stochastic solutions, the results show that those provided by the simheuristic clearly outperform the solutions of the deterministic version of the problem when these are simulated for all the considered problems, i.e., using the OBD solutions for the stochastic scenario is not a good idea, since it leads to sub-optimal solutions. On average, an improvement of about 7.91% is observed for the VRP problem, while an improvement of about 1.72% is reported for the TOP problem . These results justify the importance of using hybrid simulation-fuzzy methods when dealing with optimization problems under uncertainty.

Regarding the fuzzy scenarios, Figure 7.7 displays a summary of the presented results for

Table 7.3: Comparison of results, in terms of traveled distance, for the different VRP scenarios.

Instance	Deterministic Scenario			Stochastic Scenario		Fuzzy Scenario	
	BKS (1)	OBD Sol. (2)	GAP (%) (1)–(2)	Det Sol. (3)	Stochastic Sol. (4)	Stoch-Fuzzy Sol. (5)	Fuzzy Sol. (6)
A-n32-k5	787.1	787.2	0.01%	1117.0	797.5	1119.0	1245.4
A-n33-k5	662.1	662.1	0.00%	895.4	667.8	811.6	988.8
A-n33-k6	742.7	742.7	0.00%	860.6	755.0	868.6	990.8
A-n37-k5	672.5	674.2	0.26%	974.8	682.4	926.8	1044.3
A-n38-k5	733.9	739.7	0.79%	768.4	761.5	945.0	1118.5
A-n39-k6	833.2	835.2	0.24%	835.7	835.3	1118.1	1236.4
A-n45-k6	952.2	957.1	0.51%	1074.2	1030.0	1264.9	1510.4
A-n45-k7	1147.4	1156.4	0.79%	1251.6	1161.2	1438.8	1754.7
A-n55-k9	1074.5	1085.9	1.06%	1128.6	1126.9	1292.8	1584.8
A-n60-k9	1360.6	1365.8	0.38%	1380.4	1371.5	1791.1	2135.4
A-n61-k9	1040.3	1049.0	0.83%	1128.4	1118.6	1361.2	1622.5
A-n63-k9	1633.7	1641.0	0.45%	1720.9	1689.4	2248.6	2847.1
A-n65-k9	1184.7	1195.2	0.89%	1332.2	1271.5	1594.6	1936.4
A-n80-k10	1776.2	1792.7	0.93%	2013.0	1838.5	2761.4	3334.5
B-n31-k5	676.1	676.5	0.06%	697.5	677.3	693.2	1068.4
B-n35-k5	958.9	959.4	0.05%	1053.4	1033.2	1215.4	1588.0
B-n39-k5	553.2	553.7	0.08%	585.6	584.6	855.3	1037.3
B-n41-k6	835.8	840.8	0.60%	924.9	862.8	1152.2	1344.5
B-n45-k5	754.0	754.7	0.10%	776.4	768.3	1085.0	1202.7
B-n50-k7	744.2	744.2	0.00%	789.4	778.1	991.7	1235.8
B-n52-k7	754.5	756.8	0.31%	852.1	763.8	1084.5	1355.6
B-n56-k7	716.4	719.4	0.42%	802.6	735.5	1023.0	1340.3
B-n57-k9	1602.3	1603.8	0.09%	1915.8	1700.0	2047.6	2723.9
B-n68-k9	1300.2	1306.5	0.48%	1491.7	1359.1	1776.7	2348.9
B-n78-k10	1250.6	1256.6	0.48%	1413.0	1383.9	1759.8	2146.6
E-n22-k4	375.3	375.3	0.00%	375.3	375.3	376.4	502.3
E-n30-k3	505.0	505.0	0.00%	505.0	507.9	742.4	838.0
E-n33-k4	837.7	839.4	0.21%	839.7	839.7	1183.0	1506.9
E-n76-k10	841.3	852.0	1.27%	926.0	902.9	1194.6	1336.8
Average	**941.6**	**945.8**	**0.39%**	**1049.3**	**978.6**	**1266.3**	**1549.2**

different problems. The vertical axis represents the gap obtained for the different optimization methods with respect to the OBD solution. This figure shows that the solution quality worsens as the uncertainty level increases. This is due to route failures occurring during the execution stage, which penalize the entire route and, therefore, cause an extra cost. Figures 7.8–7.11 illustrate a numerical example for the VRP instance *A-n80-k10*. Figure 7.8 depicts the configuration of the deterministic solution and its associated cost (1797.05). This cost can be seen as a lower-bound reference value in a scenario with perfect information. Figures 7.9–7.11 show a representation of the obtained solution considering different levels of uncertainty. As the level of uncertainty increases, the total cost also increases, i.e., the highest cost (1860.94) is reached in the completely fuzzy scenario, where the solution cost has increased up to 3.43% with respect to the deterministic solution. This extra cost is mainly caused by the rise in failure costs, since a greater number of detours and round-trips are expected when the uncertainty in the demand at each node is higher. Notice also that these solutions have different configurations in each scenario, since the optimization algorithm

Table 7.4: Comparison of results, in terms of collected reward, for the different TOP scenarios.

Instance	Deterministic Scenario			Stochastic Scenario		Fuzzy Scenario	
	BKS (1)	OBD Sol. (2)	GAP (%) (1)–(2)	Det Sol. (3)	Stochastic Sol. (4)	Stoch-Fuzzy Sol (5)	Fuzzy Sol. (6)
p1.2.f	80	80	0.0	78.9	79.3	77.4	76.8
p1.2.i	135	135	0.0	127.6	129.4	125.2	117.9
p1.2.k	175	175	0.0	169.3	174.4	164.6	143.2
p1.2.n	235	235	0.0	228.8	232.7	216.2	189.7
p1.3.n	190	190	0.0	182.5	189.6	180.0	174.1
p1.4.j	75	75	0.0	59.3	63.3	51.3	45.1
p1.4.k	100	100	0.0	98.3	99.9	99.9	95.8
p1.4.l	120	120	0.0	118.9	119.2	118.6	117.1
p1.4.m	130	130	0.0	98.2	102.9	91.5	86.3
p1.4.n	155	155	0.0	99.9	104.0	107.5	98.2
p1.4.o	165	165	0.0	159.4	164.2	148.6	143.6
p1.4.p	175	175	0.0	171.3	174.4	163.7	155.5
p2.2.d	160	160	0.0	150.6	150.6	150.0	137.5
p2.2.i	230	230	0.0	223.4	226.3	228.5	204.0
p2.3.i	200	200	0.0	191.5	195.2	186.7	177.9
p3.2.c	180	180	0.0	179.1	179.2	178.9	160.2
p3.2.d	220	220	0.0	212.3	217.5	197.7	179.8
p3.2.g	360	360	0.0	358.3	358.8	308.2	297.9
p3.2.q	760	760	0.0	748.5	755.2	663.4	630.6
p3.2.r	790	790	0.0	768.3	774.9	656.1	638.5
p3.3.e	200	200	0.0	198.2	199.0	195.7	187.6
p3.4.g	220	220	0.0	212.6	217.3	205.0	191.3
p5.2.d	80	80	0.0	75.5	77.4	73.7	70.7
p5.2.k	670	670	0.0	643.3	662.1	646.0	612.5
p5.2.p	1150	1150	0.0	1135.4	1138.1	1105.1	1073.0
p5.3.f	110	110	0.0	107.4	109.1	107.6	103.0
p5.3.o	870	870	0.0	856.2	865.1	836.9	806.9
p5.4.g	140	140	0.0	135.3	137.9	134.3	129.2
p5.4.t	1160	1160	0.0	1139.5	1148.4	1107.4	1068.1
p5.4.u	1300	1300	0.0	1279.5	1286.3	1239.2	1198.2
p6.2.d	192	192	0.0	185.4	188.1	177.8	164.2
p6.2.e	360	360	0.0	276.4	297.2	285.4	277.9
p6.2.f	588	588	0.0	577.4	580.0	519.5	501.2
p6.2.g	660	660	0.0	648.3	650.5	584.9	569.2
Average	**362.8**	**362.8**	**0.0**	**349.9**	**354.3**	**333.3**	**318.3**

attempts to generate routes that reduce the risk of failure.

7.2 The LRP with Facility-Sizing Decisions and Stochastic and Fuzzy Demands

As explained in Sections 4.3 and 6.3, when designing and managing supply chains, one of the most relevant problems is the simultaneous location of distribution facilities and the routing of vehicles to deliver products to a set of geographically dispersed customers. The former is considered a strategic decision, while the latter is operational. This problem is

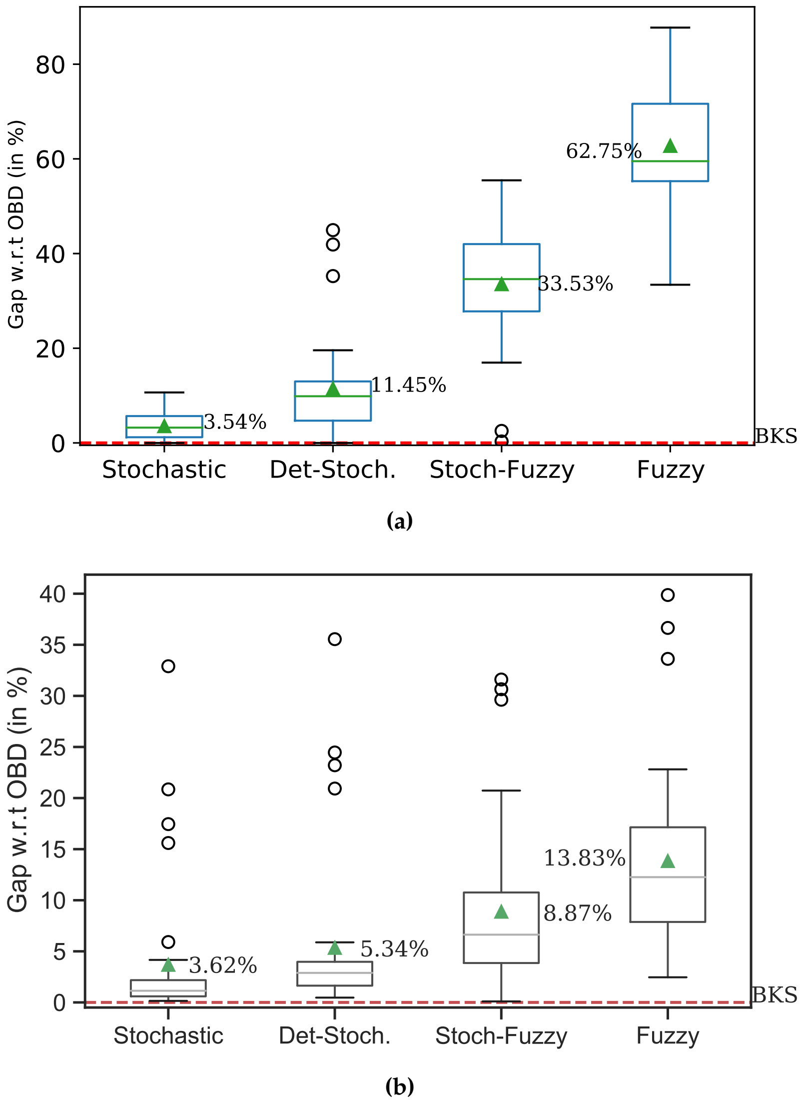


Figure 7.6: Gaps of different optimization methods with respect to the OBD solution. (**a**) Results for the *VRP* dataset. (**b**) Results for the *TOP* dataset.

known in the scientific literature as the location routing problem. One of the most studied versions of the LRP is the capacitated LRP, in which both depot and vehicle capacity constraints must be satisfied. However, the vast majority of previously published works consider the depot capacity as a fixed value for each location. This could not be a suitable approach when dealing with realistic problems, since it is usual that decision-makers can select the size of a facility from a discrete set of known available sizes, or even freely.

Traditional LRP approaches consider that parameters are deterministic or crisp, i.e., they assume that inputs are known in advance. Hence, the literature on the LRP addressing uncertain parameters is still scarce. In order to overcome this problem, articles employing stochastic approaches can be found in the literature. Customers' demand is one of the most addressed stochastic parameters (Quintero-Araujo et al., 2019b; Rabbani et al., 2019; Sun et al., 2019; Zhang et al., 2019b). Other parameters might also be considered as stochastic, such as transportation costs and travel speeds (Herazo-Padilla et al., 2015) or logistic costs

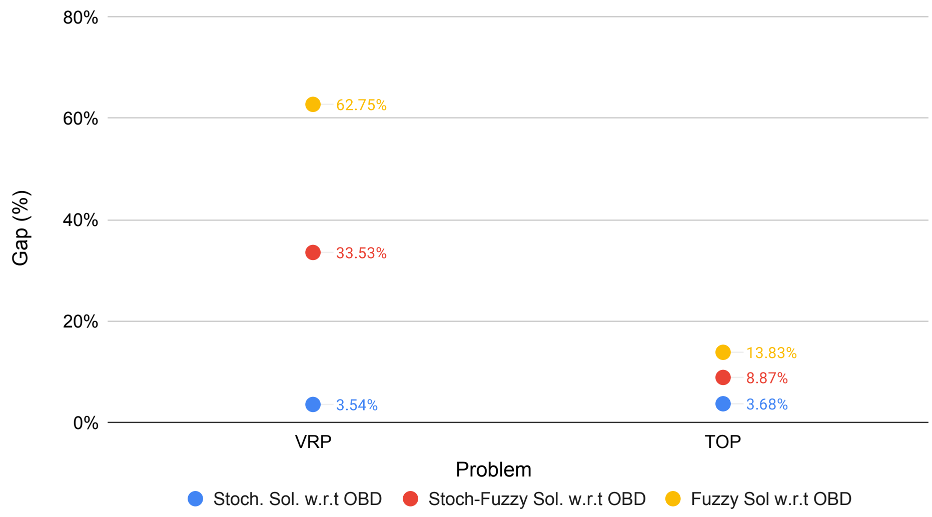


Figure 7.7: Gaps of different optimization methods with respect to the OBD solution.

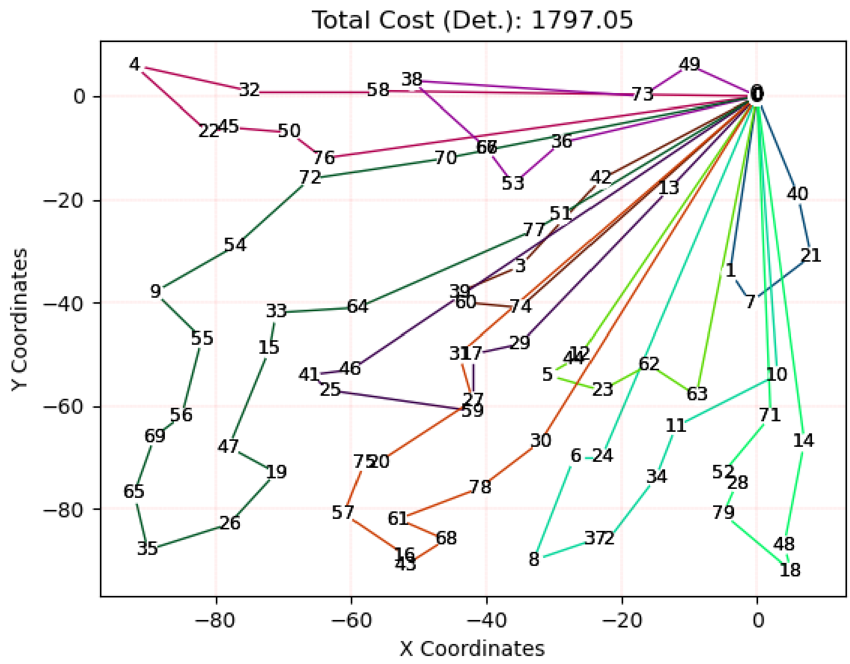


Figure 7.8: Best solution for the VRP deterministic scenario.

and travel distance (Zhang et al., 2018). In general, many articles addressing stochasticity in routing problems hybridize simulation models with heuristic or metaheuristic algorithms to tackle efficiently both uncertainty and NP-hardness. In many real-life situations, however, it might not be possible to accurately model all uncertainty sources as stochastic variables following a probability distribution. This might be the case, for instance, when the volume of observations is low or the available data does not have enough quality (Corlu et al., 2020). Hence, uncertainty in the LRP has also been tackled through the use of fuzzy sets. Parameters such as customers' demands (Zhang et al., 2020a; Mehrjerdi and Nadizadeh, 2013; Fazayeli et al., 2018; Nadizadeh and Kafash, 2019), travel times (Zarandi et al., 2011; Zarandi

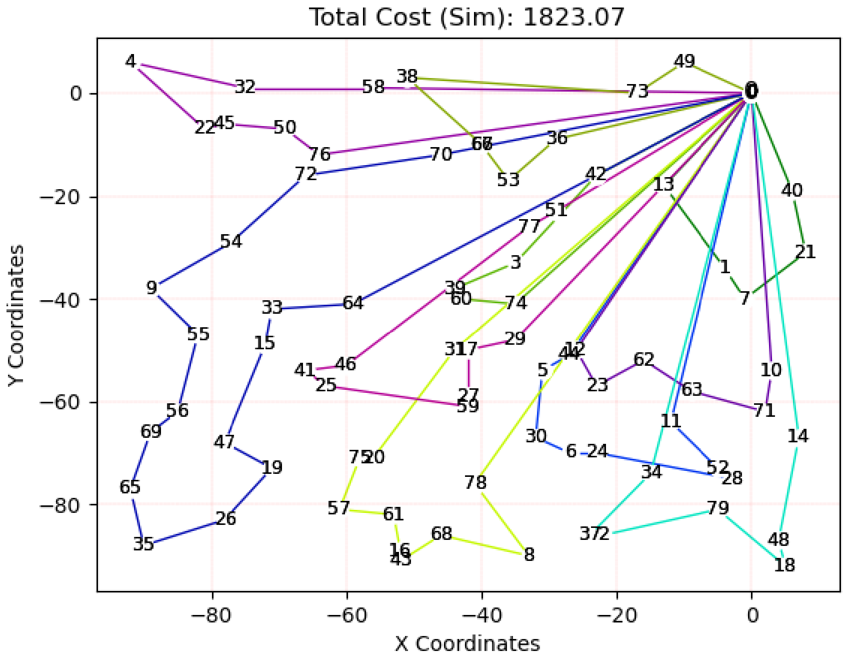


Figure 7.9: Best solution for the VRP stochastic scenario.

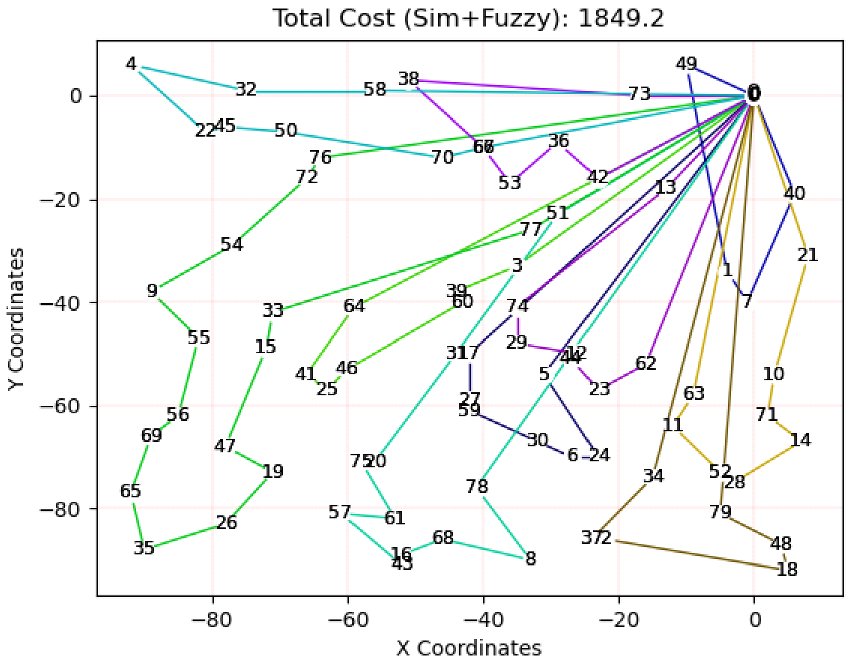


Figure 7.10: Best solution for the VRP stochastic and fuzzy scenario.

et al., 2013) or time windows (Ghezavati and Morakabatchian, 2015) have been modeled as fuzzy in several studies. Notice that, whenever possible, modeling uncertainty as stochastic variables might allow a deeper statistical analysis of the results.

To the best of our knowledge, there are no works in the literature simultaneously addressing stochastic and fuzzy approaches to model demand uncertainty in a flexible-size

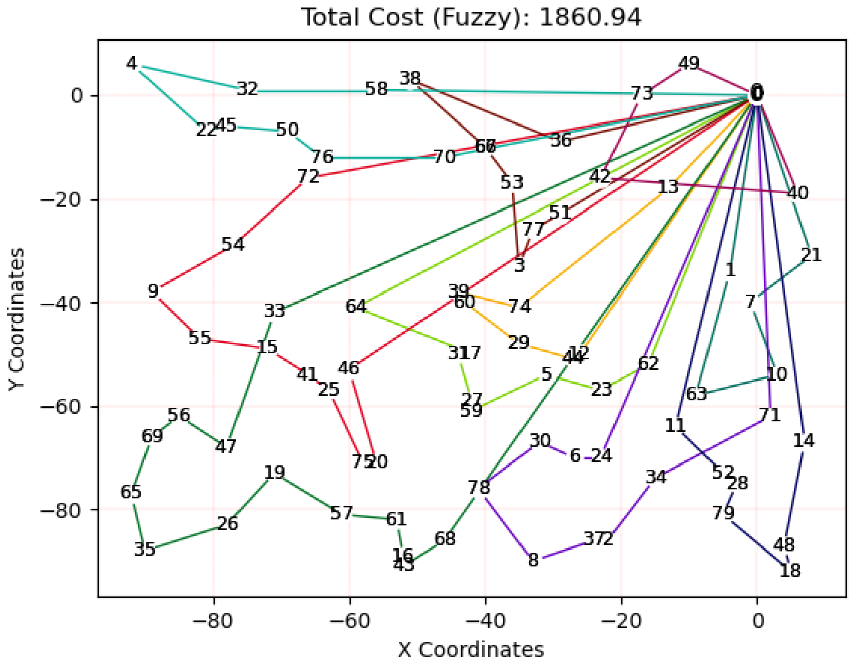


Figure 7.11: Best solution for the VRP fuzzy scenario.

LRP. This is a realistic scenario, since many companies might have historical data on trustworthy customers and not enough data on new or unreliable ones. Hence, the main contributions of this section are twofold: on the one hand, a new variant of the location routing problem is studied, where facility sizing decisions and hybrid fuzzy-stochastic demands are simultaneously considered. On the other hand, this section proposes a competitive solution approach based on the hybridization of a metaheuristic algorithm with both simulation and fuzzy logic, i.e., a so-called fuzzy simheuristic, to solve the aforementioned problem. Indeed, simheuristics have been traditionally proposed to deal with stochastic issues in hard COPs (Juan et al., 2018). However, their hybridization with fuzzy logic has been rarely studied.

7.2.1 Problem Definition

Most characteristics of the problem addressed in this section have been already described in Subsection 6.3.1. Basically, we consider an LRP with facility sizing decisions, where the size of each open facility is also a variable to decide on. Furthermore, we also consider both stochastic and fuzzy demands. If I is the set of customers, the customers' demands are uncertain and are modeled using stochastic values for a subset of customers I_1, and fuzzy values for a subset of customers I_2, such that $I_1 \cup I_2 = I$. The LRP with facility sizing decisions and uncertain demands can be formulated as a mathematical programming model, whose sets, parameters, and variables are shown in Table 7.5.

The objective is to minimize the total cost (TC), which includes opening facilities costs (OC), routing costs (RC), and failure costs (FC), i.e., $TC = OC + RC + FC$. These parts are defined in Equations (7.3)–(7.5).

Table 7.5: Sets, parameters, and variables for the LRP with facility sizing decisions and uncertain demands.

Sets
V = Set of nodes
K = Set of vehicles
L = Set of available sizes
I = Set of customers, $I \subset V$
J = Set of depots, $J \subset V$
A = Set of arcs, $A = V \times V = \{(m,n) : m \in V, n \in V \wedge m \neq n\}$
$\delta^+(S)$ = Set of arcs leaving S, $S \subset V$, $\delta^+(S) \subset A$
$\delta^-(S)$ = Set of arcs entering S, $S \subset V$, $\delta^-(S) \subset A$
Parameters
s_{jl} = Available size of type $l \in L$ for the depot $j \in J$
D_i = Uncertain demand of customer $i \in I$
f_j = Fixed opening cost of depot $j \in J$
o_{jl} = Variable opening cost of depot $j \in J$ with size of type $l \in L$
c_a = Cost of traversing arc $a \in A$
q = Capacity of each vehicle
$\%SS$ = Safety stock percentage
Variables
y_{jl} = Binary variable equal to 1 if depot $j \in J$ is open with size of type $l \in L$, 0 otherwise
x_{ij} = Binary variable equal to 1 if customer $i \in I$ is assigned to depot $j \in J$, 0 otherwise
w_{ak} = Binary variable equal to 1 if arc $a \in A$ is used in the route performed by vehicle $k \in K$, 0 otherwise

$$OC = \sum_{j \in J} \sum_{l \in L} (f_j + o_{jl}) y_{jl} \tag{7.3}$$

$$RC = \sum_{e \in E} \sum_{k \in K} c_e w_{ek} \tag{7.4}$$

$$FC = \min\{c_{reac}, c_{prev}\} \tag{7.5}$$

FC represents the cost incurred whenever the actual demand of a route is greater than the vehicle capacity, where c_{reac} and c_{prev} depend on the corrective action considered, namely:

1. A reactive strategy with a cost c_{reac}, in which a vehicle must perform a round-trip to its assigned facility for a replenishment if the actual current-customer demand is higher than the vehicle's current load.

2. A preventive strategy with a cost c_{prev}, in which a vehicle must perform a detour to the facility before visiting the next customer. The decision about performing this detour depends on the type of demand of the next customer. If the demand is stochastic, the detour is carried out whenever the expected demand of the next customer is higher than the current load of the vehicle. Alternatively, if the demand is fuzzy, this decision depends on the comparison between the fuzzy values of both the demand of the next customer and the current load.

Hence, the location routing problem with facility sizing decisions and uncertain demands can be modeled as the following integer program:

$$Minimize \;\; TC \tag{7.6}$$

$s.t.$

$$\sum_{k\in K}\sum_{a\in\delta^-(i)} w_{ak} = 1 \quad \forall i \in I \tag{7.7}$$

$$\sum_{i\in I}\sum_{a\in\delta^-(i)} D_i w_{ak} \leq (1-\%SS)q \quad \forall k \in K \tag{7.8}$$

$$\sum_{a\in\delta^+(n)} w_{ak} = \sum_{a\in\delta^-(n)} w_{ak}, \quad \forall k \in K, \forall n \in V \tag{7.9}$$

$$\sum_{a\in\delta^+(J)} w_{ak} \leq 1 \quad \forall k \in K \tag{7.10}$$

$$\sum_{a\in A(S)} w_{ak} \leq |S|-1 \quad \forall S \subseteq I, \forall k \in K \tag{7.11}$$

$$\sum_{a\in\delta^+(j)} w_{ak} + \sum_{a\in\delta^-(i)} w_{ak} \leq 1 + x_{ij} \quad \forall i \in I, \forall j \in J, \forall k \in K \tag{7.12}$$

$$\sum_{j\in J} x_{ij} = 1 \quad \forall i \in I \tag{7.13}$$

$$\sum_{i\in I} D_i x_{ij} \leq \sum_{l\in L} s_l y_{jl} \quad \forall j \in J \tag{7.14}$$

$$\sum_{l\in L} y_{jl} \leq 1 \quad \forall j \in J \tag{7.15}$$

$$\forall\, y_{jl}, x_{ij}, w_{ak} \in \{0,1\} \tag{7.16}$$

The objective function (7.6) minimizes the total cost. Constraint (7.7) ensures that each customer is served by a single route and a single vehicle. Constraint (7.8) guarantees that the total demand served by a vehicle in a route does not exceed its capacity. This limit is reduced by a safety stock, which is a percentage of the vehicle capacity reserved to respond more effectively to the uncertain demand. Constraint (7.9) guarantees the continuity of each route. Constraint (7.10) ensures the return of each vehicle to its starting depot. Constraint (7.11) guarantees the subtour elimination. Constraint (7.12) ensures that a customer is served by a route departing from an open depot only if this customer is allocated to this depot. Constraint (7.13) guarantees that a customer is assigned to only one depot. Constraint (7.14) ensures that the total demand served from a depot does not exceed its assigned size. Constraint (7.15) guarantees that only one size is assigned to an open depot. Finally, Constraint (7.16) determines that all decision variables are binary.

7.2.2 Solution Approach

Since the problem described in Section 7.2.1 is known for being NP-hard, the formulated mathematical model is not employed to find an optimal solution but just to provide a better understanding of the problem details. Hence, we propose a fuzzy simheuristic approach for minimizing the expected total cost. Traditionally, simheuristics have been used to solve optimization problems with stochastic components, such as ARPs with stochastic demands (Gonzalez-Martin et al., 2018), stochastic waste collection problems (Gruler et al., 2017a) or TOPs with stochastic travel times (Panadero et al., 2020b). We have extended the simheuristic framework by including fuzzy components in order to deal with COPs with uncertainty components of both stochastic and non-stochastic nature. In particular, our methodology combines an ILS metaheuristic with MCS and FIS to deal with stochastic and fuzzy variables, respectively. As discussed by Ferone et al. (2019), several metaheuristic frameworks offer a well-balanced combination of efficiency and relative simplicity and can be easily extended to a fuzzy simheuristic. In general, our approach is composed of three stages. During the first stage, a set of promising LRP solutions are generated using a constructive heuristic, which employs BR techniques (Ferrer et al., 2016). In the second stage, the ILS metaheuristic tries to improve each of these promising solutions by iteratively exploring the search space and conducting a short number of simulations. Finally, in the third stage, a refinement procedure using a larger number of simulation runs is applied to these elite solutions, which allows to obtain a more accurate estimation of the expected total cost.

Algorithm 11 outlines the main components of Stage 1. It generates quickly a ranked list of "promising" LRP solutions. The main input parameters of this heuristic are: the list of customers with both their demand and location in Cartesian coordinates, the list of facilities including their opening costs and the vehicle capacity. The algorithm procedure is as follows: initially, the minimum and maximum ($nbDepots_0$ and $maxNbDepots$, respectively) numbers of facilities required to serve the total demand are computed. Both bounds are calculated by dividing the total demand by the maximum available facility size, and the minimum available facility size, respectively, and they are rounded up to the next integer number. Then we run our algorithm for each number of facilities between $nbDepots_0$ and $maxNbDepots$ (line 3). Later, for each iteration of the line 4 loop, a new set of random locations are generated (line 5). This is stored in $usedOpenDepots$ to avoid repeating. Next, if the available capacity of facilities in $openDepots$ is enough to satisfy customers demand, customers' allocation and routing procedures are carried out; otherwise, $openDepots$ is rejected. The customers' allocation procedure is performed by producing a new map (line 9) where each facility has a list of all customers sorted by savings. These savings represent the benefit of allocating each customer to the current depot instead to the best alternative facility. Then a facility in $openDepots$ is selected randomly, and a BR procedure is used to allocate a customer of the list to the current depot. This procedure ends when all customers have been allocated. In the step in line 10 a VRP is solved for each subset facility-customers in the map. Finally, a feasible LRP solution is yielded and stored in the pool of solutions $poolSol$. The algorithm ends returning a top list of complete LRP solutions, assessed in terms of opening and routing costs.

Algorithm 12 outlines Stages 2 and 3. During the second stage, each "promising" map generated by the constructive heuristic is processed by the simulation and the fuzzy components to estimate its safety stock (line 4). This procedure is carried out by performing a

Algorithm 11 Constructive heuristic

```
usedOpenDepots ← ∅
⟨nbDepots_0, maxNbDepots⟩ ← computeDepotsBound(depots)
for nbDepots ← nbDepots_0 to maxNbDepots do
    for iter ← 1 to iter_max do
        openDepots ← depotsToOpen(nbDepots)
        if openDepots ∉ usedOpenDepots then
            if capacity(openDepots) ≥ demand(cust) then
                usedOpenDepots ← add(usedOpenDepots, openDepots)
                map ← allocateCustomers(openDepots, cust)
                lrpSol ← CWS(map, β, vehCap)
                poolSol ← add(poolSol, lrpSol)
            end if
        end if
    end for
end for
return sortingByCost(poolSol)
```

low number of runs, where a new value is assigned to each random or fuzzy element based on its probability distribution or fuzzy function, respectively. We use MCS in order to estimate the stochastic variables, whilst a fuzzy inference system is used to estimate the fuzzy variables. Then, the objective function and the constraints are evaluated under the random/fuzzy generated values to compute the expected cost of each promising map. Next, the ILS metaheuristic tries to improve the set of "promising" maps by iteratively exploring the search space and conducting a second process of fuzzy/simulation runs. We start the process by perturbing the current base solution *baseSol* (line 8). In this phase we use two different strategies. In the first one, the algorithm randomly selects a set of customers and tries to reassign them in a random way to another facility without violating its capacity. Regarding the second strategy, the algorithm randomly exchanges the allocation of a percentage of customers among facilities. This process is dependent on the value of k, which represents the degree of exchange to be applied. This value is updated in each iteration between K_{min} and K_{max}, i.e., it is reset to K_{min} whenever a new solution *newSol* outperforms the *baseSol*, and it is increased whenever the algorithm fails to improve the current solution until a maximum value K_{max}. The strategy to be used in each iteration of the algorithm is randomly selected.

Afterwards, the algorithm starts a local search around the perturbed solution in order to improve it (line 9). This stage consists in a *two-opt inter-route* operator, which interchanges two chains of randomly selected customers between different facilities. A *newSol* is returned whenever no more improvements are achieved. Later, whenever the deterministic cost of the *baseSol* is improved (line 10), the *newSol* is processed by the simulation and the fuzzy components to deal with the uncertainty of the proposed problem, using a low number of runs to compute the expected cost of the solution (line 11). Notice that this procedure does not only provide estimated values to the expected cost associated with the solutions generated by our approach, but it also reports feedback to the metaheuristic search process. If the *newSol* is also able to improve the expected cost of the *baseSol* (line 12), the latter is updated. In the same way, if the expected cost of the *newSol* improves the cost of the

Algorithm 12 ILS-based fuzzy simheuristic

```
1: initSol ← genInitSol(inputs,α, β)
2: baseSol ← initSol
3: bestSol ← baseSol
4: fastSimulation(baseSol)
5: T ← T_0
6: while time ≤ t_max do
7:     k ← K_min
8:     perturbationSol ← perturbation(baseSol, k, α, β)
9:     newSol ← localSearch(perturbationSol)
10:    if detCost(newSol) < detCost(baseSol) then
11:        fastSimulation(newSol)
12:        if expCost(newSol) < expCost(baseSol) then
13:            baseSol ← newSol
14:            if expCost(newSol) < expCost(bestSol) then
15:                bestSol ← newSol
16:                insert(poolBestSol, bestSol)
17:            end if
18:            k ← K_min
19:        end if
20:    else
21:        temperature ← updateTemperature(detCost(newSol), detCost(baseSol), T)
22:        if (U(0,1) ≤ temperature) then
23:            baseSol ← newSol
24:            k ← K_min
25:        else
26:            k ← min(k ∗ Inc, K_max)
27:        end if
28:    end if
29:    T ← λT
30: end while
31: for sol ∈ poolBestSol do
32:    longSimulation(sol)
33:    if expCost(sol) < expCost(bestSol) then
34:        bestSol ← sol
35:    end if
36: end for
37: return bestSol
```

best solution (*bestSol*) found so far (line 14), the latter is updated and added to the pool of elite solutions (line 16). This pool contains the best stochastic/fuzzy solutions found so far. The number of solutions in this pool is a known parameter that depends on the available computational time. Moreover, by limiting the size of this pool we ensure that we only keep track of the top solutions as the algorithm evolves. In order to further diversify the search, the algorithm might occasionally accept nonimproving solutions following an acceptance criterion (lines 20-28). Specifically, we have used a simulated annealing acceptance criterion, which contains a decaying probability that is regulated by a dynamic temperature parameter (T).

Finally, a refinement procedure using a larger number of simulation runs is executed in the third stage for each elite solution (lines 31–36). Hence, a more accurate summary of output variables can be obtained. As before, both probability distributions and fuzzy functions are employed in this simulation, depending on whether the element has a stochastic or fuzzy nature. Finally, the best-found solution (or pull of best alternative solutions) is returned, considering that the decision maker might be not only interested in the average value associated with a solution but also in its variability level. Particularly, the main output variables in our experiments are: the opening and routing costs, the cost incurred whenever a route fails and the safety stock.

7.2.3 Computational Experiments and Results

Multiple sets of instances are found in the literature to test the algorithms designed to solve the LRP (Akca et al., 2009; Barreto et al., 2007; Belenguer et al., 2011). Nevertheless, these sets do not consider characteristics such as parameters uncertainty and flexible facility sizes, i.e., instances must be adapted to our problem's features. Therefore, we use the instances found in Akca et al. (2009) and introduce the following modifications:

1. Traditional LRP instances consider that a single fixed size is available to assign to open depots. We extend this unit set to five alternative sizes, so that our algorithm selects one of them for each open depot. If s_j is the size proposed by the original instance for each potential depot $j \in J$, and L is the set of available sizes, our approach' alternative sizes are $s_{jl} \in \{(1-2r)s_j, (1-r)s_j, s_j, (1+r)s_j, (1+2r)s_j\}$, where $l \in L$, $0.0 < r < 0.5$, and r is the range of difference between available sizes. When $r = 0$, the case is the same as the traditional LRP. We consider that $r = 0.25$.

2. Traditional LRP instances consider a fixed cost (f_j) incurred whenever a depot $j \in J$ is open. We keep this parameter unaltered. Additionally, we introduce a variable cost (o_{jl}) depending on f_j and s_{jl}, namely: $o_{jl} = \dfrac{(s_{jl} - s_j)}{2s_j} \dfrac{\sum_j f_j}{|J|}$. This formula preserves o_{jl} in the same order as f_j for each depot $j \in J$. Besides, it yields negative costs whenever $s_{jl} < s_j$, positive costs whenever $s_{jl} > s_j$, and a null cost when $s_{jl} = s_j$. Thus our results can be compared with those found in the LRP literature.

3. An uncertain demand D_i for each customer $i \in I$ is considered. The demand of half of the customers is assumed to follow a log-normal probability distribution. If ϕ_i is the deterministic demand in the Akca's set, then $E[D_i] = \phi_i$. In addition, three different values of variance are considered: low, medium and high, i.e., for $\lambda \in \{0.05, 0.10, 0.20\}$, $Var[D_i] = \lambda\phi_i$. These variability values are preserved identical to the ones used in Section 6.3.3, in order to perform a suitable results comparison. The demand of the other half of the customers is considered to be fuzzy. In this case, D_i can be estimated as low (DL), medium (DM) or high (DH). The demand in each of these fuzzy sets is represented by a triangular fuzzy number $D_i = (d_{1i}, d_{2i}, d_{3i})$. If q is the vehicle total load capacity, all fuzzy demand values are expressed as a proportion of q in order to perform an appropriate comparison between the demand and the vehicle available capacity, i.e., $0 \leq D_i \leq 1$. The membership function of these fuzzy sets are displayed in Figure 7.2.

7.2.3.1 A Fuzzy Approach for the Demand and the Vehicle Available Capacity

When considering customers with stochastic demands, the decision about visiting the next customer in a route is made simply by comparing its expected demand with the vehicle's current load. If this demand is greater, the vehicle will perform a detour to the depot for a replenishment. Nevertheless, when the next customer demand is fuzzy, the decision about serving it is made employing a preference index p_i (Teodorović and Pavković, 1996). It indicates the strength of our inclination to visit the next node in a route. This index depends on both the estimated demand of the next node D_{i+1} and the vehicle capacity C_i that remains available after serving the customer $i \in I$. C_i is expressed as a proportion of q, i.e., $0 \leq C_i \leq 1$. It also can be treated as low (CL), medium (CM) or high (CH), and it is represented by a triangular fuzzy number $C_i = (c_{1i}, c_{2i}, c_{3i})$. The membership function of the capacity fuzzy sets are displayed in Figure 7.3.

The preference index is defined between 0 and 1, i.e., $0 \leq p_i \leq 1$. When $p_i = 1$, we will definitely visit the next node in a route since the vehicle available capacity can for sure meet its demand. When $p_i = 0$, we are sure that D_{i+1} exceeds C_i and the vehicle must return to the depot for a replenishment. We consider that the preference can be very low (PVL), low (PL), medium (PM), high (PH) or very high (PVH). Each of these categories is represented by a fuzzy set, whose membership function is depicted in Figure 7.4. Additionally, we define a set of reasoning rules (Table 7.1) to determine the preference to visit the next node depending on the levels of both the demand and the vehicle available capacity. Figure 7.12 displays the procedure used to compute the preference index p_i after serving the customer $i \in I$. This procedure is described as follows:

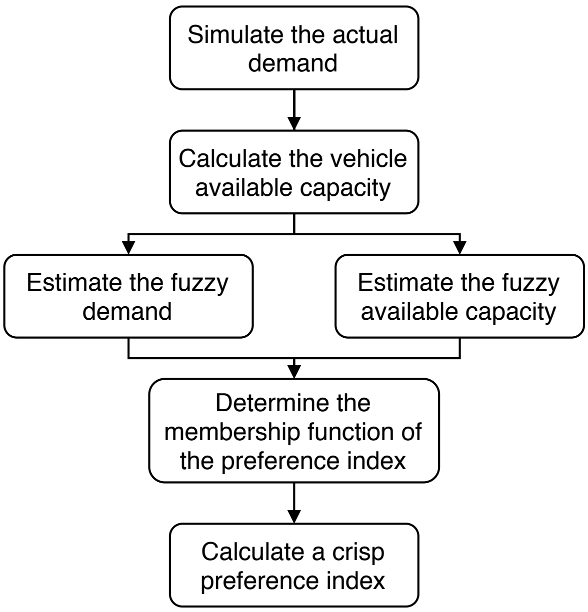


Figure 7.12: Procedure used to compute the preference index p_i.

1. Simulate the actual demand of each customer employing a fuzzy simulation approach. Based on the works by Teodorović and Pavković (1996), Sun et al. (2018) and Sun (2020), we follow the steps described below:

(a) Generate a random demand d_i between a lower bound and an upper bound. Since the objective is preserving the variability conditions similar to the stochastic demands, the lower and upper bounds are given by the expressions $\frac{\phi_i - \sqrt{3\lambda\phi_i}}{q}$ and $\frac{\phi_i + \sqrt{3\lambda\phi_i}}{q}$, respectively.

(b) Calculate the membership degree $\mu(d_i)$ of this demand. Notice that $\mu(d_i) \in [0, 1]$.

(c) Generate a random number $\rho \in [0, 1]$.

(d) Compare ρ and $\mu(d_i)$. If $\rho \leq \mu(d_i)$, then assume the actual demand of the customer i as d_i; otherwise, repeat steps (a)–(d) until this condition is fulfilled.

2. Calculate the vehicle available capacity subtracting from q the sum of the simulated demand of the first m customers visited in the current route, including the customer i. Whenever the route fails and the vehicle must perform a trip to the depot for a replenishment, the counting of m starts again from 1.

3. Estimate the fuzzy demand and the fuzzy available capacity according to the categories previously defined: low, medium or high.

4. Determine the membership function of the preference index using the reasoning rules defined in Table 7.1.

5. Calculate a crisp preference index using the center of gravity as defuzzification method. Additional methods can be found in Klir and Yuan (1995), and Opricovic and Tzeng (2003).

We define a known threshold p^*, such that $0 \leq p^* \leq 1$. The computed preference index p_i must be compared with p^* in order to make a decision about the vehicle next destination. If $p_i \geq p^*$, the vehicle should visit the next customer directly; otherwise, we estimate that the vehicle available capacity cannot meet the next customer demand. In this case, both preventive (c_{prev}) and reactive (c_{reac}) costs are calculated (see Section 7.2.1). If $c_{prev} < c_{reac}$, the vehicle should perform a detour to the depot for a preventive replenishment; otherwise, it should visit the next customer directly and react to its real demand. The lower the threshold level, the greater the inclination to unload the vehicle as much as possible before making a replenishment trip to the depot. In this case, less preventive detours are performed. Hence, the number of times that a reactive round-trip must be carried out increases. Previous tests using modified Akca's instances yielded lower costs when $p^* = 0.45$.

The following parameters are used by our algorithm to run the experiments: *(i)* 350 iterations for map perturbations; *(ii)* 150 iterations for the BR savings heuristic; *(iii)* 150 iterations for splitting; *(iv)* a random value between 0.05 and 0.80 for β_1, the parameter of the geometric distribution associated with the BR selection during the allocation map process; *(v)* a random value between 0.07 and 0.23 for β_2, the parameter of the geometric distribution associated with the BR heuristic for routing; *(vi)* $n = 100$ runs for the initial simulation stage; *(vii)* $N = 5000$ runs for the intensive simulation stage; and *(viii)* 100 iterations to estimate the safety stock (SS), testing only discrete values between 0% and 10%. Our proposed algorithm was coded as a Java application. All experiments were executed on a standard Windows PC with a Core *i*5 processor and 6 GB RAM. A total of ten different random seeds were used for each instance.

7.2.3.2 Results and Discussion

Table 7.6 shows our obtained results for 12 Akca's instances. Five main indicators are computed: depots opening costs (OC), which is formed by both fixed and variable costs; routing costs (RC); failure costs (FC), which is incurred whenever the vehicle must perform either a detour or a round-trip to the depot; total costs (TC); and the estimated safety stock (SS) level. Four types of solutions are compared. All of them are flexible, i.e., they consider facility sizing decisions. Firstly, our best deterministic solutions are shown, i.e, there is no uncertainty in the customers' demand and its realization is exactly as expected. In this case, a safety stock is not necessary and there are no failure costs. Secondly, we show the best stochastic solutions reported in Section 6.3.3, in which the exact customers' demand is not known. Instead, all of them follow a log-normal distribution with known mean and standard deviation. Thirdly, our best hybrid fuzzy-stochastic solutions are displayed, in which half of the customers' demand follows a log-normal distribution, and half of the customers' demand is considered to be fuzzy. Finally, our best fuzzy solutions are shown, in which all customers' demand is considered to be fuzzy, due to a high level of uncertainty. Additionally, results for three levels of variability (λ) are shown. Clearly, our best deterministic solutions are the same regardless of the variability level, given the total absence of uncertainty.

Results in Table 7.6 show a slight average increase in total costs when increasing the variability level for all types of solutions, except for the best deterministic solution. This growth is caused mainly by the rise in failure costs, since a greater number of detours and round-trips is expected when the demand variability level is higher. Additionally, total costs also increase when the uncertainty level is higher regardless of the variability level, i.e., the deterministic solution is the cheapest one, and the fuzzy solution is the most costly. If we compare only the average deterministic cost of each set of solutions, formed by the sum of OC and RC, we obtain values with negligible differences. Hence, the contrasts in total costs are caused mainly by failure costs. For example, for the instance *Cr30x5a-3* in the low variability scenario, 1.6% of total costs are failure costs in the best stochastic solution. However, in the best fuzzy solution this percentage rises to 3.5%. Most instances show this steady growth when increasing the uncertainty level, which confirms that fuzzy scenarios have a higher uncertainty level when compared with deterministic and stochastic scenarios. Finally, the average safety stock increases when both variability and uncertainty levels rise, since more protection against uncertainty is necessary in both cases.

Results corresponding to our best deterministic solution in Table 7.6 were yielded assuming that the realized demand is deterministic. Hence, an additional experiment has been performed, in which this solution (called henceforth OBD) is tested in a hybrid fuzzy-stochastic environment, using 0% of safety stock protection against uncertainty. Figure 7.13 compares this solution's results with our best-found hybrid fuzzy-stochastic solution (OBF)

Table 7.6: Comparative results between our flexible solutions under different uncertainty levels.

Instance	Best Deterministic Solution			Best Stochastic Solution					Best Hybrid Solution					Best Fuzzy Solution				
	OC	**RC**	**TC**	**OC**	**RC**	**FC**	**TC**	**SS**	**OC**	**RC**	**FC**	**TC**	**SS**	**OC**	**RC**	**FC**	**TC**	**SS**
Low variability																		
Cr30x5a-1	200.00	575.14	775.14	200.00	575.14	2.37	777.51	0%	200.00	575.14	3.31	778.45	2%	200.00	575.14	5.86	781.00	2%
Cr30x5a-2	200.00	607.28	807.28	200.00	607.28	0.04	807.32	3%	200.00	607.28	0.12	807.40	3%	200.00	607.28	0.12	807.40	3%
Cr30x5a-3	187.50	507.92	695.42	187.50	509.25	10.99	707.74	3%	187.50	509.25	17.48	714.22	3%	187.50	509.25	25.50	722.25	3%
Cr30x5b-1	225.00	623.22	848.22	225.00	623.22	9.37	857.59	0%	225.00	623.22	14.59	862.81	0%	225.00	623.22	22.85	871.07	1%
Cr30x5b-2	187.50	625.32	812.82	187.50	625.32	0.00	812.82	2%	187.50	625.32	0.00	812.82	2%	187.50	625.32	0.00	812.82	2%
Cr30x5b-3	187.50	684.58	872.08	187.50	684.58	2.25	874.33	1%	187.50	684.58	6.35	878.43	1%	187.50	684.58	9.50	881.58	1%
Cr40x5a-1	162.50	731.84	894.34	162.50	731.84	0.03	894.37	1%	162.50	731.84	0.07	894.41	1%	162.50	731.84	0.59	894.93	1%
Cr40x5a-2	225.00	637.26	862.26	225.00	639.02	0.10	864.12	0%	225.00	639.02	0.81	864.83	1%	225.00	642.02	0.03	867.05	3%
Cr40x5a-3	162.50	752.88	915.38	162.50	752.88	0.97	916.35	0%	162.50	752.88	3.26	918.64	0%	162.50	752.88	6.82	922.21	1%
Cr40x5b-1	162.50	852.04	1014.54	162.50	852.04	6.90	1021.45	1%	162.50	852.04	12.24	1026.78	1%	162.50	852.04	20.79	1035.33	1%
Cr40x5b-2	225.00	690.57	915.57	225.00	690.57	0.08	915.65	1%	225.00	690.57	0.62	916.18	1%	225.00	690.57	1.23	916.79	1%
Cr40x5b-3	175.00	764.33	939.33	175.00	772.87	0.07	947.93	2%	175.00	772.87	0.29	948.16	2%	175.00	772.87	0.35	948.22	2%
Average	191.67	671.03	862.70	191.67	672.00	2.76	866.43	1.17%	191.67	672.00	4.93	868.59	1.42%	191.67	672.25	7.80	871.72	1.75%
Medium variability																		
Cr30x5a-1	200.00	575.14	775.14	200.00	575.14	7.63	782.77	2%	200.00	575.14	9.67	784.81	2%	200.00	575.14	12.91	788.05	2%
Cr30x5a-2	200.00	607.28	807.28	200.00	607.28	0.46	807.74	3%	200.00	607.28	1.94	809.22	3%	200.00	607.28	1.43	808.71	3%
Cr30x5a-3	187.50	507.92	695.42	187.50	509.25	18.50	715.25	3%	187.50	509.25	24.10	720.85	3%	187.50	509.25	29.73	726.48	3%
Cr30x5b-1	225.00	623.22	848.22	225.00	623.22	14.63	862.85	0%	225.00	623.22	18.32	866.53	3%	225.00	623.22	24.23	872.45	3%
Cr30x5b-2	187.50	625.32	812.82	187.50	625.32	0.00	812.82	2%	187.50	625.32	0.00	812.82	2%	187.50	625.32	0.00	812.82	2%
Cr30x5b-3	187.50	684.58	872.08	187.50	684.58	10.21	882.28	0%	187.50	684.58	12.79	884.87	1%	187.50	684.58	12.88	884.96	1%
Cr40x5a-1	162.50	731.84	894.34	162.50	739.24	0.01	901.75	3%	162.50	739.24	0.01	901.75	3%	162.50	739.24	0.00	901.74	3%
Cr40x5a-2	225.00	637.26	862.26	225.00	643.52	3.07	871.59	1%	225.00	642.02	0.24	867.26	3%	225.00	642.02	0.57	867.59	3%
Cr40x5a-3	162.50	752.88	915.38	162.50	752.88	4.46	919.85	1%	162.50	752.88	8.57	923.95	1%	162.50	752.88	11.83	927.22	1%
Cr40x5b-1	162.50	852.04	1014.54	162.50	858.58	4.54	1025.62	2%	162.50	858.58	8.01	1029.09	2%	237.50	795.18	0.00	1032.68	4%
Cr40x5b-2	225.00	690.57	915.57	225.00	690.57	2.06	917.63	1%	225.00	690.57	3.77	919.33	0%	225.00	690.57	5.80	921.37	1%
Cr40x5b-3	175.00	764.33	939.33	175.00	772.87	1.42	949.29	2%	175.00	772.87	2.53	950.40	2%	175.00	772.87	2.96	950.82	2%
Average	191.67	671.03	862.70	191.67	673.54	5.58	870.79	1.67%	191.67	673.41	7.50	872.57	2.08%	197.92	668.13	8.53	874.57	2.33%
High variability																		
Cr30x5a-1	200.00	575.14	775.14	200.00	575.14	19.66	794.80	2%	200.00	575.14	19.82	794.96	0%	200.00	575.14	24.25	799.38	1%
Cr30x5a-2	200.00	607.28	807.28	200.00	607.74	0.02	807.76	5%	200.00	611.41	0.02	811.43	7%	200.00	607.74	0.04	807.78	5%
Cr30x5a-3	187.50	507.92	695.42	187.50	509.25	27.86	724.61	2%	187.50	509.25	29.95	726.70	4%	187.50	509.25	33.41	730.16	3%
Cr30x5b-1	225.00	623.22	848.22	225.00	623.22	19.99	868.21	10%	225.00	623.22	20.73	868.95	10%	225.00	623.22	24.86	873.08	10%
Cr30x5b-2	187.50	625.32	812.82	187.50	625.32	0.10	812.92	3%	187.50	625.32	0.20	813.02	5%	187.50	625.32	0.15	812.97	3%
Cr30x5b-3	187.50	684.58	872.08	187.50	684.58	24.93	897.00	1%	187.50	684.58	29.03	901.11	5%	187.50	684.58	34.01	906.09	5%
Cr40x5a-1	162.50	731.84	894.34	162.50	737.20	2.85	902.55	2%	162.50	735.84	7.83	906.17	1%	162.50	735.84	9.38	907.71	1%
Cr40x5a-2	225.00	637.26	862.26	225.00	642.02	1.79	868.82	3%	225.00	642.02	1.48	868.50	3%	225.00	642.02	2.25	869.27	3%
Cr40x5a-3	162.50	752.88	915.38	162.50	763.69	5.78	931.97	2%	162.50	763.69	7.76	933.96	2%	162.50	752.88	18.65	934.04	1%
Cr40x5b-1	162.50	852.04	1014.54	237.50	786.00	4.65	1028.14	3%	237.50	792.36	2.84	1032.70	4%	237.50	786.00	8.47	1031.97	3%
Cr40x5b-2	225.00	690.57	915.57	225.00	690.57	9.35	924.91	2%	225.00	690.57	12.59	928.15	2%	225.00	690.57	14.96	930.53	2%
Cr40x5b-3	175.00	764.33	939.33	175.00	780.62	4.14	959.76	3%	175.00	780.62	4.90	960.52	3%	175.00	780.62	5.86	961.48	3%
Average	191.67	671.03	862.70	197.92	668.78	10.09	876.79	3.17%	197.92	669.50	11.43	878.85	3.83%	197.92	667.77	14.69	880.37	3.33%

in terms of failure costs. Results for 12 Akca's instances are depicted for each demand variability scenario. Extreme points in dashed lines indicate the average cost for each set of data. As expected, average failure costs show an increasing trend when the variability grows, regardless of the type of solution. Conversely, Figure 7.13 shows that the OBF outperforms the OBD when tested under uncertainty conditions. This fact demonstrates the quality of our fuzzy simheuristic approach, especially in scenarios where the demand variability is high.

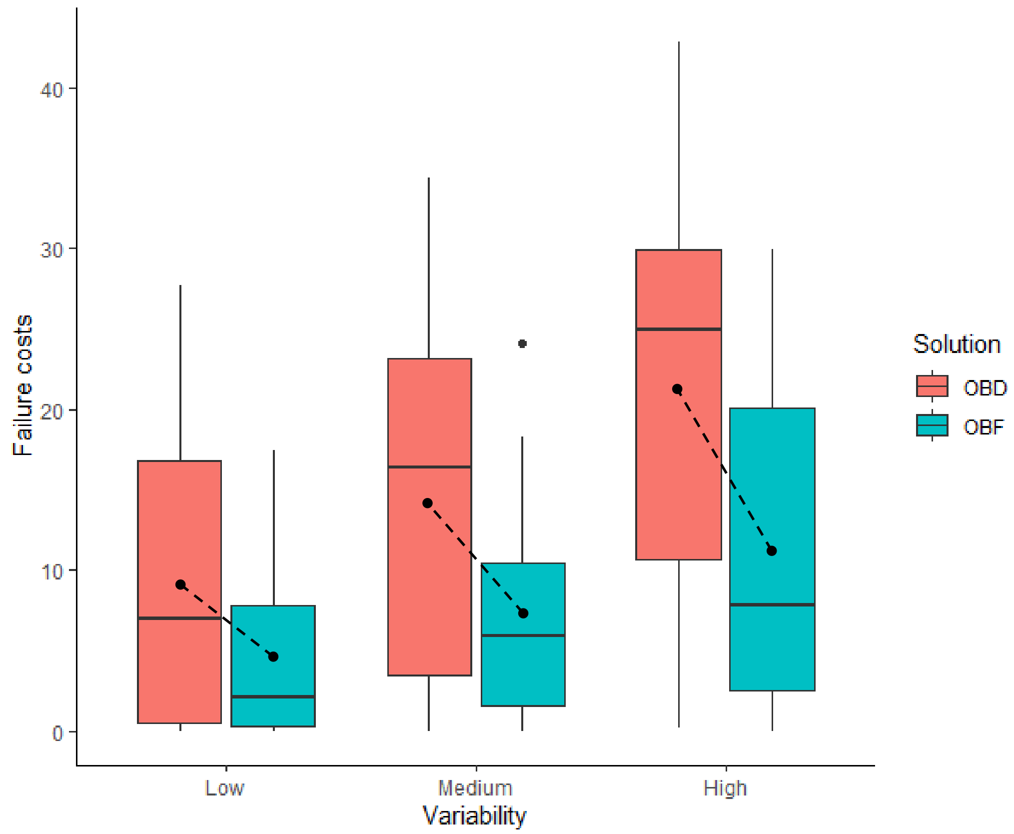


Figure 7.13: Failure costs of our best deterministic and our best hybrid solutions.

Table 7.7 compares two types of hybrid fuzzy-stochastic solutions. Firstly, we show our best solution with a single facility size alternative given by the original Akca's instances, i.e., the solution is not flexible since only one size is available to select. Secondly, we show our best flexible solution, which corresponds to our best hybrid solution in Table 7.6 When comparing the total costs of both types of solutions, the negative gap obtained for all instances and under all variability levels shows the advantages of considering facility sizing decisions. For example, we reach a maximum absolute gap of 7.71% in total cost savings for a single instance. In average, both opening and routing costs decrease whenever alternative depot sizes are available. Nevertheless, each instance shows different results regarding OC and RC. The most evident case is that in which opening costs decrease. Clearly, this is a direct result of having smaller facility size alternatives. Without loss of generality, all examples below take as reference the high variability scenario. For example, the instance *Cr30x5b-3* has a total demand of 1620. Both flexible and non-flexible approaches design the same routes and yield equal routing costs. Nevertheless, the non-flexible approach locates two depots of size 1000 each. Conversely, our flexible approach locates one depot of size 1000 and one depot of size 750. Hence, the non-flexible solution assigns an extra capacity that is not necessary under the problem's current conditions.

Table 7.7: Comparative results between our hybrid solutions when considering facility sizing decisions.

Instance	Best Non-flexible Hybrid Solution					Best Flexible Hybrid Solution					Gap
	OC	RC	FC	TC	SS	OC	RC	FC	TC	SS	TC
Low variability											
Cr30x5a-1	200.00	619.51	3.45	822.96	1%	200.00	575.14	3.31	778.45	2%	-5.41%
Cr30x5a-2	200.00	626.01	0.04	826.05	1%	200.00	607.28	0.12	807.40	3%	-2.26%
Cr30x5a-3	200.00	507.99	17.56	725.55	2%	187.50	509.25	17.48	714.22	3%	-1.56%
Cr30x5b-1	200.00	682.97	0.32	883.29	2%	225.00	623.22	14.59	862.81	0%	-2.32%
Cr30x5b-2	200.00	625.32	0.00	825.32	2%	187.50	625.32	0.00	812.82	2%	-1.51%
Cr30x5b-3	200.00	684.58	5.95	890.53	1%	187.50	684.58	6.35	878.43	1%	-1.36%
Cr40x5a-1	200.00	733.47	3.22	936.70	0%	162.50	731.84	0.07	894.41	1%	-4.51%
Cr40x5a-2	200.00	691.47	11.15	902.63	1%	225.00	639.02	0.81	864.83	1%	-4.19%
Cr40x5a-3	200.00	748.64	9.88	958.52	1%	162.50	752.88	3.26	918.64	0%	-4.16%
Cr40x5b-1	200.00	858.58	1.94	1060.53	2%	162.50	852.04	12.24	1026.78	1%	-3.18%
Cr40x5b-2	300.00	690.57	0.65	991.22	2%	225.00	690.57	0.62	916.18	1%	-7.57%
Cr40x5b-3	200.00	780.62	0.07	980.69	2%	175.00	772.87	0.29	948.16	2%	-3.32%
Average	208.33	687.48	4.52	900.33	1.42%	191.67	672.00	4.93	868.59	1.42%	-3.45%
Medium variability											
Cr30x5a-1	200.00	619.51	9.17	828.68	0%	200.00	575.14	9.67	784.81	2%	-5.29%
Cr30x5a-2	200.00	626.01	0.60	826.61	2%	200.00	607.28	1.94	809.22	3%	-2.10%
Cr30x5a-3	200.00	507.99	24.30	732.29	2%	187.50	509.25	24.10	720.85	3%	-1.56%
Cr30x5b-1	200.00	681.50	14.31	895.80	1%	225.00	623.22	18.32	866.53	3%	-3.27%
Cr30x5b-2	200.00	625.32	0.01	825.33	2%	187.50	625.32	0.00	812.82	2%	-1.52%
Cr30x5b-3	200.00	684.58	15.60	900.18	1%	187.50	684.58	12.79	884.87	1%	-1.70%
Cr40x5a-1	200.00	733.47	7.69	941.17	1%	162.50	739.24	0.01	901.75	3%	-4.19%
Cr40x5a-2	200.00	700.80	12.59	913.39	3%	225.00	642.02	0.24	867.26	3%	-5.05%
Cr40x5a-3	200.00	748.64	20.15	968.79	0%	162.50	752.88	8.57	923.95	1%	-4.63%
Cr40x5b-1	200.00	863.91	2.32	1066.23	3%	162.50	858.58	8.01	1029.09	2%	-3.48%
Cr40x5b-2	300.00	690.57	4.18	994.75	1%	225.00	690.57	3.77	919.33	0%	-7.58%
Cr40x5b-3	200.00	780.62	0.94	981.56	3%	175.00	772.87	2.53	950.40	2%	-3.17%
Average	208.33	688.58	9.32	906.23	1.58%	191.67	673.41	7.50	872.57	2.08%	-3.63%
High variability											
Cr30x5a-1	200.00	619.51	20.69	840.20	0%	200.00	575.14	19.82	794.96	0%	-5.38%
Cr30x5a-2	200.00	621.45	5.66	827.12	3%	200.00	611.41	0.02	811.43	7%	-1.90%
Cr30x5a-3	200.00	507.99	30.16	738.15	4%	187.50	509.25	29.95	726.70	4%	-1.55%
Cr30x5b-1	200.00	681.50	18.85	900.35	0%	225.00	623.22	20.73	868.95	10%	-3.49%
Cr30x5b-2	200.00	625.32	0.14	825.46	5%	187.50	625.32	0.20	813.02	5%	-1.51%
Cr30x5b-3	200.00	684.58	30.23	914.81	1%	187.50	684.58	29.03	901.11	5%	-1.50%
Cr40x5a-1	200.00	737.94	5.78	943.73	2%	162.50	735.84	7.83	906.17	1%	-3.98%
Cr40x5a-2	200.00	700.80	15.98	916.78	3%	225.00	642.02	1.48	868.50	3%	-5.27%
Cr40x5a-3	200.00	748.64	32.89	981.54	0%	162.50	763.69	7.76	933.96	2%	-4.85%
Cr40x5b-1	200.00	858.58	22.53	1081.11	2%	237.50	792.36	2.84	1032.70	4%	-4.48%
Cr40x5b-2	300.00	693.03	12.66	1005.69	0%	225.00	690.57	12.59	928.15	2%	-7.71%
Cr40x5b-3	200.00	772.87	13.22	986.09	2%	175.00	780.62	4.90	960.52	3%	-2.59%
Average	208.33	687.68	17.40	913.42	1.83%	197.92	669.50	11.43	878.85	3.83%	-3.68%

Some instances show an opposite behavior, i.e., opening costs either increase or remain the same while routing costs decrease. For example, the non-flexible solution of the instance *Cr30x5a-1* opens two depots of size 1000 each. Alternatively, the flexible solution opens one depot of size 1500 and one depot of size 500, i.e., the total capacity is equal and, given our defined costs structure, also the opening costs. However, this slight change drives a redesign of routes that decreases RC. An additional example is given by the instance *Cr40x5a-2*. Figure 7.14 depicts the best solution found by both the non-flexible approach (a) and our flexible approach (b). The solution in Figure 7.14a locates two depots of size 1750 each, and the solution in Figure 7.14b locates three depots of size 875 each. The latter case has a total capacity that is smaller than the former's; however, opening costs are higher since the fixed cost is clearly greater when 3 facilities are open instead of 2. This new configuration decreases considerably routing costs (Table 7.7), which shows that considering facility sizing decisions not only reduces total costs by decreasing depots capacity but also by increasing it, since shorter routes can be designed.

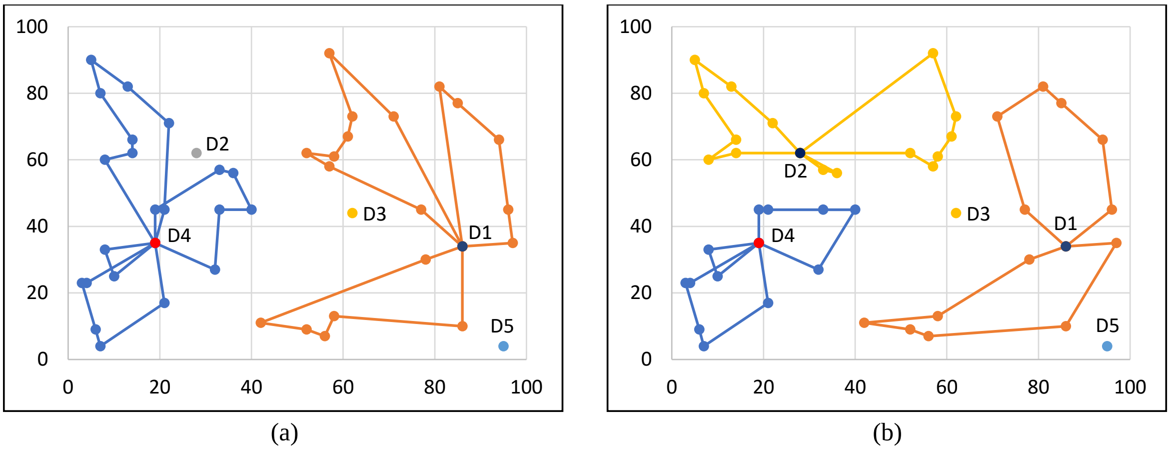


Figure 7.14: Best-found solution by a non-flexible (**a**) and a flexible (**b**) fuzzy LRP for the instance *Cr40x5a-2*.

7.3 Conclusions

This chapter has proposed a new methodology to address combinatorial T&L problems where demands of a subset of customers are stochastic, while demands of the complementary subset are fuzzy. Specifically, a VRP and an LRP with facility sizing decisions are solved employing this approach. Additionally, a TOP with fuzzy travel times is addressed as well. Particularly, in the case of the VRP and the TOP, we consider that both stochastic and fuzzy uncertainty are present in many real-life transportation systems. Hence, pure deterministic, pure stochastic, and pure fuzzy scenarios represent particular cases that can also be addressed by employing our fuzzy simheuristic methodology. Since our methodology combines metaheuristics with stochastic and fuzzy simulation, it takes the best characteristics of both worlds, i.e.: *(i)* the metaheuristics component provides the efficiency necessary to explore the solution space in order to find near-optimal solutions in short computational times. This characteristic becomes highly relevant when dealing with transportation problems, which are usually NP-hard; and *(ii)* the stochastic/fuzzy simulation component provides suitable tools to cope with different types of uncertainty, in order to provide high-quality solutions in terms of expected costs, expected profits, or risk/reliability indicators. A set of numerical instances demonstrates these advantages. The simultaneous consideration of stochastic and fuzzy uncertainty arises whenever a subset of elements in a transportation problem, e.g., customers, roads, or vehicles, allows us to model some uncertainty aspects using probability distributions, while others require fuzzy techniques due to their vagueness or to the lack of enough historical data. The well-known VRP and TOP have been useful in testing our approach. For the VRP, we have studied a numerical example in which demands associated with a group of customers are stochastic, while a different group of customers presents fuzzy demands. Regarding the TOP, we have analyzed a case study in which travel times between customers are stochastic for a group of edges, and fuzzy for another group. The obtained results show that employing our approach leads to improve the solution quality –in terms of total cost for the VRP, and total collected reward for the TOP– when uncertainty is considered. All in all, these numerical examples illustrate the efficiency of the proposed methodology to solve transportation problems combining, at the same time, deterministic, stochastic, and fuzzy elements, something that has been rarely explored in the existing literature despite its relevance in real-life applications.

Regarding the LRPFS, a fuzzy simheuristic approach is proposed to solve this problem cost- and time- efficiently. Initially, our algorithm selects the best size for each open facility from a set of provided alternatives. We perform an iterative procedure in which a set of location-allocation-routing configurations are assessed in terms of opening and routing costs. Then a top list of complete LRP solutions is iteratively perturbed and simulated. The perturbation stage is performed by employing an ILS metaheuristic. The simulation stage is carried out by running a classic MCS for the stochastic demands and a fuzzy simulation for the fuzzy demands. Failure costs are introduced as an additional performance indicator. Finally, a set of elite solutions is assessed through a refinement procedure where a larger number of simulation runs is executed. Our fuzzy simheuristic approach has been proved to be flexible enough not only to combine efficiently stochastic and fuzzy demands in a single execution but also to address less general scenarios in which demands of all customers are either deterministic or fuzzy. Our approach has also been proved to be a cost-efficient algorithm when considering uncertainty scenarios. It decreases route failure costs when compared with the best deterministic solution tested in a hybrid fuzzy-stochastic environment. The use of a safety stock policy as a protection against uncertainty has also contributed to this decrease. To the best of our knowledge, this is the first time that a hybrid fuzzy-stochastic LRP with facility sizing decisions is addressed. Medium-sized benchmark instances considering three demand variability levels were used. Obtained results show that introducing such flexibility decreases total costs in two mutually nonexclusive ways: firstly, yielding savings in opening costs by locating facilities of smaller size; and secondly, yielding savings in routing costs by locating facilities of higher size, which drives a routes redesign that reduces the total traveled distance. We also have demonstrated that these savings are always incurred regardless of the demand variability level.

Chapter 8

General Conclusions and Future Research Lines

8.1 General Conclusions

This thesis has presented a series of applications of different solution methods to multiple strategic, tactical, and operational transportation and logistics (T&L) problems. Most proposed methods are approximate, i.e., they do not guarantee that an optimal solution can be found, but they have been proved to be fast and obtain high-quality results. Comparisons between the results attained by the presented methods and those obtained by exact models –i.e., some optimal solutions are calculated by mixed-integer linear programming (MILP) models proposed in this thesis–, or those reported in the literature, demonstrate the cost- and time-efficiency of our algorithms. Applications considering deterministic, stochastic, or fuzzy parameters have been studied. The proposed solution algorithms have been adapted to each of these cases. Furthermore, we design a "fuzzy simheuristics" procedure, which combines heuristics and metaheuristics with stochastic and fuzzy simulation. Fuzzy simheuristics is a generic approach since it includes as particular cases pure deterministic, pure stochastic, and pure fuzzy problems. A vehicle routing problem with stochastic and fuzzy demands (VRP-S/F-D), a team orienteering problem with stochastic and fuzzy travel times (TOP-S/F-TT), and a location routing problem with facility-sizing decisions and stochastic and fuzzy demands (LRPFS-S/F-D) have been employed to test the fuzzy simheuristics approach. Results obtained in the three applications show that considering fuzzy simheuristic algorithms is a solid approach to solve T&L combinatorial optimization problems (COP) with a high degree of uncertainty.

Fuzzy simheuristics are composed of different layers that make them a cost-efficient approach. Each of these layers have been tested independently in other T&L problems, whose results can be seen along this document. Biased-randomized (BR) heuristics represent the "most basic" layer. Two real-world rich applications (Chapter 3) demonstrate the efficiency and flexibility of this type of algorithms. A large quantity of real constraints and conditions has been considered when solving these challenges. The applicability of BR algorithms has been demonstrated in both the transportation of protective elements in the COVID-19 lockdown period in Barcelona and a feed distribution project in the agri-food sector in Catalonia. A metaheuristic layer is then introduced to enrich the BR algorithms (Chapter 4).

Dynamic conditions, optional backhauls, and facility-sizing decisions are addressed in a dynamic ride-sharing problem (DRSP), a VRPOB, and a location routing problem (LRPFS), respectively. Results in the DRSP show that cost savings are obtained when considering a dynamic approach –in this case, a discrete-event driven metaheuristic– instead of a static approach when external conditions, such as traffic or weather, change over time. Results in the VRPOB show that including penalty costs for not serving backhaul customers leads to cost savings. Finally, results in the LRPFS demonstrate that including the facility capacity as an additional variable to model decreases the costs of the T&L activities. This is showed as well in Chapters 6 and 7, since this thesis takes the LRPFS as a core problem to demonstrate the advantages of employing metaheuristic, simheuristic, and fuzzy simheuristic approaches when deterministic, stochastic, and fuzzy/stochastic parameters, respectively, are considered. Furthermore, since the LRP includes the VRP and the FLP as particular cases, the solution approaches proposed in this thesis can also be applied to these problems.

Chapters 3 and 4 consider all inputs as deterministic. Hence, Chapter 5 serves as an introduction to consider stochastic parameters and, therefore, to the inclusion of hybrid simulation-optimization approaches to deal with this type of problems. Particularly, Monte Carlo simulation (MCS) and a MILP model are hybridized to solve an FLP with stochastic demands. This chapter also introduces the reliability as an additional indicator to assess solutions that include stochasticity. Hence, the simultaneous consideration of costs and reliability has been proved to be a good approach when studying T&L problems including stochastic inputs. Moreover, this chapter shows the effect of considering different probability distributions when modeling these parameters. Later, Chapter 6 applies simheuristic algorithms to solve an open VRP and a TOP with stochastic service and travel times, and an LRPFS with stochastic demands. The COVID-19 case instances considered in Chapter 3 are adapted to assess the proposed simheuristics in the OVRPSSTT and the TOPSSTT. In general, Chapter 6's results show that both cost savings and good reliability levels are obtained when employing simheuristics to solve T&L COPs, in comparison to the case where deterministic inputs are considered in a stochastic environment. Finally, Chapter 7 proposes the use of fuzzy simheuristics to deal with problems including fuzzy and stochastic parameters, as mentioned above. This type of algorithms represents the final layer of the approximate solution methods proposed in this thesis and, therefore, it includes BR heuristics, metaheuristics, and simheuristics in a fuzzy environment.

8.2 Future Research Lines

The results obtained after developing and testing the multiple solution methods proposed in this thesis, and their application to different T&L problems, allow to identify the following research lines, which can be explored as future work:

- Customer demands and service and travel times have been the only parameters considered as fuzzy or stochastic in the proposed methods. Additional operational parameters that can be uncertain in real-world T&L problems are facility capacities, selling prices, supply quantities, among others. Furthermore, disruption risks can also be considered as an uncertainty source. They refer to low-frequency high-impact events, such as earthquakes, floods, pandemics, or terrorist attacks, which can affect the structure of the supply chain. For instance, a group of facilities can be suddenly destroyed, or some roads cannot be longer used for transportation activities. Considering these events is a great opportunity to test the performance of both simheuristics and fuzzy simheuristics in different contexts. Humanitarian logistics is a real-world activity that can benefit from these approaches.

- Considering resilience –which is the ability of a supply chain to recover successfully after being disturbed– in designing and managing T&L activities is a successful approach to face uncertainty, operational and disruption risks. This concept implies considering key performance indicators additional to the traditional costs and profits. Multi-objective simheuristics and fuzzy simheuristics can be a successful approach to deal with these cases. Furthermore, once a multi-objective approach has been considered, sustainability issues can be easily included as well, given the concerns about the environmental and social impacts of human activities, additional to the economic goal.
- Multi-period problems can be considered to increase the quality of the obtained results. This is useful in any of the problems addressed in this thesis, but especially in those ones that consider a reward or a penalization for serving or not serving, respectively, a set of customers. The VRPOB and the TOP are examples of problems with these characteristics. For instance, if a customer in the VRPOB is not visited in a given day, the penalization for not serving it the next day should be higher. Inventory decisions can be included as well in multi-period problems.
- Section 3.2 introduced a reactive (automatic) fine-tuning process for the main parameter of the BR method, i.e., a manual time-consuming fine-tuning process is not necessary. This parameter-less procedure can be included in metaheuristic, simheuristic, and fuzzy simheuristic approaches to test its performance in terms of cost- and time-efficiency when more complex algorithms are employed.
- Further heuristic and metaheuristic approaches can be tested when solving the problems addressed in this thesis. For instance, BR versions of the iterated local search (ILS) algorithm can be employed to solve those problems where only multi-start approaches have been tested, such as the RVRP or the RTOP. Furthermore, the combination of ILS with MCS, which has been proved to be successful when solving the LRPFSSD, can be used to solve stochastic and fuzzy versions of the VRP and the TOP.

8.3 Research Outcomes

All results presented in this thesis have been published or have been accepted for publication in scientific articles or conference papers in peer-reviewed JCR- or Scopus-indexed journals. The complete list of articles is enumerated below. The covers of all articles are shown in Appendix B.2.

8.3.1 JCR/Scopus-Indexed Articles

1. **Tordecilla, R.D.**, Juan, A.A., Montoya-Torres, J., Quintero-Araujo, C., & Panadero, J. (2021). Simulation-optimization methods for designing and assessing resilient supply chain networks under uncertainty scenarios: A review. *Simulation Modelling Practice and Theory*, 106, 102166 (indexed in ISI SCI, 2020 IF = 3.272, Q1; 2020 SJR = 0.554, Q2). ISSN: 1569-190X.

2. **Tordecilla, R.D.**, Martins, L.C., Panadero, J., Copado, P.J., Perez-Bernabeu, E., & Juan, A.A. (2021). Fuzzy simheuristics for optimizing transportation systems: dealing with stochastic and fuzzy uncertainty. *Applied Sciences*, 11(17), 7950 (indexed in ISI SCI, 2020 IF = 2.679, Q2; 2020 SJR = 0.435, Q2). ISSN: 2076-3417.

3. **Tordecilla, R.D.**, Montoya-Torres, J.R., Quintero-Araujo, C.L., Panadero, J., & Juan, A.A. (2022). The location routing problem with facility sizing decisions. *International Transactions in Operational Research*, 0, 1-31 (indexed in ISI SCI, 2020 IF = 4.193, Q2; 2020 SJR = 1.032, Q1). ISSN: 0969-6016.

4. Londoño, J.C., **Tordecilla, R.D.**, Martins, L.C., & Juan, A.A. (2020). A biased-randomized iterated local search for the vehicle routing problem with optional backhauls. *TOP*, 29, 387-416 (indexed in ISI SCI, 2018 IF = 0.892, Q3; 2013 SJR = 0.600, Q2). ISSN: 1134-5764.

5. Martins, L. C., **Tordecilla, R.D.**, Castañeda, J., Juan, A.A. & Faulin, J. (2021). Electric vehicle routing, arc routing, and team orienteering problems in sustainable transportation. *Energies*, 14(16), 5131 (indexed in ISI SCI, 2019 IF = 2.702, Q3; 2019 SJR = 0.635, Q2). ISSN: 1996-1073.

6. Raba, D., **Tordecilla, R.D.**, Copado, P., Juan, A., & Mount, D. (2021). A digital twin for decision making on livestock feeding. *INFORMS Journal on Applied Analytics (Interfaces)*, 0, 1-16 (indexed in ISI SCI, 2020 IF = 1.434, Q4; 2020 SJR = 0.662, Q2). ISSN: 0092-2102.

7. Peyman, M., Copado, P.J., **Tordecilla, R.D.**, Martins, L.C., Xhafa, F., & Juan, A.A. (2021). Edge computing and IoT analytics for agile optimization in intelligent transportation systems. *Energies*, 14(19), 6309 (indexed in ISI SCI, 2020 IF = 3.004, Q3; 2020 SJR = 0.598, Q2). ISSN: 1996-1073.

8.3.2 Scopus-Indexed Articles

1. **Tordecilla, R.D.**, Martins, L.C., Saiz, M., Copado-Mendez, P.J., Panadero, J., & Juan, A.A. (2021). Agile computational intelligence for supporting hospital logistics during the COVID-19 crisis. In: *Computational Management*. Springer, pp. 383–407 (indexed in Scopus, 2019 SJR = 0.204, Q3). ISSN: 2196-7326.

2. **Tordecilla, R.D.**, Copado-Mendez, P.J., Panadero, J., Quintero-Araujo, C.L., Montoya-Torres, J.R., & Juan, A.A. (2021). Combining heuristics with simulation and fuzzy logic

to solve a flexible-size location routing problem under uncertainty. *Algorithms*, 14(2), 45 (indexed in ESCI; 2019 SJR = 0.358, Q3). ISSN: 1999-4893.

3. **Tordecilla, R.D.**, Panadero, J., Quintero-Araújo, C.L., Montoya-Torres, J.R., & Juan, A.A. (2020). A simheuristic algorithm for the location routing problem with facility sizing decisions and stochastic demands. *2020 Winter Simulation Conference (WSC)*, pp. 1265-1275 (indexed in ISI WOS and Scopus, 2019 SJR = 0.410). ISSN: 0891-7736.

4. **Tordecilla, R.D.**, Copado, P., Panadero, J., Martins, L., & Juan, A.A. (2021). An agile and reactive biased-randomized heuristic for an agri-food rich vehicle routing problem. *Transportation Research Procedia*, 58, 385-392 (indexed in Scopus, 2018 SJR = 0.601). ISSN: 2352-1465.

5. Rabe, M., **Tordecilla, R.D.**, Martins, L.C., Chicaiza-Vaca, J., & Juan, A.A. (2021). Supporting hospital logistics during the first months of the COVID-19 crisis: a simheuristic for the stochastic team orienteering problem. *2021 Winter Simulation Conference (WSC)*. Accepted conference article (indexed in ISI WOS and Scopus, 2019 SJR = 0.410). ISSN: 0891-7736.

6. Peyman, M., Li, Y., **Tordecilla, R.D.**, Copado, P., Juan, A.A., & Xhafa, F. (2021). Waste collection of medical items under uncertainty using Internet of things and city open data repositories: a simheuristic approach. *2021 Winter Simulation Conference (WSC)*. Accepted conference article (indexed in ISI WOS and Scopus, 2019 SJR = 0.410). ISSN: 0891-7736.

7. Juan, A.A., Corlu, C.G., **Tordecilla, R.D.**, de la Torre, R., & Ferrer, A. (2020). On the use of biased-randomized algorithms for solving non-smooth optimization problems. *Algorithms*, 13(1), 8 (indexed in ESCI; 2019 SJR = 0.358, Q3). ISSN: 1999-4893.

8. Rabe, M., Chicaiza-Vaca, J., **Tordecilla, R.D.**, & Juan, A.A. (2020). A simulation-optimization approach for locating automated parcel lockers in urban logistics operations. *2020 Winter Simulation Conference (WSC)*, pp. 1230-1241 (indexed in ISI WOS and Scopus, 2019 SJR = 0.410). ISSN: 0891-7736.

9. Rabe, M., Gonzalez-Feliu, J., Chicaiza-Vaca, J., & **Tordecilla, R.D.** (2021). Simulation-optimization approach for multi-period facility location problems with forecasted and random demands in a last-mile logistics application. *Algorithms*, 14(2), 41 (indexed in ISI WOS and Scopus, 2019 SJR = 0.410). ISSN: 0891-7736.

Bibliography

Aazam, Mohammad and Eui-Nam Huh (2014). "Fog computing and smart gateway based communication for cloud of things". In: *2014 International Conference on Future Internet of Things and Cloud*. IEEE, pp. 464–470.

— (2015). "Fog computing micro datacenter based dynamic resource estimation and pricing model for IoT". In: *2015 IEEE 29th International Conference on Advanced Information Networking and Applications*. IEEE, pp. 687–694.

Adenso-Diaz, Belarmino, Carlos Mena, Santiago García-Carbajal, and Merrill Liechty (2012). "The impact of supply network characteristics on reliability". In: *Supply Chain Management: An International Journal* 17.3, pp. 263–276.

Agatz, Niels A.H., Alan L. Erera, Martin W.P. Savelsbergh, and Xing Wang (2011). "Dynamic ride-sharing: A simulation study in metro Atlanta". In: *Transportation Research Part B: Methodological* 45.9, pp. 1450–1464.

Ahlgren, Bengt, Markus Hidell, and Edith C-H Ngai (2016). "Internet of things for smart cities: Interoperability and open data". In: *IEEE Internet Computing* 20.6, pp. 52–56.

Ahmadi-Javid, Amir, Pardis Seyedi, and Siddhartha S Syam (2017). "A survey of healthcare facility location". In: *Computers & Operations Research* 79, pp. 223–263.

Akca, Z, RT Berger, and TK Ralphs (2009). "A branch-and-price algorithm for combined location and routing problems under capacity restrictions". In: *Operations Research and Cyber-Infrastructure*. Springer, pp. 309–330.

Al Chami, Zaher, Hamza El Flity, Hervé Manier, and Marie-Ange Manier (2018). "A new metaheuristic to solve a selective pickup and delivery problem". In: *2018 4th International Conference on Logistics Operations Management (GOL)*. IEEE, pp. 1–5.

Albareda-Sambola, Maria and Jessica Rodríguez-Pereira (2019). "Location-routing and location-arc routing". In: *Location Science*. Springer, pp. 431–451.

Alemany, Gabriel, Jesica de Armas, Angel A Juan, Álvaro García-Sánchez, Roberto García-Meizoso, and Miguel Ortega-Mier (2016). "Combining Monte Carlo simulation with heuristics to solve a rich and real-life multi-depot vehicle routing problem". In: *2016 Winter Simulation Conference (WSC)*. IEEE, pp. 2466–2474.

Alonso-Mora, Javier, Samitha Samaranayake, Alex Wallar, Emilio Frazzoli, and Daniela Rus (2017). "On-demand high-capacity ride-sharing via dynamic trip-vehicle assignment". In: *Proceedings of the National Academy of Sciences* 114.3, pp. 462–467.

Altiparmak, Fulya, Mitsuo Gen, Lin Lin, and Turan Paksoy (2006). "A genetic algorithm approach for multi-objective optimization of supply chain networks". In: *Computers & Industrial Engineering* 51.1, pp. 196–215.

Alumur, Sibel and Bahar Y Kara (2007). "A new model for the hazardous waste location-routing problem". In: *Computers & Operations Research* 34.5, pp. 1406–1423.

— (2008). "Network hub location problems: The state of the art". In: *European journal of operational research* 190.1, pp. 1–21.

Alvarez Fernandez, Stephanie, Daniele Ferone, Angel Juan, and Daniele Tarchi (2021). "A simheuristic algorithm for video streaming flows optimisation with QoS threshold modelled as a stochastic single-allocation p-hub median problem". In: *Journal of Simulation*, pp. 1–14.

Amaran, Satyajith, Nikolaos V Sahinidis, Bikram Sharda, and Scott J Bury (2014). "Simulation optimization: a review of algorithms and applications". In: *4OR* 12.4, pp. 301–333.

Amin, Saman Hassanzadeh and Fazle Baki (2017). "A facility location model for global closed-loop supply chain network design". In: *Applied Mathematical Modelling* 41, pp. 316–330.

Amin, Saman Hassanzadeh and Guoqing Zhang (2013). "A multi-objective facility location model for closed-loop supply chain network under uncertain demand and return". In: *Applied Mathematical Modelling* 37.6, pp. 4165–4176.

Andrade, Carlos E, Thuener Silva, and Luciana S Pessoa (2019). "Minimizing flowtime in a flowshop scheduling problem with a biased random-key genetic algorithm". In: *Expert Systems with Applications* 128, pp. 67–80.

Assis, Luciana P, André L Maravilha, Alessandro Vivas, Felipe Campelo, and Jaime A Ramírez (2013). "Multiobjective vehicle routing problem with fixed delivery and optional collections". In: *Optimization letters* 7.7, pp. 1419–1431.

Augerat, Philippe, D Naddef, JM Belenguer, E Benavent, A Corberan, and Giovanni Rinaldi (1995). *Computational results with a branch and cut code for the capacitated vehicle routing problem*. Tech. rep. Universite Joseph Fourier, Grenoble, France.

Aykin, Turgut (1995). "The hub location and routing problem". In: *European Journal of Operational Research* 83.1, pp. 200–219.

Azadeh, A and H Farrokhi-Asl (2019). "The close–open mixed multi depot vehicle routing problem considering internal and external fleet of vehicles". In: *Transportation Letters* 11.2, pp. 78–92.

Bagirov, Adil M and John Yearwood (2006). "A new nonsmooth optimization algorithm for minimum sum-of-squares clustering problems". In: *European journal of operational research* 170.2, pp. 578–596.

Bagloee, Saeed Asadi, Madjid Tavana, Mohsen Asadi, and Tracey Oliver (2016). "Autonomous vehicles: challenges, opportunities, and future implications for transportation policies". In: *Journal of modern transportation* 24.4, pp. 284–303.

Balcik, Burcu and Benita M Beamon (2008). "Facility location in humanitarian relief". In: *International Journal of logistics* 11.2, pp. 101–121.

Baldacci, Roberto, Aristide Mingozzi, and Roberto Wolfler Calvo (2011). "An exact method for the capacitated location-routing problem". In: *Operations Research* 59.5, pp. 1284–1296.

Barreto, Sérgio, Carlos Ferreira, Jose Paixao, and Beatriz Sousa Santos (2007). "Using clustering analysis in a capacitated location-routing problem". In: *European Journal of Operational Research* 179.3, pp. 968–977.

Bayliss, Christopher, Roberto Guidotti, Alejandro Estrada-Moreno, Guillermo Franco, and Angel A Juan (2020a). "A biased-randomized algorithm for optimizing efficiency in parametric earthquake (Re) insurance solutions". In: *Computers & Operations Research* 123, p. 105033.

Bayliss, Christopher, Angel A Juan, Christine SM Currie, and Javier Panadero (2020b). "A learnheuristic approach for the team orienteering problem with aerial drone motion constraints". In: *Applied Soft Computing* 92, p. 106280.

Bayliss, Christopher, Leandro do C Martins, and Angel A Juan (2020c). "A Two-phase Local Search with a Discrete-event Heuristic for the Omnichannel Vehicle Routing Problem". In: *Computers & Industrial Engineering*, p. 106695.

Behzadi, Golnar, Michael Justin O'Sullivan, Tava Lennon Olsen, Frank Scrimgeour, and Abraham Zhang (2017). "Robust and resilient strategies for managing supply disruptions in an agribusiness supply chain". In: *International Journal of Production Economics* 191, pp. 207–220.

Belenguer, José-Manuel, Enrique Benavent, Christian Prins, Caroline Prodhon, and Roberto Wolfler Calvo (2011). "A branch-and-cut method for the capacitated location-routing problem". In: *Computers & Operations Research* 38.6, pp. 931–941.

Belgin, Onder, Ismail Karaoglan, and Fulya Altiparmak (2018). "Two-echelon vehicle routing problem with simultaneous pickup and delivery: Mathematical model and heuristic approach". In: *Computers & Industrial Engineering* 115, pp. 1–16.

Bellmore, Mandell and George L Nemhauser (1968). "The traveling salesman problem: a survey". In: *Operations Research* 16.3, pp. 538–558.

Belloso, Javier, Angel A Juan, and Javier Faulin (2019). "An iterative biased-randomized heuristic for the fleet size and mix vehicle-routing problem with backhauls". In: *International Transactions in Operational Research* 26.1, pp. 289–301.

Belloso, Javier, Angel A Juan, Enoc Martinez, and Javier Faulin (2017). "A biased-randomized metaheuristic for the vehicle routing problem with clustered and mixed backhauls". In: *Networks* 69.3, pp. 241–255.

Beneicke, Julia, Angel A Juan, Fatos Xhafa, David Lopez-Lopez, and Alfons Freixes (2019). "Empowering citizens' cognition and decision making in smart sustainable cities". In: *IEEE Consumer Electronics Magazine* 9.1, pp. 102–108.

Bent, Russell W and Pascal Van Hentenryck (2004). "Scenario-based planning for partially dynamic vehicle routing with stochastic customers". In: *Operations Research* 52.6, pp. 977–987.

Berbeglia, Gerardo, Jean-François Cordeau, Irina Gribkovskaia, and Gilbert Laporte (2007). "Static pickup and delivery problems: a classification scheme and survey". In: *Top* 15.1, pp. 1–31.

Bidhandi, Hadi Mohammadi, Rosnah Mohd Yusuff, Megat Mohamad Hamdan Megat Ahmad, and Mohd Rizam Abu Bakar (2009). "Development of a new approach for deterministic supply chain network design". In: *European Journal of Operational Research* 198.1, pp. 121–128.

Birge, John R and Francois Louveaux (2011). *Introduction to stochastic programming*. Springer Science & Business Media.

Bistaffa, Filippo, Christian Blum, Jesúus Cerquides, Alessandro Farinelli, and Juan A Rodríguez-Aguilar (2019). "A Computational Approach to Quantify the Benefits of Ridesharing for Policy Makers and Travellers". In: *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–12.

Blackhurst, J, T Wu, and P O'grady (2004). "Network-based approach to modelling uncertainty in a supply chain". In: *International Journal of Production Research* 42.8, pp. 1639–1658.

Booker, JM and TJ Ross (2011). "An evolution of uncertainty assessment and quantification". In: *Scientia Iranica* 18.3, pp. 669–676.

Boonmee, Chawis, Mikiharu Arimura, and Takumi Asada (2017). "Facility location optimization model for emergency humanitarian logistics". In: *International Journal of Disaster Risk Reduction* 24, pp. 485–498.

Braekers, Kris, Katrien Ramaekers, and Inneke Van Nieuwenhuyse (2016). "The vehicle routing problem: State of the art classification and review". In: *Computers & Industrial Engineering* 99, pp. 300–313.

Brandão, José (2016). "A deterministic iterated local search algorithm for the vehicle routing problem with backhauls". In: *Top* 24.2, pp. 445–465.

— (2020). "A memory-based iterated local search algorithm for the multi-depot open vehicle routing problem". In: *European Journal of Operational Research* 284.2, pp. 559–571.

Brandão, Julliany S, Thiago F Noronha, Mauricio GC Resende, and Celso C Ribeiro (2015). "A biased random-key genetic algorithm for single-round divisible load scheduling". In: *International Transactions in Operational Research* 22.5, pp. 823–839.

— (2017). "A biased random-key genetic algorithm for scheduling heterogeneous multi-round systems". In: *International Transactions in Operational Research* 24.5, pp. 1061–1077.

Brito, Julio, Airam Expósito, and José A Moreno (2016). "Solving the team orienteering problem with fuzzy scores and constraints". In: *2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. IEEE, pp. 1614–1620.

Bruck, Bruno P and Manuel Iori (2017). "Non-elementary formulations for single vehicle routing problems with pickups and deliveries". In: *Operations Research* 65.6, pp. 1597–1614.

Bruneau, Michel, Stephanie E Chang, Ronald T Eguchi, George C Lee, Thomas D O'Rourke, Andrei M Reinhorn, Masanobu Shinozuka, Kathleen Tierney, William A Wallace, and Detlof Von Winterfeldt (2003). "A framework to quantitatively assess and enhance the seismic resilience of communities". In: *Earthquake spectra* 19.4, pp. 733–752.

Caceres-Cruz, Jose, Pol Arias, Daniel Guimarans, Daniel Riera, and Angel A Juan (2014). "Rich vehicle routing problem: Survey". In: *ACM Computing Surveys (CSUR)* 47.2, pp. 1–28.

Caldeira, Rylan H and A Gnanavelbabu (2021). "A simheuristic approach for the flexible job shop scheduling problem with stochastic processing times". In: *Simulation* 97.3, pp. 215–236.

Calvet, Laura, Angel A Juan, Carles Serrat, and Jana Ries (2016). "A statistical learning based approach for parameter fine-tuning of metaheuristics". In: *SORT-Statistics and Operations Research Transactions* 1.1, pp. 201–224.

Calvet, Laura, Dandan Wang, Angel A Juan, and Lluc Bové (2019). "Solving the multidepot vehicle routing problem with limited depot capacity and stochastic demands". In: *International Transactions in Operational Research* 26.2, pp. 458–484.

Caramia, Massimiliano and Francesca Guerriero (2010). "A milk collection problem with incompatibility constraints". In: *Interfaces* 40.2, pp. 130–143.

Carvalho, Helena, Ana P Barroso, Virgínia H Machado, Susana Azevedo, and Virgilio Cruz-Machado (2012). "Supply chain redesign for resilience using simulation". In: *Computers & Industrial Engineering* 62.1, pp. 329–341.

Chao, I-Ming, Bruce L Golden, and Edward A Wasil (1996). "The team orienteering problem". In: *European journal of operational research* 88.3, pp. 464–474.

Chen, Huey-Kuo, Che-Fu Hsueh, and Mei-Shiang Chang (2009). "Production scheduling and vehicle routing with time windows for perishable food products". In: *Computers & operations research* 36.7, pp. 2311–2319.

Chen, Rui, Xinwu Qian, Lixin Miao, and Satish V Ukkusuri (2020). "Optimal charging facility location and capacity for electric vehicles considering route choice and charging time equilibrium". In: *Computers & Operations Research* 113, p. 104776.

Chiadamrong, N and V Piyathanavong (2017). "Optimal design of supply chain network under uncertainty environment using hybrid analytical and simulation modeling approach". In: *Journal of Industrial Engineering International* 13.4, pp. 465–478.

Chica, Manuel, Angel A Juan, Christopher Bayliss, O Cordon, and W. David Kelton (2020). "Why simheuristics? Benefits, limitations, and best practices when combining metaheuristics with simulation". In: *SORT-Statistics and Operations Research Transactions* 44.2, pp. 1–24.

Choi, Hosoon, Piyali Chatterjee, John D Coppin, Julie A Martel, Munok Hwang, Chetan Jinadatha, and Virender K Sharma (2021). "Current Understanding of the Surface Contamination and Contact Transmission of SARS-CoV-2 in Healthcare Settings". In: *Environmental Chemistry Letters*, pp. 1–10.

Christofides, Nicos and Samuel Eilon (1969). "An algorithm for the vehicle-dispatching problem". In: *Journal of the Operational Research Society* 20.3, pp. 309–318.

Christopher, M. and H. Peck (2004). "Building the Resilient Supply Chain". In: *The International Journal of Logistics Management* 15.2, pp. 1–14.

Çimen, Mustafa and Mehmet Soysal (2017). "Time-dependent green vehicle routing problem with stochastic vehicle speeds: An approximate dynamic programming algorithm". In: *Transportation Research Part D: Transport and Environment* 54, pp. 82–98.

Clarke, Geoff and John W Wright (1964). "Scheduling of vehicles from a central depot to a number of delivery points". In: *Operations Research* 12.4, pp. 568–581.

Coelho, Leandro C, Jean-François Cordeau, and Gilbert Laporte (2013). "Thirty years of inventory routing". In: *Transportation Science* 48.1, pp. 1–19.

Coelho, Leandro C and Gilbert Laporte (2015). "Classification, models and exact algorithms for multi-compartment delivery problems". In: *European Journal of Operational Research* 242.3, pp. 854–864.

Corlu, Canan G, Javier Panadero, Stephan Onggo, and Angel A Juan (2020). "On the scarcity of observations when modelling random inputs and the quality of solutions to stochastic optimisation problems". In: *Proceedings of the 2020 Winter Simulation Conference*. IEEE. Piscataway, New Jersey, pp. 2105–2113.

Correia, Isabel and Teresa Melo (2016). "Multi-period capacitated facility location under delayed demand satisfaction". In: *European Journal of Operational Research* 255.3, pp. 729–746.

Correia, Isabel, Stefan Nickel, and Francisco Saldanha-da Gama (2010). "Single-assignment hub location problems with multiple capacity levels". In: *Transportation Research Part B: Methodological* 44.8-9, pp. 1047–1066.

Correll, David, Yoshinori Suzuki, and Bobby J Martens (2014). "Logistical supply chain design for bioeconomy applications". In: *Biomass and Bioenergy* 66, pp. 60–69.

Costa-Salas, Yasel, William Sarache, and Margarethe Überwimmer (2017). "Fleet size optimization in the discarded tire collection process". In: *Research in Transportation Business & Management* 24, pp. 81–89.

Crainic, Teodor Gabriel, Guido Perboli, and Mariangela Rosano (2018). "Simulation of intermodal freight transportation systems: a taxonomy". In: *European Journal of Operational Research* 270.2, pp. 401–418.

Dantzig, George B and John H Ramser (1959). "The truck dispatching problem". In: *Management science* 6.1, pp. 80–91.

De Armas, Jesica, Angel A Juan, Joan M Marquès, and João Pedro Pedroso (2017). "Solving the deterministic and stochastic uncapacitated facility location problem: from a heuristic to a simheuristic". In: *Journal of the Operational Research Society* 68.10, pp. 1161–1176.

Dehghanimohammadabadi, Mohammad, Mandana Rezaeiahari, and Thomas K Keyser (2017). "Simheuristic of patient scheduling using a table-experiment approach—Simio and Matlab integration application". In: *2017 Winter Simulation Conference (WSC)*. IEEE, pp. 2929–2939.

Deif, I and L Bodin (1984). "Extension of the Clarke and Wright algorithm for solving the vehicle routing problem with backhauling". In: *Proceedings of the Babson conference on software uses in transportation and logistics management*. Babson Park, MA, pp. 75–96.

Derigs, Ulrich, Jens Gottlieb, Jochen Kalkoff, Michael Piesche, Franz Rothlauf, and Ulrich Vogel (2011). "Vehicle routing with compartments: applications, modelling and heuristics". In: *OR spectrum* 33.4, pp. 885–914.

Dharmaraj, Selvakumar, Veeramuthu Ashokkumar, Sneha Hariharan, Akila Manibharathi, Pau Loke Show, Cheng Tung Chong, and Chawalit Ngamcharussrivichai (2021). "The COVID-19 pandemic face mask waste: a blooming threat to the marine environment". In: *Chemosphere* 272, p. 129601.

Ding, Hongwei, Lyes Benyoucef, and Xiaolan Xie (2009). "Stochastic multi-objective production-distribution network design using simulation-based optimization". In: *International Journal of Production Research* 47.2, pp. 479–505.

Dixit, Vijaya, Navaneeth Seshadrinath, and MK Tiwari (2016). "Performance measures based optimization of supply chain network resilience: A NSGA-II+ Co-Kriging approach". In: *Computers & Industrial Engineering* 93, pp. 205–214.

Dolgui, Alexandre, Dmitry Ivanov, and Boris Sokolov (2018). "Ripple effect in the supply chain: an analysis and recent literature". In: *International Journal of Production Research* 56.1-2, pp. 414–430.

Dominguez, Oscar, Daniel Guimarans, Angel A Juan, and Ignacio de la Nuez (2016a). "A biased-randomised large neighbourhood search for the two-dimensional vehicle routing problem with backhauls". In: *European Journal of Operational Research* 255.2, pp. 442–462.

Dominguez, Oscar, Angel A Juan, and Javier Faulin (2014). "A biased-randomized algorithm for the two-dimensional vehicle routing problem with and without item rotations". In: *International Transactions in Operational Research* 21.3, pp. 375–398.

Dominguez, Oscar, Angel A Juan, Ignacio de la Nuez, and Djamila Ouelhadj (2016b). "An ILS-biased randomization algorithm for the two-dimensional loading HFVRP with sequential loading and items rotation". In: *Journal of the Operational Research Society* 67.1, pp. 37–53.

Döyen, Alper, Necati Aras, and Gülay Barbarosoğlu (2012). "A two-echelon stochastic facility location model for humanitarian relief logistics". In: *Optimization Letters* 6.6, pp. 1123–1145.

Drexl, Michael and Michael Schneider (2015). "A survey of variants and extensions of the location-routing problem". In: *European Journal of Operational Research* 241.2, pp. 283–308.

Dror, Moshe and Pierre Trudeau (1986). "Stochastic vehicle routing with modified savings algorithm". In: *European Journal of Operational Research* 23.2, pp. 228–235.

Ekşioğlu, Sandra D, Gökçe Palak, Andro Mondala, and Allen Greenwood (2013). "Supply chain designs and management for biocrude production via wastewater treatment". In: *Environmental Progress & Sustainable Energy* 32.1, pp. 139–147.

Elia, Valerio and Maria Grazia Gnoni (2015). "Designing an effective closed loop system for pallet management". In: *International Journal of Production Economics* 170, pp. 730–740.

Elshaer, Raafat and Hadeer Awad (2020). "A taxonomic review of metaheuristic algorithms for solving the vehicle routing problem and its variants". In: *Computers & Industrial Engineering* 140, p. 106242.

Erbeyoğlu, Gökalp and Ümit Bilge (2020). "A robust disaster preparedness model for effective and fair disaster response". In: *European Journal of Operational Research* 280.2, pp. 479–494.

Errico, Fausto, Guy Desaulniers, Michel Gendreau, Walter Rei, and L-M Rousseau (2016). "A priori optimization with recourse for the vehicle routing problem with hard time windows and stochastic service times". In: *European Journal of Operational Research* 249.1, pp. 55–66.

Eskandarpour, Majid, Djamila Ouelhadj, Sara Hatami, Angel A Juan, and Banafsheh Khosravi (2019). "Enhanced multi-directional local search for the bi-objective heterogeneous vehicle routing problem with multiple driving ranges". In: *European Journal of Operational Research* 277.2, pp. 479–491.

Estrada-Moreno, Alejandro, Albert Ferrer, Angel A Juan, Adil Bagirov, and Javier Panadero (2019a). "A biased-randomised algorithm for the capacitated facility location problem with soft constraints". In: *Journal of the Operational Research Society* 0.0, pp. 1–17.

Estrada-Moreno, Alejandro, Albert Ferrer, Angel A Juan, Javier Panadero, and Adil Bagirov (2020). "The non-smooth and bi-objective team orienteering problem with soft constraints". In: *Mathematics* 8.9, p. 1461.

Estrada-Moreno, Alejandro, Christian Fikar, Angel A Juan, and Patrick Hirsch (2019b). "A biased-randomized algorithm for redistribution of perishable food inventories in supermarket chains". In: *International Transactions in Operational Research* 26.6, pp. 2077–2095.

Fagnant, Daniel J and Kara M Kockelman (2018). "Dynamic ride-sharing and fleet sizing for a system of shared autonomous vehicles in Austin, Texas". In: *Transportation* 45.1, pp. 143–158.

Fan, Xiangxiang, Yeming Gong, Xianhao Xu, and Bipan Zou (2019). "Optimal decisions in reducing loss rate of returnable transport items". In: *Journal of cleaner production* 214, pp. 1050–1060.

Farahani, Reza Zanjirani, Nasrin Asgari, Nooshin Heidari, Mahtab Hosseininia, and Mark Goh (2012). "Covering problems in facility location: A review". In: *Computers & Industrial Engineering* 62.1, pp. 368–407.

Farahani, Reza Zanjirani, Masoud Hekmatfar, Alireza Boloori Arabani, and Ehsan Nikbakhsh (2013). "Hub location problems: A review of models, classification, solution techniques, and applications". In: *Computers & Industrial Engineering* 64.4, pp. 1096–1109.

Farahani, Reza Zanjirani, Maryam SteadieSeifi, and Nasrin Asgari (2010). "Multiple criteria facility location problems: A survey". In: *Applied mathematical modelling* 34.7, pp. 1689–1709.

Faulin, Javier, Miquel Gilibert, Angel A Juan, Xavier Vilajosana, and Ruben Ruiz (2008). "SR-1: A simulation-based algorithm for the capacitated vehicle routing problem". In: *2008 Winter Simulation Conference*. IEEE, pp. 2708–2716.

Faulin, Javier, Scott Grasman, Angel Juan, and Patrick Hirsch (2018). *Sustainable Transportation and Smart Logistics: Decision-Making Models and Solutions*. Elsevier.

Fausto, F., A. Reyna-Orta, E. Cuevas, A.G. Andrade, and M. Perez-Cisneros (2020). "From ants to whales: Metaheuristics for all tastes". In: *Artificial Intelligence Review* 53, 753—810.

Fazayeli, Saeed, Alireza Eydi, and Isa Nakhai Kamalabadi (2018). "Location-routing problem in multimodal transportation network with time windows and fuzzy demands: presenting a two-part genetic algorithm". In: *Computers & Industrial Engineering* 119, pp. 233–246.

Ferone, Daniele, Aljoscha Gruler, Paola Festa, and Angel A Juan (2019). "Enhancing and extending the classical GRASP framework with biased randomisation and simulation". In: *Journal of the Operational Research Society* 70.8, pp. 1362–1375.

Ferone, Daniele, Sara Hatami, Eliana M González-Neira, Angel A Juan, and Paola Festa (2020). "A biased-randomized iterated local search for the distributed assembly permutation flow-shop problem". In: *International Transactions in Operational Research* 27.3, pp. 1368–1391.

Ferrer, Alberto, Daniel Guimarans, Helena Ramalhinho, and Angel A Juan (2016). "A BRILS metaheuristic for non-smooth flow-shop problems with failure-risk costs". In: *Expert Systems with Applications* 44, pp. 177–186.

Figueira, Gonçalo and Bernardo Almada-Lobo (2014). "Hybrid simulation–optimization methods: A taxonomy and discussion". In: *Simulation Modelling Practice and Theory* 46, pp. 118–134.

Fikar, Christian, Angel A Juan, Enoc Martinez, and Patrick Hirsch (2016). "A discrete-event driven metaheuristic for dynamic home service routing with synchronised trip sharing". In: *European Journal of Industrial Engineering* 10.3, pp. 323–340.

Frade, Inês, Anabela Ribeiro, Gonçalo Gonçalves, and António Pais Antunes (2011). "Optimal location of charging stations for electric vehicles in a neighborhood in Lisbon, Portugal". In: *Transportation Research Record* 2252.1, pp. 91–98.

Fu, Michael C (1994). "Optimization via simulation: A review". In: *Annals of operations research* 53.1, pp. 199–247.

Fu, Michael C, Fred W Glover, and Jay April (2005). "Simulation optimization: a review, new developments, and applications". In: *Proceedings of the Winter Simulation Conference, 2005.* IEEE, 13–pp.

García-Nájera, Abel, John A Bullinaria, and Miguel A Gutiérrez-Andrade (2015). "An evolutionary approach for multi-objective vehicle routing problems with backhauls". In: *Computers & Industrial Engineering* 81, pp. 90–108.

Gavish, Bezalel and Stephen C Graves (1978). *The travelling salesman problem and related problems (Working Paper)*. Tech. rep. Massachusetts Institute of Technology, Operations Research Center.

Gendreau, Michel, Ola Jabali, and Walter Rei (2016). "50th anniversary invited article—future research directions in stochastic vehicle routing". In: *Transportation Science* 50.4, pp. 1163–1173.

Gendreau, Michel, Gilbert Laporte, and René Séguin (1996). "Stochastic vehicle routing". In: *European Journal of Operational Research* 88.1, pp. 3–12.

Gestió Extracentre dels Residus Sanitaris (2019). Available at `http://residus.gencat.cat/web/.content/home/ambits_dactuacio/tipus_de_residu/residus_sanitaris/esquema_de_gestio/sanitaris02.pdf`. Generalitat de Catalunya. (Visited on 03/24/2021).

Ghannadpour, Seyed Farid, Simak Noori, Reza Tavakkoli-Moghaddam, and Keivan Ghoseiri (2014). "A multi-objective dynamic vehicle routing problem with fuzzy time windows: Model, solution and application". In: *Applied Soft Computing* 14, pp. 504–527.

Ghezavati, Vahidreza and Shabnam Morakabatchian (2015). "Application of a fuzzy service level constraint for solving a multi-objective location-routing problem for the industrial hazardous wastes". In: *Journal of Intelligent & Fuzzy Systems* 28.5, pp. 2003–2013.

Glock, Christoph H (2017). "Decision support models for managing returnable transport items in supply chains: A systematic literature review". In: *International Journal of Production Economics* 183, pp. 561–569.

Goetschalckx, Marc and Charlotte Jacobs-Blecha (1989). "The vehicle routing problem with backhauls". In: *European Journal of Operational Research* 42.1, pp. 39–51.

Goldbeck, Nils, Panagiotis Angeloudis, and Washington Ochieng (2020). "Optimal supply chain resilience with consideration of failure propagation and repair logistics". In: *Transportation Research Part E: Logistics and Transportation Review* 133, p. 101830.

Golden, Bruce L, Larry Levy, and Rakesh Vohra (1987). "The orienteering problem". In: *Naval Research Logistics (NRL)* 34.3, pp. 307–318.

Gonçalves, José Fernando and Mauricio GC Resende (2011). "Biased random-key genetic algorithms for combinatorial optimization". In: *Journal of Heuristics* 17.5, pp. 487–525.

Gonzalez-Feliu, Jesus, Frédéric Semet, and Jean-Louis Routhier (2014). *Sustainable urban logistics: Concepts, methods and information systems*. Springer.

González-Hernández, Isidro Jesús, Rafael Granillo-Macías, José Luis Martínez-Flores, Diana Sánchez-Partida, and Damián Emilio Gibaja-Romero (2019). "HYBRID MODEL TO DESIGN AN AGRO-FOOD DISTRIBUTION NETWORK CONSIDERING FOOD QUALITY." In: *International Journal of Industrial Engineering* 26.4, pp. 588–609.

Gonzalez-Martin, Sergio, Angel A Juan, Daniel Riera, Quim Castella, Rodrigo Muñoz, and Alejandra Perez (2012). "Development and assessment of the SHARP and RandSHARP algorithms for the arc routing problem". In: *AI Communications* 25.2, pp. 173–189.

Gonzalez-Martin, Sergio, Angel A Juan, Daniel Riera, Monica G Elizondo, and Juan J Ramos (2018). "A simheuristic algorithm for solving the arc routing problem with stochastic demands". In: *Journal of Simulation* 12.1, pp. 53–66.

Gonzalez-Neira, Eliana Maria, Daniele Ferone, Sara Hatami, and Angel A Juan (2017). "A biased-randomized simheuristic for the distributed assembly permutation flowshop problem with stochastic processing times". In: *Simulation Modelling Practice and Theory* 79, pp. 23–36.

Govindan, Kannan, Mohammad Fattahi, and Esmaeil Keyvanshokooh (2017). "Supply chain network design under uncertainty: A comprehensive review and future research directions". In: *European Journal of Operational Research* 263.1, pp. 108–141.

Govindan, Kannan, Ahmad Jafarian, Roohollah Khodaverdi, and Kannan Devika (2014). "Two-echelon multiple-vehicle location–routing problem with time windows for optimization of sustainable supply chain network of perishable food". In: *International Journal of Production Economics* 152, pp. 9–28.

Govindan, Kannan and Hamed Soleimani (2017). "A review of reverse logistics and closed-loop supply chains: a Journal of Cleaner Production focus". In: *Journal of Cleaner Production* 142, pp. 371–384.

Grasas, Alex, Angel A Juan, Javier Faulin, Jesica de Armas, and Helena Ramalhinho (2017). "Biased randomization of heuristics using skewed probability distributions: a survey and some applications". In: *Computers & Industrial Engineering* 110, pp. 216–228.

Grasas, Alex, Angel A Juan, and Helena R Lourenço (2016). "SimILS: a simulation-based extension of the iterated local search metaheuristic for stochastic combinatorial optimization". In: *Journal of Simulation* 10.1, pp. 69–77.

Greene, David L, Sangsoo Park, and Changzheng Liu (2014). "Public policy and the transition to electric drive vehicles in the US: The role of the zero emission vehicles mandates". In: *Energy Strategy Reviews* 5, pp. 66–77.

Gribkovskaia, Irina, Gilbert Laporte, and Aliaksandr Shyshou (2008). "The single vehicle routing problem with deliveries and selective pickups". In: *Computers & Operations Research* 35.9, pp. 2908–2924.

Gruler, Aljoscha, Jésica de Armas Adrián, Angel A Juan, and David Goldsman (2019). "Modelling human network behaviour using simulation and optimization tools: the need for hybridization". In: *SORT: statistics and operations research transactions* 43.2, pp. 0193–222.

Gruler, Aljoscha, Christian Fikar, Angel A Juan, Patrick Hirsch, and C Contreras-Bolton (2017a). "Supporting multi-depot and stochastic waste collection management in clustered urban areas via simulation–optimization". In: *Journal of Simulation* 11.1, pp. 11–19.

Gruler, Aljoscha, Angel A Juan, Carlos Contreras-Bolton, and Gustavo Gatica (2015). "A biased-randomized heuristic for the waste collection problem in smart cities". In: *International Forum for Interdisciplinary Mathematics*. Springer, pp. 255–263.

Gruler, Aljoscha, Javier Panadero, Jesica de Armas, Jose A Moreno, and Angel A Juan (2018). "Combining variable neighborhood search with simulation for the inventory routing problem with stochastic demands and stock-outs". In: *Computers & Industrial Engineering* 123, pp. 278–288.

— (2020a). "A variable neighborhood search simheuristic for the multiperiod inventory routing problem with stochastic demands". In: *International Transactions in Operational Research* 27.1, pp. 314–335.

Gruler, Aljoscha, Antoni Pérez-Navarro, Laura Calvet, and Angel A Juan (2020b). "A simheuristic algorithm for time-dependent waste collection management with stochastic travel times". In: *SORT-Statistics and Operations Research Transactions*, pp. 285–310.

Gruler, Aljoscha, Carlos L Quintero-Araújo, Laura Calvet, and Angel A Juan (2017b). "Waste collection under uncertainty: a simheuristic based on variable neighbourhood search". In: *European Journal of Industrial Engineering* 11.2, pp. 228–255.

Guerrero, Jessica Calderón and Jenny Díaz-Ramírez (2017). "A review on transportation last-mile network design and urban freight vehicles". In: *Proceedings of the 2017 International Conference on Industrial Engineering and Operations Management*, pp. 533–552.

Guerrero, William Javier, Laura Andrea Sotelo-Cortés, and Enrique Romero-Mota (2018). "Simulation-optimization techniques for closed-loop supply chain design with multiple objectives". In: *Dyna* 85.206, pp. 202–210.

Guimarans, Daniel, Oscar Dominguez, Javier Panadero, and Angel A Juan (2018). "A simheuristic approach for the two-dimensional vehicle routing problem with stochastic travel times". In: *Simulation Modelling Practice and Theory* 89, pp. 1–14.

Gumus, Alev Taskin, Ali Fuat Guneri, and Selcan Keles (2009). "Supply chain network design using an integrated neuro-fuzzy and MILP approach: A comparative design study". In: *Expert Systems with Applications* 36.10, pp. 12570–12577.

Gutiérrez-Jarpa, Gabriel, Guy Desaulniers, Gilbert Laporte, and Vladimir Marianov (2010). "A branch-and-price algorithm for the vehicle routing problem with deliveries, selective pickups and time windows". In: *European Journal of Operational Research* 206.2, pp. 341–349.

Hanafi, Saïd, Renata Mansini, and Roberto Zanotti (2020). "The multi-visit team orienteering problem with precedence constraints". In: *European journal of operational research* 282.2, pp. 515–529.

Harrison, Colin G and Peter R Williams (2016). "A systems approach to natural disaster resilience". In: *Simulation Modelling Practice and Theory* 65, pp. 11–31.

Hashimoto, Hideki, Toshihide Ibaraki, Shinji Imahori, and Mutsunori Yagiura (2006). "The vehicle routing problem with flexible time windows and traveling times". In: *Discrete Applied Mathematics* 154.16, pp. 2271–2290.

Hatami, Sara, Laura Calvet, Victor Fernández-Viagas, José M Framiñán, and Angel A Juan (2018). "A simheuristic algorithm to set up starting times in the stochastic parallel flowshop problem". In: *Simulation Modelling Practice and Theory* 86, pp. 55–71.

Health Care Waste (2021). Available at `http://residus.gencat.cat/en/ambits_dactuacio/tipus_de_residu/sanitaris/index.html`. Generalitat de Catalunya. (Visited on 03/24/2021).

Heckmann, Iris, Tina Comes, and Stefan Nickel (2015). "A critical review on supply chain risk – Definition, measure and modeling". In: *Omega* 52, pp. 119–132. ISSN: 0305-0483.

Hemmelmayr, Vera, Karen Smilowitz, and Luis de la Torre (2017). "A periodic location routing problem for collaborative recycling". In: *IISE Transactions* 49.4, pp. 414–428.

Herazo-Padilla, Nilson, Jairo R Montoya-Torres, Santiago Nieto Isaza, and J Alvarado-Valencia (2015). "Simulation-Optimization Approach for the Stochastic Location-Routing Problem". In: *Journal of Simulation* 9.4, pp. 296–311.

Hiermann, Gerhard, Matthias Prandtstetter, Andrea Rendl, Jakob Puchinger, and Günther R Raidl (2015). "Metaheuristics for solving a multimodal home-healthcare scheduling problem". In: *Central European Journal of Operations Research* 23.1, pp. 89–113.

Hohenstein, Nils-Ole, Edda Feisel, Evi Hartmann, and Larry Giunipero (2015). "Research on the phenomenon of supply chain resilience: a systematic review and paths for further investigation". In: *International Journal of Physical Distribution & Logistics Management* 45.1/2, pp. 90–117.

Holguín-Veras, José, Miguel Jaller, Luk N Van Wassenhove, Noel Pérez, and Tricia Wachtendorf (2012). "On the unique features of post-disaster humanitarian logistics". In: *Journal of Operations Management* 30.7-8, pp. 494–506.

Homayouni, S Mahdi, Dalila BMM Fontes, and José F Gonçalves (2020). "A multistart biased random key genetic algorithm for the flexible job shop scheduling problem with transportation". In: *International Transactions in Operational Research*.

Hooftman, Nils, Maarten Messagie, Joeri Van Mierlo, and Thierry Coosemans (2020). "The Paris Agreement and Zero-Emission Vehicles in Europe: Scenarios for the Road Towards a Decarbonised Passenger Car Fleet". In: *Towards User-Centric Transport in Europe 2*. Springer, pp. 151–168.

Hornstra, Richard P, Allyson Silva, Kees Jan Roodbergen, and Leandro C Coelho (2020). "The vehicle routing problem with simultaneous pickup and delivery and handling costs". In: *Computers & Operations Research* 115, p. 104858.

Hosni, Hadi, Joe Naoum-Sawaya, and Hassan Artail (2014). "The shared-taxi problem: Formulation and solution methods". In: *Transportation Research Part B: Methodological* 70, pp. 303–318.

Hübner, Alexander and Manuel Ostermeier (2019). "A multi-compartment vehicle routing problem with loading and unloading costs". In: *Transportation Science* 53.1, pp. 282–300.

Husakou, Anatol, Lars Magnus Hvattum, Ketil Danielsen, and Arild Hoff (2020). "An application of the multi-depot heterogeneous fixed fleet open vehicle routing problem". In: *International Journal of Advanced Operations Management* 12.2, pp. 142–155.

Ilic, Alexander, Jason WP Ng, Paul Bowman, and Thorsten Staake (2009). "The value of RFID for RTI management". In: *Electronic Markets* 19.2-3, pp. 125–135.

ISO (2016). *ISO/IEC 19762:2016. Information technology – Automatic identification and data capture (AIDC) techniques – Harmonized vocabulary*.

Ivanov, Dmitry (2020). "Predicting the impacts of epidemic outbreaks on global supply chains: A simulation-based analysis on the coronavirus outbreak (COVID-19/SARS-CoV-2) case". In: *Transportation Research Part E: Logistics and Transportation Review* 136, p. 101922.

Ivanov, Dmitry and Alexandre Dolgui (2020). "Viability of intertwined supply networks: extending the supply chain resilience angles towards survivability. A position paper motivated by COVID-19 outbreak". In: *International Journal of Production Research* 58.10, pp. 2904–2915.

Iwan, Stanisław, Kinga Kijewska, and Justyna Lemke (2016). "Analysis of parcel lockers' efficiency as the last mile delivery solution–the results of the research in Poland". In: *Transportation Research Procedia* 12, pp. 644–655.

Jacobs-Blecha, Charlotte and Marc Goetschalckx (1992). "The vehicle routing problem with backhauls: properties and solution algorithms". In: *National Transportation Research Board* 13.

Jacobsen, S Kruse and Oli BG Madsen (1980). "A comparative study of heuristics for a two-level routing-location problem". In: *European Journal of Operational Research* 5.6, pp. 378–387.

Jaillet, Patrick, Jin Qi, and Melvyn Sim (2016). "Routing optimization under uncertainty". In: *Operations research* 64.1, pp. 186–200.

Ji, Xiaoyu, Xiande Zhao, and Deming Zhou (2007). "A fuzzy programming approach for supply chain network design". In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 15.supp02, pp. 75–87.

Jiménez-Meza, A, J Arámburo-Lizárraga, and E de la Fuente (2013). "Framework for estimating travel time, distance, speed, and street segment level of service (los), based on GPS data". In: *Procedia Technology* 7, pp. 61–70.

Juan, A A, J Faulin, S E Grasman, D Riera, J Marull, and C Mendez (2011). "Using safety stocks and simulation to solve the vehicle routing problem with stochastic demands". In: *Transportation Research Part C: Emerging Technologies* 19.5, pp. 751–765.

Juan, A A, A Freixes, J Panadero, C Serrat, and A Estrada-Moreno (2020a). "Routing Drones in Smart Cities: a Biased-Randomized Algorithm for Solving the Team Orienteering Problem in Real Time". In: *Transportation Research Procedia* 47, pp. 243–250.

Juan, Angel A., Javier Faulin, Albert Ferrer, Helena R. Lourenço, and Barry Barrios (2013a). "MIRHA: multi-start biased randomization of heuristics with adaptive local search for solving non-smooth routing problems". In: *TOP* 21.1, pp. 109–132.

Juan, Angel A, Javier Faulin, Scott E Grasman, Markus Rabe, and Gonçalo Figueira (2015a). "A review of simheuristics: Extending metaheuristics to deal with stochastic combinatorial optimization problems". In: *Operations Research Perspectives* 2, pp. 62–72.

Juan, Angel A, Javier Faulin, Josep Jorba, Jose Caceres, and Joan Manuel Marquès (2013b). "Using parallel & distributed computing for real-time solving of vehicle routing problems with stochastic demands". In: *Annals of Operations Research* 207.1, pp. 43–65.

Juan, Angel A, Javier Faulin, Rubén Ruiz, Barry Barrios, Miquel Gilibert, and Xavier Vilajosana (2009). "Using Oriented Random Search to Provide a Set of Alternative Solutions to the Capacitated Vehicle Routing Problem". In: *Operations Research and Cyber-Infrastructure*. Springer, pp. 331–345.

Juan, Angel A, W David Kelton, Christine SM Currie, and Javier Faulin (2018). "Simheuristics applications: dealing with uncertainty in logistics, transportation, and other supply chain areas". In: *Proceedings of the 2018 Winter Simulation Conference*. IEEE Press, pp. 3048–3059.

Juan, Angel A, Helena R Lourenço, Manuel Mateo, Rachel Luo, and Quim Castella (2014). "Using iterated local search for solving the flow-shop problem: Parallelization, parametrization, and randomization issues". In: *International Transactions in Operational Research* 21.1, pp. 103–126.

Juan, Angel A, Iñaki Pascual, Daniel Guimarans, and Barry Barrios (2015b). "Combining biased randomization with iterated local search for solving the multidepot vehicle routing problem". In: *International Transactions in Operational Research* 22.4, pp. 647–667.

Juan, Angel Alejandro, Canan Gunes Corlu, Rafael David Tordecilla, Rocio de la Torre, and Albert Ferrer (2020b). "On the Use of Biased-Randomized Algorithms for Solving Non-Smooth Optimization Problems". In: *Algorithms* 13.1, p. 8.

Juan, Angel Alejandro, Carlos Alberto Mendez, Javier Faulin, Jesica De Armas, and Scott Erwin Grasman (2016). "Electric vehicles in logistics and transportation: A survey on emerging environmental, strategic, and operational challenges". In: *Energies* 9.2, p. 86.

Jung, Jaeyoung, Joseph YJ Chow, R Jayakrishnan, and Ji Young Park (2014). "Stochastic dynamic itinerary interception refueling location problem with queue delay for electric taxi charging stations". In: *Transportation Research Part C: Emerging Technologies* 40, pp. 123–142.

Kapustin, Nikita O and Dmitry A Grushevenko (2020). "Long-term electric vehicles outlook and their potential impact on electric grid". In: *Energy Policy* 137, p. 111103.

Karunakaran, Deepak, Yi Mei, and Mengjie Zhang (2019). "Multitasking Genetic Programming for Stochastic Team Orienteering Problem with Time Windows". In: *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, pp. 1598–1605.

Keenan, Peter, Javier Panadero, Angel A Juan, Rafael Martí, and Seán McGarraghy (2021). "A strategic oscillation simheuristic for the Time Capacitated Arc Routing Problem with stochastic demands". In: *Computers & Operations Research* 133, p. 105377.

Keizer, Marlies de, René Haijema, Jacqueline M Bloemhof, and Jack GAJ Van Der Vorst (2015). "Hybrid optimization and simulation to design a logistics network for distributing perishable products". In: *Computers & Industrial Engineering* 88, pp. 26–38.

Kenyon, Astrid S and David P Morton (2003). "Stochastic vehicle routing with random travel times". In: *Transportation Science* 37.1, pp. 69–82.

Keshtkaran, Morteza, Koorush Ziarati, Andrea Bettinelli, and Daniele Vigo (2016). "Enhanced Exact Solution Methods for the Team Orienteering Problem". In: *International Journal of Production Research* 54.2, pp. 591–601.

Kim, Jinkyung, Matthew J Realff, and Jay H Lee (2011). "Optimal design and global sensitivity analysis of biomass supply chain networks for biofuels under uncertainty". In: *Computers & Chemical Engineering* 35.9, pp. 1738–1751.

Kim, Taebok, Christoph H Glock, and Yongjang Kwon (2014). "A closed-loop supply chain for deteriorating products under stochastic container return times". In: *Omega* 43, pp. 30–40.

Kizys, Renatas, Angel A Juan, Bartosz Sawik, and Laura Calvet (2019). "A Biased-Randomized Iterated Local Search Algorithm for Rich Portfolio Optimization". In: *Applied Sciences* 9.17, p. 3509.

Klir, George and Bo Yuan (1995). *Fuzzy Sets and Fuzzy Logic: Theory and Applications*. New Jersey: Pearson.

Klose, Andreas and Andreas Drexl (2005). "Facility location models for distribution system design". In: *European journal of operational research* 162.1, pp. 4–29.

Ko, Hyun Jeung, Chang Seong Ko, and Taioun Kim (2006). "A hybrid optimization/simulation approach for a distribution network design of 3PLS". In: *Computers & Industrial Engineering* 50.4, pp. 440–449.

Koç, Çağrı and Gilbert Laporte (2018). "Vehicle routing with backhauls: Review and research perspectives". In: *Computers & Operations Research* 91, pp. 79–91.

Koç, Çağrı, Gilbert Laporte, and İlknur Tükenmez (2020). "A Review on Vehicle Routing with Simultaneous Pickup and Delivery". In: *Computers & Operations Research*, p. 104987.

Koo, Lee Ying, Arief Adhitya, Rajagopalan Srinivasan, and Iftekhar A Karimi (2008). "Decision support for integrated refinery supply chains: Part 2. Design and operation". In: *Computers & Chemical Engineering* 32.11, pp. 2787–2800.

Kristianto, Yohanes and Liandong Zhu (2017). "Techno-economic optimization of ethanol synthesis from rice-straw supply chains". In: *Energy* 141, pp. 2164–2176.

Kroon, Leo and Gaby Vrijens (1995). "Returnable containers: an example of reverse logistics". In: *International Journal of Physical Distribution & Logistics Management* 25.2, pp. 56–68.

Labadie, Nacima, Renata Mansini, Jan Melechovskỳ, and Roberto Wolfler Calvo (2012). "The Team Orienteering Problem with Time Windows: An LP-Based Granular Variable Neighborhood Search". In: *European Journal of Operational Research* 220.1, pp. 15–27.

Lahyani, Rahma, Leandro C Coelho, Mahdi Khemakhem, Gilbert Laporte, and Frédéric Semet (2015a). "A multi-compartment vehicle routing problem arising in the collection of olive oil in Tunisia". In: *Omega* 51, pp. 1–10.

Lahyani, Rahma, Anne-Lise Gouguenheim, and Leandro C Coelho (2019). "A hybrid adaptive large neighbourhood search for multi-depot open vehicle routing problems". In: *International Journal of Production Research* 57.22, pp. 6963–6976.

Lahyani, Rahma, Mahdi Khemakhem, and Frédéric Semet (2015b). "Rich vehicle routing problems: From a taxonomy to a definition". In: *European Journal of Operational Research* 241.1, pp. 1–14.

Lam, Chiou-Peng, Martin Masek, Luke Kelly, Michael Papasimeon, and Lyndon Benke (2019). "A simheuristic approach for evolving agent behaviour in the exploration for novel combat tactics". In: *Operations Research Perspectives* 6, p. 100123.

Laporte, G., F. Louveaux, and H. Mercure (1989). "Models and exact solutions for a class of stochastic location-routing problems". In: *European Journal of Operational Research* 39.1, pp. 71–78.

Laporte, Gilbert (1992). "The vehicle routing problem: An overview of exact and approximate algorithms". In: *European journal of operational research* 59.3, pp. 345–358.

Laporte, Gilbert, Hélène Mercure, and Yves Nobert (1986). "An exact algorithm for the asymmetrical capacitated vehicle routing problem". In: *Networks* 16.1, pp. 33–46.

Laporte, Gilbert, Yves Nobert, and Serge Taillefer (1988). "Solving a family of multi-depot vehicle routing and location-routing problems". In: *Transportation Science* 22.3, pp. 161–172.

Lashine, Sherif H, Mohamed Fattouh, and Abeer Issa (2006). "Location/Allocation and Routing Decisions in Supply Chain Network Design". In: *Journal of Modelling in Management* 1.2, pp. 173–183.

Latorre-Biel, Juan I, Daniele Ferone, Angel A Juan, and Javier Faulin (2021). "Combining simheuristics with Petri nets for solving the stochastic vehicle routing problem with correlated demands". In: *Expert Systems with Applications*, p. 114240.

Law, Averill M (2013). *Simulation Modeling and Analysis*. 5th ed. Tucson, Arizona: McGraw-Hill.

Lenstra, Jan Karel and AHG Rinnooy Kan (1981). "Complexity of vehicle routing and scheduling problems". In: *Networks* 11.2, pp. 221–227.

Leonzio, Grazia, Pier Ugo Foscolo, and Edwin Zondervan (2019). "Sustainable utilization and storage of carbon dioxide: analysis and design of an innovative supply chain". In: *Computers & Chemical Engineering* 131, p. 106569.

Li, Feiyue, Bruce Golden, and Edward Wasil (2007). "The open vehicle routing problem: Algorithms, large-scale test problems, and computational results". In: *Computers & operations research* 34.10, pp. 2918–2930.

Li, Xiangyong, Peng Tian, and Stephen CH Leung (2010). "Vehicle routing problems with time windows and stochastic travel and service times: Models and algorithm". In: *International Journal of Production Economics* 125.1, pp. 137–145.

Li, Yongbo, Hamed Soleimani, and Mostafa Zohal (2019). "An improved ant colony optimization algorithm for the multi-depot green vehicle routing problem with multiple objectives". In: *Journal of cleaner production* 227, pp. 1161–1172.

Li, Yuhong and Christopher W Zobel (2020). "Exploring supply chain network resilience in the presence of the ripple effect". In: *International Journal of Production Economics*, p. 107693.

Lin, Jie, Wei Yu, Nan Zhang, Xinyu Yang, Hanlin Zhang, and Wei Zhao (2017). "A survey on internet of things: Architecture, enabling technologies, security and privacy, and applications". In: *IEEE internet of things journal* 4.5, pp. 1125–1142.

Lin, Shih-Wei and F Yu Vincent (2012). "A Simulated Annealing Heuristic for the Team Orienteering Problem with Time Windows". In: *European Journal of Operational Research* 217.1, pp. 94–107.

Liu, Haoxiang and David ZW Wang (2017). "Locating multiple types of charging facilities for battery electric vehicles". In: *Transportation Research Part B: Methodological* 103, pp. 30–55.

Londoño, Julio C, Rafael D Tordecilla, Leandro do C Martins, and Angel A Juan (2020). "A biased-randomized iterated local search for the vehicle routing problem with optional backhauls". In: *TOP*, pp. 1–30.

Lopes, Rui Borges, Carlos Ferreira, Beatriz Sousa Santos, and Sérgio Barreto (2013). "A taxonomical analysis, current methods and objectives on location-routing problems". In: *International Transactions in Operational Research* 20.6, pp. 795–822.

Lourenço, Helena Ramalhinho, Olivier C Martin, and Thomas Stützle (2019). "Iterated local search: framework and applications". In: *Handbook of Metaheuristics*. Springer, pp. 129–168.

Machado, Cláudia A Soares, Nicolas Patrick Marie de Salles Hue, Fernando Tobal Berssaneti, and José Alberto Quintanilha (2018). "An overview of shared mobility". In: *Sustainability* 10.12, p. 4342.

Madsen, O.B.G. (1983). "Methods for solving combined two level location-routing problems of realistic dimensions". In: *European Journal of Operational Research* 12.3, pp. 295–301.

Mahmoudi, Monirehalsadat and Irandokht Parviziomran (2020). "Reusable packaging in supply chains: A review of environmental and economic impacts, logistics system designs, and operations management". In: *International Journal of Production Economics*, p. 107730.

Mahmoudi, Monirehalsadat and Xuesong Zhou (2016). "Finding optimal solutions for vehicle routing problem with pickup and delivery services with time windows: A dynamic programming approach based on state–space–time network representations". In: *Transportation Research Part B: Methodological* 89, pp. 19–42.

Maier, Holger R, Saman Razavi, Zoran Kapelan, L Shawn Matott, J Kasprzyk, and Bryan A Tolson (2019). "Introductory overview: Optimization using evolutionary algorithms and other metaheuristics". In: *Environmental modelling & software* 114, pp. 195–213.

Maranzana, FE (1964). "On the location of supply points to minimize transport costs". In: *Journal of the Operational Research Society* 15.3, pp. 261–270.

Marinakis, Yannis, Georgia-Roumbini Iordanidou, and Magdalene Marinaki (2013). "Particle swarm optimization for the vehicle routing problem with stochastic demands". In: *Applied Soft Computing* 13.4, pp. 1693–1704.

Marinakis, Yannis, Magdalene Marinaki, and Athanasios Migdalas (2019). "A multi-adaptive particle swarm optimization for the vehicle routing problem with time windows". In: *Information Sciences* 481, pp. 311–329.

Marmol, Mage, Anita Goyal, Pedro Jesus Copado-Mendez, Javier Panadero, and Angel A Juan (2021). "Maximizing customers' lifetime value using limited marketing resources". In: *Marketing Intelligence & Planning*.

Marmol, Mage, Leandro do C Martins, Sara Hatami, Angel A Juan, and Vicenc Fernandez (2020). "Using biased-randomized algorithms for the multi-period product display problem with dynamic attractiveness". In: *Algorithms* 13.2, p. 34.

Martí, Rafael, Mauricio GC Resende, and Celso C Ribeiro (2013). "Multi-start methods for combinatorial optimization". In: *European Journal of Operational Research* 226.1, pp. 1–8.

Martins, Leandro do C, Christopher Bayliss, Pedro J Copado-Méndez, Javier Panadero, and Angel A Juan (2020). "A Simheuristic Algorithm for Solving the Stochastic Omnichannel Vehicle Routing Problem with Pick-up and Delivery". In: *Algorithms* 13.9, p. 237.

Martins, Leandro do C., Patrick Hirsch, and Angel A Juan (2021a). "Agile optimization of a two-echelon vehicle routing problem with pickup and delivery". In: *International Transactions in Operational Research* 28.1, pp. 201–221.

Martins, Leandro do C, Daniele Tarchi, Angel A Juan, and Alessandro Fusco (2021b). "Agile optimization for a real-time facility location problem in Internet of Vehicles networks". In: *Networks*, pp. 1–14.

Martins, Leandro do C, Rocio de la Torre, Canan G Corlu, Angel A Juan, and Mohamed A Masmoudi (2021c). "Optimizing ride-sharing operations in smart sustainable cities: Challenges and the need for agile algorithms". In: *Computers & Industrial Engineering* 153, p. 107080.

Martins, Sara, Pedro Amorim, Gonçalo Figueira, and Bernardo Almada-Lobo (2017). "An optimization-simulation approach to the network redesign problem of pharmaceutical wholesalers". In: *Computers & Industrial Engineering* 106, pp. 315–328.

Masiero, Gilmar, Mario Henrique Ogasavara, Ailton Conde Jussani, and Marcelo Luiz Risso (2016). "Electric vehicles in China: BYD strategies and government subsidies". In: *RAI Revista de Administração e Inovação* 13.1, pp. 3–11.

Mason, Alex, Andy Shaw, and Ahmed Al-Shamma'a (2012). "Peer-to-peer inventory management of returnable transport items: A design science approach". In: *Computers in Industry* 63.3, pp. 265–274.

Mattos, Roberto Gomes de, Fabricio Oliveira, Adriana Leiras, Abdon Baptista de Paula Filho, and Paulo Gonçalves (2019). "Robust optimization of the insecticide-treated bed nets procurement and distribution planning under uncertainty for malaria prevention and control". In: *Annals of Operations Research* 283.1, pp. 1045–1078.

Mazza, D., A. Pagès-Bernaus, D. Tarchi, A. A. Juan, and G. E. Corazza (2018). "Supporting Mobile Cloud Computing in Smart Cities via Randomized Algorithms". In: *IEEE Systems Journal* 12.2, pp. 1598–1609.

Mehrjerdi, Yahia Zare and Ali Nadizadeh (2013). "Using Greedy Clustering Method to Solve Capacitated Location-Routing Problem with Fuzzy Demands". In: *European Journal of Operational Research* 229.1, pp. 75–84.

Mei, Yi and Mengjie Zhang (2018). "Genetic Programming Hyper-heuristic for Stochastic Team Orienteering Problem with Time Windows". In: *2018 IEEE Congress on Evolutionary Computation (CEC)*. IEEE, pp. 1–8.

Melo, M Teresa, Stefan Nickel, and Francisco Saldanha-Da-Gama (2009). "Facility location and supply chain management–A review". In: *European Journal of Operational Research* 196.2, pp. 401–412.

Moroz, Miroslaw and Zdzislaw Polkowski (2016). "The last mile issue and urban logistics: choosing parcel machines in the context of the ecological attitudes of the Y generation consumers purchasing online". In: *Transportation Research Procedia* 16, pp. 378–393.

Mourad, Abood, Jakob Puchinger, and Chengbin Chu (2019). "A survey of models and algorithms for optimizing shared mobility". In: *Transportation Research Part B: Methodological*.

Mukherjee, Sanghamitra Chattopadhyay and Lisa Ryan (2020). "Factors influencing early battery electric vehicle adoption in Ireland". In: *Renewable and Sustainable Energy Reviews* 118, p. 109504.

Muñoz-Villamizar, Andrés, Carlos L Quintero-Araújo, Jairo R Montoya-Torres, and Javier Faulin (2019). "Short-and mid-term evaluation of the use of electric vehicles in urban freight transport collaborative networks: a case study". In: *International Journal of Logistics Research and Applications* 22.3, pp. 229–252.

Muyldermans, Luc and Gu Pang (2010). "On the benefits of co-collection: Experiments with a multi-compartment vehicle routing algorithm". In: *European Journal of Operational Research* 206.1, pp. 93–103.

Nadizadeh, Ali and Behzad Kafash (2019). "Fuzzy capacitated location-routing problem with simultaneous pickup and delivery demands". In: *Transportation Letters* 11.1, pp. 1–19.

Nagy, Gábor and Saïd Salhi (2007). "Location-routing: issues, models and methods". In: *European Journal of Operational Research* 177.2, pp. 649–672.

Nataraj, S, D Ferone, C Quintero-Araujo, A A Juan, and P Festa (2019). "Consolidation centers in city logistics: a cooperative approach based on the location routing problem". In: *International Journal of Industrial Engineering Computations* 10.3, pp. 393–404.

Negri, Elisa, Vibhor Pandhare, Laura Cattaneo, Jaskaran Singh, Marco Macchi, and Jay Lee (2021). "Field-synchronized Digital Twin framework for production scheduling with uncertainty". In: *Journal of Intelligent Manufacturing* 32.4, pp. 1207–1228.

Ni, Yaodong, Yi Chen, Hua Ke, and Dan A Ralescu (2018). "Models and algorithm for the orienteering problem in a fuzzy environment". In: *International Journal of Fuzzy Systems* 20.3, pp. 861–876.

Nickel, Stefan, Francisco Saldanha-da Gama, and Hans-Peter Ziegler (2012). "A multi-stage stochastic supply network design problem with financial decisions and risk management". In: *Omega* 40.5, pp. 511–524.

Nzediegwu, Christopher and Scott X Chang (2020). "Improper solid waste management increases potential for COVID-19 spread in developing countries". In: *Resources, conservation, and recycling* 161, p. 104947.

Oksuz, Mehmet Kursat and Sule Itir Satoglu (2020). "A two-stage stochastic model for location planning of temporary medical centers for disaster response". In: *International Journal of Disaster Risk Reduction* 44, p. 101426.

Oliva, Diego, Pedro Copado, Salvador Hinojosa, Javier Panadero, Daniel Riera, and Angel A Juan (2020). "Fuzzy simheuristics: Solving optimization problems under stochastic and uncertainty scenarios". In: *Mathematics* 8.12, p. 2240.

Oliveira, Josenildo Brito, Mingzhou Jin, Renato Silva Lima, John E Kobza, and José Arnaldo Barra Montevechi (2019). "The role of simulation and optimization methods in supply chain risk management: Performance and review standpoints". In: *Simulation Modelling Practice and Theory* 92, pp. 17–44.

Öncan, Temel, Kuban Altınel, and Gilbert Laporte (2009). "A comparative analysis of several asymmetric traveling salesman problem formulations". In: *Computers & Operations Research* 36.3, pp. 637–654.

O'Neill, Eoin, Dave Moore, Luke Kelleher, and Finbarr Brereton (2019). "Barriers to electric vehicle uptake in Ireland: Perspectives of car-dealers and policy-makers". In: *Case Studies on Transport Policy* 7.1, pp. 118–127.

Onggo, Bhakti Stephan, Javier Panadero, Canan G Corlu, and Angel A Juan (2019). "Agri-food supply chains with stochastic demands: A multi-period inventory routing problem with perishable products". In: *Simulation Modelling Practice and Theory* 97, p. 101970.

Oppen, Johan, Arne Løkketangen, and Jacques Desrosiers (2010). "Solving a rich vehicle routing and inventory problem using column generation". In: *Computers & Operations Research* 37.7, pp. 1308–1317.

Opricovic, Serafim and Gwo-Hshiung Tzeng (2003). "Defuzzification within a multicriteria decision model". In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 11.05, pp. 635–652.

Ortiz-Astorquiza, Camilo, Ivan Contreras, and Gilbert Laporte (2018). "Multi-level facility location problems". In: *European Journal of Operational Research* 267.3, pp. 791–805.

Owen, Susan Hesse and Mark S Daskin (1998). "Strategic facility location: A review". In: *European journal of operational research* 111.3, pp. 423–447.

Pagès-Bernaus, Adela, Helena Ramalhinho, Angel A Juan, and Laura Calvet (2019). "Designing e-commerce supply chains: a stochastic facility–location approach". In: *International Transactions in Operational Research* 26.2, pp. 507–528.

Palacios-Argüello, Laura, Jesus Gonzalez-Feliu, Natacha Gondran, and Fabien Badeig (2018). "Assessing the economic and environmental impacts of urban food systems for public school canteens: case study of Great Lyon region". In: *European Transport Research Review* 10.2, pp. 1–20.

Panadero, J., J. de Armas, C. S. M. Currie, and A. A. Juan (2017). "A simheuristic approach for the stochastic team orienteering problem". In: *2017 Winter Simulation Conference (WSC)*. IEEE Press, pp. 3208–3217.

Panadero, Javier, Jana Doering, Renatas Kizys, Angel A Juan, and Angels Fito (2020a). "A variable neighborhood search simheuristic for project portfolio selection under uncertainty". In: *Journal of Heuristics* 26.3, pp. 353–375.

Panadero, Javier, Angel A Juan, Christopher Bayliss, and Christine Currie (2020b). "Maximising reward from a team of surveillance drones: a simheuristic approach to the stochastic team orienteering problem". In: *European Journal of Industrial Engineering* 14.4, pp. 485–516.

Panwalkar, Shrikant S and Wafik Iskander (1977). "A survey of scheduling rules". In: *Operations research* 25.1, pp. 45–61.

Pariazar, Mahmood and Mustafa Y Sir (2018). "A multi-objective approach for supply chain design considering disruptions impacting supply availability and quality". In: *Computers & Industrial Engineering* 121, pp. 113–130.

Parragh, Sophie N, Karl F Doerner, and Richard F Hartl (2008). "A survey on pickup and delivery problems". In: *Journal für Betriebswirtschaft* 58.1, pp. 21–51.

Patella, Sergio Maria, Gianluca Grazieschi, Valerio Gatta, Edoardo Marcucci, and Stefano Carrese (2021). "The Adoption of Green Vehicles in Last Mile Logistics: A Systematic Review". In: *Sustainability* 13.1, p. 6.

Peck, H. (2006). "Reconciling supply chain vulnerability, risk and supply chain management". In: *International Journal of Logistics Research and Applications* 9.2, pp. 127–142.

Peng, Peng, Lawrence V Snyder, Andrew Lim, and Zuli Liu (2011). "Reliable Logistics Networks Design with Facility Disruptions". In: *Transportation Research Part B: Methodological* 45.8, pp. 1190–1211.

Penna, Puca Huachi Vaz, Anand Subramanian, Luiz Satoru Ochi, Thibaut Vidal, and Christian Prins (2019). "A hybrid heuristic for a broad class of vehicle routing problems with heterogeneous fleet". In: *Annals of Operations Research* 273.1, pp. 5–74.

Perl, J. and M.S. Daskin (1985). "A warehouse location-routing problem". In: *Transportation Research Part B* 19.5, pp. 381–396.

Pisinger, David and Stefan Ropke (2007). "A general heuristic for vehicle routing problems". In: *Computers & operations research* 34.8, pp. 2403–2435.

Ponomarov, Serhiy Y and Mary C Holcomb (2009). "Understanding the concept of supply chain resilience". In: *The international journal of logistics management* 20.1, pp. 124–143.

Prins, Christian, Caroline Prodhon, and Roberto Wolfler Calvo (2006). "Solving the capacitated location-routing problem by a GRASP complemented by a learning process and a path relinking". In: *4OR* 4.3, pp. 221–238.

Prins, Christian, Caroline Prodhon, Angel Ruiz, Patrick Soriano, and Roberto Wolfler Calvo (2007). "Solving the capacitated location-routing problem by a cooperative Lagrangean relaxation-granular tabu search heuristic". In: *Transportation Science* 41.4, pp. 470–483.

Prodhon, Caroline and Christian Prins (2014). "A survey of recent research on location-routing problems". In: *European Journal of Operational Research* 238.1, pp. 1–17.

Quintero-Araujo, Carlos L, Juan Pablo Caballero-Villalobos, Angel A Juan, and Jairo R Montoya-Torres (2017). "A biased-randomized metaheuristic for the capacitated location routing problem". In: *International Transactions in Operational Research* 24.5, pp. 1079–1098.

Quintero-Araujo, Carlos L, Aljoscha Gruler, Angel A Juan, and Javier Faulin (2019a). "Using horizontal cooperation concepts in integrated routing and facility-location decisions". In: *International Transactions in Operational Research* 26.2, pp. 551–576.

Quintero-Araujo, Carlos L, Daniel Guimarans, and Angel A Juan (2019b). "A simheuristic algorithm for the capacitated location routing problem with stochastic demands". In: *Journal of Simulation* 0, pp. 1–18.

Raba, David, Alejandro Estrada-Moreno, Javier Panadero, and Angel A Juan (2020). "A reactive simheuristic using online data for a real-life inventory routing problem with stochastic demands". In: *International Transactions in Operational Research* 27.6, pp. 2785–2816.

Rabbani, Masoud, Razieh Heidari, and Reza Yazdanparast (2019). "A Stochastic Multi-Period Industrial Hazardous Waste Location-Routing Problem: integrating NSGA-II and Monte Carlo simulation". In: *European Journal of Operational Research* 272.3, pp. 945–961.

Rabe, Markus, Jorge Chicaiza-Vaca, Rafael D Tordecilla, and Angel A Juan (2020a). "A simulation-optimization approach for locating automated parcel lockers in urban logistics operations". In: *2020 Winter Simulation Conference (WSC)*. IEEE, pp. 1230–1241.

Rabe, Markus, Maik Deininger, and Angel A Juan (2020b). "Speeding up computational times in simheuristics combining genetic algorithms with discrete-event simulation". In: *Simulation Modelling Practice and Theory* 103, p. 102089.

Rabe, Markus, Jesus Gonzalez-Feliu, Jorge Chicaiza-Vaca, and Rafael D Tordecilla (2021). "Simulation-Optimization Approach for Multi-Period Facility Location Problems with Forecasted and Random Demands in a Last-Mile Logistics Application". In: *Algorithms* 14.2, p. 41.

Rahbari, Ali, Mohammad Mahdi Nasiri, Frank Werner, MirMohammad Musavi, and Fariborz Jolai (2019). "The vehicle routing and scheduling problem with cross-docking for perishable products under uncertainty: Two robust bi-objective models". In: *Applied Mathematical Modelling* 70, pp. 605–625.

Rajagopal, Varthini, Shanmugam Prasanna Venkatesan, and Mark Goh (2017). "Decision-making models for supply chain risk mitigation: A review". In: *Computers & Industrial Engineering* 113, pp. 646–682.

Ramezankhani, MJ, S Ali Torabi, and Farhad Vahidi (2018). "Supply chain performance measurement and evaluation: A mixed sustainability and resilience approach". In: *Computers & Industrial Engineering* 126, pp. 531–548.

Reed, Martin, Aliki Yiannakou, and Roxanne Evering (2014). "An ant colony algorithm for the multi-compartment vehicle routing problem". In: *Applied Soft Computing* 15, pp. 169–176.

Resende, Mauricio GC and Celso C Ribeiro (2010). "Greedy randomized adaptive search procedures: Advances, hybridizations, and applications". In: *Handbook of metaheuristics*. Springer, pp. 283–319.

Reyes-Rubiano, L, AA Juan, Christopher Bayliss, J Panadero, J Faulin, and P Copado (2020a). "A Biased-Randomized Learnheuristic for Solving the Team Orienteering Problem with Dynamic Rewards". In: *Transportation Research Procedia* 47, pp. 680–687.

Reyes-Rubiano, Lorena, Laura Calvet, Angel A Juan, Javier Faulin, and Lluc Bové (2020b). "A biased-randomized variable neighborhood search for sustainable multi-depot vehicle routing problems". In: *Journal of Heuristics* 26.3, pp. 401–422.

Reyes-Rubiano, Lorena, Daniele Ferone, Angel A Juan, and Javier Faulin (2019). "A simheuristic for routing electric vehicles with limited driving ranges and stochastic travel times". In: *SORT-Statistics and Operations Research Transactions* 1.1, pp. 3–24.

Rezaei, Neda, Sadoullah Ebrahimnejad, Amirhossein Moosavi, and Adel Nikfarjam (2019). "A green vehicle routing problem with time windows considering the heterogeneous fleet of vehicles: two metaheuristic algorithms". In: *European Journal of Industrial Engineering* 13.4, pp. 507–535.

Riemann, Raffaela, David ZW Wang, and Fritz Busch (2015). "Optimal location of wireless charging facilities for electric vehicles: flow-capturing location model with stochastic user equilibrium". In: *Transportation Research Part C: Emerging Technologies* 58, pp. 1–12.

Ritzinger, Ulrike, Jakob Puchinger, and Richard F Hartl (2016). "A survey on dynamic and stochastic vehicle routing problems". In: *International Journal of Production Research* 54.1, pp. 215–231.

Rodríguez, Sara V, Lluis M Plà, and Javier Faulin (2014). "New opportunities in operations research to improve pork supply chain efficiency". In: *Annals of Operations Research* 219.1, pp. 5–23.

Ruggieri, Roberto, Marco Ruggeri, Giuliana Vinci, and Stefano Poponi (2021). "Electric Mobility in a Smart City: European Overview". In: *Energies* 14.2, p. 315.

Ruiz, Efrain, Valeria Soto-Mendoza, Alvaro Ernesto Ruiz Barbosa, and Ricardo Reyes (2019). "Solving the open vehicle routing problem with capacity and distance constraints with a biased random key genetic algorithm". In: *Computers & Industrial Engineering* 133, pp. 207–219.

Saberian, Mohammad, Jie Li, Shannon Kilmartin-Lynch, and Mahdi Boroujeni (2021). "Repurposing of COVID-19 single-use face masks for pavements base/subbase". In: *Science of the Total Environment* 769, p. 145527.

Saeedvand, Saeed, Hadi S Aghdasi, and Jacky Baltes (2020). "Novel hybrid algorithm for Team Orienteering Problem with Time Windows for rescue applications". In: *Applied Soft Computing* 96, p. 106700.

Saif, Ahmed and Samir Elhedhli (2016). "Cold supply chain design with environmental considerations: A simulation-optimization approach". In: *European Journal of Operational Research* 251.1, pp. 274–287.

Salehi, Faraz, Masoud Mahootchi, and Seyed Mohammad Moattar Husseini (2019). "Developing a robust stochastic model for designing a blood supply chain network in a crisis: A possible earthquake in Tehran". In: *Annals of Operations Research* 283.1-2, pp. 679–703.

Salem, Roba W and Mohamed Haouari (2017). "A simulation-optimisation approach for supply chain network design under supply and demand uncertainties". In: *International Journal of Production Research* 55.7, pp. 1845–1861.

Salhi, Said and Graham K Rand (1989). "The effect of ignoring routes when locating depots". In: *European journal of operational research* 39.2, pp. 150–156.

Samanlioglu, Funda (2013). "A multi-objective mathematical model for the industrial hazardous waste location-routing problem". In: *European Journal of Operational Research* 226.2, pp. 332–340.

Sánchez-Oro, Jesús, Ana D López-Sánchez, and J Manuel Colmenar (2020). "A general variable neighborhood search for solving the multi-objective open vehicle routing problem". In: *Journal of Heuristics* 26.3, pp. 423–452.

Schneider, Johannes and Scott Kirkpatrick (2007). *Stochastic Optimization*. Springer Science & Business Media.

Schneider, Michael and Michael Drexl (2017). "A survey of the standard location-routing problem". In: *Annals of Operations Research* 259.1-2, pp. 389–414.

Schrage, Linus (1981). "Formulation and structure of more complex/realistic routing and scheduling problems". In: *Networks* 11.2, pp. 229–232.

Sharma, Satyendra Kumar, Srikanta Routroy, and Utkarsh Yadav (2018). "Vehicle routing problem: recent literature review of its variants". In: *International Journal of Operational Research* 33.1, pp. 1–31.

Shavarani, Seyed Mahdi, Mazyar Ghadiri Nejad, Farhood Rismanchian, and Gokhan Izbirak (2018). "Application of hierarchical facility location problem for optimization of a drone delivery system: a case study of Amazon prime air in the city of San Francisco". In: *The International Journal of Advanced Manufacturing Technology* 95.9, pp. 3141–3153.

Shen, Zuo-Jun Max, Roger Lezhou Zhan, and Jiawei Zhang (2011). "The reliable facility location problem: Formulations, heuristics, and approximation algorithms". In: *INFORMS Journal on Computing* 23.3, pp. 470–482.

Shi, Yong, Toufik Boudouh, and Olivier Grunder (2017). "A hybrid genetic algorithm for a home health care routing problem with time window and fuzzy demand". In: *Expert Systems with Applications* 72, pp. 160–176.

Silvestrin, Paulo Vitor and Marcus Ritt (2017). "An iterated tabu search for the multi-compartment vehicle routing problem". In: *Computers & Operations Research* 81, pp. 192–202.

Simangunsong, Eliot, Linda C Hendry, and Mark Stevenson (2012). "Supply-chain uncertainty: a review and theoretical foundation for future research". In: *International Journal of Production Research* 50.16, pp. 4493–4523.

Simeonova, Lina, Niaz Wassan, Naveed Wassan, and Said Salhi (2020). "Recent Developments in Real Life Vehicle Routing Problem Applications". In: *Green Transportation and New Advances in Vehicle Routing Problems*. Springer, pp. 213–228.

Snyder, Lawrence V (2006). "Facility location under uncertainty: a review". In: *IIE transactions* 38.7, pp. 547–564.

Snyder, Lawrence V, Zümbül Atan, Peng Peng, Ying Rong, Amanda J Schmitt, and Burcu Sinsoysal (2016). "OR/MS models for supply chain disruptions: A review". In: *Iie Transactions* 48.2, pp. 89–109.

Snyder, Lawrence V and Mark S Daskin (2005). "Reliability models for facility location: the expected failure cost case". In: *Transportation Science* 39.3, pp. 400–416.

Soysal, Mehmet (2016). "Closed-loop Inventory Routing Problem for returnable transport items". In: *Transportation Research Part D: Transport and Environment* 48, pp. 31–45.

Stefanovic, Dusan, Nenad Stefanovic, and Bozidar Radenkovic (2009). "Supply network modelling and simulation methodology". In: *Simulation Modelling Practice and Theory* 17.4, pp. 743–766.

Stewart Jr, William R and Bruce L Golden (1983). "Stochastic vehicle routing: A comprehensive approach". In: *European Journal of Operational Research* 14.4, pp. 371–385.

Stiglic, Mitja, Niels Agatz, Martin Savelsbergh, and Mirko Gradisar (2015). "The benefits of meeting points in ride-sharing systems". In: *Transportation Research Part B: Methodological* 82, pp. 36–53.

Sun, Yan (2020). "A Fuzzy Multi-Objective Routing Model for Managing Hazardous Materials Door-to-Door Transportation in the Road-Rail Multimodal Network With Uncertain Demand and Improved Service Level". In: *IEEE Access* 8, pp. 172808–172828.

Sun, Yan, Martin Hrušovskỳ, Chen Zhang, and Maoxiang Lang (2018). "A time-dependent fuzzy programming approach for the green multimodal routing problem with rail service capacity uncertainty and road traffic congestion". In: *Complexity* 2018.

Sun, Zhuo, Ni Yan, Yining Sun, and Haobin Li (2019). "Location-Routing Optimization with Split Demand for Customer Self-Pickup via Data Analysis and Heuristics Search". In: *Asia-Pacific Journal of Operational Research* 36.06, p. 1940013.

Talbi, El-Ghazali (2009). *Metaheuristics: from design to implementation*. Vol. 74. John Wiley & Sons.

Tang, Chaogang, Shixiong Xia, Chunsheng Zhu, and Xianglin Wei (2019). "Phase timing optimization for smart traffic control based on fog computing". In: *IEEE Access* 7, pp. 84217–84228.

Taş, Duygu, Nico Dellaert, Tom Van Woensel, and Ton De Kok (2013). "Vehicle routing problem with stochastic travel times including soft time windows and service costs". In: *Computers & Operations Research* 40.1, pp. 214–224.

Tavakkoli-Moghaddam, Reza, Mohammadreza Meskini, Hadi Nasseri, and Haed Tavakkoli-Moghaddam (2019). "A Multi-Depot Close and Open Vehicle Routing Problem with Heterogeneous Vehicles". In: *2019 International Conference on Industrial Engineering and Systems Management (IESM)*. IEEE, pp. 1–6.

Tekin, Eylem and Ihsan Sabuncuoglu (2004). "Simulation optimization: A comprehensive review on theory and applications". In: *IIE transactions* 36.11, pp. 1067–1081.

Teodorović, Dušan and Goran Pavković (1996). "The fuzzy set theory approach to the vehicle routing problem when demand at nodes is uncertain". In: *Fuzzy Sets and Systems* 82.3, pp. 307–317.

Ting, Chuan-Kang, Xin-Lan Liao, Yu-Hsuan Huang, and Rung-Tzuo Liaw (2017). "Multi-vehicle selective pickup and delivery using metaheuristic algorithms". In: *Information Sciences* 406, pp. 146–169.

Tordecilla-Madera, Rafael, Andrés Polo, Dairo Muñoz, and Leonardo González-Rodríguez (2017). "A robust design for a Colombian dairy cooperative's milk storage and refrigeration logistics system using binary programming". In: *International Journal of Production Economics* 183, pp. 710–720.

Tordecilla-Madera, Rafael, Andrés Polo Roa, John Willmer Escobar, and Nicolas Clavijo Buriticá (2018). "A mathematical model for collecting and distributing perishable products by considering costs minimisation and CO_2 emissions". In: *International Journal of Services and Operations Management* 31.2, pp. 207–234.

Toth, Paolo and Daniele Vigo (1997). "An exact algorithm for the vehicle routing problem with backhauls". In: *Transportation science* 31.4, pp. 372–385.

— (2014). *Vehicle routing: problems, methods, and applications*. SIAM.

Tricoire, Fabien, Martin Romauch, Karl F Doerner, and Richard F Hartl (2010). "Heuristics for the Multi-Period Orienteering Problem with Multiple Time Windows". In: *Computers & Operations Research* 37.2, pp. 351–367.

Tunalıoğlu, Renan, Çağrı Koç, and Tolga Bektaş (2016). "A multiperiod location-routing problem arising in the collection of Olive Oil Mill Wastewater". In: *Journal of the Operational Research Society* 67.7, pp. 1012–1024.

Tuzun, Dilek and Laura I Burke (1999). "A two-phase tabu search approach to the location routing problem". In: *European Journal of Operational Research* 116.1, pp. 87–99.

Ukkusuri, Satish V and Wilfredo F Yushimito (2008). "Location Routing Approach for the Humanitarian Prepositioning Problem". In: *Transportation Research Record* 2089.1, pp. 18–25.

Vakulenko, Yulia, Daniel Hellström, and Klas Hjort (2018). "What's in the parcel locker? Exploring customer value in e-commerce last mile delivery". In: *journal of Business Research* 88, pp. 421–427.

Vansteenwegen, Pieter, Wouter Souffriau, Greet Vanden Berghe, and Dirk Van Oudheusden (2009). "Iterated Local Search for the Team Orienteering Problem with Time Windows". In: *Computers & Operations Research* 36.12, pp. 3281–3290.

Verlinde, Sara, César Rojas, Heleen Buldeo Rai, Bram Kin, and Cathy Macharis (2018). "E-Consumers and Their Perception of Automated Parcel Stations". In: *City Logistics 3: Towards Sustainable and Liveable Cities*, pp. 147–160.

Verma, Madhushi and Kaushal K Shukla (2015). "Application of fuzzy optimization to the orienteering problem". In: *Advances in Fuzzy Systems* 2015.

Vidal, Thibaut, Gilbert Laporte, and Piotr Matl (2019). "A concise guide to existing and emerging vehicle routing problem variants". In: *European Journal of Operational Research*.

Vidović, Milorad, Dražen Popović, and Branislava Ratković (2014). "Mixed integer and heuristics model for the inventory routing problem in fuel delivery". In: *International Journal of Production Economics* 147, pp. 593–604.

Villarinho, Pedro A, Javier Panadero, Luciana S Pessoa, Angel A Juan, and Fernando L Cyrino Oliveira (2021). "A simheuristic algorithm for the stochastic permutation flow-shop problem with delivery dates and cumulative payoffs". In: *International Transactions in Operational Research* 28.2, pp. 716–737.

Vlajic, Jelena V, Jack GAJ Van der Vorst, and René Haijema (2012). "A framework for designing robust food supply chains". In: *International Journal of Production Economics* 137.1, pp. 176–189.

Wang, Qian, Qingkai Ji, and Chun-Hung Chiu (2014). "Optimal routing for heterogeneous fixed fleets of multicompartment vehicles". In: *Mathematical Problems in Engineering* 2014.

Wassan, N (2007). "Reactive tabu adaptive memory programming search for the vehicle routing problem with backhauls". In: *Journal of the Operational Research Society* 58.12, pp. 1630–1641.

Wu, Tai-Hsi, Chinyao Low, and Jiunn-Wei Bai (2002). "Heuristic solutions to multi-depot location-routing problems". In: *Computers & Operations Research* 29.10, pp. 1393–1415.

Xu, Jie, Edward Huang, Chun-Hung Chen, and Loo Hay Lee (2015). "Simulation optimization: A review and exploration in the new era of cloud computing and big data". In: *Asia-Pacific Journal of Operational Research* 32.03, p. 1550019.

Xu, Wenzheng, Zichuan Xu, Jian Peng, Weifa Liang, Tang Liu, Xiaohua Jia, and Sajal K Das (2020). "Approximation algorithms for the team orienteering problem". In: *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, pp. 1389–1398.

Yazdani, Maziar, Kamyar Kabirifar, Boadu Elijah Frimpong, Mahdi Shariati, Mirpouya Mirmozaffari, and Azam Boskabadi (2021). "Improving construction and demolition waste collection service in an urban area using a simheuristic approach: A case study in Sydney, Australia". In: *Journal of Cleaner Production* 280, p. 124138.

Yazdani, Maziar, Mohammad Mojtahedi, and Martin Loosemore (2020). "Enhancing evacuation response to extreme weather disasters using public transportation systems: a novel simheuristic approach". In: *Journal of Computational Design and Engineering* 7.2, pp. 195–210.

Yoo, Taejong, Hyunbo Cho, and Enver Yücesan (2010). "Hybrid algorithm for discrete event simulation based supply chain optimization". In: *Expert Systems with Applications* 37.3, pp. 2354–2361.

Yoon, Moonyoung and James Bekker (2020). "Multi-objective simulation optimisation on discrete sets: a literature review". In: *International Journal of Operational Research* 39.3, pp. 364–405.

Yousefikhoshbakht, Majid and Azam Dolatnejad (2017). "A column generation for the heterogeneous fixed fleet open vehicle routing problem". In: *International Journal of Production Management and Engineering* 5.2, pp. 55–71.

Yu, Vincent F, Shih-Wei Lin, Wenyih Lee, and Ching-Jung Ting (2010). "A simulated annealing heuristic for the capacitated location routing problem". In: *Computers & Industrial Engineering* 58.2, pp. 288–299.

Yu, Yang, Sihan Wang, Junwei Wang, and Min Huang (2019). "A branch-and-price algorithm for the heterogeneous fleet green vehicle routing problem with time windows". In: *Transportation Research Part B: Methodological* 122, pp. 511–527.

Zachariadis, Emmanouil E and Chris T Kiranoudis (2012). "An effective local search approach for the vehicle routing problem with backhauls". In: *Expert Systems with Applications* 39.3, pp. 3174–3184.

Zarandi, Mohammad Hossein Fazel, Ahmad Hemmati, and Soheil Davari (2011). "The multi-depot capacitated location-routing problem with fuzzy travel times". In: *Expert Systems with Applications* 38.8, pp. 10075–10084.

Zarandi, Mohammad Hossein Fazel, Ahmad Hemmati, Soheil Davari, and I Burhan Turksen (2013). "Capacitated location-routing problem with time windows under uncertainty". In: *Knowledge-Based Systems* 37, pp. 480–489.

Zhang, Bo, Hui Li, Shengguo Li, and Jin Peng (2018). "Sustainable Multi-Depot Emergency Facilities Location-Routing Problem with Uncertain Information". In: *Applied Mathematics and Computation* 333, pp. 506–520.

Zhang, Haoran, Yongtu Liang, Qi Liao, Jinyu Chen, Wan Zhang, Yin Long, and Chen Qian (2019a). "Optimal design and operation for supply chain system of multi-state natural gas under uncertainties of demand and purchase price". In: *Computers & Industrial Engineering* 131, pp. 115–130.

Zhang, Huizhen, Fan Liu, Liang Ma, and Ziying Zhang (2020a). "A hybrid heuristic based on a particle swarm algorithm to solve the capacitated location-routing problem with fuzzy demands". In: *IEEE Access* 8, pp. 153671–153691.

Zhang, Shuai, Mingzhou Chen, and Wenyu Zhang (2019b). "A Novel Location-Routing Problem in Electric Vehicle Transportation with Stochastic Demands". In: *Journal of Cleaner Production* 221, pp. 567–581.

Zhang, Shuai, Mingzhou Chen, Wenyu Zhang, and Xiaoyu Zhuang (2020b). "Fuzzy optimization model for electric vehicle routing problem with time windows and recharging stations". In: *Expert systems with applications* 145, p. 113123.

Zheng, Yongshuang and Baoding Liu (2006). "Fuzzy vehicle routing model with credibility measure and its hybrid intelligent algorithm". In: *Applied Mathematics and Computation* 176.2, pp. 673–683.

Zhong, Shaopeng, Rong Cheng, Yu Jiang, Zhong Wang, Allan Larsen, and Otto Anker Nielsen (2020). "Risk-averse optimization of disaster relief facility location and vehicle routing under stochastic demand". In: *Transportation Research Part E: Logistics and Transportation Review* 141, p. 102015.

Zhou, Lin, Roberto Baldacci, Daniele Vigo, and Xu Wang (2018). "A multi-depot two-echelon vehicle routing problem with delivery options arising in the last mile distribution". In: *European Journal of Operational Research* 265.2, pp. 765–778.

Zhou, Lin, Yun Lin, Xu Wang, and Fuli Zhou (2019). "Model and algorithm for bilevel multisized terminal location-routing problem for the last mile delivery". In: *International Transactions in Operational Research* 26.1, pp. 131–156.

Appendices

A.1 Complementary Information

A.1.1 MILP Model of the Vehicle Routing Problem with Optional Backhauls

This appendix shows a mathematical model for the problem described in Section 4.2.1. Related sets, parameters, and variables are displayed in Table 1.

Table 1: Sets, parameters, and variables for our VRPOB model.

Sets
V = Set of nodes
L = Set of LH customers
B = Set of BH customers
A = Set of edges, $A \subseteq V \times V = \{(i,j) : i,j \in V, i < j\}$
K = Set of vehicles
Parameters
d_i = Demand of customer $i \in V$
c_{ij} = Cost of travel from node $i \in V$ to node $j \in V$
h_i = Unitary penalty cost (e.g., unitary inventory cost) per RTI not collected from customer $i \in B$
q = Capacity of each vehicle
Variables
x_{ijk} = Binary variable equal to 1 if edge $(i,j) : i \in L \cup \{0\}, j \in L$ is in the LH route traveled by vehicle $k \in K$, 0 otherwise
y_{ijk} = Binary variable equal to 1 if edge $(i,j) : i \in L \cup B, j \in B \cup \{0\}$ is in the BH route traveled by vehicle $k \in K$, 0 otherwise
z_{ik} = Binary variable equal to 1 if customer $i \in V \setminus \{0\}$ is visited by vehicle $k \in K$, 0 otherwise
f_{ijk} = Variable to eliminate subtours in the LH route visited by vehicle $k \in K$ and for each edge $(i,j) : i \in L \cup \{0\}, j \in L$
g_{ijk} = Variable to eliminate subtours in the BH route visited by vehicle $k \in K$ and for each edge $(i,j) : i \in L \cup B, j \in B \cup \{0\}$

$$Minimize \ \sum_{i \in B} h_i d_i \left(1 - \sum_{k \in K} z_{ik}\right) + \sum_{k \in K} \sum_{i \in L \cup \{0\}} \sum_{j \in L, i \neq j} c_{ij} x_{ijk} + \sum_{k \in K} \sum_{i \in L \cup B} \sum_{j \in B \cup \{0\}, i \neq j} c_{ij} y_{ijk} \quad (1)$$

s.t.

$$\sum_{k \in K} z_{ik} = 1, \ \forall i \in L \quad (2)$$

$$\sum_{k \in K} z_{ik} \leq 1, \ \forall i \in B \quad (3)$$

$$\sum_{j \in L} x_{0jk} = 1, \ \forall k \in K \quad (4)$$

$$\sum_{i \in V} y_{i0k} = 1, \ \forall k \in K \quad (5)$$

$$\sum_{i \in L \cup \{0\}, i \neq h} x_{ihk} + \sum_{j \in L, j \neq h} x_{hjk} + \sum_{j \in B \cup \{0\}, j \neq h} y_{hjk} = 2 z_{hk}, \ \forall h \in L, \forall k \in K \quad (6)$$

$$\sum_{i \in L \cup B, i \neq h} y_{ihk} + \sum_{j \in B \cup \{0\}, j \neq h} y_{hjk} = 2z_{hk}, \ \forall h \in B, \forall k \in K \tag{7}$$

$$\sum_{i \in L \cup \{0\}, i \neq h} x_{ihk} = \sum_{j \in L, j \neq h} x_{hjk} + \sum_{j \in B \cup \{0\}, j \neq h} y_{hjk} \ \forall h \in L, \forall k \in K \tag{8}$$

$$\sum_{i \in L \cup B, i \neq h} y_{ihk} = \sum_{j \in B \cup \{0\}, j \neq h} y_{hjk}, \ \forall h \in B, \forall k \in K \tag{9}$$

$$\sum_{j \in L, j \neq h} x_{hjk} + \sum_{j \in B \cup \{0\}, j \neq h} y_{hjk} \leq 1, \ \forall h \in L, \forall k \in K \tag{10}$$

$$\sum_{i \in B \cup \{0\}, i \neq h} y_{ihk} \leq 1, \ \forall h \in B, \forall k \in K \tag{11}$$

$$\sum_{i \in L} d_i z_{ik} \leq q, \ \forall k \in K \tag{12}$$

$$\sum_{i \in B} d_i z_{ik} \leq q, \ \forall k \in K \tag{13}$$

$$f_{ijk} \leq n x_{ijk}, \ \forall i \in L \cup \{0\}, \forall j \in L, i \neq j, \forall k \in K \tag{14}$$

$$\sum_{i \in L \cup \{0\}, i \neq j} f_{ijk} - \sum_{h \in L, h \neq j} f_{jhk} = z_{jk}, \ \forall j \in L, \forall k \in K \tag{15}$$

$$\sum_{j \in L} f_{0jk} = \sum_{j \in L} z_{jk}, \ \forall k \in K \tag{16}$$

$$g_{ijk} \leq m y_{ijk}, \ \forall i \in L \cup B, \forall j \in B \cup \{0\}, i \neq j, \forall k \in K \tag{17}$$

$$\sum_{h \in B \cup \{0\}, h \neq j} g_{jhk} - \sum_{i \in L \cup B, i \neq j} g_{ijk} = z_{jk}, \ \forall j \in B, \forall k \in K \tag{18}$$

$$\sum_{j \in B} g_{j0k} = \sum_{j \in B} z_{jk}, \ \forall k \in K \tag{19}$$

$$x_{ijk}, y_{ijk}, z_{ik} \in \{0, 1\}, \ \forall i, j \in V, k \in K \tag{20}$$

$$f_{ijk}, g_{ijk} \in \mathbb{Z}^+, \ \forall i, j \in V, k \in K \tag{21}$$

In this model, Equation (1) minimizes the total routing plus penalty cost. Constraints (2) ensure that every LH customer is visited by only one vehicle. Constraints (3) make certain that at most only one vehicle visits a BH customer. Constraints (4) and (5) warrant that each vehicle leaves and returns to the depot. Constraints (6) and (7) certify that two edges (one entering and one leaving) are assigned to a customer only if this is serviced. Constraints

(8) and (9) establish that if one vehicle enters a customer node, it must departure from it as well. Constraints (10) assure that after departing from each LH customer, a vehicle must either serve another LH customer or make an empty trip either to the depot or to a BH customer. Constraints (11) make sure that before a vehicle visits a BH customer, it must have serviced either another BH customer or a LH customer. Constraints (12) and (13) ensure that the total quantity of product to deliver or pickup does not exceed a single vehicle capacity, both in LH and BH groups, respectively. Based on the formulation proposed by Gavish and Graves (1978), constraints (14), (15), and (16) eliminate subtours for LH customers. Likewise, constraints (17), (18), and (19) do the same regarding subtours for BH customers. Other formulations for subtour elimination can be found in Öncan et al. (2009). Finally, constraints (20) and (21) indicate the variables that are binary and integer, respectively.

A.1.2 Alternative MILP Models for the LRP with Facility Sizing Decisions

Section 4.3.1 shows a model that yields optimal solutions relatively quickly for our small newly-created instances. Nevertheless, alternative formulations can be made for our addressed problem. This appendix shows a comparison between three MIP models. The first model is the one shown in Section 4.3.1 (called the *3-index model* henceforth). The second model is a modification of the first one, in which available sizes are not considered as an independent set (*2-index model*). Instead, we consider multiple copies of each facility, and each copy has a different capacity. Finally, the third model is an adaptation of a set-partitioning model (Baldacci et al., 2011), where a set of alternative sizes is included (*SP model*). Tables 2 and 3 show the sets, parameters, and variables of the 2-index model and the SP model, respectively. The 3-index and 2-index models are very similar, however, the 2-index model (Equations (22)-(25)) requires a set of dummy depots. For example, if an instance has 3 potential depot locations and there are 5 alternative sizes for each open depot, the set J has 15 dummy depots.

$$Minimize \quad \sum_{j \in J} (f_j + o_j) y_j + \sum_{a \in A} \sum_{k \in K} c_a w_{ak} + \sum_{a \in \delta^+(J)} \sum_{k \in K} v w_{ak} \tag{22}$$

s.t.

$$Constraints\ (4.6) - (4.12)$$

$$\sum_{i \in I} d_i x_{ij} \leq s_j y_j, \quad \forall j \in J \tag{23}$$

$$\sum_{j \in J_p} y_j \leq 1, \quad \forall p \in P \tag{24}$$

$$\forall\ y_j, x_{ij}, w_{ak} \in \{0, 1\} \tag{25}$$

Table 2: Sets, parameters, and variables of a 2-index model for the LRP with facility sizing decisions.

Sets
V = Set of nodes
K = Set of vehicles
I = Set of customers, $I \subset V$
J = Set of dummy depots, $J \subset V$
P = Set of potential depot locations, $P \subset V$
J_p = Set of dummy depots in each location $p \in P$, $J_p \subset J$
A = Set of arcs, $A = V \times V = \{(m, n) : m \in V, n \in V \wedge m \neq n\}$
$\delta^+(S)$ = Set of arcs leaving S, $S \subset V$, $\delta^+(S) \subset A$
$\delta^-(S)$ = Set of arcs entering S, $S \subset V$, $\delta^-(S) \subset A$
Parameters
s_j = Available size of depot $j \in J$
d_i = Demand of customer $i \in I$
f_j = Fixed opening cost of depot $j \in J$
o_j = Variable opening cost of depot $j \in J$
c_a = Cost of traversing arc $a \in A$
v = Fixed cost for using a vehicle
q = Capacity of each vehicle
M = A very large number when compared to the magnitude of the rest of the parameters
Variables
y_j = Binary variable equal to 1 if depot $j \in J$ is open, 0 otherwise
x_{ij} = Binary variable equal to 1 if customer $i \in I$ is assigned to depot $j \in J$, 0 otherwise
w_{ak} = Binary variable equal to 1 if arc $a \in A$ is used in the route performed by vehicle $k \in K$, 0 otherwise
u_{ik} = Accumulated deliveries by vehicle $k \in K$ until customer $i \in I$

The objective function (22) minimizes the total cost, formed by the depot fixed and variable opening costs, the routing costs, and the vehicles fixed costs. Constraints (23) guarantee that the total demand of the customers assigned to an open depot does not exceed its capacity. Constraints (24) ensure that at most one depot is open in each location. Finally, Constraints (25) indicate the variables that are binary.

The SP model (Equations (26)-(30)) requires as an input a set of all feasible routes in the problem, i.e., these routes must be constructed before each instance is run in the optimization software. Additionally, each route has both a cost (or distance) and a demand, formed by the addition of all customers' demands in that route. These routes must be feasible, i.e., the vehicle capacity is used to construct them. After this procedure finishes, the vehicle capacity is not used further.

$$Minimize \sum_{j \in J} \sum_{l \in L} \left(f_j + o_{jl}\right) y_{jl} + \sum_{j \in J} \sum_{r \in R_j} \left(c_{rj} + v\right) x_{rj} \tag{26}$$

s.t.

$$\sum_{j \in J} \sum_{r \in R_{ij}} x_{rj} = 1, \quad \forall i \in I \tag{27}$$

$$\sum_{r \in R_j} d_r x_{rj} \leq \sum_{l \in L} s_{jl} y_{jl}, \quad \forall j \in J \tag{28}$$

$$\sum_{l \in L} y_{jl} \leq 1, \quad \forall j \in J \tag{29}$$

Table 3: Sets, parameters, and variables of a set-partitioning model for the LRP with facility sizing decisions.

Sets
V = Set of nodes
R = Set of feasible routes
L = Set of available sizes
I = Set of customers, $I \subset V$
J = Set of depots, $J \subset V$
R_j = Set of feasible routes passing through the depot $j \in J$, $R_j \subset R$
R_{ij} = Set of feasible routes of depot $j \in J$ passing through the customer $i \in I$, $R_{ij} \subset R$
Parameters
s_{jl} = Available size of type $l \in L$ for the depot $j \in J$
d_r = Demand of route $r \in R$
f_j = Fixed opening cost of depot $j \in J$
o_{jl} = Variable opening cost of depot $j \in J$ with size of type $l \in L$
c_{rj} = Cost of route $r \in R_j$ of depot $j \in J$
v = Fixed cost for using a vehicle
Variables
y_{jl} = Binary variable equal to 1 if depot $j \in J$ is open with size of type $l \in L$, 0 otherwise
x_{rj} = Binary variable equal to 1 if route $r \in R$ of depot $j \in J$ is included in the solution, 0 otherwise

$$\forall\, y_{jl}, x_{rj} \in \{0, 1\} \tag{30}$$

The objective function (26) minimizes the total cost, formed by the depot fixed and variable opening costs, the distance-based costs of the routes, and the vehicles fixed costs. Constraints (27) guarantee that each customer is served by only one route. Constraints (28) ensure that the total demand of the routes assigned to each open depot does not exceed its assigned capacity. Constraints (29) guarantee that at most one depot is open in each potential location. Finally, Constraints (30) indicate that all variables are binary. All experiments in this appendix were run in a PC with an Intel Core i7 processor with 16 GB RAM, and using Windows 10 as operating system.

Table 4 displays the obtained results for our newly-created instances introduced in Section 4.3.3.1. Regardless of the MIP model, the optimal solution has always been found. Nevertheless, both the total necessary time to find these solutions and the number of single variables and equations are noticeably different for the three models. The 2-index model shows a higher number of variables and equations than the 3-index model. Despite the 2-index has one index less than the 3-index model, in the former the number of elements of the set J is multiplied by 5, which affects the size of the entire model. Additionally, the SP model shows both a significantly smaller number of variables and a greater number of equations than the other two models. The total time is formed by 3 terms:

1. *Instance generation time (IGT):* it is the time required to generate a file readable by the optimization software (e.g., GAMS). Since the SP model requires a list of all feasible routes (sets R, R_j and R_{ij}) as an input, as well as the parameters d_r and c_{rj}, the number of single input parameters can be really large. Hence, an application in Python was programmed to generate this instance file. Conversely, the 3-index and 2-index models do not require an automatic instance generation procedure, since the number of single input parameters is significantly smaller than the inputs for the SP model.

2. *Model generation time (MGT):* it is the time employed by GAMS to read and check the syntax of the input code, as well as the time spent to generate the model before it can be solved.

3. *Solving time (ST):* it is the time employed by GAMS to find the optimal solution after the model has been generated.

Table 4: Comparison of our MIP models using newly-created instances.

Instance	Optimal solution	Single equations	Single variables	Discrete variables	IGT (s)	MGT (s)	ST (s)	Total time (s)
3-index model								
tor08x2a	751.23	273	315	290	-	0.20	0.68	0.88
tor08x2b	747.05	273	315	290	-	0.13	1.41	1.54
tor08x2c	664.56	273	315	290	-	0.15	13.52	13.67
tor08x2d	606.30	273	315	290	-	0.12	2.84	2.96
tor08x2e	815.57	273	315	290	-	0.12	3.58	3.70
tor10x3a	878.93	432	526	495	-	0.15	83.45	83.60
tor10x3b	652.50	432	526	495	-	0.14	188.02	188.16
tor10x3c	948.99	432	526	495	-	0.18	1027.90	1028.08
tor10x3d	742.36	432	526	495	-	0.16	19.54	19.70
tor10x3e	788.30	432	526	495	-	0.14	31.11	31.25
Average		353	421	393	-	0.15	137.21	137.35
2-index model								
tor08x2a	751.23	497	763	738	-	0.13	4.51	4.64
tor08x2b	747.05	497	763	738	-	0.16	21.05	21.21
tor08x2c	664.56	497	763	738	-	0.14	43.73	43.87
tor08x2d	606.30	497	763	738	-	0.14	21.90	22.04
tor08x2e	815.57	497	763	738	-	0.15	11.42	11.57
tor10x3a	878.93	840	1366	1335	-	0.14	373.01	373.15
tor10x3b	652.50	840	1366	1335	-	0.17	2663.10	2663.27
tor10x3c	948.99	840	1366	1335	-	0.18	4246.76	4246.94
tor10x3d	742.36	840	1366	1335	-	0.17	75.89	76.06
tor10x3e	788.30	840	1366	1335	-	0.17	355.92	356.09
Average		669	1065	1037	-	0.15	781.73	781.88
SP model								
tor08x2a	751.23	13	2939	2938	0.11	0.22	0.13	0.46
tor08x2b	747.05	13	26097	26096	0.66	1.97	0.12	2.75
tor08x2c	664.56	13	5171	5170	0.16	0.23	0.09	0.48
tor08x2d	606.30	13	2219	2218	0.12	0.15	0.07	0.34
tor08x2e	815.57	13	4259	4258	0.16	0.24	0.09	0.49
tor10x3a	878.93	17	265843	265842	16.65	218.28	0.83	235.76
tor10x3b	652.50	17	441673	441672	26.80	790.59	1.35	818.74
tor10x3c	948.99	17	86941	86940	3.39	18.73	0.29	22.41
tor10x3d	742.36	17	152089	152088	4.91	45.37	0.48	50.76
tor10x3e	788.30	17	134626	134625	4.48	38.46	0.42	43.36
Average		15	112186	112185	5.74	111.42	0.39	117.56

The 2-index model's average total time is about 6 times longer than the 3-index model's time, which shows how inefficient the 2-index model is. Additionally, the SP model's average total time is slightly smaller than the 3-index model's, despite the addition of the IGT. Given the large size of the input file for the SP model, the average MGT is significantly greater than the average ST. Contrarily, the average MGT is significantly smaller than the average ST for the 3-index and 2-index models. That is, the latter models are really easy to read and hard to solve, and the SP model shows an opposite performance.

Table 5 displays our obtained results for 9 small benchmark instances. Concretely, we use the Barreto's and Prodhon's instances whose number of customers is smaller than 30. The second column in these tables shows the Best found solution (BFS). An asterisk indicates that the BFS is the optimal solution. Since instances have been modified to include the alternative sizes, there is no reference in the literature where the optimal solution is provided. Hence, the 3-index and 2-index models found efficiently the optimal solution for the instance *Perl-12x2*, and the SP model found it for the instances *Coord20-5-1* and *Coord20-5-2*. The rest of the BFSs are obtained employing our metaheuristic approach (Tables 4.10 and 4.11). The solving time limit was set on 10,000 seconds. Three new indicators are added to Table 5: *(i)* the MIP solution, which is the best integer solution found by CPLEX when reaching the time limit, *(ii)* the optimality gap tolerance obtained by CPLEX when reaching the time limit, and *(iii)* the gap between the MIP solution and the BFS. The smaller these gaps, the better the results. Hence, the average gaps show again a higher efficiency of the 3-index model.

Table 5: Comparison of the 3-index and 2-index models using small benchmark instances.

Instance	BFS	Single equations	Single variables	Discrete variables	MIP solution	CPLEX gap	BFS gap	MGT (s)	ST (s)	Total time (s)
3-index model										
Perl-12x2	203.98*	545	611	574	203.98	0.00%	0.00%	0.17	21.81	21.98
Coord20-5-1	51165.49*	2586	3126	3025	58661.08	43.48%	14.65%	0.33	10000.00	10000.33
Coord20-5-2	43426.36*	2586	3126	3025	45322.41	32.25%	4.37%	0.32	10000.00	10000.32
Coord20-5-1b	34998.10	1572	1926	1865	34499.32	23.58%	-1.43%	0.31	10000.00	10000.31
Coord20-5-2b	33403.25	1572	1926	1865	33912.96	20.39%	1.53%	0.34	10000.00	10000.34
Gaskell-21x5	418.65	2265	2735	2650	458.26	24.35%	9.46%	0.39	10000.00	10000.39
Gaskell-22x5	578.86	1858	2248	2181	580.45	4.98%	0.27%	0.58	10000.00	10000.58
Min-27x5	2960.02	3549	4157	4048	3475.48	37.59%	17.41%	0.43	10000.00	10000.43
Gaskell-29x5	493.35	4041	4695	4578	501.83	26.88%	1.72%	0.35	10000.00	10000.35
Average	18627.52	2286	2728	2646	19735.09	23.72%	5.33%	0.36		
2-index model										
Perl-12x2	203.98*	865	1283	1246	203.98	0.00%	0.00%	0.15	38.69	38.84
Coord20-5-1	51165.49*	4706	7526	7425	65550.80	50.27%	28.12%	0.36	10000.00	10000.36
Coord20-5-2	43426.36*	4706	7526	7425	53105.74	42.08%	22.29%	0.44	10000.00	10000.44
Coord20-5-1b	34998.10	2852	4726	4665	37535.61	31.52%	7.25%	0.40	10000.00	10000.40
Coord20-5-2b	33403.25	2852	4726	4665	33789.50	20.68%	1.16%	0.50	10000.00	10000.50
Gaskell-21x5	418.65	4045	6515	6430	483.71	29.15%	15.54%	0.37	10000.00	10000.37
Gaskell-22x5	578.86	3258	5328	5261	593.65	11.58%	2.56%	0.36	10000.00	10000.36
Min-27x5	2960.02	5809	9017	8908	4185.74	49.40%	41.41%	0.41	10000.00	10000.41
Gaskell-29x5	493.35	6461	9915	9798	804.77	55.48%	63.12%	0.67	10000.00	10000.67
Average	18627.52	3950	6285	6203	21805.94	32.24%	20.16%	0.41		

Table 6 shows the results obtained by the SP model for the same benchmark instances of Table 5. In this case, the problem size increases dramatically. In order to illustrate this statement, an upper bound (UB) for the number of feasible set partitions of customers – i.e., depots are not included, is calculated as follows:

1. Sort the customers' demands in ascending order.

2. Determine the maximum number of customers (N) that can be served in a single route. To attain this, add iteratively the demands of the first h elements until the vehicle capacity is reached, such that $\sum_{h=1}^{N} d_h \leq q$ and $\sum_{h=1}^{N+1} d_h > q$.

3. Calculate the number p of h-permutations of $|I|$, where $|I|$ is the instance's total number of customers, according to Equation (31). Notice that this expression still does not include the set partitions with one single customer.

$$p = \sum_{h=2}^{N} P(|I|, h) \tag{31}$$

4. Calculate UB according to Equation (32). Two terms are included: firstly, the set partitions with one single customer, and secondly, the division of p by 2, which is useful to decrease UB. Since all arcs in the network are assumed to be symmetric, a route traversed in one direction has the same distance-based cost and demand than the same route traversed in the opposite direction.

$$UB = |I| + \frac{p}{2} \tag{32}$$

Table 6: Results of the SP model using small benchmark instances.

Instance	Optimal solution	UB Number of feasible set partitions	Single equations	Single variables	Discrete variables	IGT (s)	MGT (s)	ST (s)	Total time (s)
Perl-12x2	203.98	2.38 x 10^{6}	17	4757295	4757294	4219.16	168769.50	19.19	173007.86
Coord20-5-1	51165.49	9.92 x 10^{5}	31	495551	495550	31.67	816.26	11.55	859.48
Coord20-5-2	43426.36	9.92 x 10^{5}	31	779411	779410	44.70	1867.81	5.25	1917.76
Coord20-5-1b	-	3.72 x 10^{12}	Out of memory to generate the instance file						
Coord20-5-2b	-	3.72 x 10^{12}							
Gaskell-21x5	-	6.98 x 10^{11}							
Gaskell-22x5	-	1.23 x 10^{20}							
Min-27x5	-	4.44 x 10^{15}							
Gaskell-29x5	-	1.00 x 10^{27}							
Average		1.11 x 10^{26}	26	2010752	2010751	1431.84	57151.19	12.00	58595.03

Calculated UBs show the reason why a computer with the aforementioned characteristics is not even able to generate the instance file. For example, the IGT for the instance *Perl-12x2* is greater than one hour, with a UB equal to 2.38 x 10^6. In turn, the instance *Gaskell-21x5* has a UB about $300,000$ times greater than the *Perl-12x2*'s, which shows the large size of that instance, as well as the size of the rest of instances whose optimal solution is not known. The MGT in Table 6 also shows how large the instance files are. For instance, GAMS took more than 46 hours only to generate the *Perl-12x2* model. Conversely, solving times are quite short in comparison.

Multiple conclusions can be drawn from the study shown in this appendix. Firstly, the 3-index model shows a better performance than the 2-index model under all considered indicators. Secondly, when considering our newly-created small instances, the SP model is 14% more time-efficient than the 3-index model in finding the optimal solution. However, the SP model efficiency is lost when increasing slightly the instance size, given the sharp rise in the size of the feasible routes set. Finally, after a solving time of $10,000$ seconds, the 3-index model did not reach the best found solution for most benchmark instances. Our metaheuristic approach obtained these BFSs in less than 15 seconds (Tables 4.10 and 4.11), which shows its high time- and cost-efficiency.

B.2 Complete Production

B.2.1 JCR-Indexed Articles

Simulation Modelling Practice and Theory 106 (2021) 102166

Contents lists available at ScienceDirect

Simulation Modelling Practice and Theory

journal homepage: www.elsevier.com/locate/simpat

Simulation-optimization methods for designing and assessing resilient supply chain networks under uncertainty scenarios: A review

Rafael D. Tordecilla[*,a,b], Angel A. Juan[a,c], Jairo R. Montoya-Torres[b], Carlos L. Quintero-Araujo[d], Javier Panadero[a,c]

[a] IN3 – Computer Science Dept., Universitat Oberta de Catalunya, Barcelona, Spain
[b] Faculty of Engineering, Universidad de La Sabana, Chia, Colombia
[c] Euncet Business School, Terrassa, Spain
[d] International School of Economics and Administrative Sciences, Universidad de La Sabana, Chia, Colombia

ARTICLE INFO

Keywords:
Resilient supply chain networks design
Simulation-optimization methods
Uncertainty scenarios
Metaheuristics

ABSTRACT

The design of supply chain networks (SCNs) aims at determining the number, location, and capacity of production facilities, as well as the allocation of markets (customers) and suppliers to one or more of these facilities. This paper reviews the existing literature on the use of simulation-optimization methods in the design of resilient SCNs. From this review, we classify some of the many works in the topic according to factors such as their methodology, the approach they use to deal with uncertainty and risk, etc. The paper also identifies several research opportunities, such as the inclusion of multiple criteria (e.g., monetary, environmental, and social dimensions) during the design-optimization process and the convenience of considering hybrid approaches combining metaheuristic algorithms, simulation, and machine learning methods to account for uncertainty and dynamic conditions, respectively.

1. Introduction

A supply chain network (SCN) is a typical example of a complex and large-scale system [11] define it as a network of suppliers, manufacturing plants, warehouses, and distribution channels organized to acquire raw materials, convert these raw materials into finished products, and distribute these products among customers. Many decisions must be made in such a complex system in order to guarantee a good performance. However, the more complex a system is, the more imprecise or inexact is the information available to characterize it and, therefore, the greater the uncertainty level [15].

Supply chain network design (SCND) is a concept broadly studied during the last decades, both from a qualitative and a quantitative perspective. Authors have referred to it by using the terms *supply chain design* and *supply chain network design* [23] state that a SCND problem “comprises the decisions regarding the number and location of production facilities, the amount of capacity at each facility, the assignment of each market region to one or more locations, and supplier selection for sub-assemblies, components and materials”. These decisions are related to a strategic level, and must be optimized considering a long-term (usually several years) efficient operation of the supply chain as a whole [6]. One of the more challenging responsibilities in SCND is addressing uncertainty.

* Corresponding author.
E-mail addresses: rtordecilla@uoc.edu (R.D. Tordecilla), ajuanp@uoc.edu (A.A. Juan), jairo.montoya@unisabana.edu.co (J.R. Montoya-Torres), carlos.quintero5@unisabana.edu.co (C.L. Quintero-Araujo), jpanaderom@uoc.edu (J. Panadero).

https://doi.org/10.1016/j.simpat.2020.102166
Received 18 March 2020; Received in revised form 6 July 2020; Accepted 7 August 2020
Available online 11 August 2020

applied sciences

MDPI

Article

Fuzzy Simheuristics for Optimizing Transportation Systems: Dealing with Stochastic and Fuzzy Uncertainty

Rafael D. Tordecilla [1,2], Leandro do C. Martins [1], Javier Panadero [1,3], Pedro J. Copado [1,3], Elena Perez-Bernabeu [4] and Angel A. Juan [1,3,*]

[1] IN3–Computer Science Department, Universitat Oberta de Catalunya, 08018 Barcelona, Spain; rtordecilla@uoc.edu (R.D.T.); leandrocm@uoc.edu (L.d.C.M.); jpanaderom@uoc.edu (J.P.); pcopadom@uoc.edu (P.J.C.)
[2] School of Engineering, Universidad de La Sabana, Chia 250001, Colombia
[3] Euncet Business School, 08225 Terrassa, Spain
[4] Department of Applied Statistics and Operations Research, Universitat Politècnica de València, 03801 Alcoy, Spain; elenapb@eio.upv.es
* Correspondence: ajuanp@uoc.edu

Abstract: In the context of logistics and transportation, this paper discusses how simheuristics can be extended by adding a fuzzy layer that allows us to deal with complex optimization problems with both stochastic and fuzzy uncertainty. This hybrid approach combines simulation, metaheuristics, and fuzzy logic to generate near-optimal solutions to large scale NP-hard problems that typically arise in many transportation activities, including the vehicle routing problem, the arc routing problem, or the team orienteering problem. The methodology allows us to model different components–such as travel times, service times, or customers' demands–as deterministic, stochastic, or fuzzy. A series of computational experiments contribute to validate our hybrid approach, which can also be extended to other optimization problems in areas such as manufacturing and production, smart cities, telecommunication networks, etc.

Keywords: transportation; vehicle routing problems; metaheuristics; simulation-optimization; fuzzy techniques

Citation: Tordecilla, R.D.; Martins, L.d.C.; Panadero, J.; Copado, P.J.; Perez-Bernabeu, E.; Juan, A.A. Fuzzy Simheuristics for Optimizing Transportation Systems: Dealing with Stochastic and Fuzzy Uncertainty. *Appl. Sci.* **2021**, *11*, 7950. https://doi.org/10.3390/app11177950

Academic Editor: Ludmila Dymova

Received: 24 July 2021
Accepted: 26 August 2021
Published: 28 August 2021

1. Introduction

Managers tend to rely on analytical methods that allow them to make informed decisions. This explains why optimization models play a key role in many industries and business, including the logistics and transportation sector. Whenever accurate information on the inputs and constraints of the optimization problem is available, the resulting deterministic models can be solved by using well-known methods, either of exact or approximate nature.

Many optimization problems in real-life transportation involve taking into account a large number of variables and rich constraints, which often makes them to be NP-hard [1]. When this is the case, the computational complexity makes it difficult to obtain optimal solutions in a short computational time. At this point, heuristic approaches can provide near-optimal solutions that, in turn, cover all the requirements of the problem [2]. When dealing with challenging optimization problems, there is a tendency to divide them into sub-problems, which simplifies the difficulty but might also lead to sub-optimal solutions [3,4]. Given the increase in computational power experienced during the last decade, and also the development of advanced metaheuristic algorithms, it is possible nowadays to solve rich and large-scale problems that were intractable in the past [5].

In the scientific literature on combinatorial optimization problems, it is often assumed that the input values are constant and known. However, in a real-world scenario this is rarely the case, since uncertainty is often present and affects these inputs. In the context

INTERNATIONAL TRANSACTIONS
IN OPERATIONAL RESEARCH

WILEY

INTERNATIONAL TRANSACTIONS IN OPERATIONAL RESEARCH

Intl. Trans. in Op. Res. 0 (2022) 1–31
DOI: 10.1111/itor.13125

The location routing problem with facility sizing decisions

Rafael D. Tordecilla[a,b,*], Jairo R. Montoya-Torres[b], Carlos L. Quintero-Araujo[c], Javier Panadero[a,d] and Angel A. Juan[e]

[a] *IN3 – Computer Science Department, Universitat Oberta de Catalunya, Barcelona, Spain*
[b] *School of Engineering, Universidad de La Sabana, Chia, Colombia*
[c] *International School of Economics and Administrative Sciences, Universidad de La Sabana, Chia, Colombia*
[d] *Euncet Business School, Terrassa, Spain*
[e] *Department of Applied Statistics and Operations Research, Universitat Politècnica de València, Alcoy, Spain*
E-mail: rtordecilla@uoc.edu [Tordecilla]; jairo.montoya@unisabana.edu.co [Montoya-Torres]; carlosqa@unisabana.edu.co [Quintero-Araujo]; jpanaderom@uoc.edu [Panadero]; ajuanp@eio.upv.es [Juan]

Received 6 August 2020; received in revised form 21 September 2021; accepted 25 January 2022

Abstract

The location routing problem (LRP) integrates operational decisions on vehicle routing operations with strategic decisions on the location of the facilities or depots from which the distribution will take place. In other words, it combines the well-known vehicle routing problem (VRP) with the facility location problem (FLP). Hence, the LRP is an *NP-hard* combinatorial optimization problem, which justifies the use of metaheuristic approaches whenever large-scale instances need to be solved. In this paper, we explore a realistic version of the LRP in which facilities of different capacities are considered, i.e., the manager has to consider not only the location but also the size of the facilities to open. In order to tackle this optimization problem, three mixed-integer linear formulations are proposed and compared. As expected, they have been proved to be cost- and time- inefficient. Hence, a biased-randomized iterated local search algorithm is proposed. Classical instances for the LRP with homogeneous facilities are naturally extended to test the performance of our approach.

Keywords: location routing problem; heterogeneous facilities; biased randomization; metaheuristics

1. Introduction

The location routing problem (LRP) is a traditional strategic-tactical-operational problem that considers a set of potential facilities and a set of customers with a known demand, whose main decisions are: (i) the number and location of facilities to open, (ii) the allocation of customers to open facilities, and (iii) the design of routes to serve customers from each facility using a fleet of vehicles.

*Corresponding author.

TOP
https://doi.org/10.1007/s11750-020-00558-x

ORIGINAL PAPER

A biased-randomized iterated local search for the vehicle routing problem with optional backhauls

Julio C. Londoño[1] · Rafael D. Tordecilla[2,3] · Leandro do C. Martins[2] · Angel A. Juan[2]

Received: 29 July 2019 / Accepted: 18 April 2020

Abstract

The vehicle routing problem with backhauls integrates decisions on product delivery with decisions on the collection of returnable items. In this paper, we analyze a scenario in which collection of items is optional—but subject to a penalty cost. Both transportation costs and penalties associated with non-collecting decisions are considered. A mixed-integer linear model is proposed and solved for small instances. Also, a metaheuristic algorithm combining biased randomization techniques with iterated local search is introduced for larger instances. Our approach yields cost savings and is competitive when compared to other state-of-the-art approaches.

Keywords Vehicle routing problem with optional backhauls · Returnable transport items · Biased randomization · Iterated local search

Mathematics Subject Classification 90B06 · 90C11 · 90C59

✉ Rafael D. Tordecilla
rtordecilla@uoc.edu

Julio C. Londoño
julio.londono@correounivalle.edu.co

Leandro do C. Martins
leandrocm@uoc.edu

Angel A. Juan
ajuanp@uoc.edu

1 Faculty of Engineering, Universidad del Valle, Cali, Colombia

2 IN3-Computer Science Department, Universitat Oberta de Catalunya, 08860 Castelldefels, Spain

3 Faculty of Engineering, Universidad de La Sabana, Km 7 Autopista norte de Bogota, D.C., Chia, Colombia

energies

Article

Electric Vehicle Routing, Arc Routing, and Team Orienteering Problems in Sustainable Transportation

Leandro do C. Martins [1], Rafael D. Tordecilla [1,2], Juliana Castaneda [1], Angel A. Juan [1,*] and Javier Faulin [3]

[1] IN3–Computer Science Department, Universitat Oberta de Catalunya, 08018 Barcelona, Spain; leandrocm@uoc.edu (L.d.C.M.); rtordecilla@uoc.edu (R.D.T.); jcastanedaji@uoc.edu (J.C.)
[2] School of Engineering, Universidad de La Sabana, Chia 250001, Colombia
[3] Institute of Smart Cities, Department Statistics, Computer Sciences, and Mathematics, Public University of Navarre, 31006 Pamplona, Spain, javier.faulin@unavarra.es
* Correspondence: ajuanp@uoc.edu

Abstract: The increasing use of electric vehicles in road and air transportation, especially in last-mile delivery and city mobility, raises new operational challenges due to the limited capacity of electric batteries. These limitations impose additional driving range constraints when optimizing the distribution and mobility plans. During the last years, several researchers from the Computer Science, Artificial Intelligence, and Operations Research communities have been developing optimization, simulation, and machine learning approaches that aim at generating efficient and sustainable routing plans for hybrid fleets, including both electric and internal combustion engine vehicles. After contextualizing the relevance of electric vehicles in promoting sustainable transportation practices, this paper reviews the existing work in the field of electric vehicle routing problems. In particular, we focus on articles related to the well-known vehicle routing, arc routing, and team orienteering problems. The review is followed by numerical examples that illustrate the gains that can be obtained by employing optimization methods in the aforementioned field. Finally, several research opportunities are highlighted.

Keywords: electric batteries; vehicle routing problem; arc routing problem; team orienteering problem

Citation: Martins, L.C.; Tordecilla, R.D.; Castaneda, J.; Juan, A.A.; Faulin, J. Electric Vehicle Routing, Arc Routing, and Team Orienteering Problems in Sustainable Transportation. *Energies* **2021**, *14*, 5131. https://doi.org/10.3390/en14165131

Academic Editors: Daniel J. Auger and Jorge Barreras

Received: 18 July 2021
Accepted: 17 August 2021
Published: 19 August 2021

1. Introduction

With the goal of promoting sustainability, many cities in the world are observing an increasing use of electric vehicles (EVs), both for citizens' mobility [1] and for last-mile logistics [2]. The use of zero-emission technologies is supported by governmental plans in regions such as Europe [3], North America [4], and Asia [5]. According to Kapustin and Grushevenko [6], EVs will account for a noticeable share (between 11% and 28%) of the road transportation fleet by 2040. Still, many authors point out batteries' driving range anxiety, high recharging times, scarcity of recharging stations, and lack of effective financial incentives that compensate for the higher cost of most EV models as some of the main barriers for the generalization of EVs in our cities [7–9].

In urban, peri-urban, and metropolitan areas, many activities related to freight transportation and citizens' mobility are carried out by fleets of vehicles [10]. The efficient coordination of these fleets becomes necessary in order to reduce monetary costs, operation times, energy consumption, and environmental/social impacts on the city. However, this coordination constitutes a relevant challenge that is typically modeled as a mathematical optimization problem. Depending on the specific characteristics of the transportation activity, different families of problems can be found in the scientific literature. Among the most popular ones, we can include vehicle routing problems (VRPs) [11–13], arc routing problems (ARPs) [14,15], and team orienteering problems (TOPs) [16,17]. These problems, which can model scenarios involving both road and aerial EVs, are NP-hard even in their

INFORMS JOURNAL ON APPLIED ANALYTICS
Articles in Advance, pp. 1–16
ISSN 2644-0865 (print), ISSN 2644-0873 (online)
http://pubsonline.informs.org/journal/inte

A Digital Twin for Decision Making on Livestock Feeding

David Raba,[a] Rafael D. Tordecilla,[a] Pedro Copado,[a] Angel A. Juan,[a] Daniel Mount[b]

[a] Universitat Oberta de Catalunya, 08060 Barcelona, Spain; [b] INSYLO Technologies, Inc., 17002 Girona, Spain

Contact: draba@uoc.edu, https://orcid.org/0000-0002-4181-5279 (DR); rtordecilla@uoc.edu, https://orcid.org/0000-0003-3028-1686 (RDT); pcopadom@uoc.com, https://orcid.org/0000-0003-4219-5056 (PC); ajuanp@uoc.com, https://orcid.org/0000-0003-1392-1776 (AAJ); daniel.mount@insylo.com, https://orcid.org/0000-0001-8412-805X (DM)

Received: February 4, 2021
Revised: May 26, 2021; August 9, 2021
Accepted: September 2, 2021
Published Online in Articles in Advance: December 30, 2021

https://doi.org/10.1287/inte.2021.1110

Abstract: This work is part of the IoFEED project, which aims at monitoring approximately 325 farm bins and investigates business processes carried out between farmers and animal feed producers. We propose a computer-aided system to control and optimize the supply chain to deliver animal feed to livestock farms. Orders can be of multiple types of feed and shipped from multiple depots by using a fleet of heterogeneous vehicles with multiple compartments. Additionally, this case considers some business-specific constraints, such as product compatibility, facility accessibility restrictions, prioritized locations, or biosecurity constraints. A digital twin–based approach is implemented at the farm level by installing sensors to remotely measure the inventories. Our approach combines biased-randomization techniques with a simheuristic framework to make use of data provided by the sensors. The analysis of results is based on these two real pilots and showcases the insights obtained during the IoFEED project. The results of this work show how the internet of things and simulation-based optimization methods combine successfully to optimize the feeding operations of livestock farms.

Funding: This work was partially supported by the EU-IoF2020 project [Grant 731884], the Spanish Ministry of Economy and Competitiveness [Grant DI-15-08176], and the Catalan Agency for Management of University and Research Grants [Grant 2016-DI-038].

Keywords: vehicle routing problem • internet of things • animal farming • feeding • heuristics

Livestock production in the European Union represents 40% of the overall agriculture output. The European feed sector is of utmost importance to the livestock industry. Farm animals in the European Union consume an estimated 478 million metric tons of feed a year, of which 163 million metric tons are produced by compound feed manufacturers (FEFAC 2018). The European feed industry is a growing industry, with an estimated turnover of €50 billion, that directly employs approximately 110,000 people, most of them in rural areas where employment offers are usually scarce. Even though most of the compound feed plants are small and medium enterprises (SMEs), they have an average production volume of 40,000 tons (t) of compound feed per plant (FEFAC 2019). The quality of this compound feed is really important to farmers because it directly correlates with milk or meat quality. A better knowledge of the farms' nutritional needs gives the feed manufacturers the best position to plan raw material procurement and gives them a reliable supply chain with short lead times that will allow them to replenish their silos before they run out of feed. Final delivery is often done by trucks. Hence, an efficient distribution relies on how routes are planned. The same truck will cover a wider product variety for the same trip, depending on the number of compartments. Moreover, this transport fleet can be totally or partially owned by the feed manufacturer. Outsourcing is commonly used to increase service capacity during peak periods. At the feed mill, raw materials are processed into grain or pellets. The feed mill produces a certain number of products—according to the demand, which varies throughout the week—and keeps them in stock. The more orders per day that a feed manufacturer receives, the more complicated it is to achieve optimal production and distribution. For make-to-order processes, it is of utmost importance to have demand forecasts, precisely for adopting make-to-stock processes that will smooth peaks in production. As a result, being able to serve large orders and unexpected demands will depend on these decisions.

In this paper, we describe the implementation of a computer-based solution to address the problem of delivering animal feed to farms. Additionally, we discuss the benefits of integrating digital twins (DTs) with system simulation and the internet of things (IoT). This paper aims to identify and quantify yield savings generated by suppressing stock runouts (up to £28,000/year), better inventory management at farms (£30,000/year), and automatic order scheduling. This work is part of the IoFEED project

energies

Article

Edge Computing and IoT Analytics for Agile Optimization in Intelligent Transportation Systems

Mohammad Peyman [1,*], Pedro J. Copado [1,2], Rafael D. Tordecilla [1,3], Leandro do C. Martins [1], Fatos Xhafa [4,*] and Angel A. Juan [1,2]

1 IN3—Computer Science Department, Universitat Oberta de Catalunya, 08018 Barcelona, Spain; pcopadom@uoc.edu (P.J.C.); rtordecilla@uoc.edu (R.D.T.); leandrocm@uoc.edu (L.d.C.M.); ajuanp@uoc.edu (A.A.J.)
2 Department of Data Analytics&Business Intelligence, Euncet Business School, 08018 Barcelona, Spain
3 School of Engineering, Universidad de La Sabana, Chia 250001, Colombia
4 Computer Science Department, Universitat Politècnica de Catalunya, 08034 Barcelona, Spain
* Correspondence: mpeyman@uoc.edu (M.P.); fatos@cs.upc.edu (F.X.)

Abstract: With the emergence of fog and edge computing, new possibilities arise regarding the data-driven management of citizens' mobility in smart cities. Internet of Things (IoT) analytics refers to the use of these technologies, data, and analytical models to describe the current status of the city traffic, to predict its evolution over the coming hours, and to make decisions that increase the efficiency of the transportation system. It involves many challenges such as how to deal and manage real and huge amounts of data, and improving security, privacy, scalability, reliability, and quality of services in the cloud and vehicular network. In this paper, we review the state of the art of IoT in intelligent transportation systems (ITS), identify challenges posed by cloud, fog, and edge computing in ITS, and develop a methodology based on agile optimization algorithms for solving a dynamic ride-sharing problem (DRSP) in the context of edge/fog computing. These algorithms allow us to process, in real time, the data gathered from IoT systems in order to optimize automatic decisions in the city transportation system, including: optimizing the vehicle routing, recommending customized transportation modes to the citizens, generating efficient ride-sharing and car-sharing strategies, create optimal charging station for electric vehicles and different services within urban and interurban areas. A numerical example considering a DRSP is provided, in which the potential of employing edge/fog computing, open data, and agile algorithms is illustrated.

Keywords: fog; edge computing; Internet of Things; intelligent transportation systems; smart cities; machine learning; agile optimization

Citation: Peyman, M.; Copado, P.J.; Tordecilla, R.D.; Martins, L.d.C.; Xhafa, F.; Juan, A.A. Edge Computing and IoT Analytics for Agile Optimization in Intelligent Transportation Systems. *Energies* **2021**, *14*, 6309. https://doi.org/10.3390/en14196309

Academic Editor: Sergio Saponara

Received: 2 August 2021
Accepted: 29 September 2021
Published: 2 October 2021

1. Introduction

In today's modern society, urban centers are facing the so-called booming of information. Due to the population growth in many countries around the globe, and recent innovations in information and telecommunication technologies, several activities and related challenges have jointly arisen. People are increasingly consuming more information through their mobile devices, vehicles are equipped with different intelligent systems, devices are distributed around the cities for gathering and generating information, and urban areas are continuously taking advantage of these information technologies and big data. Consequently, so-called smart cities have emerged, whose scope combines sustainable development with the intelligent management of gathered data in order to enhance the operation of different services within urban areas, such as waste collection management [1], car-sharing/ride-sharing activities [2], the optimal location of recharging stations for electric vehicles (EVs), among others. In this matter, during the past few years, the Internet of things (IoT) has become a popular term that plays a significant role to expand and produce a lot of data through sensors and allows citizens and things to be connected in any situation

B.2.2 Scopus-Indexed Articles

Chapter 18
Agile Computational Intelligence for Supporting Hospital Logistics During the COVID-19 Crisis

Rafael D. Tordecilla, Leandro do C. Martins, Miguel Saiz, Pedro J. Copado-Mendez, Javier Panadero, and Angel A. Juan

Abstract This chapter describes a case study regarding the use of 'agile' computational intelligence for supporting logistics in Barcelona's hospitals during the COVID-19 crisis in 2020. Due to the lack of sanitary protection equipment, hundreds of volunteers, the so-called "Coronavirus Makers" community, used their home 3D printers to produce sanitary components, such as face covers and masks, which protect doctors, nurses, patients, and other civil servants from the virus. However, an important challenge arose: how to organize the daily collection of these items from individual homes, so they could be transported to the assembling centers and, later, distributed to the different hospitals in the area. For over one month, we have designed daily routing plans to pick up the maximum number of items in a limited time—thus reducing the drivers' exposure to the virus. Since the problem characteristics were different each day, a series of computational intelligence algorithms was employed. Most of them included flexible heuristic-based approaches and biased-randomized

R. D. Tordecilla ( ) · L. C. Martins · M. Saiz · P. J. Copado-Mendez · J. Panadero · A. A. Juan
IN3—Computer Science Department, Universitat Oberta de Catalunya, 08018 Barcelona, Spain
e-mail: rtordecilla@uoc.edu

L. C. Martins
e-mail: leandrocm@uoc.edu

M. Saiz
e-mail: msaiz@weoptimize.net

P. J. Copado-Mendez
e-mail: pcopadom@uoc.edu

J. Panadero
e-mail: jpanaderom@uoc.edu

A. A. Juan
e-mail: ajuanp@uoc.edu

R. D. Tordecilla
Universidad de La Sabana, Chía, Cundinamarca, Colombia

M. Saiz
weOptimize, 43700 El Vendrell, Spain

P. J. Copado-Mendez · J. Panadero · A. A. Juan
Euncet Business School, Terrassa 08225, Spain

S. Patnaik et al. (eds.), *Computational Management*, Modeling and Optimization in Science and Technologies 18, https://doi.org/10.1007/978-3-030-72929-5_18 *algorithms*

Article

Combining Heuristics with Simulation and Fuzzy Logic to Solve a Flexible-Size Location Routing Problem under Uncertainty

Rafael D. Tordecilla [1,2,*], Pedro J. Copado-Méndez [1,3], Javier Panadero [1,3], Carlos L. Quintero-Araujo [4], Jairo R. Montoya-Torres [2] and Angel A. Juan [1,3]

[1] IN3–Computer Science Department, Universitat Oberta de Catalunya, Barcelona 08018, Spain; pcopadom@uoc.edu (P.J.C.-M.); jpanaderom@uoc.edu (J.P.); ajuanp@uoc.edu (A.A.J.)
[2] School of Engineering, Universidad de La Sabana, Chia 250001, Colombia; jairo.montoya@unisabana.edu.co
[3] Department of Data Analytics & Business Intelligence, Euncet Business School, 08221 Terrassa, Spain
[4] International School of Economics and Administrative Sciences, Universidad de La Sabana, Chia 250001, Colombia; carlosqa@unisabana.edu.co
* Correspondence: rtordecilla@uoc.edu or rafael.tordecilla@unisabana.edu.co

Abstract: The location routing problem integrates both a facility location and a vehicle routing problem. Each of these problems are *NP-hard* in nature, which justifies the use of heuristic-based algorithms when dealing with large-scale instances that need to be solved in reasonable computing times. This paper discusses a realistic variant of the problem that considers facilities of different sizes and two types of uncertainty conditions. In particular, we assume that some customers' demands are stochastic, while others follow a fuzzy pattern. An iterated local search metaheuristic is integrated with simulation and fuzzy logic to solve the aforementioned problem, and a series of computational experiments are run to illustrate the potential of the proposed algorithm.

Keywords: location routing problem; uncertainty; heuristics; simulation; fuzzy logic

Citation: Tordecilla, R.D.; Copado-Méndez, P.J.; Panadero, J.; Quintero-Araujo, C.L.; Montoya-Torres, J.R.; Juan, A.A. Combining Heuristics with Simulation and Fuzzy Logic to Solve a Flexible-Size Location Routing Problem under Uncertainty. *Algorithms* **2021**, *14*, 45. https://doi.org/10.3390/a14020045

Academic Editor: Javier Del Ser Lorente

Received: 21 January 2021
Accepted: 26 January 2021
Published: 30 January 2021

1. Introduction

When designing and managing supply chains, one of the most relevant problems is the simultaneous location of distribution facilities and the routing of vehicles to deliver products to a set of geographically dispersed customers. The former is considered a strategic decision, while the latter is operational. This problem is known in the scientific literature as the location routing problem (LRP). The LRP addresses these two types of decisions in an integrated manner. From the formal view of the operational research community, the LRP is known to be *NP-hard*, since it can be reduced to either the facility location problem (FLP), the vehicle routing problem (VRP) or the multidepot VRP, which are all known to be *NP-hard*. This computational complexity means that optimal solutions are really difficult to obtain in a reasonable computational time. Thus, heuristic approaches are required to solve medium- and large-sized instances. Due to its complexity, some of the first studies tackled the problem by splitting it into the corresponding subproblems [1,2]. Nevertheless, this approach might lead to suboptimal solutions.

Due to the increase in computational power and the development of fast heuristic approaches, the LRP has been studied in an integrated way, which clearly has improved the obtained results [3]. One of the most studied versions of the LRP is the capacitated LRP, in which both depot and vehicle capacity constraints must be satisfied (the acronym LRP will henceforth refer to this version). However, all previous works consider the depot capacity as a fixed value for each location. This could not be a suitable approach when dealing with realistic problems, since it is usual that decision-makers can select the size of a facility from a discrete set of known available sizes, or even freely. For real-world problems, this set is usually associated with investment activities, such as building facilities [4], purchasing

Proceedings of the 2020 Winter Simulation Conference
K.-H. Bae, B. Feng, S. Kim, S. Lazarova-Molnar, Z. Zheng, T. Roeder, and R. Thiesing, eds.

2020 Winter Simulation Conference (WSC) | 978-1-7281-9499-8/20/$31.00 ©2020 IEEE | DOI: 10.1109/WSC48552.2020.9384053

A SIMHEURISTIC ALGORITHM FOR THE LOCATION ROUTING PROBLEM WITH FACILITY SIZING DECISIONS AND STOCHASTIC DEMANDS

Rafael D. Tordecilla
Javier Panadero
Angel A. Juan

IN3 – Computer Science Dept.
Universitat Oberta de Catalunya
Avda. Carl Friedrich Gauss, 5
Castelldefels, 08860, SPAIN

Carlos L. Quintero-Araújo
Jairo R. Montoya-Torres

Int. School of Economics and Admin. Sci.
Faculty of Engineering
Universidad de La Sabana
Chia, COLOMBIA

ABSTRACT

Location routing is a well known problem in which decisions about facility location and vehicle routing must be made. Traditionally, a fixed size or capacity is assigned to an open facility as the input parameter to the problem. However, real-world cases show that decision-makers usually have a set of size options. If this size is selected accurately according to the demand of allocated customers, then location decisions and routing activities would raise smaller cost. Nevertheless, choosing this size implies additional variables that make an already *NP-hard* problem even more challenging. In addition, considering stochastic demands contributes to making the optimization problem more difficult to solve. Hence, a simheuristic algorithm is proposed in this work. It combines the efficiency of metaheuristics and the capabilities of simulation to deal with uncertainty. A series of computational experiments show that our approach can efficiently deal with medium-large instances.

1 INTRODUCTION

The Location Routing Problem (LRP) is one of the most complete problems in logistics optimization, since it includes all decision levels, i.e., strategic, tactical, and operational. From an Operational Research perspective, it can be seen as the combination of the Facility Location Problem (FLP) and the Vehicle Routing Problem (VRP), which are both *NP-hard* problems. Hence, the LRP is also *NP-Hard*, and heuristic approaches are required for solving medium- and large-sized instances. Due to its complexity, the first reported studies on the LRP tackled it by separating the corresponding sub-problems (Salhi and Rand 1989; Nagy and Salhi 2007). As expected, this approach led to sub-optimal solutions. More recently, given the increase in computational power and the development of non-exact approaches, such as heuristic and metaheuristic algorithms, the LRP has been studied in an integrated way, which has clearly improved the obtained results (Prodhon and Prins 2014). The LRP has been used to support decision-making processes related to supply chain network design (Lashine et al. 2006), humanitarian logistics (Ukkusuri and Yushimito 2008), horizontal cooperation (Quintero-Araujo et al. 2019), and city logistics (Nataraj et al. 2019), among others. One of the most studied versions of the LRP is the Capacitated LRP, in which both depot and vehicle capacity constraints must be satisfied (the acronym LRP will henceforth refer to this problem). However, all previous works consider the depot capacity as a fixed value. This could not be a suitable approach when dealing with realistic problems, since it is usual that decision-makers can select the size of a facility from a discrete set of known available sizes, or even freely. For real-world problems, this set is usually associated with investment activities, such as building facilities (Zhou et al. 2019), purchasing equipment (Tordecilla-Madera et al. 2017), or qualifying workforce (Correia and Melo 2016). From an academic point of view, the consideration of flexible sizes in the facilities has been rarely addressed in the literature.

Available online at www.sciencedirect.com

ScienceDirect

Transportation Research Procedia 58 (2021) 385–392

Transportation Research Procedia

www.elsevier.com/locate/procedia

14th Conference on Transport Engineering: 6th – 8th July 2021

An Agile and Reactive Biased-Randomized Heuristic for an Agri-Food Rich Vehicle Routing Problem

Rafael D. Tordecilla[a,b,*], Pedro J. Copado-Méndez[a], Javier Panadero[a], Leandro do C. Martins[a], Angel A. Juan[a]

[a]IN3 – Computer Science Dept., Universitat Oberta de Catalunya, Barcelona 08018, Spain
[b]School of Engineering, Universidad de La Sabana, Chia 250001, Colombia

Abstract

Operational problems in agri-food supply chains usually show characteristics that are scarcely addressed by traditional academic approaches. These characteristics make an already NP-hard problem even more challenging; hence, this problem requires the use of tailor-made algorithms in order to solve it efficiently. This work addresses a rich vehicle routing problem in a real-world agri-food supply chain. Different types of animal food products are distributed to raising-pig farms. These products are incompatible, i.e., multi-compartment heterogeneous vehicles must be employed to perform the distribution activities. The problem considers constraints regarding visit priorities among farms, and not-allowed access of large vehicles to a subset of farms. Finally, a set of flat tariffs are employed to formulate the cost function. This problem is solved employing a reactive savings-based biased-randomized heuristic, which does not require any time-costly parameter fine-tuning process. Our results show savings in both cost and traveled distance when compared with the real supply chain performance.

Keywords: Rich Vehicle Routing Problem, Agri-Food Supply Chain, Biased-Randomized Heuristic.

1. Introduction

Feeding pigs in the pork production industry is a highly relevant activity to successfully achieve the supply chain goals (Rodríguez, 2014). Such activity requires a precise logistics from the production plant to the farms where the pigs are raised. Hence, our work consists in designing a set of vehicle routes that meet the feed demand of a set of pig farms, considering the real case of a pork production company in Spain. From an academic point of view, the analyzed

* Corresponding author.
E-mail address: rtordecilla@uoc.edu, rafael.tordecilla@unisabana.edu.co

10.1016/j.trpro.2021.11.052

Proceedings of the 2021 Winter Simulation Conference
S. Kim, B. Feng, K. Smith, S. Masoud, Z. Zheng, C. Szabo, and M. Loper, eds.

SUPPORTING HOSPITAL LOGISTICS DURING THE FIRST MONTHS OF THE COVID-19 CRISIS: A SIMHEURISTIC FOR THE STOCHASTIC TEAM ORIENTEERING PROBLEM

Markus Rabe
Jorge Chicaiza-Vaca

Department of IT in Production and Logistics
Technical University Dortmund
Leonhard-Euler-Str. 5
Dortmund, 44227, GERMANY

Rafael D. Tordecilla
Leandro do C. Martins
Angel A. Juan

IN3 – Computer Science Dept.
Universitat Oberta de Catalunya
Rambla Poblenou, 156
Barcelona, 08018, SPAIN

ABSTRACT

The unexpected crisis posed by the COVID-19 pandemic since March 2020 caused that items such as face shields, ear savers, and door openers were in high demand. In the area of Barcelona, thousands of volunteers employed their home 3D printers to produce these elements. Due to the lockdown, they had to be collected at each individual house by a reduced group of volunteer drivers, who transported them to several consolidation centers. These activities required a daily agile design of efficient routes, especially considering that drivers' exposure should be minimized – i.e., routes should not exceed a maximum time threshold. These constraints limit the number of individual houses that could be visited every day in order to collect newly produced items. Moreover, being a real-life environment, travel and service times are better modeled as random variables, which increases the problem complexity. This logistics challenge can be modeled as a stochastic team orienteering problem, with the objective of maximizing the total collected reward while satisfying the constraints on the fleet size and the maximum travel time per route. In order to solve this stochastic optimization problem, a simheuristic algorithm is proposed. Our approach, which also makes use of biased-randomization techniques, is able of generating high-quality solutions in short computing times.

1 INTRODUCTION

The COVID-19 pandemic crisis is one of the more recent greatest global challenges. The exponential increase in cases requiring medical care led to a sudden shortage of protective materials, putting medical and support staff at high risk of becoming infected as well. This not only jeopardized necessary attention in hospitals, but also accelerated the spread of COVID-19. Since March 2020, the pandemic has also had a strong impact in countries such as Germany and Spain. As in other regions, a community of volunteers called "Coronavirus Makers" was created in the Barcelona region to provide protective materials to staff in hospitals, nursing homes, and emergency medical care. The main tool was domestic 3D printers, which allowed a very quick design and elaboration of elements such as face shields, ear savers, or door openers. The bottleneck in this context was mainly a logistics one, as the lockdown meant that each 3D printer was in a single home, and collecting the items required optimizing the routing plans to maximize the added value of the items gathered while keeping drivers' safety. This paper describes the experience of bringing together different professional and personal profiles such as academics, volunteers, makers, and entrepreneurs, who typically employ different approaches when dealing with the pandemic. In this case, there was a need to find a quick way to apply knowledge accumulated over years of research to an urgent need, where every day counts. The goal was to support the Makers community in their volunteer initiative

Proceedings of the 2021 Winter Simulation Conference
S. Kim, B. Feng, K. Smith, S. Masoud, Z. Zheng, C. Szabo, and M. Loper, eds.

WASTE COLLECTION OF MEDICAL ITEMS UNDER UNCERTAINTY USING INTERNET OF THINGS AND CITY OPEN DATA REPOSITORIES: A SIMHEURISTIC APPROACH

Mohammad Peyman
Yuda Li
Rafael D. Tordecilla
Pedro J. Copado-Mendez
Angel A. Juan

IN3 – Computer Science Department
Universitat Oberta de Catalunya
Rambla Poblenou, 156
Barcelona, 08018, SPAIN

Fatos Xhafa

Computer Science Department
Universitat Politècnica de Catalunya
Carrer de Jordi Girona, 3
Barcelona, 08034, SPAIN

ABSTRACT

In the current pandemic situation, a large quantity of medical items are being consumed by citizens all over the world. If not properly collected and processed, these items can be pollutant or even dangerous for many people. Inspired by a real case study in the city of Barcelona, and assuming that data from container sensors are available in the city open repository, this work addresses a medical waste collection problem both with and without uncertainty. The waste collection process is modeled as a rich version of the open vehicle routing problem, where the capacity constraints are not in the loading dimension but in the maximum time each vehicle can circulate without having to perform a mandatory stop, with the goal of minimizing the total time required to complete the waste collection process. To provide high - quality solutions to this complex problem, a biased - randomized heuristic is initially proposed. This heuristic is then combined with simulation to provide effective collection plans in scenarios where travel times and pick - up times are modeled as random variables.

1 INTRODUCTION

The outbreak of the COVID - 19 pandemic not only has caused a significant global social and economic crisis but also has dramatic effects on the environment. To fight the spread of COVID - 19, governments and health officials around the globe have introduced mandatory policies including lock - downs, quarantines, and border closures. While these measures have positive impacts on the environment due to the reductions in air pollution, they are most likely temporary as pollution levels may rise again when the world recovers from the pandemic. However, consumption of personal protective equipment (PPE), such as masks and gloves, during the pandemic has already generated more than billions of contaminated waste. To date, COVID - 19 continues to be a challenge to global public health. Saberian et al. (2021) estimate that 6.88 billion – approximately 206,470 tons – face masks are generated around the world each day. In many cities, the daily face mask usage (in terms of pieces quantity) can be roughly estimated by simply multiplying the city population size by the acceptance rate of masks (Nzediegwu and Chang 2020).

For instance, the population in Barcelona, Spain, is equal to $1,664,182$ people (Statistical Institute of Catalonia 2020). Despite wearing a face mask in public is mandatory in Barcelona, the acceptance rate is estimated to be about 80 %. Hence, we estimate the daily face mask usage in Barcelona to be about $1,331,345$ pieces. Many of these masks are made of petroleum - based non - renewable polymers that are non - biodegradable (Dharmaraj et al. 2021), which means that they take hundreds or even thousands

algorithms

MDPI

Article

On the Use of Biased-Randomized Algorithms for Solving Non-Smooth Optimization Problems

Angel Alejandro Juan [1,*], Canan Gunes Corlu [2], Rafael David Tordecilla [3] and Rocio de la Torre [4] and Albert Ferrer [5]

1 Internet Interdisciplinary Institute (IN3), Department of Computer Science, Multimedia and Telecommunication, Universitat Oberta de Catalunya & Euncet Business School, 08018 Barcelona, Spain
2 Metropolitan College, Boston University, Boston, MA 02215, USA; canan@bu.edu
3 Internet Interdisciplinary Institute (IN3), Universitat Oberta de Catalunya & Universidad de La Sabana, 08018 Barcelona, Spain; rtordecilla@uoc.edu
4 Institute for Advanced Research in Business and Economics (INARBE), Business Administration Department, Public University of Navarre, 31006 Pamplona, Spain; rocio.delatorre@unavarra.es
5 Department of Applied Mathematics, Universitat Politecnica de Catalunya, 08028 Barcelona, Spain; alberto.ferrer@upc.edu
* Correspondence: ajuanp@uoc.edu; Tel.: +34-932-53-2300

Received: 13 December 2019; Accepted: 23 December 2019; Published: 25 December 2019

Abstract: Soft constraints are quite common in real-life applications. For example, in freight transportation, the fleet size can be enlarged by outsourcing part of the distribution service and some deliveries to customers can be postponed as well; in inventory management, it is possible to consider stock-outs generated by unexpected demands; and in manufacturing processes and project management, it is frequent that some deadlines cannot be met due to delays in critical steps of the supply chain. However, capacity-, size-, and time-related limitations are included in many optimization problems as hard constraints, while it would be usually more realistic to consider them as soft ones, i.e., they can be violated to some extent by incurring a penalty cost. Most of the times, this penalty cost will be nonlinear and even noncontinuous, which might transform the objective function into a non-smooth one. Despite its many practical applications, non-smooth optimization problems are quite challenging, especially when the underlying optimization problem is *NP-hard* in nature. In this paper, we propose the use of biased-randomized algorithms as an effective methodology to cope with *NP-hard* and non-smooth optimization problems in many practical applications. Biased-randomized algorithms extend constructive heuristics by introducing a nonuniform randomization pattern into them. Hence, they can be used to explore promising areas of the solution space without the limitations of gradient-based approaches, which assume the existence of smooth objective functions. Moreover, biased-randomized algorithms can be easily parallelized, thus employing short computing times while exploring a large number of promising regions. This paper discusses these concepts in detail, reviews existing work in different application areas, and highlights current trends and open research lines.

Keywords: non-smooth optimization; biased-randomized algorithms; heuristics; soft constraints

1. Introduction

Optimization models are used in many practical situations to represent decision-making challenges in areas such as computational finance, transportation and logistics, telecommunication networks, smart cities, etc. [1]. Many of these challenges can be transformed into optimization problems (OPs) that can be then solved using a plethora of methods of both exact and approximate

Proceedings of the 2020 Winter Simulation Conference
K.-H. Bae, B. Feng, S. Kim, S. Lazarova-Molnar, Z. Zheng, T. Roeder, and R. Thiesing, eds.

A SIMULATION-OPTIMIZATION APPROACH FOR LOCATING AUTOMATED PARCEL LOCKERS IN URBAN LOGISTICS OPERATIONS

Markus Rabe
Jorge Chicaiza-Vaca

Department of IT in Production and Logistics
Technical University Dortmund
Leonhard-Euler-Str. 5
Dortmund, 44227, GERMANY

Rafael D. Tordecilla
Angel A. Juan

IN3 – Computer Science Dept.
Universitat Oberta de Catalunya
Euncet Business School
Barcelona, 08018, SPAIN

2020 Winter Simulation Conference (WSC) | 978-1-7281-9499-8/20/$31.00 ©2020 IEEE | DOI: 10.1109/WSC48552.2020.9384087

ABSTRACT

Experts propose using an automated parcel locker (APL) for improving urban logistics operations. However, deciding the location of these APLs is not a trivial task, especially when considering a multi-period horizon under uncertainty. Based on a case study developed in Dortmund, Germany, we propose a simulation-optimization approach that integrates a system dynamics simulation model with a multi-period capacitated facility location problem (CFLP). First, we built the causal-loop and stock-flow diagrams to show the APL system's main components and interdependencies. Then, we formulated a multi-period CFLP model to provide the optimal number of APLs to be installed in each period. Finally, Monte Carlo simulation was used to estimate the cost and reliability level for different scenarios with random demands. In our experiments, only one solution reaches a 100% reliability level, with a total cost of 2.7 million euros. Nevertheless, if the budget is lower, our approach offers other good alternatives.

1 INTRODUCTION

Researchers have used simulation-optimization (SO) techniques for solving complex transportation and logistics problems for many years (Figueira and Almada-Lobo 2014). Exploring the behavior of logistics systems, and estimating their response to various changes in their environment, is the primary purpose behind the use of simulation (Crainic et al. 2018). In logistics systems, SO enables to represent and estimate different scenarios for policy changes and environmental regulations, enabling better accommodation of logistics schemes. In this context, we focus on SO models in urban logistics (UL) systems. Urban logistics has been a subject of interest for researchers during the last decades. UL is defined by Gonzalez-Feliu et al. (2014) as "The multi-disciplinary field that aims to understand, study and analyze the different organizations, logistics schemes, stakeholders and planning actions related to the improvement of the different goods transport systems in an urban zone and link them in a synergic way to decrease the main nuisances related to it". Hence, UL includes different stakeholders who are seen in urban logistics, as well as a wide variety of aims, which imposes challenges to decision makers.

This paper focuses on the usage of automated parcel locker (APL) systems, such as pack-stations or locker boxes, as one of the most promising initiatives to improve the UL activities. The APL has electronic lockers with variable opening codes, so that it can be used by different consumers whenever it is convenient for them. APLs are located in apartment blocks, workplaces, railway stations, or near to consumers' homes, to which parcels are delivered. The costs of APL deliveries are lower than those of home deliveries, and the risk of missed deliveries is avoided. Some studies confirm that online shoppers will use APLs more frequently in the future (Moroz and Polkowski 2016). Despite there are limitations to the concept, many third-party logistics providers, such as DHL, InPost, Norway Post, PostDanmark, UPS, or Amazon continue

algorithms

MDPI

Article

Simulation-Optimization Approach for Multi-Period Facility Location Problems with Forecasted and Random Demands in a Last-Mile Logistics Application

Markus Rabe [1], Jesus Gonzalez-Feliu [2], Jorge Chicaiza-Vaca [1,*] and Rafael D. Tordecilla [3,4]

[1] IT in Production and Logistics, Faculty of Mechanical Engineering, TU Dortmund, 44227 Dortmund, Germany; markus.rabe@tu-dortmund.de
[2] Centre de Recherche en Intelligence et Innovation Managériales, Excelia Business School, 17000 La Rochelle, France; gonzalezfeliuj@excelia-group.com
[3] IN3—Computer Science Department, Universitat Oberta de Catalunya, 08018 Barcelona, Spain; rtordecilla@uoc.edu
[4] School of Engineering, Universidad de La Sabana, 53753 Chia, Colombia; rafael.tordecilla@unisabana.edu.co
* Correspondence: jorge.chicaiza@tu-dortmund.de

Abstract: The introduction of automated parcel locker (APL) systems is one possible approach to improve urban logistics (UL) activities. Based on the city of Dortmund as case study, we propose a simulation-optimization approach integrating a system dynamics simulation model (SDSM) with a multi-period capacitated facility location problem (CFLP). We propose this integrated model as a decision support tool for future APL implementations as a last-mile distribution scheme. First, we built a causal-loop and stock-flow diagram to show main components and interdependencies of the APL systems. Then, we formulated a multi-period CFLP model to determine the optimal number of APLs for each period. Finally, we used a Monte Carlo simulation to estimate the costs and reliability level with random demands. We evaluate three e-shopper rate scenarios with the SDSM, and then analyze ten detailed demand configurations based on the results for the middle-size scenario with our CFLP model. After 36 months, the number of APLs increases from 99 to 165 with the growing demand, and stabilizes in all configurations from month 24. A middle-demand configuration, which has total costs of about 750, 000€, already locates a suitable number of APLs. If the budget is lower, our approach offers alternatives for decision-makers.

Keywords: hybrid modeling; system dynamics; facility location problems; Monte Carlo simulation; automated parcel lockers; last-mile delivery

Citation: Rabe, M.; Gonzalez-Feliu, J.; Chicaiza-Vaca, J.; Tordecilla, R.D. Simulation-Optimization Approach for Multi-Period Facility Location Problems with Forecasted and Random Demands in a Last-Mile Logistics Application. *Algorithms* **2021**, *14*, 41. https://doi.org/10.3390/a14020041

Academic Editor: Alberto Policriti
Received: 14 December 2020
Accepted: 24 January 2021
Published: 28 January 2021

1. Introduction

Last-mile logistics (LML) is known as the least efficient and most complex part of the supply chain. LML activities have negative impacts on pollution and traffic congestion in urban areas [1]. The growth of e-commerce activities has increased the number of individual home deliveries, thus driving up LML flows. Improving the efficiency of LML in urban areas through research is an important driver for the success of e-commerce and helps to reduce the negative externalities associated with urban logistics (UL).

An automated parcel locker (APL) is a potential solution to LML challenges. In our current work, we analyze the use of APLs such as packstations or locker boxes as a promising alternative to improve UL activities [2]. An APL is a group of electronic lockers with variable opening codes. APLs can be used by different consumers whenever it is convenient for them. APLs are located near consumers' homes, workplaces, and train stations, where online shoppers deliver or ship packages. The costs of home delivery and the related risk of missed delivery are likely to be higher compared APL systems. Online shoppers are likely to use APLs more often in the future [3]. Third-party logistics providers such as DHL, InPost, Norway Post, UPS, or Amazon continue to invest in APLs to gain a competitive advantage [3].