A Metaheuristic and Simheuristic Approach for the *p*-Hub Median Problem from a Telecommunication Perspective

By

STEPHANIE MILENA ALVAREZ FERNANDEZ

Department of Computer Science, Multimedia and Telecommunications
UNIVERSITAT OBERTA DE CATALUNYA

A thesis submitted in fulfillment of the requirements for the degree of DOCTOR OF PHILOSOPHY IN NETWORK & INFORMATION TECHNOLOGIES

20 DECEMBER, 2018

Thesis committee
Prof. Dr. Daniel Riera i Terrén
Prof. Dr. Jésica de Armas Adrián
Prof. Dr. Daniel Guerreiro e Silva

A mis padres Yadire y Ovani, les debo todo.

E ao Marco, seu amor me sustenta.

- Stephanie

Abstract

Recent advances in the telecommunication industry offer great opportunities to citizens and organizations in a globally-connected world, but they also arise a vast number of complex challenges that decision makers must face. Some of these challenges can be modeled as optimization problems. Examples include the framework of network utility maximization for resource allocation in communication networks, finding a network topology that satisfies certain properties associated with quality of service requirements, overlay multicast networks, and other important features for source to destination communication.

First, this thesis provides a review on how metaheuristics have been used so far to deal with optimization problems associated with telecommunication systems, detecting the main trends and challenges. Particularly the analysis focuses on the network design, routing, and allocation problems. In addition, due to the nature of these challenges, this work discusses how the hybridization of metaheuristics with methodologies such as simulation can be employed to extend the capabilities of metaheuristics when solving stochastic optimization problems.

Then, a popular optimization problem with practical applications to the design of telecommunication networks: the Uncapacitated Single Allocation p-Hub Median Problem (USApHMP) where a fixed number of hubs have unlimited capacity, each non-hub node is allocated to a single hub and the number of hubs is known in advance is analyzed in deterministic and stochastic scenarios. p-hub median problems are concerned with optimilaty of telecommunication and transshipment networks, and seek to minimize the cost of transportation or establishing.

Next, Two immune inspired metaheuristics are proposed to solve the USApHMP, besides that, a two-stage metaheuristic which relies on the combination of biased-randomized techniques with an iterated local search framework and its integration with simulation Monte Carlo technique for solving the same problem is proposed. In order to show their efficiency, a series of computational tests are carried out using small and large size instances from the literature. These results contribute to a deeper understanding of the effectiveness of the

employed metaheuristics for solving the USApHMP in small and large networks. Finally, an illustrative application of the USApHMP is presented as well as some insights about some new possibilities for it, extending the proposed methodology to real-life environments.

Resumen

Los recientes avances en la industria de las telecomunicaciones ofrecen grandes oportunidades para ciudadanos y organizaciones en un mundo globalmente conectado, pero también presentan una gran cantidad de desafíos complejos que son diariamente enfrentados por técnicos e ingenieros. Algunos de estos desafíos se pueden modelar como problemas de optimización. Algunos ejemplos incluyen la asignación de recursos en redes de comunicación, la búsqueda de una topología de red que satisfaga ciertas propiedades asociadas con los requisitos de calidad de servicio, redes de multidifusión superpuestas y otras características importantes para la comunicación de origen a destino.

El primer objetivo de esta tesis es proporcionar una revisión de la literatura de cómo se han utilizado estas técnicas, tradicionalmente, para tratar los problemas de optimización asociados a sistemas de telecomunicaciones, detectando las principales tendencias y desafíos. En particular, el estudio se centra en los problemas de diseño de red, enrutamiento y problemas de asignación de recursos. Debido a la naturaleza de estos problemas, este trabajo también analiza cómo se pueden combinar las técnicas metaheurísticas con metodologías de simulación para ampliar las capacidades de resolver problemas de optimización estocásticos.

En seguida, se trata un popular problema de optimización con aplicaciones prácticas para redes de telecomunicaciones, el problema de la p mediana no capacitado, analizándolo desde escenarios deterministas y estocásticos. Este problema consiste en determinar el número de instalaciones (medianas) en una red, minimizando la suma de todos los costes o distancias desde un punto de demanda a la instalación más cercana. En general, el problema de la p mediana está ligado con la optimización de redes de telecomunicaciones y de transporte, y buscan minimizar el costo de transporte o establecimiento.

Luego, para resolverlo se proponen dos algoritmos inmunológicos y un algoritmo metaheurístico de dos etapas basado en la combinación de técnicas aleatorias sesgadas con un marco de búsqueda local iterado y su integración con la técnica de simulación de Monte Carlo. La eficiencia de los algoritmos se prueba realizando una serie de test computacionales uti-

lizando algunas de las instancias más empleadas en la literatura, obteniendo unos resultados que demostraron el óptimo desempeño de los algoritmos propuestos en instancias pequeñas y grandes al resolverlas en cuestión de segundos y a un bajo costo computacional. Finalmente, se presenta una aplicación ilustrativa del problema de la p mediana, así como algunas ideas sobre nuevas posibilidades para ello, que extienden la metodología propuesta a problemas de la vida real.

Resum

Els últims avenços en la industria de les telecomunicacions ofereixen grans oportunitats per ciutadans i organitzacions en un món globalment connectat, però a la vegada, presenten reptes als que s'enfronten tècnics i enginyers que prenen decisions. Alguns d'aquests reptes es poden modelitzar com problemes d'optimització. Exemples inclouen l'assignació de recursos a les xarxes de comunicació, trobant una topologia de xarxa que satisfà certes propietats associades a requisits de qualitat de servei, xarxes multicast superposades i altres funcions importants per a la comunicació origen a destinació.

El primer objectiu d'aquest treball és proporcionar un revisió de la literatura sobre com s'han utilitzat aquestes tècniques, tradicionalment, per tractar els problemes d'optimització associats a sistemes de telecomunicació, detectant les principals tendències i desafiaments. Particularment, l'estudi es centra en els problemes de disseny de xarxes, enrutament i problemes d'assignació de recursos. Degut a la naturalesa d'aquests problemes, aquest treball també analitza com es poden combinar les tècniques metaheurístiques amb metodologies de simulació per ampliar les capacitats de resoldre problemes d'optimització estocàstics.

A més, es tracta un popular problema d'optimització amb aplicacions pràctiques per xarxes de telecomunicació, el problema de la p mediana no capacitat, analitzant-lo des d'escenaris deterministes i estocàstics. Aquest problema consisteix en determinar el nombre d'instal·lacions (medianes) en una xarxa, minimitzant la suma de tots els costs o distàncies des d'un punt de demanda a la instal·lació més propera. En general, el problema de la p mediana està lligat amb l'optimització de xarxes de telecomunicacions i de transport, i busquen minimitzar el cost de transport o establiment de la xarxa.

Es proposa dos algoritmes immunològics i un algoritme metaheurístic de dues etapes basat en la combinació de tècniques aleatòries amb simulacions Monte Carlo. L'eficiència de les algoritmes es posa a prova mitjançant alguns dels test computacionals més utilitzats a la literatura, obtenint uns resultats molt satisfactoris, ja que es capaç de resoldre casos petits i grans en qüestió de segons i amb un baix cost computacional. Finalment, es presenta

una aplicació il·lustrativa del problema de la p mediana, així com algunes noves idees sobre aquest, que estenen la metodologia proposta a problemes de la vida real.

Resumo

Avanços recentes no setor das telecomunicações oferecem grandes oportunidades para cidadãos e organizações em um mundo globalmente conectado, ao mesmo tempo em que surge um vasto número de desafios complexos que os engenheiros devem enfrentar. Alguns desses desafios podem ser modelados como problemas de otimização. Alguns exemplos incluem o problema de alocação de recursos em redes de comunicações, desenho de topologias de rede que satisfaça determinadas propriedades associadas a requisitos de qualidade de serviço, sobreposição de redes multicast e outros recursos importantes para comunicação de origem a destino.

O primeiro objetivo desta tese é fornecer uma revisão sobre como as metaheurísticas têm sido usadas até agora para lidar com os problemas de otimização associados aos sistemas de telecomunicações, detectando as principais tendências e desafios. Particularmente, a análise enfoca os problemas de desenho, roteamento e alocação de redes. Além disso, devido á natureza desses desafios, o presente trabalho discute como a hibridização de metaheurísticas com metodologias como simulação pode ser empregada para ampliar as capacidades das metaheurísticas na resolução de problemas de otimização estocásticos na indústria de telecomunicações.

Logo, é analisado um problema de otimização com aplicações práticas para redes de telecomunicações: o problema das p medianas não capacitado em que um número fixo de hubs tem capacidade ilimitada, cada nó não-hub é alocado para um único hub e o número de hubs é conhecido de antemão, sendo analisado em cenários determinísticos e estocásticos. Dada a sua variedade e importância prática, o problema das p medianas vem sendo aplicado e estudado em vários contextos.

Seguidamente, propõe-se dois algoritmos imune-inspirados e uma meta-heurística de dois estágios, que se baseia na combinação de técnicas tendenciosas e aleatórias com uma estrutura de pesquisa local iterada, além de sua integração com a técnica de simulação de Monte Carlo para resolver o problema das p medianas. Para demonstrar a eficiência dos

algoritmos, uma série de testes computacionais é realizada, utilizando instâncias de grande porte da literatura. Estes resultados contribuem para uma compreensão mais profunda da eficácia das metaheurísticas empregadas para resolver o problema das p medianas em redes pequenas e grandes. Por último, uma aplicação ilustrativa do problema das p medianas é apresentada, bem como alguns *insights* sobre novas possibilidades para ele, estendendo a metodologia proposta para ambientes da vida real.

Acknowledgements

First of all, thanks to God, for enlightening and guiding me in all moments of my life.

I want to express my gratitude to my family, especially to my parents Yadire and Ovani, whom have sacrificed so much for me to be here today, their lives and marriage are an inspiration to me. Thank you both for giving me the strength to reach the stars and chase my dreams. To my brothers Harol and Giovanni whom are always taking care of me ever since I can remember, thank you for giving me the joy of being the aunt of such incredible children. Also a big thanks to my new family Marco, Marcelita, Marco Antonio and Murilo for all the love and big support during all this time.

I would like to thank Prof. Daniel Guerreiro e Silva, my mentor in this long walk. Thank you for your patience and support during my research. Working with you has been an incredibly rewarding experience and I hope this partnership will last for a long time. I am also deeply grateful to Prof. Angel A. Juan for the incredible opportunity to work under his excellent stewardship at the UOC research group ICSO, for his patience over extensive discussions, for his great support and encouragement. Also, I would like to thank Prof. Daniel Riera who has greatly contributed to this work through his guidance and support. Thank you for being supportive when I felt desperate. In the same way, I would like to thank my co-director Prof. Jésica de Armas, for the availability, accompaniment and for all the corrections that allowed me to always seek higher standards in my work. I would like to mention Daniele Ferone for the confidence and his contribution in this work. And Prof. Daniele Tarchi from the Università di Bologna for his invitation to visit his research group and university.

Moreover, I want to express my gratitude to the members of the ENE Postgraduate Program and to the members of the UOC OSRT department, for their constant performance and collaboration during the development of my doctorate. In the same way, I want to thank the financial support of CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior), which was of vital importance for the accomplishment of this work. This thesis would never have been possible without the support of the UOC Doctoral School, the Internet

Interdisciplinary Institute (IN3) and the Erasmus+ Program which funded me during my stay in Barcelona.

Finally, I want to thank all my friends in Spain, Brazil, Colombia and Germany whom have always tried to cheer me up when I thought I was going to quit. Your friendship makes my life a wonderful experience. And, to the colleagues of ICSO, GPDS and DIGICOMM for their friendship and fellowship, for collaboration and for providing fun times in difficult times.

Table of Contents

Page

List of Tables

List of Figures

Figure **Page**

List of Algorithms

Algorithm | **Page**

Nomenclature

ACO Ant Colony Optimization

AP Australian Post data set

BCO Bee Colony Optimization

BKS Best Known Solution

BLS Breakout Local Search

BR Biased Randomization

CAB Civil Aeronautics Board data set

CL Candidate List

COP Combinatorial Optimization Problem

$DNDR$ Directed Network Design with Relays Problem

DSN Distributed Sensor Network

EA Evolutionary Algorithm

GA Genetic Algorithm

$GRASP$ Greedy Randomized Adaptive Search Procedure

$GVNS$ General Variable Neighborhood Search

HLP Hub Location o Problem

ICT Information and Communication Technologies

ILS Iterated Local Search

$IS-IS$ Intermediate System-Intermediate System

LR Lagrangian Relaxation

LS Local Search

$MANET$ Mobile *Ad hoc* Network

$MCFP$ Multi-Commodity Flow Problem

MRP Multi-cast Routing Problem

MST Minimum-cost Spanning Tree

$NITSS$ Nature Inspired Tool for Sensor Simulation

NSF National Science Foundation

$OFDM$ Orthogonal Frequency Division Multiplexing

$OSPF$ Open Shortest Path First

$PLDs$ Permanent Lightpath Demands

PSO Particle Swarm Optimization

QoS Quality of Service

RC Restricted Clustering

RCL Restricted Candidate List

$RLDs$ Random Lightpath Demands

$RNDP$ Robust Network Design Problem

$RSAP$ Reliable Server Assignment Problem

$RWAP$ Routing and Wavelength Assignment Problem

SA Simulated Annealing

$SATL$ Simulated Annealing Tabu Search

SI Swarm Intelligence

$SLDs$ Scheduled Lightpath Demands

TCO Termite Colony Optimization

TDM Time Division Multiplexing

TS Tabu Search

$VANET$ Vehicular *Ad hoc* Network

VNS Variable Neighborhood Search

$VTDP$ Virtual Topology Design Problem

WDM Wavelegth Division Multiplexing

WMN Wireless Mesh Network

WSN Wireless Sensor Network

USApHMP Uncapacitated Single Assignment p-Hub Median Problem

CHAPTER 1

INTRODUCTION

1.1 Motivation

In today's complex world, there is a visible trend in the implementation of *smart* technologies to city planning and management, leading to greater optimization of time and resources, and resulting in more efficiency. In its usage as an adjective and in the context of modern technology, *smart* implies that following processes of computer programming or guidance, some level of intelligent autonomy or automation in action is involved[1]. In effect, innovative projects comprising information and communication technologies (ICT) have been included in the political agenda of many governments as a key program to enable a vision where municipalities can use technology to meet sustainability goals, boost local economies and improve urban services.

Thus, technological advancements and changes in the social and political sector have had great effect on telecommunication growth. The word telecommunication is derived from the Greek word *tele*, which means over a distance, while *communication* simply means the sharing of information or messages between two or more entities (Sapolsky et al. 2018), putting the two words together gives telecommunication. Telecommunication includes voice, video and internet communication services and telecommunication services involve voice communication, video streaming, graphics and television services at high speed. Telecommu-

[1]Sources: Collins Dictionary, 2017, *Smart*, adj.; Merriam-Webster Dictionary, 2017, *Smart*.

nication services always improve in their quality of connectivity as technologies advances. As effect, in recent years, the world have seen a tremendous growth in mobile telecommunication networks, so much so that the number of mobile phone users are now greater than the number of land-line phone numbers.

As a consequence of the evident increase in mobile communications and services, the world is widely connected. Cellular phone networks, power grids, and the Internet are examples of vital infrastructures that connect everything on this planet. These crucial systems share one striking feature: they are composed of subsystems that make local decisions and coordinate with other subsystems to accomplish their tasks. Such systems can be denominated as networked systems. Networked systems are not limited to telecommunications, they are also present, for example, in modern cars and aircraft where several controllers are working together to make the operations more flexible and easier.

Furthermore, due to the immense appetite for communication and bandwidth, the demand seems to only increase. At the same time, the physical resources are limited, for instance, there is only a limited spectrum available for wireless communications. Therefore, there are financial incentives to use current and future communications hardware and the available resources as efficiently as possible. If it were entirely known which services are going to make use of the network resources in term of duration, location, and required capacity, the resources could of course be used and adapted accordingly. All these form a series of very challenging problems. One of the main aims of when designing communication networks is to transmit signals with the minimum noise, least energy consumption, and optimal transmission quality. Thus, there is a need for designing such large scale systems and networks so that costs and resources are optimized. Hence, in this thesis, decision problems that are posed as optimization problems and where there is a natural communication structure imposed on the disparate parts of the systems are considered. A particular application motivates this thesis: resource location-allocation in communication networks.

An optimization problem is always associated with a set of decision variables, i.e., a set of variables should be chosen such that the solution is feasible and optimal. An important step when approaching optimization problems is to identify the type of problem we are handling with, since algorithms for solving optimization problems are tailored to a particular type of problem. Thus, optimization problems can be divided in continuous and discrete. Models with continuous variables are continuous optimization problems and problems with discrete variables are discrete optimization problems.

Network design problems considered in this thesis can be fitted into combinatorial optimization problems (COPs). A formal definition of COPs can be stated as follows (Aarts et al. 2003).

Definition 1.1. A COP is specified by a set of problem instances and is either a minimization problem or a maximization problem.

Unless stated otherwise, we consider in this chapter only minimization problems.

Definition 1.2. An instance of a COP is a pair $(\mathscr{S}, f)$, where the solution set $\mathscr{S}$ is the set of feasible solutions and the cost function f is a mapping $f : \mathscr{S} \to \mathbb{R}$. The problem is to find a globally optimal solution, i.e., an $s' \in \mathscr{S}$ such that $f(s') \leq f(s)$ for all $s \in \mathscr{S}$. Furthermore, $f' = f(s')$ denotes the optimal cost, and $\mathscr{S}' = \{s \in \mathscr{S} | f(s) = f'\}$ denotes the set of optimal solutions.

The space of possible solutions is typically too large to search exhaustively using pure brute force. In some cases, problems can be solved exactly using Branch and Bound techniques. However in other cases no exact algorithms are feasible, and the implementation of other methodologies such as metaheuristics results more appealing. Several studies have been dedicated to global optimization techniques and most of them are useful for solving COPs (Gray et al. 1997, Mahdavi et al. 2015, Ronellenfitsch and Katifori 2016). When solving these problems, the main objective is to find the best possible solution, from among a usually vast set of feasible combinations. Thus, for instance, we might be interested in: *(i)* minimizing the cost or energy consumption of some activity or system; or *(ii)* maximizing the profit, output, performance, or efficiency of some activity or system. The variables, the constraints and the objective function of a COP define its combinatorial structure. This is a semi-formal way to indicate the main characteristics of the problem that affect the effectiveness of different solutions procedures.

COPs are not limited to networked systems and ICT and can be found anywhere, from logistics and transportation to health care, production, and finance. Although COPs have been studies so many decades ago, they have received special attention in the last decades, mainly due to the strong connection between this type of problems and the problems found in real world and its complexity in the resolution by exact methods. Real-life COPs commonly found in telecommunication systems demands real-time information as is the case of real-time streaming, which means that the transmission must be quickly relayed by all nodes. Besides, those problems are usually NP-hard and large scale, which means that they cannot be solved exclusively using exact methods unless a considerable amount of computing time and efforts are invested. Some examples of COPs belonging to the NP-hard class are the Traveling Salesman Problem (Lawler 1985, Cook 2011) and Quadratic Assignment Problem (Finke et al. 1987, Drezner 2015), among others. In these cases, heuristics and metaheuristics reveal themselves as good alternatives to exact methods, since they can find high-quality or even near-optimal solutions in low computing times.

On the one hand, the term heuristic is derived from the Greek word *heuriskein*, which means "the art of discovering new strategies to solve problems" and is typically employed to describe procedures that make use of the problem characteristics to generate reasonably good solutions sometimes in just a few milliseconds (Romanycia and Pelletier 1985). On the other hand, metaheuristics would be considered as higher level algorithms that coordinate simple heuristics and rules to find near-optimal solutions to COPs, typically after some seconds (Glover and Kochenberger 2006). A review of several metaheuristics applied to COPs is presented by Glover et. al. (Glover and Kochenberger 2006).

Due to the rapid deployment of *smart* technologies, telecommunication industry is one of the main area where metaheuristics are applied and a large number of routing and network design technologies have been developed and updated. The related optimization problems found in the literature cover many types of networks. A generic network design infrastructure and a routing method must ensure that data can travel through the network between arbitrary end points. The resulting network should be able to support some of many scenarios, among others: heavily and loaded networks, traffic patterns, multicast, or point-to-point traffic. Likewise, the rapid growth of telecommunication capacity –driven in part by the wide ranging deployment of optic fiber as well as the expansion of wireless technologies– has led to the increasing concern regarding the traffic characteristics and different quality of service (QoS) demands.

The aforementioned demands that the design of communication networks and routing schemes should guarantee some subset of the following requirements: *(i)* all-time connectivity, providing at the same time the required robustness, high reliability, and availability; *(ii)* efficient energy management and other issues at the lowest possible cost; *(iii)* QoS guarantees or, at least, a high probability of good service; *(iv)* prevention of routing oscillations, loops, and overloaded network situations. In this sense, the use of metaheuristic algorithms in many COPs found in telecommunication area offers an alternative solution.

Although, metaheuristics constitute a powerful tool to tackle complex optimization problems in telecommunication area, some of these methodologies have been developed considering deterministic problems when, real-world communication scenarios are plenty of uncertainty.

Unfortunately, the simplification of scenarios, i.e., assuming no uncertainty, can lead to poor-quality solutions. Simheuristics (Juan et al. 2015*a*) combine metaheuristics and simulation specially designed to tackle COP instances that contain stochastic components. This hybrid method not only can deal with uncertainty and real-time decision making but they can also consider another aspects such as richer objective functions, dynamism, etc.

With this in mind, this thesis presents an analysis of some research challenges related to metaheuristics and its applications to optimization problems in telecommunication networks design issues. Also, it studies and integrates powerful and well-known methodologies such as metaheuristics and simulation techniques and its applications for solving an specific network design problem relevant on the design of communication networks.

1.2 Research Questions

The present thesis intends to answer the following overarching questions, hereafter referred to as the main research questions.

First, the emergence of trends such of new Internet-based applications, video streaming, and content distribution have generated a great demand for the design and development of the network infrastructure. The continued growth of problems in term of size has led researchers to propose alternatives to traditional exact methods to solve complex problems in "real time". In this context, several heuristics have been applied to the problem of designing reliable communication networks. This brings us the next question:

Which are the most used methodologies found in the literature by researchers when solving optimization problems related to telecommunications systems?

Second, the main concern when thinking on telecommunication problems is to provide new fast and efficient methodologies or improve the methodologies already existing that allow decision makers and engineers to find approximate solutions to big problems in a short period of time. This concern led us to the following question:

Are those referred methodologies fast enough and efficient when applied to problems related to the design of the communication networks that engineers and decision makers face?

Finally, new heuristics need to be developed to solve associated problems in telecommunication systems. Especially, those that require an effective solution approach to handle a larger problem in practice in a considerable small amount of time. The point of real-time decisions is that the environment is full of stochastic variables such as weather, energy failure and natural events which usually we have not control of and, today's world demands fast solutions every day, which make us think about the last question:

How the hybridization of metaheuristic-based methodologies with simulation techniques (simheuristics) can help to solve "real world" problems commonly found in telecommunication systems?

Within this context and taking into account the previous questions, this thesis study the application of metaheuristics and simheurisitcs for the Uncapacitated Single Assignment p-Hub Median Problem (USApHMP), this problem have the characteristic that is suitable for modeling real-life telecommunications problems, such as the frequency assignment problem and antenna positioning problem.

1.3 Research Objectives

As already introduced, the main goal of this research is to study the application of metaheuristics and to explore its combination with simulation techniques in order to deal with stochastic version of optimization problems such as the USApHMP which is related to telecommunication systems. In order to reach this goal, the following specific objectives with is corresponding specific research question should be achieved.

I ***Which are the most used methodologies found in the literature by researchers when solving optimization problems related to telecommunications systems?***

  1 Identifying the main challenges related to the implementation of the metaheuristic methodologies in assessing their suitability for using in telecommunication systems.

II ***Are those referred methodologies fast enough and efficient when applied to problems related to the design of the communication networks that engineers and decision makers face?***

  2 Investigating the potential heuristics approaches for a classic problem of combinatorial optimization commonly applied to the design of communication networks, specifically, the USApHMP.

III ***How the hybridization of metaheuristic-based methodologies with simulation techniques (simheuristics) can help to solve "real world" problems commonly found in telecommunication systems?***

3 Designing new and efficient approach that combines metaheuristic methodologies with simulation techniques as a promising strategy for solving the aforementioned problem under uncertainty scenarios.

4 Implementing such algorithms, test them in some data available in literature and compare them with existing ones, concluding on the contribution.

The main contributions of this dissertation will be on one hand, a review and analyses of some of the most common challenges found when solving COPs related to telecommunication systems and, on the other hand, due the importance of p-hub median problems in real world applications, the implementation of metaheuristics and of hybrid techniques such as simheuristics (metaheuristics and simulation) to deal with deterministic and stochastic versions of the USApHMP.

1.4 Research Strategy

Two research strategies have been used in order to structure the thesis. Once we have identified the challenges in the context of telecommunications, in order to fulfill the second, third and fourth objectives, it is applied Design and Creation and Experiment methodologies (Oates 2005).

1.4.1 Design and Creation Strategy

The problem solving approach associated to this strategy employs an iterative procedure that can be summarized in five steps:

- Awareness
  Most of the existing methodologies to solve real-life COPs consider scenarios with a reduced size and supposing all information is known, which is usually a quite unrealistic assumption.

- Suggestion
  The suggestion is to explore the application of metaheuristics to solve an identified challenge related to telecommunication systems: the USApHMP, and to study the application of simheuristics for solving the identified problem in a stochastic scenario.

- Development
  Algorithms for solving the identified problem commonly found in the telecommunication field relying on the application of metaheuristics and simheuristics are designed.

- Evaluation
  Initially, the algorithms are tested. Then, they are used to solve the identified challenge, and a comparison between the obtained results and those from other authors is performed.

- Conclusion
  The quality of the proposed algorithms is assessed by examining the results. Main advantages and disadvantages are described. Finally, results also enable the identification of areas for future research.

1.4.2 Experiment Strategy

Another strategy considered in this thesis is related to experiments. Accordingly, researchers start by developing a theory about their topic of interest, which leads to a statement based on the theory that can be tested empirically via an experiment. In the context of this thesis, after reviewing the literature on metaheuristics applied for solving telecommunication problems, and elaborating a specific proposal (algorithm) for solving an identified problem, the hypothesis associated will be one or several among the following:

- The proposed algorithm should be able to achieve better objective solutions than other state-of-the-art algorithms.

- The proposed algorithm should be able to achieve better or similar objective solutions than other state-of-the-art algorithms, requiring less time.

- The algorithm should be capable of dealing with more rich and realistic challenges and/or even new problems not previously mentioned.

The main algorithm outcomes of interest will be the objective solution and the computational time. However, some other desirable characteristics will be also taken into account: flexible, simplicity, and ease of implement, for instance.

1.5 Thesis Structure

The diagram in Figure 1.1 provides guidance for readers to access the contents of this thesis. A brief description of each chapter is as follows.

This chapter introduced important concepts to the care and justification of this work and presented the motivation, the research questions and objectives and strategy as well as the meaningful of the thesis. Considering the proposal of this thesis, the remainder of this text is organized as follows:

In the first part, Chapter 2, presents how metaheuristics have been gaining attention over the last years and the main areas where have been applied. It also introduces the existing methodology employed throughout the thesis. In particular, metaheuristics and other methodologies employed for solving COPs are presented, describing their context, reviewing the main definitions and classifications.

Supported by the concepts already introduced on the previous chapter, Chapter 3 gives a detailed literature review on how metaheuristics have been used so far to deal with COPs associated with telecommunications systems. Particularly, the analysis focuses on the network design, routing and allocation problems. Through the extensive analysis of the literature, different metaheuristic techniques employed for solving problems are identified as well as different trends are presented regarding the solution of those identified problems.

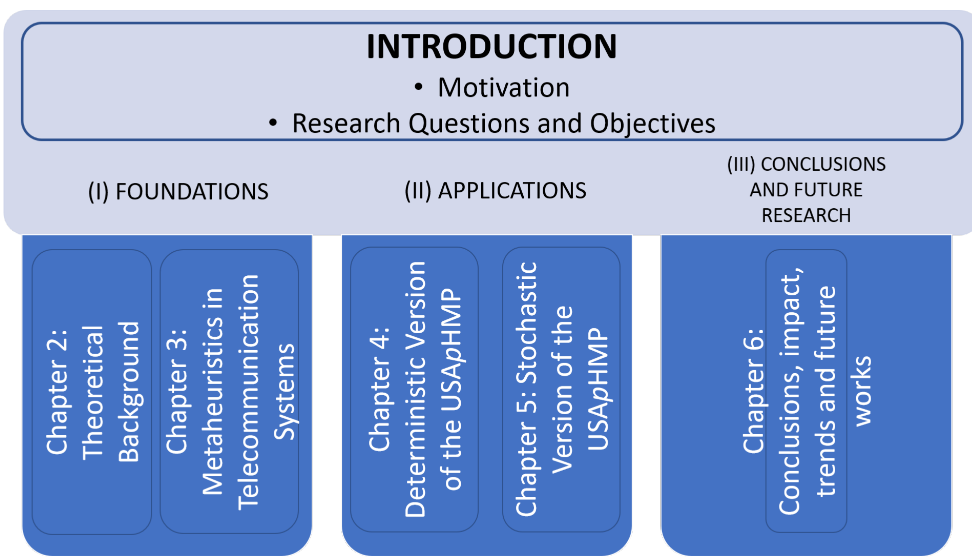


Figure 1.1: The structure of the thesis

In the second part, Chapter 4 presents three different approaches for solving a common optimization problem when considering the design of telecommunication networks: the USApHMP. First, a two-stage metaheuristic based on the combination of biased-randomized technique with an iterated local search framework is presented . Following, two artificial immune systems are employed in order to address the same problem. In all cases computational results that validate the methodology for small and large-size instances from the literature are presented.

Chapter 5 analyzes the USApHMP under uncertainty assumptions with the purpose of extending the methodology presented in the previous chapter to a simheuristic algorithm. Additionally, an extensive numerical experiment is included with the purpose of analyzing the efficiency of the described methodology under uncertainty scenarios.

Finally, in the third part, Chapter 6, presents the final remarks about the work, the publications associated to the thesis as well as the research impact and future perspectives. The chapter discusses how the hybridization of metaheuristics with methodologies such as simulation and machine learning can be employed to extend the capabilities of metaheuristics when solving stochastic and dynamic COPs in telecommunications.

(I) FOUNDATIONS

Chapter 2

Theoretical Background

Usually, when designing COP algorithms, one can choose between complete or incomplete algorithms. Complete algorithms are guaranteed to find for every finite size instance of a COP an optimal solution in bounded time, however, for NP-hard problems, complete methods might need exponential computational time in the worst case. In incomplete methods, such as metaheuristics, they may fail to find an optimal solution but in compensation, such methods are able to find "good" solutions in a significantly reduced amount of time. The last fact makes these methods effective solution strategies for solving COPs found in a very large variety of areas and situations.

Particularly, real-life telecommunication systems are subject to a large set of environmental conditions, and many factors and parameters may affect the performance of these systems. Usually, processes in telecommunication systems can be slow and expensive. Thus, any savings in terms of time and resources will make network designs more sustainable. One important issue to be addressed while designing the right optimization algorithm for solving a COP in the telecommunication field is the balance between the algorithm's performance and the computational time it requires.

Since there are so many optimization problems in the field of telecommunication systems and due the constant necessity for greater optimization of time and resources, it is not surprising that the use of metaheuristics has been gaining attention over the last years. This is evidenced in Figure 2.1, which shows a time series chart of the number of indexed journal

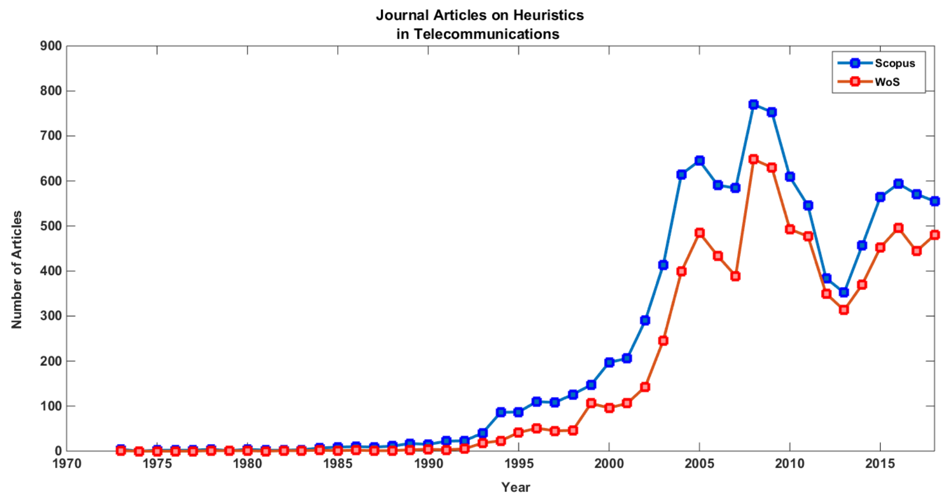


Figure 2.1: Time series of journal articles indexed in Scopus and Web of Science

articles (both in Scopus and in Web of Science) during the last decades. Notice the increase in the number of published articles since the early 2000s.

The data for the previous time series was obtained by running a search procedure including the following logical condition: *((*heuristic) OR "genetic algorithm") AND telecommunication)*. Note that, the term *genetic algorithm* is included in the logical condition due to the vast amount of works proposing such algorithms for solving optimization problems in telecommunications and related areas. Using the same search criteria, Figure 2.2 shows a Pareto chart of Scopus-indexed articles by subject area. Here, it is clear that the areas of Computer Science and Engineering are the predominant ones in terms of number of publications related to applications of heuristic optimization in telecommunication systems. Finally, Table 2.1 summarizes the number of publications on these topics in each Scopus-indexed journal. Observe that most of these journals belong to the areas of Telecommunication (including wireless and computer networks), Computer Science, and Operations Research.

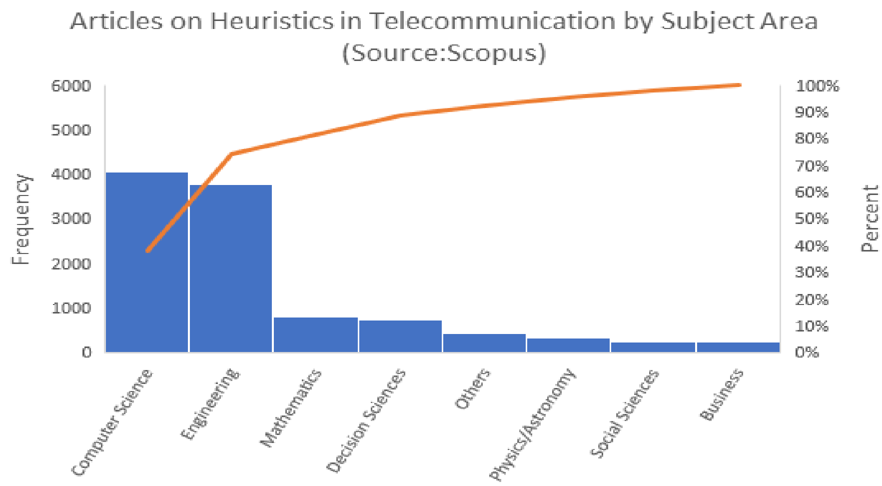


Figure 2.2: Pareto chart for Scopus-indexed articles by subject area

A huge number of optimization algorithms have been devised through the years, and the idea is not to make any attempt at all to provide a survey here. However, some of the most standard algorithms for solving optimization problems that are relevant for this thesis are presented. With this in mind, this chapter summarizes the theoretical background of metaheuristics algorithms commonly found when solving COPs in telecommunications. Additionally, a detailed description of the main metaheuristics and techniques applied in the latter chapters, including, Iterated Local Search, Artificial Immune Systems and Biased Randomization is included. Despite here is presented a relatively small group, a vast variety of metaheuristic methodologies exist, especially through hybridization. Thus, an insight of hybrid metaheuristics is also presented. Finally, this chapter outlines simheuristics as general simulation-optimization approach to solve complex COPs under uncertainty.

Table 2.1: Articles on heuristics in telecommunication by journal

Journal	**Articles**
Computer Communications	121
Computer Networks	100
European Journal Of Operational Research	88
IEEE ACM Transactions On Networking	85
IEEE Journal On Selected Areas In Communications	85
IEEE Transactions On Vehicular Technology	73
IEEE Transactions On Wireless Communications	72
Computers And Operations Research	61
IEICE Transactions On Communications	58
Photonic Network Communications	54
IEEE Transactions On Communications	46
Journal Of Lightwave Technology	44
Wireless Networks	43
IEEE Transactions On Mobile Computing	41
Journal Of China Universities Of Posts And Telecommunications	41
Telecommunication Systems	41
IEEE Transactions On Parallel And Distributed Systems	37
Wireless Personal Communications	36
Ad Hoc Networks	32
IEEE Communications Letters	31
Applied Soft Computing Journal	30
Eurasip Journal On Wireless Communications And Networking	30
Expert Systems With Applications	30
Networks	30
Journal Of Beijing University Of Posts And Telecommunications	29
Wireless Communications And Mobile Computing	29
Electronics Letters	28
International Journal Of Communication Systems	27
Journal Of Heuristics	25
Journal Of Optical Communications And Networking	20
Iet Communications	19
Operations Research	18
Ruan Jian Xue Bao Journal Of Software	18
IEEE Transactions On Computers	17
Journal Of Parallel And Distributed Computing	17
Information Sciences	16
Journal Of The Operational Research Society	16
Mobile Networks And Applications	15
Optical Switching And Networking	15
IEEE Transactions On Reliability	14

2.1 A Brief Overview on Metaheuristics

Metaheuristics can be defined as general high-level methodologies that can be used as guiding strategies in the design of heuristics to solve specific optimization problems (Talbi 2009). The term *metaheuristic* was first introduced by Glover (1986). Exact methods are guaranteed to find an optimal solution and to prove its optimality for every instance of a COP. The run-time, however, often increases dramatically with a problem instance's size, and often only small or moderately-sized instances can be practically solved to proven optimality. Thus, for larger instances, one attractive possibility is usually to turn to metaheuristic algorithms that trade optimality for run-time since, they allow the resolution of large-scale instances of a specific problem by providing satisfactory solutions in a reasonable execution time. Metaheuristics offer many advantages over conventional algorithms (Konak and Smith 1999, Yang et al. 2013). They are among the most promising and successful techniques and have gained popularity in the last few decades, with many conferences, journals and books especially dedicated to them. Blum and Roli (2003) outline nine properties of metaheuristics, as follows:

- Metaheuristics are strategies that guide a search process.
- The goal is to efficiently explore the search space in order to find (near-)optimal solutions.
- Techniques which constitute metaheuristic algorithms range from simple local search procedures to complex learning processes.
- Metaheuristic algorithms are approximate.
- They may incorporate mechanisms to avoid getting trapped in confined areas of the search space.
- The basic concepts of metaheuristics permit an abstract level description.
- Metaheuristics are not problem-specific
- Metaheuristics may make use of domain-specific knowledge in the form of heuristics that are controlled by the upper level strategy.

Metaheuristics are particularly suited for solving highly complex optimization problems including many decision variables with non-linear interactions among them, multiple objectives and/or constraints, and ill-understood structures. Many classification criteria have been

Algorithm 1: LocalSearch

Generate an initial solution s_o
$s \leftarrow s_o$
$N \leftarrow$ neighborhood operator
while *(Stop condition is not met)* **do**
  $s' \leftarrow$ Improve($N(s)$) such that $f(s') \leq f(s)$
  $s \leftarrow s'$
end
Return s

proposed to differentiate metaheuristics (Talbi 2009). In our case we employ the classification that label them as single-solution metaheuristics –which maintain a single solution–, population-based metaheuristics –which work with a set of individual solutions– and hybrid metaheuristics. The most important commonly employed are highlighted next.

2.1.1 Single-Solution Metaheuristics

These metaheuristics operate with a single solution per iteration and one or more neighborhood structures. Algorithms working on single solutions are also known as trajectory methods, which consider a single search point at a time during the search process. Single-solution metaheuristics focus on exploitation (intensification), applying local search within a limited region. Local Search procedures (LS) are optimization methods that maintain a solution, known as current solution, and explore the search space by steps within its neighborhood. They usually go from the current solution to a better close solution, which is used, in the next iteration, as current solution. This process is repeated until a stop condition is fulfilled, e.g. there is no better solution within the neighborhood of the current solution.

The pseudo-code of a basic LS is detailed in Algorithm 1. Given a finite set of neighborhood structures $\mathcal{N}_k (k = 1, \cdots, k_{max})$, with $\mathcal{N}_k(s)$ the set of solutions in the $k-$th neighborhood of $s \in \mathcal{S}$, and an objective function f, the LS works as follows. If a better solution $s' \in \mathcal{N}(s)$ is found, s' becomes the new current solution and the neighborhood search starts again. If no further improvement can be made, then, a local or global optimum which minimizes f has been found. Most local search heuristics use only one neighborhood structure, i.e., $k_{max} = 1$. This being said, the $k-$th neighborhood in the LS described below is ignored.

Particularly, in the area of telecommunication systems, the most frequently-used single-solution metaheuristics are Tabu Search, Iterated Local Search, Greedy Randomized Adaptive

Search Procedure, Variable Neighborhood Search and Simulated Annealing. Following, we present the basis of those methods. Additionally, a detailed description of Biased Randomization is presented, which serve as underlying engine of a developed methodology presented in Chapter 4. Despite the methodologies presented here seem relatively simple, many state-of-the-art optimization methods are based on them.

Tabu Search

Tabu Search (TS) is a local search procedure that also includes a "tabu" list of recently-visited solutions to increase efficiency (Glover 1997). Contrary to memoryless metaheuristics, which make use of the search experience (memory, in the widest sense) to influence the future search direction (Birattari et al. 2001, Arin and Rabadi 2013), TS makes use of flexible and adaptive memory designs.

Tabus are one of the distinctive elements of TS when compared to LS. This procedure uses a selection rule based on the concept of *best improvement*, where the best solution in the neighborhood is chosen to replace the current reference solution. This acceptance criterion allows the search to avoid getting trapped into a local optimum, although a cyclic search could happen. To avoid this undesirable behavior, recently-visited solutions are forbidden from being visited during a certain number of iterations, thus avoiding cyclical paths in the neighborhood set.

The basic steps of a simple TS heuristic for minimization are described in Algorithm 2. It begins in the same way as ordinary LS, proceeding iteratively from one point (s) to another until chosen termination criteria is satisfied. Each solution s has associated neighborhood $N(s)$, and each solution s' is reached from s by an operation called "move". To carry out the exploration process, recently visited solutions should be avoided. To this aim a tabu list T is maintained. Therefore once a solution is visited, the move from which it was obtained is considered tabu. A more detailed explanation of TS can be found in the works presented by Glover (1989, 1990).

The TS procedure above described is certainly an effective approach for solving hard COPs. However, in most cases, additional elements have to be included in the search strategy to make it full effective. The most important of these are intensification, diversification, allowing infeasible solutions, surrogate and auxiliary objectives.

Algorithm 2: TabuSearch

```
Generate an initial solution s_o
s ← s_o
s' ← s_o
f(s') ← f(s_o)
T ← [ ] // the tabu list is null

while (Stop condition is not met) do
    select the best candidate s
    if f(s) < f(s') then
        f(s') ← f(s)
        s' ← s
    end
    update T
end
Return s'
```

Iterated Local Search

The Iterated Local Search (ILS) (Lourenço et al. 2010) is another possible approach for solving COPs in the area of telecommunications. ILS algorithm is based on the so called Hill Climbing algorithm (Langley et al. 1987). It tries to search for local optima but in a more intelligent way by using the hill climbing technique. ILS explores a sequence of solutions created as perturbations of the current best solution, the result of which is refined using an embedded heuristic.

In practice, the ILS algorithm consists of a sequence of local queries interspersed by perturbations. Its pseudo-code is given by Algorithm 3. Let f be the cost function on our COP; f is to be minimized. For a given input s, it always outputs the same solution s' whose cost is less than or equal to $f(s)$. Given the current s, we first apply a change or perturbation that leads to an intermediate state s', which belongs to $\mathscr{S}$. Then LocalSearch is applied to s' and we reach a solution s'^* in $\mathscr{S}^*$. If s'^* passes an acceptance test, it becomes the next element of the walk in $\mathscr{S}^*$; otherwise, we return to s. Perturbations are functions that modify the solution in order to generate neighboring solutions that are not so close that they can fall into the same local optimum nor as far away as they can behave as random, generating a traditional local search from a new initial random solution. The stopping criterion is defined by the number of iterations without improvement. When a large sequence of iterations fails

Algorithm 3: IteratedLocalSearch

```
Generate an initial solution s_o
s ← LocalSearch(s_o)
while (Stop condition is not met) do
    s' ← Perturbate(s)
    s'* ← LocalSearch(s')
end
Return s'*
```

to reduce the cost of the solution, the algorithm is interrupted and returns the best solution obtained during its execution.

Greedy Randomized Adaptive Search Procedure

The Greedy Randomized Adaptive Search Procedure (GRASP) (Feo and Resende 1995) is another single-location metaheuristic, originally proposed for the Operations Research practitioners and have been employed in the optimization of telecommunication systems. It is an iterative procedure, where each iteration consists of two phases: construction and local search. The objective of GRASP is to repeatedly sample stochastically-greedy solutions, and then use a local search to refine them to a local optimum. The procedure is centered on a stochastic and greedy step-wise construction mechanism, which drives the selection of the components of a solution based on a given sorting criteria.

Algorithm 4: GreedyRandomizedAdaptiveSearchProcedure

```
Generate an initial solution s_o
s ← LocalSearch(s_o)
while (Stop condition is not met) do
    s ← GreedyRandomizedConstruction(s) // construction phase

    s' ← LocalSearch(s)
    if f(s') < f(s) then
        f(s') ← f(s)
        s ← s'
    end
end
Return s'
```

The basic steps of this procedure are shown in Algorithm 4. First a candidate solution s is generated using a randomised construction search procedure. Then, a local search procedure is applied, yielding an improved candidate solution s'. This two-phase process is iterated until a termination condition is satisfied.

In the construction phase, a feasible solution is iteratively constructed, one element at a time. At each construction iteration, the choice of the next element to be added is determined by ordering all elements in a restricted candidate list (RCL) formed by the best elements, i.e. those whose incorporation to the current partial solution results in the smallest incremental cost (this is the greedy aspect of the algorithm). The element to be incorporated into the partial solution is randomly selected from those in the RCL (this is the probabilistic aspect of the algorithm) according to a uniform distribution. Once the selected element is incorporated into the partial solution, the candidate list is updated and the incremental costs are reevaluated (this is the adaptive aspect of the heuristic). The above steps are repeated while there exists at least one candidate element.

In the search phase, a local search is implemented to try to improve the performance of the current solution by searching better solutions in its neighborhood. Once the termination condition is met, the best overall solution is kept as the result. The effectiveness of this procedure depends of some aspects, such as the initial solution. The construction phase plays an important role with respect to this aspect. Therefore, it is important to construct a number of good initial solutions before employing it.

Variable Neighborhood Search

The Variable Neighborhood Search (VNS) (Mladenović and Hansen 1997, Hansen and Mladenović 1999, 2002, Hansen et al. 2010) is a local search method that consists of exploring the solution space through systematic exchanges of neighborhood structures. The VNS exploits systematically the idea of neighborhood change, both in the descent to local optima and in the escape from the valleys which contain them. Contrary to other metaheuristics based on local search methods, the VNS method does not follow a trajectory, but instead explores increasingly different neighborhoods of the current solution.

The method also includes a local search procedure to be applied over the current solution. This local search routine can also use different neighborhood structures. The pseudo-code of the metaheuristic is presented in Algorithm 5. Additional details can be found in Mladenović and Hansen (1997).

Algorithm 5: VariableNeighborhoodSearch

```
Generate an initial solution s_o
s ← s_o
k ← 1
while (Stop condition is not met) do
    s' ∈ N^(k)(s)
    s'* ← LocalSearch(s')
    if f(s'*) ≤ f(s') then
        s ← s'*
        k ← 1
    else
        k ← k + 1
    end
end
Return s
```

Let us denote $N^{(k)}(k = 1, \cdots, k_{max})$, a finite set of pre-selected neighborhood, in this metaheuristic, part of a solution is chosen and, at each iteration, s' is randomly selected within the neighborhood $N^{(k)}(s)$ of the current solution s. This neighbor is then subjected to a local search procedure. If $s^{'*}$ is better than the current solution, the search continues from $s^{'*}$ starting from the first neighborhood structure $N^{(1)}(s)$. Otherwise, the search continues from the next neighborhood structure $N^{(k+1)}(s)$. This procedure is terminated when a stop condition is reached, such as the maximum allowed CPU time, the maximum number of iterations, or maximum number of consecutive iterations without improvements.

Simulated Annealing

Simulating Annealing (SA) is another single-solution metaheuristic. It has its origin in the analogy between the physical process of cooling a metal in a fusion state and an optimization problem. Based on ideas of statistical mechanics proposed by Metropolis et al. (1953), and simulation techniques. SA was initially presented as a combinatorial optimization technique by Kirkpatrick et al. (1983), who used it in the design of electronic systems.

SA consists of first "melting" the system to be optimized at an elevated temperature and then reducing the temperature until the system "freezes" and no improvement in the value of the objective function occurs. The temperature sequence and the number of rearrangements x_o attempted at each temperature for equilibrium represents the SA annealing scheme. The pseudo-code of SA is presented in Algorithm 6. In this algorithm, for each neighbor s' of s, $\Delta = (f(s') - f(s))$ is calculated. In Case $\Delta < 0$, the algorithm makes the solution s' the current solution, because there was an improvement. If $\Delta \geq 0$, the solution s' can be accepted with a probability of acceptance $x \in [0,1)$. T is a parameter of the method, called temperature which regulates the probability of acceptance of solutions with worse cost than the current solution and α is the Boltzmann constant.

Algorithm 6: SimulatedAnnealing

```
Generate an initial solution s_o
s ← s_o
s* ← s_o
T ← T_o // selecting an initial temperature

while (T > 0) do
    while (Stop condition is not met) do
        Built s' ∈ N(s)
        Δ = f(s') − f(s)
        if (Δ < 0) then
            s ← s'
            if f(s') < f(s*) then
                s* ← s'
            end
        else
            x ∈ [0,1)
            if x < e^(−Δ/T) then
                s ← s'
            end
        end
        T ← α × T
        s ← s*
        Update Solution
    end
end
```

Each configuration of a solution in the search space represents a different internal energy of the system. Heating the system results in a relaxation of the acceptance criteria of the samples taken from the search space. As the system is cooled, the acceptance criteria of samples is narrowed to focus on improving movements. Once the system has cooled, the configuration will represent a sample at or close to a global optimum. The advantage of this technique is its property of using a descent strategy, but allowing random ascending movements, thus, avoiding local optima.

Biased Randomization Techniques

Any constructive heuristic can be seen as an iterative greedy procedure, which builds a feasible good solution to the problem at hand by selecting, at each iteration, the best option from a list, sorted according to some logical criterion. Biased randomization (BR) techniques refer to the introduction of randomization in the construction phase and/or neighborhood search of optimization algorithms (Grasas et al. 2017). The application of BR techniques guides the search process by selecting a candidate other than the next option. BR makes use of probability distributions, other than uniform, which do not distribute probabilities in a symmetric shape but in a non-symmetric or skewed one. An example of sorted solution elements can be seen in Figure 2.3. Each potentially eligible element of the candidate list is represented on the x-axis. They are ranked according to some criteria (e.g., priority rule, heuristic value), which defines the selection probability according to some theoretical skewed probability function. The specific distribution to employ will depend upon the specific COP being considered. Some candidate distributions to be considered are the geometric and a discrete version of the descent triangular. These distributions can also be used to induce biased randomness into an algorithm.

Algorithm 7 describes the steps required to implement biased randomization, where CL is a set of all elements potentially eligible. First, a constructive heuristic is selected, once the base heuristic is selected, the algorithm should follow some kind of iterative process. At each round of this iterative process, a new complete solution is generated. For the construction of this solution, the base heuristic is randomized applying a non-symmetric probability distribution to the elements of the CL. Optionally, a local search process can be added to the algorithm in order to improve the solution provided at each round of the iterative process.

As mentioned before, GRASP proposes to consider a restricted list and then apply a uniform randomization in the order the elements of the RCL are selected. BR techniques goes one step further, and instead of restricting the list of candidates, it assigns different probabilities of being selected to each potential movement in the sorted list. In this way the elements at the top of the list, but potentially all elements could be selected.

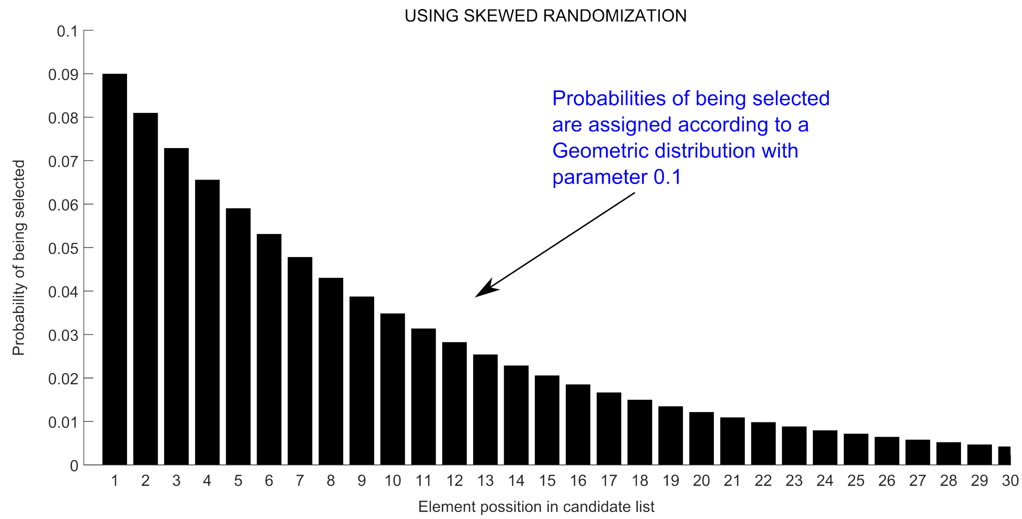


Figure 2.3: Use of Geometric distribution in Biased Randomization. Retrieved from Grasas et al. (2017)

Algorithm 7: BiasedRandomizationProcedure

Generate an initial solution s_o
$s \leftarrow s_o$
CL $\leftarrow$ initialize candidate set
$\mu \leftarrow$ get random number uniformly distributed in $[0,1)$ given a specific seed
$\rho \leftarrow$ get random number from a distribution $Dis(parameter, \mu)$
while *(Stop condition is not met)* **do**
    $l \leftarrow$ get the ρ element of the CL `// construction phase`

    $s \leftarrow s \cup \{\text{CL}\}[l]$
    CL $\leftarrow$ CL $\{\text{CL}\}[l]$
    Reorder CL
    LocalSearch(s)
end
Return s

2.1.2 Population-Based Metaheuristics

The popularity of population-based metaheuristics has grown rapidly among both the scientific community and practitioners. Research fields in which they are commonly and highly successfully employed include logistics and transportation, computer vision, cryptography and telecommunications.

Particularly, due to the increasing complexity of communication infrastructures, many problems presented in the telecommunication area have become extremely complex. Such problems have to deal with a variety of complicated side constraints, for instance, the presence of noise, multiple objectives, dynamically changing parameters, large solution space and others, that in many cases cannot be handled effectively by the existing single-solution based optimization techniques.

Population-based metaheuristics operate with a set of solutions per iteration, and may have different denominations, such as colony, cloud, swarm or even population, depending on the case. They make use of learning factors as they attempt to understand the correlation between the design variables in order to identify the regions of the search space in high-quality solutions.

Most of the population-based algorithms are nature-inspired. Natural computing algorithms is a process of extracting ideas from nature to develop artificial computational systems, or using natural media to perform computation. They are also referred as nature-inspired algorithms or clever algorithms. They have proved themselves to provide optimal solutions in reasonable time for a broad range of optimization problems (Yang 2014).

According to the No Free Lunch theorem (Wolpert and Macready 1997), not all optimization problems can be solved by a single algorithm. As a result, different researchers have developed different natural computing algorithms and their variations. Natural computing algorithms encompass a variety of Evolutionary Algorithms (EAs) that share a common underlying idea of survival-of-the-fittest.

Nature-inspired algorithms are usually well-suited for applications like planning, design, control, classification and clustering, time series modeling, music composing, etc. Unlike most optimization techniques, these algorithms maintain a population of tentative solutions that are manipulated competitively by applying some variation operators to find a global optimum (Brabazon et al. 2015). The following subsections describe the main natural computing paradigms that have being identified as the most applied for solving COPs related to the design of communication networks and other related problems.

Genetic Algorithms

The most common evolutionary computing implementations are Genetic Algorithms (GAs), which model genetic evolution (Goldberg 2006). GA is inspired by Darwin's theory of evolution.

It mimics the process of natural selection for survival of the fittest individual. GA applies various operators such as selection, crossover and mutation.

Therefore, a GA is based on the generation of an initial population, which is evolved by means of the application of evolutionary mechanisms implemented in a loop process. Algorithm 8 describes the pseudo-code for the GA. The initialization phase generates the individuals and it estimates the grade of adaptability to the environment, that is, it calculates the fitness function value of each individual. Once the initialization phase ends, the process enters into the evolutionary loop, where the evolution operators are applied to the individuals in a iterative way until a stop condition is met.

In the case of GAs, a population of strings is used, and these strings are often referred to in the GA literature as chromosomes. The recombination of strings is carried out using simple analogies of genetic crossover and mutation, and the search is guided by the results of evaluating the objective function for each string in the population. Based on this evaluation, strings that have higher fitness (i.e., represent better solutions) can be identified, and these are given more opportunities to breed. Crossover is a matter of replacing some of the genes in one parent by corresponding genes of the other. The other common operator is mutation in which a gene is chosen randomly and the value of the chose gene is changed.

Swarm Intelligence

Another bioinspired approach is Swarm Intelligence (SI). The term is used to describe any attempt to design algorithms or problem-solving devices inspired by the collective behavior of social organisms, from insect colonies to human societies. SI has two main areas: algorithms

Algorithm 8: GeneticAlgorithm

$S \leftarrow$ Generate an initial population of chromosomes
fitness(S)
while *(Stop condition is not met)* **do**
  $S_p \leftarrow$ selectParents(S)
  $S_r \leftarrow$ reproduction(S_p)
  crossover(S_r)
  mutate(S_r)
  fitness(S_r)
  Update Solution
end

based on the collective behavior of social insects –such as Ant Colony (ACO) (Dorigo and Birattari 2011), Termite Colony (TCO)(Hedayatzadeh et al. 2010) or Bee Colony (BCO) (Karaboga and Basturk 2007) Optimization–, and algorithms based on cultures of socio-cognition as the Particle Swarm Optimization (PSO) (Kennedy and Shi 2001).

On one hand, ACO is inspired by a foraging behavior of real ants. Ants posses a natural ability to find the shortest tour between the food source and their nest. If a problem can be converted to a graph, ACO can be applied to find the optimal solution. On the other hand, BCO is inspired by the behavior of the social insect population. A colony of honey bees can extend itself over long distances and in multiple directions simultaneously to exploit a large number of food sources. A colony prospers by deploying its foragers to good fields.

Additionally, TCO is inspired from intelligent behaviors of termites. Termites moves randomly in the search space, but their trajectories are biased towards regions with more pheromones. Finally, PSO is inspired by the flocking behavior of birds. It applies an iterative approach, where, in each iteration, a candidate solution (referred as particle) is moved around the search space based on its position and velocity.

The Algorithm 9 shows the basic structure of SI procedures. In this algorithm, S denotes the population of individuals. These individuals are not necessarily solutions to the considered problem. They may be partial solutions, or set of solutions, or any object which can be transformed into one or more solutions in a structured way. Then, at each iteration of the algorithm, the following three major operations are performed. First, a set of individuals S' are selected from the current population S by applying an evaluation function. Second, a population S'' is generated from S' by the application of a construction phase. Finally, the current population is evaluated. The process is repeated until certain termination criteria are satisfied.

Algorithm 9: SwarmIntelligenceAlgorithm

$S \leftarrow$ Generate an initial population of individuals
while *(Stop condition is not met)* **do**
  $S' \leftarrow$ evaluate(S)
  $S'' \leftarrow$ constructionPhase(S')
  $S'' \leftarrow$ evaluate(S'')
end
Return best individual s''

Artificial Immune Systems

Artificial Immune Systems (AIS) (De Castro and Timmis 2002) are algorithms and systems that use the human immune system as inspiration. The human immune system is robust, error tolerant and extremely adaptive. Such properties are highly desirable for the development of novel computer systems. All AIS algorithms mimic the behavior and properties of immunological cells, specifically B-cells (a particular type of lymphocyte, white blood cell), T-cells (a type of white blood cell that plays a central role in cell-mediated immunity) and dendritic cells (DCs), but the resultant algorithms exhibit different levels of complexity and can perform a wide range of tasks.

A general procedure of AIS is shown in Algorithm 10. At each step (iteration) an antibody's concentration is increased by an amount dependent on its matching to each antigen. In absence of matching, an antibody's concentration will slowly decrease over time. An important characteristic of AIS is its memory-based detection system which is based on the adaptive response of the natural immune systems that enables it to learn protein structures that characterize pathogens it encounters, and "remember" those structures so that future responses to the same pathogens will be very rapid and efficient.

Unlike some other bio-inspired techniques, such as genetic algorithms, the field of AIS encompasses a spectrum of algorithms that exist because different algorithms implement different properties of different cells. Modern AIS are inspired by one of three sub-fields:

Clonal selection: The theory suggests that starting with an initial repertoire of general immune cells, the system is able to change itself (the compositions and densities of cells and their receptors) in response to experience with the environment. The information processing principles of the clonal selection theory describe a general learning strategy. This strategy

Algorithm 10: ArtificialImmuneSystem

Ab ← Generate an initial population of antibodies
Solve $fit \leftarrow$ affinity(Ab)
while *(Stop condition is not met)* **do**
  $C \leftarrow$ clone(Ab)
  $C^* \leftarrow$ apply genetic operator to (C,fit)
  $fit' \leftarrow$ affinity(C^*)
  Ab ← select(C^*,fit')
end
Return Ab

involves a population of adaptive information units (each representing a problem-solution or component) subjected to a competitive processes for selection, which together with the resultant duplication and variation ultimately improves the adaptive fit of the information units to their environment.

Negative selection: The focus of the negative selection algorithm is on anomaly detection problems such as computer and network intrusion detection. The information processing principles of the self-nonself discrimination process via negative selection are that of a anomaly and change detection systems that model the anticipation of variation from what is known. The principle is achieved by building a model of changes, anomalies or unknown data by generating patterns that do not match an existing corpus of available patterns.

Immune network algorithms: The objective of the immune network process is to prepare a repertoire of discrete pattern detectors for a given problem domain, where better performing cells suppress low-affinity (similar) cells in the network. This principle is achieved through an interactive process of exposing the population to external information.

2.1.3 Hybrid Metaheuristics

Over the last few years, an expressive number of algorithms that do not follow purely the paradigm of traditional metaheuristics have been reported. This has been done, specially through hybridization of metaheuristics with other methodologies. The motivation behind implementing hybrid algorithms is usually to obtain better performance approaches that take advantages of each of the single strategies (Raidl 2006). In fact, choosing an adequate combination of multiple algorithmic concepts is often the key for achieving state-of-the-art performance in solving most difficult COPs.

One can distinguish three types of hybridization following Blum's taxonomy (Blum et al. 2011): The first consists of introducing concepts or strategies from either class of algorithms into the other. For example, some population-based metaheuristics make use of this kind of hybridization by incorporating local search procedures. The second is based on the exchange of information of two or more algorithms which are typically run in parallel, helping each other to cover the search space. Finally, the last approach integrates metaheuristics with exact methods (Boschetti et al. 2009), simulation (Juan et al. 2015*a*), and even machine learning techniques (Calvet et al. 2017).

Hybridization of metaheuristics has got an important role for optimization, since a hybrid algorithm often presents a more efficient performance than algorithms based on plain metaheuristics. This is because when hybridizing, the most promising characteristics of each metaheuristic are combined, strengthening the strategy as a whole. In fact, the idea of hybridizing metaheuristics is not new, but dates back to the origins of metaheuristics themselves. Another different taxonomy with various types of hybrid metaheuristics is presented by Talbi (2002) and Raidl (2006). Following Talbi's hybrid metaheuristics classifications, one can distinguish two different groups illustrated in Figure 2.4: the hierarchical and the flat hybrid metaheuristics.

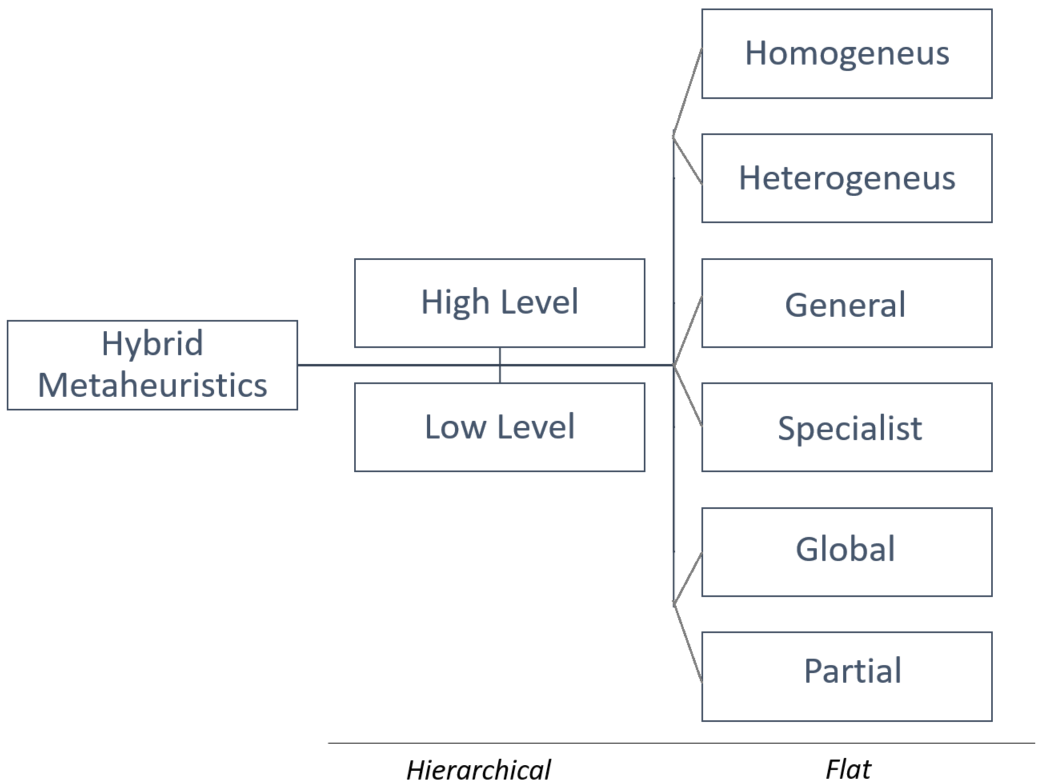


Figure 2.4: Classification of hybrids metaheuristics following Talbi's taxonomy (Talbi 2002)

Hierarchical Hybrid Metaheuristics

The hierarchical component captures the structure of the algorithm implementation. In high-level combinations, at first the algorithms identities are maintained and there is a well-defined form of cooperation between them, but there is no strong direct relationship between the internal mechanisms of the algorithms. The algorithms that participate in low-level combinations strongly depend on each other, since individual component or function of a metaheuristic is replaced from components of the other metaheuristic.

Flat Hybrid Metaheuristics

The flat component specifies the features involved in the algorithm and several dichotomies are defined:

- Homogeneous vs Heterogeneous: The first involves the use of different instances of the same algorithm, while the second involves different algorithms.
- General vs Specialist: All the hybridization mentioned in the hierarchical classes are considered as general, since all the algorithms solve the same optimization problem. Specialist hybridization combine algorithms that solve different problems.
- Global vs Partial: In the global form, all algorithms search the entire search space, that is, they all try to solve the optimization problem globally. In partial mode each algorithm searches a different portion of the search space and can provide a local solution.

The hybrid metaheuristics community has now became notorious, and has its own set of scientific events and journals such as the *Workshop on Hybrid Metaheuristics* (Blesa et al. 2016) and the *International Journal of Applied Metaheuristics Computing*. Moreover, many well-established hybrid search techniques, that can benefit from the complementary capabilities of a wide range of algorithms, have been developed, such as the incorporation of exact algorithms in metaheuristics or viceversa. For a survey dedicated to combinations of metaheuristics with integer linear programming techniques see (Raidl and Puchinger 2008), for an overview on combinations of local search methods with constraint programming see (Focacci et al. 2003), and for a review on combinations of local search methods with exact techniques see (Dumitrescu and Stützle 2003).

2.2 Optimization under Uncertainty and Simheuristics

Despite the fact that metaheuristics are able to efficiently solve large-scale COPs in short computing times, they frequently assume that the problem inputs, the underlying objective function(s), and the set of optimization constraints are deterministic. However, real-life is plenty of uncertainty, which often makes deterministic models to be oversimplified versions of the real stochastic problems faced by decision makers. There are several consolidated methodologies to deal with optimization problems under uncertainties. Among the most used are: sensitivity analysis, stochastic programming and robust optimization.

Sensitivity analysis (Gal 1997) is well known for its application on financial modeling and have be extended to a wide range of fields. Many attempts are made to investigate the problem's behavior when the input data changes. Sensitivity analysis is carried out after the optimum solution of a given problem is obtained. In most problems the variations occur in the right side of the constraints and/or the objective function coefficients, a new constraint may be added, constraints may not be rigid or data are not known exactly. The objective of sensitivity analysis is to find a new optimal solution for a given problem when some of the problem data changes without resolving the problem from scratch.

Stochastic programming (Charnes and Cooper 1959, Dantzig 2010) is another methodology commonly employed for solving COPs when uncertainty matters. When some of the data elements in a linear program are most appropriately described using random variables, a stochastic linear program results. In this sense, stochastic programs involve an artful blend of traditional deterministic mathematical programs and stochastic model.

Also, robust optimization (Mulvey et al. 1995) deals with uncertain data in decision making processes. Robust optimization addresses the uncertain nature of an optimization problem without making specific assumptions on probability distributions: the uncertain parameters are assumed to belong to a deterministic uncertainty set. Robust optimization adopts a min-max approach that addresses uncertainty by guaranteeing the feasibility and optimality of the solution against all instances of the parameters within the uncertainty set.

In sum, every one of these methodologies have their markable characteristics. Sensitivity analysis is a post-optimality study to determine the impact that perturbations cause on the nominal problem. In stochastic programming, for example, it is assumed that the probability distribution of the uncertain parameters (random variables) is known or can be reasonably well estimated. Finally, robust optimization is concerned with developing models and methods so that solutions are feasible for any realizations of a convex, previously given set of random variables and violations of constraints are not tolerated. A very clear difference between the methodology and stochastic programming is that the first one does not require knowledge of the probability distribution of the uncertain parameter, while the second one needs (Ben-Tal and Nemirovski 2000, Bertsimas and Sim 2003).

However, some of the methodologies mentioned above usually assume a Normal or Exponential behavior under the presence of historical data. An emerging methodology called simheuristics combining metaheuristics with simulation techniques have been applied to solve complex COPs in different areas (Juan et al. 2011, 2014*a*, Michalak and Knowles 2016, de Armas et al. 2017, Gonzalez-Martin et al. 2018). It allows for developing accurate and flexible models. Specifically, randomness can be modeled throughout a best-fit probability distribution -either theoretical or empirical. The combination of metaheuristics with simulation also promotes the use of risk-analysis criteria during the evaluation of alternative solutions to stochastic COPs.

Simheuristic algorithms belong to the simulation-optimization techniques (Chica et al. 2017), which have been used for some decades. These algorithms are the result of the efficient combination of simulation techniques with already existing metaheuristics to efficiently solve complex stochastic COPs. It is clear that the results are not expected to be optimal. However, simheuristics provide high-quality solutions to complex real-life problems in reasonable computing times. It is better to obtain an approximate solution to an accurate model than the optimal solution for an oversimplified model (Juan et al. 2015*a*). In particular, the simheuristic approach is aimed at solving COPs of the form:

$$\text{minimize} \quad f(s) = E[C(s)] \tag{2.1a}$$

$$\text{maximize} \quad f(s) = E[B(s)] \tag{2.1b}$$

$$\text{subject to:} \quad P(q_i(s) \geq l_i) \geq k_i \ \text{ for all } i = 1, 2, \cdots, n, \tag{2.1c}$$

$$h_j(s) \leq r_j \qquad \text{for all } j = 1, 2 \cdots, m, \tag{2.1d}$$

$$s \in S, \tag{2.1e}$$

where S represents a discrete space of possible solutions s, $C(s)$ represents a stochastic cost function and conversely, $B(s)$ represents a stochastic profit or income function. $E[C(s)]$ represents a probabilistic measure of interest associated with the cost function (e.g., the expected value of $C(s)$). Also, Eq. 2.1c represents probabilistic constraints related to the problem and Eq. 2.1d represents typical deterministic constraints in COPs.

Simheuristic approaches assume that, in scenarios with moderate uncertainty, good quality solutions for the deterministic version of a COP are also likely to be good quality solutions for the stochastic version of the same problem. A general overview of simheuristics is presented in Figure 2.5. Given a stochastic COP, the idea is to obtain the corresponding deterministic version. This can be done, for example, by replacing the stochastic variables by their expected values. Random variables can be modeled by means of probability distributions of the historical data. Then a metaheuristic-based algorithm is used to find a set of high quality feasible solutions for the deterministic COP. After that, a fast simulation is applied to these solutions to see if they are also good solutions for the stochastic COP. At this step we do not need intensive simulation, which will require a great amount of time. The estimated values generated by simulation can be used to rank the elite solutions for the stochastic COP. Then we can execute intensive simulation on the top simulations found with the fast simulation. It must be noticed that these final simulations can also be used to obtain information on the probability distribution of the quality of each solution.

According to Juan et al. (2015*a*) the most relevant advantage of applying simheuristics are that *(i)* they allow the construction and study of valid complex system models; *(ii)* the outputs of the simulation can be employed to generate information about the probability distribution of the quality of each solution and *(iii)* an analysis of the input/output variables space of a model may strengthen trust in the solving approach.

2.3 Concluding Remarks

Due to the increasing number of users and new technologies, telecommunication systems and their associated COPs have received much attention during the last two decades. Taking into account the complexity and dimension of this kind of issues, this chapter presented the main metaheuristics mostly applied when solving those problems. After introducing the most popular metaheuristics, the main classification criteria were discussed and other emerging methodologies such as hybrid metaheuristics and simheuristics were presented. Actually, a part of this thesis focused on integrating BR techniques into search metaheuristic to reduce running time will be presented in Chapter 4. Moreover, a brief but useful taxonomy of search methods for combinatorial optimization which will be useful on the context of this thesis was provided. In the next chapter, a literature review of metaheuristics in telecommunication systems is presented.

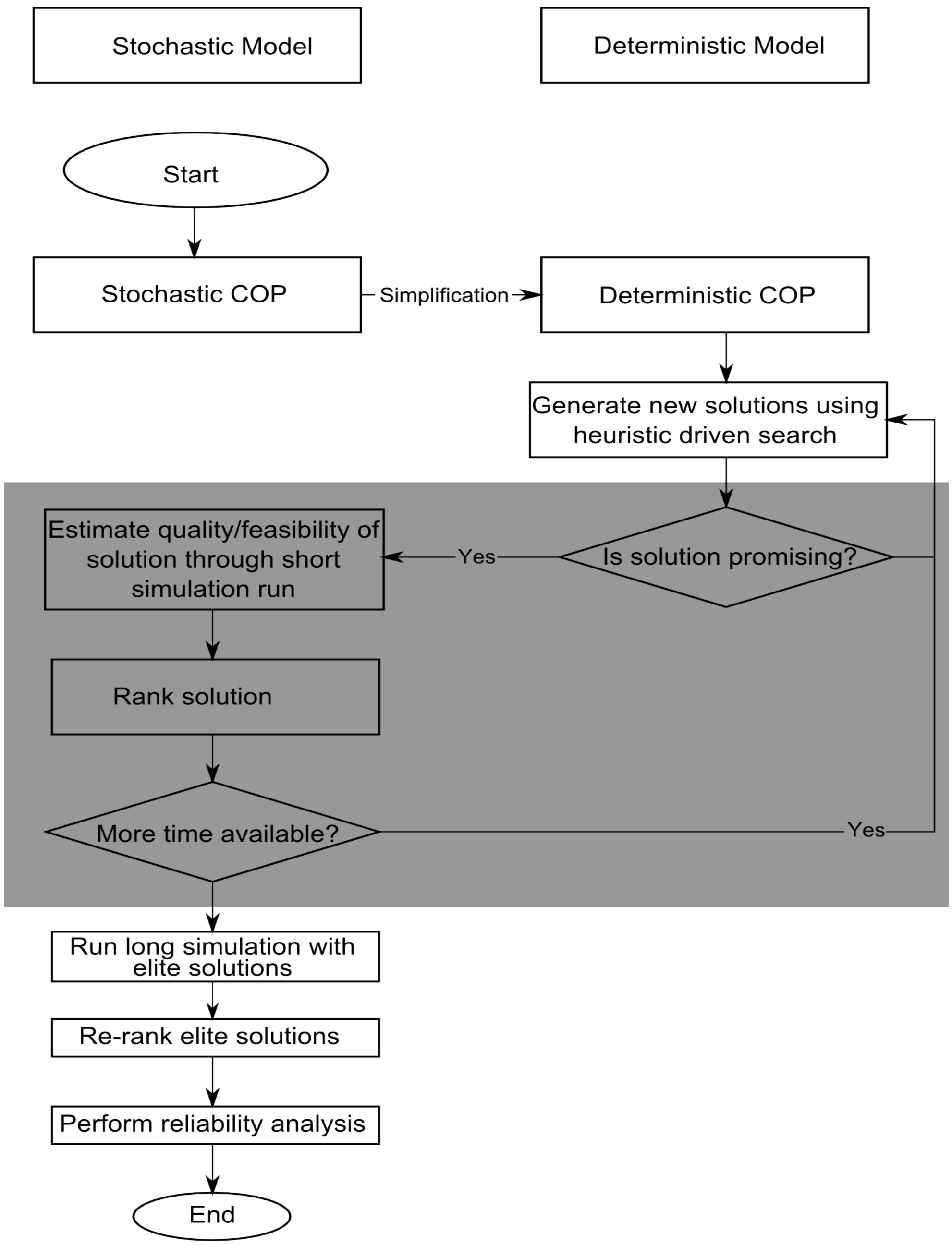


Figure 2.5: Overview of the Simheuristic framework. Retrieved from Juan et al. (2015*a*)

Chapter 3

Metaheuristics in Telecommunication Systems: network design, routing and allocation problems

As stated before, in the information era, telecommunication systems are all around us. Showing an increasing number of users, telecommunication services raise many challenges to the optimization research community (Resende and Pardalos 2008, Donoso and Fabregat 2016, Evans 2017). In effect, some of these challenges can be formulated as COPs, as in the case of the frequency-assignment problem in radio networks, the network design problem, the routing problem, or the optimization of allocation channels (Martins and Ribeiro 2006, Ahmad et al. 2015, Soua and Minet 2015, Medhi and Ramasamy 2017). Since most of these problems are NP-hard and large scale, heuristics and metaheuristics have been increasingly used to deal with them. Often, these metaheuristic approaches have been combined with other soft computing methods, thus generating hybrid algorithms (Blum and Roli 2003)

Network design decisions also affect other managerial decisions, such as repository (or hub) location and routing paths. In recent years, a large number of routing and network design technologies have been developed and updated (Pióro and Medhi 2004, Bidgoli 2016). The diversity of deployed networks and the rapid pace of technological change rise the need for new optimization approaches that support smart decision making. Some of the major driving forces behind these requirements are the need for QoS guarantees and the explosive growth in network size and usage. Likewise, the demand for mobile communication has increased. However, there is a finite spectrum allocated to such services, which raises the question of channel allocation in mobile radio systems (Martins and Ribeiro 2006, Soua and Minet 2015, Yin et al. 2016).

Additionally, since architectural re-design efforts are time consuming and hence expensive, there is a critical need for efficient approaches to support the proper design decisions. Thus, the use of metaheuristic algorithms is required in solving COPs in the telecommunication area.

Table 3.1 gives a general overview over the discussed application fields. For the scope of this work, we focus on the challenges and properties of the optimization problems previously mentioned. Hence, this chapter presents an analysis of some research challenges related to metaheuristics and their applications to optimization problems in telecommunication network design, routing and allocation of resources.

3.1 Network Design

As mentioned before, telecommunication network-design problems have gained attention over the last decades. The classical version of this problem consists in finding a network design that minimizes total costs while satisfying users' demands, providing a minimum QoS, and respecting the capacity constraints of each link. In the past, the design of communication networks was solved as a single objective optimization problem, using the cost of the network as the objective to be minimized and considering constraints such as reliability, maximum delay, etc (Dengiz et al. 1997).

However, there is a clear trend to consider the design of a telecommunication network as a multi-objective optimization problem. A typical architecture for such a network consists of tributary networks –which connect nodes to hubs– and a backbone network –which interconnects the hubs. Depending on the application, hub nodes are called by various names, including gates, concentrators, switches, control points, or even access points (Klincewicz 1998). Tributary networks are also called local or access networks, while backbone networks may be referred sometimes as hub-level networks.

Frequently, due to the size of the problem, the design of the backbone network is considered independently from that of the tributary networks. Thus, a solution approach proposed by Chamberland et al. (2000) for the integrated design was based on a TS algorithm. The authors dealt with the problem of how to expand a Metropolitan Area Network in a cost effective way. The proposed model considered the update of the access network with a star

Table 3.1: Overview of optimization challenges in telecommunications

Optimization Problem	**Challenges and requirements**
Network Design	• Location of hosts, servers, terminals, and other end nodes • Projected traffic for the network • Projected costs for delivering different service levels • Limitation of hardware resources • Security • QoS requirements • Reliability and availability
Routing	• Efficient routing with mesh infrastructure • Scalability • Robustness • Guarantee security • Overloaded situations
Channel Allocation	• Adaptability • Channel utilization • QoS support • Fault tolerance • Self-configurability • Packet flow assignments

topology and the expansion of the backbone network with various types of topologies. Results proved that the TS approach presents a relatively stable behavior in terms of closeness to the optimal solution. The results were obtained on instances up to 500 users in a reasonable amount of time. As the authors concluded based on the results they obtained, the limitation to the application of their methodology if the difficult of evaluating the lower bound since such evaluation becomes time-consuming as the problem increases in size.

Another COP that arises in the design of telecommunication networks is the star ring problem: the aim is to locate a simple cycle through a subset of vertices of a graph with the objective of minimizing the total cost of all connections in a ring star topology. A hybrid metaheuristic approach to solve this problem was proposed by Dias et al. (2006). The approach combined a VNS with a GRASP algorithm. The hybrid metaheuristic was tested on the Traveling Salesman Problem library (Reinelt 1991), with instances involving between 50 and 100 vertices.

Even though the star topology is a distinct choice for many types of access networks, there are situations where other layouts might be more appropriate, as the tree topology. Girard et al. (2001) presented a fast TS algorithm for the design of access tree networks. They described in detail the parameters used for the data structure and indicated how that could lead to substantial improvements of the overall computational time while providing costs lower than those obtained using traditional methods. The authors evaluated the proposed methodology on artificial networks varying from 20 to 100 nodes and on a real network with 46 users represented as nodes and 4 switches.

Another structure that have received attention is the minimal spanning tree (MST). MST problems are considered to lie in the core of network systems design. Few years ago, Ruiz et al. (2015) proposed a biased randomization genetic algorithm which evolves a population of random vectors that encode solutions to the capacitated minimum spanning tree problem. The authors used the sets of instances with a number of terminals $n \in \{80, 120, 160\}$.

The work presented by Carello et al. (2004*a*) dealt with a hub location problem (HLP) in which the topologies of the backbone and the tributary networks are given. The locations of the hubs must be chosen among the terminal nodes, and each terminal node must be assigned to exactly one hub. The authors addressed the problem of planning a two-level network as illustrated in Figure 3.1, i.e.: a network in which the traffic collected by the access nodes must be routed through a backbone network, whose nodes are called transit nodes. In order to solve the problem, the authors proposed a TS-based algorithm. The proposed algorithm was tested on 19 instances with up to 49 nodes.

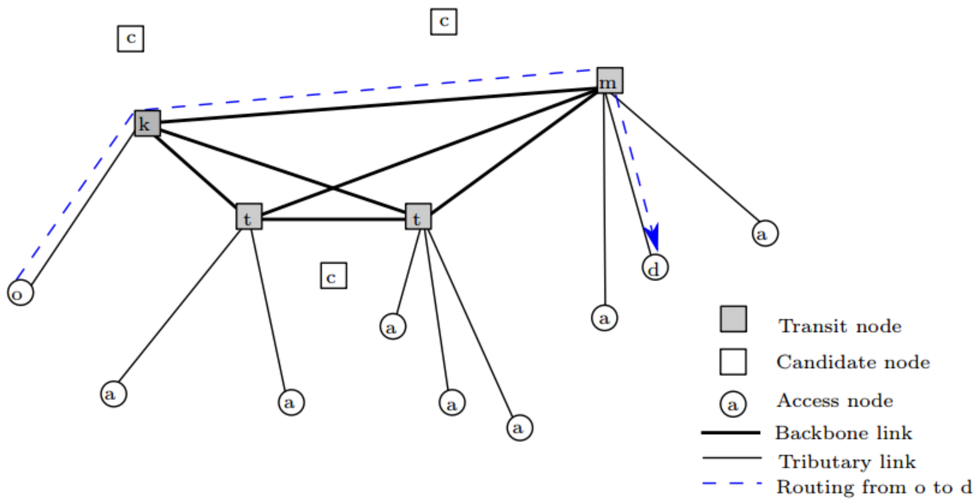


Figure 3.1: Example of a backbone/tributary network problem, in which the backbone network is fully connected while the tributary networks are star networks (Carello et al. 2004*a*)

The importance of telecommunication networks has dramatically increased over the past few years. Today's networks require a significant amount of investment in order to guarantee QoS and performance. Network design and planning in engineering systems requires policy decisions, analysis of investment strategies, and technical development plans. The multi-level network optimization problem (MNOP) is a network design model that raises optimization aspects of dimension, topological design, and facility location.

MNOPs appear in many contexts such as telecommunication, logistics, transportation, and electric power systems. Flores et al. (2003) proposed parallel asynchronous versions of promising multi-objective EAs, with the aim of designing an optimal telecommunication network in the presence of multiple conflicting objectives as cost and performance. The test problem chosen for testing the proposed algorithms it is a simplified version of a real network design problem conceived to link, using distinct types of fiber, 19 universities and research centers located in 9 different cities of Turkey.

As already mentioned, the topological design problem is NP-hard, and thus it becomes intractable using exact methods as the number of nodes increases. With the emergence of some applications –such as teleconferencing, interactive simulation, or distributed content systems and multi-games–, the number of people demanding multi-cast services is growing. Multi-cast communication refers to the delivery of information to many receivers simultaneously. This information may come just from one source (one-to-many multi-cast) or many sources as well (many-to-many multi-cast).

In general, there are several potential applications of multi-cast, e.g.: news delivery, stock quotes distribution, software updates, audio and video streaming, etc. Li and Pan (2011) formulated this problem and then solve it using a parallel GA. In their computational experiments the algorithm was tested in a network with 25 nodes. The GA-based solution could generate a lower link cost to achieve multi-cast. An issue regarding multi-cast is how to provide a reliable service.

The provisioning of reliable multi-cast service deals with how to handle packet re-transmissions. Santos et al. (2006) focused on the server replication method, wherein data is replicated over a subset of the multi-cast-capable relaying hosts and re-transmission request from receivers which are handled by the nearest replicated server. They proposed a hybrid metaheuristic to find near-optimal solutions to this problem. Experiments were conducted on multi-cast scenarios with 2000 transit nodes and 200 servers created with the Georgia Tech Internetwork Topology Models (GT-ITM) toolkit (Calvert et al. 1997).

As an alternative of connecting each pair of demand nodes with a direct connection, a hub-and-spoke topology is used in a number of networks. In this type of networks, direct communication between pairs of demand nodes is usually pricey. The flow of information, goods, or passengers from different origins can be obtained at hub nodes before transmission to their destination. Performance of these networks relies mainly on the use of consolidation, switching, or transshipment points, knows as the hub facilities, where the flows from several origins are consolidated and rerouted to their destinations, sometimes via another hub. Since HLP involves the movement of commodities, information and people, it is not surprising that the design of hub-and-spoke networks is widely applied in telecommunication industry, logistical systems, airline industry and postal companies.

Today, there are many other areas that can take advantage of the hub concept like maritime industry, freight transportation companies, public transit and message delivery networks (Farahani et al. 2013). The design of hub-and-spoke networks is known to be also an NP-hard problem, and it has recently been tackled by approaches based on metaheuristics. Thus, Gomes et al. (2013) proposed an efficient GA for the design of hub-and-spoke with single allocation, where the creation of the initial population is based on a GRASP metaheuristic. The authors tested the proposed GA on the Civil Aeronautics Board (CAB) and Australian Post (AP) data sets. Likewise, Sun (2012) considered a capacitated asymmetric allocation hub problem. The approach determined the number of hubs, their location, and the asymmetric

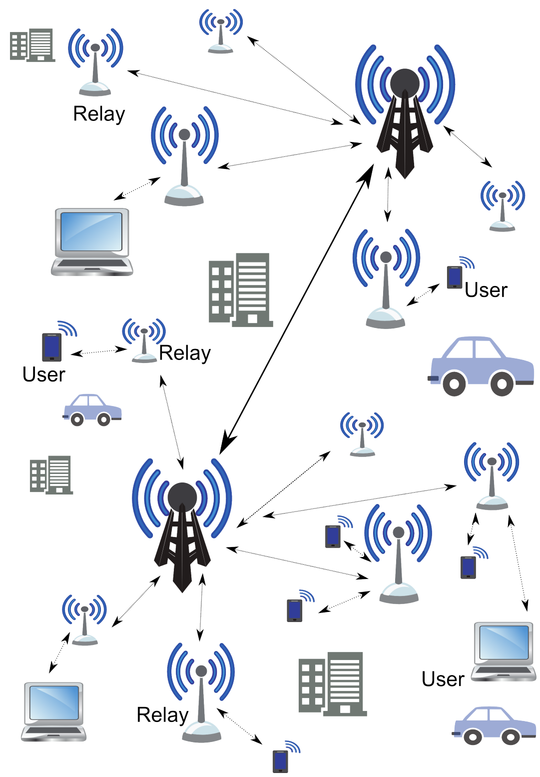


Figure 3.2: Example of a network with relays

allocation of non-hubs nodes to hub with the objective of minimizing total transportation costs while satisfying the required service level. The proposed solution method is based on combining ACO and GA.

Another hub location related problem that arises in telecommunication systems is the directed network design problem with relays (DNDR). The problem is illustrated in Figure 3.2. Given a directed network and a set of commodities, the DNDR consists of introducing a subset of arcs and locating relays on a subset of nodes such that in the resulting network, the total cost is minimized. Recently, Li et al. (2017) presented an ILS algorithm in order to solve the aforementioned problem. The algorithm was tested on a set of instances with nodes up to 160.

On the other hand, networks may become partially disconnected due to catastrophic component failures, and it is important for network users to be able to access some network services even under such circumstances. Availability of network services can be increased by strategically allocating servers over a network. In that sense, Kulturel-Konak and Konak

(2010) developed a simulation-optimization approach combining Monte Carlo simulation with PSO to solve the reliable server assignment problem (RSAP). The proposed approach was tested using random problems ranging from 30 to 100 nodes. Then, the same authors defined this problem as determining a deployment of identical servers to maximize a measure of service availability, and solved it using nature-inspired metaheuristic approaches such as ACO and PSO (Konak and Kulturel-Konak 2011).

The RSAP in networks has been addressed by a limited number of works presented in the literature. This is closely related to the $p-$median problem (Hakimi 1964), and is concerned with locating p identical servers at p distinct nodes of a network to minimize the total weighted distance between the nodes and the closest servers. Most network reliability problems are NP-hard and, because of its difficulty, the $p-$median problem has been usually studied under simplifying assumptions. Few years ago, Konak et al. (2015) introduced the RSAP considering attacks, which seeks to choose the locations of servers on a network in order to maximize the network reliability that results from a worst-case attack on the edges of the network. They introduced a GA that embeds the game-theoretic structure of the problem into the algorithm. The GA-based algorithm was tested on problems size with 11, 25 and 30 nodes.

Moreover, telecommunication service operators are increasing their investment in solving the network design problem in order to deal with any traffic requirement under certain bounds and physical network conditions. Diaz-Baez et al. (2013) proposed a GA to solve the RNDP with optimal capacity of links, considering a stable routing with uncertain traffic that can be divided into k sub-routes. However, raising the k value implies an increase in the number of viable solutions. Thus, a trade-off between k and the quality of the solutions obtained by the proposed algorithm was detected. The authors presented the results obtained with a network with 14 nodes and 21 links using the National Science Foundation topology (NSF)(Chinoy and Braun 1992).

Likewise, given the importance and complexity of the robust network design problem (RNDP), Arteta and Pinto-Roa (2015) studied the RNDP subject to guarantee certain QoS level. By reserving an adjustable bandwidth for each node, the network is not negatively influenced by traffic from the rest of the network. Hence, a multi-objective EA was proposed to solve and find a robust network design, which also minimizes the cost of the network, minimizes the inequity of traffic, and maximizes the traffic service in the worst-case scenario. The experiments were carried using the NSF topology.

Pressure to reduce costs is also adding new urgency to the search of practical communication network design and optimization algorithms that can pack traffic into fewer or less expensive facilities without requiring new technology or capital purchases. Thus, Cox and Sanchez (2000) presented in their work a metaheuristic algorithm based on a short term TS approach for designing least-cost telecommunication networks that carry cell site traffic to wireless switches while meeting survivability, capacity, and technical compatibility constraints.

With an explosive increase in data traffic over recent years, it has become increasingly difficult to rely on outdoor base stations to support the traffic generated indoors mainly due to the penetration issue of wireless signals. In the past, outdoor cellular networks had been very successful to provide wide coverage area. However due to physical barriers of buildings such as walls and windows, it becomes very difficult for wireless signal to penetrate buildings and achieve the required signal strength for the indoor mobile users. Shakya et al. (2018) proposed a GA for designing an in-building distributed antenna system based on the real world requirements of a telecommunication service provider. The algorithm was tested on a instance of a building with $12, 25, 48$ and 60 floors.

Also, many research works have been done for optimizing average network delay and, thereby, designing either minimal-cost reliable networks or maximal-reliable economic networks. A hybrid PSO algorithm was implemented by Papagianni et al. (2008) to design a network infrastructure including decisions concerning the locations and sizes of the links. The authors presented the results from a network with 16 nodes and 120 edges. The results indicated an improvement in the optimization process in comparison to GA. Similarly, Dasgupta et al. (2012), the authors proposed a modeling of data networks with delay and packet loss ratio. They minimized network cost using GA. For that, three objective functions were defined. The proposed solution was tested with a 5-node network.

Tables 3.2 and 3.3 summarize the main articles that have proposed metaheuristics applied to network design problems in telecommunications. The first column indicates the article in which the metaheuristic-based methodology was employed. The next two columns (Single-solution and Population-based) indicates the classification of the metaheuristic in hand. The last column introduces some notes regarding the limitations of the presented methodologies. The average running time is indicated in seconds (s), minutes (m) or hours (h). Notice that TS is the most popular single-solution search metaheuristic to approach network design problems. However, population-based metaheuristics, especially GAs, are the most employed methodologies. As a matter of fact, we can see that the majority of studies have dealt with small and medium large-size problems.

Table 3.2: Metaheuristics applied to network design optimization problems (I)

Article	**Optimization problem**	**Single solution**	**Population based**	**Notes**
Cox and Sanchez (2000)	Design least-cost network	TS		Network up to 25 nodes and average time 1 to 3 (m)
Chamberland et al. (2000)	Topology design	TS		Network up to 500 nodes and time consuming when evaluating lower bounds
Girard et al. (2001)	Topology design	TS		20 to 100 nodes network and average time 137 (s)
Flores et al. (2003)	Optimal network design		EA	19 nodes network and average time 4.54 (h)
Carello et al. (2004a)	HLP	TS		49 nodes network and average time 73.02 (s)
Dias et al. (2006)	Topology design	GRASP		50 to 100 vertices and average time 11.39 (s)
Santos et al. (2006)	Reliable MRP	GRASP		Network up to 2000 nodes and average time 9586.83 (s)
Papagianni et al. (2008)	Topology design		PSO	26 nodes network and time not reported
Konak et al. (2010)	RSAP		PSO	30 to 100 nodes network and average time 540.28 (s)
Konak et al. (2011)	RSAP		PSO	11 to 100 nodes network and time not reported

Table 3.3: Metaheuristics applied to network design optimization problems (II)

Article	**Optimization problem**	**Single solution**	**Population based**	**Notes**
Li and Pan (2011)	Coding topology design		GA	25 nodes network and and use of slave processors
Dasgupta et al. (2012)	Topology design		GA	5 nodes network and time not reported
Gomes et al. (2013)	HLP		GA	10 to 200 nodes network and longest time 1.95 (*s*)
Diaz-Baez et al. (2013)	Robust design		GA	14 nodes network and 60 (*m*)
Konak et al. (2015)	RSAP		GA	11 to 30 nodes and average time 6122.7 (*s*)
Ruiz et al. (2015)	MST		Hybrid	network up to 160 nodes and time less than 2200 (*s*)
Arteta and Pinto-Roa (2015)	Robust design		EA	14 nodes and time not reported
Li et al. (2017)	DNDR	ILS		Network up to 160 nodes and time less than 1800 (*s*)
Shakya et al. (2018)	Distributed Antenna System		GA	Network up to 60 nodes and time not reported

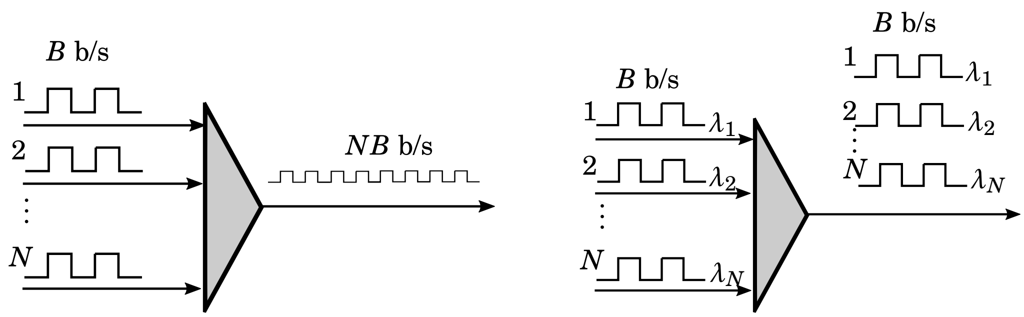
3.2 Routing

In the last decades, we have seen dramatic changes in the telecommunication industry that have far-reaching implications for our life-styles. There are many drivers for those changes: there is a continuing and relentless need for more capacity in the network and, at the same time, business today rely on high-speed networks to conduct their business. Not too many years ago, wire and radio technologies were the choices to send messages effectively.

Today, optical fiber has been displacing wire in many applications and, with wireless, is emerging as one of the dominant transmission technologies. The aforementioned factors have driven the development of high-capacity optical-fiber networks and their remarkably rapid transition from the research laboratories into commercial deployment. Optical-fiber networks offer the promise to solve many of the problems we have mentioned.

In addition to providing enormous capacities in the network, an optical-fiber network provides a common infrastructure over which a variety of services can be delivered. In the first generation of these networks, optical fiber was essentially used for transmission, and simply to provide capacity. The second-generation of optical-fiber networks has also routing and switching capabilities, as well as intelligence in the optical layer (Ramaswami et al. 2009). In both generations, multiplexing techniques provide an increase in transmission capacity. The need for multiplexing is driven by the fact that, in most applications, it is much less expensive to transmit data at higher rates over a single fiber than to transmit at lower rates over multiple fibers. There are basically two ways of increasing the capacity transmission on a fiber as depicted in Figure 3.3 the Wavelegth Division Multiplexing (WDM) and the Time Division Multiplexing (TDM).

The idea of WDM is to transmit data simultaneously at multiple carrier wavelength (or equivalently, frequencies or colors) over a fiber. In a wavelength-routed WDM network, end users communicate with one another via all-optical WDM channels, which are referred to as light-paths (Chlamtac et al. 1992). A light-path is viewed as a point-to-point light connection from a source to its destination. The concept of light-path can be extended to a light-tree, where a point-to-multipoint connection is set up using a single or multiple wavelengths (Sahasrabuddhe and Mukherjee 1999). Siregar et al. (2005), considered the Multi-cast Routing Problem (MRP) in large-scale WDM optical-fiber networks, where transmission requests are established by point-to-multipoint connections. The authors proposed a GA that exploits the combination of alternative shortest paths for the given multi-cast requests in order to minimize the number of required split-capable nodes.



(a) TDM multiplexing

(b) WDM multiplexing

Figure 3.3: Different multiplexing techniques for increasing the transmission capacity on an optical fiber. Both multiplexing techniques take in N data streams, each of B b/s, and multiplex into a single fiber with a total aggregate rate of NB b/s

Given a set of connections, the problem of setting up light-paths by routing and assigning a wavelength to each connection is called the Routing and Wavelength Assignment Problem (RWAP) (Figure 3.4). An algorithm to solve the dynamic RWAP in a distributed manner was proposed by Aragón et al. (2007). It uses ACO to obtain updated information about the network state, which is then employed to find the routes and wavelength that allow to establish new connections through an adaptive process able to deal with dynamic changes in the network state. Hassan and Phillips (2008) proposed a PSO metaheuristic approach for solving the static RWAP. To help the particles to converge towards an optimal solution quickly, a novel scheme is devised for route selection during the particles search. Later, the same authors addressed the dynamic RWAP in WDM optical networks (Hassan and Phillips 2009). For solving this variant, they proposed a Chaotic PSO. The static RWAP is often referred to as the Virtual Topology Design Problem (Dutta and Rouskas 2002). Ghose et al. (2005), the authors considered the problem of designing virtual topologies for multi-hop optical WDM networks. To analyze this problem, two metaheuristic algorithms were introduced: GRASP and EA. An ACO routing algorithm was proposed by Pavani and Waldman (2006) for transparent optical-fiber networks. It took into consideration the bit error rate of the connections that is derived from amplified spontaneous emission noise accumulated along the light-path.

In order to maximize the usage of the light-paths, telecommunication carriers adopt a technique that consists in grooming low speed traffic streams into high capacity channels. This technique is referred to as the RWAP with traffic grooming. Few years ago, Wu et al.

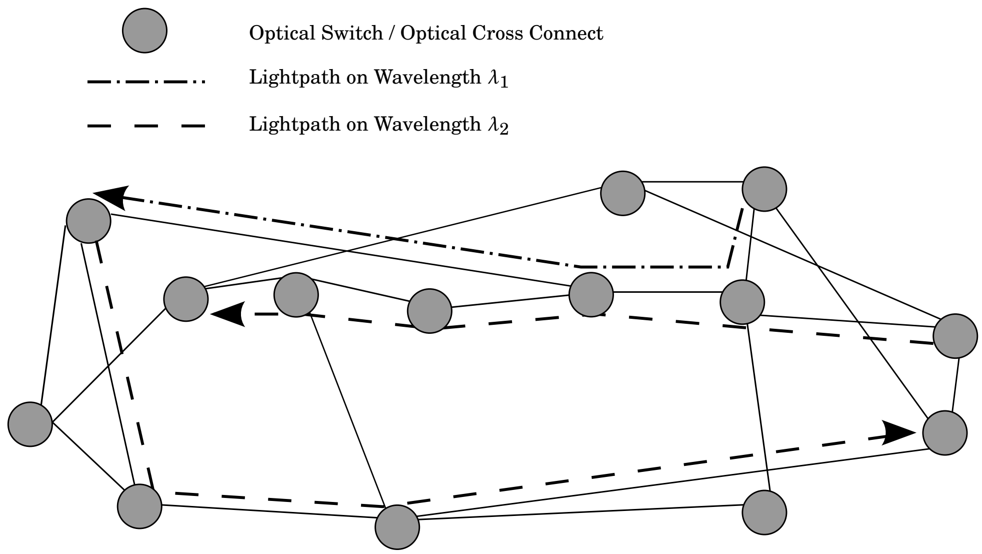


Figure 3.4: Wavelength-routed optical WDM network with light-path connections (Hassan and Phillips 2008)

(2015) proposed a GRASP algorithm for solving the traffic grooming and routing problem with simple path constraints in WDM mesh networks, and introduced a mechanism to tackle the interaction between the grooming problem and the routing problem. In addition, traffic demands in WDM optical-fiber networks can be classified into three categories (Gagnaire et al. 2007): permanent or static lightpath demands (PLDs), scheduled lightpath demands (SLDs), and random lightpath demands (RLDs). SLDs are known in advance, and are supposed to be active only for a limited period of time (hours, days, or weeks). In this regard, Marković et al. (2012) studied the RWAP of SLDs in all-optical WDM networks with no wavelength conversion capability. They proposed an algorithm based on the BCO metaheuristic. It was shown that, by applying the proposed algorithm, significant improvements in terms of the number of established light-paths could be achieved by taking into account temporal information of light-path demands compared to the case when this information is not considered.

Authors agree that optimizing information exchange and routing is a challenging problem with implications of many aspects of the network: determining the required paths while minimizing the cost, minimizing the delay or maximizing the reliability, etc. A commodity represents a certain demand of telecommunication traffic between two nodes. If multiple pairs of source and destinations have to be managed, the problem is defined as a multi-commodity flow problem (MCFP). Masri et al. (2011) extended the MCFP by considering multiple sources for each flow, and a solving ACO metaheuristic was proposed.

Orthogonal Frequency Division Multiplexing (OFDM) was proposed as a modulation technique for optical networks due to its good spectral efficiency. Optical OFDM is more flexible compared to traditional WDM systems, enabling elastic bandwidth transmissions. Christodoulopoulos et al. (2010) introduced the routing and spectrum allocation problem in OFDM-based optical networks, as opposed to the typical RWAP in traditional WDM networks. The objective is to serve the connections through adequate spectrum allocation, with the constraint that no spectrum overlapping is allowed among those connections, and minimize the utilized spectrum. These authors proposed a SA metaheuristic to find good orderings that yield near optimal performance.

In addition, many researches have proposed metaheuristics for solving different variants of network routing problems. Thus, Sim and Sun (2002) introduced an algorithm based on ACO for solving various types of routing and congestion problems in computer networking. Likewise, Vaezpour and Dehghan (2014) addressed the joint channel and routing problem for multi-cast applications. In their work, a technique based on a multi-objective GAs was proposed to build a delay-constrained multi-cast tree with minimum interference. Similarly, Kusetogullari et al. (2011) proposed a GA and a PSO metaheuristic algorithms to maximize utilization and improve QoS in expanding networks.

In a data communication network, nodes and arcs represent routers and transmission links, respectively. Intra-domain traffic engineering aims at making a more efficient use of network resources within autonomous systems (Buriol et al. 2005). Interior gateway protocols, such as the Open Shortest Path First (OSPF) and the Intermediate System-Intermediate System (IS-IS) are commonly used to select the paths along which traffic is routed within an autonomous system. Given a set of traffic demands between origin-destination pairs, the OSPF weight-setting problem consists on determining weights to be assigned to the links so as to optimize a cost function, typically associated with a network congestion measure. The work presented by Fortz and Thorup (2004) was one of the first considering even traffic splitting in OSPF weight setting. Later, Buriol et al. (2005) proposed a GA with a local search procedure for the OSPF weight-setting problem.

Ad hoc networks are a type of wireless networks that do not require any infrastructure such as a backbone or configured access points. The great advantage of *ad hoc* networks is the high flexibility they offer, even when there is no fixed communication infrastructure, there are high installation costs, or the reliability levels are lower than used to be in other networks. Other advantage of *ad hoc* networks is their robustness. Due to these characteristics, there are several applications of *ad hoc* networks: they can be used in places where rapid installation is required.

A mobile *ad hoc* network (MANET) is a collection of mobile nodes which communicate over radio. In a MANET, no infrastructure is required to allow the exchange of information between mobile device users. Since the nodes are mobile, the network topology may change rapidly and in an unpredictable way over time. The major challenge found in this kind of networks is to find a path between the communication end points satisfying QoS requirements. The difficulty of this challenge is increased due to the mobility of the nodes. Gunes et al. (2002) introduced an ACO metaheuristic to handle the routing problem in a multi-hop MANET. This approach consisted of three phases: route discovery, route maintenance, and handling of route failures.

Multi-cast routing is one type of data transmission service in MANET where the data are sent from source node to many destination nodes through more than on path. In the MRP, QoS mainly depends on cost, delay, jitter and bandwidth. Recently, Wei et al. (2018) proposed a multi-objective multi-cast routing optimization GA-based algorithm

Likewise, Deepalakshmi and Radhakrishnan (2009) proposed years ago an ACO-based routing protocol for a MANET supporting multimedia communications. In fact, Multi-casting plays and important role in the *ad hoc* wireless networks. Mangai et al. (2008) introduced an ACO metaheuristic for multi-cast routing in *ad hoc* wireless networks. A vehicular *ad hoc* network (VANET) is a subclass of a MANET frequently employed in intelligent transportation systems. Multi-casting provides different traffic information to a limited number of vehicle drivers by a parallel transmission. However, it represents a very important challenge in the application of VANETs, especially in the case of the network scalability. Bitam and Mellouk (2013) proposed a BCO algorithm to solve the QoS MRP for VANETs with multiple constraints. A few years ago, Nancharaiah and Mohan (2014) proposed a hybrid routing intelligent algorithm that combines a PSO approach and an ACO approach with the goal of improving various performance metrics in MANET routing, including: end-to-end delay, power consumption, and communication costs. Also, Kim (2014) introduced a multi-path routing scheme employing SA to deal with a hostile dynamic real-world situation into the conflict MANET routing problem.

Advances in sensor technology and computer networks have enabled distributed sensor networks (DSN) to evolve from static network topology to dynamically changing topology. In such dynamic environment design and development of efficient routing protocol remain a challenge for researches. Mann et al. (2016) presented a population-based metaheuristic based on swarm intelligence for energy-efficient hierarchical routing protocol for WSN. The proposed protocol was simulated on the Nature Inspired Tool for Sensor Simulation (NITSS) with a network up to 285 nodes. Similarly, Das et al. (2016) proposed a GA-based model to further maximize the network lifetime. The proposed model implements a multihop routing mechanism for data dissemination from source to the sink. The algorithm was evaluated in the range of 20 to 60 sensor nodes.

Tables 3.4, 3.5 and 3.6 summarize the articles that have proposed metaheuristics applied to routing network design problems. The first column indicates the article in which the metaheuristic-based methodology was employed. The next two columns (Single-solution and Population-based) indicates the classification of the metaheuristic in hand. And the last column, denoted by "Notes" gives a brief description of the limitations found on the analyzed article. Up to now, population-based metaheuristics like GAs and ACOs are the most employed ones in these routing problems.

3.3 Channel Allocation and other Problems

Another relevant problem is the channel allocation in mobile radio system. The rapid growth of cellphone users brought the need for efficient reuse of the limited frequency spectrum allocated to cellular mobile communications (Lee 1995, Tragos et al. 2013). The efficient reuse of this spectrum is also important from the financial point of view, since less spectrum required to offer services to the same number of users means a lower cost. In the current scenario of cellular mobile services, the transmission frequencies are grouped into bands which are usually codified in a set of channels. Thus, each base station receives a portion of the total number of channels available to the entire system.

The channel allocation task in wireless communication deals with the allocation of available communication channels to any mobile host within its network coverage. A base station is responsible for controlling the allocation of channels in a cell. Communication between base stations enables unused channels to be temporarily transferred from one cell to another. Suliman et al. (2016) proposed an AIS-based Algorithm for bandwidth spectrum allocation in wireless communication. Their model works by allocating the unused channels from the cells with less demand to overloaded cells. The proposed algorithm was tested on networks with 50 and 100 mobile hosts.

Tackling interference is an essential issue in wireless communications to which interference alignment provides a promising solution. Interference alignment aligns, at each receiver,

Table 3.4: Metaheuristics applied to routing network design optimization problems (I)

Article	**Optimization problem**	**Single solution**	**Population based**	**Notes**
Sim and Sun (2002)	Network routing		ACO	Path may become congested if all ants travel along it
Gunes et al. (2002)	MANET routing		ACO	In cases with high node mobility results show a higher variation
Fortz and Thorup (2004)	OSPF weight setting	LS		Algorithm tested with a network up to 100 nodes
Buriol et al. (2005)	Weight setting		GA	Algorithms were run for 1-hour trials
Siregar et al. (2005)	MRP		GA	The GA was tested over a small network under 20 nodes
Ghose et al. (2005)	Virtual routing	GRASP	EA	The traffic in single-hop increases with the increase in the number of wavelengths
Pavani and Waldman (2006)	Adaptive RWA		ACO	The longest feasible route has less hops than the pre-established number
Aragón et al. (2007)	Distributed RWA		ACO	The algorithm presents a higher and more unpredictable maximum setup time

Table 3.5: Metaheuristics applied to routing network design optimization problems (II)

Article	Optimization problem	Single solution	Population based	Notes
Hassan and Phillips (2008)	Static RWA		PSO	The algorithm was tested using a network under 20 nodes
Mangai et al. (2008)	Multi-cast routing		ACO	Packet delivery ratio of the protocols slightly decrease when increase the speed
Hassan and Phillips (2009)	Dynamic RWA		PSO	At very high traffic the availability of free wavelengths becomes very limited
Deepalakshmi et al. (2009)	MANET routing		ACO	End-to-end delay tends to increase with decrease in number of nodes
Christodoulopoulos et al. (2010)	MCFP	SA		The algorithm was evaluated using a 14 nodes network
Masri et al. (2011)	MCFP		ACO	The CPU time is influenced by the number of messages to be routed
Kusetogullari et al. (2011)	MANET routing		Hybrid	Increasing the population intersection produces information loss and distortion
Marković et al. (2012)	RWA of SLDs		BCO	The number of bees significantly affects the required CPU times

Table 3.6: Metaheuristics applied to routing network design optimization problems (III)

Article	Optimization problem	Single solution	Population based	Notes
Bitam and Mellouk (2013)	QoS-MRP VANET		BCO	The algorithm was tested over a small network under 20 nodes
Kim (2014)	MANET routing	SA		The algorithm was evaluated using a network of 100 nodes
Vaezpour and Dehghan (2014)	Multi-cast routing		GA	Algorithm is limited when using aggregate objective function
Nancharaiah and Mohan (2014)	MANET routing		Hybrid	When density of nodes increase decrease the number of rounds
Wu et al. (2015)	Traffic groaming	GRASP		Poorly performance whit larger ratio and with sparse networks
Mann et al. (2016)	WSN routing		BCO	The hybrid algorithm was evaluated using a network of 100 nodes
Das et al. (2016)	WSN routing		GA	The hybrid algorithm was evaluated using a network up to 60 nodes
Wei et al. (2018)	MANET routing		GA	Algorithm tested with a network up to 50 nodes

the interference in the smallest possible part of the space formed by available signaling dimensions; the remaining part will contain the useful signal that will be recovered using a decoding matrix. Recently, Messaoud et al. (2017) performed an interference alignment optimization using PSO and BCO algorithms. The proposed algorithms were tested on different scenarios in which transmitters and receivers were equipped with 5 antennas each.

For various reasons and especially because of user's mobility, the signals between the mobile unit and the base station may become weaker while interference from adjacent cells increases. When a user in communication crosses the line between adjacent cells, occurs a handoff, i.e., the mobile network automatically supports the transferring of a communication from one cell to another adjacent cell. Efficient assigning of cells to switches can have a significant impact on handoff and cable costing. In order to solve the assignment of cell to switches problems, Mirsaleh and Meybodi (2018) proposed a hybrid metaheuristic obtained from the combination of learning automation and local search. The experiments for testing the proposed algorithm were carried out on a cellular network with the number of cells up to 100 and the number of switches up to 5.

Keeping the required hardware investments to a minimal level while achieving a high QoS is the basic principle of network planning; the planning process can be split into two main steps: dimensioning and frequency assignment. Bedoui et al. (2014) proposed a hybrid metaheuristic for solving the multi-objective frequency assignment problem in broadcasting. They employed a variant of TS, which uses a probabilistic aspiration criterion, the probabilistic TS combined with a GA and SA algorithms. Similarly, Chen et al. (2016), designed a GA-based resource allocation scheme under different frequency bands and in the OFDM system for the downlink of Long Term Evolution advanced. The results were obtained by testing the proposed algorithm on a scenario with 64 user equipment.

To provide flawless services, managing a stable network bandwidth is a major concern in wireless network. In this sense, wireless mesh network (WMN) is an initial step towards providing cost effective, dynamic, high bandwidth networks over a specific coverage area. However, minimizing the influence of interference is a prime challenge. The multichannel-multiradio concept have been introduced to alleviate this problem with the help of efficient channel allocation algorithms. Chakraborty and Debbarma (2017) proposed a GA-based algorithm in order to solve the MRP and channel assignment in multichannel multiradio wireless mesh network. The analysis of the algorithm performance was based on a 14 available channel network. Likewise, in Ohatkar and Bormane (2014) a GA is introduced for hybrid channel allocation. It focuses on reducing the interference in cellular networks.

Therefore, it is important to establish a strategy in order to reduce the total use of the available frequencies. Unfortunately, when it comes to wireless communication there is the interference problem –a phenomenon that represents the superposition of two or more electromagnetic waves at the same point. Some approaches have been developed in order to solve it. Soo-Hyun et al. (2000) considered the channel allocation problem. In such problem, the goal is to minimize the weighted average blocking probability subject to co-channel interference constraints in a cellular mobile system. These authors simplified the problem using the concept of pattern, and applied a SA procedure to deal with it. Similarly, He et al. (2018) proposed a GA-based algorithm to solve the channel allocation and power control problem for cognitive radio networks with multiple constraints and get the optimal channel allocation strategy. The proposed algorithm was tested on a network with number of users up to 6.

Cellular networks are at the center of this expansion. With the growing of the number of users and services provided by cellular networks, the cellular network should be highly efficient to satisfy the users' needs. The radio network design is a key step in the life cycle of a network. It involves many sub-problems, like frequency assignment problem and antenna positioning problem. The latter consists in finding the locations for the base stations to ensure a maximum radio coverage, while minimizing the number of the used base stations. Recently, Benmezal et al. (2017) an hybrid metaheuristic framework based on ILS and Breakout Local Search (BLS) . The proposed algorithm was tested on a realistic instance with a total of 135000 points, among which 1000 points are chosen to be potential transmitter locations.

Along with the fast-increasing in mobile cellular networks, wireless sensor networks (WSN) have been evolving. A WSN is usually described as a network of nodes that track physical or environmental conditions, enabling interactions between persons or computers and the enclosing environment (Bröring et al. 2011). During the deployment of a WSN, one key objective is the full coverage of the monitoring region with a minimal number of sensors and a minimal energy-consumption of the network. Fidanova et al. (2014) proposed a multi-objective ACO to solve this problem. Later, Das et al. (2015) proposed a similar algorithm to minimize the number of required sensors while covering a maximum area. They developed an algorithm based on TCO.

Due to spectrum scarcity and the high demand for services, the same problem that is presented in mobile networks is also a problem in satellite communication networks. The challenge of assigning telecommunication services to satellite spectrum resources have been modeled as a packing problem. Thus, Wille et al. (2005) introduced a packet network design

Table 3.7: Metaheuristics applied to other telecommunication optimization problems (I)

Article	**Optimization problem**	**Single solution**	**Population based**	**Notes**
Soo-Hyun et al. (2000)	Allocation resource	SA		Poor Relative quality as the traffic load increases
Wille et al. (2005)	IP networks	TS	GA	Algorithm tested on a 40-nodes network
Fidanova et al. (2014)	Positioning problem		ACO	Execution time influenced by the number of ants
Ohatkar and Bormane (2014)	Allocation resource		GA	Algorithm tested on a 25 cell array
Bedoui et al. (2014)	Frequency assignment	Hybrid		Runtime
Cabrera et al. (2014)	Allocation resource		Hybrid	Evaluation of greedy solutions and resource usage
Das et al. (2015)	Positioning problem		TCO	Increasing the number of termite may reduce the size of the coverage
Mazza et al. (2016)	Allocation resource		Hybrid	Algorithm tested with a small size network

Table 3.8: Metaheuristics applied to other telecommunication optimization problems (II)

Article	**Optimization problem**	**Single solution**	**Population based**	**Notes**
Kutucu et al. (2016)	Allocation Resource	SA		Algorithm tested on a 96-nodes network
Suliman et al. (2016)	Channel allocation		AIS	Algorithm tested on a 100-nodes network
Chen et al. (2016)	Allocation resource		GA	Algorithm tested on a 64-nodes network
Messaoud et al. (2017)	Interference channel		PSO BCO	Scenarios with 5 antennas at transmitters and receivers
Benmezal et al. (2017)	Antenna positioning	LS		rate of coverage area
chakraborty et al. (2017)	Channel allocation		GA	analysis based on total number of 14 available channels
Mirsaleh and Meybodi (2018)	Cells assignment		Hybrid	Network with up to 100 nodes and 5 switches
He et al. (2018)	Channel allocation		GA	analysis based on total number of 6 users

methodology to assign flow and capacities under end-to-end QoS constraints. The solving approach is based on the use of a GA and a TS. Likewise, Kutucu et al. (2016) consider the band collocation problem which may find an application in telecommunication networks, to design an optimal packing of information flows on different wavelengths into groups for obtaining the highest available cost reduction using WDM technology. The authors proposed a SA algorithm tested on a set of instances with a network up to 96 base stations.

With the development of new technologies, Internet-distributed computing has become increasingly popular, which has brought new emergent paradigms. However, the scarce availability level of non-dedicated resources constitutes an important challenge to the range of possible applications of these systems. Cabrera et al. (2014) proposed a simulation-optimization approach for cost-effective and availability-aware service deployment. Likewise, Mazza et al. (2016) dealt with the resource allocation problem for supporting fast access of mobile devices. They presented a BR-based heuristic to support efficient and fast link selection.

Table 3.7 summarizes the information reviewed in this section. The first column indicates the article in which the metaheuristic-based methodology was employed. The next two columns (Single-solution and Population-based) indicates the classification of the metaheuristic in hand. And the last column, denoted by "Notes" gives a brief description of the limitations found on the analyzed article. Notice the diversity of metaheuristics that have been used to deal with these problems, including hybrid ones.

3.4 Concluding Remarks

This chapter offered a review of recent works proposing different metaheuristic approaches to efficiently deal with the aforementioned challenges. Three main telecommunication fields have been the focus of this chapter: networks design, routing, and allocation. Inside these fields several sub-problems have been identified, since the design of telecommunication systems comprises many different aspects, including: topologies, location of resources in links and nodes, uni-cast vs. multi-cast routing, reliability and availability of networks, etc. From the literature review, it can be concluded that, so far, population-based metaheuristics –such as Genetic Algorithms, Ant Colony Optimization, and Particle Swarm Optimization– have been more popular than single-solution ones –such as Tabu Search, GRASP, or Simulated Annealing. However, this also offers a good chance to researchers in other fields –e.g., Operations Research or Industrial Engineering– to propose new single-solution approaches that typically require less parameters and might be easier to implement in real-life scenarios.

Moreover, it should be noted that studies have been done on a considerable range of HLPs, however, some of the novel formulations and their solution techniques are not pertinent to real-world problems since they consider small size instances. In the following, a hub location related problem: the USApHMP is subject of study.

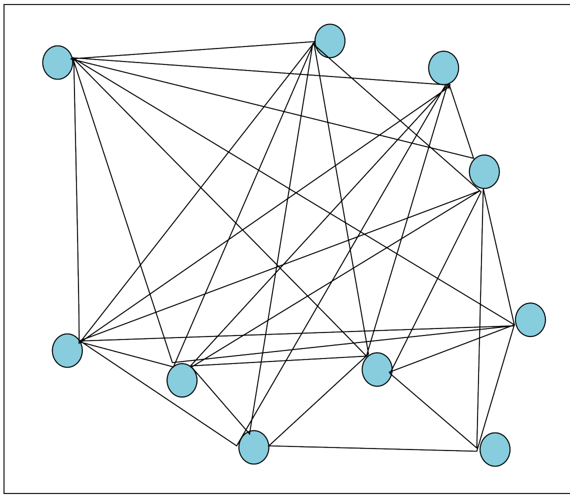
(II) APPLICATIONS

CHAPTER 4

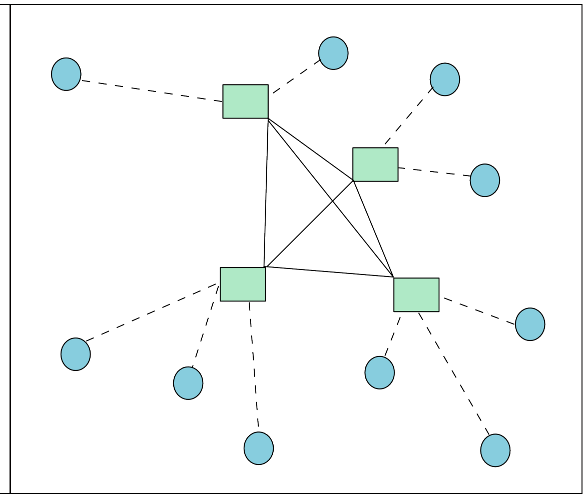
DETERMINISTIC VERSION OF THE UNCAPACITATED SINGLE-ALLOCATION p-HUB MEDIAN PROBLEM

Hub location related problems have gained attention over the last decades. In general, the HLP can be stated as follows: given a set of nodes, their locations, an origin-destination pair traffic requirements matrix specifying the level of interaction between each pair of nodes, and the number of hubs that are to be employed, p, determine the location of hubs and the assignment of the non-hub nodes to them to enable the interaction of the non-hub nodes at minimum cost. Non-hub nodes are connected directly to hubs, and hubs are directly connected to one another. Links between non-hub nodes are not allowed, so all traffic must be routed via, at least, one hub.

In telecommunication networks, a hub is a place of concurrence where the work of the network is centralized with the purpose of delivering out the data that arrives from one or more directions to other destinations. In this sense, hubs are applied to decrease the number of transportation links between origin and destination nodes. For example, a fully interconnected network with n nodes and without hub nodes has $n(n-1)$ origin-destination links as illustrated in Figure 4.1a. However, if a hub node is selected to connect all other nodes (i.e., non-hub nodes also known as spokes) with each other, there will be only $2(n-1)$ connections to serve all origin-destinations pairs. This idea can be extended to a network with more than a unique hub node, sometimes referred as a multiple-hub network, such as the



(a) A fully interconnected network with n nodes



(b) A multiple-hub network

Figure 4.1: Comparison between a fully interconnected network with non-hub nodes and a multiple-hub network

one shown in Figure 4.1b. Note that by using fewer resources, demand pairs can be served more efficiently with a hub network than with a fully connected structure.

Costs of a hub network depend on its proper structure. The total distances of links (known as arcs) connecting the whole pairs of origin-destination points might be less in a hub network, but the total travel distance may be larger since the number of people, commodities, merchandise, or information moving on the hub-to-hub connections could be greater than those moving between hub and spokes. It seems that the telecommunication industry is originally one of the oldest user of hub network concept. However, logistics systems, airline industry, and postal companies are main users of this concept on these days.

There are many variants of HLP (O'kelly 1987): the p-hub median problem, the p-hub center problem, the capacitated/uncapacitated hub location problem, and the hub covering problem. In the p-hub median problem, which has applications on telecommunication networks and transportation, the objective is to minimize the total cost of movement flow (data, packets, passengers, etc.). In the p-hub center problem the main objective is to minimize the maximum cost (or distance) between each pair of requested points. The capacitated/uncapacitated hub location problem differs from the p-hub median one in that the number of hubs is not known beforehand, and a fixed cost is associated with each potential hub location. The covering hub problems are particularly applied to the delivery of time-sensitive items.

Moreover, hub location problems may be classified by the way in which the requested points are assigned or allocated to hubs. In this sense, they can assume one of the two allocation schemes: *(i)* single allocation scheme, where each node must be assigned to exactly one hub node (i.e., all flows from/to each node go only via assigned hub); and *(ii)* multiple allocation scheme, where nodes are allowed to communicate with more than one hub. Furthermore, different constraints may arise, including capacities restrictions on the volume of traffic a hub can concentrate. A straightforward version of this problem is the uncapacitated hub location problem, which assumes that the capacity of each hub is virtually unlimited or, at least, far beyond the expected demand. In these types of configurations, the hubs serve as connection points between two installations, allowing to replace a large amount of direct connections between all pair of the nodes, hence, minimizing the total transportation cost of the network. As expressed on Chapter 3, since network design problems are time consuming and hence expensive, there is a critical need for efficient approaches to support the proper design decisions in short times.

By relying on the above explanations, in this chapter, the USApHMP is addressed by a two-stage metaheuristic. The developed metaheuristic will be referred as the Biased Randomization Iterated Local Search, or BRILS as an abbreviation. Additionally, two algorithms inspired by AIS are proposed to solve the same problem, namely CLONALG and optAiNet. In short, the most significant contributions of this chapter to the literature are: (*i*) developing an efficient metaheuristic algorithm by including BR techniques into the framework of ILS; (*ii*) putting forward two promising algorithms based on AIS methodology, and (*iii*) providing numerous experiments to show the quality of these proposals and pointing out several promising directions to future research.

4.1 Literature Review

The USApHMP was initially presented by O'kelly (1987) in order to study airline passenger flow in which overall transportation cost between origin and destination points is minimized. The authors were the first to introduce the CAB data set, which is based on airline passengers interactions between 25 US cities in 1970. This data has become the most used by the hub location researchers, although nowadays can be considered as medium-size network.

Few years later, the multiple allocation p-hub median problem was introduced by Campbell (1992). Later, the same author presented a linear model approach for solving the single allocation p-hub median problem (Campbell 1994). He also formulated the problem with flow thresholds, which is defined as the minimum flow value needed to allow service on a link. When flow thresholds are set to their maximum values, each demand node is assigned to a single hub and the formulation reduces to the single allocation p-hub median problem. After that, Skorin-Kapov et al. (1996) proposed new very tight linear programming formulations for the multiple and single allocation p-hub median problem. The authors showed that the LP relaxation of Campbell (1994) formulation resulted in highly fractional solutions. They presented the optimal solutions for the CAB data set by using CPLEX solver (IBM inc.). Ernst and Krishnamoorthy (1998) presented an integer linear formulation for single allocation p-hub median problem with fewer variables and constraints in an attempt to solve larger problems, the authors employed SA algorithm for solving problem up to 50 nodes. They were the first authors to use the AP data set, which is based on a postal delivery in Sydney, Australia, and consists of 200 nodes representing the postal districts.

Ebery (2001) investigated the single allocation p-hub median problem in the presence of two or three hubs where the locations of hubs are fixed. Authors solved these problems using new mixed integer linear programming formulations with the AP data set up to 200 nodes. However, the computational time required to solve their new formulation was greater than the required to solve the formulation in Ernst and Krishnamoorthy (1998). Also, the formulation was only effective at solving problems with at most three hubs. In the heuristic side, Chen and Wu (2008) dealt with the USApHMP by adapting a hybrid heuristic from previous research which was designed mainly for the multiple allocation version. The adapted heuristic used a tabu list in its SA algorithm to overcome the problem which was tested using the AP data set.

Likewise, Ilić et al. (2010) introduced a General VNS (GVNS) algorithm to solve the USApHMP which is tested in instances with up to 400 nodes and reported new results for a few problems with up to 1000 nodes. Similarly, Kratica (2013) solved this problem by an electromagnetism-like metaheuristic, which can solve nonlinear global optimization problems converging rapidly to an optimum. The author demonstrated this approach is capable to solve the problem up to 1000 nodes. Alternatively, Peiró et al. (2014) proposed a heuristic approach based on GRASP metaheuristic that employs three local search procedures to solve the uncapacitated r-allocation p-hub median problem, in which every node is assigned to r of the selected p hubs ($r \leq p$). The proposed algorithm was tested in instances with up to 200 nodes. They introduced a mechanism to eliminate low-quality solutions during the greedy phase. Therefore, they selectively apply local search to promising solutions. The authors tested their algorithm in the USA423 data set, which is based on real airline data with up to

423 nodes.

Furthermore, Martí et al. (2015) demonstrated that their Scatter Search-based methodology was superior compared to GRASP methodology when solving uncapacitated r-allocation p-hub median problem. They also enhanced their algorithm by hybridization with Path Relinking. They presented solutions with the CAB, AP and USA423 data sets. The later, is based on real airline data concerning 423 cities in the United States. Later, Amin-Naseri et al. (2016) introduced a robust bi-objective USApHMP in which travel time has non-deterministic nature. They developed a hybrid metaheuristic based on Scatter Search and Variable Neighborhood Descent tested in problems up to 50 nodes. Rostami et al. (2015) addressed a new version of the USApHMP in which transportation costs of each edge are given by piecewise constant cost functions. Authors proposed an exact Branch-and-Bound algorithm applied in instances with up to 50 nodes.

Recently, Sun et al. (2017) presented an evaluation of different methods from the state-of-the-art for solving five problems belonging to the hub location problem, among them the single and multiple allocation p-hub median problem. The techniques authors chose for studying the scalability and solution qualities were GA, Lagrangian Relaxation (LR) and Restricted Clustering (RC) based methods were applied. The authors used as case studies three data sets: Turkish Postal System, AP and CAB. Authors concluded that GA provides good solution to single allocation problems as LR to multiple allocation problems. RC methodology provided solutions within a shorter time than the others, however, the optimality of the solution could not be guaranteed.

Recent successful approaches using exact methods are Meier and Clausen (2015) and Meier et al. (2016). In these works, linearizations from the Euclidean structure presented in famous instances of hub location problems are constructed, leading to compact formulations with few variables. The authors presented results for instances with up to 200 nodes. For a more extensive literature review on the HLP, the reader is addressed to Alumur and Kara (2008), in where the authors reviewed over 70 articles on hub network optimization, and more recently to Farahani et al. (2013), in which the authors review models, classifications, solution techniques and applications of hub location problems. In order to provide an overview of the literature review related to the p-hub median problem, the paramount feature of each studies mentioned above are encapsulated in Table 4.1.

Although the USApHMP has received various attention from the heuristics and exact methods community, and ILS has been applied to other variants of HLP, no BR techniques implementation exists for the USApHMP investigated here. This chapter aims to bridge

Table 4.1: Deterministic p-hub median literature

Article	**Classification**	**Characteristics**
Campbell (1994)	Single allocation	Different types of formulations
Skorin-Kapov et al. (1996)	Multiple and Single allocation	Very tight formulations
Ernst and Krishnamoorthy (1998)	Single allocation	Integer formulations
Ebery (2001)	Single allocation	Mixed integer formulations only with 3 hubs
Chen and Wu (2008)	Single allocation and uncapacitated	Hybrid metaheuristic
Ilić et al. (2010)	Single allocation and uncapacitated	GVNS algorithm
Kratica (2013)	Single allocation and uncapacitated	Electromagnetism-like metaheuristic
Peiró et al. (2014)	r allocation and uncapacitated	GRASP
Martí et al. (2015)	r allocation and uncapacitated	Scatter search-based algorithm
Amin-Naseri et al. (2016)	Single and uncapacitated	Hybrid metaheuristic
Rostami et al. (2015)	Single and uncapacitated	Branch-and-bound algorithm
Sun et al. (2017)	Single and Multiple allocation	Tested three different approaches
Meier and Clausen (2015), Meier et al. (2016)	Hub location problem	Exact methods

this gap by proposing the BRILS approach to solve USApHMP. The algorithms designed applying BR of classical heuristics allows to obtain "high-quality" solutions to realistic problems in reasonable computing times. Moreover, they tend to use a reduced number of parameters, which make them simple to implement and set up in most practical applications. Also, CLONALG and optAiNet are proposed to solve the same problem. In contrast with standard GA and other bioinspired proposals which have been applied to solve other variants of the problem, immune inspired algorithms such as CLONALG and optAiNet present an intrinsic capacity of maintaining diversity of solutions during the execution, which can be decisive to increase the probability that the global optimum or a good local optimum be reached. Although it is known that there is an specific global search strategy with superior performance for a wide class of problems, these arguments support the application of such metaheuristics in the context of USApHMP.

4.2 Formal Problem Description

The USApHMP consists in selecting p nodes of a given network as hubs and allocating the remaining nodes to them, such that each non-hub node is allocated to a single hub and the overall transportation cost is minimized. The number of required hubs to locate is given in advance. The mixed integer linear programming formulation proposed by O'kelly (1987) is used in this work, since it has been widely applied for solving from small to large instances of the problem.

According to this formulation, $N = \{1, \cdots, n\}$ is a set of n distinct nodes in the network, where each node refers to origin/destination points or possible hub location. The distance between each origin node i and destination node j is C_{ij}, and w_{ij} is the amount of flow from node i to node j. A variable $H_{ij} \in \{0, 1\}$ has value 1 if node i is allocated to a hub node j and 0 otherwise. The condition $H_{kk} = 1$ implies that the node k is a hub. Given that hub nodes are fully interconnected, every link between an origin node and a destination node will contain at least one hub (in case a mail is sent from postcode district to itself) and at most two hubs (in case the origin and destination nodes belong to different postcode districts). Thus, parameters $\mathscr{X}$, τ and δ represent the unit rates (costs) for collection (origin-to-hub), transfer (hub-to-hub) and distribution (hub-to-destination) along the path. Generally, τ is used as a discount factor to provide reduced unit costs on arcs between hubs to reflect scale savings, so $\tau < \mathscr{X}$ and $\tau < \delta$. The factor τ was originally proposed by O'kelly (1987) to represent scale savings on the transportation cost between hubs; the two remaining factors, $\mathscr{X}$ and δ were later introduced by Ernst and Krishnamoorthy (1998) to represent the reality of postal service costs. Given the notation above, the problem is formulated as follows:

$$\text{(4.1a)} \quad \text{minimize} \quad \sum_{i,j,k,l \in N} w_{ij}(\mathscr{X} C_{ik} H_{ik} + \tau C_{kl} H_{ik} H_{lj} + \delta C_{lj} H_{lj})$$

$$\text{(4.1b)} \quad \text{subject to:} \quad \sum_{k=1}^{n} H_{kk} = p,$$

$$\text{(4.1c)} \quad \sum_{k=1}^{n} H_{ik} = 1, \qquad \text{for all } i = 1, \cdots, n,$$

$$\text{(4.1d)} \quad H_{ik} \leq H_{kk} \qquad \text{for all } i,k = 1, \cdots, n,$$

$$\text{(4.1e)} \quad H_{ik} \in \{0,1\} \qquad \text{for all } i,k = 1, \cdots, n,$$

The objective function 4.1a guides the search for the minimum total cost of origin-to-hub, hub-to-hub, and hub-to-destination flow costs multiplied by $\mathscr{X}$, τ and δ factors respectively. Constraint 4.1b specifies that the exactly given number of p hubs are found. Constraint 4.1c ensures that the flow from any node i is sent through exactly one hub node in which the node i is allocated. By constraint 4.1d we prevent nodes being allocated to hubs that are not being activated. Constraint 4.1e enforces that the flow is sent only via activated hubs, thus avoiding direct transfer between non-hub nodes. Figure 4.2 illustrates the problem given a network of $n = 20$ nodes and $p = 4$ hubs. In the example, origin node i sends the flow w_{ij} to the destination node j through the hubs every node is allocated.

4.3 A Two-Stage Biased Randomized Iterated Local Search for Solving the USA*p*HMP

4.3.1 Solving Methodology

Two main ideas are the basis of our approach. Firstly, using a simple (with few parameters to avoid long fine-tuning processes) and fast heuristic that allows the generation of feasible solutions and selection of the most promising ones. Secondly, considering all decisions together at each phase of the algorithm, i.e., tackling the p hubs location taking into account assignment costs when deciding the nodes allocated to the p hubs, as well as the transportation costs of sending the flow traffic from the origin node to its destination (or at least an estimated cost).

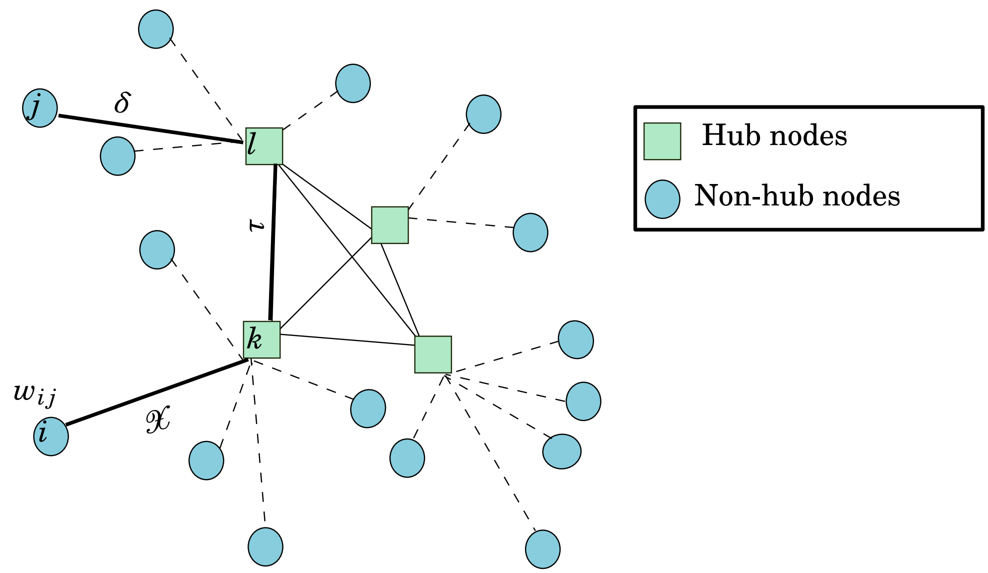


Figure 4.2: Illustrative example of the USApHMP with a network of $n = 20$ and $p = 4$

As a result of combining these two ideas, we have developed an effective algorithm that can be repeatedly executed without consuming too much computational time.

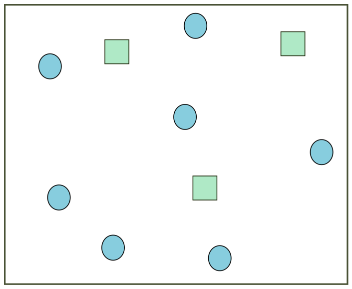
The use of BR techniques inside a constructive heuristic to obtain new solutions is an appealing aspect of our approach. It means that some bias is introduced into the random process in order to guide the selection of a movement, in such a way that the basic logic behind the deterministic heuristic is conserved (Grasas et al. 2017, Juan et al. 2013). If higher probabilities of selection are assigned to the most promising movements during the constructive phase of the heuristic, the logic of the original procedure is respected. Indeed, asymmetric probability distributions, for instance, the geometric distribution or the decreasing-triangular one, are preferred (Juan et al. 2015*b*,*c*). Furthermore, if we execute it several times, we can obtain different solutions with similar costs. Several authors have successfully applied this methodology in literature (Dominguez et al. 2014, 2016*a*,*b*, Juan et al. 2014*b*, Quintero-Araujo et al. 2017). This is the first time that a BR technique is implemented into a solution procedure for the USApHMP.

Additionally, the choice of ILS as building block to tackle USApHMP is motivated by the large number of studies adopting ILS in hub location-like problems (Carello et al. 2004*b*, Teymourian et al. 2011, Davari and Fazel Zarandi 2013, Fazel Zarandi et al. 2015). As explained before in Section 2, ILS is a single-solution metaheuristic that starts applying a

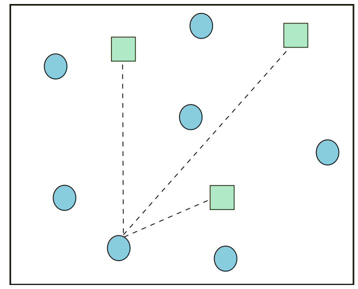
Algorithm 11: BRILS(nodes, *p*Hubs, per, maxIterations, beta)

```
initialSolution ← genRandSolution(nodes, pHubs, connectionCostMatrix, beta)
baseSolution ← initSolution
solutionSet ← {}
numberOfIterations ← 0
credit ← 0
while (numberOfIterations ≤ maxIterations) do
    newSolution ← perturb(baseSolution, connectionCostMatrix, beta, per)
    newSolution ← localSearch(newSolution, connectionCostMatrix)
    delta ← cost(newSolution) – cost(baseSolution)
    if (newSolution in the best n solutions) then
        add(newSolution, solutionSet)
    end
    if (delta ≥ 0) then
        credit ← delta
        baseSolution ← newSolution
        if (cost(newSolution) < cost(bestSolution)) then
            bestSolution ← newSolution
        end
    else
        if -(delta) ≤ 0 then
            credit ← 0
            baseSolution ← newSolution
        end
    end
end
numberOfIterations ← numberOfIterations + 1
bestSolution ← null
for (solution in solutionSet) do
    solution ← fullLocalSearch(solution)
    if (bestSol = null or cost(solution) ≤ cost(bestSolution)) then
        bestSolution ← solution
    end
end
return bestSolution
```

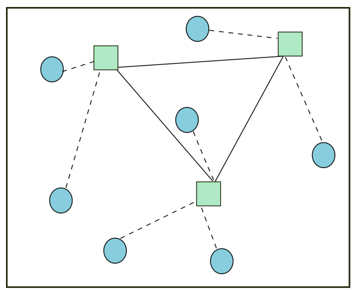
local search to an initial solution. Then, a loop starts where at each iteration a perturbation phase is applied to the current local optima, followed by a local search phase.



(a) Initial distribution of hubs and nodes after Eq. 4.2 is computed for every node i



(b) The biased randomization is applied



(c) Network after the total transportation cost is minimized

Figure 4.3: Generation of the initial solution using BRILS algorithm

Algorithm 11 depicts our integration of BR into the ILS scheme. This allows us to take advantage of BR features to improve the search space generated by the ILS and, thus, to complement the search. The algorithm receives the following input parameters: *(i)* the number of nodes as well as the number of hubs of the problem; *(ii)* the limit percentage of hubs to be deleted from the current base solution during the perturbation phase; *(iii)* the stopping criterion, i.e., the maximum number of iterations to be executed; *(iv)* the parameter β for the geometric distribution. The procedure starts generating an initial solution from function genRandSol (line 1). Figure 4.3 illustrates this function. Firstly, a set of initial hubs are randomly chosen, using a uniform probability distribution as shown in Figure 4.3a. Let k be a candidate location for a hub and that any node i could be assigned to k, then, we consider that all the flow transferred from node i to any node j has to be sent through k. The cost associated with such a flow is given by w_{ij} multiplied by the distance C_{ik} from node i to the assigned hub k. Accordingly, we have the calculation of this assignment, $a(i,k)$, as

$$a(i,k) = C_{ik} \sum_{j=1}^{n} w_{ij}. \tag{4.2}$$

In order to evaluate the potential of k as a possible hub, we compute (4.2) for every node i in the graph. Then, for each non-hub node we add the selected hubs in a list which is sorted according to the $a(i,k)$ by allocating the non-hub node to the chosen hubs. Then, the BR technique is applied by using the geometric distribution to give higher probabilities to the non-hub node with less connection cost, as illustrated in Figure 4.3b. The geometric distribution has only one parameter, β, which in our case represents the probability given to the first element of the list. Consequently, when β becomes closer to 1, the greedy behavior of the heuristic is retrieved. This way, at each step, instead of inserting the "best candidate node for being a hub", all candidates are considered with nodes having better odds of being selected. Notice that no time-costly fine-tuning process is needed here. Then, an array as the bunch of nodes allocated to the same hub is found. Note that for each node j that receives flow from any node i, this flow has to be transferred through their assigned hub. That is, to transport the w_{ij} flow, a path $i \to k_i \to k_j \to j$ will be used. Then, the transportation cost of routing all flows is minimized (see Figure 4.3c).

Algorithm 12: perturb(baseSolution, connectionCostMatrix, beta, per)

```
newSolution ← baseSolution
// destruction phase
hubs ← randSelect(newSolution, per)
newSolution ← delete(newSolution,hubs)
hubs ← randSelect(nodes(newSolution - hubs(newSolution)), per)
newSolution ← add(newSolution, hubs)
// construction phase
for (node in nodes) do
    hubs(node) ← sortByConnectionCost(node, hubs, connectionCostMatrix)
    hub(node) ← biasedRandSelect(hubs(node), beta)
    newSolution ← add(newSolution, hub(node))
end
estimatedCost(newSolution) ← estimateCost(newSolution, connectionCostMatrix)
return newSolution
```

Once the initial solution has been generated, the ILS main loop starts (lines 6-23). Inside it, the current base solution is disturbed in order to improve the non-hub nodes allocations obtained so far, thus generating a new candidate solution (line 7). As it can be seen from Algorithm 12, a percentage *per* of the current base solution is destroyed and then reconstructed to generate a new solution during this stage. In order to do that, we generate a new assignment of nodes to hubs searching into the neighborhood of the current solution. This new assignment is constructed as follows. A random number of r different hubs are chosen. The number of r hubs to be removed is determined by the perturbation operator per. These hubs are excluded from the current base solution. Then, the same number r of hubs are randomly chosen to be included in the current new solution. Finally, in order to reconstruct the solution, the same logic used to generate the initial solution is applied, i.e., the non-hub nodes are sorted in a list depending of the connection cost, the nodes in the list are randomized, and the array of nodes allocated to every hub is found. The use of BR techniques in this phase results appealing because it enhances the performance of the heuristic.

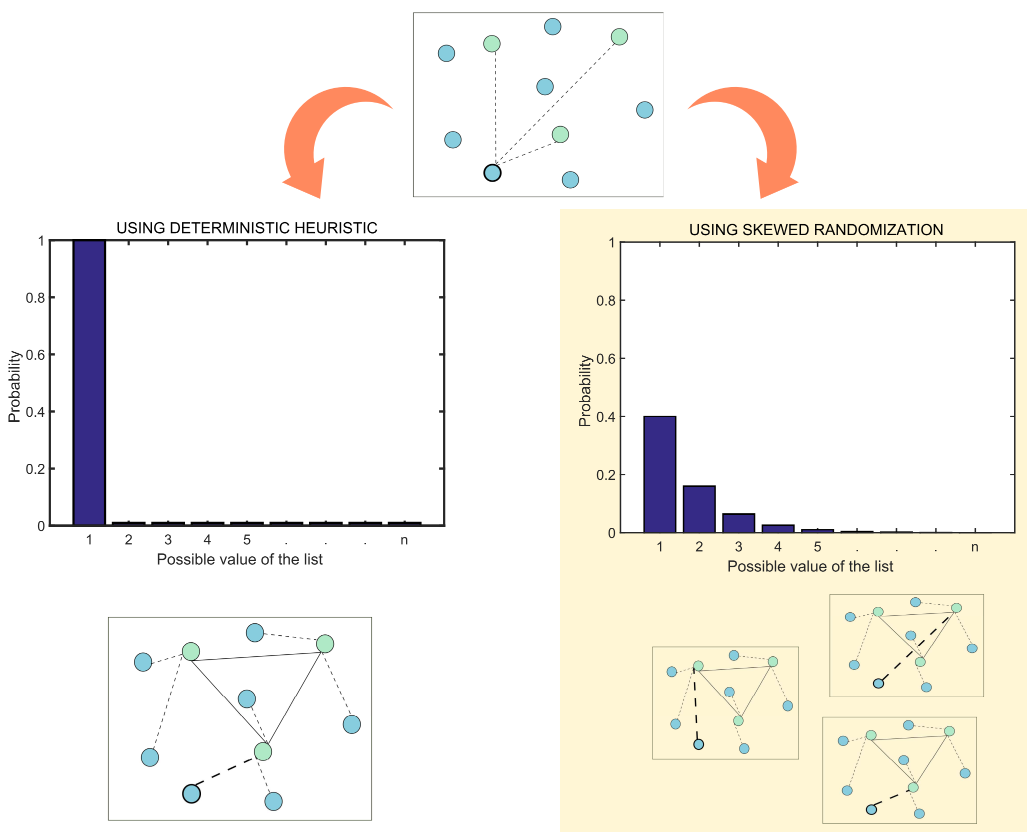


Figure 4.4: Use of skewed distributions to generate alternative solutions

A deterministic heuristic provides one possible quality solution . However, as discussed by Grasas et al. (2016), by randomizing the deterministic approach using a skewed probability distribution, we can go further and obtain different quality solutions while keeping the logic behind it (Figure 4.4). As already stated in Chapter 2, in other approaches such as the GRASP (Feo and Resende 1995), the "best next" elements to choose in each step are collected in a RCL, and then the next element to include in the partial solution is randomly selected using a uniform distribution. However, this approach requires to determine the proper size of the RCL and it is not necessary if we use the BR technique, that seems to perform equal or even better than GRASP, as depicted in (Grasas et al. 2016).

Algorithm 13: localSearch(newSolution, connectionCostMatrix)

```
for (node in nodes(newSolution)) do
    for hub in hubs(newSolution - hub(node)) do
        savings ← distance(node, hub)*(collectionCost*totalOutFlow(node) +
          distributionCost*totalInFlow(node))
        savings ← savings - distance(node, hub(node))
          *(collectionCost*totalOutFlow(node) + distributionCost*totalInFlow(node))
        if (savings < 0) then
            hubs(node) ← hub
            estimatedCost(newSolution) ← estimatedCost(newSolution) + savings
        end
    end
end
return newSolution
```

Then, a local search process is carried out (line 8). This procedure is described in Algorithm 13. It is a simple and easy-to-implement procedure. By following the steps in the procedure the solution is moved approaching the closest local optima by using a neighborhood search scheme. The local search tries to improve the solution with a one-trade in each iteration. During the one-trade step the procedure tries to alternate the selected hub k with one of the remaining hubs in the current new solution. After that, the estimated cost of the solution containing the new hub configuration is compared to the solution that contains the original hub. Notice that the concept of *savings* is proposed, which is associated with the estimated cost of sending flow from the origin node i to the destination node j through their hubs yet ignoring the transfer (hub-to-hub) cost τ (by ignoring the transfer cost the strategy becomes much less computational expensive). If the savings value is smaller than 0, one-trade is performed, otherwise a new hub configuration is tested. Within the proposed local search procedure, every time an improvement is detected, this is rapidly performed and the procedure continues. If, for each node, one-trade produces a savings value greater than zero, the local search procedure concludes with no improvements. Next, the new solution obtained is compared against the current base solution. Notice that, at any time the *newSolution* is found, this is saved in solutionSet (line 9-11). The acceptance criterion (line 12-19) uses the *credit* strategy, which means that, whenever an improvement in the current solution is performed, a credit is assigned by the algorithm. This credit has the same value of the improvement and limits the total length that the current base solution gets worse in the following iteration. Then, for every *solution* contained in solutionSet, a local search procedure denoted as fullLocalSearch and presented in Algorithm 14 is applied. This procedure differs from the one carried out in line 8 in that the fullLocalSearch takes into account the objective function value as in 4.1a, i.e., the sum of origin-to-hub, hub-to-hub and hub-to-destination flow costs multiplied by $\mathscr{X}$, τ and δ factors respectively. Finally, if the former *solution* is better than the *bestSolution*, then the latter is updated to proceed with the search from a more likely point within the solution space. At any time, the best solution found is updated and saved, thus, this is the solution returned at the end. Note that by running the algorithm multiple times or until a limited computation time, each run will produce different solutions in a fast way and one can take the best from the set of solutions generated.

Algorithm 14: fullLocalSearch(newSolution)

```
for (node in nodes(newSolution)) do
    for hub in hubs(newSolution - hub(node)) do
        oldHub ← hubs(node)
        oldCost ← cost(newSolution)
        hubs(node) ← hub
        if (cost(newSolution) > oldCost) then
            hubs(node) ← oldHub
        end
    end
end
return newSolution
```

4.3.2 Experiments and Results

In this section, the computational experiments to test the performance of the proposed BRILS algorithm are illustrated. The procedures in this method have been implemented as Java® 7SE applications. The present final results were obtained with an Intel® Xeon E5-2630 v4 at 2.20 GHz running the Ubuntu 14.04 operating system.

In order to investigate the efficiency of the BRILS algorithm, the AP data set has been used. As mentioned before, the AP data set was first introduced by Ernst and Krishnamoorthy (1998) and is derived from a postal delivery system in Australia. Accordingly, an evaluation involving nodes (representing postcode districts), as well as their coordinates and flow volumes is performed. The size of the original data file is 200 nodes. Smaller instances can be obtained using a C® program from ORLIB. According to previous works, the values of the collection (node-to-hub), transfer (hub-to-hub), and distribution (hub-to-node) costs have been set to $\mathscr{X} = 3$, $\tau = 0.75$ and $\delta = 2$, which holds $\mathscr{X} > \delta > \tau$. The AP data set problems have asymmetric flows ($w_{ij} \neq w_{ji}$) and a mail can be sent from postcode district to itself, i.e., $w_{ii} \neq 0$.

For each instance, 20 different seeds generated randomly were employed in order to execute the test of the BRILS algorithm. Before starting the complete analysis of our algorithm, we run several executions so we could empirically define appropriate values for the following parameters in terms of solutions quality and time consumed: (*i*) Maximum number of iterations (*maxIterations*) = 10000; (*ii*) Geometric distribution parameter for biased randomized allocation of the nodes to the hubs (β) = $0.1 \leq \beta \leq 0.3$; (*iii*) Percentage of hubs to remove (*per*) = $0.05 \leq per \leq 0.1$; and (*iv*) Maximum number of best solutions (*solutionSet*) = 50.

The computational results of the proposed BRILS on small and large instances are presented in Tables 4.2 and 4.3. On one hand, Table 4.2 shows a comparison of BRILS with state-of-the-art heuristics methods: SA method proposed by Ernst and Krishnamoorthy (1998), a SA and TL hybrid heuristic SATL considered by Chen and Wu (2008), and a GVNS heuristic presented by Ilić et al. (2010). This table is organized as follows: (*i*) The first two columns show the number of nodes represented with n and the number of hubs p to be selected; (*ii*) The third column, denoted by *Cost (1)* gives the best known solutions (BKS) for the given instances. Bold values represent the optimal solutions of the current solution, if it is previously known, if they are not optimal, is is not highlighted; (*iii*) The next three columns (*Cost (2)*, *Time* and *Gap (1)-(2)*) contain the best solution value obtained by SA algorithm, its average running time and the percentage gap with respect to the BKS given in column *Cost(1)*, respectively; (*iv*) The following seventh, eighth and ninth columns denoted by *Cost(3)*, *Time* and *Gap (1)-(3)* show the reported best solutions and average running time from the SATL algorithm as well as the percentage gap with respect to the BKS; (*v*) The next three columns (*Cost (4)*, *Time* and *Gap (1)-(4)*) indicate the solutions, average running time and percentage gap of the GVNS with respect to the BKS; (*vi*) The last three columns denoted by *Cost (5)*, *Time* and *Gap (1)-(5)* give the found solution, time needed to obtain the corresponding solution and the corresponding percentage gap with respect to the BKS found when applying our proposed BRILS heuristic for the aforementioned sets of benchmark instances. On the other hand, Table 4.3 presents the solution found by the BRILS algorithm in terms of the hubs found. Regarding the size of the instances, they have been classified as small ($10 \leq n \leq 50$) and large ($100 \leq n \leq 200$). The elapsed times for each heuristic are shown in seconds and we calculate the percentage gaps as follows:

$$Gap\ (Cost, Cost_{BKS}) = 100 \times \left(\frac{Cost - Cost_{BKS}}{Cost_{BKS}}\right). \tag{4.3}$$

From the results presented in the first part of Table 4.2, it follows that BRILS is capable of obtaining the optimal solutions for every small-size AP problems, while the SA solutions are almost always optimal but for the instance with sizes $n = 40$ and $p = 5$. Furthermore, regarding the running time, the results with BRILS are significantly better than those obtained by SA and SATL. In this regard, the results obtained by BRILS have a similar behavior with respect to the ones provided by GVNS.

The second part of Table 4.2 shows the computational results for large instances. From the results, one can see that BRILS obtains high-quality solutions. The results provided by the GVNS, which were slightly superior than those obtained by BRILS, were based on running the algorithm 20 times each with different random numbers. Moreover, the GVNS used a shaking parameter $k_{max} = p$, the stopped rule were defined as $n/2$ iterations, and also three local search procedures l_{max} were implemented: *(i) allocate* which randomly change the clusters; *(ii) alternate* which change the hub in the cluster and *(iii) locate* which split clusters adding a new hub. By contrast, BRILS uses only one local search procedure and the parameter settings were exactly the same as solving the small size AP instances. On average terms, one can conclude that the best solutions obtained by SA and SATL are improved by BRILS and its runtime is significantly smaller than GVNS, at the cost of an affordable small solution gap with respect to BKS (see Figures 4.5 and 4.6).

Table 4.2: Comparison of results obtained by the BRILS algorithms and best known solutions for the AP instances

Instance		BKS	SA (Ernst *et. al.* '98)			SATL (Chen *et. al.* '08)			GVNS (Ilić *et. al.* '10)			BRILS		
n	*p*	*Cost (1)*	*Cost (2)*	*Time*	*(1)-(2) %*	*Cost (3)*	*Time*	*(1)-(3) %*	*Cost (4)*	*Time*	*(1)-(4) %*	*Cost (5)*	*Time*	*(1)-(5) %*
10	2	**167493.06**	167493.06	0.25	0.00	167493.06	0.05	0.00	167493.06	0.00	0.00	167493.08	0.01	0.00
	3	**136008.13**	136008.13	0.50	0.00	136008.13	0.08	0.00	136008.13	0.00	0.00	136008.14	0.01	0.00
	4	**112396.07**	112396.07	0.85	0.00	112396.07	0.08	0.00	112396.07	0.00	0.00	112396.08	0.01	0.00
	5	**91105.37**	91105.37	1.34	0.00	91105.37	0.09	0.00	91105.37	0.00	0.00	91105.38	0.01	0.00
20	2	**172816.69**	172816.69	1.10	0.00	172816.69	0.16	0.00	172816.69	0.00	0.00	172816.69	0.02	0.00
	3	**151533.08**	151533.08	1.67	0.00	151533.08	0.20	0.00	151533.08	0.00	0.00	151533.08	0.02	0.00
	4	**135624.88**	135624.88	2.56	0.00	135624.88	0.28	0.00	135624.88	0.00	0.00	135624.88	0.03	0.00
	5	**123130.09**	123130.09	3.61	0.00	123130.09	0.33	0.00	123130.09	0.00	0.00	123130.09	0.03	0.00
25	2	**175541.98**	175541.98	2.14	0.00	175541.98	0.20	0.00	175541.98	0.00	0.00	175541.98	0.03	0.00
	3	**155256.32**	155256.32	2.65	0.00	155256.32	0.50	0.00	155256.32	0.00	0.00	155256.32	0.03	0.00
	4	**139197.17**	139197.17	3.37	0.00	139197.17	0.45	0.00	139197.17	0.00	0.00	139197.17	0.03	0.00
	5	**123574.29**	123574.29	4.33	0.00	123574.29	0.55	0.00	123574.29	0.00	0.00	123574.29	0.03	0.00
40	2	**177471.67**	177471.67	4.34	0.00	177471.67	0.58	0.00	177471.67	0.00	0.00	177471.68	0.04	0.00
	3	**158830.54**	158830.54	6.92	0.00	158830.54	1.06	0.00	158830.54	0.00	0.00	158830.55	0.04	0.00
	4	**143968.88**	143968.88	9.07	0.00	143968.88	1.47	0.00	143968.88	0.00	0.00	143968.88	0.05	0.00
	5	**134264.97**	151719.42	12.08	0.13	134264.97	2.27	0.00	134264.97	0.00	0.00	134264.97	0.05	0.00
50	2	**178484.29**	178484.29	8.68	0.00	178484.29	1.09	0.00	178484.29	0.00	0.00	178484.29	0.05	0.00
	3	**158569.93**	158569.93	12.02	0.00	158569.93	2.38	0.00	158569.93	0.00	0.00	158569.94	0.05	0.00
	4	**143378.05**	143378.05	17.22	0.00	143378.05	2.73	0.00	143378.05	0.00	0.00	143378.05	0.08	0.00
	5	**132366.95**	132366.95	19.53	0.00	132366.95	3.98	0.00	132366.95	0.00	0.00	132366.96	0.06	0.00
100	5	136929.44	136929.44	80.60	0.00	136929.44	24.28	0.00	136929.44	0.08	0.00	136984.70	0.88	0.04
	10	106469.57	106469.57	161.90	0.00	106469.57	109.45	0.00	106469.57	0.67	0.00	106469.57	0.50	0.00
	15	90533.52	90605.10	279.80	0.08	90534.78	156.34	0.00	90533.52	3.22	0.00	90534.79	0.46	0.00
	20	80270.96	80682.71	522.70	0.51	80270.96	298.55	0.00	80270.96	3.58	0.00	80305.10	3.86	0.04
200	5	140062.65	140409.41	399.70	0.25	140472.77	156.80	0.29	140062.65	5.16	0.00	140062.65	1.08	0.00
	10	110147.66	111088.33	776.60	0.85	110254.99	679.30	0.10	110147.66	5.60	0.00	110147.66	0.39	0.00
	15	94459.20	95460.54	1105.30	1.06	94878.63	885.83	0.44	94459.20	17.66	0.00	94695.79	0.87	0.25
	20	84955.33	85560.39	1555.90	0.71	85345.90	1695.08	0.46	84955.33	12.98	0.00	85006.05	9.11	0.06
Average				178.45	0.59%		143.72	0.05%		1.76	0.00%		0.64	0.01%

Table 4.3: Solutions found by BRILS algorithm for the AP instances.

p	***n***						
	10	**20**	**25**	**40**	**50**	**100**	**200**
2	[3,7]	[6,14]	[8,18]	[12,28]	[14,35]	–	–
3	[3,4,7]	[6,12,14]	[7,14,18]	[12,22,28]	[14,28,35]	–	–
4	[3,4,7,8]	[2,6,12,14]	[2,7,14,18]	[12,22,26,28]	[14,28,33,35]	–	–
5	[1,3,4,7,8]	[2,6,12,14,15]	[2,7,14,17,18]	[3,12,22,26,28]	[4,14,28,33,35]	[7,28,55,63,70]	[14,61,113,131,141]
10	–	–	–	–	–	[7,10,18,29,45, 50,57,66,70,75]	[14,35,48,49,60, 99,113,131,140,149]
15	–	–	–	–	–	[7,10,13,16,20, 22,28,36,45,49, 57,66,70,75,85]	[14,19,28,40,45, 56,72,90,99,119,131, 140,149,155,170]
20	–	–	–	–	–	[2,4,7,10,13, 16,20,25,28,36, 42,49,59,63,67, 70,75,78,85,98]	[4,8,14,19,25, 31,40,50,57,72, 83,98,126,134,141, 149,155,159,165,170]

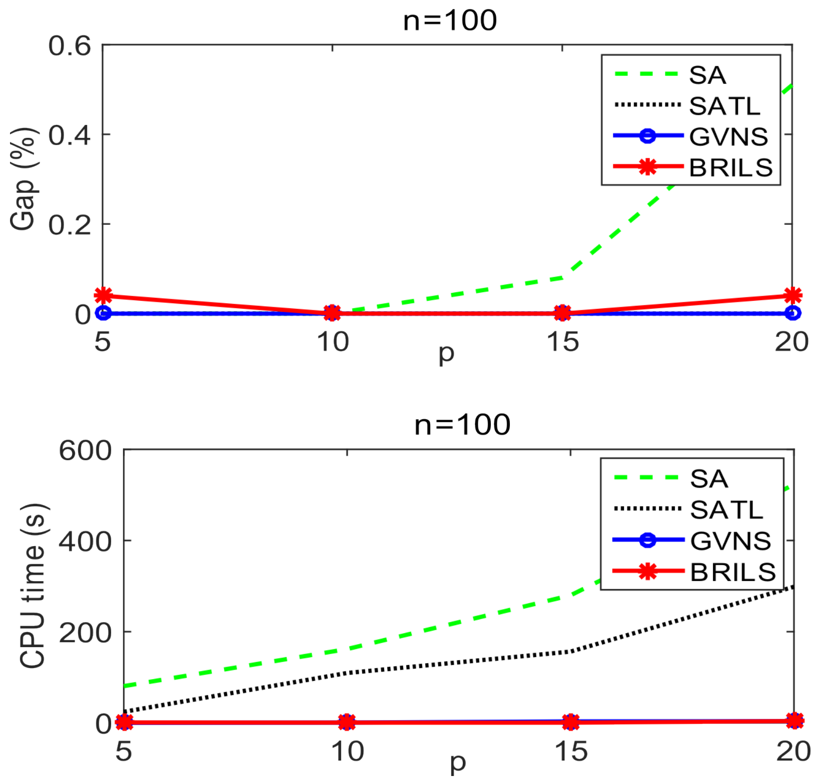


Figure 4.5: Visual comparison among the SA, SATL, GVNS and BRILS heuristics for instances with $n = 100$ and $p = \{5, 10, 15, 20\}$

In order to further demonstrate the attractiveness of the BRILS algorithm, Table 4.3 reports the nodes that were selected as hub nodes in the network. For the small instances the hubs given by the solution correspond to the optimal ones. Moreover, for illustration purposes, Figure 4.7a displays the distribution of the nodes in the network for the instance with $n = 50$ and $p = 5$. In Figure 4.7b each set of nodes assigned to a hub has been represented with a form. The nodes selected as hubs and its assignments given by the solution are illustrated in Figure 4.7c. The position of the nodes in the latter have been changed to better depict the assignments.

To discuss the convergence of the BRILS algorithm under diverse random number seeds, Figures 4.8 and 4.9 show the time evolution of the gaps with respect to the BKS for the instances with $n = \{50, 100, 200\}$ and $p = \{5, 10\}$. The seeds were randomly selected and the results correspond to the procedure explained in Algorithm 14, Section 4.3.1. After reaching the maximum number of iterations the best solutions are submitted through a local search.

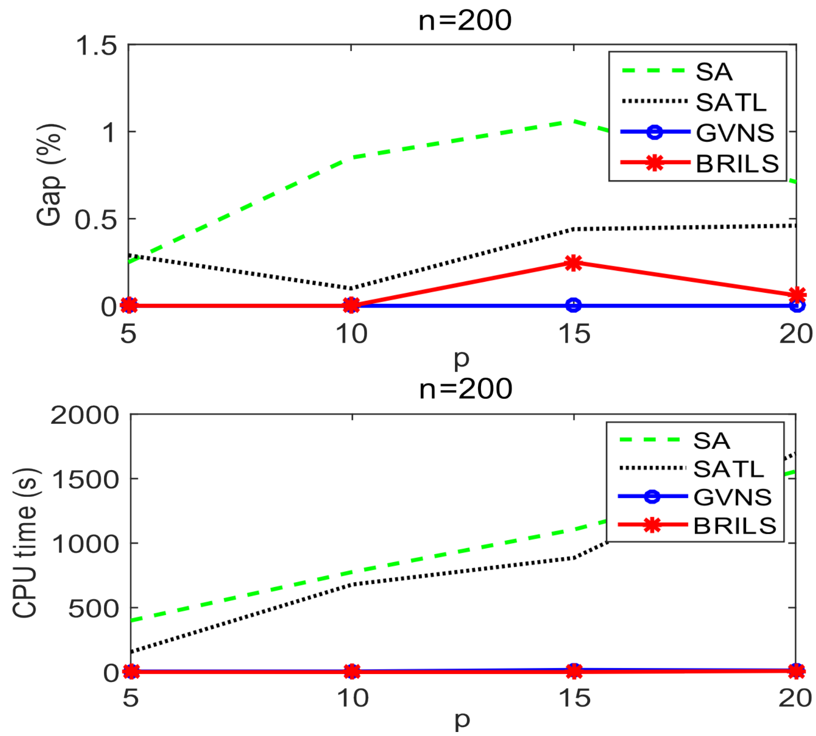
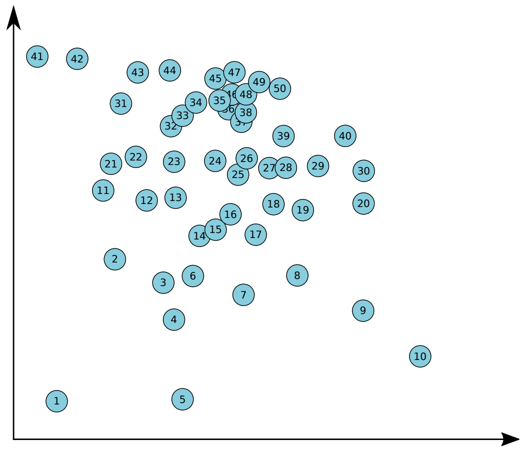


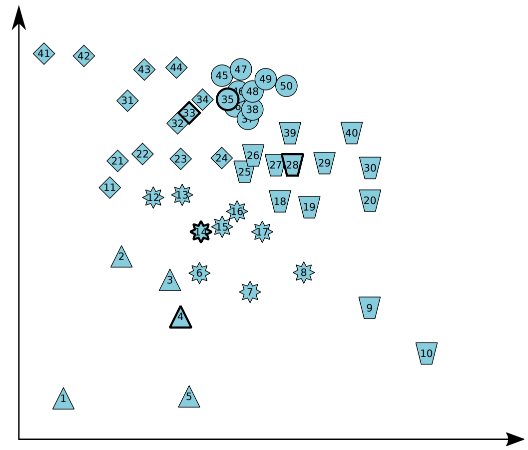
Figure 4.6: Visual comparison among the SA, SATL, GVNS and BRILS heuristics for instances with $n = 200$ and $p = \{5, 10, 15, 20\}$

Notice that BRILS algorithm quickly diminishes the gap in a few computational seconds.

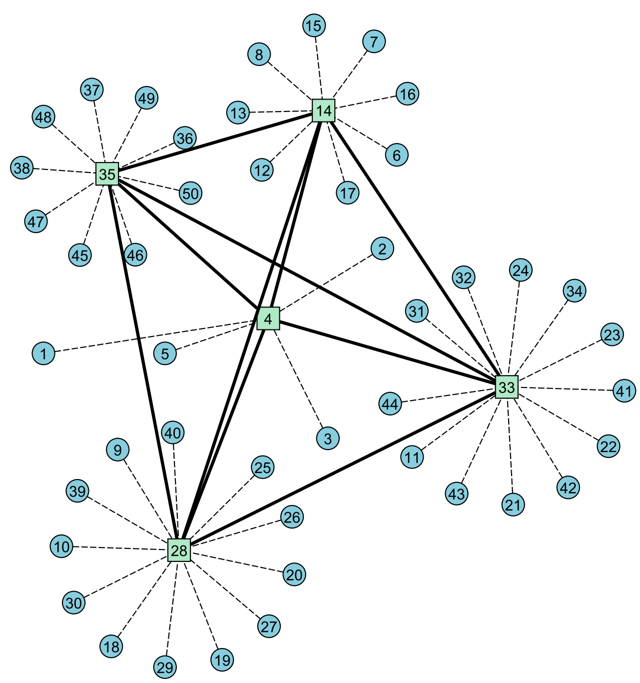
Finally, Figure 4.10 shows a multiple boxplot which allows an alternative comparison of the algorithms performance for the AP data set. Notice again that the computing times of the proposed BRILS algorithm were much shorter than the ones spent by SA, SATL and GVNS, which confirm that the usage of skewed probability distributions in the randomization process within heuristics can provide state-of-the-art results in terms of quality and runtime.



(a) Initial distribution of nodes for the instance $n = 50$ and $p = 5$



(b) Distribution of nodes after BRILS finds the best assigments



(c) Network fully interconneted after BRILS find the best assignments

Figure 4.7: Network with 50 nodes and its structure for $p = 5$

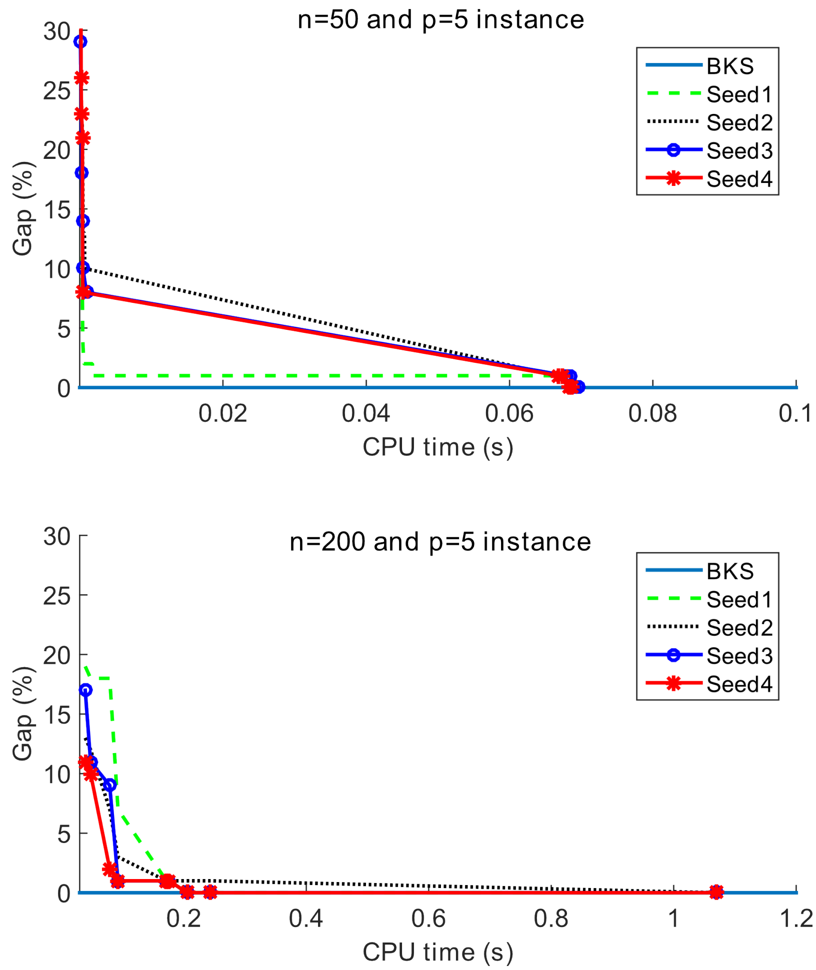
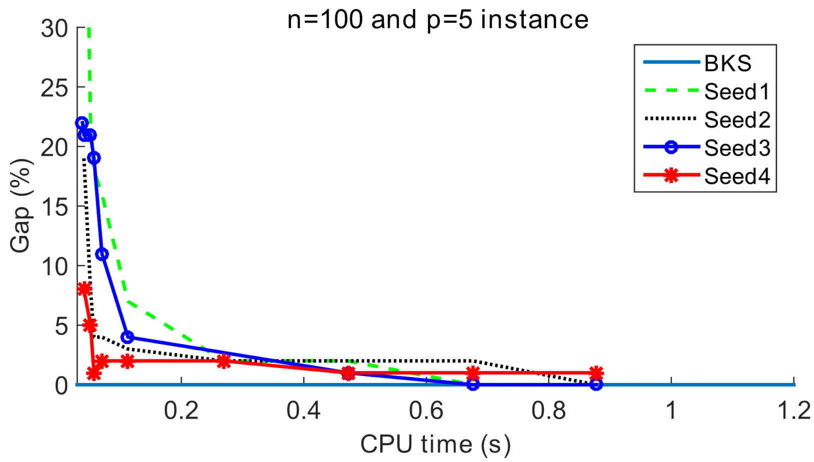


Figure 4.8: Visual performance of BRILS among different seeds for the instance with $n = \{50, 200\}$ and $p = 5$

4.3.3 Discussion

The proposed algorithm was implemented as a Java® 7SE application. Even though Java is a programming language executed in virtual machine (JVM) and we are aware it may show poorer performance like others than C, C++ or Python, the vast amount of tools available in the standard API and its object-orientation eased the development process. In addition, the execution on the JVM offers better replicability and repeatability than other languages.

To test the efficiency of the proposed algorithm, the AP data set was chosen with the criteria of testing the algorithm against instances of small and large size in terms of number



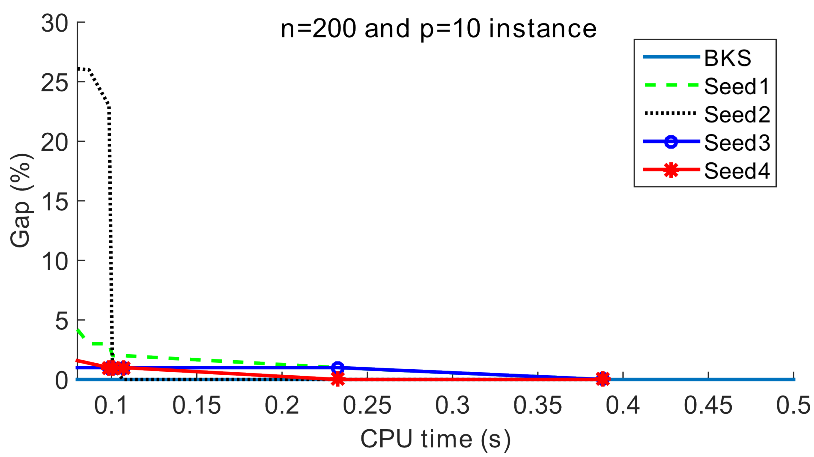


Figure 4.9: Visual performance of BRILS among different seeds for the instance with $n = \{100, 200\}$ and $p = \{5, 10\}$

of hubs and nodes included in the graph. In addition, most of the BKS found in the literature employed the aforementioned data set. So that we can compare our results against those published results.

The results confirm the efficiency of the BRILS algorithm, since our proposed methodology found the BKS for most of the instances. Although for the larger instances the BRILS algorithm did not reach all of the BKS reported using GVNS, the results shown in Table 4.2 can be considered very satisfactory. Besides, BRILS is designed to obtain good quality solutions (in some cases the optimal solution) in *all* variants of this problem (i.e., for different

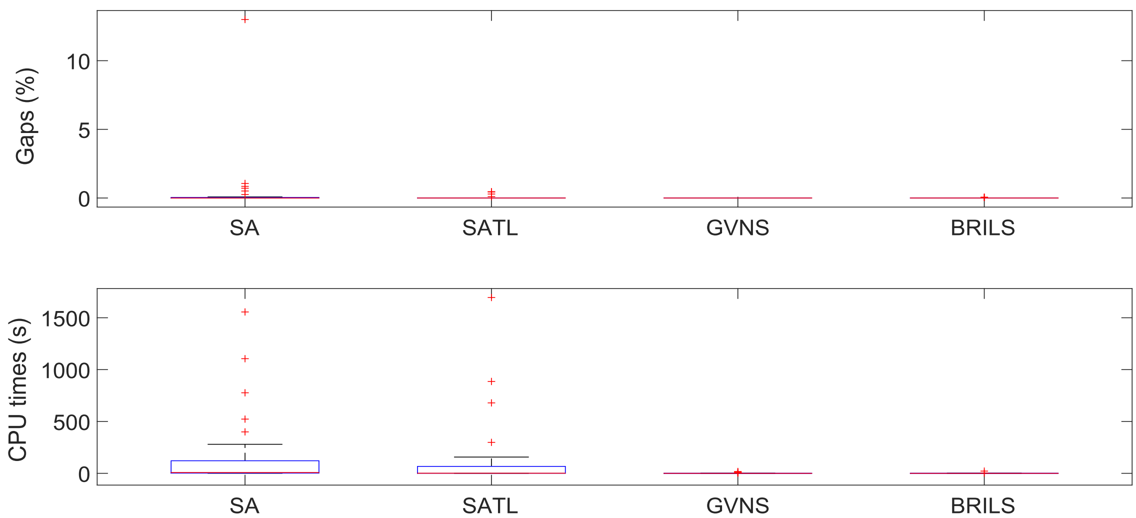


Figure 4.10: Visual comparison among the SA, SATL, GVNS and BRILS heuristics

values of n and p). Moreover, the average computation time is 0.64 seconds which is 36% of the average time consumed provided by the GVNS.

With all the shown instances, we can conclude our methodology outperforms the existing state-of-the-art heuristics in small size and some large size instances in terms of time and solutions. Furthermore, it is worth mentioning that our algorithm is relatively simple: it has been applied to the tested instances without requiring any special fine-tuning or set-up process, which makes BRILS an appealing tool to support the solution of the USApHMP. We expect to apply the presented methodology to larger network instances and to study its application in real system deployments.

Finally, the proposed algorithm can be divided among multiple processors in a natural way with the objective of running it in less time. This is due to the nature of the BRILS framework, which ILS stage can be easily parallelizable. With parallelization the results obtained so far by BRILS can be improved by decreasing the required execution time when having a machine with multiple execution cores.

4.4 Artificial Immune Systems for Solving the USApHMP

4.4.1 Solving Methodology

As mentioned in Chapter 2, AIS are a class of algorithms inspired by the natural immune system. Applied in the context of engineering and computing, the immunological principles of decentralization and maintenance of diversity are often useful in solving problems with large solutions spaces, such as the USApHMP.

Immune system features have seen increased interest from the research community. Features such as: recognition of antigen characteristics, pattern memorization capabilities, self-organizing memory, adaptation ability, immune response shaping, learning from examples, distributed and parallel data processing, multilayer structure and generalization capability (Dudek 2012) have been applied to solve problems with large solutions spaces, such as the USApHMP. AIS are not limited for solving optimization problems, in fact, AIS-based algorithms have been applied for solving identification of non-linear systems and (Silva et al. 2015, Fernandez et al. 2018), and machine learning related problems (Aydin et al. 2011).

In the context of USApHMP, the cells and their antibodies are analogous to a candidate solution for the problem. Thus, it is possible to immunologically simulate such problem, seeking solutions effectively.

With this in mind, two immune-inspired metaheuristics are proposed for solving the USApHMP: (*i*) the CLONALG algorithm, which simulates the rapid response of immune systems when presented with a pathogenic agent, and *(ii)* the optAiNet algorithm inspired by the theory of the immune network, which simulates a self-regulatory network of antibodies where each member has the capacity to affect the whole.

4.4.1.1 CLONALG Algorithm

The clonal selection principle was initially based on works carried in the seventies by Burnet (1978). This work served as inspiration for CLONALG (de Castro and Von Zuben 2002), a popular AIS algorithm involving an abstract version of the cloning and hypermutation process. All clonal selection-based algorithms essentially gravitate around a repeated cycle of match, clone, mutate and replace, and numerous parameters can be tuned, including the cloning rate, the initial number of antibodies, and the mutation rate for the clones.

In particular, CLONALG has the intrinsic ability of balancing the exploitation of the best solutions with the exploration of the search space, which can be very important to increase the probability of finding the global optimum or a good solution. The key steps of CLONALG are selection and mutation process. The selection process is expressed as selecting a number of best affinity elements from the population. During the mutation process, a common mutation rate is applied to the antibodies that are selected.

For solving the USApHMP, the CLONALG algorithm, described in Algorithm 15 receives the following input parameters: (*i*) the number of nodes as well as the number of hubs given by the problem; (*ii*) the clonal factor β; (*iii*) parameter ρ which controls the shape of the mutation rate; (*iv*) the size of the antibody pool $Ninitial$; (*v*) the number of clones nC; and (*vi*) the parameter $range$ for selecting the number of Ab that will replace the lowest affinity Ab. The procedure starts generating a pool of antibodies[1] with fixed size $Ninitial$, in which every Ab_i representing an element from the parameter space. de Castro and Von Zuben (2002) proposed that the generation of those antibodies occurs randomly in order to have a great diversity of population.

In the case of the USApHMP, every Ab represents a solution for the problem. First, the given number of p hubs are randomly chosen following a uniform probability distribution, then, the remaining nodes are allocating to their nearest hubs. Next, every antibody Ab is evaluated by the fitness function $f^{Ag}(Ab_i)$, in which Ag represents the antigens, and the fitness is the cost function given by Eq. 4.1a. This way, the Ab (solutions) with highest affinity will survive during the next step. Following, the amount of clones nC to be generated for each individual is calculated as follows:

$$nC = \text{round}(\beta \cdot Ninitial), \tag{4.4}$$

[1]For simplicity, the terms "cell" / "antibody" will be considered equivalent within AISs.

Algorithm 15: CLONALG(nodes, pHubs, β, ρ, Ninitial, nC, b, range)

```
Ab ← random(Ninitial, range)
while Stop condition is not met do
    Solve fit ← affinity(Ab)
    C ← clone(Ab, nC, β)
    C* ← mutate(C, fit, ρ)
    Fit' ← affinity(C*)
    R ← select(C*, Fit')
    Ab ← replace(R, random(b, range))
end
return Ab
```

where β is the clonal factor. Then, the new set of nC clones performs an affinity maturation process. The parameter ρ controls the shape of the mutation rate with respect to the following equation:

$$\alpha = e^{(-\rho \cdot fit)}, \tag{4.5}$$

where α represents the mutation rate, and fit is the fitness function $f^{Ag}(Ab_i)$ value normalized in $[0,1]$. Note that the mutation rate is inversely proportional to their parent's affinity, i.e., the greater the affinity, the lower the mutation intensity. After that, the affinity of every clone is calculated. Next, a new set of R individuals is formed. In this process, which represents the implementation of the immune memory of the system, the individuals (solutions) with the highest affinities and diversity are kept. Finally, the main loop is concluded with a random generation of b new antibodies that will replace the lowest affinity individuals in the current population. The process repeats itself until a pre-established stopping criteria is met.

As already stated, the CLONALG is highly affected by the parameter mutation. The aim of this operator is to increase the diversity of the candidate solutions and the exploration of the search space. It is based on a slightly modification of a individual's specific characteristic, and it occurs with a very low probability, around 1%. Considering that an antibody will represent an allocation matrix of the hubs and nodes, i.e., the matrix H in Equation 4.1a, Algorithm 16 presents the pseudo-code for this operator. The procedure, selects between $pHubs$ or $node$ mode, by doing so, local regions are explored, helping the algorithm escape out of local minima. Once the set of clones have been generated, each clone is then mutated. This ensures that the current solutions generated after the clone step have, on average, higher affinities than those of the early primary response. Random changes are introduced and cause structurally different solutions. One such event or change will lead to an increment in the affinity of the antibody.

It is worth stressing that the mutation operator explained above follows the restrictions imposed by the problem formulation, seen in Equation 4.1a.

4.4.1.2 optAiNet Algorithm

The immunological network theory (which serves as an inspiration to the AiNet family of algorithms) differs from the clonal selection theory by proposing antibodies capable of identifying and interacting with each other, continually stimulating and suppressing other individuals. The set of antibodies forms a decentralized and autonomous immunological

Algorithm 16: mutate(Ab,pHubs,nodes)

```
initialSolution ← genRandSol(nodes, pHubs)
greatpHub ← randSelect(pHub from initialSolution)
while Stop condition is not met do
    select mode pHubs or nodes
    if pHubs then
        pHub* ← select one pHub randomly
        disconnect all nodes allocated to pHub*
        greatpHub ← nodes
        newpHub ← select one of the nodes allocated to the greatpHub
        connect the non allocated nodes to newpHub
    else
        node ← select one node randomly
        disconnect node from its corresponding pHub
        connect node to a new randomly selected pHub
    end
end
return Ab*
```

network in which each member is able to influence what will happen to the network. Thus, in addition of being able to react to the antigens, the antibodies here also react to themselves (De França et al. 2010).

Antibody-antibody relations depend only on the individual performance and degree of similarity between them, occurring as follows: (*i*) being sufficiently similar, the worst performing antibody is deleted; (*ii*) otherwise, clones are generated with a mutation inversely proportional to their performance, removing those that do not have beneficial mutations, i.e., the theory of clonal selection applies. Finally, in order to maintain the cellular diversity analogous to that seen in the immune system, a percentage amount of random antibodies is added to the set. With this, the algorithm encourages aspects of global space exploration while avoiding premature convergences to a local optimum.

Keeping that in mind, optAiNet is a variation of the immunological network algorithm with a special focus on optimizing functions, where the main characteristics lie on the way in which new antibodies are introduced to the system and in the mechanisms of mutation and evaluation of antibodies. The pseudo-code is presented in Algorithm 17. The input parameters are the following: (*i*) the number of nodes as well as the number of hubs given by the problem; (*ii*) the parameter $elite$ that controls the similarity operator; (*iii*) the clonal factor β; (*iv*) the parameter σ that works as a suppression threshold; (*v*) parameter ρ which controls the shape of the mutation rate; (*vi*) the size of the antibody pool $Ninitial$; (*vii*) the number of clones nC; and (*viii*) the parameter $range$ for selecting the number of Ab that will replace the lowest affinity Ab.

Algorithm 17: optAiNet(nodes, pHubs, elite, β, σ, ρ, Ninitial, nC, b, range)

```
Ab ← random(Ninitial, range)
while Stopping criteria not met do
    Solve fit ← affinity(Ab)
    Ab ← remove(Ab, elite)
    C ← clone(Ab, nC)
    C* ← mutate(C, fit, β)
    Fit' ← affinity(C*)
    R ← select(C*, Fit')
    R* ← remove(R, σ)
    Ab ← replace(R*, random(b, range))
end
return Ab
```

The logic behind the optAiNet algorithm is somehow similar to the one developed in CLONALG, i.e., a pool of size $Ninitial$ of initial solutions Ab is generated and every Ab is evaluated following the affinity function. A remarkable difference with respect to CLONALG is the similarity operator. Before cloning the antibodies with highest affinity, the optAiNet algorithm applies the similarity operator and removes the antibodies with less affinity in a subset of antibodies sufficiently similar, then, a number of clones nC are generated and an affinity process is performed, the clones with highest affinity remain and the similarity operator is applied to them, following, the clones highly similar are removed taking into account the affinity function and finally, the removed individuals are replaced with new antibodies. Another difference between the optAiNet with respect to CLONALG is that the number of antibodies depends on the similarity operator, the algorithm starts with the pool of antibodies of size $Ninitial$ but it changes during execution.

Some important points should be taking into account when implementing the optAiNet algorithm: (*i*) The fitness function, which is the one being optimized, is in fact, a measure of affinity between antibody and antigen; (*ii*) Each solution corresponds to the information contained in a given receptor (network cell); and (*iii*) The affinity between cells is measured by a simple Euclidean distance.

It is worth mentioning that, because USA*p*HMP is a combinatorial problem, it is not appropriate to use a typical approach such as Euclidean distance to measure the degree of similarity between solutions. Thus, the similarity operator must be customized accordingly, operating as follows. For each p hub of a given candidate solution (Ab), the unweighted geometric center between its connections is calculated. Then, with all the centers obtained, we add the distance between the centers closest to both solutions. This way, the operator is expected to return a measure where similar solutions are in fact similar in terms of the problem, i.e., hubs and nodes similarly allocated. Algorithm 18 presents the pseudo-code for this operator.

Algorithm 18: calcDistance(Ab1, Ab2)

```
totalDistance ← 0 ;
for every hub in Ab1 do
│  calculate geometric center between interhub connections ;
end
for every hub in Ab2 do
│  calculate geometric center between interhub connections ;
end
for every geometric center of Ab1 do
│  calculate geometric center of the closest Ab2 ;
│  sum distances between centers to totalDistance ;
end
Update totalDistance ;
return totalDistance;
```

In general, the performance of AIS is mainly affected by the variations of critical parameters. In particular for CLONALG and optAiNet algorithms, the crucial parameters that control the exploration capacity of the algorithms are: (*i*) The antibody population size; (*ii*) The number of clones; (*iii*) The remainder replacement size; and (*iv*) The termination condition. Besides these parameters, optAiNet can be highly affected by two more key parameters, the threshold of performance and threshold of similarity. Thus, the consequences of changing the settings of each parameter are presented.

- Antibody population size, *Ninitial*: Specifies the total amount of individuals. Note that each candidate antibody must satisfy the constraint conditions of the specific problem. If excessive, an overloaded number of redundant antibodies can be generated, which will be eliminated by the threshold of similarity in the case of optAiNet algorithm.

- Number of clones, nC: It is a set of antibodies which are selected from a whole antibody pool that has the highest affinity with the antigen. In the cloning phase, antibodies with higher affinity should be more reproductive than those with lower affinity. Many clones may be redundant, while few clones may be ineffective in the search for a favorable mutation.

- Remainder replacement size, b: Specifies the total number of lowest affinity antibodies to replace with random antibodies on each antigen exposure. As mentioned before, after the mutation operator, some candidate antibodies may become better in affinity while the others may become worse in affinity. Therefore, it is essential to update the antibody spaces and to allow those better/worse antibodies to upgrade/degrade. Excessive addition of antibodies increases the computational time without bringing equivalent benefits; on the other hand, adding few antibodies impairs the search for the optimal overall.

- Termination condition, $maxIt$: Specifies the total number of algorithm iterations to perform. The parameter controls the amount of learning the system will perform on the problem domain. The entire process is completed after a number of iterations is executed.

- Threshold of performance, *elite*: It is the minimum threshold for the relative performance of an antibody before it is eliminated; high values decrease the ability to restart the algorithm, putting possible solutions at risk; low values allow the existence of unfeasible solutions.

- Threshold of similarity, σ: This parameter determines the maximum limit of similarity so that two antibodies can coexist in the solution set without being eliminated; if too small, computational power is lost in solutions around the same local optimum and if excessively large, the local optimum would be ignored.

4.4.2 Experiments and Results

In this section, some numerical experiments are considered to check the performance of the proposed CLONALG and optAiNet algorithms to solve the USApHMP. For the sake of examination, four benchmark problems are selected from the AP data set. The problems generated from the AP data set are of size $n = \{10, 20, 50\}$ and $p = \{2, 3, 5\}$. As reported previously, the details for the AP data set are described in (Ernst and Krishnamoorthy 1998). All computational experiments were performed on an Intel® Xeon E5-2630 v4 at 2.20 GHz running the Ubuntu 14.04 operating system. The procedures in our method have been implemented as Python®applications.

The crucial parameters mentioned in the last section were defined with the aid of a fine-tuning process, which comprised 10 trials of both algorithms for each possible configuration of $Ninitial \in \{10,20,30,40\}$, $b \in \{0.1,0.2,0.3,0.4\}$, $nC \in \{2,4,6,8\}$, $maxIt \in \{100,150,200,250\}$, $elite \in \{0.01,0.03,0.05,0.07\}$ and $\sigma \in \{1,3,5,7\}$. The fine-tuning tests were executed in the scenario where $n = 10$ and $p = 5$. The average cost of the solution was calculated for each instance, and the parameters which led to the best results are displayed in Table 4.4. After set the values for each parameter, we have run each algorithm 10 times as in the fine-tuning process due to computational time consumed by each algorithm execution. That way we can collect the information regarding the performance of both algorithms for solving the USA*p*HMP using the AP data set. The quality of each solution is evaluated as a percentage *gap* between the BKS and the solution found by the algorithm in hand using Eq. 4.3.2. In addition, run time information has been used to compare the efficiency of the methods presented above.

Table 4.4: CLONALG and optAiNet parameters

Parameter	**Value**
Ninitial	20
nC	4
b	0.1
maxIt	200
elite	0.05
σ	5

With this in mind, the analysis starts with Figure 4.11, which depicts the cost curve for the best solution in a test simulation with an instance of the problem of size $n = 50$, $p = 5$ and 100 iterations until met the stop condition. Notice that, as expected, there are *steps* in the cost curve. This phenomenon occurs when the search is able to locate a new optimum point, either by mutation or by restarting. Also, it can be seen that the width of the *steps* increases with the current execution time, meaning a longer search time between finding new ones and a great difficult level for finding them. It is a consequence of the exploration of search space process by the populations of the two methods.

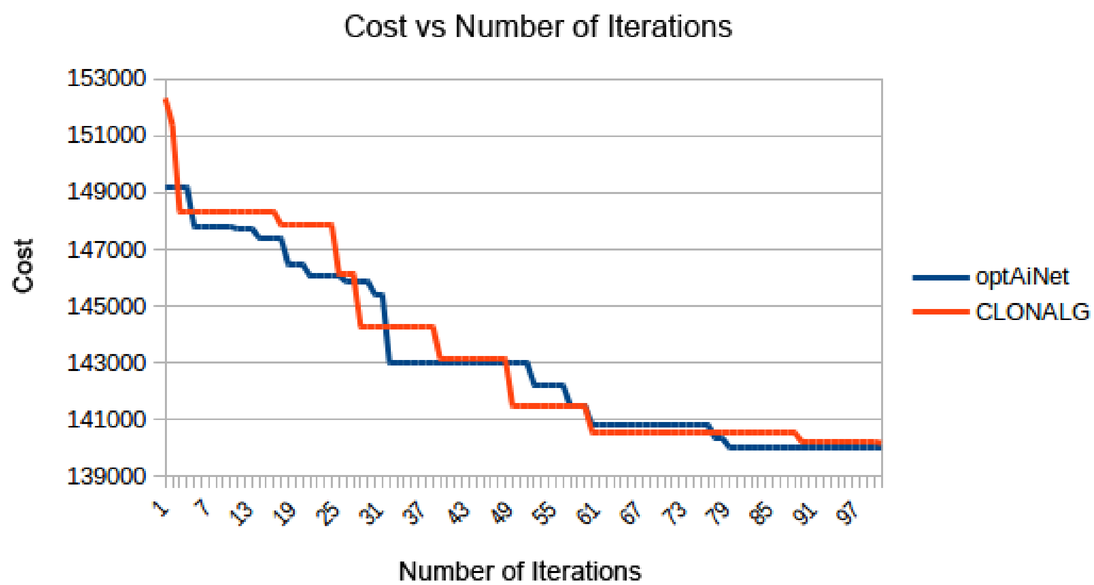


Figure 4.11: Cost of the best solution compared along the iterations by the CLONALG and optAiNet algorithms

Following, results obtained for the set of instances are summarized in Table 4.5. The first column presents the algorithm tested. The following two columns, "n" and "p" indicates the number of nodes and hubs, respectively. The last three columns present the percentage gaps regarding the BKSs. Particularly, the *average gap* column shows the average of the gaps found for each instance using the optAiNet and CLONALG algorithms. The *best gap* presents the gap between the best solution found by the proposed algorithms with respect to the BKS. The *max gap* indicates the gap between the worst solution found by the algorithm and the BKS and *Time* shows the average running time needed to obtain the corresponding solution among the 10 executions.

Considering the effectiveness of the technique, the optimization quality of optAiNet is better with respect to the CLONALG, in spite of the higher computational time presented by optAiNet. Furthermore, it is clear that CLONALG, although less effective, has significantly lower execution times. Failing to embrace more solutions space to optimize the problem, CLONALG has encountered adversity to get out of local optima. Such difficulties were

Table 4.5: Solutions found by optAiNet and CLONALG for the AP instances

	n	**p**	**Average gap** (%)	**Best gap** (%)	**Max gap** (%)	**CPU time** (s)
optAiNet	10	2	0	0	0	4.20
optAiNet	20	3	0	0	0	9.90
optAiNet	50	3	0	0	0	61.47
optAiNet	50	5	1.05	0.22	3.51	210.97
CLONALG	10	2	0	0	0	1.16
CLONALG	20	3	0.03	0	0.34	3.48
CLONALG	50	3	0.68	0	1.71	18.13
CLONALG	50	5	3.92	2.62	5.99	18.44

expected (de Morais et al. 2011), since the algorithm does not have population control capacity like optAiNet and depends exclusively on the cloning/selection/mutation processes to escape from search regions already exploited, but in compensation, the amount of computational time consumed for the execution of the technique was superior when compared to optAiNet.

Furthermore, a detailed comparison of the proposed AIS metaheuristics with the results obtained when applying BRILS for solving the USApHMP is presented in Table 4.6. The best CLONALG and optAiNet results on the AP data set were compared with results obtained by the BRILS algorithm introduced in Section 4.3. As it can be seen from Table 4.6, BRILS method obtained optimal solutions for all instances while AIS metaheuristics methods obtain optimal solutions for the first three instances. Regarding the running times, the CLONALG and optAiNet are several times slower in comparison with BRILS method. Those results simply manifest the high performance of the proposed BRILS method on AP instances. Although AIS did not reach optimal solutions for all instances, both methods are a valuable addition to the repertoire of algorithms for solving the USApHMP, and the empirical results point out the need for a future investigation on the application of such methodologies in order to seek further improvements of the algorithms.

Table 4.6: Solutions found by optAiNet and CLONALG compared to BRILS for the AP instances

Instance		**CLONALG**		**optAiNet**		**BRILS**	
n	p	*Gap* (%)	*Time* (s)	*Gap* (%)	*Time* (s)	*Gap* (%)	*Time* (s)
10	2	0.00	1.16	0.00	4.20	0.00	0.00
20	3	0.00	3.48	0.00	9.90	0.00	0.00
50	3	0.00	18.13	0.00	61.47	0.00	0.00
	5	2.62	18.44	0.22	210.97	0.00	0.00
Average			10.30		71.64		0.00

4.4.3 Discussion

As a result of the experiments carried out in the last section, some questions can be derived. The performances illustrated in Figure 4.11 can be considered typical due to the nature of the algorithm. In both cases, the number of iterations performed until the algorithms were capable of locating the best solution is also depicted. Note that the CLONALG requires a higher number of iterations than optAiNet. Nevertheless, the CLONALG is still capable of locating feasible solutions and in some cases the optimal solutions.

Furthermore, it is important to evaluate the average performance of each algorithm. Table 4.5 provides the average value obtained over 10 runs of the algorithm for each of the generated instances from the AP data set. With greater capacity of search and exploration of the diversity of solutions, optAiNet achieved superior results, although, higher computational times were observed with respect to CLONALG. In this regard, CLONALG shows itself as superior since the average time spent by optAiNet for solving the four instances was 6 times greater than the one spent by CLONALG for solving the same instances.

Finally, one can note that CLONALG it is not too sensitive when the number of p hubs is increased from 3 to 5. In the case involving 50 nodes the situation changes, since the algorithm has a limit parameter for the size of the population. That particular situation can be bypassed by adjusting such parameter and does not characterize a theoretical limitation of the algorithm. Additionally, there is evidence to suggest that both, CLONALG and optAiNet algorithms are stochastic in nature. Each time they are run a different result is obtained.

In sum, by looking at the results presented above, we can conclude that AIS can operate efficiently for solving the presented problem as long as the proper parameters are set. In other words, the performance of the algorithm is essentially limited by tuning of parameters process. The last, is a disadvantage when comparing to BRILS method, which does not need time-costly fine-tuning process. As it is implied throughout this section, the work developed is germinal and thus, it provides several perspectives for future studies, such as the details of the experiments for scenarios with more nodes and hubs. Additionally, the algorithm sensitive to its tuning parameters must be assessed. Second, the present findings might have important implications for solving variations of the USApHMP.

4.5 Concluding Remarks

In this chapter we have proposed three different methodologies for solving the USApHMP: (*i*) a Biased Randomized Iterated Local Search algorithm denominated as BRILS, which combines Biased Randomization techniques with an Iterated Local Search approach, and (*ii*) two Artificial Immune Systems CLONALG and optAiNet. The most popular benchmark problem found in the literature, the AP data set was considered. A series of tests were carried out, which highlighted in particular the performance of the BRILS algorithm in the 28 instances that were evaluated.

On the one hand, the particular characteristic of the BRILS approach is its robustness in terms of solution quality and time consumed. Additionally, it is worth mentioning that it is easy to implement and it does not require any complex or extensive parameter calibration in order to reach good quality solutions. On the other hand, after comparing the proposed method with the best known solutions and other state-of-the-art methods, the results show that the use of the BRILS algorithm allows to obtain significant quality solutions within a reasonably short computational effort across the USApHMP.

Regarding the AIS algorithms, despite the immune inspiration of both algorithms, optAiNet and CLONALG are very similar to evolutionary algorithms, as the main steps of Clonal Selections (cloning, hypermutation and selection) correspond to a micro-evolution that occurs in the immune system and, so, they are modeled in optAiNet and CLONALG with operators similar to those already present in evolutionary computing. The Clonal Selection theory for acquired immunity has been shown to be a useful metaphor for solutions for complex engineering problems such as pattern recognition, identification of non-linear systems and classification. The methodology presented in this chapter has demonstrated that the powerful Darwinian-like metaphor can be used to solve design network related problems. Regarding this matter, it was demonstrated that the USApHMP can be approached efficiently via optAiNet and CLONALG given its NP-hard nature. Standing out for being able to find feasible solutions, the optAiNet algorithm was able to optimally solve 3 of the 4 analyzed instances of the AP data set. In turn, the CLONALG presented a singular capacity to converge with speed, however finding worse quality solutions with respect to optAiNet for the same instances. In general, both techniques proved to be viable in solving the smaller instances of the AP data set, and the perspective of deepening the experimental analyzes for the larger instances of the problem is open.

In summary, the findings presented in this chapter add to a growing body of literature on the USA*p*HMP. Comprehensive results showed that the ILS metaheuristic may significantly benefit from the integration of BR into its framework. Furthermore, AIS metaheuristics employed, reveal themselves as potential techniques for solving the USA*p*HMP. Regarding the AIS implementations, the proposed algorithms clearly have some limitations when comparing to the BRILS algorithm since requires the cross-validation of several parameters. The parameter fine-tuning of a metaheuristic may be a time-consuming and complex, but may have a high effect on the quality of the solutions found. In the next chapter the same problem is considered under uncertainty conditions and a simheuristic methodology is proposed.

Chapter 5

Stochastic Version of the Uncapacitated Single-Allocation p-Hub Median Problem

As already introduced in previous chapters, hub location related problems are of great importance due to the necessity of design low cost and high efficiency networks in multiple applications such as telecommunication and computer networks (Resende and Pardalos 2008). In recent years, a large increase in the traffic flow have been experienced. The massive use of video on demand to the countless number of mobile devices exchanging text and video, the huge impact of companies such as Amazon with air transportation and trucking and the increased number of network technologies for vehicular transportation, have influenced the fact that traffic has reached a scale that can exceed the capacity of today's networks. In order to find solutions to those realistic scenarios, it is crucial to consider the stochastic behavior of the main parameters. Thus, companies around the world that make use of design network techniques have been forced to increase the capacity of their networks to serve this growing demands as well of the inclusion of stochasticity on the key parameters which foresees the behavior of the users' demands. As the cost of installing a network infrastructure is usually very large, network design heavily use optimization tools to keep costs as low as possible. This process is usually conducted using hub localization problems.

The design of hub networks has its advantages, because depending on the context of the application can reduce infrastructure costs and enable economies of scale through the consolidation of traffic flows with the use of technologies for the transportation or transmission of large volumes.

Hub-and-spoke networks represented by hub location problems are commonly studied in location theory. The characteristic of these networks rely on the use of hub facilities. Rather than connecting each demand point by a direct link, a hub network replace those direct connections by setting a group of hubs (Kara and Tansel 2001). The hubs allow that the traffic flow among many points being concentrated and conducted through a small number of links achieving scale economies. Moreover, depending on the applications, the use of hub-and-spoke networks can reduce infrastructure costs.

The most common HLP is the p-hub median problem. As already introduced in Chapter 4, in such problems, the objective is to minimize the total cost of movement flow. Furthermore, these problems may be classified by the way in which demand points are assigned or allocated to hubs (single and multiple schemes) and also by the objective cost functions. An example of these type of problems is the USApHMP.

It is common to find in the literature studies considering the deterministic version of HLPs, leaving behind the study considering the uncertainty present on real-world problems. In this paper, in order to overcome this gap, a formulation for the USApHMP under uncertainty assumptions is proposed, where a fixed number of hubs have unlimited capacity and each non-hub node is allocated to a single hub. In the USApHMP every node is a potential hub and the number of hubs is known *a priori*. Instead of considering that the demands are deterministic values, the demand is modeled as a random variable.

Additionally, in order to make more realistic the stochastic scenario, a QoS thresholds related to all nodes (potential hubs) of the network is included. Thus, every node has a transmission capacity. Consequently, given a threshold, if a node is selected as hub, the total amount of flow passing through such hub because the flow every origin node i sends and receive to/from the network is calculated, if this total flow surpass the given threshold, the flow may be directed through the hub that supports the transmission, otherwise, a penalty cost is added. Figure 5.1, illustrate this scenario, the squares represent the hubs, the circles represent the origin and destination points (a same point has the two roles) and the lines between hubs represent the links which have scale economics. Note, that depending of the given thresholds, a node i can be allocated to a hub that support the total amount of flow. That way, the network may no longer be limited by the amount of flow every node wants to transmit. A realistic scenario application could be that of video delivery. Up to a given amount of video, each hub can work in a linear way without any penalty. When the number of videos (i.e., the demands) exceed the hub transmission capacity, their quality will be reduced. This is model through the use of a penalty that corresponds to manage all the demanded videos while with a lower quality.

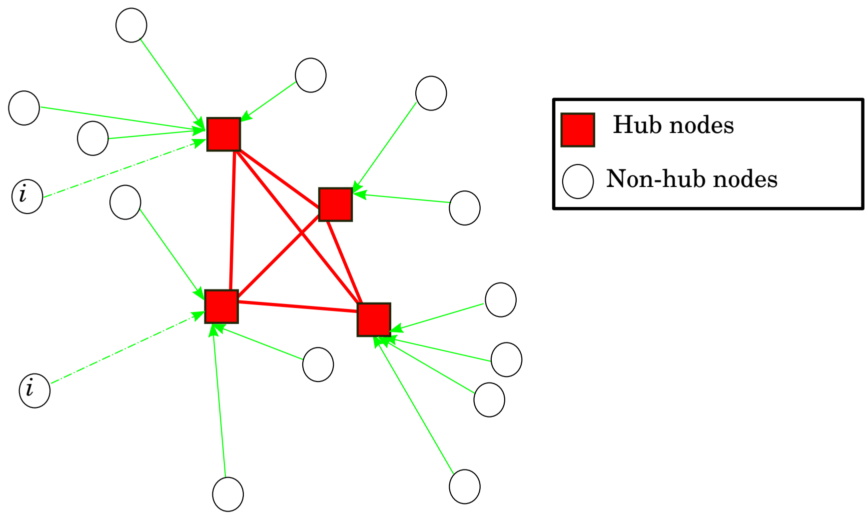


Figure 5.1: Illustrative example of hub network with single assignment and QoS thresholds. In such example, node i can be allocated to different hubs depending on the ammount of stochastic flow that hubs support

With this in mind, a fast, easy to implement simheuristic algorithm (Juan et al. 2015*b*) to solve the stochastic version of the aforementioned problem is developed, based on the integration of simulation techniques and a two-stage metaheuristic methodology. Simherusitics belong to the optimization-simulation techniques, and have been applied to a considerable range of optimization problems that consider uncertainty on the inputs parameters. Accordingly, the main contributions of this chapter are (i) extending the formulation of the USApHMP to a stochastic model; (ii) coming up with a metaheuristic framework which implement the simulation and (iii) discussing under a set of computational experiments the promising results of our proposed simheuristic algorithm to the USApHMP.

5.1 Literature Review

This section presents a review of different variants of the stochastic p-hub median problem. As mentioned before, hub location related problems are receiving increased attention in telecommunications, transportation and logistics and the values the inputs can take in such problems are usually not deterministic. For example, a common scenario in telecommunications is the user's request of data packets (voice, image, video, etc.), thus, the service costs and even the requests are constantly changing which can be represented by random variables instead of constant values. This scenario is also feasible for logistics and transportation, where vehicles have to be dispatched in response to real-time requests.

There are few published papers addressing the uncertainty issue in the context of hub location related problems, with respect to the number of publications considering the same problems and its deterministic versions. In this context, Marianov and Serra (2003) modeled hub location problem as an M/D/c queuing network for airline transportation. In this approach, the authors established a probabilistic capacity constraint in order to limit the number of airplanes waiting on queue. They formulated a mixed integer programming and proposed a heuristic based on TS tested on different networks up to 30 nodes. Sim et al. (2009) focused on the stochastic p-hub center problem in which the authors incorporated stochastic travel times to model the minimum service level required. The authors proposed a linear mixed integer programming formulation for the problem assuming that travel times are independent variables with normal distribution. They proposed three different heuristics; a radio-based heuristic, a randomized local search heuristic and a combination of both to obtain feasible solution for test solving the CAB instances.

Later, Yang (2009) analyzed the air freight hub location problem and flight routes planning under seasonal demand variations. Author presented a mixed integer programming formulation for the problem considering the demand as a discrete random variable, also, the model permitted direct connection between non-hub nodes. The solver GAMS was employed to solve one case study from the air freight market in Taiwan and China up to 10 nodes (airports). In the same way, Contreras et al. (2011) studied stochastic uncapacitated multiple allocation HLPs in which uncertainty is present in the transportation costs and demands. It is shown that with uncertain demands or dependent transportation costs, stochastic problems are equivalent to their corresponding deterministic expected value problem, in which random variables are replaced by their expectations. In the case of uncertain independent transportation costs, they suggested an algorithm based on Monte Carlo simulation and Benders decomposition to solve instances up to 50 nodes.

Alternatively, Alumur et al. (2012) addressed single and multiple HLP under uncertainty. The authors presented a modeling framework in which the uncertainty associated with set-up costs and the demands is represented by a finite set of scenarios. They tested the model on the CAB instances by using CPLEX. Mohammadi et al. (2013) proposed a stochastic multi-objective model considering risk for solving the p-hub covering problem. The authors assumed that every path of origin-destination nodes had a risk factor and presented an imperialist competitive algorithm inspired by imperialistic competition. The results were obtaining by using their own data set including a risk factor and number of nodes up to 100.

Hult et al. (2014) considered the single allocation p-hub center problem taking into consideration the stochastic nature of travel times. The authors proposed an exact computation approach to solve it. It was tested on the standard AP and CAB data sets and assuming the distance as normally distributed. Furthermore, Sadeghi et al. (2015) proposed a chance-constraint approach for a p-hub covering location problem. They considered stochastic degradation on the link capacities by following a truncated Erlang distribution function. They proposed a mixed integer linear programming formulation, and to solve it, a metaheuristic algorithm tested on generated instances up to 200 nodes was developed.

Recently, Adibi and Razmi (2015) proposed a two-stage stochastic programming approach for solving the uncapacitated multiple allocation hub problem. The authors proposed a formulation assuming that demands and transportation costs were independent. The proposed formulations and methodology were used to design a 10-nodes network of air passengers in Iran. Zhai et al. (2016) considered as well a two-stage uncapacitated HLP in which uncertain demands follows Normal distributions. The authors proposed a hybrid metaheuristic which combined GA, VNS and fuzzy simulation. The results were presented using a data set they generated with instances from 20 to 50 nodes and 5 to 10 hubs.

Rahimi et al. (2016) presented on their work a multi-objective HLP modeling the congestion as stochastic. The authors developed a hybrid metaheuristic as a result of the integration of SA with differential EA methods. The proposed framework was tested using generated data sets with problems containing up to 200 nodes.

Also, Meraklı and Yaman (2017) proposed a special polyhedral uncertainty model known as the hose model, and two different Benders decomposition algorithms for solving the capacitated hub location with uncertain demand. Authors tested the algorithms on the AP and CAB data sets using the instances with 25, 40 and 50 nodes, and generated the traffic bounds for the hose model.

Recently, Correia et al. (2018) investigated the capacitated multiple HLP assuming stochasticity on the flow to be sent from the origin node to the destination node. The authors proposed a two-stage stochastic modeling framework. The computational tests were performed using the CAB data set which allow the authors to generate instances with 15, 20 and 25 nodes. Regarding the stochastic flows, the authors considered 5 scenarios representing in this way an evolution trend for the flows and assuming that each scenario occurred with probability of 1/5. Likewise, Azizi et al. (2018) considered the problem of designing hub-and-spoke networks with stochastic demands and congestion. The authors proposed an exact method and a GA. The computational experiments were performed using the CAB and Turkish data sets. The congestion on the network was represented using the number of users as number of hubs.

In particular, regarding the USApHMP, the quantity of works found in the literature addressing uncertainty is very limited. Ahmadi et al. (2015) formulated the USApHMP as a two-stage stochastic programming framework in which uncertainty is associated with flows between the nodes. The authors integrated a basic financial risk measure into the model framework to make robust decisions, and a practical case study of a distribution system of automobile parts in Iran up to 10 nodes was tested using GAMS. Although the problem was stated as USApHMP, authors considered set up costs of opening hub facilities at potential nodes which is not a faithful model of the problem.

Likewise, Ghaderi and Rahmaniani (2016) addressed the USApHMP under uncertainty in which demands and travel times are assumed to be stochastic. The authors proposed a hybrid heuristics based on VNS, PSO and TS to properly resolve the uncertainty in the parameters. They tested the proposed approach on extra scenarios generated by multiplying random numbers which followed uniform function distribution to the AP and CAB data sets.

Recently, Amin-Naseri et al. (2016) came up with a robust bi-objective model for the USApHMP. The authors considered uncertainty on the travel times. They oriented the problem to the telecommunication area and assumed that uncertainty was caused by noise factors. In order to solve the problem, the authors proposed a hybrid heuristic based on the integration of Scatter Search and VNS. The proposed methodology was tested using the CAB and AP data sets.

Later, Qin and Gao (2017) considered the USApHMP with stochastic flows and took into account fixed costs. The authors designed a GA and tested it on their own instances consisting on a network with 10 nodes and the amount of flow between origin node and destination node were described by uncertain variables.

To provide an overview of the literature review related to HLPs when taking into consideration uncertainty on some components, the paramount feature of each studies mentioned above are encapsulated in Table 5.1. Although uncertainty has been addressed in the context of the USApHMP, the aforementioned works do not give an overview of how a multiple-hub network can be affected by the stochasticity represented in the users' demands, since they

Table 5.1: Stochastic p-hub median literature

Article	**Classification**	**Characteristics**
Marianov and Serra (2003)	Hub location Problem	Probabilistic capacity
Sim et al. (2009)	p-hub center problem	Stochastic travel times
Yang (2009)	Hub location problem	Stochastic demands
Contreras et al. (2011)	Uncapacitated multiple hub location problem	Stochastic costs and demands
Alumur et al. (2012)	Single and multiple hub location problem	Stochastic set-up costs
Mohammadi et al. (2013)	p-hub covering problem	risk factor on origin-destination path
Hult et al. (2014)	Single allocation p-hub center problem	Stochastic travel times
Sadeghi et al. (2015)	p-hub covering problem	Stochastic link capacities
Ahmadi et al. (2015)	USApHMP	Stochastic demands
Adibi and Razmi (2015)	Uncapacitated multiple allocation hub location problem	Stochastic costs
Zhai et al. (2016)	Uncapacitated hub location problem	Stochastic demands
Rahimi et al. (2016)	Hub location problem	Stochastic congestion
Ghaderi and Rahmaniani (2016)	USApHMP	Stochastic demands and travel times
Amin-Naseri et al. (2016)	USApHMP	Stochastic demands
Meraklı and Yaman (2017)	Uncapacitated p-hub problem	Stochastic demands
Qin and Gao (2017)	USApHMP	Stochastic demands
Correia et al. (2018)	Capacitated multiple hub location problem	Stochastic demands
Azizi et al. (2018)	Hub location problem	Stochastic demands

only take into account pre-established scenarios where uncertainty is represented by a finite set of cases which does not represent real-world situations. Moreover, such assumptions can find good solutions for some scenarios but poor for others. Furthermore, some authors make effective use of robust optimization and/or stochastic programming. In robust optimization, the aim is to find near-optimal solutions for any set of scenarios no matter which of them are realized, leading us to consider feeble solutions from a scenario that may be unlikely to occur. On the other hand, stochastic programming mostly seek to find close or almost optimal solutions across all scenarios.

With this in mind, this study is motivated by the fact that there exists a lower number of works dealing with uncertainty in HLPs. To partially close this gap, a simheuristic algorithm in order to deal with the USApHMP under uncertainty assumptions is proposed. This is the first time that BR methodology in combination with Monte Carlo techniques are implemented into a procedure for solving the problem. In this regard, service quality thresholds are introduced, by doing so, the influence of variations in the flow and how this impact the decision making process is considered, which was neglected in the previous studies.

5.2 Formal Problem Description

As mentioned before in Chapter 4, (Ernst and Krishnamoorthy 1998) proposed a mixed integer linear programming formulation for the USApHMP, which requires fewer constraints and variables. They also published the popular AP dataset, which is based on the postal delivery in Sydney. Their formulation is still the best formulation in terms of computation time requirements. Hence in this chapter, we use the same formulation and propose an extension taking into consideration data uncertainty. For the sake of clarity, the variables regarding the formulation are given in Table 5.2.

Table 5.2: Set of indexes and input variables

Symbol	Description
N	Set of potential nodes; $i,j,k,l \in \{1,2,\cdots,\lvert N\rvert\}$
C_{ij}	The transportation cost per unit flow between nodes i and j
W_{ij}	Stochastic amount of flow between nodes i and j
p	Number of hubs to be located
$\mathscr{X}$	Cost multiplier in the collection
τ	Cost multiplier in the transfer
δ	Cost multiplier in the distribution

And the involved decision variables are presented as follows:

$$\sum_{k=1}^{n} H_{kk} = p, \tag{5.1a}$$

$$\sum_{k=1}^{n} H_{ik} = 1, \quad \text{for every } i = 1,\cdots,n, \tag{5.1b}$$

$$H_{ik} \leq H_{kk} \quad \text{for every } i,k = 1,\cdots,n, \tag{5.1c}$$

$$H_{ik} \in \{0,1\} \quad \text{for every } i,k = 1,\cdots,n, \tag{5.1d}$$

As introduced before in Chapter 4, constraint 5.1a ensures that the number of given hubs p are chosen, constraint 5.1b establishes that a node is allocated to only one hub. By constraint 5.1c we prevent non-hub nodes being allocated to other non-hub nodes. Constraint 5.1d enforces that the flow is sent only via the selected hubs, thus, preventing direct transmission between non-hub nodes.

The distance from the origin node i to its destination node j is C_{ij}, and the cost of the route $i \to k \to l \to j$ is decomposed by three segments, $\mathscr{X} C_{ik}$ which is the transport cost from origin node i to its hub k, τC_{kl} which represents the transport cost from hub k to hub l and δC_{lj} which is the transport cost from hub l to the destination node j. Generally, τ is used as a discount factor between hubs k and l which aims to represent economy of scale between hubs. If only one hub is used, this means that $k = l$ and no discount factor is applied. The two remaining factors, $\mathscr{X}$ which represents the unit rates (costs) for collection (origin-hub) and δ which represent the costs for distribution (hub-destination) along the path were introduced by Ernst and Krishnamoorthy (1998) to represent the reality on postal service costs, and can also represent the reality in telecommunication applications, since the amount of flow a node i sends to node j does not mean that is the same amount node i will receive.

In order to introduce stochasticity to the aforementioned problem, now it is assumed that the flow sent from one node i to node j follows a known probability distribution represented by a random variable W_{ij} instead of assuming a deterministic demand parameter w_{ij}. Thus, the flow W_{ij} sent through the path $i \to k \to l \to j$ is considered as a distributed random variable with media $E[W_{ij}] = w_{ij}$. Notice that, in the basic stochastic USA*p*HMP with a linear objective function, the solution minimizing the total expected cost will be the same as the optimal solution for the deterministic USA*p*HMP, this property will not hold if, for instance, a nonlinear penalty cost is added in the stochastic USA*p*HMP objective function to account for the flow passing through hubs knowing the flow from an origin node to an end node, is higher than a threshold. Such assumption also represents a real world application, such as the design of a hub-and-spoke communication network during social events. In this type of events, we cannot assure the behavior of the assistants, so, sending flow from one origin node to a destination node through the hub network may vary, in some cases, exceeding a given transmission capacity. To prevent some overloaded situations, companies penalizes users that exceed the transmission capacity.

With this in mind, a vector of QoS thresholds is also generated, so that every potential hub has a given threshold regarding the quantity of flow is collected and transmitted, and every time this threshold is surpassed, a penalty cost is added. Notice that, this does not mean capacity in terms of allocation, indeed, a hub can still allocate unlimited number of nodes, however, the penalization costs related to every time the total amount of flow going trough a potential hub is surpassed will impact the total transportation cost.

As mentioned before, traffic volume and unforeseen factors have a great impact on the transmission. Thus, the transmission of flow from origin node i to destination node j is affected by service quality thresholds. Based on these assumptions and notations, Eq. 4.1b can be reformulated as follows:

$$\text{minimize} \quad \sum_{i,j,k,l \in N} W_{ij}(\mathcal{X} C_{ik} H_{ik} + \tau C_{kl} H_{ik} H_{lj} + \delta C_{lj} H_{lj}) + \phi \tag{5.2}$$

$$\phi = \begin{cases} 0 & \text{if } F_k \le F_k^0 \quad (5.3a) \\ (F_k - F_k^0)^2 & \text{otherwise} \quad (5.3b) \end{cases}$$

In the above formulation, the second part of the objective function given by Eq. 5.2 is regarding the penalty cost ϕ, where F_k represents the total amount of stochastic flow passing through hub k due to the flow every node i allocated to k sends and receives from the rest of the network and is given by:

$$F_k = \sum_{k \in N} (W_{ij} Hik + W_{k'j} H_{k'j}) \tag{5.4}$$

The penalty cost can assume two values: 0 if $F_k \leq F_k^0$, where F_k^0 is the given threshold at hub k and $(F_k - F_k^0)^2$, otherwise, that is, if the total amount of flow F_k is greater than a given threshold F_k^0. Accordingly, value ϕ given by Eq. 5.3 represents the penalty cost per unit flow between nodes i and j at hubs that surpass the QoS threshold. Note that, when $\phi = 0$, that means that the variation on the stochastic flow is too small, then, the problem becomes equivalent to a deterministic USApHMP.

5.3 Solving Methodology

In this chapter, the fast algorithm based on BR techniques and ILS is extended to a simheuristic algorithm to jointly solve the USApHMP under uncertainty assumptions. Due to the successful results presented by BRILS - already introduced in Chapter 4 - for solving the deterministic version of the USApHMP, this work aims to follow the same methodology. Now, instead of considering a deterministic demand w_{ij}, this parameter is modeled as a random variable W_{ij}. Our approach is motivated by the fact that, frequently most of HLPs are assumed that all inputs are deterministic, however, reality imposes that many input variables are of stochastic nature, which requires the implementation of heuristics to deal with such randomness. In this context, simheuristics integrates simulation techniques into metaheuristic driven frameworks to tackle COPs under uncertainty.

In general, the concept of simheuristic is based on the integration of simulation in any of its forms into metaheuristic optimization methodologies to account for stochasticity in different COPs. The general framework of simheuristics, already introduced in Chapter 2 can be summarized as given in the pseudocode presented in Algorithm 19. First, a deterministic counterpart of the instance is obtained. Then, an iterative process is started, where a metaheuristic-driven algorithm to perform an efficient search inside the solution space associated with the deterministic COP is run. Once a stopping criteria is met, estimation of the quality or feasibility of each of the promising solutions when considered as solutions of the stochastic COP, are computed using simulation techniques.

Algorithm 19: generalSimheuristicProcedure(stochastic COP, parameters metaheuristic, objF, statistics)

```
Transform stochastic COP into deterministic counterpart
baseSol ← GenerateSolution
baseSol ← ConstructiveHeuristic(baseSol)
(baseSol, objF(baseSol), statistics) ← simulation(baseSol, fast)
eliteSolutions ← {}
while (metaheuristic stopping criterion not reached) do
    newSol ← GenerateSolution
    newSol ← SearchProcedure(newSol)
    (newSol, objF(newSol), statistics) ← simulation(newSol, fast)
    if (objF(newSol) < objF(baseSol)) then
        eliteSolutions ← add(newSol)
        baseSol ← newSol
    end
end
foreach (solution in eliteSolutions) do
    (solution, objF(solution), statistics) ← simulation(solution, long)
end
return set of stochastic solutions
```

The flowchart illustrated in Figure 5.2 presents how the proposed simheuristic methodology works for the USApHMP. First, a USApHMP instance defined by a set of n nodes and p hubs is considered. Each node i has associated a flow $W_{ij}(1 \leq i \leq n))$ that follows a known probability distribution with an existing mean $E[W_{ij}] = w_{ij}$ where $E[W_{ij}]$ represents the mean or expected value of each random flow, in fact, one of the potential benefits of the present approach is that the methodology is valid for any statistical distribution with a known mean, either theoretical or experimental and also with a given variance (Juan et al. 2015*a*). In particular, if the variances associated with node flows are not too large, it seems natural to expect a deterministic behavior.

In order to make more realistic the stochastic scenario, we have included a penalty cost every time a given QoS threshold at hubs is surpassed. The inclusion of this penalty cost also breaks the lineal property of the objective function. So if the transportation cost of sending flow from the origin node i to the end node j through their respective hubs is met given a predefined threshold, the transportation cost remains and the allocation of the non-hubs to the chosen hub is done, otherwise this allocation will be penalized by adding a penalty cost ϕ

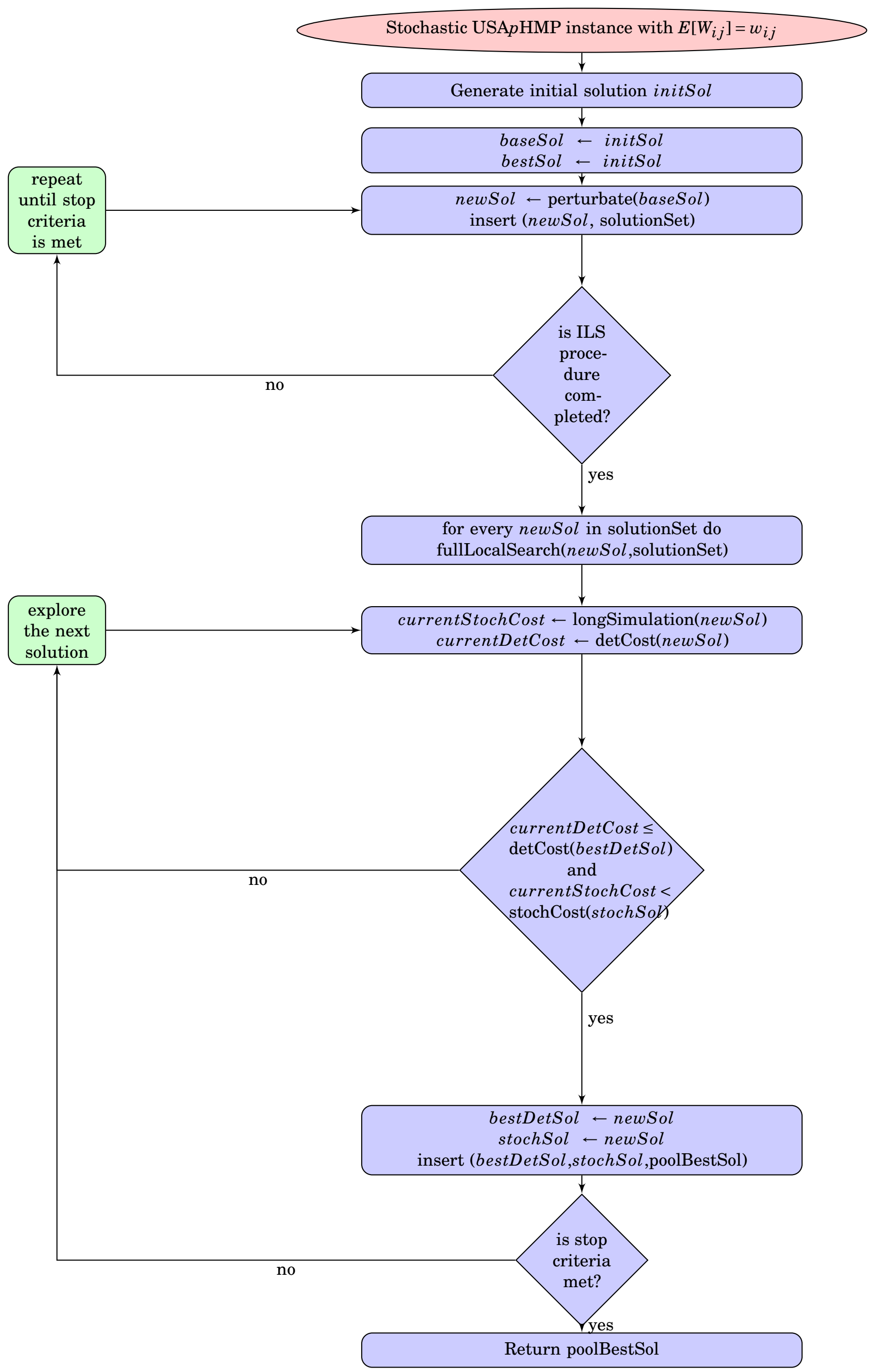


Figure 5.2: Flow-chart diagram of the stochastic USApHMP simheuristic algorithm

to the transportation cost in hand. Following that statement, for the stochastic scenario, in the case where the flow exceeds the threshold F_k, the penalty ϕ is attributed as given in Eq. 5.2.

In the following, the USApHMP with deterministic flow is considered and solved by applying any efficient methodology. Note that the solution for the deterministic USApHMP is an aprioristic solution for the stochastic version of the same problem. In our case, we apply the aforementioned BRILS algorithm in order to generate an initial solution. In the first part of the BRILS algorithm, we use BR to select the p hubs based on an assignment cost criterion. In the second phase, we perturb the selected p hubs of the promising solutions and refine the non-hub nodes allocations obtained so far allowing the generation of feasible solutions. Once an initial solution is generated the ILS loop starts. Inside it, the current base solution is disturbed in order to improve the non-hub nodes allocations obtained so far, thus generating a new candidate solution. During this stage, a percentage *per* of the current base solution is destroyed and then reconstructed to generate a new solution. Then, in order to reconstruct the solution, the same logic used to generate the initial solution is applied, i.e., the non-hub nodes are sorted in a list depending of the connection cost, the sequence of elements on the list is randomized, and the array of nodes allocated to every hub is found. The use of BR techniques in this phase results appealing because it enhances the performance of the heuristic.

After considering that a new solution is better than the current base solution with $deltaDet$ indicating the difference, the current base solution is updated an a $credit$ is fixed and any time a *newSol* is found, this is saved in a pool of best solutions in order to analyze them in the future. Finally, larger simulations are executed over the reduced set of best found solutions. At this stage Monte Carlo simulations over every current solution contained in the *solutionSet* are performed. After run the simulations, every solution would be a candidate solution for the stochastic problem. Thus, sample observations for the deterministic as well for the stochastic problem are obtained and, the best solution of each scenario will be set. Notice, that the methodology presented here can be applied to USApHMP instances of virtually any size (e.g. problems with hundreds or even thousands of nodes) since complexity due to size can be managed by efficient meta-heuristics and Monte Carlo simulation.

5.4 Experiments and Results

This section describes the computation experiments performed to test the efficiency of the proposed simheuristic algorithm. The procedures in our method have been implemented as Java® 7SE applications. The results presented in this section were obtained with a standard desktop computer with a 2.7 GHz Intel® Core™ i5 processor. There was no processing to parallel execution, without GPU inclusion in any part of the process.

The test problem used is the AP set of instances (Ernst and Krishnamoorthy 1998). This is the first study that has implemented simheuristic approach for solving the USA*p*HMP with stochastic demands and service quality thresholds, and consequently, no comparison with other literature studies is possible. By fully mimic USA*p*HMP, the AP data set provides the following information: *(i)* location of each node; *(ii)* flow between all nodes; *(iii)* collection cost; *(iv)* transfer cost and *(v)* distribution cost.

In order to test our simheuristic approach in the USA*p*HMP under uncertainty, we have extended the AP instances by employing the Log-normal probability distribution function for modeling the stochastic flow. In a real world application, historical data could be used to model each flow from one node to other by a different probability distribution function. As discussed by Juan et al. (2011), the Log-normal distribution is a more natural choice than the Normal distribution when modeling non-negative random variables. In particular, in the scenario already mentioned at the beginning of this chapter, the aim is to model the flow as a non-negative variable since it is supposed that users are always demanding content. The Log-normal has two parameters namely; the location parameter, μ_{ij} and the scale parameter σ_{ij}. These parameters are given by the following expressions:

$$\mu_{ij} = \ln(E[W_{ij}]) - \frac{1}{2} \cdot \ln\left(1 + \frac{Var[W_{ij}]}{E[W_{ij}]^2}\right) \tag{5.5}$$

$$\sigma_{ij} = \left|\sqrt{\ln\left(1 + \frac{Var[W_{ij}]}{E[W_{ij}]^2}\right)}\right| \tag{5.6}$$

Furthermore, the QoS thresholds are generated as follows: Given the best known solutions (BKS) reported in Alvarez Fernandez et al. (2018), we take the hub locations and calculate the total flow going through every hub of the solution due to the quantity of flow every node assigned to them, sends and receives, that is, if nodes i and j are allocated to hub k, we sum the amount of flow that goes through k taking into account the flow that nodes i and j send and receive to/from the remaining nodes of the network. After we calculate this total flow for every hub from the BKS, we set those total flows as thresholds for the given hubs. In order to generate the vector of thresholds, we set for every node (non-hub nodes) of the network the maximum values we calculated before.

Before providing and discussing the results for all instances from the AP dataset, the parameters and input values used on the implementation of our methodology are given. For each instance, 10 different seeds generated randomly were employed in order to execute the test of the proposed algorithm. As presented previously in Chapter 4, the maximum number of iterations on the first part is set at (*maxIterations*) = 5000; the Geometric distribution parameter for biased randomized allocation of the nodes to the hubs (β) = $0.1 \leq \beta \leq 0.3$; the percentage of hubs to remove (*per*) = $0.05 \leq per \leq 0.1$; the maximum number of best solutions (*solutionSet*) = 20. Besides, the number of run employed in the simulation phase (*longSimulation*) = 10000.

We considered three different scenarios regarding the variance levels k of each instance assuming relatively low, medium and high variances. The variances were set at the following values $Var[W_{ij}] = \{5, 10, 20\}[w_{ij}]$. In order to see the variation on the flow when the three different levels mentioned before were applied, we made a test employing the AP instance with $n = 10$ nodes and $p = 2$ hubs and the results are depicted in Figure 5.3. It is observed that as variance decreases the curve rises more sharply.

The computational analysis are summarized in Table 5.3 which report the results for

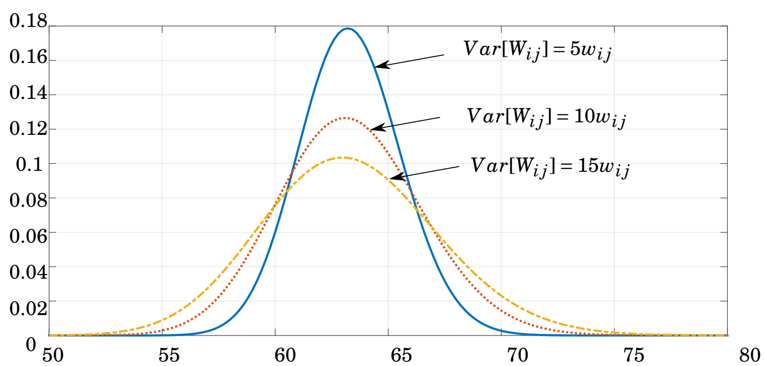


Figure 5.3: The stochastic flow on the AP instance with $n = 10$ and $p = 2$ modeled with the Log-normal probability distribution function for different kinds of variance levels k, specifically $k = \{5, 10, 15\}$.

Table 5.3: Results for the Australian Post instances considering the demand as stochastic

				k=5				k=10				k=20			
Instance			**BDS-D**	**BDS-S**		**BSS-S**		**BDS-S**		**BSS-S**		**BDS-S**		**BSS-S**	
	n	*p*	*Cost(1)*	*Cost(2)*	*(2) - (1)*	*Cost(3)*	*(3) - (1)*	*Cost(4)*	*(4) - (1)*	*Cost(5)*	*(5) - (1)*	*Cost(6)*	*(6) - (1)*	*Cost(7)*	*(7) - (1)*
Small	10	2	167493.08	177385.77	5.91%	175629.47	4.86%	187637.83	12.03%	183284.21	9.43%	208928.08	24.74%	186915.84	11.60%
		3	136008.13	146407.53	7.65%	146407.53	7.65%	157227.14	15.60%	146142.85	7.45%	179802.18	32.20%	146142.78	7.45%
		4	112396.07	123029.54	9.46%	122612.66	9.09%	134266.34	19.46%	133515.83	18.79%	158037.22	40.61%	144462.98	28.53%
		5	91105.38	101551.14	11.47%	101338.33	11.23%	112537.73	23.52%	110547.87	21.34%	135618.99	48.86%	120016.81	31.73%
	20	2	172816.68	182918.78	5.85%	182135.60	5.39%	193340.08	11.88%	192130.84	11.18%	215015.29	24.42%	194190.68	12.37%
		3	151533.08	161894.10	6.84%	159974.41	5.57%	172649.40	13.94%	168700.12	11.33%	195082.75	28.74%	181664.12	19.88%
		4	135624.88	146334.42	7.90%	145317.20	7.15%	157658.15	16.25%	144066.14	6.22%	181609.94	33.91%	147082.74	8.45%
		5	123130.09	134012.34	8.84%	132948.75	7.97%	145549.59	18.21%	143823.70	16.81%	169911.94	37.99%	160157.71	30.07%
	25	2	175541.98	186003.19	5.96%	183797.62	4.70%	197038.36	12.25%	190998.64	8.81%	220104.12	25.39%	193529.07	10.25%
		3	155256.32	165887.00	6.85%	162475.02	4.65%	177212.46	14.14%	173567.54	11.79%	201233.19	29.61%	189842.63	22.28%
		4	139197.17	149811.40	7.63%	145306.88	4.39%	161367.61	15.93%	154271.14	10.83%	186717.50	34.14%	174502.34	25.36%
		5	123574.29	134273.32	8.66%	133845.02	8.31%	145915.99	18.08%	141528.60	14.53%	171093.16	38.45%	161408.64	30.62%
	40	2	177471.68	187907.30	5.88%	187028.27	5.38%	198934.12	12.09%	195224.85	10.00%	222093.47	25.14%	202617.10	14.17%
		3	158830.55	169789.00	6.90%	168257.85	5.94%	181627.22	14.35%	177891.50	12.00%	206805.38	30.21%	186311.15	17.30%
		4	143968.88	154605.38	7.39%	154142.95	7.07%	166074.07	15.35%	146449.80	1.72%	190172.15	32.09%	149978.03	4.17%
		5	134264.97	142513.75	6.14%	138362.87	3.05%	147318.92	9.72%	138980.12	3.51%	158500.39	18.05%	142793.15	6.35%
	50	2	178484.28	189146.41	5.97%	188919.71	5.85%	200330.90	12.24%	198347.42	11.13%	223735.73	25.35%	212085.33	18.83%
		3	158569.93	169203.59	6.71%	168512.69	6.27%	180551.49	13.86%	176837.90	11.52%	204584.59	29.02%	181048.31	14.18%
		4	143378.04	154293.27	7.61%	153904.98	7.34%	166284.89	15.98%	163013.25	13.69%	192108.39	33.99%	182790.70	27.49%
		5	135587.81	142504.11	5.10%	142007.08	4.73%	149606.32	10.34%	139818.99	3.12%	167868.07	23.81%	138733.94	2.32%
Large	100	5	143382.99	145746.07	1.65%	145178.46	1.25%	150317.46	4.84%	145919.20	1.77%	162631.99	13.42%	150365.92	4.87%
		10	106469.57	118158.72	10.98%	115153.22	8.16%	120385.14	13.07%	119346.82	12.09%	146556.07	37.65%	139285.38	30.82%
		15	90522.52	100869.52	11.43%	97083.28	7.25%	106711.86	17.88%	105342.41	16.37%	107798.49	19.08%	99998.60	10.47%
		10	80270.96	95947.88	19.53%	93062.93	15.94%	92479.01	15.21%	84610.26	5.41%	99485.36	23.94%	83883.10	4.50%
	200	5	140062.65	181129.02	29.32%	179353.42	28.05%	159860.07	14.13%	155249.17	10.84%	159141.29	13.62%	150218.32	7.25%
		10	110147.66	144304.45	31.01%	141892.28	28.82%	125134.45	13.61%	119062.27	8.09%	150202.60	36.36%	136300.00	23.74%
		15	94459.20	131336.07	39.04%	129395.15	36.99%	101126.38	7.06%	96771.65	2.45%	119535.88	26.55%	109665.95	16.10%
		20	84955.33	121978.86	43.58%	117051.01	37.78%	96999.34	14.18%	92334.60	8.69%	106456.20	25.31%	100371.30	18.15%

both the deterministic and the stochastic USApHMP with low, medium and high variance level. The first three columns indicate the size of the instances which are classified as *small* and *large* instances, specifically *small* includes networks with size up to 50 nodes and *large* consider networks with nodes from 100 to 200 nodes. In the following, the number of nodes "n" and the number of hubs "p" given by each instance are given. Next, in the column "*Cost (1)*", the best deterministic solution for the deterministic version of the USApHMP is given, these solutions are also referred as *best deterministic solution - deterministic* scenario, "BDS-D". Next, the four following columns are the obtained results with low variance scenario, i.e., $k = 5$. The column "*Cost(2)*" includes the best expected deterministic costs obtained by Monte Carlo simulation. Also, the percentage gap when compared against the BKS of the deterministic USApHMP is given in the column "*(2) - (1)*". The results related to expected costs obtained when applied simulation to the deterministic scenario (columns *Cost(2)* and *(2) - (1)*) are referred as *best deterministic solution - stochastic scenario* "BDS-S". Following, the total expected costs "*Cost (3)*" when considering that flows are stochastic instead of deterministic and the percentage gap are reported (column *(3) - (1)*). This gap is with respect to the best found solution given in *Cost (1)* that is, the real gap between our best solution obtained when applied the simheuristic algorithm on the stochastic scenario and the deterministic USApHMP BKS. These results (columns *Cost (3)* and *(3) - (1)*) are referred as *best stochastic solution - stochastic scenario* "BSS-S". The following columns report the BDS-S and BSS-S for medium and high variance level scenarios with their referred gaps, respectively.

The Figure 5.4 presents a multiple boxplot comparison of the results obtained when applied the proposed simheuristic methodology for solving the AP dataset instances among different variance level scenarios. The multiple boxplot comparison summarizes the BSS-S from Table 5.3. This boxplot gives a general perspective of how the different level of variance presented in flows impact the choice of the most reliable p-hub design network.

Moreover, for illustration purposes, Figure 5.5 displays the configuration of the network when the algorithm is applied for solving the instance with $n = 20$ and $p = 3$. On one hand, 5.5a displays the distribution of the nodes in the network for the instance when the variance presented on the flow is very small, that is, are considered as deterministic. Figure 5.5b shows the distribution of the nodes for the same network but in that case, the flow is stochastic with a low-level variance, each set of nodes assigned to a hub has been represented with a circle and hubs are the squared ones, additionally, the nodes selected as hubs and its assignments are also depicted. The position of the nodes in the plots have been truly maintained as given in the AP instances to better depict the assignments. On the other hand, Figure 5.6

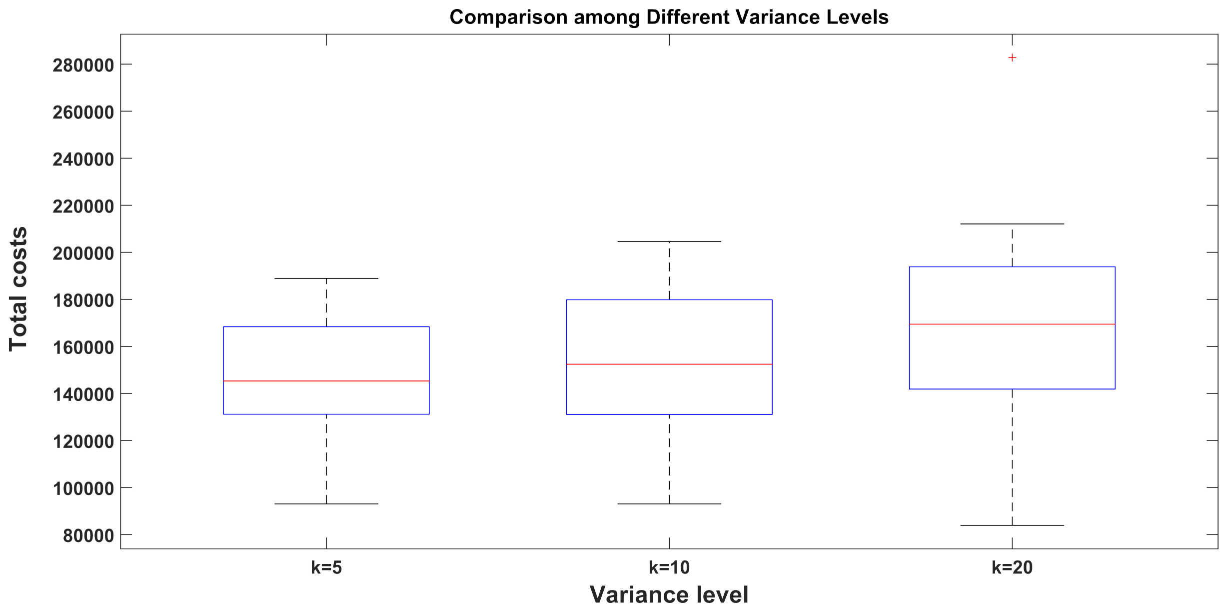


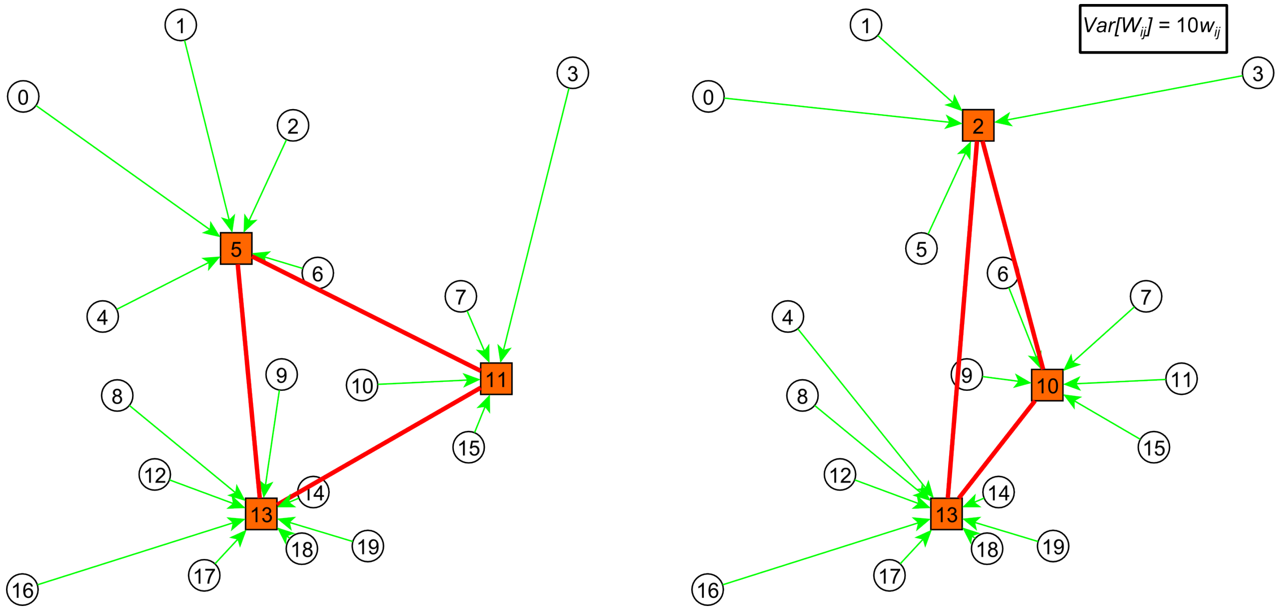
Figure 5.4: Boxplot comparing the BSS-S solution of all AP instances among different level variance scenarios

depicts the configuration of the same network when the variance presented on the flow is medium and high level. Note that both configurations are slightly different from the ones reported in Figure 5.5. In Figure 5.6a the selected hubs are nodes 13, 10 and 1, however, this configuration changes when variance increase from $k = 16$ to $k = 20$, precisely, the nodes selected as hubs in the last scenario are 14, 13 and 5.

5.5 Discussion

As the results indicate, the algorithm is able to find all optimal solutions for small size problems considering $n = \{10, 20, 25\}$ and all BKS for medium and large size problems ($n = \{40, 50, 100, 200\}$) when the demand is deterministic. This was expected since the simheuristic algorithm was built upon the BRILS methodology presented in Chapter 4. The solutions for the deterministic version of the problem were obtained in order to test the efficiency of the simheuristic algorithm to find the deterministic solution under the stochastic scenario.

The solutions obtained when considered the demand as stochastic with low, medium and high level variance scenarios were also compared. As seen in Table 5.3, the results show that assuming a problem being deterministic can lead to solutions with poor performance



(a) Network configuration of the BDS-D with no variance. Cost = 151533.08

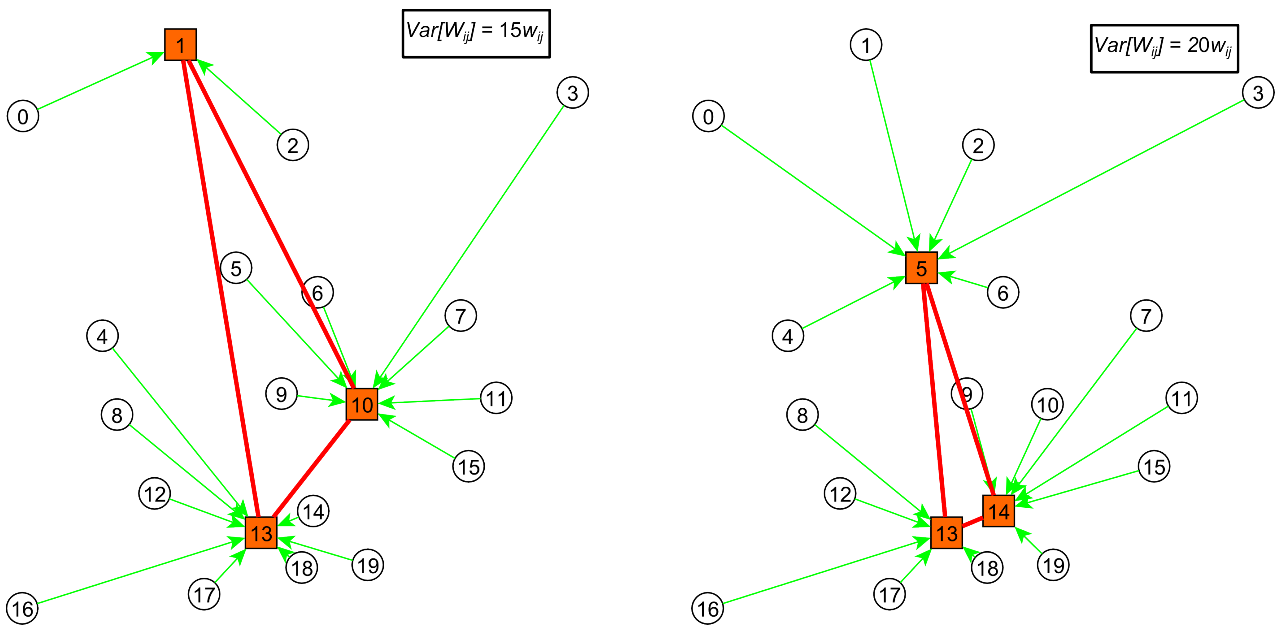
(b) Network configuration of the BSS-S with $k = 10$. Cost = 168700.12

Figure 5.5: Comparison between solutions for the deterministic and stochastic USApHMP, respectively, on a network with 20 nodes and its structure for $p = 3$.

even in scenarios with a relatively low variance. In all experiments, the expected total costs obtained with the BSS-S are better that ones obtained with the BDS-S. The reason is that the deterministic solution is not balanced, and a high variance results in an increasing of the expected total cost. Furthermore, Figures 5.5 and 5.6 illustrate the case of the instance with 20 nodes and 3 hubs without and with variance. In Figure 5.5 the left and right plots represent the best deterministic and the best stochastic solution, respectively. Note that, the variance presented on the flow transited among the network impact the hubs location and the allocation of the nodes to them. Moreover, in Figure 5.6 the differences in the configuration when variance levels increased is observed.

5.6 Concluding Remarks

This chapter considered the USApHMP under uncertainty assumptions. A simheuristic approach, which combines BRILS with Monte Carlo simulation was proposed. The proposed methodology seeks location of p-hubs among nodes in a network and the allocation of the



(a) Network configuration of the BSS-S with $k = 15$ Cost = 178691.05

(b) Network configuration of the BSS-S with $k = 20$ Cost = 181664.12

Figure 5.6: Comparison between solutions stochastic USApHMP with different variance levels, respectively, on a network with 20 nodes and its structure for $p = 3$.

remaining nodes to them so that the overall transportation cost and uncertainty presented in flow are minimized. The mentioned uncertainty comes from incoming traffic to hubs. Also, QoS thresholds are proposed so to mimic a communication network design problem. The simheuristic approach provides the decision maker with a set of alternative solutions considering levels of variance presented on flows. Each of the solutions, characterized by their total estimated costs reflecting the possibility of that solution being a feasible one. Although other previous works have proposed to benefit from the stochastic programming and/or robust optimization, they usually consider a limited set of scenarios no matter which of them are realized. On the contrary, the simheuristic approach presented in this chapter relaxes most on the assumptions and, therefore, it allows for considering more realistic user demand scenarios. Thus, the simheuristic approach is valid for virtually any statistical distribution - the one that best fits historical data on user demands. A complete set of tests have been performed to illustrate the methodology and analyze its efficiency as well as its potential benefits.

(III) CONCLUSIONS AND FUTURE RESEARCH

Chapter 6

Conclusions

Due to the increasing amount of new optimization challenges in the telecommunication industry, metaheuristic algorithms have received an increasing attention during the last two decades. This thesis offers a review of recent works proposing different metaheuristic approaches to efficiently deal with most of the recent challenges. Three main telecommunication fields have been the focus of this work: networks design, routing, and allocation. Inside these fields several sub-problems have been identified, since the design of telecommunication systems comprises many different aspects, including: topologies, location of resources in links and nodes, uni-cast vs. multi-cast routing, reliability or availability of networks.

After the introduction chapter, Chapter 2 presents a description of the main metaheuristics cited throughout the dissertation, and later Chapter 3 presented a literature review. From this literature review, it can be concluded that, so far, population-based metaheuristics –such as Genetic Algorithms, Ant Colony Optimization, and Particle Swarm Optimization– have been more popular than single-solution ones –such as Tabu Search, GRASP, or Simulated Annealing. This is probably due to the fact that most authors belong to Computer Science area, where the use of population-based metaheuristics is widely extended. However, this also offers a good chance to researchers in other fields –e.g., Operations Research or Industrial Engineering– to propose new single-solution approaches that typically require less parameters and might be easier to implement in real-life scenarios.

Besides telecommunications and computing networks, hub location problems are receiving increased attention in transportation and logistics. In the literature review presented in the previous chapter one can find many types of hub location related problems and they differ from each other depending by the way in which the requested points are assigned or allocated to hubs. In Chapter 4, we consider a well-known optimization problem when it comes to designing telecommunication networks: the USApHMP, where a predefined number of hubs have unlimited capacity, each non-hub node is allocated to a unique hub and the number of hubs is given in advance. In order to solve the USApHMP three methodologies where proposed. The first one consisted on a two-stage metaheuristic based on the combination of biased-randomization technique with an iterated local search framework and the other two methodologies were based on the artificial immune systems, the CLONALG and the optAiNet algorithms. The proposed algorithms were tested using a well-known benchmark, the AP data set, which contribute to validate the proposed methodologies.

Finally, Chapter 5 presents an algorithm which combines the algorithm designed for the deterministic version of the USApHMP, the BRILS, with ideas from the simheuristic framework, in order to evaluate the performance of the resulted algorithm when applied for solving the USApHMP under uncertainty scenarios. The performance of the algorithm is evaluated with a set of benchmarks available on the literature adapted for the uncertainty scenario, using statistical techniques for evaluate the performance of the designed solution.

6.1 Accomplishments and Main Contributions

The main objective of this thesis was to study the application of metaheuristics and simheuristics methodologies in order to deal with deterministic and stochastic versions of combinatorial optimization problems related to telecommunication systems. In the course of accomplishing the research questions and its related research objectives introduced in Chapter 1, a series of original contributions were generated.

Which are the most used methodologies found in the literature by researchers when solving problems related to telecommunications systems?

The use of metaheuristics has increased during the last two decades to address many telecommunication problems, such as hub location, topology design, reliable server assignment, frequency assignment and wavelength allocation, routing, etc. *Chapter 3 is a contribution to the literature of the application and development of metaheuristics for solving current optimization problems in the telecommunication field*. Although many network design models reviewed addressed construction of a better system in terms of network topology and routing schemes, the models are often created under simple assumptions that ignore the substantial aspects of current communication networks. The literature review showed the need for developing faster heuristic and metaheuristic algorithms, which might be able to provide good solutions in almost real time for large-sized instances with thousands or even millions of nodes. On the other hand, it is clear the high level of dynamism around the telecommunication problems reviewed. However, there is a lack of proposals to solve these problems considering the dynamism that appear in real-world telecommunication problems.

Are those referred methodologies fast enough and efficient when applied to problems related to the design of the communication networks that engineers and decision makers face?

From the literature review one can conclude that the location of resources in telecommunications has been identified as a significant operational and defense strategy because the performance of current networks is highly reliant of resources locations and any malfunction at a resource may cause degradation of the entire network's ability to transfer flow. In particular, Hubs are critical elements of telecommunication and transportation networks since they play a vital role as a switching or transshipment point, allowing mass traffic movement. In this regard, *Chapter 4 is a computational contribution applied on the deterministic version of the Uncapacitated Single Allocation p-Hub Median Problem*. The contributions are highlighted in three ways. Accordingly:

- the development of a fast and easy-to-implement algorithm, based on biased randomization techniques (Grasas et al. 2017) and iterated local search methodology, which allows to generate "good" solutions to the USA*p*HMP in just a few milliseconds;

- the demonstration of the potential advantage of the proposed methodology by finding optimal solutions for a networks up to 100 nodes in size and near-optimal solutions to 200 nodes network, in all cases in low computing times; and

- the development of two algorithms based on the artificial immune systems framework. The CLONALG and the optAiNet heuristics with a series of numerical experiments on a well-known benchmark, which contribute to validate the proposed methodologies and highlight future research lines.

How the hybridization of metaheuristic-based methodologies with simulation techniques (simheuristics) can help to solve "real world" problems commonly found in telecommunication systems?

With the emergence of new mobile and decentralized telecommunication systems, aspects such as uncertainty and dynamism are more relevant than ever. For this reason, *Chapter 5 is a computational contribution applied on a stochastic Uncapacitated Single Assignment p-Hub Median Problem*. The problem considers the stochastic nature of the demand of sending flow from origin nodes to destination nodes and a simheuristic method to find approximate solutions to networks up to 200 nodes in size is developed. Additionally, the benefits of accounting for stochastic demands are presented.

In the following, we include a list containing the publications generated during the development of this thesis:

Published JCR Indexed Papers

- **S. Alvarez Fernandez**, A. A. Perez, J. de Armas, D. G. Silva, and D. Ferone, "A Two-Stage Biased-Randomized Iterated Local Search for the Uncapacitated Single Allocation p-Hub Median Problem", published in *Transactions on Emerging Telecommunications Technologies*, ISSN: 2161-3915, 2018.

- **S. Alvarez Fernandez**, A. A. Perez, J. de Armas, D. G. Silva, and D. Riera, "Metaheuristics in Telecommunication Systems: network design, routing and allocation problems", published in *IEEE Systems Journal*, ISSN: 1932-8184, 2018.

Under review Journal Papers

- **S. Alvarez Fernandez**, D. Ferone, A. A. Perez, and D. Tarchi, "Single Allocation p-Hub Median Problem with Stochastic Demands and Service Quality Thresholds". 2018.

Conference Papers Indexed in ISI-WoS or Scopus

- **S. Alvarez Fernandez**, R. F. Attux, D. G. Fantinato, J. R. Montalvão, D. G. Silva, "Immune-Inspired Optimization with Autocorrentropy Function for Blind Inversion of Wiener Systems," *Proceedings of the 2018 IEEE Congress on Evolutionary Computation (CEC)*, 2018.

Conference Papers/Abstracts with Peer-reviewing Process

- **S. Alvarez Fernandez**, A. A. Perez, D. Riera, J. de Armas and D. G. Silva, "A Simheuristic Algorithm for the Uncapacitated and Stochastic Hub Location Problem", presented in *12th Metaheuristics International Conference, (MIC'17)*, Barcelona, Spain, 2017.

- G. L. Nobrega, V. J. Tasso, A. Souza, **S. Alvarez Fernandez**, D. G. Silva, "Artificial Immune Systems for Solving the Uncapacitated Single-Allocation p-Hub Median Problem", presented in *Encontro Nacional de Inteligência Artificial e Computacional (ENIAC) 2018*, 2018.

6.2 Research Impact

The results presented in this thesis impacts a wide range of applications. In particular, with the results and findings presented, this thesis impacts the field of combinatorial optimization problems related to telecommunication systems by contributing a methodology which combines a biased randomization technique with an iterated local search approach to solve the USApHMP as well as the combination of the proposed methodology with Monte Carlo simulation techniques which yields significant gains in solution quality and computational times as compared to related works. Moreover, two AIS-based metaheuristics are presented revealing themselves as efficient approaches for solving the same problem.

Besides, this thesis allows the telecommunication research community to know the current state of art, the possibilities and trends related to the application of new methodologies involving metaheuristics in order to solve optimization related problems. The hope is that future research will continue developing stochastic optimization models and methods building on, among others, the model designs and methodologies proposed in the preceding chapters. In this respect, there are a number of possible work to be done, and some of them are introduced below.

6.3 Emerging Trends and Future Work

After the work done in the thesis, some trends in the field have been detected. One trend in the area of telecommunication networks is the search for more effective designs. On one hand, in the resource allocation problem for robust network design, a mis-allocation of capacities could have two side effects: *(i)* significant data loss for certain traffic –i.e., a non-robust design; and (*ii*) over-sizing of installed capacities –i.e., a robust design at a high investment cost. On the other hand, Regarding the test cases and performance evaluation of the proposed algorithms found in the literature review, one can notice the predominance of small- and medium-sized instances. This is probably due to the computing time required to solve large-sized instances. However, in most real-life telecommunication systems a solution might be required after a few seconds or milliseconds. In this context, the use of fast metaheuristic algorithms becomes necessary to deal with the associated design communication issues.

Another observed trend is the current predominance of population-based metaheuristics over single-solution ones. In our view, both are valid approaches and, therefore, much research can be done yet regarding the use of single-solution metaheuristics in the telecommunication field. In fact, single-solution metaheuristics might offer some advantages over population-based approaches, since the former typically employ less parameters and might be easier to implement in practical applications.

Also, due the growing developments in metering and digital technologies, the amount of different electronic mobile devices around us is increasing. This mobile technology is massively being used today, which arises several challenges related to the use of 5G communication technologies, cloud computing services, security, trust and privacy, etc. During the next years, new challenging problems will emerge in the telecommunication industry. For instance, we observe an expansion of the wireless technologies, an increasing demand for higher QoS, a continuous raising in traffic flow, and a strong growth in the use of mobile ad hoc and peer-to-peer networks. These emerging challenges raise new COPs characterized by uncertainty and dynamic conditions, since this way the modeling of the problems is more realistic and solutions are more flexible when the environment changes. While hybridization of simulation and optimization has recently been developed and gained popularity in solving stochastic COPs (Juan et al. 2015*a*), the majority of telecommunications problems mentioned above have not yet been extensively addressed by simheuristics.

Recently, an interesting methodology for addressing dynamic COPs has been formally proposed in (Calvet et al. 2017). The authors explain how the combined used of machine learning techniques and metaheuristics — i.e., learnheuristics — can be useful to deal with the dynamism that appears in many realistic COPs. Therefore, the application of these hybrid methodologies opens new possibilities when solving real-world telecommunication problems.

After identifying the emerging trends and challenges, there are yet a number of possible future work to be done, and below some of them are listed.

- The algorithms proposed in chapters 4 are not limited to the USApHMP and can be extended to different directions. First, by considering larger problems than those solved in this thesis. Secondly, by applying the algorithm to other hub location related problems, for instance, the multiple as well as the capacitated allocation hub-and-spoke location problems. Figure 6.1 is an example of a multiple allocation hub-and-spoke network where collection and distribution for any node may be performed using different hubs in order to minimize costs.

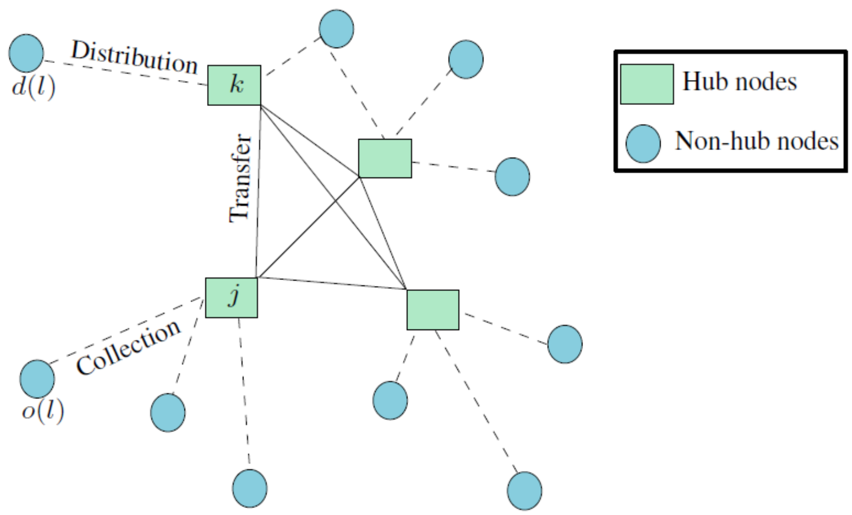


Figure 6.1: Multiple allocation hub-and-spoke network

- In addition, thinking about strategies to decrease execution time for optAiNet and implement effective restarts without significant increase in the execution cost of CLON-ALG are issues to be discussed in future work, which may bring even more growth to the metaheuristics discussed and the application itself. Finally, the results still need to be compared with those obtained from other approaches in the literature.

- Moreover, the application of USApHMP can be evaluated as a decision-making tool for Software-Defined Network (SDN) projects (Lange et al. 2015), where the network engineer is involved with crucial aspects like defining the number and position of each SDN controller — which plays a role equivalent to a hub in USApHMP— such that throughput and quality of service metrics must be optimized. Figure 6.2 illustrates the SDN context.

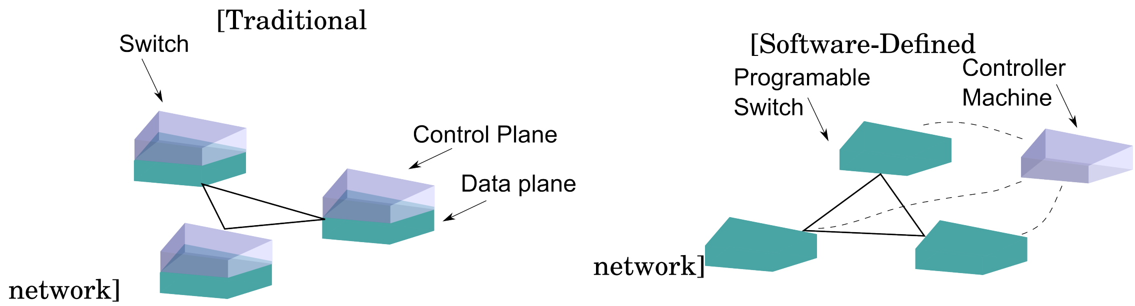


Figure 6.2: Comparison between a traditional network and a software defined network (Maleki et al. 2017)

- There are two types of channel allocation; fixed and dynamic. When the demand is known, fixed channel allocation is the best option. However, in the real world, demand is unlikely to be known *a priori*. Demand changes on a daily basis, therefore it is impractical to limit the number of available channels within a cell. Figure 6.3 illustrates a dynamic channel allocation, in which channels are dynamically assigned to cells are used. In dynamic channel allocation, the trends of past allocations are examined and used. The supply of future channels is done based on the analysis, and at the same time extra channels are provided to deal with unexpected scenarios such as network breakdown and a sudden surge of demand. This is another problem in which the application of simheuristics have a visible potential.

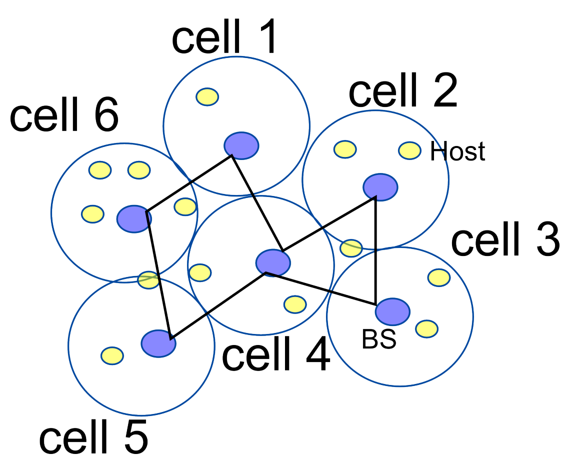


Figure 6.3: Channel allocation in base station scenario (Mishra and Saxena 2012)

- Likewise, in order to include dynamic inputs in the optimization models, learnheuristic algorithms combining metaheuristics with machine learning techniques can be studied in the field of hub location problems as well as other network design problems since some problems in the telecommunication field are characterized by inputs that are not fixed in advance (e.g., the performance of transmission devices). Summarizing, the development of effective metaheuristics methods which include the analysis of dynamic and uncertainty inputs seems to be necessary for the future cases (Figure 6.4). In this context, the simheuristic algorithm proposed in Chapter 5 may be also extended in several directions.

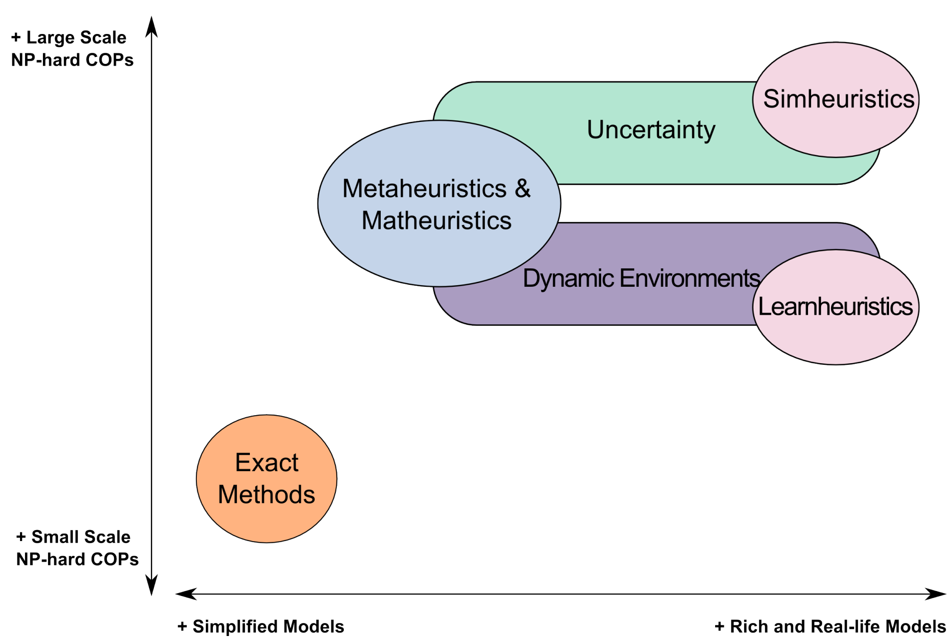


Figure 6.4: Hybridization of metaheuristics with simulation and machine learning techniques

6.4 Thesis Committee and Research Groups

The thesis committee for this research is composed by:

Prof. Dr. Daniel Riera i Terrén is the director of Computer Science Engineering in the Department of Computer Science, Multimedia and Telecommunication (EIMT) by the Universitat de Catalunya (UOC). Dr. Riera holds a Ph.D. in Computer Science by the Universitat Autónoma de Barcelona (UAB). His main research scopes include the model of discreet systems using Petri nets, the optimization using Constraint Programming techniques, the use of metaheuristics and simulation, and the gamification of learning.

Prof. Dr. Jésica de Armas Adrián is Assistant Professor at the Department of Economics and Business, Universitat Pompeu Fabra. She holds a Ph.D. in Computer Science by the Universidad de La Laguna. Her research interests include Operations Research, Combinatorial Optimization, Metaheuristics, Machine Learning, Simulation, Vehicle Routing, Scheduling, Logistics, Production, Transportation, and Health Care Optimization. As a result of her research, she has published a large number of articles in high impact factor journals and she has participated in a large number of relevant conferences in the field.

Prof. Dr. Daniel Guerreiro e Silva is Assistant Professor at the Department of Electrical Engineering (ENE) of University of Brasília (UnB), Brazil. Dr. Silva holds a Ph.D. in Electrical Engineering by the University of Campinas. His main research areas are information theoretic learning, adaptive signal processing, bio-inspired computing and signal processing over finite fields. His teaching activities mainly concentrate in operating systems, algorithms and data structures, probability and introduction to stochastic processes and information theory.

Prof. Dr. Angel A. Juan P. is a Full professor in the Computer Science, Multimedia and Telecommunication Department at the Open University of Catalonia (UOC). He is also a Researcher at the Internet Interdisciplinary Institute (IN3). Dr. Juan holds a Ph.D. in Applied Computational Mathematics (UNED). He has also completed a postdoctoral internship at the MIT Center for Transportation & Logistics. His research interests include Randomized Algorithms and Simheuristics in Logistics, Production, and Internet Computing, as well as Educational Data Analysis & Mathematical e-Learning.

My research was carried out within the Internet Computing Systems Optimization (ICSO) group from the UOC and the Digital Signal Processing Group (GPDS) from the UnB. The ICSO has two research lines as its name suggested; Internet computing and systems optimization. The first line focuses on the development of distributed and parallel systems at different scales (small groups, clusters, or Internet). The later one focuses on the use of industrial analytic (algorithms and software solutions) to support complex decision-making in the areas of transportation and logistics, production, real-time positioning, smart cities, and finance. The GPDS is part of the ENE at the Faculty of Technology of UnB, although some members are experts in Computer Science and related fields. The focus of GPDS is the development of new methods for signal processing with several applications in a variety of areas. Among the group's areas of study, digital media processing, data compression, digital network communication can be mentioned.

Bibliography

Aarts, E., Aarts, E. H. and Lenstra, J. K. (2003), *Local search in combinatorial optimization*, Princeton University Press.

Adibi, A. and Razmi, J. (2015), '2-stage stochastic programming approach for hub location problem under uncertainty: A case study of air network of iran', *Journal of Air Transport Management* **47**, 172–178.

Ahmad, A., Ahmad, S., Rehmani, M. H. and Hassan, N. U. (2015), 'A survey on radio resource allocation in cognitive radio sensor networks', *IEEE Communications Surveys & Tutorials* **17**(2), 888–917.

Ahmadi, T., Karimi, H., Davoudpour, H. and Hosseinijou, S. A. (2015), 'A robust decision-making approach for p-hub median location problems based on two-stage stochastic programming and mean-variance theory: a real case study', *The International Journal of Advanced Manufacturing Technology* **77**(9-12), 1943–1953.

Alumur, S. A., Nickel, S. and Saldanha-da Gama, F. (2012), 'Hub location under uncertainty', *Transportation Research Part B: Methodological* **46**(4), 529–543.

Alumur, S. and Kara, B. Y. (2008), 'Network hub location problems: The state of the art', *European journal of operational research* **190**(1), 1–21.

Alvarez Fernandez, S., Ferone, D., Juan, A. A., Silva, D. and Armas, J. (2018), 'A two-stage biased-randomized iterated local search for the uncapacitated single-allocation p-hub median problem', *Transactions on Emerging Telecommunications Technologies* pp. 1–14.

Amin-Naseri, M. R., Yazdekhasti, A. and Salmasnia, A. (2016), 'Robust bi-objective optimization of uncapacitated single allocation p-hub median problem using a hybrid heuristic algorithm', *Neural Computing and Applications* pp. 1–22.

Aragón, V. M., de Miguel, I., Durán, R. J., Merayo, N., Aguado, J. C., Fernández, P., Lorenzo, R. M. and Abril, E. J. (2007), *A New Algorithm for the Distributed RWA Problem in WDM Networks Using Ant Colony Optimization*, Springer Berlin Heidelberg, pp. 299–308.

Arin, A. and Rabadi, G. (2013), Memory and learning in metaheuristics, *in* 'Artificial Intelligence, Evolutionary Computing and Metaheuristics', Springer, pp. 435–476.

Arteta, A. and Pinto-Roa, D. P. (2015), Multiobjective robust network design under uncertain traffic an approach based on evolutionary algorithm, *in* 'Latin American Computing Conference', pp. 1–10.

Aydin, I., Karakose, M. and Akin, E. (2011), 'A multi-objective artificial immune algorithm for parameter optimization in support vector machine', *Applied Soft Computing* **11**(1), 120–129.

Azizi, N., Vidyarthi, N. and Chauhan, S. S. (2018), 'Modelling and analysis of hub-and-spoke networks under stochastic demand and congestion', *Annals of Operations Research* **264**(1-2), 1–40.

Bedoui, A., Debreux, P., Idoumghar, L., Schott, R. and Schott, T. (2014), Steps toward the design of hybrid metaheuristics for the multi-objective frequency assignment problem in broadcasting, *in* 'Telecommunication Technologies (ISTT), 2014 IEEE 2nd International Symposium on', IEEE, pp. 208–213.

Ben-Tal, A. and Nemirovski, A. (2000), 'Robust solutions of linear programming problems contaminated with uncertain data', *Mathematical programming* **88**(3), 411–424.

Benmezal, L., Benhamou, B. and Boughaci, D. (2017), Some neighbourhood approaches for the antenna positioning problem, *in* 'Tools with Artificial Intelligence (ICTAI), 2017 IEEE 29th International Conference on', IEEE, pp. 1001–1007.

Bertsimas, D. and Sim, M. (2003), 'Robust discrete optimization and network flows', *Mathematical programming* **98**(1-3), 49–71.

Bidgoli, H. (2016), 'Handbook of computer networks: Key concepts, data transmission, and digital and optical networks', *Networks (PDF Download Available)* **20**, 6.

Birattari, M., Paquete, L., Stützle, T. and Varrentrapp, K. (2001), 'Classification of metaheuristics and design of experiments for the analysis of components', *Teknik Rapor, AIDA-01-05*.

Bitam, S. and Mellouk, A. (2013), 'Bee life-based multi constraints multicast routing optimization for vehicular ad hoc networks', *Journal of Network and Computer Applications* **36**(3), 981–991.

Blesa, M. J., Blum, C., Cangelosi, A., Cutello, V., NUOVO, A. D., Pavone, M. and Talbi, E.-G., eds (2016), *Hybrid Metaheuristics*, Springer International Publishing.

Blum, C. and Roli, A. (2003), 'Metaheuristics in combinatorial optimization: Overview and conceptual comparison', *ACM Compututing Surveys* **35**(3), 268–308.

Blum, C., Puchinger, J., Raidl, G. R. and Roli, A. (2011), 'Hybrid metaheuristics in combinatorial optimization: A survey', *Applied Soft Computing* **11**(6), 4135–4151.

Boschetti, M. A., Maniezzo, V., Roffilli, M. and Röhler, A. B. (2009), Matheuristics: Optimization, simulation and control, *in* 'International Workshop on Hybrid Metaheuristics', Springer, pp. 171–177.

Brabazon, A., O'Neill, M. and McGarraghy, S. (2015), *Natural Computing Algorithms*, Springer Publishing Company, Incorporated.

Bröring, A., Echterhoff, J., Jirka, S., Simonis, I., Everding, T., Stasch, C., Liang, S. and Lemmens, R. (2011), 'New generation sensor web enablement', *Sensors* **11**(3), 2652–2699.

Buriol, L., Resende, M., Ribeiro, C. and Thorup, M. (2005), 'A hybrid genetic algorithm for the weight setting problem in OSPF/IS-IS routing', *Networks* **46**(1), 36–56.

Burnet, F. M. (1978), 'Clonal Selection and After', *Theoritical Immunology* pp. 63–85.

Cabrera, G., Juan, A. A., Láázaro, D., Marquès, J. M. and Proskurnia, I. (2014), 'A simulation-optimization approach to deploy internet services in large-scale systems with user-provided resources', *Simulation* **90**(6), 644–659.

Calvert, K. L., Doar, M. B. and Zegura, E. W. (1997), 'Modeling internet topology', *IEEE Communications Magazine* **35**(6), 160–163.

Calvet, L., Armas, J. d., Masip, D. and Juan, A. A. (2017), 'Learnheuristics: hybridizing metaheuristics with machine learning for optimization with dynamic inputs', *Open Mathematics* **15**(1), 261–280.

Campbell, J. F. (1992), 'Location and allocation for distribution systems with transsshipments and transportion economies of scale', *Annals of operations research* **40**(1), 77–99.

Campbell, J. F. (1994), 'Integer programming formulations of discrete hub location problems', *European Journal of Operational Research* **72**(2), 387–405.

Carello, G., Della Croce, F., Ghirardi, M. and Tadei, R. (2004*a*), 'Solving the hub location problem in telecommunication network design: A local search approach', *Networks* **44**(2), 94–105.

Carello, G., Della Croce, F., Ghirardi, M. and Tadei, R. (2004*b*), 'Solving the hub location problem in telecommunication network design: A local search approach', *Networks* **44**(2), 94–105.

Chakraborty, D. and Debbarma, K. (2017), 'Q-car: an intelligent solution for joint qos multicast routing and channel assignment in multichannel multiradio wireless mesh networks', *Applied Intelligence* **47**(1), 13–27.

Chamberland, S., Sansò, B. and Marcotte, O. (2000), 'Topological design of two-level telecommunication networks with modular switches', *Operations Research* **48**(5), 745–760.

Charnes, A. and Cooper, W. W. (1959), 'Chance-constrained programming', *Management science* **6**(1), 73–79.

Chen, J.-F. and Wu, T.-H. (2008), 'A note on solution of the uncapacitated single allocation p-hub median problem', *Journal of the Chinese Institute of Industrial Engineers* **25**(1), 11–17.

Chen, L.-J., Chang, G.-W., Pai, H.-T., Yen, L. and Lin, H.-P. (2016), Resource allocation algorithms for lte over wi-fi spectrum, *in* 'Computer Symposium (ICS), 2016 International', IEEE, pp. 701–705.

Chica, M., Juan Pérez, A. A., Cordon, O. and Kelton, D. (2017), 'Why simheuristics? benefits, limitations, and best practices when combining metaheuristics with simulation'.

Chinoy, B. and Braun, H.-W. (1992), The national science foundation network, Technical report, Technical Report GA-A21029, SDSC.

Chlamtac, I., Ganz, A. and Karmi, G. (1992), 'Lightpath communications: an approach to high bandwidth optical WAN's', *IEEE Transactions on Communications* **40**(7), 1171–1182.

Christodoulopoulos, K., Tomkos, I. and Varvarigos, E. A. (2010), Routing and spectrum allocation in OFDM-based optical networks with elastic bandwidth allocation, *in* 'IEEE Globecom', pp. 1–6.

Contreras, I., Cordeau, J.-F. and Laporte, G. (2011), 'Stochastic uncapacitated hub location', *European Journal of Operational Research* **212**(3), 518–528.

Cook, W. J. (2011), *In pursuit of the traveling salesman: mathematics at the limits of computation*, Princeton University Press.

Correia, I., Nickel, S. and Saldanha-da Gama, F. (2018), 'A stochastic multi-period capacitated multiple allocation hub location problem: Formulation and inequalities', *Omega* **74**, 122–134.

Cox, L. A. and Sanchez, J. R. (2000), 'Designing least-cost survivable wireless backhaul networks', *Journal of Heuristics* **6**(4), 525.

Dantzig, G. B. (2010), Linear programming under uncertainty, *in* 'Stochastic programming', Springer, pp. 1–11.

Das, P. P., Chakraborty, N. and Allayear, S. M. (2015), Optimal coverage of wireless sensor network using termite colony optimization algorithm, *in* 'International Conference on Electrical Engineering and Information Communication Technology', pp. 1–6.

Das, S., Barani, S., Wagh, S. and Sonavane, S. (2016), Energy efficient and trustable routing protocol for wireless sensor networks based on genetic algorithm (e2trp), *in* 'Automatic Control and Dynamic Optimization Techniques (ICACDOT), International Conference on', IEEE, pp. 154–159.

Dasgupta, M., Biswas, G. P. and Bhar, C. (2012), Optimization of multiple objectives and topological design of data networks using genetic algorithm, *in* 'International Conference on Recent Advances in Information Technology', pp. 256–262.

Davari, S. and Fazel Zarandi, M. (2013), 'The single-allocation hierarchical hub-median problem with fuzzy flows', *Advances in Intelligent Systems and Computing* **195 AISC**, 165–181.

de Armas, J., Juan, A. A., Marquès, J. M. and Pedroso, J. P. (2017), 'Solving the deterministic and stochastic uncapacitated facility location problem: from a heuristic to a simheuristic', *Journal of the Operational Research Society* **68**(10), 1161–1176.

De Castro, L. and Timmis, J. (2002), 'Artificial immune systems: A new computational approach, september 2002'.

de Castro, L. N. and Von Zuben, F. J. (2002), 'Learning and Optimization Using the Clonal Selection Principle', *IEEE Transactions on Evolutionary Computation* **6**(3), 239–251.

De França, F. O., Coelho, G. P., Castro, P. A. D. and Von Zuben, F. J. (2010), 'Conceptual and Practical Aspects of the aiNet Family of Algorithms', *International Journal of Natural Computing Research* **1**(1), 1–35.

de Morais, V. W. C., Maravilha, A. L., de Assis, L. P. and Andrade, A. V. (2011), Abordagem Imunológica ao Problema de Roteamento de Veículos com Coleta e Entrega Simultâneas, *in* 'XLIII Simpósio Brasileiro de Pesquisa Operacional (SBPO)', Ubatuba-SP, pp. 1422–1433.

Deepalakshmi, P. and Radhakrishnan, S. (2009), QoS routing algorithm for mobile ad hoc networks using ACO, *in* 'Int. Conference on Control, Automation, Communication and Energy Conservation', pp. 1–6.

Dengiz, B., Altiparmak, F. and Smith, A. E. (1997), 'Local search genetic algorithm for optimal design of reliable networks', *IEEE Transactions on Evolutionary Computation* **1**(3), 179–188.

Dias, T. C. S., de Sousa Filho, G. F., Macambira, E. M., Lucidio dos Anjos, F. C. and Fampa, M. H. C. (2006), An efficient heuristic for the ring star problem, *in* 'International Workshop on Experimental and Efficient Algorithms', Springer, pp. 24–35.

Diaz-Baez, B., Pinto-Roa, D. P. and von Lucken, C. (2013), Robust network design under uncertain traffic an approach based on genetic algorithm, *in* 'Latin American Computing Conference', pp. 1–8.

Dominguez, O., Juan, A. A. and Faulin, J. (2014), 'A biased-randomized algorithm for the two-dimensional vehicle routing problem with and without item rotations', *International Transactions in Operational Research* **21**(3), 375–398.

Dominguez, O., Juan, A. A., Barrios, B., Faulin, J. and Agustin, A. (2016*a*), 'Using biased randomization for solving the two-dimensional loading vehicle routing problem with heterogeneous fleet', *Annals of Operations Research* **236**(2), 383–404.

Dominguez, O., Juan, A., Nuez, I. and Ouelhadj, D. (2016*b*), 'An ils-biased randomization algorithm for the two-dimensional loading hfvrp with sequential loading and items rotations', *Journal of the Operational Research Society* **67**(1), 37–53.

Donoso, Y. and Fabregat, R. (2016), *Multi-objective optimization in computer networks using metaheuristics*, Auerbach Publications.

Dorigo, M. and Birattari, M. (2011), Ant colony optimization, *in* 'Encyclopedia of machine learning', Springer, pp. 36–39.

Drezner, Z. (2015), The quadratic assignment problem, *in* 'Location Science', Springer, pp. 345–363.

Dudek, G. (2012), 'An artificial immune system for classification with local feature selection', *IEEE Transactions on Evolutionary Computation* **16**(6), 847–860.

Dumitrescu, I. and Stützle, T. (2003), Combinations of local search and exact algorithms, *in* 'Workshops on Applications of Evolutionary Computation', Springer, pp. 211–223.

Dutta, R. and Rouskas, G. N. (2002), *Design of Logical Topologies for Wavelength Routed Networks*, Springer US, pp. 79–102.

Ebery, J. (2001), 'Solving large single allocation p-hub problems with two or three hubs', *European Journal of Operational Research* **128**(2), 447–458.

Ernst, A. T. and Krishnamoorthy, M. (1998), 'Exact and heuristic algorithms for the uncapacitated multiple allocation p-hub median problem', *European Journal of Operational Research* **104**(1), 100–112.

Evans, J. (2017), *Optimization algorithms for networks and graphs*, Routledge.

Farahani, R. Z., Hekmatfar, M., Arabani, A. B. and Nikbakhsh, E. (2013), 'Hub location problems: A review of models, classification, solution techniques, and applications', *Computers & Industrial Engineering* **64**(4), 1096–1109.

Fazel Zarandi, M., Davari, S. and Haddad Sisakht, S. (2015), 'An empirical comparison of simulated annealing and iterated local search for the hierarchical single allocation hub median location problem', *Scientia Iranica* **22**(3), 1203–1217.

Feo, T. A. and Resende, M. G. (1995), 'Greedy randomized adaptive search procedures', *Journal of global optimization* **6**(2), 109–133.

Fernandez, S. A., Fantinato, D. G., Montalvao, J., Attux, R. and Silva, D. G. (2018), Immune-inspired optimization with autocorrentropy function for blind inversion of wiener systems, *in* '2018 IEEE Congress on Evolutionary Computation (CEC)', IEEE, pp. 1–7.

Fidanova, S., Marinov, P. and Paprzycki, M. (2014), *Influence of the Number of Ants on Multi-objective Ant Colony Optimization Algorithm for Wireless Sensor Network Layout*, Springer, pp. 232–239.

Finke, G., Burkard, R. E. and Rendl, F. (1987), 'Quadratic assignment problems', *Annals of Discrete Mathematics* **31**(1), 61–82.

Flores, S. D., Cegla, B. B. and Cáceres, D. B. (2003), Telecommunication network design with parallel multi-objective evolutionary algorithms, *in* 'Proceedings. Latin America Conference on Towards a Latin American Agenda for Network Research', ACM, pp. 1–11.

Focacci, F., Laburthe, F. and Lodi, A. (2003), Local search and constraint programming, *in* 'Handbook of metaheuristics', Springer, pp. 369–403.

Fortz, B. and Thorup, M. (2004), 'Increasing internet capacity using local search', *Computational Optimization and Applications* **29**(1), 13–48.

Gagnaire, M., Koubaa, M. and Puech, N. (2007), 'Network dimensioning under scheduled and random lightpath demands in all-optical WDM networks', *J. on Selected Areas in Communications* **25**(9), 58–67.

Gal, T. (1997), A historical sketch on sensitivity analysis and parametric programming, *in* 'Advances in Sensitivity Analysis and Parametic Programming', Springer, pp. 1–10.

Ghaderi, A. and Rahmaniani, R. (2016), 'Meta-heuristic solution approaches for robust single allocation p-hub median problem with stochastic demands and travel times', *The International Journal of Advanced Manufacturing Technology* **82**(9-12), 1627–1647.

Ghose, S., Kumar, R., Banerjee, N. and Datta, R. (2005), 'Multihop virtual topology design in WDM optical networks for self-similar traffic', *Photonic Network Communications* **10**(2), 199–214.

Girard, A., Sanso, B. and Dadjo, L. (2001), 'A tabu search algorithm for access network design.', *Annals of Operations Research* **106**(1-4), 229–262.

Glover, F. (1986), 'Future paths for integer programming and links to artificial intelligence', *Computers & operations research* **13**(5), 533–549.

Glover, F. (1989), 'Tabu search-part i', *ORSA Journal on computing* **1**(3), 190–206.

Glover, F. (1990), 'Tabu search—part ii', *ORSA Journal on computing* **2**(1), 4–32.

Glover, F. (1997), Tabu search and adaptive memory programming advances, applications and challenges, *in* 'Interfaces in computer science and operations research', Springer, pp. 1–75.

Glover, F. W. and Kochenberger, G. A. (2006), *Handbook of metaheuristics*, Vol. 57, Springer Science & Business Media.

Goldberg, D. E. (2006), *Genetic algorithms*, Pearson Education India.

Gomes, B. N., Martins, A. X., de Camargo, R. S. and Ramirez, J. A. (2013), 'An efficient genetic algorithm for the design of hub-and-spoke networks', *IEEE Communications Letters* **17**(4), 793–796.

Gonzalez-Martin, S., Juan, A. A., Riera, D., Elizondo, M. G. and Ramos, J. J. (2018), 'A simheuristic algorithm for solving the arc routing problem with stochastic demands', *Journal of Simulation* **12**(1), 53–66.

Grasas, A., Juan, A. A. and Ramalhinho, H. (2016), 'Simils: A simulation-based extension of the iterated local search metaheuristic for stochastic combinatorial optimization', *Journal of Simulation* **10**(1), 69–77.

Grasas, A., Juan, A. A., Faulin, J., de Armas, J. and Ramalhinho, H. (2017), 'Biased randomization of heuristics using skewed probability distributions: a survey and some applications', *Computers & Industrial Engineering*.

Gray, P., Hart, W., Painton, L., Phillips, C., Trahan, M. and Wagner, J. (1997), 'A survey of global optimization methods', *Sandia National Laboratories, Albuquerque, USA http://www. cs. sandia. gov/opt/survey*.

Gunes, M., Sorges, U. and Bouazizi, I. (2002), ARA-the ant-colony based routing algorithm for MANETs, *in* 'Proceedings. International Conference on Parallel Processing Workshop', pp. 79–85.

Hakimi, S. L. (1964), 'Optimum locations of switching centers and the absolute centers and medians of a graph.', *Operations Research* **12**(3), 450–459.

Hansen, P. and Mladenović, N. (1999), An introduction to variable neighborhood search, *in* 'Meta-heuristics', Springer, pp. 433–458.

Hansen, P. and Mladenović, N. (2002), Developments of variable neighborhood search, *in* 'Essays and surveys in metaheuristics', Springer, pp. 415–439.

Hansen, P., Mladenović, N., Brimberg, J. and Pérez, J. A. M. (2010), Variable neighborhood search, *in* 'Handbook of metaheuristics', Springer, pp. 61–86.

Hassan, A. and Phillips, C. (2008), Static routing and wavelength assignment inspired by particle swarm optimization, *in* 'International Conference on Information and Communication Technologies', pp. 1–6.

Hassan, A. and Phillips, C. (2009), Chaotic particle swarm optimization for dynamic routing and wavelength assignment in all-optical wdm networks, *in* 'Int. Conf. on Signal Processing and Communication Systems.', pp. 1–7.

He, X., Jiang, H., Song, Y., Luo, Y. and Zhang, Q. (2018), 'Joint optimization of channel allocation and power control for cognitive radio networks with multiple constraints', *Wireless Networks* pp. 1–20.

Hedayatzadeh, R., Salmassi, F. A., Keshtgari, M., Akbari, R. and Ziarati, K. (2010), Termite colony optimization: A novel approach for optimizing continuous problems, *in* 'Electrical Engineering (ICEE), 2010 18th Iranian Conference on', IEEE, pp. 553–558.

Hult, E., Jiang, H. and Ralph, D. (2014), 'Exact computational approaches to a stochastic uncapacitated single allocation p-hub center problem', *Computational Optimization and Applications* **59**(1-2), 185–200.

Ilić, A., Urošević, D., Brimberg, J. and Mladenović, N. (2010), 'A general variable neighborhood search for solving the uncapacitated single allocation p-hub median problem', *European Journal of Operational Research* **206**(2), 289–300.

Juan, A. A., Faulin, J., Ferrer, A., Lourenço, H. R. and Barrios, B. (2013), 'Mirha: multi-start biased randomization of heuristics with adaptive local search for solving non-smooth routing problems', *Top* **21**(1), 109–132.

Juan, A. A., Faulin, J., Grasman, S. E., Rabe, M. and Figueira, G. (2015*a*), 'A review of simheuristics: Extending metaheuristics to deal with stochastic combinatorial optimization problems', *Operations Research Perspectives* **2**, 62–72.

Juan, A. A., Faulin, J., Grasman, S. E., Rabe, M. and Figueira, G. (2015*b*), 'A review of simheuristics: Extending metaheuristics to deal with stochastic combinatorial optimization problems', *Operations Research Perspectives* **2**(1), 62–72.

Juan, A. A., Faulin, J., Grasman, S. E., Riera, D., Marull, J. and Mendez, C. (2011), 'Using safety stocks and simulation to solve the vehicle routing problem with stochastic demands', *Transportation Research Part C: Emerging Technologies* **19**(5), 751–765.

Juan, A. A., Grasman, S. E., Caceres-Cruz, J. and Bektaş, T. (2014*a*), 'A simheuristic algorithm for the single-period stochastic inventory-routing problem with stock-outs', *Simulation Modelling Practice and Theory* **46**, 40–52.

Juan, A. A., Lourenço, H. R., Mateo, M., Luo, R. and Castella, Q. (2014*b*), 'Using iterated local search for solving the flow-shop problem: Parallelization, parametrization, and randomization issues', *International Transactions in Operational Research* **21**(1), 103–126.

Juan, A. A., Pascual, I., Guimarans, D. and Barrios, B. (2015*c*), 'Combining biased randomization with iterated local search for solving the multidepot vehicle routing problem', *International Transactions in Operational Research* **22**(4), 647–667.

Kara, B. and Tansel, B. (2001), 'The latest arrival hub location problem', *Management Science* **47**(10), 1408–1420.

Karaboga, D. and Basturk, B. (2007), 'A powerful and efficient algorithm for numerical function optimization: artificial bee colony (abc) algorithm', *Journal of global optimization* **39**(3), 459–471.

Kennedy, J. and Shi, Y. (2001), *Swarm Intelligence*, Elsevier Science & Technology.

Kim, S. (2014), 'Adaptive MANET multipath routing algorithm based on the simulated annealing approach', *The Scientific World Journal*.

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

Klincewicz, J. G. (1998), 'Hub location in backbone/tributary network design: a review', *Location Science* **6**(1–4), 307 – 335.

Konak, A. and Kulturel-Konak, S. (2011), 'Reliable server assignment in networks using nature inspired metaheuristics', *IEEE Transactions on Reliability* **60**(2), 381–393.

Konak, A. and Smith, A. E. (1999), A hybrid genetic algorithm approach for backbone design of communication networks, *in* 'Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406)', Vol. 3, p. 1823 Vol. 3.

Konak, A., Kulturel-Konak, S. and Snyder, L. V. (2015), 'A game-theoretic genetic algorithm for the reliable server assignment problem under attacks', *Computers & Industrial Engineering* **85**, 73–85.

Kratica, J. (2013), 'An electromagnetism-like metaheuristic for the uncapacitated multiple allocation p-hub median problem', *Computers & Industrial Engineering* **66**(4), 1015–1024.

Kulturel-Konak, S. and Konak, A. (2010), Simulation optimization embedded particle swarm optimization for reliable server assignment, *in* 'Proceedings. Winter Simulation Conference', pp. 2897–2906.

Kusetogullari, H., Leeson, M. S., Ren, W. and Hines, E. L. (2011), K- shortest path network problem solution with a hybrid genetic algorithm: Particle swarm optimization algorithm, *in* '13th International Conference on Transparent Optical Networks', pp. 1–4.

Kutucu, H., Gursoy, A., Kurt, M. and Nuriyev, U. G. (2016), The band collocation problem: a library of problems and a metaheuristic approach., *in* 'DOOR (Supplement)', pp. 464–476.

Lange, S., Gebert, S., Zinner, T., Tran-Gia, P., Hock, D., Jarschel, M. and Hoffmann, M. (2015), 'Heuristic Approaches to the Controller Placement Problem in Large Scale SDN Networks', *IEEE Transactions on Network and Service Management* **12**(1), 4–17.

Langley, P., Gennari, J. and Iba, W. (1987), Hill climbing theories of learning, *in* 'Proceedings of the Fourth International Workshop on Machine Learning', pp. 312–323.

Lawler, E. L. (1985), 'The traveling salesman problem: a guided tour of combinatorial optimization', *Wiley-Interscience Series in Discrete Mathematics*.

Lee, W. C. (1995), *Mobile Cellular Telecommunications: Analog and Digital Systems*, 2nd edn, McGraw-Hill Professional.

Li, J. and Pan, Y. (2011), Network coding oriented topology design based on parallel genetic algorithm, *in* '4th International Joint Conference on Computational Sciences and Optimization'.

Li, X., Lin, S., Chen, S., Aneja, Y. P., Tian, P. and Cui, Y. (2017), 'An iterated metaheuristic for the directed network design problem with relays', *Computers & Industrial Engineering* **113**, 35–45.

Lourenço, H. R., Martin, O. C. and Stützle, T. (2010), Iterated local search: Framework and applications, *in* 'Handbook of metaheuristics', Springer, pp. 363–397.

Mahdavi, S., Shiri, M. E. and Rahnamayan, S. (2015), 'Metaheuristics in large-scale global continues optimization: A survey', *Information Sciences* **295**, 407–428.

Maleki, A., Hossain, M. M., Georges, J.-P., Rondeau, E. and Divoux, T. (2017), An sdn perspective to mitigate the energy consumption of core networks–géant2, *in* 'International SEEDS conference 2017'.

Mangai, S., Tamilarasi, A. and Venkatesh, C. (2008), Dynamic core multicast routing protocol implementation using ant colony optimization in ad hoc wireless networks, *in* 'International Conference on Computing, Communication and Networking', pp. 1–5.

Mann, P. S., Singh, S. and Kumar, A. (2016), Computational intelligence based metaheuristic for energy-efficient routing in wireless sensor networks, *in* 'Evolutionary Computation (CEC), 2016 IEEE Congress on', IEEE, pp. 4460–4467.

Marianov, V. and Serra, D. (2003), 'Location models for airline hubs behaving as m/d/c queues', *Computers & Operations Research* **30**(7), 983–1003.

Marković, G., Aćimović-Raspopović, V. and Radojičić, V. (2012), 'A heuristic algorithm for lightpath scheduling in next-generation WDM optical networks', *Photonic Network Communications* **23**(3), 272–284.

Martí, R., Corberán, Á. and Peiró, J. (2015), 'Scatter search for an uncapacitated p-hub median problem', *Computers & Operations Research* **58**, 53–66.

Martins, S. L. and Ribeiro, C. C. (2006), Metaheuristics and applications to optimization problems in telecommunications, *in* 'Handbook of optimization in telecommunications', Springer, pp. 103–128.

Masri, H., Krichen, S. and Guitouni, A. (2011), An ant colony optimization metaheuristic for solving bi-objective multi-sources multicommodity communication flow problem, *in* '4th Joint IFIP Wireless and Mobile Networking Conference', pp. 1–8.

Mazza, D., Pagès-Bernaus, A., Tarchi, D., Juan, A. A. and Corazza, G. E. (2016), 'Supporting mobile cloud computing in smart cities via randomized algorithms', *IEEE Systems Journal*.

Medhi, D. and Ramasamy, K. (2017), *Network routing: algorithms, protocols, and architectures*, Morgan Kaufmann.

Meier, J. and Clausen, U. (2015), Solving classical and new single allocation hub location problems on euclidean data, Technical report, Tech. rep. Tech. rep., Optimization Online, http://www. optimization-online. org/DB_HTML/2015/03/4816. html.

Meier, J. F., Clausen, U., Rostami, B. and Buchheim, C. (2016), 'A compact linearisation of euclidean single allocation hub location problems', *Electronic Notes in Discrete Mathematics* **52**, 37–44.

Meraklı, M. and Yaman, H. (2017), 'A capacitated hub location problem under hose demand uncertainty', *Computers & Operations Research* **88**, 58–70.

Messaoud, L. A., Merazka, F. and Massicotte, D. (2017), A novel metaheuristic based interference alignment for k-user interference channel: A comparative study, *in* 'Detection Systems Architectures and Technologies (DAT), Seminar on', IEEE, pp. 1–6.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. and Teller, E. (1953), 'Equation of state calculations by fast computing machines', *The journal of chemical physics* **21**(6), 1087–1092.

Michalak, K. and Knowles, J. D. (2016), Simheuristics for the multiobjective nondeterministic firefighter problem in a time-constrained setting, *in* 'European Conference on the Applications of Evolutionary Computation', Springer, pp. 248–265.

Mirsaleh, M. R. and Meybodi, M. R. (2018), 'Assignment of cells to switches in cellular mobile network: a learning automata-based memetic algorithm', *Applied Intelligence* pp. 1–17.

Mishra, M. and Saxena, P. (2012), 'Survey of channel allocation algorithms research for cellular systems', *International Journal of Networks and Communications* **2**(5), 75–104.

Mladenović, N. and Hansen, P. (1997), 'Variable neighborhood search', *Computers & operations research* **24**(11), 1097–1100.

Mohammadi, M., Jolai, F. and Tavakkoli-Moghaddam, R. (2013), 'Solving a new stochastic multi-mode p-hub covering location problem considering risk by a novel multi-objective algorithm', *Applied Mathematical Modelling* **37**(24), 10053–10073.

Mulvey, J. M., Vanderbei, R. J. and Zenios, S. A. (1995), 'Robust optimization of large-scale systems', *Operations research* **43**(2), 264–281.

Nancharaiah, B. and Mohan, B. C. (2014), 'The performance of a hybrid routing intelligent algorithm in a mobile ad hoc network', *Computers & Electrical Engineering* **40**(4), 1255–1264.

Oates, B. J. (2005), *Researching information systems and computing*, Sage.

Ohatkar, S. N. and Bormane, D. S. (2014), 'An optimization technique for efficient channel allocation in cellular network', *Journal of Communications Technology and Electronics* **59**(11), 1225–1233.

O'kelly, M. E. (1987), 'A quadratic integer program for the location of interacting hub facilities', *European Journal of Operational Research* **32**(3), 393–404.

Papagianni, C., Papadopoulos, K., Pampas, C., Tselikas, N. D., Kaklamani, D. T. and Venieris, I. S. (2008), Communication network design using particle swarm optimization, *in* 'International Multiconference on Computer Science and Information Technology', pp. 915–920.

Pavani, G. S. and Waldman, H. (2006), Adaptive routing and wavelength assignment with power constraints using ant colony optimization, *in* 'International Telecommunications Symposium', pp. 637–642.

Peiró, J., Corberán, Á. and Martí, R. (2014), 'Grasp for the uncapacitated r-allocation p-hub median problem', *Computers & Operations Research* **43**, 50–60.

Pióro, M. and Medhi, D. (2004), *Routing, flow, and capacity design in communication and computer networks*, Elsevier.

Qin, Z. and Gao, Y. (2017), 'Uncapacitated p-hub location problem with fixed costs and uncertain flows', *Journal of Intelligent Manufacturing* **28**(3), 705–716.

Quintero-Araujo, C. L., Caballero-Villalobos, J. P., Juan, A. A. and Montoya-Torres, J. R. (2017), 'A biased-randomized metaheuristic for the capacitated location routing problem', *International Transactions in Operational Research* **24**(5), 1079–1098.

Rahimi, Y., Tavakkoli-Moghaddam, R., Mohammadi, M. and Sadeghi, M. (2016), 'Multi-objective hub network design under uncertainty considering congestion: An m/m/c/k queue system', *Applied Mathematical Modelling* **40**(5-6), 4179–4198.

Raidl, G. R. (2006), 'A unified view on hybrid metaheuristics', *Hybrid metaheuristics* **4030**, 1–12.

Raidl, G. R. and Puchinger, J. (2008), Combining (integer) linear programming techniques and metaheuristics for combinatorial optimization, *in* 'Hybrid metaheuristics', Springer, pp. 31–62.

Ramaswami, R., Sivarajan, K. and Sasaki, G. (2009), *Optical networks: a practical perspective*, Morgan Kaufmann.

Reinelt, G. (1991), 'TSPLIB– a traveling salesman problem library.', *ORSA Journal on Computing* **3**(4), 376.

Resende, M. G. and Pardalos, P. M. (2008), *Handbook of optimization in telecommunications*, Springer Science & Business Media.

Romanycia, M. H. and Pelletier, F. J. (1985), 'What is a heuristic?', *Computational Intelligence* **1**(1), 47–58.

Ronellenfitsch, H. and Katifori, E. (2016), 'Global optimization, local adaptation, and the role of growth in distribution networks', *Physical review letters* **117**(13), 138301.

Rostami, B., Meier, J., Buchheim, C. and Clausen, U. (2015), The uncapacitated single allocation p-hub median problem with stepwise cost function, Technical report, Technical report, Optimization Online.

Ruiz, E., Albareda-Sambola, M., Fernández, E. and Resende, M. G. (2015), 'A biased random-key genetic algorithm for the capacitated minimum spanning tree problem', *Computers & Operations Research* **57**, 95–108.

Sadeghi, M., Jolai, F., Tavakkoli-Moghaddam, R. and Rahimi, Y. (2015), 'A new stochastic approach for a reliable p-hub covering location problem', *Computers & Industrial Engineering* **90**, 371–380.

Sahasrabuddhe, L. H. and Mukherjee, B. (1999), 'Light trees: optical multicasting for improved performance in wavelength routed networks', *IEEE Communications Magazine* **37**(2), 67–73.

Santos, L. F., Milagres, R., Albuquerque, C. V., Martins, S. and Plastino, A. (2006), NGL01-4: A hybrid GRASP with data mining for efficient server replication for reliable multicast, *in* 'IEEE Globecom', pp. 1–6.

Sapolsky, H. M., Crane, R. J., Neuman, W. R. and Noam, E. M. (2018), *The Telecommunications Revolution: Past, Present and Future*, Routledge.

Shakya, S., Poon, K. and Ouali, A. (2018), A ga based network optimization tool for passive in-building distributed antenna systems, *in* 'Proceedings of the Genetic and Evolutionary Computation Conference', ACM, pp. 1371–1378.

Silva, D. G., Montalvão, J., Attux, R. and Coradine, L. C. (2015), 'An immune-inspired, information-theoretic framework for blind inversion of wiener systems', *Signal Processing* **113**, 18–31.

Sim, K. M. and Sun, W. H. (2002), Multiple ant-colony optimization for network routing, *in* '1st International Symposium on Cyber Worlds, Proceedings.', pp. 277–281.

Sim, T., Lowe, T. J. and Thomas, B. W. (2009), 'The stochastic p-hub center problem with service-level constraints', *Computers & Operations Research* **36**(12), 3166–3177.

Siregar, J. H., Zhang, Y. and Takagi, H. (2005), 'Optimal multicast routing using genetic algorithm for WDM optical networks', *IEICE transactions on communications* **88**(1), 219–226.

Skorin-Kapov, D., Skorin-Kapov, J. and O'Kelly, M. (1996), 'Tight linear programming relaxations of uncapacitated p-hub median problems', *European Journal of Operational Research* **94**(3), 582–593.

Soo-Hyun, K., Chang, K.-N. and Kim, S. (2000), 'A channel allocation for cellular mobile radio systems using simulated annealing', *Telecommunication Systems* **14**(1-4), 95–106.

Soua, R. and Minet, P. (2015), 'Multichannel assignment protocols in wireless sensor networks: A comprehensive survey', *Pervasive and Mobile Computing* **16**, 2–21.

Suliman, S. I., Kendall, G. and Musirin, I. (2016), Implementing immune-based algorithm for bandwidth spectrum allocation in wireless communication, *in* 'Digital Information and Communication Technology and its Applications (DICTAP), 2016 Sixth International Conference on', IEEE, pp. 90–96.

Sun, J. U. (2012), 'A combined meta-heuristic approach to optimize a hub and spoke network design problem', *IASTED International Conference on Modelling and Simulation* pp. 1–10.

Sun, X., Dai, W., Zhang, Y. and Wandelt, S. (2017), 'Finding-hub median locations: An empirical study on problems and solution techniques', *Journal of Advanced Transportation*.

Talbi, E.-G. (2002), 'A taxonomy of hybrid metaheuristics', *Journal of heuristics* **8**(5), 541–564.

Talbi, E.-G. (2009), *Metaheuristics: from design to implementation*, Vol. 74, John Wiley & Sons.

Teymourian, E., Sadeghi, A. and Taghipourian, F. (2011), A dynamic virtual hub location problem in airline networks - formulation and metaheuristic solution approaches, pp. 1061–1068.

Tragos, E. Z., Zeadally, S., Fragkiadakis, A. G. and Siris, V. A. (2013), 'Spectrum assignment in cognitive radio networks: A comprehensive survey', *IEEE Communications Surveys & Tutorials* **15**(3), 1108–1135.

Vaezpour, E. and Dehghan, M. (2014), 'A multi-objective optimization approach for joint channel assignment and multicast routing in multi-radio multi-channel wireless mesh networks', *Wireless Personal Communications* **77**(2), 1055–1076.

Wei, W., Qin, Y. and Cai, Z. (2018), 'A multi-objective multicast routing optimization based on differential evolution in manet', *International Journal of Intelligent Computing and Cybernetics* **11**(1), 121–140.

Wille, E. C. G., Mellia, M., Leonardi, E. and Ajmone Marsan, M. (2005), *Topological Design of Survivable IP Networks Using Metaheuristic Approaches*, Springer Berlin Heidelberg, pp. 191–206.

Wolpert, D. H. and Macready, W. G. (1997), 'No free lunch theorems for optimization', *IEEE transactions on evolutionary computation* **1**(1), 67–82.

Wu, X., Ye, T., Guo, Q. and Lü, Z. (2015), 'GRASP for traffic grooming and routing with simple path constraints in WDM mesh networks', *Computer Networks* **86**, 27–39.

Yang, T.-H. (2009), 'Stochastic air freight hub location and flight routes planning', *Applied Mathematical Modelling* **33**(12), 4424–4430.

Yang, X.-S. (2014), *Nature-inspired optimization algorithms*, Elsevier.

Yang, X.-S., Cui, Z., Xiao, R., Gandomi, A. H. and Karamanoglu, M. (2013), *Swarm intelligence and bio-inspired computation: theory and applications*, Elsevier.

Yin, R., Zhong, C., Yu, G., Zhang, Z., Wong, K. K. and Chen, X. (2016), 'Joint spectrum and power allocation for d2d communications underlaying cellular networks', *IEEE Transactions on Vehicular Technology* **65**(4), 2182–2195.

Zhai, H., Liu, Y.-K. and Yang, K. (2016), 'Modeling two-stage uhl problem with uncertain demands', *Applied Mathematical Modelling* **40**(4), 3029–3048.