MATHEMATICAL DECOMPOSITION TECHNIQUES FOR RESOURCE ALLOCATION IN OPTICAL AND 5G NETWORKS

Quang Anh Nguyen

A thesis
in
The Department
of
Computer Science and Software Engineering

Presented in Partial Fulfillment of the Requirements
For the Degree of Doctor of Philosophy
Concordia University
Montréal, Québec, Canada

August 2025

Concordia University
School of Graduate Studies

This is to certify that the thesis prepared

By: **Mr. Quang Anh Nguyen**

Entitled: **Mathematical Decomposition Techniques for Resource Allocation in Optical and 5G Networks**

and submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy (Doctor of Philosophy (Computer Science))

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining commitee:

Dr. Claudio Contardo	Chair
Dr. Krzysztof Walkowiak	External Examiner
Dr. Denis Pankratov	Arms-Length Examiner
Dr. Chadi Assi	Examiner
Dr. Tristan Glatard	Examiner
Dr. Brigitte Jaumard	Supervisor

Approved by

Dr. Chen, Tse-Hsun (Peter), Graduate Program Director

August 5, 2025

Dr. Mourad Debbabi, Dean
Gina Cody School of Engineering and Computer Science

Abstract

Mathematical Decomposition Techniques for Resource Allocation in Optical and 5G Networks

Quang Anh Nguyen, Ph.D.
Concordia University, 2025

Telecommunication companies use software with heuristic algorithms to plan their routing. However, with network demands increasing at such a rapid pace, the effectiveness of these heuristics becomes a critical issue. Therefore, our research focuses on designing large-scale optimization models and algorithms for solving provisioning problems in 5G networks. Our works can be categorized into two main topics: provisioning problems at the physical layer and provisioning problems at the logical layer.

The first topic of the thesis focuses on the Routing and Spectrum Assignment (RSA) problem and is structured into three parts. In the first part, we propose a new decomposition exact modeling of the RSA problem, based on a link decomposition, in order to further improve the scalability of previous exact methods. Solution requires a column generation (CG) algorithm, a powerful decomposition technique, to derive proven ε-optimal solutions, with small ε values.

The second part presents a decomposition model that still aimed at maximizing throughput in the RSA problem, but subject to additional interference (also called. Optical Signal-to-Noise Ratio (OSNR)) constraints using the Gaussian Noise (GN) model. It is built upon the link-based decomposition model from the first part. The solution combines a Tabu Search (TS) to handle non-linear components within a Column Generation algorithm.

In the third part, we address the limitations observed in the second part, specifically the suboptimal solution of subproblems resulting from using TS. To overcome this, we propose a reformulation of the subproblems as Maximum Weight Independent Set (MWIS) problems to more effectively handle the non-linearities, and improve on both the scalability and the accuracy of the solutions.

The second topic addresses the challenge of ensuring protection for Service Function Chaining (SFC) requests in an Open Radio Access Network (O-RAN). We investigate two protection schemes (dedicated vs. shared) and two distinct objective functions (availability vs. latency), which both require handling non-linearities in an efficient manner in order to remain with scalable exact solution schemes.

Acknowledgments

I would like to express my deepest and most sincere gratitude to my supervisor, Dr. Brigitte Jaumard, for her exceptional guidance, encouragement, and unwavering support throughout my journey at Concordia University. Her expertise, insightful feedback, and thoughtful comments have shaped my research and taught me not only how to conduct rigorous scientific work, but also how to think critically, communicate effectively, and remain persistent in the face of challenges. I am truly grateful for her patience, openness to my ideas, and trust in my ability to explore new directions.

I would also like to sincerely thank Dr. Abdelhak Bentaleb for his valuable advice in structuring the writing of this thesis, and Mohammad Sheikh Zefreh, whose expertise helped me overcome challenges in areas beyond my initial field of knowledge. I am equally grateful to my friends Quang Huy Duong, for his constructive feedback and constant support, and Adham Mohammed, for his emotional support and insightful, knowledgeable feedback. Their collective support made this journey far less daunting and greatly enriched my work.

I am grateful to the members of my thesis committee for their constructive comments, challenging questions, and valuable suggestions, which significantly improved the quality of this work.

My appreciation extends to my colleagues and friends at the lab, whose stimulating discussions, technical assistance, and warm friendship made this research journey both productive and enjoyable.

I gratefully acknowledge the financial support provided by MITACS and Concordia University, without which this work would not have been possible.

Finally, I owe my deepest thanks to my family – to my mom, my dad, and my brother – for their constant support and belief in me. Their love and faith have been my greatest source of strength and motivation.

Contribution of Authors

This thesis consists of four manuscripts. Author contributions are as follows.

Chapter 3: Brigitte Jaumard and Quang Anh Nguyen, Link and Node Decomposition for Efficient Provisioning in Elastic Optical Networks
I was responsible for defining the mathematical models and its implementation, generating data and running experiments, writing and editing the draft of the manuscript. Brigitte Jaumard provided supervision, ideas for the mathematical models, implementation improvement, and manuscript editing.

Chapter 4: Brigitte Jaumard and Quang Anh Nguyen, Interference Aware Provisioning in Flexible Optical Networks
I was responsible for defining the mathematical models and its implementation, running experiments, writing and editing the draft of the manuscript. Brigitte Jaumard provided supervision, implementation improvement, and manuscript editing. We received help from Mohammad Sheikh Zefreh, CIENA, in understanding the Gaussian Noise model and how to build the reach table.

Chapter 5: Quang Anh Nguyen, Brigitte Jaumard, Decomposition Model for Interference-Aware RSA in Elastic Optical Network
I was responsible for designing the solution, implementation, running experiments, writing and editing the draft of the manuscript. Brigitte Jaumard provided supervision, implementation improvement, and manuscript review.

Chapter 6: Quang Anh Nguyen, Brigitte Jaumard, Availability vs. Latency and Shared vs. Dedicated Protection for O-RAN
I was responsible for defining mathematical models, providing mathematical proofs, generating data, running experiments, writing and editing the draft of the manuscript. Brigitte Jaumard provided supervision, and manuscript editing.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Internet demanding keep growing intensively, according to Cisco report [2], global internet protocol traffic was about 400 Exabytes per month in 2022, when it was 310, 250 and 200 Exabytes in 2021, 2020 and 2019. More recently, Sandvine's 2024 report [1] estimates daily traffic at 33 Exabytes–nearly 990 Exabytes per month– with 22 Exabytes over cable networks and 11 Exabytes over mobile networks. This remarkable growth in IP traffic over just five years underscores the pressing need to efficiently allocate network resources, which has become one of the most critical challenges in network planning.

Traditionally, Dense Wavelength Division Multiplexing (DWDM) has been used to transmit signals in optical networks. However, with the explosive increase in network demand, there is a need for new technologies that can effectively utilize network resources. This has led to the emergence of Elastic Optical Networks (EONs), which provide enhanced flexibility and efficiency in the allocation of optical resources.

While Elastic Optical Networks (EONs) enable improved utilization of network resources, they also introduce greater complexity in resource planning compared to traditional DWDM networks. In EONs, network provisioning involves the challenging task of identifying connection paths for each network demand and allocating the available spectrum along these paths. This problem is commonly known as Routing and Spectrum Assignment (RSA).

As the Routing and Spectrum Assignment (RSA) problem has been known to be

NP-complete [76], several studies have proposed heuristic solutions, hybrid approaches combining Integer Linear Programming (ILP) and heuristics, and machine learning methods to tackle it. Little emphasis has been placed on exact solutions for RSA, as these approaches often face scalability challenges when dealing with large networks, with many requests or frequency slots. To evaluate the effectiveness of non-exact solutions and provide a benchmark, there is a need for an exact solution, which requires addressing the scalability of the current RSA models and algorithms. To evaluate the effectiveness of non-exact solutions and provide a benchmark, there is a need for an exact solution, which requires addressing the scalability of the current RSA models and algorithms. This is the first objective of this thesis, with the concern of taking into account interference constraints, often neglected because of their non-linearity.

In parallel with the evolution of the physical layer, the transformation of mobile networks toward Open Radio Access Networks (O-RAN) introduces new challenges in the logical layer. In O-RAN, Service Function Chains (SFCs) are formed by deploying Virtualized Network Functions (VNFs) on cloud-based infrastructure. Ensuring high reliability and performance for these SFCs under varying network conditions and service-level objectives (e.g., low latency, high availability) leads to a complex provisioning problem. Therefore, the second objective of this thesis is to develop scalable optimization models for SFC provisioning in O-RAN, taking into account various protection schemes and performance metrics.

1.2 Thesis's Plan

This thesis consists eight chapters in total including Introduction (Chapter 1), Background (Chapter 2), State-of-art (Chapter 2.2), four manuscripts (Chapter 3, 4, 5, and 6) and Conclusion (Chapter 7).

Chapter 2 provides an overview of the background information relevant to the thesis, including concepts such as EON, RSA, O-RAN, and Column Generation (CG). Chapter 2.2 summarizes the existing body of knowledge surrounding the topics addressed in this thesis.

Chapter 3 introduces the fundamental mathematical models and the innovative decomposition method proposed for solving the RSA problem. The methodologies

and insights presented in this chapter serve as the cornerstone for our subsequent studies throughout this thesis. Expanding upon the foundations established in Chapter 3, Chapter 4 extends the problem statement by incorporating the consideration of physical interferences.

While the solution presented in Chapter 4 demonstrates effectiveness, it is important to acknowledge the limitations that still exist. Chapter 5 addresses one of these limitations by proposing an exact solution to the interference-aware RSA problem. By leveraging advanced mathematical formulations, this chapter introduces an approach that overcomes the challenges faced in the previous chapter, ultimately achieving optimal solutions.

Chapter 6 introduces the concepts of Open Radio Access Network (O-RAN), Service Function Chain (SFC), and Virtual Network Function (VNF). Additionally, this chapter focuses on two protection schemes for the SFC placement problem under single-server failure scenarios, accompanied by mathematical models and solution approaches.

Chapter 7 provides a comprehensive synthesis of the thesis, summarizing the key findings and contributions of each chapter. Additionally, this chapter outlines possible future research directions and highlights the significance of the research conducted in addressing critical challenges in network resource allocation.

Figure 1.1 highlights challenges and solutions related to RSA problem.

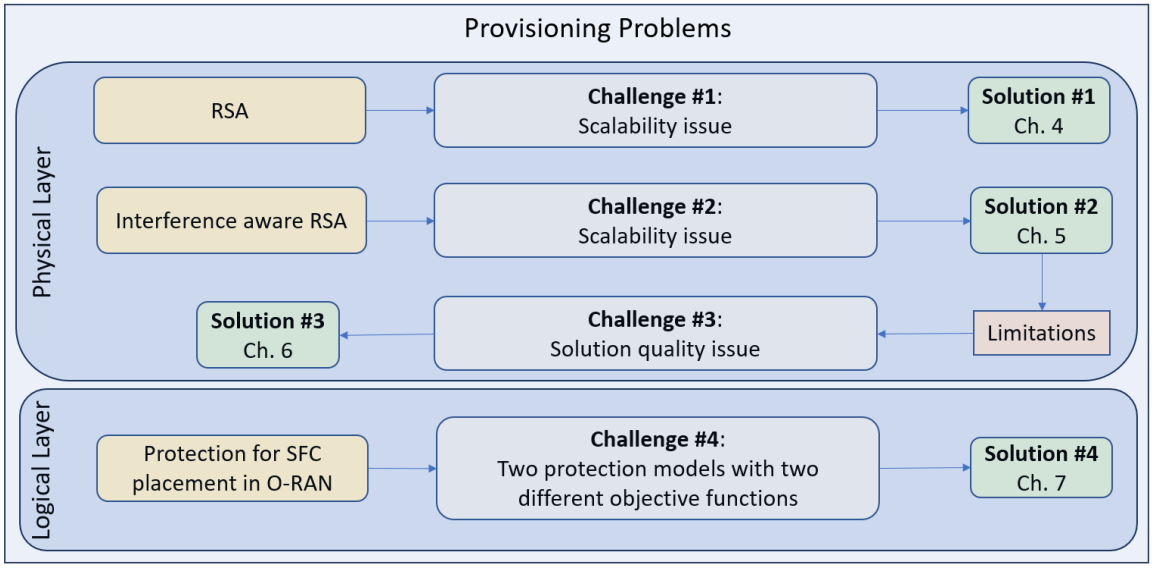


Figure 1.1: The main challenges and solutions of this thesis.

Challenge #1: We address the *Scalability issues* in existing works when seeking optimal solutions for the RSA problem in the context of large scale networks, a high number of network demands, or an extensive range of frequency slots. To tackle this challenge, Solution #1 presents solutions utilizing the Column Generation method.

Challenge #2: We encounter similar issues as in Challenge #1, but with the added complexity of considering interference in the RSA problem. To overcome this challenge, Solution #2 combines the power of Column Generation and Tabu Search methods to provide a comprehensive approach. However, this approach suffers from two drawbacks. Firstly, Tabu Search is unable to find the optimal solution for the problem, leading to sub-optimal results. Secondly, the runtime of the approach is significantly long, primarily attributed to the utilization of Tabu Search, which hinders its practicality in real-time scenarios.

Challenge #3: Our primary focus lies in addressing the *sub-optimal solution* presented in Challenge #2. In Solution #3, we aim to overcome this challenge by introducing a reformulation of the sub-problems as an Integer Linear Programming (ILP) problem, specifically as a Maximum Weight Independent Set (MWIS) problem.

Challenge #4: Our primary objective is to ensure that VNF provisioning maintains the functionality of all Service Function Chains (SFCs) in the event of a single-server failure. To achieve this, we propose two protection schemes: dedicated protection and shared protection, supported by mathematical models. Additionally, we develop heuristic approaches to address the non-linear characteristics of the problems.

Chapter 2

Background and Literature Review

2.1 Background

2.1.1 Elastic Optical Network

Elastic Optical Networks (EONs) present a new solution to the network planning problem, as compared to traditional networks that utilize Dense Wavelength Division Multiplexing (DWDM) technologies [7, 38]. EON uses Orthogonal Frequency Division Multiplexing (OFDM) technology, which allows for more flexible spectrum usage. While DWDM is limited to approximately 100 wavelengths on a 50GHz frequency grid, OFDM operates on a 12.5GHz frequency grid, with the amount of spectrum usage being flexible based on the demand capacity per channel. For a more comprehensive understanding of Elastic Optical Networks (EONs), further details can be explored in the references [45, 85]. Figure 2.1 demonstrates difference in spacing between WDM networks and EON.

Light propagating through optical fiber undergoes attenuation, which is dependent on the wavelength. The permissible signal loss ranges are categorized into different bands, namely O, E, S, C, and L-bands. Figure 2.2 illustrates these five wavelength bands in optical communication. Among these bands, the C-band is widely utilized in various optical transmission systems due to its minimal loss characteristics. The primary focus of our thesis research is addressing RSA problems specifically within the C-band, which encompasses approximately 387 frequency slots, each valued at 12.5 GHz.

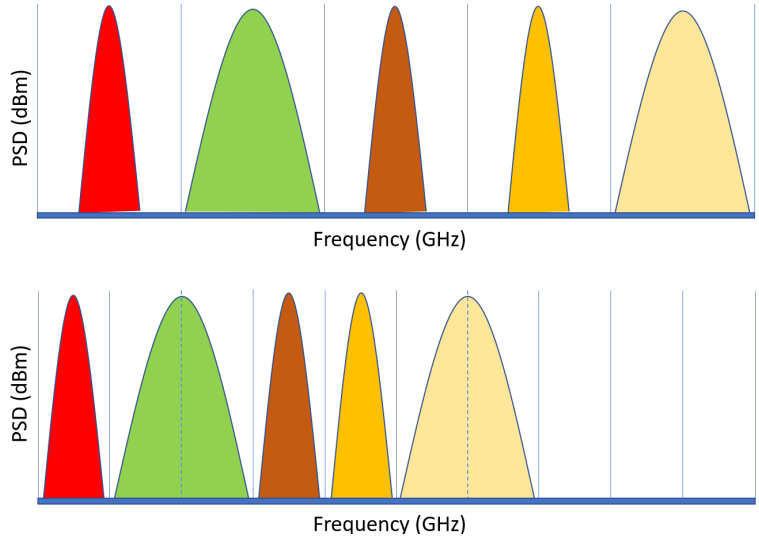


Figure 2.1: Spacing in traditional WDM network vs EON.

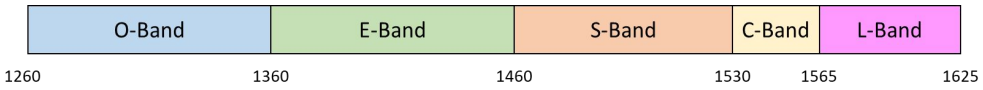


Figure 2.2: Wavelength bands. The numbers represent the wavelength values in nanometers (nm).

2.1.2 Routing and Spectrum Assignment

The problem of network provisioning in an EON is commonly referred to as the Routing and Spectrum Assignment (RSA) problem. RSA plays a crucial role in network planning as it enables the optimal utilization of available resources. In RSA, the goal is to determine the most efficient routes for transmitting data and allocate contiguous portions of the optical spectrum to meet the traffic demands. Additional information and comprehensive discussions on RSA topics can be explored in references such as [3, 15]. The objective of the RSA problem can vary, including maximizing network throughput, minimizing spectrum usage, or reducing link congestion. In our research, we specifically focus on maximizing network throughput as the primary objective.

In the present study, the RSA problem can be formulated as follows: Given a network $G(V, L)$ where V is the set of nodes, L is the set of links. With the set of demands K where each demand characterized by its source, destination, and data

rate, find a provisioning that maximize the network throughput and satisfies the following constraints:

- Continuous: for each channel, the allocated spectrum must be the same on its route
- Contiguous: for each channel, the allocated spectrum must be adjacent to each other.

2.1.3 Interference in Optical Network

In optical network, interference is a critical factor that has strong impact on the performance and reliability of data transmission. Interference refers to the unwanted disturbances that leads to degradation in signal quality and potential data loss. Optical interference can occur due to various reasons, in the context of this study, we consider two sources of interference: Amplified Spontaneous Emissions (ASE) noise and Nonlinear Interference (NLI). For a more comprehensive understanding of interference in optical fiber, a detailed exploration of this topic can be found in the reference [6, 13]. This source provides in-depth information and analysis regarding the various aspects of interference in optical fiber, offering valuable insights into its causes, effects, and mitigation techniques.

ASE noise is an unavoidable noise effect, caused by the emission of photons, or light, within the optical amplifier. This noise introduced random fluctuations in the optical signal and degrading its quality. ASE noise becomes stronger as the signal encounters multiple amplifiers along its path.

On the other hand, NLI occurs because of the nonlinear properties of optical fibers, and interaction between optical signals within the fibers leads to additional interference. In this study, to quantify the NLI, we use Gaussian Noise model in [65]. More details about this can be found in Chapter 4.

2.1.4 Open Radio Access Network

Open Radio Access Network (O-RAN) [77] is an innovative approach to telecommunication networks that promotes synergy between different vendors. By encouraging collaboration and competition, O-RAN drives innovation, leading to smarter, more

efficient, and cost-effective network solutions. The O-RAN architecture consists of four key components:

- **O-RU** (Radio Unit): Responsible for transmitting and receiving radio signals between the network and user equipment, acting as the interface to the physical layer.
- **O-DU** (Distributed Unit): Handles lower-layer data processing tasks, such as preparing data for transmission, managing connections with the O-RU, and performing scheduling and resource allocation.
- **O-CU** (Central Unit): Manages higher-layer network functions, including signaling, mobility management, and coordination across the network. It focuses on broader data flow and works closely with the RIC for optimization
- **RIC** (RAN Intelligent Controller): Provides intelligent, data-driven optimization for the network, improving performance through real-time analytics and control

With advancements in virtualization and cloud computing, these components can now be implemented as software solutions, such as Virtual Network Function (VNF), replacing traditional hardware-dependent infrastructure.

2.1.5 Column Generation

Column Generation (CG) is a powerful optimization technique and it has been used in various fields such as operations research [14, 89], transportation [31, 54] and communication networks [44, 48]. It is a method that used to solve large scale optimization problems efficiently by generating and selecting *columns* that are most relevant to the problem. For a comprehensive understanding of Column Generation and its practical application, we recommend referring to the references [10, 30]. These sources provide detailed explanations of the underlying concepts and techniques of Column Generation, along with illustrative examples that showcase its effectiveness in various contexts.

In traditional optimization, all variables are considered simultaneously, resulting in very high complexity especially for problems with a large number of variables.

On the other hand, CG addresses this issue by considering a restricted set of variables, or columns, and progressively adding new column based on their relevant and contribution to the objective function.

The core principle behind CG involves decomposing the problem into smaller sub-problems associated with a subset of variables. This decomposition allows a more efficient way to solve big problem since only a fraction of the variable set considered at any given time. By iteratively solving the sub-problems and adding new columns that contribute relevant value to the restricted master problem, CG gradually converges towards the optimal solution of the original problem. Figure 2.3 describes the work flow of CG in general.

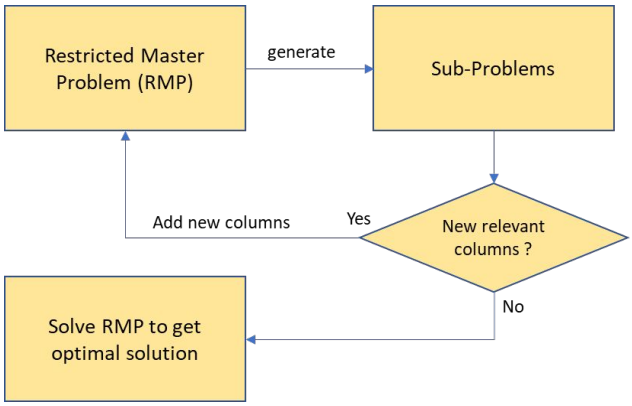


Figure 2.3: Column Generation (CG) process.

The successful application of CG is demonstrated by its real-world achievements in diverse domains such as workforce assignment, vehicle routing, resource allocation, and shift scheduling. Given the demonstrated track record of CG's success, we employed CG as the main framework for all of the projects.

2.2 Literature Review

To avoid redundancy and maintain a cohesive narrative, all literature reviews relevant to the research topics covered in Chapter 3, 4, and 5 are consolidated into this chapter. The related works are categorized into three sections, aligning with the content of the manuscripts. The first section provides an overview of studies addressing the Routing and Spectrum Assignment (RSA) problem. The second section focuses on studies tackling interference-aware RSA.

2.2.1 Routing and Spectrum Assignment

There is a wide range of solutions available in the literature to address the RSA problem. These solutions span from heuristics to exact algorithms. Several studies have explored heuristic approaches, such as the works in [37], [5], and [3].

Alaskar *et al.* [5] proposed heuristics based on priority allocation algorithms with the objective of minimizing the total amount of spectrum required to serve the demand. Their approach successfully solved data instances consisting of up to 182 requests in a network comprising 14 nodes, 20 bidirectional links, and 100 frequency slots. Additional heuristic-based solutions can be found in the comprehensive survey conducted by [3].

Due to the NP-complete nature of the RSA problem–mathematically proven by Christodoulopoulos *et al.* [19]–obtaining optimal solutions is challenging. One approach to achieve optimality is solving the Mixed Integer Linear Programming (MILP) [59] formulation of the problem using techniques like Branch and Cut (BC) [55], which combines Branch and Bound (BB) [52] with cutting planes. In this regard, Bianchetti *et al.* [12] proposed a BC approach for the RSA problem with the objective to minimize total length of the lightpaths, where various families of inequalities were added as cutting planes. The proposed approach was evaluated on different network topologies, including a European network with 43 nodes, 176 links, 384 frequency slots, and 100 demands, which was the largest considered in the experiments. The results showed that certain families of inequalities improved the performance compared to a simple branch and cut approach. However, it is noteworthy that the original function pre-solve and heuristics of the CPLEX solver still outperformed the proposed BC approach overall.

Diarrassouba *et al.* [22] adopted a similar approach to [12], but with different families of inequalities to address the RSA problem with the objective of minimizing the overall cost of routing paths. In this study, an additional constraint was introduced, where each request was associated with a maximum transmission reach, in addition to the source, destination, and number of required slots. The objective of the generated inequalities in this work was to enhance the performance of the BC approach by reducing the number of nodes in the branching tree. By incorporating these inequalities, the researchers were able to obtain optimal solutions in 137 instances, compared to 101 instances without the addition of cuts. The authors conducted experiments on various datasets, with the largest dataset consisting of 100 nodes, 136 links, 200 demands, and 280 frequency slots.

An alternative approach for obtaining optimal solutions to MILP problems is through the use of Column Generation (CG). Ruiz *et al.* [70,71] introduced decomposition models utilizing CG algorithms to solve the RSA problem with two objectives: minimizing the number of blocked demands as the primary objective and reducing the amount of unserved bit-rate as the secondary objective. Their approach successfully solved data instances with up to 64 requests in a network consisting of 21 nodes, 37 links, and a limited number of 96 frequency slots.

Klinkowski *et al.* [49] designed a CG-based solution for the RSA problem, combined with clique cuts generated by solving the maximum weighted clique problem. The objective of this approach was to minimize the total amount of unserved bit-rate. The proposed method was successfully applied to the Spain network, which consisted of 21 nodes, 35 links, 96 frequency slots, and up to 160 demands. In another study by Klinkowski *et al.* [50], with the objective was to minimize the total number of allocated frequency slots, a CG approach, combined with a simulated annealing-based heuristic for solving sub-problems, was presented. This approach was applied to the Germany network, which comprised 12 nodes, 20 links, and 60 demands. Furthermore, Klinkowski *et al.* [51] proposed an improved exact solution by combining BB with CG. At each node of the BB tree, the restricted master problem was solved using CG. This approach allowed for solving larger data instances, including the European network with 28 nodes, 41 links, and 200 demands. Goscien *et al.* [36] developed a request-based column generation method for unicast, anycast, and multicast traffic, outperforming First-Fit [46], Most-Subcarriers-First, and Longest-Path-First [19],

and achieving performance closest to the optimum. Another CG-based solution was proposed by Nguyen *et al.* in [60] to minimize spectrum usage, assuming that the network spectrum was sufficient to accommodate all requests. The model was tested on three networks: COST239 (11 nodes, 52 links), NSF (14 nodes, 42 links), and Italy (14 nodes, 58 links), with 200 frequency slots and 182 requests. The model's solution reduced spectrum usage by 16.01% in COST239, 27.34% in NSF, and 8.04% in the Italy network compared to the First-Fit heuristic.

In the aforementioned CG-based works, each sub-problem solution represented a new lightpath for each demand. Mohammed *et al.* [57] presented a novel CG approach, where the solution to each pricing problem was a sub-provisioning with allocated lightpaths having the same starting slot. The proposed solution successfully handled data instances with up to 690 requests on a USA network with 24 nodes, 85 links, and 380 frequency slots, achieving 90% throughput in less than an hour.

In recent years, there have been several notable studies on RSA topics, including works by Zhao *et al.* [101], Nassima *et al.* [58], and Paredes *et al.* [88]. Zhao *et al.* [101] presented a fuzzy logic control system for online Routing, Modulation, and Spectrum Assignment (RMSA). The effectiveness of their solution was evaluated on NSFNET (14 nodes, 22 links) and USNET (24 nodes, 43 links). When compared to traditional RMSA and RSA methods, their proposed solution achieved a blocking rate reduction of 10% and 30% respectively in NSFNET, and 10% and 25% respectively in USNET. Nassima *et al.* [58] proposed a Genetic Algorithm-based (GA) [78] approach to minimize the maximum number of allocated frequency slots on each link. The solution was evaluated on a set of randomly generated graphs, consisting of 17 instances. Among these graphs, the proposed approach outperformed the ILP model in 11 instances, while demonstrating inferior performance compared to the ILP model in the remaining cases. Another GA solution was introduced in Paredes *et al.*. This solution suggested three different strategies for GA and was compared against an ILP solution. The objective of these strategies was to minimize the maximum index of frequency slots. Among the strategies, two performed on par with the ILP model, demonstrating comparable results while significantly reducing the runtime. However, when evaluating additional metrics such as the average number of frequency slots per path and the average number of hops per path, the ILP model consistently outperformed the proposed strategies. The experiments were conducted on two network

topologies: the Abeline topology [62] with 12 nodes and 30 links, and the Nobel-EU topology [4] with 28 nodes and 82 links.

Despite notable advancements in the field of RSA, scalability remains a crucial area that requires further exploration. Additionally, to establish the groundwork for our research in Chapter 4, we introduce an innovative link-based decomposition approach for RSA in Chapter 3. This solution serves as a foundation for our subsequent analysis and contributes to addressing the scalability challenges in this field.

2.2.2 Interference aware RSA

Interference-aware RSA has received significant attention in research. While many heuristic solutions have been proposed, the focus on exact solutions has been relatively limited. One notable exact approach was presented by Yan *et al.* [96], where they employed the Gaussian Noise (GN) model to represent interference in optical fibers. They formulated the Optical Signal to Noise Ratio (OSNR) constraints as a mixed integer nonlinear problem (MINLP). However, due to the high complexity of the MINLP formulation, the model's performance was evaluated solely on tandem and ring networks with a maximum of 20 nodes. In a subsequent work, Yan *et al.* [95] introduced a linearization technique to handle the OSNR constraints by employing up to 60 linear functions. Although this approach represented an improvement, the problem complexity remained high. To address this, they developed a heuristic method to compute all binary variables and subsequently solve the problem as a Linear Programming (LP) model. The proposed approach was evaluated on the Germany network (50 nodes, 175 links) and NSF (14 nodes, 42 links) with 300 requests. Mehrabi et al. also presented in [56] an ILP solution, but the model's extensive complexity limited its testing to a smaller network, consisting of only 10 demands and 15 frequency slots. Additionally, the researchers proposed three additional heuristics, one of which performed closely to the ILP model. The heuristics were evaluated on a larger-scale Deutsche Telekom network with 14 nodes, 23 links, 500 demands, and 916 frequency slots (C+L bands).

In addition to exact solutions, several heuristics have been proposed in the literature. Hadi *et al.* introduced a 2-step algorithm in [39], which involved Routing/Traffic ordering and Power/Spectrum assignment with OSNR constraints approximated using posynomial functions. Compared to the approach presented in [96], this method

achieved an OSNR relative error of 1.09% while significantly reducing the runtime. The experiments were conducted on a COST239 network with 110 requests. Another 2-step algorithm was presented by Ives *et al.* [41]. This approach involved Launch power and Spectral assignment to increase OSNR margin, and Static Routing, Modulation, and Channel Assignment to trade OSNR margin for throughput. The algorithm was evaluated under scenarios with fixed modulation format and fixed power, as well as in the Google B4 network consisting of 12 nodes and 38 links. In the Google B4 network, the proposed algorithm demonstrated a remarkable 300% increase in throughput with 132 requests.

In recent times, there has been a significant focus on addressing RMSA problems in multi-band scenarios. Zhang *et al.* [100] introduced a GA-based heuristic to tackle the RMSA problem specifically in S+C+L band networks. The proposed solution was evaluated on the NSFNET network and demonstrated an impressive 2.5 dB improvement in the Signal to Noise Ratio (SNR) across the entire network. Yao *et al.* [97] presented a heuristic approach for addressing the Routing, Band, Modulation, and Spectrum Assignment (RBMSA) problem in C+L band networks. The proposed heuristic was compared to the First-Fit (FF) heuristic on three different network topologies: NSFNET with 14 nodes and 21 links, German network with 17 nodes and 26 links, and CMCC network with 31 nodes and 64 links. The experimental results showed that the proposed solution achieved a 9% lower blocking rate compared to FF, while utilizing less than half of the spectrum resources used by FF.

While the literature offers numerous solutions for interference-aware RSA, there is still a lack of scalable approaches that can solve the problem optimally. To address this gap, we propose a solution in Chapter 4 and an enhanced solution in Chapter 5. These proposed solutions aim to overcome the scalability limitations and provide more efficient methods to achieve optimal interference-aware RSA.

2.2.3 Learning Solution for RSA

As the field of Machine Learning (ML) continues to advance and find applications in various domains, the domain of telecommunication has also witnessed a significant increase in studies utilizing ML. In this section, we will explore several ML applications in the context of solving RSA-related topics.

Rottondi *et al.* [69] proposed a machine learning classifier that predicts whether

the bit error rate of a candidate lightpath exceeds the system threshold. The model demonstrated high accuracy, indicating its potential usefulness in RSA decision algorithms. Building upon this work, Salani *et al.* presented a hybrid approach in [73] that combines the classifier from [69] with an ILP model to solve the RSA problem. The iterative steps of this approach involve incorporating additional constraints whenever the classifier yields a negative outcome. This algorithm was able to achieve a 30% reduction in spectrum coverage compared to the traditional method of adding margin based on channel reach. However, the scalability of the approach was limited when applied to larger networks. The experiments conducted in this study involved a Japanese network consisting of 14 nodes and 22 links, with a request set generated using all-to-all connections, resulting in 182 requests and 320 available frequency slots.

Chen *et al.* [17] proposed a deep reinforcement learning (DRL) approach, named DeepRMSA, to tackle the RMSA problem with the objective of reducing the blocking rate. Evaluated on NSFNET (14 nodes) and COST239 (11 nodes) networks, each with 100 frequency slots, DeepRMSA outperformed the k-shortest path first-fit (KSP-FF) algorithm. After training on 2 million requests, it achieved a blocking rate reduction of 20.3% on NSFNET and 14.3% on COST239. More recently, Asiri *et al.* [8] extended DeepRMSA by incorporating QoT considerations into the decision-making process, further lowering the blocking rate compared to the original study. Another DRL solution named DRL-Cut was presented by Tang *et al.* in [82]. Unlike DeepRMSA, DRL-Cut employed a heuristic to compute the reward, moving away from a simple binary reward of 1 or -1. DRL-Cut achieved a significantly improved blocking rate, surpassing DeepRMSA by approximately 30.7% after being trained with approximately 1 million requests. Cheng *et al.* [18] proposed a pointer network-based approach that achieved higher OSNR and lower blocking rates than both KSP-FF and shortest-path first-fit (SP-FF) algorithms.

With the growing adoption of graph-based machine learning methods, recent studies have leveraged these techniques to enhance the learning capability of ML models. Xu *et al.* [93] proposed a DRL approach that integrates a Graph Convolutional Network (GCN) with a Recurrent Neural Network (RNN) to extract features from the network topology. Their results showed a slight improvement over DeepRMSA in terms of blocking rate, with a reduction of less than 1%. Building on this, Xiong *et al.* [92] incorporated a Graph Attention Network (GAT) into a DRL framework,

achieving a further decrease in blocking rate compared to [93]. Similarly, Quang *et al.* [68] combined GCN with multi-agent reinforcement learning, achieving a 16.62% improvement in blocking rate over DeepRMSA.

Overall, the application of machine learning to the RSA problem has evolved significantly over time. Recent ML-based solutions have shown notable progress, with some ML models outperform traditional heuristics such as the widely used k-shortest path algorithm. However, only a few studies compare their results to optimal solutions (or efficient heuristics, often called baselines in the machine learning context) or report the optimality gap (or a bound on it), leaving uncertainty about their true performance. In addition, the largest instances tested with machine learning algorithms are still smaller than the largest instances solved by optimization techniques. Moreover, it is difficult to compare their performance, as we need to discuss training and testing (to be done for each different topology with the current state of the art, with possibly some fine tuning for each topology and traffic characteristics) on the one hand, vs computational times for optimization, assuming a good initial solution is sought with a heuristic. This highlights the continued relevance and potential of mathematical models capable of solving the RSA problem optimally.

Chapter 3

Efficient Modeling of the Routing and Spectrum Allocation Problem for Flexgrid Optical Networks

This chapter corresponds to the manuscript of paper submitted for publication. As indicated in Chapter 3, we omit the literature review section in order to avoid repetitions in Chapters 3, 4 and 5.

3.1 Introduction

With the raising of networks demand, satisfying all of clients' requests becomes a hard problem and advanced networking technologies are needed to resolve such problem with more efficiency, flexibility and scalability. Elastic optical network is one of the promising solution for the high speed optical networks [16].

An elastic optical network divides its spectrum into narrow slots (e.g., 12.5GHz or 6.25GHz) and allocates these slots to lightpaths depending on its clients' requirement. Consequently, the spectrum usage is more efficient than the DWDM (Dense Wavelength Division Multiplexing) due to flexible resource allocation compare to fixed resource allocation in DWDM.

Many research efforts have been put into designing solution techniques capable of efficiently solving realistic data sets, both heuristic and exact methods. A wide range of heuristics have been proposed. Although heuristics can generally be designed

to provide a solution with reasonable computational times, or if necessary in real time, they generally share the disadvantage of having little or no information on the accuracy of their solutions, or even on their quality. Most exact methods rely on compact Integer Linear Programming solutions, which while with a polynomial number of variables and constraints, are not able to scale beyond 10 nodes with a limited number of connection requests. Few proposals have been made for large-scale optimisation models, to be solved with a decomposition algorithm. Yet, very few authors have been able to solve exactly medium size instances.

We propose two novel large-scale optimization models: the ℓ-configuration model, which groups lightpaths whose first link is ℓ, and the v-configuration model, which groups lightpaths whose first node is v. We evaluate and compare the performance of both models, highlighting their respective advantages and disadvantages. For datasets with fiber links supporting a standard transport capacity of 384 frequency slots, networks with up to 24 nodes can be solved exactly—or nearly exactly (with accuracy $\varepsilon < 10^{-2}$)—within minutes.

This chapter is organized as follows. We provide a formal statement of the Routing and Spectrum Allocation problem, together with a definition of the notations, in Section 3.2. The original new nested decomposition model is proposed in Section 3.3. Solution scheme is developed in Section 3.5. Numerical experiments are described in Section 3.6. Conclusions are drawn in the last section.

3.2 RSA Provisioning: Problem Statement

An Elastic Optical Network (EON) can be represented by a directed graph $G = (V, L)$, where V is the set of nodes and L is the set of optical fiber links. The frequency spectrum of the links is divided into a set of slices (S), also called frequency slots, indexed by s.

The network traffic (demand) is represented by a set of requests, K. Each request $k \in K$ has: *(i)* a source node $v_s \in V$ and a destination node $v_d \in V$, such that $(v_s, v_d) \in \mathcal{SD}$, where $\mathcal{SD}$ is the set of source-destination node pairs with some traffic; *(ii)* a required data rate denoted by d_k.

Provisioning a request k means: *(i)* Selecting a path from the source to the destination node of k ; *(ii)* Assigning frequency slots on every link of that path so as

to satisfy the continuity and contiguity constraints, which are next described.

Continuity constraints require that a request is assigned the same frequency slots on all its path links from source to destination.

On the other hand, **Contiguity constraints** require that the assigned frequency slots are contiguous (adjacent to each other) in the spectrum.

An illustration of a request provisioning is depicted in Figure 3.1 with a link transport capacity of 24 frequency slots and the provisioning of 5 requests. While

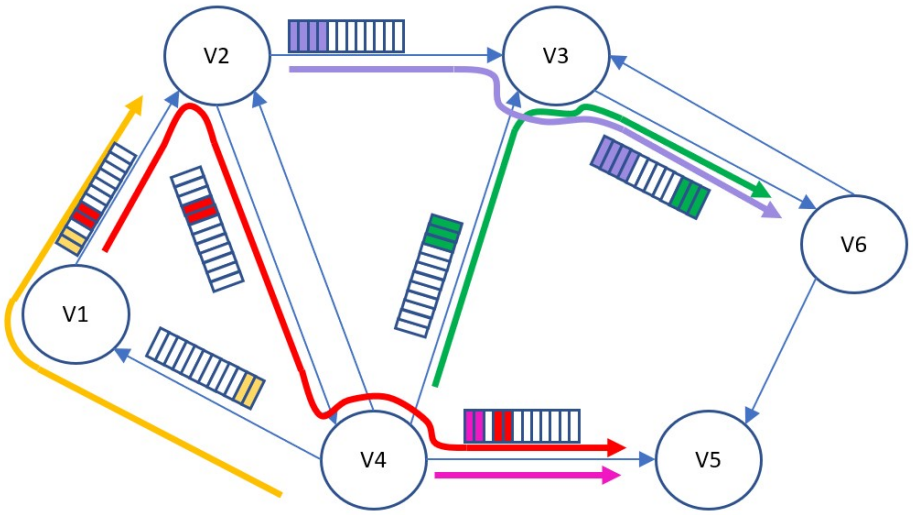


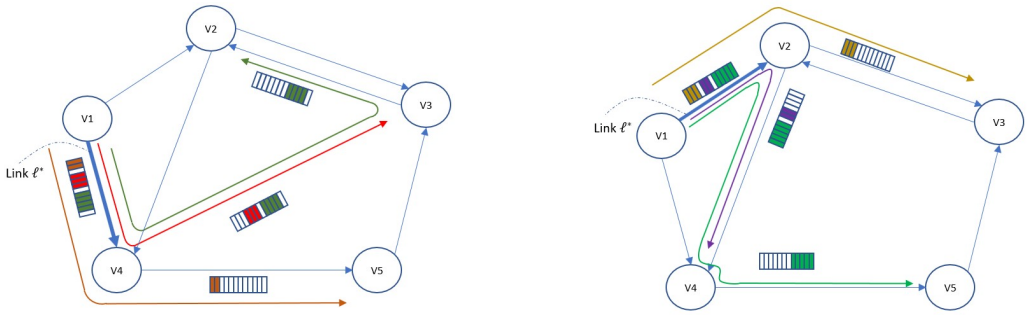
Figure 3.1: An illustrative example.

the objective function varies from one study to the next (see the papers cited in the literature review), we choose here to maximize the throughput as expressed by the weighted number of granted requests, with weights equal to the demand, i.e., the required number of frequency slots.

In the sequel, we define a lightpath by the combination of a path and a set of assigned slots satisfying the continuity and contiguity constraints.

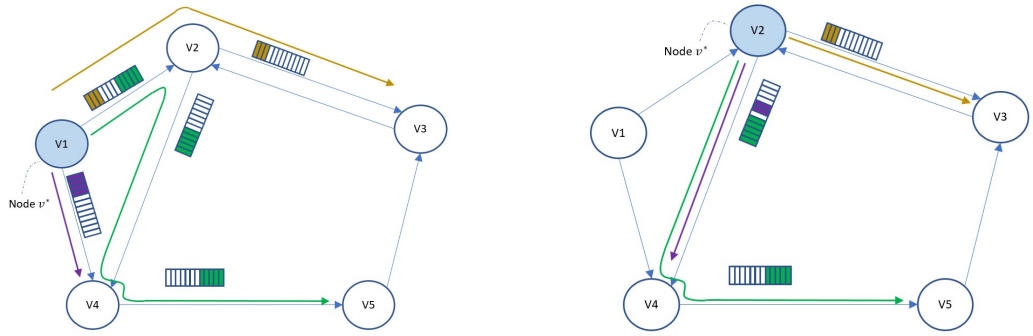
3.3 Mathematical Model

The mathematical model proposed in the previous section has an exponential number of variables, and therefore is not scalable if solved using classical ILP (Integer



(a) Configuration 1. (b) Configuration 2.

Figure 3.2: Two Link-Configuration Examples.



(a) Configuration 1. (b) Configuration 2.

Figure 3.3: Two Node-Configuration Examples.

Linear Programming) tools. Indeed, we need to use column generation techniques in order to manage a solution process that only requires an implicit enumeration of the configuration (CF) in the restricted master problem (interested readers may refer to Chvatal [20]). Each CF is generated using a link-based formulation (LConfig$_{\ell^\star}$, Figure 3.2) or node-based formulation (VConfig$_{v^\star}$, Figure 3.3) with the input is a set of pre-computed paths.

The next proposed mathematical model has two sets of variables. The first set of variables corresponds to decision variables z_γ, whose values depend on whether or not CF γ is selected. The second set of variables also corresponds to decision variables x_k, whose values depend on whether or not request k is granted.

A CF γ is a provisioning and characterized by:

- $a^\gamma_{\ell s} = 1$ if slot s is used on link ℓ in CF γ, 0 otherwise.

- $a_k^\gamma = 1$ if demand k is granted in CF γ, 0 otherwise.

The mathematical model is written as follows:

$$\max \sum_{k \in K} d_k x_k \qquad \text{(Throughput)} \tag{3.1}$$

$$\text{subject to:} \sum_{\gamma \in \Gamma_w} z_\gamma \leq 1 \qquad w \in W \tag{3.2}$$

$$\sum_{\gamma \in \Gamma} a_{\ell s}^\gamma z_\gamma \leq 1 \qquad \ell \in L, s \in S \tag{3.3}$$

$$x_k \leq \sum_{\gamma \in \Gamma} a_k^\gamma z_\gamma \qquad k \in K \tag{3.4}$$

$$z_\gamma \in \{0,1\} \qquad \gamma \in \Gamma \tag{3.5}$$

$$x_k \in \{0,1\} \qquad k \in K. \tag{3.6}$$

Constraints (3.2) enforce the selection of at most one CF for each set W, where $W = L$ in case of Link Decomposition model and $W = V$ in case of Node Decomposition model.

Constraints (3.3) make sure that if a configuration is chosen, then all the frequency slots used in that configuration are not re-used in another selected configuration.

Constraints (3.4) allow the identification of the connection requests that are granted and provisioned. Constraints (3.5) and (3.6) define the domains of the variables.

Note that, without loss of generality, we could assume $0 \leq x_k \leq 1, k \in K$. Considering the maximization objective and constraints (3.4), even if we assume $x_k \in [0,1]$, x_k can only take values 0 or 1 in the optimal solution. It has therefore the advantage of reducing the integer explicit requirements, without impacting the integer requirements of the optimal solution.

3.4 Heuristic Solution

We used first-fit strategy (Algorithm 3.1) in the heuristic solution. Each request has k potential routed paths, generated by k-shortest path algorithm (by Eppstein [27]).

Algorithm 3.1 RSA First Fit.

P: set of k-shortest path for every request k in K, ordered by $h(p) \times d_k$ where $h(p)$ is the number of hops of path p
$l(p)$: return a set of links that p go through
n_p: the number of slots required for p
k_p: request k associated with path p
t: a 2D array
for s from 1 to $|S|$ **do**:
    for $p \in P$ **do**:
        if $l(p) \cap t[s] = \emptyset$ **then**
            if r_p is not granted **then**:
                Assign slots $[s, s + n_p - 1]$ to p
                for $i \in [s, s + n_p - 1]$ **do**:
                    $t[i] \leftarrow t[i] \cup l(p)$
                end for
            end if
        end if
    end for
end for

3.5 Exact Solution

3.5.1 Column Generation

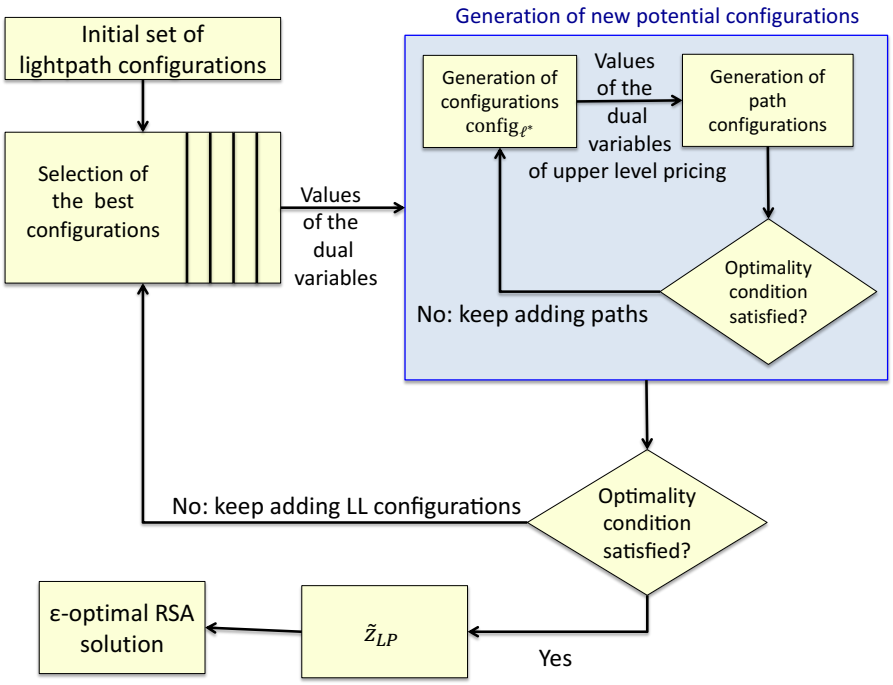


Figure 3.4: Column Generation flowchart.

We applied column generation method to solve the RSA problem. The process is described in figure 3.4. After each time we solve the relaxation of the master problem, i.e. LP problem, described in section 3.3, the dual values are used to generate the pricing problems. A pricing problem will generate a new column to be added back to the master problem, if its reduced cost, i.e. its objective function, is positive. When there are no more column generated, the master problem can be solved to achieve the ϵ-LP optimal.

We consider a Lagrangian bound, in order to compute an upper bound on the LP solution of the master problem, valid for any set of generated columns. At each iteration τ of the column generation, a Lagrangian relaxation bound LR can be calculated

as follows, using Vanderbeck [86]:

$$\begin{aligned}\text{LR}^{\tau}(x,z,u) = &\sum_{k\in K} d_k x_k + \sum_{w\in W} u_w^{(3.2)}(1-\sum_{\gamma\in\Gamma_w} z_\gamma) + \sum_{\ell\in L}\sum_{s\in S} u^{(3.3)}(1-\sum_{\gamma\in\Gamma} a_{\ell s}^{\gamma} z_\gamma)\\ &+ \sum_{k\in K} u^{(3.4)}(-x_k + \sum_{\gamma\in\Gamma} a_k^{\gamma} z_\gamma)\end{aligned} \tag{3.7}$$

subject to:

$$\sum_{\gamma\in\Gamma_w} z_\gamma \leq 1 \qquad w \in W \tag{3.8}$$

$$z_\gamma \geq 0 \qquad \gamma \in \Gamma \tag{3.9}$$

$$x_k \geq 0 \qquad k \in K, \tag{3.10}$$

where $u_\ell^{(3.2)}, u^{(3.3)}$, and $u^{(3.4)}$ are the values of the dual variables associated with constraints (3.2), (3.3), and (3.4). W is L if we choose link decomposition model, or V if we choose node decomposition model.

Let us choose link decomposition and expand the expression of $\text{LR}^{\tau}(x,z,u)$ (the node decomposition case has the same process):

$$\text{LR}^{\tau}(x,z,u) = \sum_{k\in K} d_k x_k + \underbrace{\sum_{\ell\in L} u_\ell^{(3.2)} + \sum_{\ell\in L}\sum_{s\in S} u_{\ell s}^{(3.3)}}_{ub} \tag{3.11}$$

$$\begin{aligned}&- \sum_{\ell\in L} u_\ell^{(3.2)} \sum_{\gamma\in\Gamma_\ell} z_\gamma - \sum_{\ell\in L}\sum_{s\in S} u_{\ell s}^{(3.3)} \sum_{\gamma\in\Gamma} a_{\ell s}^{\gamma} z_\gamma\\ &+ \sum_{k\in K} u_k^{(3.4)}(-x_k + \sum_{\gamma\in\Gamma} a_k^{\gamma} z_\gamma)\end{aligned} \tag{3.12}$$

$$\begin{aligned}\text{LR}^{\tau}(x,z,u) = &\sum_{k\in K} \underbrace{(d_k - u_k^{(3.4)})}_{\text{RCOST}(x_k)\leq 0} x_k + ub\\ &+ \sum_{\ell\in L}\sum_{\gamma\in\Gamma_\ell} \underbrace{\left(-u_\ell^{(3.2)} - \sum_{s\in S} a_{\ell s}^{\gamma} u_{\ell s}^{(3.3)} + \sum_{k\in K} u_k^{(3.4)} a_k^{\gamma}\right)}_{\text{RCOST}_{\gamma,\ell}^{\text{LP},\tau}} z_\gamma\\ &\leq ub + \sum_{\ell\in L}\sum_{\gamma\in\Gamma_\ell} \text{RCOST}_{\gamma,\ell}^{\text{LP},\tau}.\end{aligned} \tag{3.13}$$

Moreover, $\sum_{\ell\in L:\text{RCOST}_{\gamma,\ell}^{\text{LP},\tau}>0} \text{RCOST}_{\gamma,\ell}^{\text{LP},\tau}$ is the summation of all the reduced cost of the pricing computed at iteration τ associated with each link ℓ of the network. In other

words, we only consider non negative $\text{RCOST}^{\text{LP},\tau}_{\gamma,\ell}$, where $\text{RCOST}^{\text{LP},\tau}_{\gamma,\ell}$ is the optimal LP value.

The Lagrangian bound value is chosen by:

$$\text{LR}(x,z,u) = \min_{\tau} \text{LR}^{\tau}(x,z,u). \tag{3.14}$$

The resulting accuracy of the solution, ε, is then computed as follows:

$$\varepsilon = \frac{\min\{\text{Offered Load, LR}\} - \text{OBJ}^{\text{LB}}_{\text{ILP}}}{\text{OBJ}^{\text{LB}}_{\text{ILP}}}, \tag{3.15}$$

where the offered load is equal to $\sum_{k \in K} d_k$.

3.5.2 Nested Column Generation

In other to avoid the bias caused by using a set of pre-computed paths, such as favoring certain links and depleting resource on those links while other links still have many resources, a nested column generation was used to handle such situation. For each pricing problem, there're lower level path formulation pricing problems to generate more paths as new input for former one.

3.5.3 Integer Solution

Once the optimal solution of the LP (Linear Programming) relaxation ($\text{OBJ}^{\star}_{\text{LP}}$) has been reached, we solve exactly the last restricted master problem, i.e., the restricted master problem of the last iteration in the column generation solution process, using a branch-and-bound method, leading then to an ε-optimal ILP solution ($\text{OBJ}^{\text{LB}}_{\text{ILP}}$), where

$$\varepsilon = \frac{\text{OBJ}^{\star}_{\text{LP}} - \text{OBJ}^{\text{LB}}_{\text{ILP}}}{\text{OBJ}^{\text{LB}}_{\text{ILP}}},$$

where the optimal value of the linear relaxation ($\text{OBJ}^{\star}_{\text{LP}}$) provides an upper bound on the optimal value of the ILP ($z^{\star}_{\text{ILP}}$).

3.5.4 Pricing Problem - Link Decomposition Model

We call a configuration generated by this model $\text{LConfig}_{\ell^{\star}}$. A $\text{LConfig}_{\ell^{\star}}\gamma$ is a provisioning such that all of the provision lightpaths in γ must go through the link $\ell^{\star}$, figure 3.2 is an example of two $\text{LConfig}_{\ell^{\star}}$. To generate a $\text{LConfig}_{\ell^{\star}}$, a following pricing problem must be solved:

Maximize the reduced cost, i.e.,

$$\text{RCOST}_\gamma = -u_{\ell^\star}^{(3.2)} - \sum_{s\in S}\sum_{\ell\in L} u_{s\ell}^{(3.3)} \underbrace{\sum_{k\in K}\sum_{p\in P_k} \delta_\ell^p a_{s\ell^\star}^p}_{a_{\ell s}} + \sum_{k\in K} u_k^{(3.4)}\, a_k. \tag{3.16}$$

subject to:

$$\sum_{p\in P_k} y_p = a_k \qquad k \in K \tag{3.17}$$

$$a_s^p \le y_p \qquad p \in P_k, k \in K, \; s \in S \tag{3.18}$$

$$\sum_{k\in K}\sum_{p\in P_k} a_s^p \le 1 \qquad s \in S \tag{3.19}$$

$$\sum_{p\in P_k} \frac{1}{n_p} \sum_{s\in S} a_s^p = a_k \qquad k \in K \tag{3.20}$$

$$\sum_{p\in P_k} \sum_{s\in[1,|S|-n_p+1]} b_s^p = a_k \qquad k \in K \tag{3.21}$$

$$\sum_{i=0}^{n_p-1} a_{t+i}^p \ge n_p b_t^p \qquad t \in [1, |S| - n_p + 1], \; k \in K, p \in P_k \tag{3.22}$$

$$y_p \in \{0,1\} \qquad p \in P_k, k \in K \tag{3.23}$$

$$a_k \in \{0,1\} \qquad k \in K \tag{3.24}$$

$$a_s^p \in \{0,1\} \qquad p \in P_k, k \in K, \; s \in S. \tag{3.25}$$

$$b_s^p \in \{0,1\} \qquad p \in P_k, k \in K, \; s \in S. \tag{3.26}$$

where:

- $y_p = 1$ if path p is selected in $\text{LConfig}_{\ell^\star}$, 0 otherwise.
- $a_k = 1$ if request k is granted in the configuration under construction, 0 otherwise.
- $a_{\ell s} = 1$ if for link ℓ, slot s is occupied, 0 otherwise.
- $a_s^p = 1$ if slot s is assigned to p, 0 otherwise.

- $b_s^p = 1$ if slot s is the starting slot of p, 0 otherwise.

In addition, there is parameter $\delta_\ell^p = 1$ if path p goes through link ℓ, 0 otherwise. We denote by P_k the set of paths for routing connection request k: we do not need to pre-compute it, thanks to the nested column generation framework, where paths are online computed only once as needed.

Constraints (3.17) ensure that we select at most one path (routing) for request k if it is granted in the configuration under construction. Constraints (3.18) force variable $y_p = 1$ if provisioning of path p uses any slot s on link $\ell^\star$. Constraints (3.19) ensure that each slot is used at most once in the overall set of connection requests. Note that every selected path p must go through link $\ell^\star$, so there's no need to check for every link in the network. Constraints (3.20) make sure the total number of slots for p matches n_p. Constraints (3.21) ensure a unique starting slot for each request. Constraints (3.22) express the contiguity constraints on link $\ell^\star$.

The number of variables in the pricing problem can be further reduced by limiting the set of requests to $K_{\ell^\star}$, with $K_{\ell^\star}$ only made up of requests whose origin is equal to the node source of $\ell^\star$.

3.5.5 Pricing Problem - Node Decomposition Model

A configuration generated by this model is called a VConfig$_{v^\star}$. Similar to a LConfig$_{\ell^\star}$, a VConfig$_{v^\star}\gamma$ is a provisioning such that all of the lightpaths provisioned in γ must have the same starting node $v^\star$, firgure 3.3 is an example of VConfig$_{v^\star}$.

We obtained a VConfig$_{v^\star}$by solving the following pricing problem: Maximize the reduced cost:

$$\text{RCOST}_\gamma = -u_{v^\star}^{(3.2)} - \sum_{s\in S}\sum_{\ell\in L} u_{s\ell}^{(3.3)} \underbrace{\sum_{k\in K}\sum_{p\in P_k} \delta_\ell^p a_s^p}_{a_{\ell s}} + \sum_{k\in K} u_k^{(3.4)}\, a_k. \tag{3.27}$$

subject to:

$$a_k = \sum_{p \in P_k} y_p \qquad k \in K \tag{3.28}$$

$$a_s^p \leq y_p \qquad p \in P_k, k \in K, s \in S \tag{3.29}$$

$$\sum_{k \in K} \sum_{p \in P_k} \delta_\ell^p a_s^p \leq 1 \qquad s \in S, \ell \in L \tag{3.30}$$

$$\sum_{p \in P_k} \frac{1}{n_p} \sum_{s \in S} a_s^p = a_k \qquad k \in K \tag{3.31}$$

$$\sum_{p \in P_k} \sum_{s \in [1, |S| - n_p + 1]} b_s^p = a_k \qquad k \in K \tag{3.32}$$

$$n_p b_t^P \leq \sum_{i=0}^{n_p - 1} a_{t+i}^P \qquad t \in [1, |S| - n_p + 1]$$

$$k \in K, p \in P_k \tag{3.33}$$

$$y_p \in \{0, 1\} \qquad p \in P_k, k \in K \tag{3.34}$$

$$a_k \in \{0, 1\} \qquad k \in K \tag{3.35}$$

$$a_s^p \in \{0, 1\} \qquad p \in P_k, k \in K, \; s \in S. \tag{3.36}$$

$$b_s^p \in \{0, 1\} \qquad p \in P_k, k \in K, \; s \in S. \tag{3.37}$$

In the node decomposition, most of the constraints stay the same as in link decomposition model. However, for checking overlapped frequency slot, constraint (3.19) in link decomposition only need to check for every frequencies slots on a single link associated with the pricing problem, while constraint (3.30) in node decomposition model need to check for every frequencies slots on every links. This is the reason that makes the pricing problem of node decomposition more complex than that of link decomposition model.

Table 3.1: Comparison between pricing problem in link decomposition vs node decomposition.

	Link's pricing	Node's pricing
Number of variables	$O(\|K\| + \|P\| + \|P\|\|S\|)$	
Number of constraints	$O(\|K\| + \|P\|\|S\| + \|S\|)$	$O(\|K\| + \|P\|\|S\| + \|S\|\|L\|)$

3.5.6 Nested Pricing Problem

The lowest level pricing problem, i.e. path formulation pricing problem described in section 3.5.2, consists in computing weighted paths, and can be formulated as follows. Maximize the reduced cost:

$$\text{RCOST}_p = -\sum_{s\in S}\sum_{\ell\in L} u_{s\ell}^{(3.3)} \qquad p \in P_k, k \in K \tag{3.38}$$

which is equivalent to solving a weighted shortest path problem from v_s^k to v_d^k (source and destination node of request k, respectively) with weight $\text{WEIGHT}_\ell = \sum_{s\in S} u_{s\ell}^{(3.3)}$ for link ℓ.

3.6 Computational Results

The model and algorithm described in the previous sections was implemented on a 3.6-4.0 GHz 4-cores machine with 32 GB of RAM, with the use of CPLEX (version 12.8.0.0) for solving the (integer) linear programs.

3.6.1 Computational Comparisons on Spain Network

In a first set of experiments, we conducted experiments in order to assess the scalability of our solution process, and the accuracy of the RSA solutions that were output, in comparison with previous works. We used the same set of data instances as [42], i.e., the Spain network with 21 nodes and 35 links [70], and the same demand sets. Results are shown in Table 3.2, and include a comparison with those of [42].

In Table 3.2 we run the algorithms on many instances of the Spain network and compare it with the previous results of [42], improved in [26]. The performance of our new model and algorithm is better in terms of the quality of the solutions and of computational times.

Table 3.2: Computational comparison on Spain network.

Data Instances			$\tilde{z}_{\text{LP}}$	$\text{OBJ}_{\text{ILP}}^{\text{LB}}$	LR	ε	CPU (sec.)	from [42]		**from [26]**
Offered Load (Tbps)	$\lvert\mathcal{SD}\rvert$	$\lvert S\rvert$						$\text{OBJ}_{\text{ILP}}^{\text{LB}}$	CPU (sec.)	$\text{OBJ}_{\text{ILP}}^{\text{LB}}$ [1]
3.675	35	50	3.675	**3.675**	3.875	0	3.6	**3.17**	50	**3.675**
4.750	45	60	4.750	**4.750**	5.750	0	3.1	**4.15**	86	**4.750**
6.775	60	75	6.738	**6.725**	6.825	0.007	8.4	**5.75**	147	**6.775**
7.450	64	85	7.450	**7.450**	9.775	0	5.9	**6.00**	176	**7.450**
7.375	70	100	7.375	**7.375**	7.450	0	11.6	**6.17**	263	**7.375**
9.675	80	120	9.675	**9.675**	9.775	0	45.4	**8.15**	323	**9.675**
7.450	35	80	7.050	**7.050**	9.100	0.057	3.7	**6.70**	134	**7.450**
9.750	45	110	9.750	**9.750**	11.900	0	5.5	**8.80**	177	**9.750**
10.700	60	156	10.700	**10.700**	10.850	0	18.8	**9.45**	261	**10.700**
15.500	64	170	15.500	**15.500**	15.550	0	16.1	**12.95**	630	**15.500**
15.100	70	236	15.025	**14.950**	15.014	0.004	86.8	**13.10**	1342	**15.100**
16.850	80	256	16.700	**16.600**	16.800	0.012	61.3	**14.45**	1419	**16.850**

$|\mathcal{SD}|$ denotes the number of requests, $|S|$ designates the number of frequency slots

3.6.2 Computational Results on Larger Datasets

We consider here larger instances, both on the Spain network (Table 3.3) of the previous section, and the USA network [11] with 24 nodes and 86 links (Table 3.4). For all the experiments with the USA network, we used $|S| = 380$. While computational times are increasing, they remain reasonable for a planning problem, i.e., less than 1 hour. Note that the accuracy of the solutions is always smaller than 1%.

3.6.3 Computational of Node Decomposition model

Our goal in modeling the RSA using node decomposition is to reduce the number of pricing problems in each iteration so that the algorithm would converge faster. However what we discovered is that the amount of time to finish a run is extremely high, due to the slow convergence of each pricing problem and the increasing number of iterations of the master problem. In Table 3.5, data instance (500 requests, 200 frequency slots), the program stopped after 53 hours.

In Figure 3.5a we compared the convergence rate of the pricing problems in both link and node decomposition. The convergence rate of link decomposition is reduced

Table 3.3: Computational Results with Larger Instances on Spain network.

Data Instances			$\tilde{z}_{\text{LP}}$	$\text{OBJ}_{\text{ILP}}^{\text{LB}}$	LR	ε	CPU (sec.)
Offered Load (Tbps)	$\lvert\mathcal{SD}\rvert$	$\lvert S\rvert$					
8.075	100	300	8.075	**8.075**	8.263	0	82.5
9.625	120	300	9.600	**9.600**	9.635	0.003	102.7
11.225	140	380	11.225	**11.225**	11.255	0	147.4
13.300	160	380	13.188	**13.150**	13.225	0.006	228.1
21.925	100	380	21.925	**21.925**	21.958	0	89.9
25.600	120	380	25.413	**25.275**	25.479	0.008	182.7
29.675	140	380	29.525	**29.525**	29.525	0	405.5
33.675	160	380	32.875	**32.875**	32.925	0.002	364.8

gradually while the convergence rate of node decomposition took more time to converge. However, the total runtime of the pricing problems in each iteration are shorter in node decomposition model as shown in Figure 3.5b, because of smaller number of pricing problems in node decomposition compare to link decomposition.

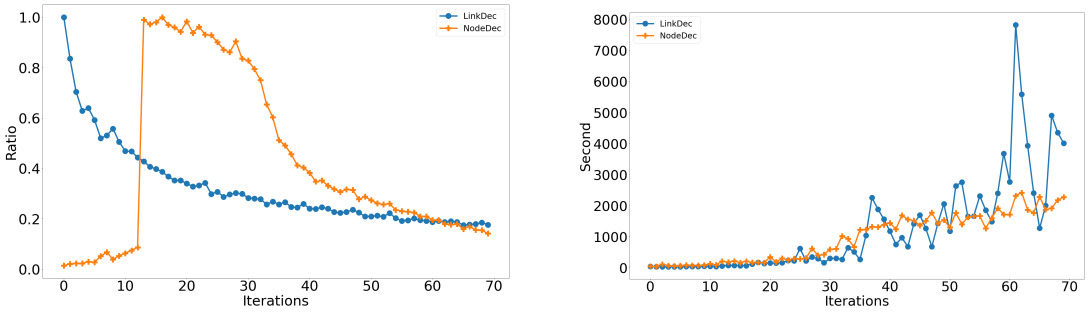
In Table 3.6 we run the two models on CONUS network (75 nodes, 99 undirected links), the datasets generated are uniform. In Table 3.7, normal directed graph (50 nodes, 88 undirected links), and 3.8, directed multigraph (50 nodes, 129 undirected links), we compared the results between node and link decomposition on the Germany network. For the datasets with the number of request less than 1000, the source and destination of each request are chosen uniformly, while for the datasets with the number of request bigger or equal 1000, the probability of choosing a pair of source and destination is derived from Tinbergen's Gravity Model [83], enhanced with logarithmic scaling to reduce disparities between rural and urban areas, following a similar approach to that of Gattuso *et al.* [33], as follows:

$$\begin{aligned} g(s,d) &= \log\left(10 + \frac{N(s)N(d)}{D(s,d)}\right) \\ p(s,d) &= \frac{g(s,d)}{\sum\limits_{(u,v)\in V^2, u\neq v} g(u,v)} \end{aligned} \tag{3.39}$$

where $N(v)$ is the population of at city v and $D(s,d)$ is the geographic distance

Table 3.4: Numerical experiments on different traffic instances on USA network.

Data Instances		$\text{OBJ}_{\text{ILP}}^{\text{LB}}$	LR	ε	CPU
Offered Load (Tbps)	$\lvert\mathcal{SD}\rvert$				(Sec.)
21.925	100	**21.925**	21.925	0	305.9
25.600	120	**25.350**	25.875	0.010	138.7
29.675	140	**29.675**	29.700	0	215.4
33.675	160	**33.550**	33.725	0.004	242.5
43.075	160	**41.650**	41.950	0.007	589.3
49.250	180	**47.150**	47.575	0.009	834.3
54.675	200	**53.425**	53.635	0.004	1458.6



(a) Convergence rate of pricing problem.

(b) CPU time.

Figure 3.5: Comparing convergence rate and runtime of link vs. node decomposition models

between s and d. Because the population between cities are too unbalanced, so certain pairs of cities were chosen many times more than others, so in Table 3.8, for each city with the population greater than 500000, we duplicate the outgoing and ingoing of those cities. The results obtained shown that the greedy heuristic's results are as good as the ILP one.

Table 3.5: Computational Results of Node Decomposition model with Larger Instances on Spain network.

Data Instances			$\tilde{z}_{\text{LP}}$	$\text{OBJ}_{\text{ILP}}^{\text{LB}}$	LR	ε
Offered Load (Tbps)	$\lvert\mathcal{SD}\rvert$	$\lvert S\rvert$				
44.3	200	50	39.309	**32.8**	44.3	0.351
44.3	200	100	44.3	**44.3**	44.3	0
55.3	250	100	55.3	**53.3**	55.3	0.038
65.4	300	100	64.036	**58.8**	65.3	0.112
65.4	300	200	65.4	**65.4**	65.4	0
110.8	500	200	109.9	**109.4**	110.8	0.013
110.8	500	400	110.8	**110.8**	110.8	0
129.7	600	200	123.9	**123.9**	129.7	0.129
129.7	600	400	129.7	**129.7**	129.7	0
151.1	700	400	151.1	**151.1**	151.1	0

Table 3.6: Comparison between Node and Link decomposition for CONUS network.

Data Instance			Node decomposition			Link decomposition			Greedy heuristic
Offer loads (Tbps)	$\lvert\mathcal{SD}\rvert$	$\lvert S\rvert$	ILP	LP	LR bound	ILP	LP	LR bound	
140.6	500	380	129.3	129.3	140.2	129.3	129.3	270.1	136.3
168.3	600	380	140.8	140.8	168.1	140.8	140.8	153.9	140.8
194.9	700	380	153.1	153.1	194.6	153.1	153.1	327.8	172.8
282.7	1000	380	161.6	161.6	275.5	161.6	161.6	327.8	189.1

3.7 Conclusion

In this chapter we proposed a new decomposition model for the RSA problem, which can be solved using a nested column generation technique. The advantage of such a link-based decomposition is that the number of pricing problems is significantly less than the number of pricing problems in a slot-based decomposition as in [42], as

Table 3.7: Comparison between Node and Link decomposition for Germany network.

Data Instance			Node decomposition			Link decomposition			Greedy heuristic
Offer loads (Tbps)	$\lvert\mathcal{SD}\rvert$	$\lvert S\rvert$	ILP	LP	LR bound	ILP	LP	LR bound	
112.4	400	400	**112.4**	112.4	112.4	**112.4**	112.4	156.2	**112.4**
126.9	450	400	**126.9**	126.9	126.9	**126.9**	126.9	182.3	**126.9**
140.9	500	400	**140.9**	140.9	140.9	**140.9**	140.9	196.1	**140.9**
169.6	600	400	**169.6**	169.6	169.6	**169.6**	169.6	241.5	**169.6**
199.2	700	400	**197.6**	197.6	198.7	**197.6**	197.6	274.0	**197.6**
279.6	1000	380	**228.9**	228.9	278.6	**228.9**	228.9	278.0	**228.9**

Table 3.8: Comparison between Node and Link decomposition for Germany multi-graph network.

Data Instance			Node decomposition			Link decomposition			Greedy heuristic
Offer loads (Tbps)	$\lvert\mathcal{SD}\rvert$	$\lvert S\rvert$	ILP	LP	LR bound	ILP	LP	LR bound	
279.6	1000	380	**230.8**	230.8	267.2	**230.8**	230.8	270.1	**230.8**
333.6	1200	380	**252.4**	252.4	309.6	**252.4**	252.4	315.3	**252.4**
360.1	1300	380	**265.4**	265.4	312.5	**265.4**	265.4	327.8	**265.4**

the number of links is less than the number of frequency slots. In addition, pricing problems are less complex to solve, and therefore can be solved faster, and in parallel. It therefore offers a promising solution scheme, with an enhanced scalability in comparison with the previous RSA decomposition schemes of the literature. We also discovered that for big datasets, the heuristic first-fit performed as good as the exact model.

Chapter 4

Interference Aware Provisioning in Flexible Optical Networks

4.1 Introduction

With the growth of Internet and network traffic demands, the efficient and cost-effective usage of bandwidth and spectrum in optical networks plays an important role in improving service provisioning. Elastic Optical Networks (EONs) define the new generation of optical networks with a higher flexibility and scalability in spectrum allocation and data rate accommodation to support different traffic types. Indeed, EONs are based on orthogonal frequency division multiplexing (OFDM), in which the network spectrum is divided into finer spectrum slots, called Frequency Slots (FSs), with the bandwidth of 12.5GHz or smaller, so that narrower spectrum slots can be allocated to lower bit rate traffic, thereby improving the spectrum resource utilization Routing is done with lightpaths, i.e., optical paths established between a given source–destination pair with a predefined data rate. Frequency slots assigned to a lightpath must satisfy two conditions: continuity and contiguity of spectrum resources. According to the continuity constraint, the same frequency slots need to be used from source to destination for a given demand request. According to the contiguity constraint, the allocated frequency slots must be pairwise contiguous in any given lightpath.

Many of the proposed RSA solutions consider the optical fiber as anideal channel and do not consider the QoS requirements in their optimization analysis. Indeed,

fiber is a non-ideal channel and optical fiber communication requires detailed attention to the channel effects such as Amplified Spontaneous Emission (ASE) noise and NonLinear Interferences (NLI). Consequently, in this study, we focus on the design of an optimization RSA model with a decomposition structure that includes the OSNR.

This chapter is organized as follows. In 4.2, we describe the RSA problem statement, introduce the GN model and reach table. We develop our mathematical model in Section 4.3 and expose the solution process in Section 4.4. Numerical results are presented in Section 4.5. Conclusions are drawn in the last section.

4.2 Problem Statement

4.2.1 GN Model for the OSNR

Studies with an OSNR mathematical expression rely on the Gaussian Noise (GN) model. Slightly different expressions of the GN model can be found, e.g., Yan *et al.* [94, 95], Poggiolini *et al.* [65]. In this study, we use the original GN model of [65] (formula (41)) under the assumption that the Power Spectral Density (PSD) of a channel per fiber span, and the parameters of the fiber are identical on every fiber span. Under these assumption, expression (41) of [65] can be written:

$$G_{\text{NLI}}(f_c) = \frac{16}{27} \sum_{n_{\text{span}}=1}^{N_{\text{span}}} \gamma^2 \, L_{\text{eff, a}}^2 \times (\Gamma^3 \, e^{-6\alpha L_{\text{span}}})^{n_{\text{span}}-1} \times (\Gamma \, e^{-2\alpha L_{\text{span}}})^{N_{\text{span}}-n_{\text{span}}-1} \times \sum_{i=1}^{N_{ch}} G_i^2 G_c \times (2 - \delta_{i,c}) \psi_{i,c,n_{\text{span}}} \tag{4.1}$$

where

$G_{\text{NLI}}(f_c)$	non linear interference (NLI) at the center frequency f_c of channel c
G_c	PSD of channel c
N_{span}	number of span of channel c
γ	non linear coefficient
$L_{\text{eff, a}}$	asymptotic effective length
L_{span}	span length, in our study, a span has a length of 80km

α power attenuation

Γ Erbium-Doped Fiber Amplifier (EDFA) gain

$\delta_{i,c}$ $= 1$ if $i = c$, 0 otherwise

$\gamma = 1.3 \cdot 10^{-3}\,\mathrm{mW}^{-1}\,\mathrm{km}^{-1}, \quad \alpha = 0.023\,\mathrm{km}^{-1},$

$L_{\text{eff, a}} = 1/(2\alpha), \quad \Gamma = e^{2\alpha L_s}$

$$\psi_{i,c,n_{\text{span}}} \approx \begin{cases} \frac{1}{4\pi(2\alpha)^{-1}|\beta_2|} \ln\left(\frac{|f_i - f_c| + B_i/2}{|f_i - f_c| - B_i/2}\right) & \text{for } i \neq c \\ \frac{\operatorname{asinh}(\frac{\pi^2}{2}\,|\beta_2|\,(2\alpha)^{-1}\,B_c^2)}{2\pi(2\alpha)^{-1}|\beta_2|} & \text{for } i = c. \end{cases}$$

Formula (4.1) then becomes:

$$G_{\text{NLI}}(f_i) = \frac{16}{27}\frac{\gamma^2\,L_{\text{eff, a}}^2\,\alpha}{\pi\,|\beta_2|} \times \left[\sum_{j=1, j\neq i}^{N_{\text{channel}}} G_j^2 G_i N_{\text{span}}^{ij} \ln\left(\frac{|f_j - f_i| + B_j/2}{|f_j - f_i| - B_j/2}\right) + N_{\text{span}}\,G_i^3\,\operatorname{asinh}\left(\frac{\pi^2|\beta_2|}{4\alpha}\,B_i^2\right)\right] \tag{4.2}$$

where N_{span}^{ij} is the number of span shared between channel i and j. Next is the ASE noise:

$$G_{\text{ASE}}(f_i) = N_s(e^{2\alpha L_{span}} - 1)h\,n_{sp}\,f_i \tag{4.3}$$

where $h = 6.62607004 \times 10^{-34}$ m^2 kg s^{-1} is Planck's constant, $n_{sp} = 5.01$ is spontaneous noise factor, its value is given by CIENA. The OSNR is then calculated by:

$$\text{OSNR}_i = \frac{G_i}{G_{\text{ASE}}(f_i) + G_{\text{NLI}}(f_i)} \tag{4.4}$$

From (4.2), we have:

$$G_{\text{SCI}}(f_i) = \frac{16}{27}\frac{\gamma^2\,L_{\text{eff, a}}^2\,\alpha}{\pi\,|\beta_2|} \times N_{\text{span}}\,G_i^3\,\operatorname{asinh}\left(\frac{\pi^2|\beta_2|}{4\alpha}\,B_i^2\right) \tag{4.5}$$

$$G_{\text{XCI}}(f_j, f_i) = \frac{16}{27}\frac{\gamma^2\,L_{\text{eff, a}}^2\,\alpha}{\pi\,|\beta_2|} \times G_j^2 G_i N_{\text{span}}^{ij} \times \ln\left(\frac{|f_j - f_i| + B_j/2}{|f_j - f_i| - B_j/2}\right) \tag{4.6}$$

$$\Rightarrow G_{\text{XCI}}(f_i) = \sum_{j=1, j\neq i}^{N_{\text{channel}}} G_{\text{XCI}}(f_j, f_i) \tag{4.7}$$

where $G_{\text{SCI}}(f_i)$ is Self-Channel Interference (SCI) of channel i, $G_{\text{XCI}}(f_j, f_i)$ is Cross-Channel Interference (XCI) caused by channel j to channel i, and $G_{\text{XCI}}(f_i)$ is total cross interference of channel i.

4.2.2 Reach Table

For each channel there's a certain condition in which described how should we assign bandwidth, the chosen bandwidth is within the set $B = \{37.5, 62.5, 87.5, 112.5\}$ GHz, the demanded bit rate of each channel is within the set $C = \{100, 200, 400\}$ Gbps. We used the Shannon's formula and (4.4) to compute the possible bandwidth could be assigned for a channel. Shannon's formula said:

$$C_i = B_i \log_2(1 + \text{OSNR}_{T,i}) \tag{4.8}$$

where C_i is the bit rate of the channel i, $\text{OSNR}_{T,i}$ is the OSNR threshold of channel i, and the OSNR constraint said that:

$$\text{OSNR}_i \geq \text{OSNR}_{T,i} \tag{4.9}$$

From (4.8) we have:

$$\text{OSNR}_{T,i} = 2^{C_i/B_i} - 1 \tag{4.10}$$

In industrial practice–specifically in the method proposed by CIENA–the threshold is adjusted by multiplying it by a correction factor:

$$\text{OSNR}_{T,i} = (2^{C_i/B_i} - 1) \times \frac{1}{0.85} \tag{4.11}$$

The scenario we used to calculate the reach table is fulfilled, that is, for a single optical fiber with C-band spectrum, consists of 380 frequency slots, each slot is 12.5 GHz, this spectrum will be filled with the same type of channels, that is channels with the same bandwidth and bit rate. For positioning a channel in the spectrum, we need guardbands on the left and right side of each channel, in industrial practice, the size of a guardband is 6.25 GHz, that means each channel will take 1 more frequency slot beside the actual one to use as guardbands.

An example of the aforementioned setup is shown in Figure 4.1, where b corresponds to a bandwidth of 37.5 GHz, which is equivalent to 3 FSs. The parameter g represents the guardband with a value of 6.25 GHz, while $n_p = 4$ denotes the total number of FSs allocated to one channel.

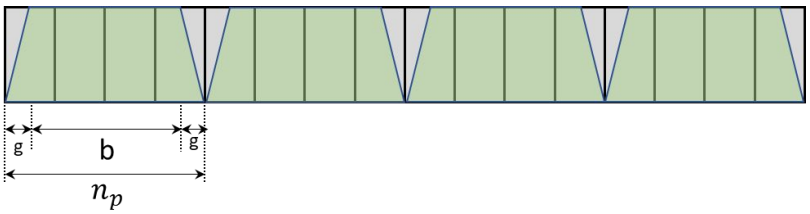


Figure 4.1: An example of a fulfilled setup.

Bit rate C_{ch}	Bandwidth B_{ch}	Bandwidth + guardbands	Number of spans N_s
100	37.5	50	57
	62.5	75	137
	87.5	100	222
	112.5	125	307
200	37.5	50	7
	62.5	75	34
	87.5	100	69
	112.5	125	107
400	62.5	75	3
	87.5	100	11
	112.5	125	24

Table 4.1: Reach Table.

Looking at the formula (4.4), (4.3) and (4.2), when the variables B_i, f_i are known, and all the launch powers of each channel is identical, $G_i = P_i/B_i$ where P_i is the launch power, so every channels have the same PSD value, then the only variable left is N_s the number of span (note that our scenario consist of a single optical fiber link, so $N_s^{i,n} = N_s$), we need to find the maximum value of N_s that still satisfies the SNR constraint. The noise of the middle channel of the spectrum is the one with the highest noise, thus the lowest OSNR value, and all of the threshold $\text{OSNR}_{T,i}$ of every channel is identical because of the same value of B_i, f_i, so to ensure the OSNR constraint of the middle channel is satisfy is the same as to ensure the OSNR constraints of every channel in the spectrum. The result of the computation is in Table 4.1.

4.3 Mathematical Model

4.3.1 OSNR Constraint

The OSNR formula is not additive, so we used its inverse. The OSNR constraint is expressed as:

$$\text{OSNR}_\pi = \frac{G_\pi}{G_{\text{ASE},\pi} + G_{\text{SCI},\pi} + G_{\text{XCI},\pi}} \geq T_\pi \tag{4.12}$$

$$\iff \frac{1}{\text{OSNR}_{T,\pi}} \geq \frac{G_{\text{ASE},\pi} + G_{\text{SCI},\pi} + G_{\text{XCI},\pi}}{G_\pi}$$

$$\iff \frac{G_{\text{XCI},\pi}}{G_\pi} \leq \underbrace{\frac{1}{T_\pi} - \frac{G_{\text{ASE},\pi} + G_{\text{SCI},\pi}}{G_\pi}}_{C_\pi} \tag{4.13}$$

The inequality (4.13) is the transformed OSNR constraint that we will use in our model.

4.3.2 Variables and parameters

We define a lightpath configuration γ as a sub-provisioning associated with link ℓ such that all the lightpaths in γ must have the same starting link as ℓ. Figure 3.2 in Chapter 3 is a visual example of a lightpath configuration associated with link ℓ^*. For the definition of variables and parameters in 4.3.3 and 4.3.4, please refer to Table 4.2.

4.3.3 Master Problem

$$\max \sum_{k \in K} d_k x_k \qquad \text{(Throughput)} \tag{4.14}$$

k	a request
s	a frequency slot
π	a lightpath
γ	a lightpath configuration
K	set of all requests
S	set of all frequency slots
Γ_k	a set of lightpaths associated with request k
z_γ	=1 if γ is selected, 0 otherwise
x_k	=1 if request k is granted, 0 otherwise
d_k	data rate of request k
a_k^γ	=1 if request k is granted in γ, 0 otherwise
$a_{\ell s}^\gamma$	=1 if slot s at link ℓ in γ is occupied, 0 otherwise
y_π^γ	=1 if π appear in γ, 0 otherwise
θ_π^γ	total cross interference (XCI) cause by all the lightpaths in γ to π
v_p	=1 if path p is selected, 0 otherwise
a_k	=1 if request k is granted, 0 otherwise
$a_{s\tilde{\ell}}^p$	=1 if path p occupy slot s in link $\tilde{\ell}$, 0 otherwise
$b_{s\tilde{\ell}}^p$	=1 if slot s is the starting slot of path p in link $\tilde{\ell}$, 0 otherwise
n_p	is the number of slot that path p require.

Table 4.2: Variables and parameters

subject to:

$$\sum_{\gamma\in\Gamma_\ell} z_\gamma \leq 1 \qquad \ell \in L \tag{4.15}$$

$$\sum_{\gamma\in\Gamma} a_{\ell s}^\gamma z_\gamma \leq 1 \qquad \ell \in L, s \in S \tag{4.16}$$

$$x_k \leq \sum_{\gamma\in\Gamma} a_k^\gamma z_\gamma \qquad k \in K \tag{4.17}$$

$$\sum_{\gamma\in\Gamma} z_\gamma(\theta_\pi^\gamma + (M - C_\pi)y_\pi^\gamma) \leq M \qquad \pi \in \Pi_k, k \in K \tag{4.18}$$

$$z_\gamma \in \{0,1\} \qquad \gamma \in \Gamma \tag{4.19}$$

$$x_k \in \{0,1\} \qquad k \in K \tag{4.20}$$

In the master problem, since configuration γ is computed, then C_π is a fixed

number. M is a constant big enough to be an upper bound of $G_{\mathrm{XCI},\pi}/G_{\pi}$, in our implementation we set $M = 1000$. Constraints (4.15) ensure that for each link, there's at most 1 configuration associated with selected. Constraints (4.16) ensure that each slot on each link occupied by at most 1 configuration, thus satisfy the continuity constraint. Constraints (4.17) check whether request k is granted. Constraints (4.18) check the SNR constraint of lightpath π.

4.3.4 Pricing Problem

Each link ℓ in the network has a pricing problem associated with and this problem is updated after every iteration. By solving the pricing problem, a new lightpath configuration is generated.

$$\begin{aligned}\max\ \overline{\mathrm{COST}}_{\gamma} = -u^{(4.15)} &- \sum_{s\in S}\sum_{\ell\in L} u_{s\ell}^{(4.16)} \underbrace{\sum_{k\in K}\sum_{p\in P_k} \delta_{\ell}^{p} a_{s\tilde{\ell}}^{p}}_{a_{\ell s}} \\ &- \sum_{k\in K}\sum_{\pi\in\Pi_k} u_{\pi}^{(4.18)}(\theta_{\pi} + (M - C_{\pi})y_{\pi}) \\ &+ \sum_{k\in K} u_{k}^{(4.17)}\, a_k \end{aligned} \tag{4.21}$$

subject to:

$$\sum_{p \in P_k} v_p = a_k \qquad k \in K \tag{4.22}$$

$$a^p_{s\tilde{\ell}} \leq v_p \qquad p \in P_k, k \in K, s \in S \tag{4.23}$$

$$\sum_{p \in P_k} \frac{1}{n_p} \sum_{s \in S} a^p_{s\tilde{\ell}} = a_k \qquad k \in K \tag{4.24}$$

$$\sum_{p \in P_k} \sum_{s \in [1, |S| - n_p + 1]} b^p_{s\tilde{\ell}} = a_k \qquad k \in K \tag{4.25}$$

$$\sum_{i=0}^{n_p - 1} a^p_{t+i,\tilde{\ell}} \geq n_p b^p_{t\tilde{\ell}} \qquad t \in [1, |S| - n_p + 1],$$

$$k \in K, p \in P_k \tag{4.26}$$

$$\sum_{k \in K} \sum_{p \in P_k} a^p_{s\tilde{\ell}} \leq 1 \qquad s \in S \tag{4.27}$$

$$v_p \in \{0, 1\} \qquad p \in P_k, k \in K \tag{4.28}$$

$$a_k \in \{0, 1\} \qquad k \in K \tag{4.29}$$

$$a^p_{s\tilde{\ell}} \in \{0, 1\} \qquad p \in P_k, k \in K, s \in S \tag{4.30}$$

$$b^p_{s\tilde{\ell}} \in \{0, 1\} \qquad p \in P_k, k \in K, s \in S \tag{4.31}$$

where constraints (4.22) ensure that we select at most one path (routing) for request k if it is granted in the γ under construction. Constraints (4.23) ensure that variable $y_p = 1$ if path p occupies any slot s on link $\ell^\star$. Constraints (4.24) ensure the total number of slots for p match with n_p. Constraints (4.25) ensure a unique starting slot for each request. Constraints (4.26) express the contiguity constraints on link $\ell^\star$. Constraints (4.27) ensure that each slot is used at most once in the overall set of connection requests.

The term θ_π in (4.21) express the interference caused by γ which is under construction to lightpath π, and is written as:

$$\begin{aligned}
\theta_\pi^\gamma &= \sum_{\substack{\pi' \in \gamma \\ \pi' \neq \pi}} \frac{G_{\mathrm{XCI},\pi}^{\pi'}}{G_\pi} \\
&= M \sum_{\substack{\pi' \in \gamma \\ \pi' \neq \pi}} N_s^{\pi,\pi'} \ln\left(\frac{|f_\pi - f_{\pi'}| + B_{\pi'}/2}{|f_\pi - f_{\pi'}| - B_{\pi'}/2}\right) \\
&= M \sum_{\substack{\pi' \in \gamma \\ \pi' \neq \pi}} N_s^{\pi,\pi'} \ln\left(1 + \frac{B_{\pi'}}{|f_\pi - f_{\pi'}| - B_{\pi'}/2}\right) \\
&= M \sum_{\substack{\pi' \in \gamma \\ \pi' \neq \pi}} N_s^{\pi,\pi'} \ln\left(1 + \frac{1}{|f_\pi - f_{\pi'}|/B_{\pi'} - 1/2}\right)
\end{aligned}$$

where $f_{\pi'} = \sum_{s \in S} b_{sl}^{p'} * (2s + n_{p'})/2$, f_π is a known value. To linearize $|f_\pi - f_{\pi'}|$:

$$\begin{aligned}
& f_\pi - f_{\pi'} = t_{\pi\pi'}^+ \\
& f_{\pi'} - f_\pi = t_{\pi\pi'}^- \\
& b t_{\pi\pi'}^+ + (1-b) t_{\pi\pi'}^- \geq 0 \\
\Rightarrow & |f_\pi - f_{\pi'}| = b t_{\pi\pi'}^+ + (1-b) t_{\pi\pi'}^- \qquad (4.32)
\end{aligned}$$

With (4.32), the pricing problem becomes a normal non linear programming problem.

We will prove that the pricing problem is a convex optimization problem.

$$\begin{aligned}
\overline{\mathrm{COST}}_\gamma &= -u^{(4.15)} - \sum_{s \in S} \sum_{\ell \in L} u_{s\ell}^{(4.16)} \sum_{k \in K} \sum_{p \in P_k} \delta_\ell^p a_{s\tilde{\ell}}^p - \sum_{k \in K} \sum_{\pi \in \Pi_k} u_\pi^{(4.18)} (\theta_\pi + (M - C_\pi) y_\pi) \\
&\quad + \sum_{k \in K} u_k^{(4.17)} a_k \\
&= -u^{(4.15)} - \sum_{s \in S} \sum_{\ell \in L} u_{s\ell}^{(4.16)} \sum_{k \in K} \sum_{p \in P_k} \delta_\ell^p a_{s\tilde{\ell}}^p - \sum_{k \in K} \sum_{\pi \in \Pi_k} u_\pi^{(4.18)} y_\pi (M - C_\pi) + \sum_{k \in K} u_k^{(4.17)} a_k \\
&\quad - \sum_{k \in K} \sum_{\pi \in \Pi_k} u_\pi^{(4.18)} \theta_\pi \qquad (4.33)
\end{aligned}$$

Pose:

$$
\begin{aligned}
h \quad &= -u^{(4.15)} - \sum_{s\in S}\sum_{\ell\in L} u_{s\ell}^{(4.16)} \sum_{k\in K}\sum_{p\in P_k} \delta_\ell^p a_{s\tilde{\ell}}^p - \sum_{k\in K}\sum_{\pi\in\Pi_k} u_\pi^{(4.18)} y_\pi (M - C_\pi) + \sum_{k\in K} u_k^{(4.17)}\, a_k \\
g \quad &= -\sum_{k\in K}\sum_{\pi\in\Pi_k} u_\pi^{(4.18)} \theta_\pi \\
\Rightarrow &\overline{\mathrm{COST}}_\gamma = h + g
\end{aligned}
$$

The term g is non-linear while h is linear, so the convexity of (4.33) is depends on g.

Consider 2 lightpaths π and π'. Pose $d_{\pi'}^{\pi} = |f_\pi - f_{\pi'}|/B_{\pi'}$. Because f_π and $f_{\pi'}$ are center frequencies of π and π', we have:

$$
\begin{aligned}
&|f_\pi - f_{\pi'}| \geq (B_\pi + B_{\pi'})/2 > B_{\pi'}/2 \\
\Rightarrow &|f_\pi - f_{\pi'}|/B_{\pi'} > 1/2 \\
\Rightarrow &d_{\pi'}^{\pi} > 1/2
\end{aligned}
$$

θ_π could be written as:

$$
\begin{aligned}
\theta_\pi &= f(\pi_1, \pi_2, ..., \pi_t) \\
&= \sum_{i=1}^{t} A_{\pi_i} \ln\left(1 + \frac{1}{d_{\pi_i}^{\pi} - 1/2}\right)
\end{aligned}
$$

with A_{π_i} is a non-negative constant, $d_{\pi_i}^{\pi} > 1/2$. Consider $f(x) = \ln(1 + 1/(x - 1/2))$ with $x > 1/2$. Second degree derivative of f is:

$$
f''(x) = \frac{32x}{(4x^2 - 1)^2}
$$

Because $x > 1/2$, $f''(x) > 0$, so $f(x)$ is a convex function. θ_π is a summation of many convex functions, so θ_π is also a convex function. Consider the last term g of (4.33), the terms $u_\pi^{(4.18)}$ are positive constants, so $-u_\pi^{(4.18)}\theta_\pi$ is a concave function. g is a summation of many concave functions, so g is also a concave function. h is a linear term, so it could be considered as both concave and convex, so $\overline{\mathrm{COST}}_\gamma = h + g$ is a concave function.

4.3.5 Lagrangian bound

To measure the effectiveness of the model, we decided to use Lagrangian bound as the upper bound of the master problem. The bound can be computed using Vanderbeck [86]. Lagrangian relaxation of the master problem at each iteration τ is written as:

$$\begin{aligned} LR_\tau = &\sum_{k\in K} d_k x_k + \sum_{\ell\in L} u^{(4.15)}(1 - \sum_{\gamma\in\Pi_\ell} z_\gamma) + \sum_{\ell\in L}\sum_{s\in S} u^{(4.16)}(1 - \sum_{\gamma\in\Gamma} a^\gamma_{\ell s} z_\gamma) \\ &+ \sum_{k\in K} u^{(4.17)}(\sum_{\gamma\in\Gamma} a^\gamma_k z_\gamma - x_k) + \sum_{k\in K}\sum_{\pi\in\Pi_k} u^{(4.18)}(M - \sum_{\gamma\in\Gamma} z_\gamma(\theta^\gamma_\pi + (M - C_\pi)y^\gamma_\pi)) \end{aligned} \tag{4.34}$$

subject to:

$$z_\gamma \in [0,1] \qquad \gamma \in \Gamma \tag{4.35}$$

$$x_k \in [0,1] \qquad k \in K \tag{4.36}$$

From (4.34) we have:

$$\begin{aligned} LR_\tau = &\sum_{k\in K} x_k \underbrace{(d_k - u^{(4.17)})}_{=RCOST(x_k)\leq 0} + \sum_{\ell\in L} u^{(4.15)} + \sum_{\ell\in L}\sum_{s\in S} u^{(4.16)} + \sum_{k\in K}\sum_{\pi\in\Pi_k} u^{(4.18)} M \\ &+ \sum_{\ell\in L}\sum_{\gamma\in\Gamma_\ell} z_\gamma(-u^{(4.15)} - \sum_{s\in S}\sum_{\ell\in L} u^{(4.16)}_{s\ell} \sum_{k\in K}\sum_{p\in P_k} \delta^p_\ell a^p_{s\tilde{\ell}} \\ &- \sum_{k\in K}\sum_{\pi\in\Pi_k} u^{(4.18)}_\pi(\theta_\pi + (M - C_\pi)y_\pi) + \sum_{k\in K} u^{(4.17)}_k\, a_k) \end{aligned}$$

With

$$u_\tau b = \sum_{\ell\in L} u^{(4.15)} + \sum_{\ell\in L}\sum_{s\in S} u^{(4.16)} + \sum_{k\in K}\sum_{\pi\in\Pi_k} u^{(4.18)} M$$

where u_τ and b is the dual vector at iteration τ and right hand side constant vector of the constraints of the master problem, we have:

$$\begin{aligned} LR_\tau &= \sum_{k\in K} x_k \underbrace{(d_k - u^{(4.17)})}_{=RCOST(x_k)\leq 0} + ub + \sum_{\ell\in L}\sum_{\gamma\in\Gamma_\ell} z_\gamma \overline{\text{COST}}_\gamma \\ &\leq u_\tau b + \sum_{\gamma\in\Gamma_\ell} \overline{\text{COST}}_\gamma \end{aligned} \tag{4.37}$$

From (4.37), Lagrangian bound (LRB) at iteration τ is computed by:

$$LRB_\tau = ub + \sum_{\gamma \in \Gamma_\ell} \overline{\text{COST}}_\gamma$$

where $\overline{\text{COST}}_\gamma$ is the non-negative solution of each solved pricing problem. Then the final LRB is the minimum between the values of LRB_τ:

$$LRB = \min_\tau \ LRB_\tau \tag{4.38}$$

4.4 Solution Process: Column Generation & Tabu Search

4.4.1 Column Generation

The process of column generation (CG) is described in Fig. 3.4 Chapter 3.

In each iteration, the master problem is solved as a Linear Program (LP) problem, then its dual values is feed forward the pricing models associated with each link of the network. If the pricing pricing problem has a feasible solution with a positive reduced cost then the ILP result is returned as a new column (lightpath configuration) for the master, if not then a lower level pricing problem, which purpose is to generate more path for the pricing, will generate more column (routing path) for the pricing. This process is repeated until no more lightpath configuration is generated, then we solve the master problem as an ILP, the result obtained is ϵ-optimal solution.

However, because of the θ_π in (4.21), the number of logarithm terms is very big, and we have no access to any optimisation solver capable of solving the problem with efficient time, so we resort to resolve the problem with a meta heuristic.

4.4.2 Tabu Search

We designed the Tabu Search (TS) [34, 35] process as follows:
Call $\textsc{rcost}(c)$ the reduced cost of solution (configuration) c. The goal is to find c^* such that $\textsc{rcost}(c^*) \geq 0$.
Call $\textsc{tabu}(\pi)$ the tabu status of lightpath π, $\textsc{tabu}(\pi) \geq 0$. If $\textsc{tabu}(\pi) > 0$ then the status of π is Tabu (we cannot consider changing π), normal otherwise.

Call $f(\pi, \pi')$ the cross interference caused by π that affect π'.
Call $N(c)$ the neighborhood of c, each neighbor (configuration) created by:

- Shifting 1 lightpath in c up or down by 1 frequency slot, if there's conflict with other lightpaths, then shift them in the same direction, if there's a lightpath that get out of the spectrum, then delete that lightpath
- With a random selection, switch 1 of the lightpath $\pi_{sd} = (p, s)$ (path, starting slot) in c with other lightpath $\pi'_{sd} = (p', s)$ in the pool
- Select randomly two lightpaths $\pi = (p, s, \#slots(\pi))$ and $\pi' = (p', s', \#slots(\pi'))$ in c. Assume wlog that $s \geq s'$. Exchange positions of π, π' (wrp to their frequency slots), if there're conflicts in the process, try to shift the lightpaths in between.
- With a random selection, add a path into biggest available chain of frequency slots
- With a random selection, remove a random lightpath
- Choose only 1 out of 5 strategies above to generate 1 neighbor. The created configuration will be abandoned if a lightpath π with $\text{TABU}(\pi) > 0$ is moved in the process.

1: Find an initial solution c_0 (by greedy or by solving the ILP without the log terms)
2: $c_{best} \leftarrow c_0$
3: $c_{current} \leftarrow c_0$
4: $c_t \leftarrow \text{argmax}\, r(c), c \in N(c_{current})$
5: $\forall \pi \in c_t$, if $\text{TABU}(\pi) > 0$ then $\text{TABU}(\pi) \leftarrow \text{TABU}(\pi) - 1$
6: If $\text{RCOST}(c_t) < \text{RCOST}(c_{current})$ (the solution didn't improve):

- choose the lightpath π' in c_t such that the value $\sum_{\pi \in \Pi} f(\pi, \pi')$ is the smallest and $\text{TABU}(\pi') == 0$
- $\text{TABU}(\pi') \leftarrow 20$

7: $c_{current} \leftarrow c_t$
8: If $r(c_{current}) > r(c_{best})$ then $c_{best} \leftarrow c_{current}$
9: If $r(c_{best}) > 0$ then stop, c_{best} is the solution. If the number of iteration exceed 100 then stop, there is no solution, else repeat step 4.

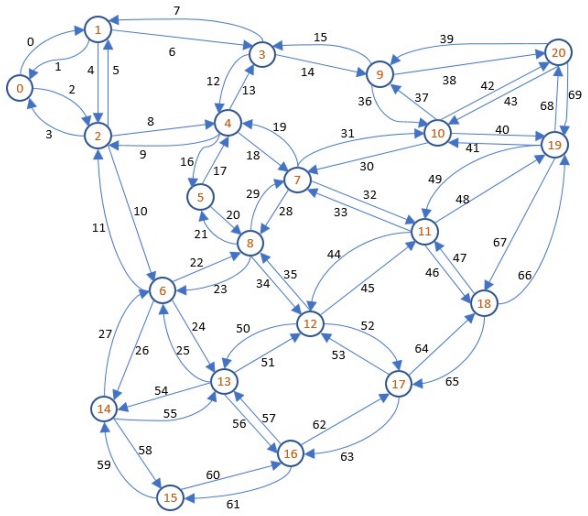


Figure 4.2: Spain network.

4.5 Numerical Result

4.5.1 Data Sets

We experimented using different set of requests and different amount of frequency slots on the Spain network topology. The data rate's value is generated within the set {100, 200, 400} Gbps with the respected distribution 40%, 30%, 30%.

4.5.2 Performance of the Model

In Table 4.3 is the numerical result we ran through several data set, we compared the throughput, spectrum usage (SU) between OSNR and non-OSNR context.

We compare the result of our model against First-Fit (Algorithm 4.1) and Best-Fit (Algorithm 4.2), we also include the result when the OSNR constraints are ignored. The result obtained shows that our model outperforms both heuristics First-Fit and Best-Fit, especially when in high number of FSs and high number of requests. When running on low number of slots (smaller than 100), the SU dropped between non-OSNR and OSNR is around 5% and raise to more than 10% when the number of slots is higher. Where the provisioning is most packed (200 slots, 600 requests) the

Nb slots	Nb re--quest	Total load	No SNR				With SNR					
			z_{ILP}	z_{LP}	LRB	SU	First fit	Best fit	z_{ILP}	z_{LP}	SU	LRB
50	200	44.3	**33.0**	33.0	49.005	**0.73**	**19.8**	**16.5**	**23.5**	28.9	**0.60**	28.9
100	200	44.3	**44.3**	44.3	46.592	**0.54**	**22.2**	**22.3**	**33.2**	44.0	**0.52**	44.0
100	250	55.3	**52.6**	52.6	70.500	**0.64**	**18.9**	**23.2**	**36.7**	51.3	**0.52**	51.3
100	300	65.4	**58.8**	58.8	79.237	**0.70**	**22.0**	**21.7**	**32.9**	52.1	**0.45**	52.1
200	300	65.4	**65.4**	65.4	71.798	**0.40**	**22.2**	**34.3**	**42.1**	65.4	**0.33**	65.4
200	500	110.8	**109.4**	109.4	146.401	**0.66**	**25.8**	**31.7**	**55.3**	103.7	**0.38**	103.7
200	600	129.7	**123.9**	123.9	182.250	**0.75**	**41.4**	**29.4**	**57.6**	113.3	**0.37**	113.3
400	500	110.8	**110.8**	110.8	119.934	**0.34**	**26.4**	**48.4**	**61.5**	110.8	**0.23**	110.8
400	600	129.7	**129.7**	129.7	144.513	**0.40**	**41.8**	**50.2**	**68.8**	129.7	**0.25**	129.7

Table 4.3: Numerical result.

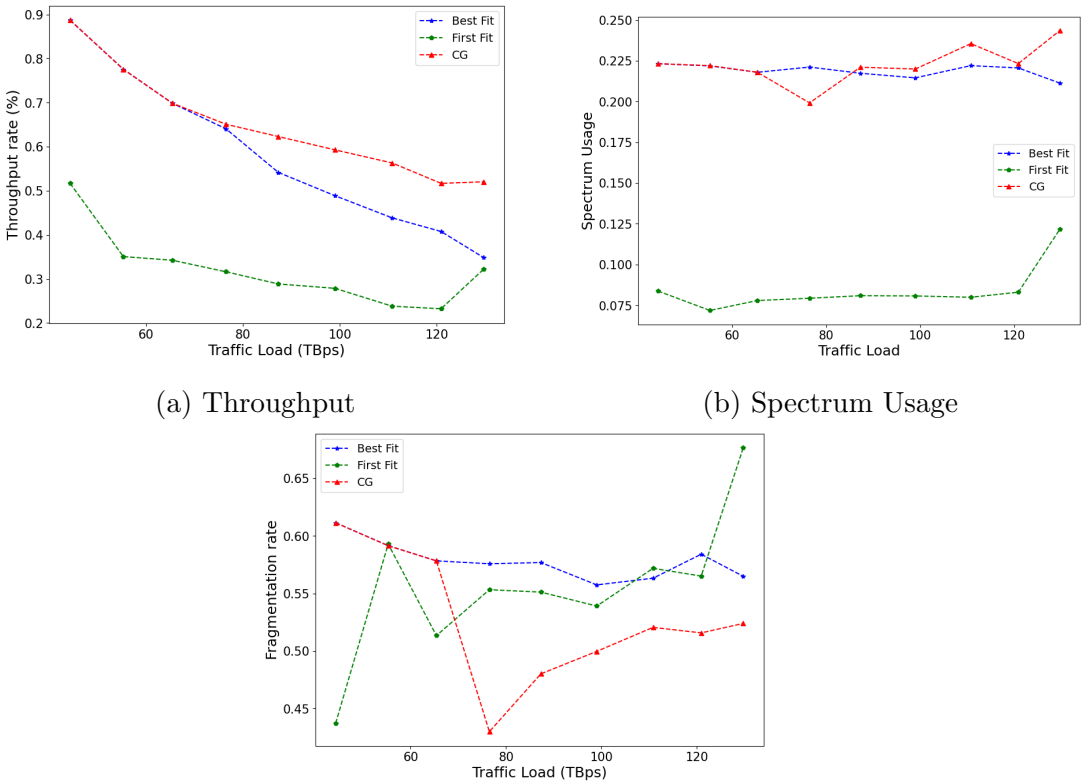
SU dropped 38% (from 75% to 37%). Another interesting point is that the LRB value is almost equal to the LP solution, it means that the LP solution is optimal.

Figures 4.3a and 4.3b compare the throughput and spectrum usage among column generation solution (CG), FF, and BF algorithms. While the CG solution achieves the highest throughput–on average, CG performs better than FF by 105% and better than BF by 16%, its spectrum usage is nearly identical to that of the BF approach.

Figure 4.3c presents the fragmentation rates–computed using the F^{RSS} formula as defined in [53]–for the three algorithms: Column Generation (CG), First-Fit (FF), and Best-Fit (BF). Although fragmentation is not explicitly minimized in the optimization constraints, we observe that CG consistently achieves a lower fragmentation rate compared to BF. This can be attributed to the behavior of BF, which always selects the frequency slots that offer the best immediate OSNR during lightpath computation. However, this short-sighted selection strategy often leads to inefficient spectrum utilization in the long term.

Algorithm 4.1 First-Fit algorithm.

```
K: set of requests
S: set of frequency slots
k(p): status of request k associated with path p
P ← ∅
for k ∈ K do
    g ← k−shortest path of k
    P ← P ∪ g
end for
P′ ← ∅
for s ← 0, s < |S|, s ← s + 1 do
    for p ∈ P do
        if p cannot be routed or k(p) = GRANTED then
            Continue
        else
            Skip ← FALSE
            for p′ ∈ P′ do
                if OSNR constraint of p′ is not satisfied then
                    Skip ← TRUE
                    Break
                end if
            end for
            if not Skip then
                Route p with s is its first slot
                P′ ← P′ ∪ {p}
                k(p) ← GRANTED
            end if
        end if
    end for
end for
```



(a) Throughput

(b) Spectrum Usage

(c) Fragmentation Rate

Figure 4.3: Different traffic loads in the Spain network with 385 frequency slots

4.5.3 Channel Spacing

Figure 4.4 presents the RSA provisioning solution generated by our model for the Spain network with 250 requests and 100 frequency slots. Each column corresponds to a single link, numbered as in Figure 4.2. Compared to Figure 4.7, where SNR is not considered, the allocated frequency slots are more sparse, reflecting the tighter feasibility imposed by OSNR constraints (i.e., some spectral locations are left unused because they would not meet the required quality along the selected paths). In Figure 4.6 (Best-Fit), the FS allocation is more evenly distributed, improving spectrum utilization compared to Figure 4.5 (First-Fit). The decomposition model produces an even denser allocation than Best-Fit, which suggests higher packing efficiency and potentially reduced spectrum fragmentation.

Algorithm 4.2 Best-Fit algorithm.

K: set of requests

S: set of frequency slots

$k(p)$: status of request k associated with path p

$P \leftarrow \emptyset$

for $k \in K$ **do**

$g \leftarrow k-$shortest path of k

$P \leftarrow P \cup g$

end for

$P' \leftarrow \emptyset$

for $p \in P$ **do**

if $k(p) =$ GRANTED **then**

Continue

else

for $s \leftarrow 0, s < |S|, s \leftarrow s + 1$ **do**

$V_s \leftarrow$ OSNR value of p if all routed paths still satisfy their OSNR constraints, 0 otherwise

end for

$s' \leftarrow \text{argmax}_s V_s$

Route p with s' as its first slot

$P' \leftarrow P' \cup \{p\}$

$k(p) \leftarrow$ GRANTED

end if

end for

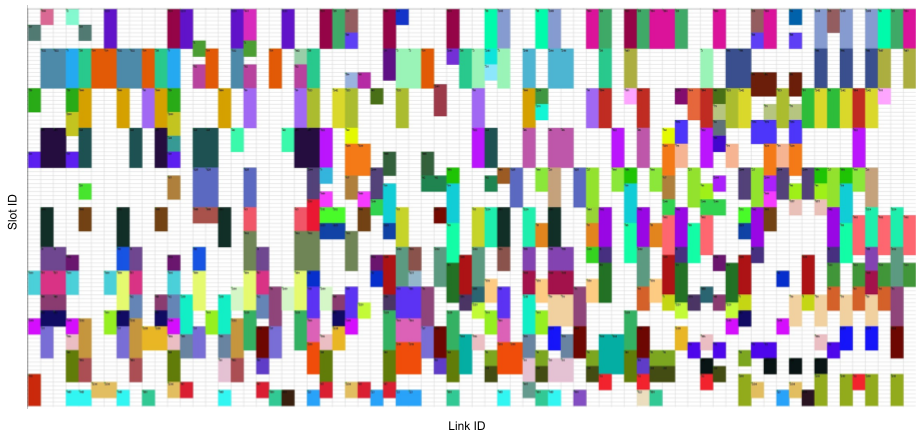


Figure 4.4: Provisioning of Spain network using decomposition model, 250 requests, 100 slots, SNR constraints accounted.

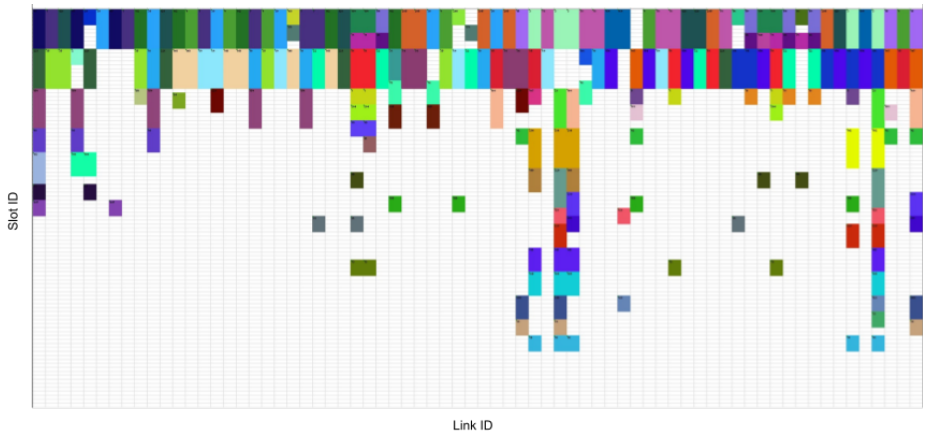


Figure 4.5: Provisioning of Spain network using First-Fit, 250 requests, 100 slots, SNR constraints accounted.

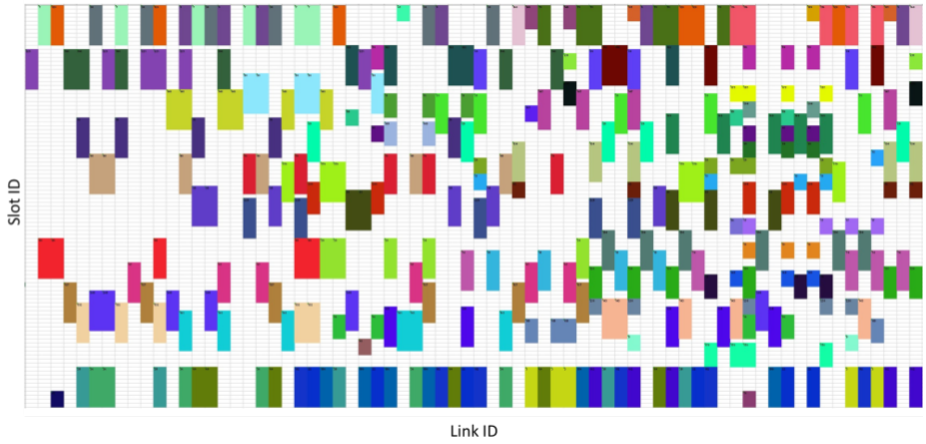


Figure 4.6: Provisioning of Spain network using Best-Fit, 250 requests, 100 slots, SNR constraints accounted.

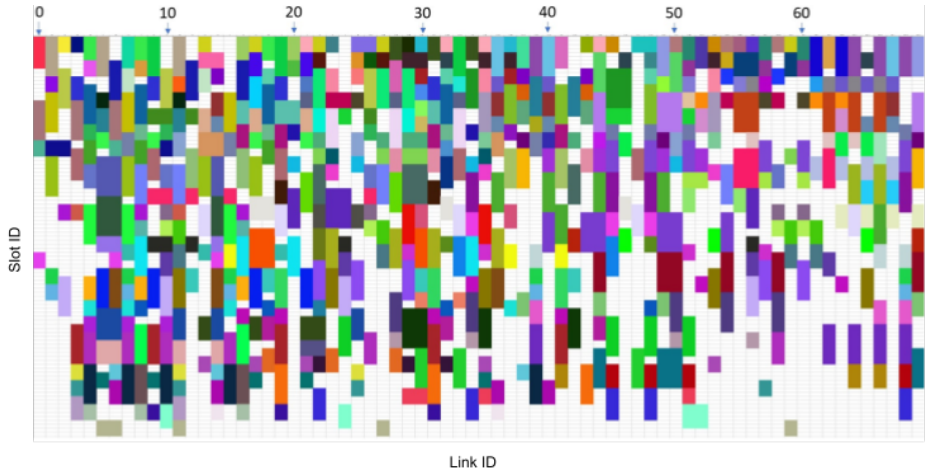


Figure 4.7: Provisioning of Spain network, 250 requests, 100 slots, SNR constraints not accounted.

4.6 Conclusion

We have seen that SNR constraints have big impact on the provisioning of the requests: the positions of the routed lightpaths become sparser. The mathematical model's performance is also better than First-Fit algorithm, however, the computation time is costly because of the amount of TS processes and the time each TS process took in each pricing, one of a method to resolve it is to let many tabu search process run on parallel at the same time.

4.7 Acknowledgment

I would like to express my sincere gratitude to Mohammad Sheikh Zefreh, CIENA, for his valuable guidance and support in helping me understand the GN model. His insights and explanations were instrumental in deepening my comprehension of the underlying concepts and their applications.

Chapter 5

Decomposition Model for Interference-Aware RSA in Elastic Optical Networks

5.1 Introduction

In the preceding chapter, a decomposition method was introduced to tackle the interference aware Routing and Spectrum Assignment (RSA) problem. However, the employed Tabu Search (CG-TS) algorithm exhibits certain limitations. Firstly, CG-TS consumes significant computational resources when dealing with the numerous sub-problems that require solving. Secondly, it falls short of achieving optimality in solving these sub-problems. This inadequacy ultimately influences the quality of the master problem's solution.

The present chapter is dedicated to addressing one of the limitations highlighted in the previous chapter. While CG-TS algorithm has proven effective, it fails to deliver optimal outcomes, resulting in inferior quality of generated columns and impacting the final solution. To surmount this challenge, this study introduces a novel solution specifically designed to rectify the optimization issues neglected by CG-TS.

5.2 Problem Statement

Given an optical network $G = (V, L)$, where V represents the optical nodes (e.g., Reconfigurable Optical Add-Drop Multiplexer ROADMs) and L represents the fiber links. The frequency spectrum in each fiber link is segmented into a collection of frequency slots, each with a bandwidth of 12.5 GHz.

Each fiber link is divided into a number of spans, with stations (ROADM/Amplifier) at both ends, see Figure 5.1.

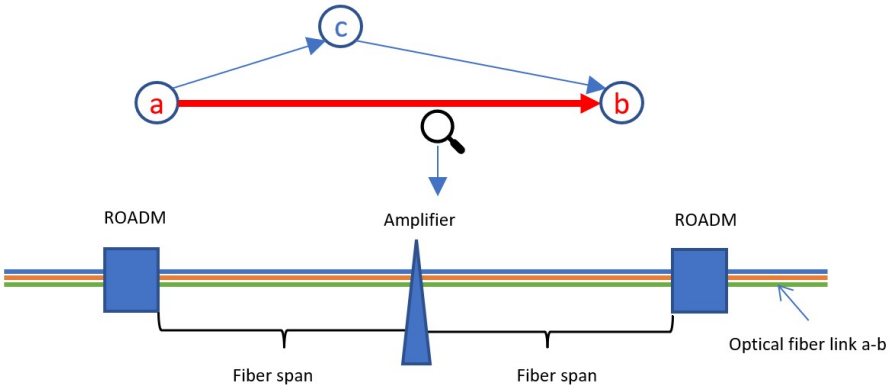


Figure 5.1: Span vs. Link.

We define the set of traffic requests as K, where each request is indexed as k, and is identified by its source, destination, and data rate d_k. Our focus lies on resolving the Routing and Spectrum Allocation (RSA) problem with the goal to optimize the network throughput while considering the interference estimated by the OSNR.

The successful fulfillment of a request requires compliance with both contiguity and continuity constraints, meaning that the assigned frequency slots (FSs) for the request must be contiguous within the assigned channel and consistent across all optical fibers through which the channel is routed. The continuity constraint is expressed mathematically in (5.3.3)-(5.11) and the contiguity constraint is located at 5.3.4-(5.22).

In addition, the OSNR constraint must be satisfied by each granted request k,

which means that the OSNR value of the channel i assigned to request k should be greater than or equal to its corresponding threshold value calculated using the Shannon formula. This constraint is explained in detail in 5.2.2.

5.2.1 Physical Layer Impairment

In optical fiber, PLI is composed of:

- Additive Spontaneous Emission noise (ASE) caused by the amplifiers
- Non-Linear Interference (NLI) caused by the Kerr effect consist of self-channel interference (SCI) and cross channel interference (XCI)

The ASE noise is calculated by:

$$G_{\text{ASE}}(f_i) = N_s^i(e^{2\alpha L} - 1)h\, n_{sp}\, f_i \tag{5.1}$$

The Gaussian Noise (GN) model in [47, 64] expressed the NLI as:

$$G_{\text{NLI}}(f_i) = G_{\text{SCI}}(f_i) + G_{\text{XCI}}(f_i) \tag{5.2}$$

$$G_{\text{SCI}}(f_i) = \mu N_s^i G_i^3 \text{asinh}(\rho B_i^2) \tag{5.3}$$

$$G_{\text{XCI}}(f_i) = \mu \sum_{j \neq i} G_j^2 G_i N_s^{ij} \ln\left(\frac{|f_j - f_i| + B_j/2}{|f_j - f_i| - B_j/2}\right) \tag{5.4}$$

where μ, ρ are related to fiber parameters:

$$\mu = \frac{8}{27}\frac{\gamma^2}{\pi\alpha|\beta_2|}$$

$$\rho = \frac{\pi^2|\beta_2|}{4\alpha}$$

The OSNR is then calculated by:

$$\text{SNR}_i = \frac{G_i}{G_{\text{ASE}}(f_i) + G_{\text{NLI}}(f_i)} \tag{5.5}$$

5.2.2 Reach Table

A reach-based approach was employed to determine the most appropriate modulation format. To generate the reach table, a two-step process was carried out. Firstly, a single optical fiber with C-band spectrum consisting of 380 frequency slots, each with

Table 5.1: Parameters and Variables in PLI formulation

Symbol	Meaning
	VARIABLES
f_i	center frequency of channel i
N_s^i	number of spans of channel i
N_s^{ij}	number of spans shared between channel i and j
G_i	Power Spectrum Density (PSD) of channel i
B_i	Bandwidth of channel i
	PARAMETERS
L	span length (km)
α	power attenuation (dB/km)
h	Planck's constant
n_{sp}	spontaneous noise factor
γ	non-linear coefficient
β_2	group velocity dispersion

a bandwidth of 12.5 GHz, was populated with channels having identical bandwidth b, data rate d and power by the same G. Subsequently, the maximum number of spans on the fiber that satisfied the OSNR constraints for each of the channels was computed. The bandwidth b took values from the set $\{37.5, 62.5, 87.5, 112.5\}$ GHz and the data rate d ranged from $\{100, 200, 400\}$ Gbps.

An example of the aforementioned setup is shown in Figure 5.2, where b corresponds to a bandwidth of 37.5 GHz, which is equivalent to 3 FSs. The parameter g represents the guardband with a value of 6.25 GHz, while $n_p = 4$ denotes the total number of FSs allocated to one channel.

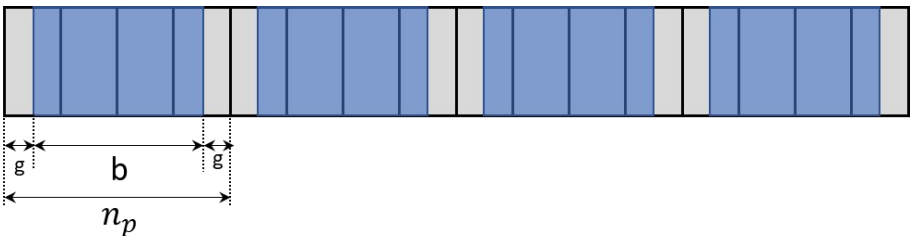


Figure 5.2: A link provisioning.

Shannon's formula states that the maximum channel capacity C computed by

$$C = B \log_2(1 + \text{SNR}) \tag{5.6}$$

where B is the bandwidth, SNR is the signal to noise ratio. Then OSNR constraints are described as:

$$\begin{aligned} &\text{SNR}_i \geq \text{SNR}_{T,i} \\ &\text{SNR}_{T,i} = 2^{d/b} - 1 \end{aligned} \tag{5.7}$$

where $\text{SNR}_{T,i}$ represents the minimum required value of SNR for channel i to ensure the quality of the signal is sufficient for successful decoding. By combining (4.4) and (5.7), the final reach table has been generated, which is presented in Table 5.2.

Table 5.2: Reach Table.

Bit rate d	Bandwidth b	Bandwidth + guardbands	Number of spans N_s
100	37.5	50	57
	62.5	75	137
	87.5	100	222
	112.5	125	307
200	37.5	50	7
	62.5	75	34
	87.5	100	69
	112.5	125	107
400	62.5	75	3
	87.5	100	11
	112.5	125	24

5.3 Mathematical Model

The column generation (CG) method was chosen to solve the problem due to our primary concern being the scalability issue. CG is a widely recognized method for tackling this problem.

5.3.1 OSNR constraint

Since the OSNR formula is not additive, we took its inverse into consideration. This led us to express the OSNR constraint as follows:

$$\begin{aligned} \frac{1}{\mathrm{SNR}_{\pi}} &\leq \frac{1}{\mathrm{SNR}_{T,\pi}} \\ \iff \frac{1}{\mathrm{SNR}_{T,\pi}} &\geq \frac{G_{\mathrm{ASE}}(f_{\pi}) + G_{\mathrm{SCI}}(f_{\pi}) + G_{\mathrm{XCI}}(f_{\pi})}{G_{\pi}} \\ \iff \frac{G_{\mathrm{XCI}}(f_{\pi})}{G_{\pi}} &\leq \underbrace{\frac{1}{\mathrm{SNR}_{T,\pi}} - \frac{G_{\mathrm{ASE}}(f_{\pi}) + G_{\mathrm{SCI}}(f_{\pi})}{G_{\pi}}}_{C_{\pi}} \end{aligned} \tag{5.8}$$

5.3.2 Variables and parameters

In the context of this chapter, a lightpath configuration γ is defined as a sub-provisioning linked to a specific link ℓ, wherein all the lightpaths within γ must originate from the same starting link ℓ. An illustration of a lightpath configuration associated with a link ℓ^* can be seen in Figure 5.3. For the complete definition of the variables and parameters in Sections 5.3.3 and 5.3.4, kindly refer to Table 5.3.

Table 5.3: Variables and parameters.

Symbol	Meaning
	VARIABLES
k	a request
s	a frequency slot
π	a lightpath/channel
γ	a lightpath configuration
K	set of all requests
S	set of all frequency slots
Γ_k	a set of lightpaths associated with request k
z_γ	=1 if γ is selected, 0 otherwise
x_k	=1 if request k is granted, 0 otherwise
a_k^γ	=1 if request k is granted in γ, 0 otherwise
$a_{\ell s}^\gamma$	=1 if slot s at link ℓ in γ is occupied, 0 otherwise
y_π^γ	=1 if π appear in γ, 0 otherwise
θ_π^γ	total cross interference (XCI) cause by all the lightpaths in γ to π
v_p	=1 if path p is selected, 0 otherwise
a_k	=1 if request k is granted, 0 otherwise
$a_{s\tilde{\ell}}^p$	=1 if path p occupy slot s in link $\tilde{\ell}$, 0 otherwise
$b_{s\tilde{\ell}}^p$	=1 if slot s is the starting slot of path p in link $\tilde{\ell}$, 0 otherwise
$u^{(i)}$	dual value of constraint i
	PARAMETERS
d_k	data rate of request k
n_p	the number of slot that path p require
M	a big enough number
C_π	calculate using 4.13

5.3.3 Master problem

$$\max \sum_{k \in K} d_k x_k \qquad \text{(Throughput)} \tag{5.9}$$

subject to:

$$\sum_{\gamma \in \Gamma_\ell} z_\gamma \leq 1 \qquad \ell \in L \tag{5.10}$$

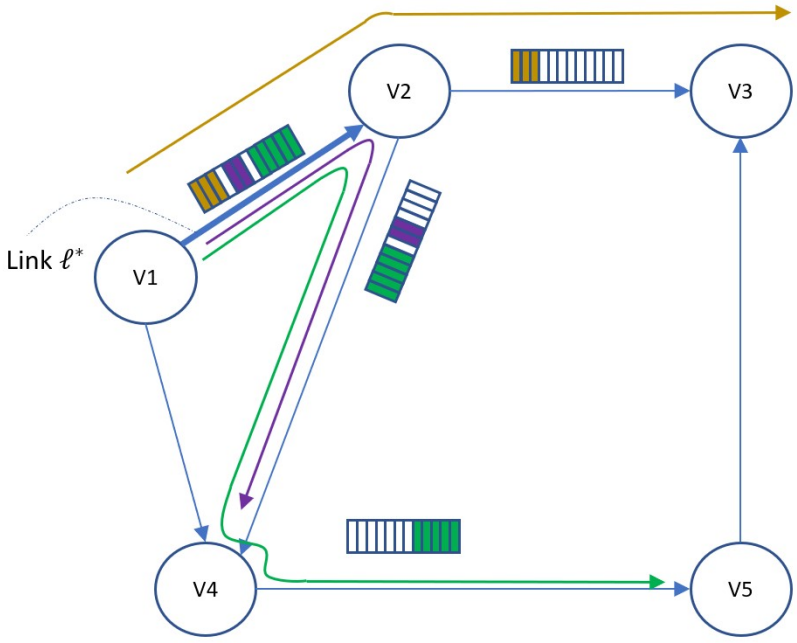


Figure 5.3: Lightpath configuration example.

Constraints (5.10) ensure that for each link, there's at most one configuration associated with selected.

$$\sum_{\gamma \in \Gamma} a_{\ell s}^{\gamma} z_{\gamma} \leq 1 \qquad \ell \in L, s \in S \tag{5.11}$$

Constraints (5.11) ensure no-overlapping FSs between selected configurations.

$$x_k \leq \sum_{\gamma \in \Gamma} a_k^{\gamma} z_{\gamma} \qquad k \in K \tag{5.12}$$

Constraints (5.12) make sure there exists a configuration that grants request k if $x_k = 1$.

$$\sum_{\gamma \in \Gamma} z_{\gamma} (\theta_{\pi}^{\gamma} + (M - C_{\pi}) y_{\pi}^{\gamma}) \leq M \qquad \pi \in \Pi_k, k \in K \tag{5.13}$$

Constraints (5.13) check the SNR constraint of every selected lightpath.

$$z_\gamma \in \{0,1\} \qquad \gamma \in \Gamma \tag{5.14}$$

$$x_k \in \{0,1\} \qquad k \in K \tag{5.15}$$

5.3.4 Pricing problem

Each link $\ell \in L$ is associated with a pricing problem that is updated after every iteration. The solution to the pricing problem generates a new lightpath configuration. Given a maximization $c^T x$ subject to $Ax \leq b, x \geq 0$, the objective value of the pricing is equivalent to the reduced cost of the new column, presented by z_γ, which is calculated as $c - A^T y$, where y represents the dual vector. In the present study, the reduced cost can be rewrote as:

$$\begin{aligned} \text{COST}_\gamma = -u^{(5.10)} &- \sum_{s\in S}\sum_{\ell\in L} u_{s\ell}^{(5.11)} \underbrace{\sum_{k\in K}\sum_{p\in P_k} \delta_\ell^p a_{s\tilde{\ell}}^p}_{a_{\ell s}} \\ &- \sum_{k\in K}\sum_{\pi\in\Pi_k} u_\pi^{(5.13)}(\theta_\pi + (M - C_\pi)y_\pi) \\ &+ \sum_{k\in K} u_k^{(5.12)}\, a_k \end{aligned} \tag{5.16}$$

The following is the pricing problem:

$$\max \text{COST}_\gamma \tag{5.17}$$

subject to:

$$\sum_{p\in P_k} v_p = a_k \qquad k \in K \tag{5.18}$$

Constraints (5.18) guarantee that at most one path (routing) is selected for request k if it is approved in the ongoing γ.

$$a_{s\tilde{\ell}}^p \leq v_p \qquad p \in P_k, k \in K, s \in S \tag{5.19}$$

Constraints (5.19) ensure that variable $y_p = 1$ if path p occupies any slot s on link $\ell^\star$.

$$\sum_{p \in P_k} \frac{1}{n_p} \sum_{s \in S} a^p_{s\tilde{\ell}} = a_k \qquad k \in K \tag{5.20}$$

Constraints (5.20) ensure the total number of slots for p match with n_p.

$$\sum_{p \in P_k} \sum_{s \in [1, |S| - n_p + 1]} b^p_{s\tilde{\ell}} = a_k \qquad k \in K \tag{5.21}$$

Constraints (5.21) ensure a unique starting slot for each request.

$$\sum_{i=0}^{n_p - 1} a^p_{t+i,\tilde{\ell}} \geq n_p b^p_{t\tilde{\ell}} \qquad t \in [1, |S| - n_p + 1]$$
$$k \in K, p \in P_k \tag{5.22}$$

Constraints (5.22) express the contiguity constraints on link $\ell^\star$.

$$\sum_{k \in K} \sum_{p \in P_k} a^p_{s\tilde{\ell}} \leq 1 \qquad s \in S \tag{5.23}$$

Constraints (5.23) ensure that each slot is used at most once in the overall set of connection requests.

$$v_p \in \{0, 1\} \qquad p \in P_k, k \in K \tag{5.24}$$
$$a_k \in \{0, 1\} \qquad k \in K \tag{5.25}$$
$$a^p_{s\tilde{\ell}} \in \{0, 1\} \qquad p \in P_k, k \in K, s \in S \tag{5.26}$$
$$b^p_{s\tilde{\ell}} \in \{0, 1\} \qquad p \in P_k, k \in K, s \in S \tag{5.27}$$

The interference caused by the under-construction configuration γ to lightpath π

is expressed as the term θ_π in (5.16), and is defined as:

$$\begin{aligned}
\theta_\pi^\gamma &= \sum_{\substack{\pi' \in \gamma \\ \pi' \neq \pi}} \frac{G_{\text{XCI},\pi}^{\pi'}}{G_\pi} \\
&= M \sum_{\substack{\pi' \in \gamma \\ \pi' \neq \pi}} N_s^{\pi,\pi'} \ln \left(\frac{|f_\pi - f_{\pi'}| + B_{\pi'}/2}{|f_\pi - f_{\pi'}| - B_{\pi'}/2} \right) \\
&= M \sum_{\substack{\pi' \in \gamma \\ \pi' \neq \pi}} N_s^{\pi,\pi'} \ln \left(1 + \frac{B_{\pi'}}{|f_\pi - f_{\pi'}| - B_{\pi'}/2} \right) \\
&= M \sum_{\substack{\pi' \in \gamma \\ \pi' \neq \pi}} N_s^{\pi,\pi'} \ln \left(1 + \frac{1}{|f_\pi - f_{\pi'}|/B_{\pi'} - 1/2} \right)
\end{aligned}$$

5.3.5 Lagrangian bound

We opted to utilize the Lagrangian bound as the upper limit of the master problem to gauge the model's efficacy. Consider a general problem:

$$\begin{aligned}
&\text{maximize } f(x) \\
&\text{subject to } g(x) \leq 0 \\
&x \in S
\end{aligned} \tag{5.28}$$

The Lagrangian relaxation of (5.28) can be written as:

$$\begin{aligned}
&\text{maximize } f(x) - \lambda g(x) \\
&x \in S
\end{aligned} \tag{5.29}$$

where λ is a non-negative vector. The Lagrangian relaxation of the master problem (5.9)-(5.15) can be written as:

$$\begin{aligned}
LR_\tau = &\sum_{k \in K} d_k x_k + \sum_{\ell \in L} u_\ell^{(5.10)} (1 - \sum_{\gamma \in \Pi_\ell} z_\gamma) \\
&+ \sum_{\ell \in L} \sum_{s \in S} u_{s\ell}^{(5.11)} (1 - \sum_{\gamma \in \Gamma} a_{\ell s}^\gamma z_\gamma) \\
&+ \sum_{k \in K} u_k^{(5.12)} (\sum_{\gamma \in \Gamma} a_k^\gamma z_\gamma - x_k) \\
&+ \sum_{k \in K} \sum_{\pi \in \Pi_k} u_\pi^{(5.13)} (M - \sum_{\gamma \in \Gamma} z_\gamma (\theta_\pi^\gamma + (M - C_\pi) y_\pi^\gamma))
\end{aligned} \tag{5.30}$$

subject to:

$$z_\gamma \in \{0,1\} \qquad \gamma \in \Gamma \tag{5.31}$$

$$x_k \in \{0,1\} \qquad k \in K \tag{5.32}$$

From (5.30) we have:

$$\begin{aligned} LR_\tau = & \sum_{k\in K} x_k \underbrace{(d_k - u_k^{(5.12)})}_{=\text{RCOST}(x_k)} + \sum_{\ell\in L} u_\ell^{(5.10)} + \sum_{\ell\in L}\sum_{s\in S} u_{s\ell}^{(5.11)} \\ & + \sum_{k\in K}\sum_{\pi\in\Pi_k} u_\pi^{(5.13)} M \\ & + \sum_{\ell\in L}\sum_{\gamma\in\Gamma_\ell} z_\gamma(-u_\ell^{(5.10)} - \sum_{s\in S}\sum_{\ell\in L} u_{s\ell}^{(5.11)} \sum_{k\in K}\sum_{p\in P_k} \delta_\ell^p a_{s\tilde{\ell}}^p \\ & - \sum_{k\in K}\sum_{\pi\in\Pi_k} u_\pi^{(5.13)}(\theta_\pi + (M - C_\pi)y_\pi) + \sum_{k\in K} u_k^{(5.12)}\, a_k) \end{aligned} \tag{5.33}$$

The reduced cost of variable x_k is denoted as $\text{RCOST}(x_k)$ in equation (5.33). If x_k is a non-basic variable, that implies $x_k = 0$ so $t = x_k\text{RCOST}(x_k) = 0$. If x_k is a basic variable, it may or may not be equal to 0, but the reduced cost $\text{RCOST}(x_k)$ is 0 in either case, and hence $t = 0$. Therefore, the sum $\sum_{k\in K} x_k\text{RCOST}(x_k) = 0$.

Denote u_τ the dual vector at iteration τ and b the right hand side constant vector of the constraints of the master problem, we have:

$$u_\tau^T b = \sum_{\ell\in L} u_\ell^{(5.10)} + \sum_{\ell\in L}\sum_{s\in S} u_{s\ell}^{(5.11)} + \sum_{k\in K}\sum_{\pi\in\Pi_k} u_\pi^{(5.13)} M$$

Then LR_τ can be rewritten as:

$$\begin{aligned} LR_\tau &\le u_\tau^T b + \sum_{\ell\in L}\sum_{\gamma\in\Gamma_\ell} z_\gamma \text{COST}_\gamma \\ &\le u_\tau^T b + \sum_{\ell\in L}\sum_{\gamma\in\Gamma_\ell} \overline{\text{COST}}_\gamma \end{aligned} \tag{5.34}$$

From (5.34), Lagrangian bound (LRB) at iteration τ is computed by:

$$\text{LRB}_\tau = u_\tau^T b + \sum_{\ell\in L}\sum_{\gamma\in\Gamma_\ell} \overline{\text{COST}}_\gamma$$

where $\overline{\text{COST}}_\gamma$ is the non-negative solution of each solved pricing problem. Then the final LRB is the minimum between the values of LRB_τ:

$$\text{LRB} = \min_\tau \text{LRB}_\tau \tag{5.35}$$

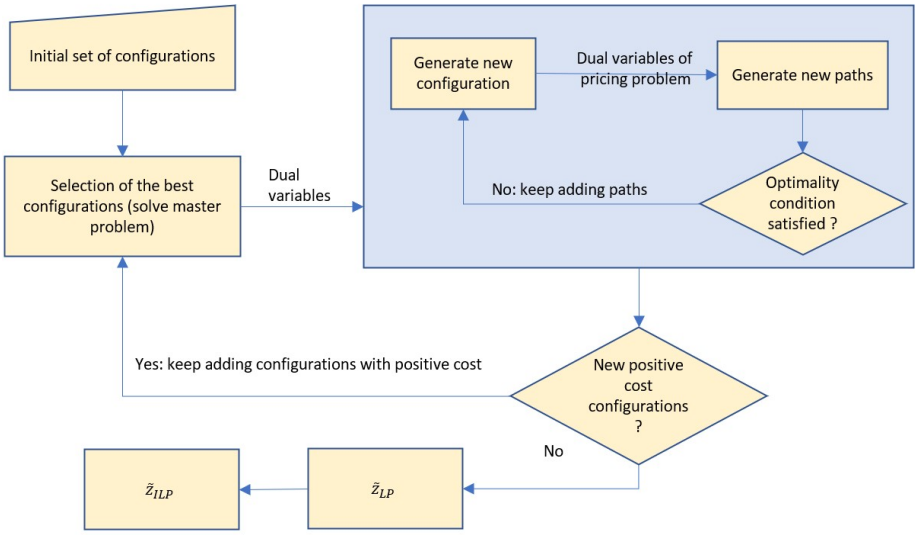


Figure 5.4: Column Generation flowchart.

5.4 Solution Process

5.4.1 Column Generation

Figure 5.4 illustrates the process of column generation (CG). At each iteration, the master problem is solved as a Linear Program (LP), and its dual values are used to solve the pricing models associated with each link of the network. If the pricing problem has a feasible solution with a positive reduced cost, the resulting ILP solution is returned as a new column (lightpath configuration) for the master problem. If not, a lower level pricing problem generates more routing paths to generate additional columns for pricing. This process continues until no more lightpath configurations are generated, after which the master problem is solved as an ILP to obtain an ϵ-optimal solution.

The increase in the number of generated lightpaths results in a proportional increase in the number of logarithmic terms in θ_π as stated in (5.16). Due to the high cost and inefficiency of solving the pricing problem as it is, an alternative approach becomes necessary to solve the problem.

5.4.2 Solving Pricing Problem

In this section, we address the pricing problem (5.16)–(5.27) by transforming it into a Maximum Weighted Independent Set problem [72, 91] through the use of a conflict graph.

We construct a conflict graph $G'(E, V')$, where $V' = \{v_{root}\} \cup \{(k, p, s) | k \in K, p \in P, s \in S\} \cup \{(k, -1, -1) | k \in K\}$, E is the set of edges such that for every $(u, v) \in E$, u and v are mutually exclusive in the final provisioning. A vertex represented by (k, p, s) denotes that request k will be accommodated by routing path p at slot s, while a vertex represented by $(k, -1, -1)$ indicates that request k will not be granted. We can find a feasible solution for the original pricing problem by identifying a set of vertices in G', denoted as $V'' \subset V'$, that satisfies both the independent set and vertex cover properties.

Figure 5.5 depicts an example of the conflict graph generated by our approach. In this graph, each request k_i is associated with a component, where each node (p_j, s_h) represents the granting of k_i using path p_j starting at slot s_h, except for the node (None, None) which represents that request k_i will not be granted. Each component is a complete graph, implying that every node pair u, v within the same component is connected by an edge $(u, v) \in E$. Moreover, if two nodes (k_i, p_j, s_h) and (k'_i, p'_j, s'_h) have overlapping slots when routing p_j at s_h and p'_j at s'_h, respectively, then an edge $e \in E$ connects these two nodes.

For each vertex, its value will be computed by:

$$h_v = \begin{cases} -u^{(5.10)} & \text{if } v = v_{root} \\ 0 & \text{if } v = (k, -1, -1) \\ u^{(5.12)}_{k'} - \sum\limits_{s \in S} \sum\limits_{\ell \in L} u^{(5.11)}_{s\ell}\, \delta^{p'}_{\ell}\, a^{\pi'}_{s} - & \text{if } v = (k', p', s') \\ \sum\limits_{k \in K} \sum\limits_{\pi \in \Pi_k} u^{(5.13)} (\theta^{\pi'}_{\pi} + (M - C_\pi) y_\pi) & \end{cases} \tag{5.36}$$

where $\pi' = (p', s')$ denotes a lightpath with path p' and starting slot s'. $a^{\pi'}_{s} = 1$ if π' occupies slot s, 0 otherwise. $\theta^{\pi'}_{\pi}$ represents the interference caused by π' to π, if π and π' have conflicting slots, then $\theta^{\pi'}_{\pi} = 0$. $y_\pi = 1$ if $\pi' = \pi$, 0 otherwise.

We propose a MILP model:

$$\max \sum_{v \in V'} h_v x_v \tag{5.37}$$

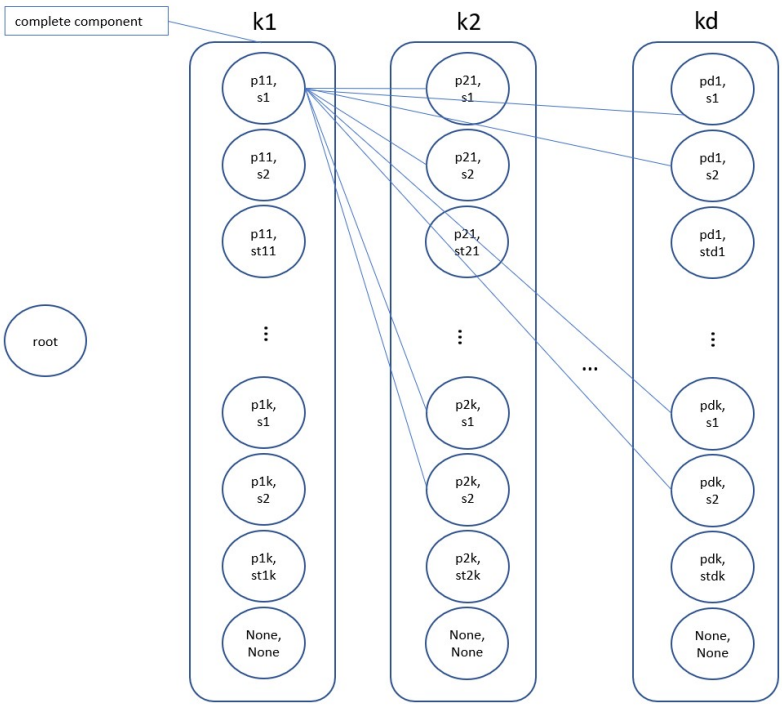


Figure 5.5: Conflict Graph.

subject to:

$$
\begin{aligned}
x_{v_{root}} &= 1 \\
\sum_{v \in V_k} x_v &= 1 \qquad k \in K \\
\sum_{v \in V_s} x_v &\leq 1 \qquad s \in S \\
x_v &\in \{0,1\} \qquad v \in V
\end{aligned}
$$

where V_k is the set of nodes associated with request k. V_s represents the set of nodes where if a node $v = (k', p', s')$ and $v \in V_s$, then the lightpath $\pi' = (p', s')$ will occupy slot s. The feasible provisioning's objective value expressed in equation (5.37) will be equivalent to the original pricing objective value stated in equation (5.16). We call this solution as Conflict Graph MILP (CG-MILP).

5.4.3 Nested Column Generation

We introduce the nested column generation, aimed at generating additional columns (i.e., new paths) to be used in the pricing. The main objective of this model is to

maximize the reduced cost of $\delta_\ell^{p'}$, which can be found in equations (5.36) and (5.37). For each pricing linked to a specific link ℓ and for every request with a source and destination (v_s, v_d), the nested CG is implemented as follows:

$$\max \; -\sum_{s\in S}\sum_{l\in L} u_{sl}^{(5.11)}\delta_l \tag{5.38}$$

subject to:

$$\delta_\ell = 1 \tag{5.39}$$

$$\delta_l = 0 \qquad l \in v_s^+ \cup v_s^- \setminus \{\ell\} \tag{5.40}$$

$$\sum_{l\in v_d^-}\delta_l = 1 + \sum_{l\in v_d^+}\delta_l \tag{5.41}$$

$$\sum_{l\in v_i^+}\delta_l = \sum_{l\in v_i^-}\delta_l \qquad i \in V \setminus \{v_s, v_d\} \tag{5.42}$$

$$\delta_l \in \{0,1\} \qquad l \in L \tag{5.43}$$

where v_i^+, v_i^- are the sets of outgoing and incoming links of node v_i. This problem can be solved as a shortest path problem with the weight of each link $w_l = \sum_{s\in S} u_{sl}^{(5.11)}$, and the first link must be ℓ.

5.5 Numerical Results

5.5.1 Data Sets

We experimented using different set of requests and different amount of frequency slots on the Spain network topology. The data rate's value is generated within the set {100, 200, 400} Gbps with the respected distribution 40%, 30%, 30%.

5.5.2 Performance of the Model

In Table 5.4 is the numerical result we ran through several data set, we compared the throughput, spectrum usage (SU) between Best Fit (BF) heuristic, Tabu Search (CG-TS) in [43] and CG-MILP. The Relative Gain (RG) column computed by:

$$GP = \frac{z_{ILP}^2 - z_{ILP}^1}{z_{ILP}^1}$$

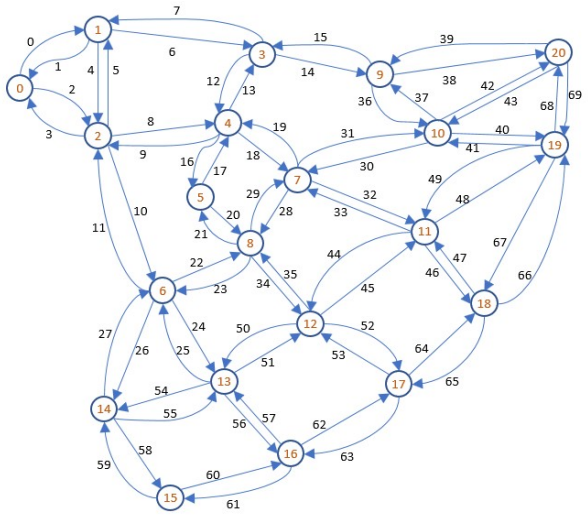


Figure 5.6: Spain network.

where z^1_{ILP} is the ILP result of Tabu Search and z^2_{ILP} is the ILP result of CG-MILP.

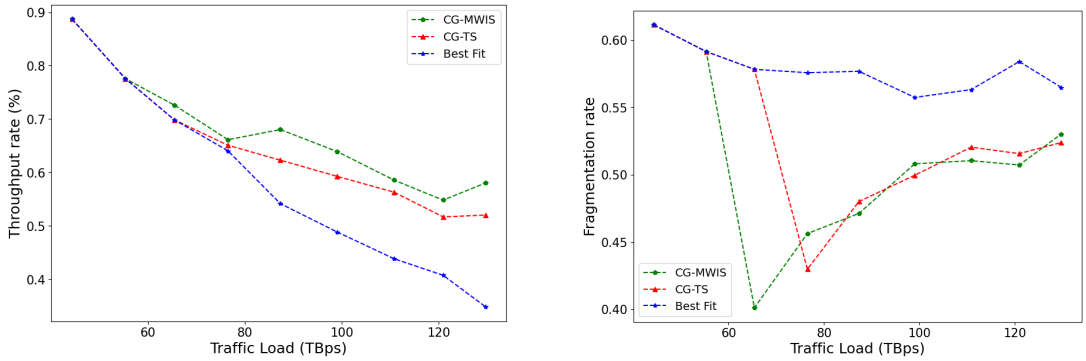
The performance of our proposed model is evaluated against two algorithms, Best-Fit and CG-TS. The experimental results demonstrate the superiority of our new approach over these algorithms. The generated columns by the proposed model exhibit better objective values, thus leading to a more effective master ILP solution. The comparison between the solutions obtained from CG-TS and CG-MILP for the same pricing problems in the initial iteration of the master problem is presented in Figure 5.8. It is evident that our proposed CG-MILP approach outperforms CG-TS in most cases, and even provides positive reduced cost columns that CG-TS fails to identify, such as in link 11 and link 28.

Figures 5.7a and 5.7b highlight the significant throughput improvement achieved by CG-MWIS on the Spain network with 385 frequency slots. Compared to CG-TS, CG-MWIS delivers an average throughput gain of 5%, with a maximum increase of 11%. More notably, CG-MWIS outperforms the Best Fit heuristic by a substantial margin, achieving an average improvement of 22% and reaching up to 66.6% in the best case. While the fragmentation rates of CG-MWIS and CG-TS are comparable, CG-MWIS consistently maintains better fragmentation performance than Best Fit.

One of the limitations of the CG-MILP method is its runtime, which increases

Nb slots	Nb re-quest	Total load	Best fit	Tabu Search				CG-MILP				RG versus	
				z_{ILP}	z_{LP}	SU	LRB	z_{ILP}	z_{LP}	SU	LRB	TS	BF
50	200	44.3	16.5	23.5	28.9	0.60	28.9	25.7	28.9	0.67	32.8	**0.09**	**0.56**
100	200	44.3	22.3	33.2	44.0	0.52	44.0	35.5	44.0	0.57	47.0	**0.07**	**0.59**
100	250	55.3	23.2	36.7	51.3	0.52	51.3	38.9	51.3	0.58	68.2	**0.06**	**0.68**
100	300	65.4	21.7	32.9	52.1	0.45	52.1	39.1	52.1	0.55	67.1	**0.19**	**0.80**
200	300	65.4	34.3	42.1	65.4	0.33	65.4	49.4	65.4	0.39	68.0	**0.17**	**0.44**
200	500	110.8	31.7	55.3	103.7	0.38	103.7	60.9	103.7	0.43	105.4	**0.10**	**0.92**
200	600	129.7	29.4	57.6	113.3	0.37	113.3	61.8	113.3	0.39	162.2	**0.07**	**1.10**
400	500	110.8	48.4	61.5	110.8	0.23	110.8	68.4	110.8	0.24	112.8	**0.11**	**0.41**
400	600	129.7	50.2	68.8	129.7	0.25	129.7	76.0	129.7	0.28	132.1	**0.10**	**0.51**

Table 5.4: Numerical result.



(a) Throughput rate

(b) Fragmentation

Figure 5.7: Comparing results of CG-MWIS with CG-TS and BF on Spain network with 385 frequency slots

as the iterations of the master problem progress and the number of binary variables grows. For instance, in the largest dataset consisting of 600 requests and 400 slots, link 22 requires a considerable amount of time to solve, as it accumulates 38,318 binary variables and takes 788 seconds to complete. This significantly affects the total running time of the model. As shown in Figure 5.9, beyond the sixth iteration, 25% of the pricing problems take more than 400 seconds to solve, and some problems require over 700 seconds to solve. This limitation is evidenced by Figure 5.10, as the number of iterations increases, the time required to complete each iteration also increases. In the final iteration, the model requires over two hours to execute.

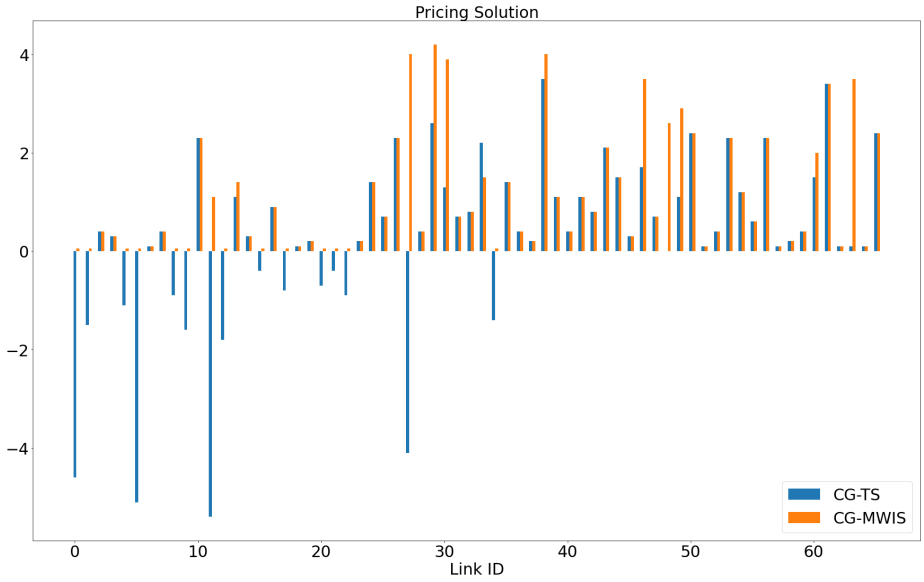


Figure 5.8: Comparing Pricing Objective Values of first iteration.

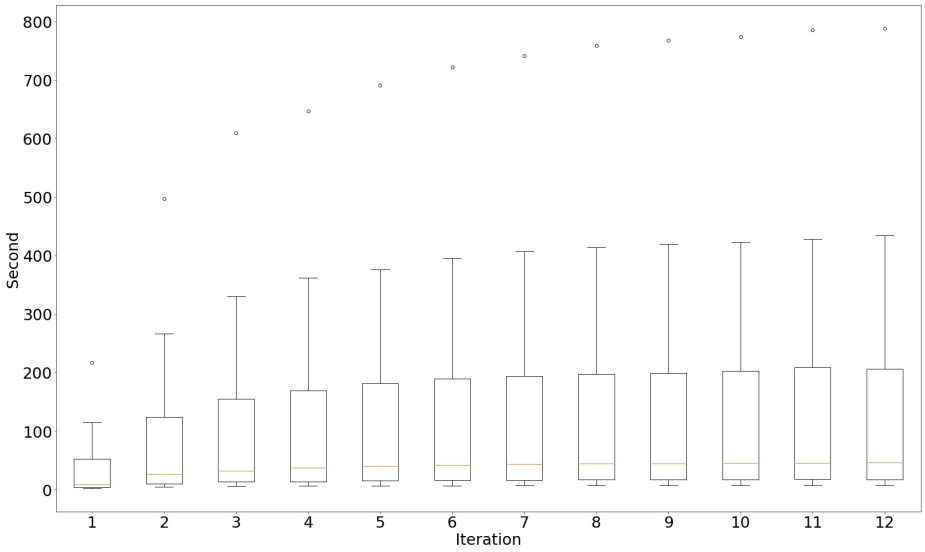


Figure 5.9: Runtime distribution.

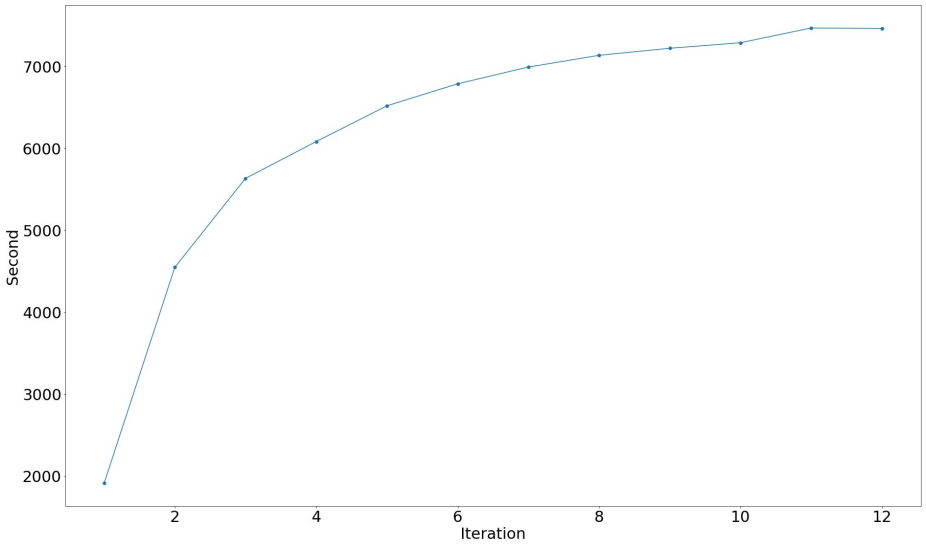


Figure 5.10: Runtime per Iteration.

Algorithm 5.1 Best-Fit algorithm.

K: set of requests
S: set of frequency slots
$k(p)$: status of request k associated with path p
$P \leftarrow \emptyset$
for $k \in K$ **do**
    $g \leftarrow k-$shortest path of k
    $P \leftarrow P \cup g$
end for
$P' \leftarrow \emptyset$
for $p \in P$ **do**
    if $k(p) =$ GRANTED **then**
        Continue
    else
        for $s \leftarrow 0, s < |S|, s \leftarrow s + 1$ **do**
            $V_s \leftarrow$ SNR value of p if all routed paths still satisfy their SNR constraints, 0 otherwise
        end for
        $s' \leftarrow \text{argmax}_s V_s$
        Route p with s' as its first slot
        $P' \leftarrow P' \cup \{p\}$
        $k(p) \leftarrow$ GRANTED
    end if
end for

5.6 Conclusions

Our research introduces a near-optimal solution for interference-aware large-scale RSA without considering the power variable, which is an improvement over our previous work in terms of solution quality. However, the runtime of the proposed model remains a significant concern, as indicated by the increasingly long computation times required for larger datasets. Despite this limitation, our work offers a benchmark model for evaluating the effectiveness of heuristics commonly used in the telecommunications

industry. Our findings underscore the importance of developing optimization models that strike a balance between performance and runtime, which is an area for future research.

5.7 Acknowledgment

I express my sincere gratitude to Dr. Abdelhak Bentaleb for his valuable time, insightful feedback, and thoughtful suggestions, which greatly contributed to improving the clarity and quality of the writing in this chapter.

Chapter 6

Availability vs. Latency and Shared vs. Dedicated Protection for O-RAN

6.1 Introduction

The rollout of 5G—and the imminent arrival of 6G—has enabled the telecommunications industry to effectively manage the rapid global growth of connected devices. Projections indicate that this trend will continue, with the total number of Internet of Things (IoT) devices expected to reach 18.4 billion by 2026 [79]. To support this surge, researchers and network operators are increasingly focused on integrating a wider range of vertical applications into 5G/6G infrastructure. These applications span across the service categories of enhanced mobile broadband (eMBB), ultra-reliable low-latency communications (URLLC), and massive machine-type communications (mMTC) for 5G, as well as immersive experiences, joint communications and sensing, and advanced sensing capabilities for 6G. While core networks are largely ready to meet these demands, Radio Access Networks (RANs) still require further innovation and optimization. One of the key enablers in this transformation is the Open Radio Access Network (O-RAN) architecture [32], especially when combined with technologies like Network Function Virtualization (NFV), Virtualized Network Functions (VNFs), and Service Function Chaining (SFCs). O-RAN aims to bridge the gap between access and core networks, enabling scalable, flexible deployment of

services.

O-RAN brings openness and intelligence to RANs by moving network functions into a virtualized environment and running them on a standardized cloud platform called O-Cloud. This open design makes it easier for different vendors' systems to work together. For example, providers can use standardized virtual functions, like the O-RAN Central Unit (O-CU), Distributed Unit (O-DU), and near-Real-Time RAN Intelligent Controller (NRT_RIC), on a variety of hardware, as long as it follows O-RAN specifications. By separating software from specific hardware brands, this approach makes it easier to expand the network and roll out new services quickly. Despite its

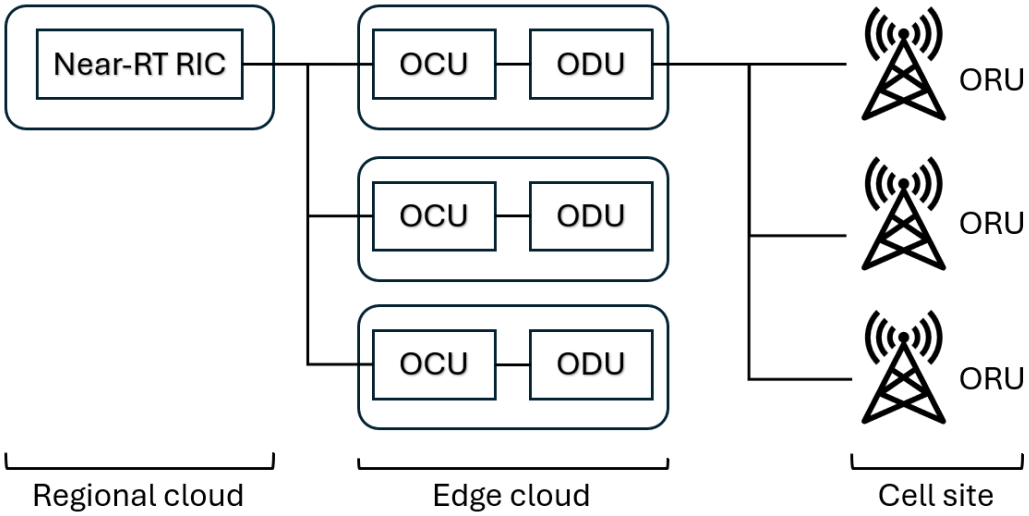


Figure 6.1: ORAN deployment, Scenario B

benefits, the openness and modularity of O-RAN introduce new challenges in ensuring reliability. Failures may arise not only from hardware but also from virtualized network components. Therefore, ensuring high availability in O-RAN deployments is both critical and complex.

To meet service quality goals, O-RAN operators use protection strategies that help keep virtual network functions (VNFs) running reliably and with minimal downtime. Since the functions in a typical O-RAN chain, comprising NRT_RIC, O-CU, and O-DU, are arranged in a straight line, they can use the same kinds of protection methods commonly used in core networks, such as *(i)* dedicated protection, *(ii)* shared protection and *(iii)* joint protection [28, 29, 74, 75, 80, 98]. While prior works have proposed heuristic and approximation-based solutions for these schemes, the literature lacks a

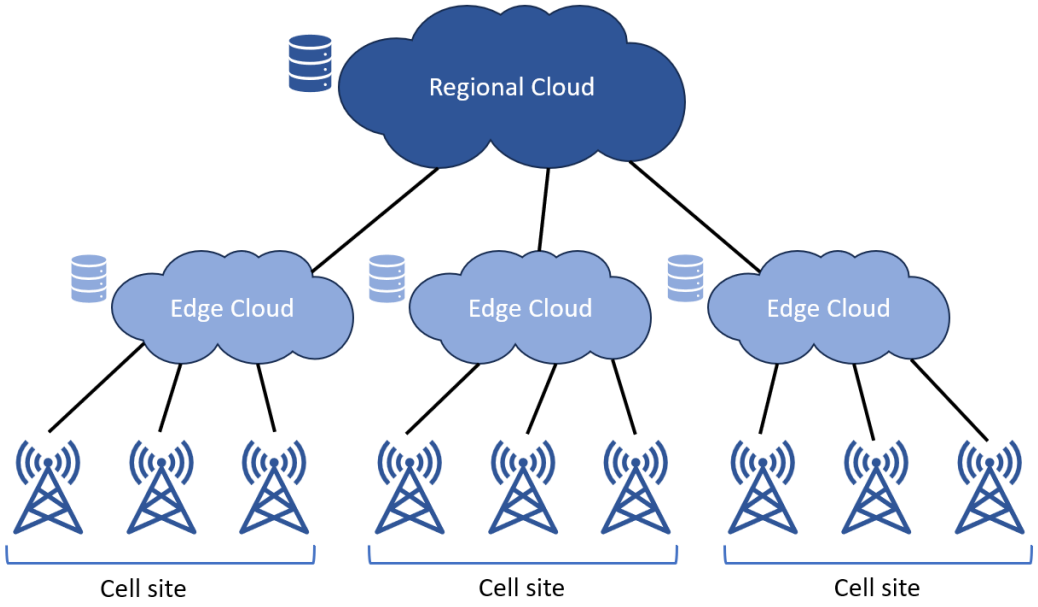


Figure 6.2: Cell site, regional and edge clouds

scalable, exact solution that can serve as a performance benchmark. In this study, we aim to fill this gap by focusing on the dedicated protection and shared protection schemes, which offers the highest reliability among the available options. Duong *et al.* [24] previously introduced a scalable column generation algorithm for dedicated protection, but under a simplified model where only one O-CU connects to each NRT_RIC. In contrast, we consider the more general and realistic Scenario B from the official O-RAN Alliance specification [61], where the connectivity structure is more complex.

To address these challenges, our work offers the following contributions:

- Introduce a powerful new mathematical formulation for SFC placement in O-RAN, seamlessly integrating both dedicated and shared protection schemes through decomposition modeling.
- Design a scalable and effective column generation, enabling the solution of large-scale instances with up to 620 servers and 590 VNFs—well beyond the reach of existing methods.
- Provide ILP heuristics specifically designed to address the nonlinearities present in the proposed models.

- Compare the proposed model and algorithm with a state-of-the-art Binary Integer Programming (BIP) model for O-RAN protection [80], and demonstrate that our approach achieves near-optimal performance on significantly larger datasets.

- Establish the proposed solution as a robust benchmark for future research, supporting the investigation of additional protection schemes and optimization variants.

The rest of the chapter is organized as follows. Section 6.2 surveys related work. Section 6.3 introduces a formal problem statement and the notations of the study. Section 6.4 details the problem in terms of mathematical formulation using decomposition modeling. Section 6.5 describes our column generation algorithm to solve the proposed decomposition model. Section 6.6 presents a comparison and demonstrates that our model provides tighter bounds than state-of-the-art ILP model. Section 6.7 presents the data generation, model validation and experimental results. The last section concludes the chapter and discusses our future work.

6.2 Related Work

Reliable and efficient VNF deployment in O-RAN networks remains a critical challenge in the 5G/6G era. To address this, researchers have proposed various solutions, including mathematical models, heuristics, and machine learning methods, that aim to ensure availability, scalability, and compliance with strict service requirements such as low latency. In the following section, we summarize key contributions in each category, highlighting their strengths and limitations.

Our problem formulation builds upon the foundational work presented in [25, 80], which address the reliability of individual VNFs. However, unlike these studies, our approach emphasizes end-to-end service reliability from the NRT_RIC to the O-DU, capturing the full scope of O-RAN service chains. Hmaity *et al.* [40] proposed an ILP model integrating both link and node protection with the goal of minimizing server usage. Despite its comprehensive formulation, the model suffers from scalability limitations, requiring approximately 13 hours to solve an instance with fewer than 20 servers and five SFC requests. In addition to protecting computing components,

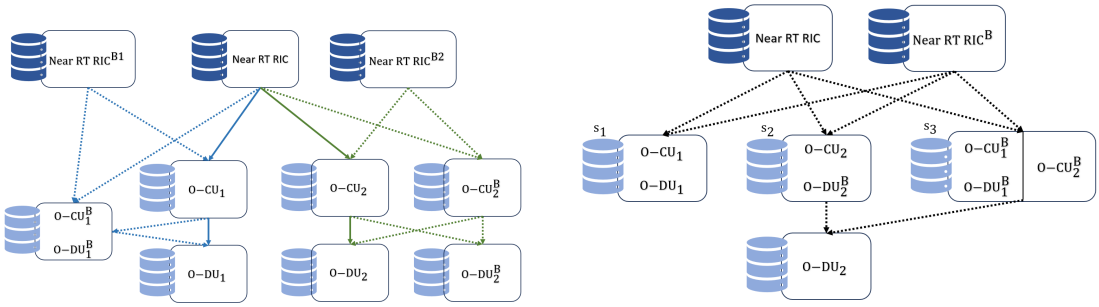
Tomassilli *et al.* [84] and Duong *et al.* [23] focused on ensuring reliable communication paths between VNFs in the core or logical network using decomposition-based models to design dedicated backup paths. Qu *et al.* [67] proposed a hybrid approach that integrates a greedy algorithm with a MILP model to address provisioning while considering delay and reliability requirements. In a subsequent work [66], the authors refined their method by developing an enhanced greedy algorithm, which achieved higher solution quality through improved resource allocation.

In addition to mathematical models, several heuristic approaches have been proposed. Fan *et al.* [28] introduced an online heuristic for joint-protection in SFC mapping, while Zhang *et al.* [99] tackled the challenge of heterogeneous VNF resource requirements beyond compute capacity, albeit without considering availability or delay constraints. Askari *et al.* [9] focused on dynamic SFC provisioning with node and link protection, but similarly neglected delay requirements, a key factor in URLLC and other 5G/6G applications.

Beyond traditional mathematical and heuristic methods, machine learning-based approaches have also emerged to tackle VNFs placement challenges. In [81], Tanim *et al.* proposed a Reinforcement Learning-based approach to address VNF placement in O-RAN while aiming to maximize service availability. Although the method demonstrates fast runtime, it does not provide insights into how close the results are to an optimal solution. Park *et al.* [63] introduced a Graph Neural Network-based solution for online SFC placement. Their model achieved approximately 90% of the optimal performance; however, the presence of imbalanced data suggests there is still room for improvement. Wu *et al.* [90] presented a probability-based approach that selects VNF placements by updating a probability vector in combination with heuristic techniques. While the optimality gap remains unclear, this method outperformed comparable solutions based on Genetic Algorithms and Tabu Search.

6.3 O-RAN Protection Problem Statement and Notations

Multiple O-RAN architecture scenarios have been proposed. In this study, we use Scenario B in the O-RAN specification [61]. Scenario B is the initial use case selected by the O-RAN Alliance. It enables the near-RT RICs to have a broader view of the



(a) Dedicated protection (b) Shared protection

Figure 6.3: Reliable O-RAN

network while maintaining low latency between O-CUs and O-DUs. In this scenario, each near-RT RIC unit can be linked with multiple O-CUs, while each O-CU is uniquely linked with one O-DU.

We now introduce the various parameters to represent the given information about network entities and their relationships.

VNF and SFC. Let V (indexed by v) be the set of all VNFs (also called O-RAN units) that are being installed in the network. V is partitioned into three subsets: $V = V^{\text{NRT\_RIC}} \cup V^{\text{O-CU}} \cup V^{\text{O-DU}}$, where $V^{\text{NRT\_RIC}}$, $V^{\text{O-CU}}$, and $V^{\text{O-DU}}$ are sets of near-RT RIC (NRT_RIC), O-CU, and O-DU VNFs, respectively. Each of these subsets is further decomposed into primary and backup VNFs: $V^{\square} = V^{\square,\text{P}} \cup V^{\square,\text{B}}$, where $\square \in \{\text{NRT\_RIC}, \text{O-CU}, \text{O-DU}\}$, with V^{P}, V^{B} being the set of primary and backup VNFs, respectively.

Each SFC request is characterized by a primary tuple (r, c, d) where r, c, d are primary near-RT RIC, O-CU, and O-DU. Similarly, the backup SFC is characterized by $r^{\text{B}}, c^{\text{B}}, d^{\text{B}}$, in which some or all backup VNFs are used depending on the failed server and on the VNF hosting on the servers. For example, in Figure 6.4, if server s_1 fails then the O-CU on s_1 will be substituted by O-CU$^{\text{B}}$ on s_3. We denote $V^{rc} = \{r, c, r^{\text{B}}, c^{\text{B}}\}$ and $V^{cd} = \{c, d, c^{\text{B}}, d^{\text{B}}\}$.

VNF dependency. We define the dependencies of a VNF v as a set of connected VNFs that have a lower hierarchy than v, denoted as V_v^{DEP}. For instance, O-CU is a dependency of NRT_RIC, O-DU is a dependency of O-CU. Given a SFC (r, c, d) and its backup $(r^{\text{B}}, c^{\text{B}}, d^{\text{B}})$, then the dependencies of c is $V_c^{\text{DEP}} = \{d, d^{\text{B}}\}$, and it is the same for the backup of c, $V_{c^{\text{B}}}^{\text{DEP}} = \{d, d^{\text{B}}\}$.

Server characteristic. Each VNF or O-RAN unit is hosted on a server, which may be prone to failure. Let S be the set of all available servers, potentially located in different data centers (or clouds). Note that a given server can host both primary and backup VNFs. Let $S = S^{\text{R}} \cup S^{\text{E}}$ be the set of all servers, regional and edge servers, respectively. Call S_v the set of servers suitable for VNF v, then for a NRT_RIC r, $S_r = S^{\text{R}}$, for an O-CU c and an O-DU d, $S_c = S_d = S^{\text{E}}$. Additionally, each server s has limited resources, symbolized by $\text{CAP}_s^{\square}$ where $\square \in \{\text{CPU}, \text{RAM}\}$, and resources required by a VNF v are presented by $\tau_v^{\square}$. Furthermore, we denote by $\delta_{ss'}$ the delay between server s and s'.

Delay requirements. In a SFC request (r, c, d), connectivity between 2 VNFs v, v', whether primary or backup, must satisfy the delay requirement. Specifically, the delay $\delta_{ss'}^{\text{server}}$ of the servers s, s' where v, v' are placed must not exceed the delay threshold $\delta_{vv'}^{\text{L}}$.

Problem statement: SFC protection under single server failure. For each SFC (r, c, d), allocate servers for r, c, d and to their respective backups $r^{\text{B}}, c^{\text{B}}, d^{\text{B}}$, such that in the event of a single-server failure, the backup VNFs can replace the primary ones installed on the failed server. This assignment must respect both the delay constraints and the server capacity limitations.

Protection scheme. We present two schemes: dedicated protection and shared protection, see Figures 6.3a and 6.3b. In the dedicated protection scheme, each backup VNF has its own dedicated resources reserved whether failure occurs or not. In contrast, the shared protection scheme reserves resources for backup VNFs only when a failure (one server at a time) occurs. It is assumed that there is enough time to repair one failure before another occurs. For example, in Figure 6.3b, the reserved resources for backup VNFs on server s_3 only need to be sufficient to run O-CU_1^{B} and O-DU_1^{B} if s_1 fails, or to run O-CU_2^{B} if s_2 fails (i.e., reserved resource is the maximum of $\tau_{\text{O-CU}_1}^{\square} + \tau_{\text{O-DU}_1}^{\square}$, and $\tau_{\text{O-CU}_2}^{\square}$). Unlike shared protection, in the case of dedicated protection, server s_3 must have enough resources to run all O-CU_1^{B}, O-DU_1^{B}, and O-CU_2^{B} (i.e., reserved resource is the summation of $\tau_{\text{O-CU}_1}^{\square}, \tau_{\text{O-DU}_1}^{\square}$, and $\tau_{\text{O-CU}_2}^{\square}$).

6.4 Mathematical Models

In order to cope with the combinatorial aspect of the O-RAN protection problem, we propose to investigate decomposition mathematical models. To this end, we introduce the concept of SFC configuration.

6.4.1 Decomposition Scheme: SFC Configurations

We first define an SFC configuration as follows. It consists of an O-RAN SFC (NRT_RIC, O-CU, O-DU) and a complete backup SFC$^{\text{B}}$ (NRT_RIC$^{\text{B}}$ O-CU$^{\text{B}}$, O-DU$^{\text{B}}$) with a proper server assignment for each of the VNF components, i.e., a primary VNF cannot be hosted on the same server as its backup, see Figure 6.4 for an illustration. Note however, that any combination of backup and primary VNFs can be used, e.g., NRT_RIC$^{\text{B}}$, O-CU, O-DU, to answer to any single server failures. Let Γ be the set of all

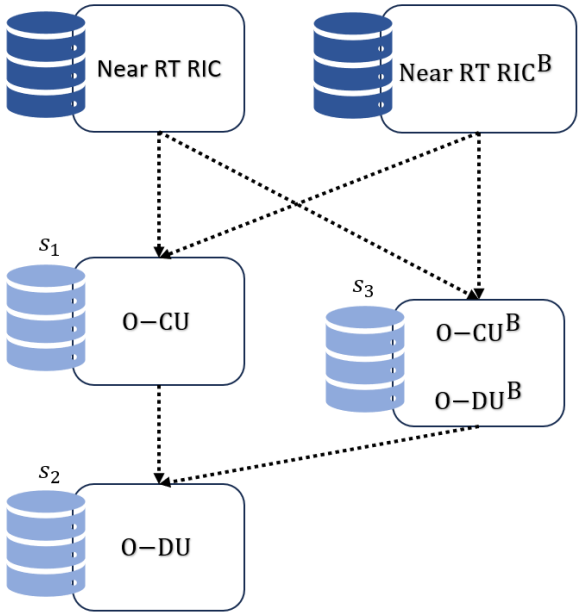


Figure 6.4: An SFC Configuration

SFC configurations. We have:

$$\Gamma = \bigcup_{r \in V^{\text{NRT\_RIC}}, c \in V_r^{\text{DEP}}} \Gamma_{rc},$$

where Γ_{rc} is the set of all SFC configurations with r and c as primary NRT_RIC and O-CU, respectively. Note that we do need to add O-DU d into the notation due to the

one-to-one relation due to Scenario B of O-RAN (see Section 6.3).

A configuration $\gamma \in \Gamma_{rc}$ is characterized by:

- A primary SFC = (NRT_RIC (r for short), O-CU (c for short), O-DU (d for short)). Let parameter $\theta_{vs}^{\gamma} = 1$ if in configuration γ, O-RAN unit v is installed on server s, 0 otherwise.
- A backup SFC$^{\text{B}}$ = (NRT_RIC$^{\text{B}}$ (r^{B}), O-CU$^{\text{B}}$ (c^{B}), O-DU$^{\text{B}}$ (d^{B}))
- A server allocation of the VNFs satisfying backup constraints (a primary VNF and its backup cannot be hosted on the same server)
- Inter VNF delay threshold requirements.

6.4.2 Variables

Mathematical models that will be developed in Sections 6.4.5 and 6.4.5 share the following sets of variables.

- z_{γ}: decision variable that is 1 if SFC configuration $\gamma \in \Gamma$ is selected, 0 otherwise.
- x_{rs}: decision variable that is 1 if primary NRT_RIC r is installed on server s, 0 otherwise.
- $y_s^{\text{RAM,B}}$ and $y_s^{\text{CPU,B}}$: required CPU and RAM resources on server s for all types of O-RAN backup units.

6.4.3 Parameters

Parameters are as follows:

- θ_{vs}^{γ}: equals 1 if VNF v is installed on server s in SFC configuration γ, 0 otherwise.
- τ_v^{CPU}, τ_v^{RAM}: required processing and memory resources of unit $v \in V$.
- $\text{CAP}_s^{\text{CPU}}$, $\text{CAP}_s^{\text{RAM}}$: processing and memory capacities of server $s \in S$.

6.4.4 Objectives

Our study is directed towards addressing two primary objectives: the minimization of total delay and the maximization of availability for the components within O-RAN. Each objective is expressed as a function of the decision variables z^γ, whether a given SFC configuration is selected or not.

Objective 1: Minimize total delay

The purpose of this function is to minimize the overall delay between NRT_RICs, O-CUs, and O-DUs. Denote the total delay of configuration γ as:

$$\delta_\gamma = \sum_{v\in\{r,r^{\mathrm{B}}\}} \sum_{v'\in\{c,c^{\mathrm{B}}\}} \delta_{vv'} + \sum_{v\in\{c,c^{\mathrm{B}}\}} \sum_{v'\in\{d,d^{\mathrm{B}}\}} \delta_{vv'}. \tag{6.1}$$

The first objective function will be formulated as:

$$\min \quad \mathrm{OBJ}^{\mathrm{DELAY}} = \sum_{\gamma\in\Gamma} \delta_\gamma z_\gamma. \tag{6.2}$$

Objective 2: Maximize availability

For a given configuration γ and a primary VNF v belong to γ, let a_v be the availability that the primary VNF v or its backup component v^{B} is functioning normally. If in γ, v is hosted on server s and v^{B} is hosted on server s^{B}, the overall availability of v in γ is:

$$a_v^\gamma = 1 - (1 - a_{vs})(1 - a_{v^{\mathrm{B}},s'}), \tag{6.3}$$

where a_{vs} (resp. $a_{v^{\mathrm{B}},s'}$) is the availability of VNF v (resp. v^{B}, the backup component of v) when hosted on server s (resp. s').

For a given 2-tree configuration γ and its corresponding SFC (r, c, d), the average availability of γ is computed as:

$$a^\gamma = a_r^\gamma a_c^\gamma a_d^\gamma. \tag{6.4}$$

The computational resources under consideration in this study encompass CPU and RAM. Recognizing the higher cost associated with CPU resources compared to RAM, our initial focus is on minimizing CPU utilization. These objectives are formally articulated through the following mathematical expressions:

$$\max \quad \text{OBJ}^{\text{AVAIL}} = \sum_{\gamma \in \Gamma} a^{\gamma} z_{\gamma}. \tag{6.5}$$

While the delay objective (6.2) is a simple linear expression, the availbility one (6.5) introduces complexity. Upon decomposition in the sub-problem, a^{γ} become a quadratic expression as in (6.4).

6.4.5 Constraints

We investigate two protection schemes: dedicated protection and shared protection. The upcoming sections will present their mathematical modeling, i.e., their set of constraints.

Dedicated Protection

We now describe the model for an O-RAN deployment with a dedicated protection (DP) scheme.

The mathematical model for O-RAN dedicated protection (DP_M) can be written as follows.

$$\sum_{\gamma \in \Gamma} \sum_{s \in S} \theta_{cs}^{\gamma} z_{\gamma} = 1 \qquad c \in V^{\text{P,O-CU}} \tag{6.6}$$

$$\sum_{\gamma \in \Gamma} \theta_{rs}^{\gamma} z_{\gamma} \leq n_r^{\text{O-CU}} x_{rs} \qquad r \in V^{\text{NRT\_RIC}}, s \in S \tag{6.7}$$

$$\sum_{s \in S} x_{rs} \leq 1 \qquad r \in V^{\text{P,NRT\_RIC}} \tag{6.8}$$

$$\sum_{\gamma \in \Gamma} \sum_{v \in V} \tau_v^{\square} \, \theta_{vs}^{\gamma} z_{\gamma} \leq \text{CAP}_s^{\square} \quad s \in S^{\text{E}}, \square \in \{\text{CPU}, \text{RAM}\} \tag{6.9}$$

$$\sum_{v \in V^{\text{NRT\_RIC}}} \tau_v^{\square} \, x_{vs} \leq \text{CAP}_s^{\square} \quad s \in S^{\text{R}}, \square \in \{\text{CPU}, \text{RAM}\} \tag{6.10}$$

$$z_{\gamma} \in \{0, 1\} \qquad \gamma \in \Gamma \tag{6.11}$$

$$x_{rs} \in \{0, 1\} \qquad r \in V^{\text{NRT\_RIC}}. \tag{6.12}$$

Constraints (6.6) ensure that every primary O-CUinstance is installed at exactly one location. Due to the structure of the selected SFC configuration γ , the unique assignment of O-CU c also guarantees the unique placement of its backup as well as its dependencies primary and backup O-DU.

Constraints (6.7) ensure that x_{rs} is one if there is one configuration of $r \in V^{\text{NRT\_RIC}}$ is selected and r is installed at $s \in S$ in the configuration.

It also ensures that each NRT_RIC r is connected to at most $n_r^{\text{O-CU}}$ primary O-CUs. Note that backup O-CUs are only activated when required, and therefore the limit remains the same whether under normal operation or when one failure occurs.

Constraints (6.8) ensure that a NRT_RIC $r \in V^{\text{P,NRT\_RIC}}$ is installed at no more than one server. Note that this set of constraint is not applied for the backup NRT_RIC since the backup one may be installed on multiple servers.

Constraints (6.9) and (6.10) make sure that CPU and memory resources are not exceeded on the edge and regional servers, respectively.

The remaining sets of constraints specify the domains of the variables.

Shared Protection

The shared protection (SP) model uses the same decomposition as the dedicated protection model except for the constraints governing the sharing of the protection units.

Constraints can be written as follows:

$$\sum_{\gamma \in \Gamma} \sum_{s \in S_v} \theta_{cs}^{\gamma} z_{\gamma} = 1 \qquad c \in V^{\text{P,O-CU}} \tag{6.13}$$

$$\sum_{\gamma \in \Gamma} \theta_{rs}^{\gamma} z_{\gamma} \leq n_r^{\text{O-CU}} x_{rs} \qquad r \in V^{\text{NRT\_RIC}}, s \in S \tag{6.14}$$

$$\sum_{s \in S} x_{rs} \leq 1 \qquad r \in V^{\text{P,NRT\_RIC}} \tag{6.15}$$

$$\sum_{\gamma \in \Gamma} \sum_{v \in V} \tau_v^{\square} \theta_{vs}^{\gamma} \theta_{v^{\text{B}}s'}^{\gamma} z_{\gamma} \leq y_{s'}^{\square,\text{B}} \qquad s, s' \in S^{\text{E}}, s \neq s', \square \in \{\text{CPU}, \text{RAM}\} \tag{6.16}$$

$$\sum_{\gamma \in \Gamma} \sum_{v \in V^{\text{P}}} \tau_v^{\square} \theta_{vs}^{\gamma} z^{\gamma} + y_s^{\square,\text{B}} \leq \text{CAP}_s^{\square} \quad s \in S^{\text{E}}, \square \in \{\text{CPU}, \text{RAM}\} \tag{6.17}$$

$$\sum_{r \in V^{\text{NRT\_RIC}}} \tau_r^{\square} x_{rs} x_{r^{\text{B}}s'} \leq y_{s'}^{\square,\text{B}} \qquad s, s' \in S^{\text{R}}, s \neq s', \square \in \{\text{CPU}, \text{RAM}\} \tag{6.18}$$

$$\sum_{r \in V^{\text{P,NRT\_RIC}}} \tau_r^{\square} x_{rs} + y_s^{\square,\text{B}} \leq \text{CAP}_s^{\square} \qquad s \in S^{\text{R}}, \square \in \{\text{CPU}, \text{RAM}\} \tag{6.19}$$

$$y_s^{\text{CPU,B}} \geq 0, y_s^{\text{RAM,B}} \geq 0 \qquad s \in S \tag{6.20}$$

$$z_{\gamma} \in \{0, 1\} \qquad \gamma \in \Gamma \tag{6.21}$$

$$x_{rs} \in \{0, 1\} \qquad r \in V^{\text{NRT\_RIC}}, s \in S^{\text{R}}. \tag{6.22}$$

Constraints (6.13) are indentical to Constraints (6.6) for dedicated protection.
Constraints (6.14) ensure that $x_{rs} = 1$ if one configuration of $r \in V^{\text{P,NRT\_RIC}}$ is selected and r is installed at $s \in S$ in the configuration. It also ensures that each NRT_RIC r connected with at most $n_r^{\text{O-CU}}$ O-CUs, as in Constraints (6.7).
Constraints (6.15) are identical to Constraints (6.8) for dedicated protection.
Constraints (6.16) and (6.17) together ensure that the CPU and RAM capacity resources at the edge servers are not exceeded. In (6.16), $y_{s'}^{\text{CPU,B}}$ defines the amount of resources needed for the backup server in case that one of the other servers failed. Assuming s is the failed server, $y_{s'}^{\text{CPU,B}}$ should be big enough to run all the VNF backups on s that are also installed on s'. In other words,

$$y_{s'}^{\text{CPU,B}} \geq \max_{s \in S^{\text{E}}} \sum_{\gamma \in \Gamma} \sum_{v \in V} \tau_v^{\text{CPU}} \theta_{v,s}^{\gamma} \theta_{v^{\text{B}},s'}^{\gamma} z_{\gamma}.$$

Constraints (6.17) express that on server s, the total amount of resources needed to run all primary VNFs and the resource reserved for the backup units must not exceed the capacity $\text{CAP}_s^{\text{CPU}}$ of s.
Justifications are similar for the RAM resources.
Constraints (6.18) and (6.19) together ensure that the CPU and RAM capacity resources at the regional servers are not exceeded. Constraints (6.18) ensure that, if server s fails, backup resources on server s' are sufficient in order that all NRT_RICs installed on s can have enough backup resources available on s'. In other words

$$y_{s'}^{\square,\text{B}} = \max_{s \in S^{\text{R}}} \sum_{r \in V^{\text{NRT\_RIC}}} \tau_r^{\square} x_{rs} x_{r^{\text{B}}s'},$$

which is expressed by (6.18). Constraints (6.19) indicate that both the resource needed to run the primary NRT_RIC and the reserved resource for backup together must not superior than the capacity $\text{CAP}_s^{\square}$ on server s.

Linearize Shared Protection

The non-linearity of (6.18) can be resolved by introducing a new set of variables

$$\omega_{s,s'}^{r} = x_{rs} x_{r^{\text{B}}s'},$$

with the following set of constraints:

$$\begin{aligned}
&\omega^r_{s,s'} && \leq x_{rs} \\
&\omega^r_{s,s'} && \leq x_{r^{\mathrm{B}}s'} \\
&\omega^r_{s,s'} && \geq x_{rs} + x_{r^{\mathrm{B}}s'} - 1 \\
&\omega^r_{s,s'} \in \{0,1\}. &&
\end{aligned} \tag{6.23}$$

Constraints (6.18) become:

$$\sum_{r \in V^{\mathrm{NRT\_RIC}}} \tau^{\square}_r \omega^r_{s,s'} \leq y^{\square,\mathrm{B}}_{s'}. \tag{6.24}$$

The sets of constraints (6.20), (6.21), (6.22), and (6.23) specify the domains of the variables.

6.5 Solution Scheme

In the next part, we present the column generation-based solution framework to solve the SFC placement problem. We elaborate on how the solution process adapts to different objectives and protection schemes and detail the techniques used to manage problem-specific nonlinearities.

6.5.1 Column Generation Framework

Column generation (CG) is an efficient algorithm used for solving large linear programming problems by decomposing a large problem (i.e., master problem) into multiple sub-problems (i.e., pricing problem). Desrosiers *et al.* [21] provide a comprehensive introduction to the CG methodology and its various applications. In the context of this study, the so-called master problems, i.e., problems (6.6) - (6.12) and (6.13) - (6.22) correspond to the selection of the best configuration, while the sub-problems are associated with the generation of "augmenting" configurations, i.e., dynamic generation of configurations that improve the value of the objective of the master problem.

The CG process, as described in Figure 6.5, starts with an initial set of SFC configurations and the solution process alternates between the master problem and

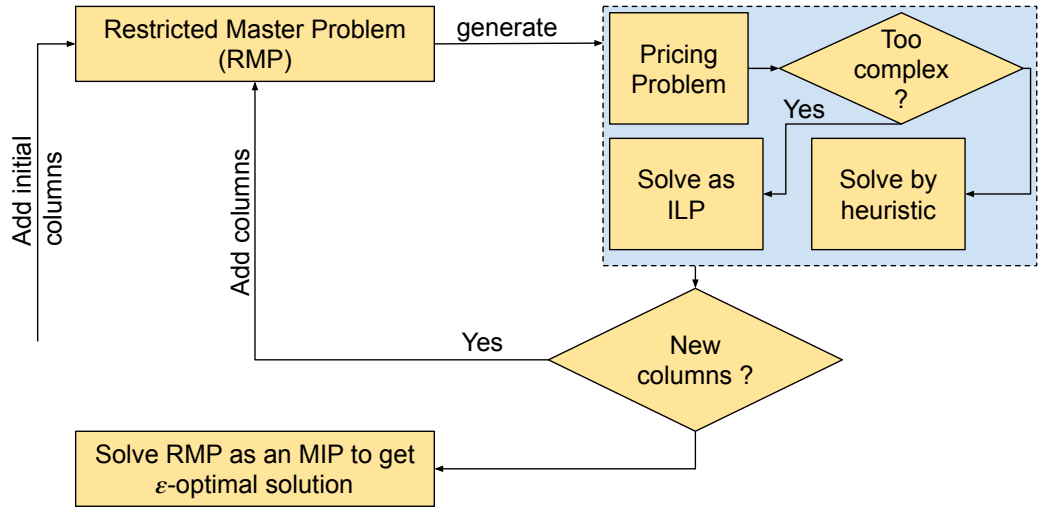
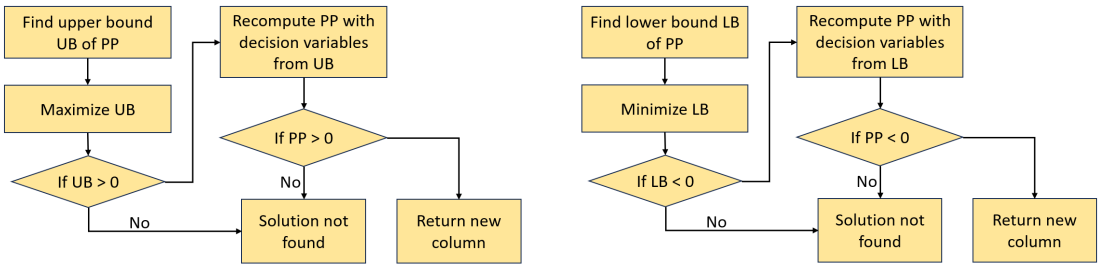


Figure 6.5: General CG flowchart



(a) Maximization (b) Minimization

Figure 6.6: Heuristic flowchart

the pricing problem, until the optimality condition is satisfied. The restricted master problem (RMP), made of a subset of configurations, is first relaxed (continuous relaxation) and solved. Based on the solution, a separate pricing problem is formulated for each individual SFC, resulting in a total number of pricing problems equal to the number of SFCs. Each subproblem generates a new configuration; if its reduced cost is negative (in case of minimization) or positive (in case of maximization), it is added to the RMP as it is then an improving one. This iterative process continues until no improving configurations are found. Finally, the RMP is solved as an MIP to obtain an ϵ-optimal solution.

In this study, the pricing objective functions are non-linear, which makes using a MILP solver computationally expensive. Therefore, we introduce a heuristic approach to address this issue. Figures (6.6a) and (6.6b) illustrate the heuristic process used

to solve pricing problems. Additionally, the upper and lower bounds obtained from the process can be used to estimate the Lagrangian bound.

6.5.2 Pricing with Objective 1: Dedicated Protection

The objective of a pricing problem is to generate an SFC configuration with the selection of servers for hosting specified VNFs denoted as r, c, and d. Different objectives will have distinct pricing problems accordingly. With a slight abuse of notation, we use the variable θ_{vs} to indicate whether VNF v is placed on server s, where $\theta_{vs} = 1$ if v is installed on s, 0 otherwise.

Exact Solution

$$\min \quad \text{PP}^{\text{DP\_DELAY}} = \delta_\gamma - D_{\text{DP}}, \tag{6.25}$$

subject to:

$$\sum_{s \in S} \theta_{vs} = 1 \qquad v \in V \tag{6.26}$$

$$\delta^{\text{server}}_{ss'}(\theta_{vs} + \theta_{v's'} - 1) \leq \delta^{\text{L}}_{vv'} \quad v \in V^{\text{P}}, v' \in V^{\text{DEP}}_v s \in S_v, s' \in S_{v'} \tag{6.27}$$

$$\theta_{vs} + \theta_{v^{\text{B}}s} \leq 1 \qquad v \in V^{\text{P}}, s \in S_v \tag{6.28}$$

$$\theta_{vs} \in \{0, 1\} \qquad v \in V, s \in S_v, \tag{6.29}$$

where D_{DP} is the duality term in dedicated protection scheme, defined by the following formula:

$$\begin{aligned} D_{\text{DP}} = u_c^{(6.6)} \sum_{s \in S^{\text{E}}} \theta_{cs} - \sum_{v \in \{r, r'\}} \sum_{s \in S} u_{v,s}^{(6.7)} \theta_{vs} \\ - \sum_{v \in V^{cd}} \sum_{s \in S_v} \sum_{\square \in \{\text{CPU,RAM}\}} u_s^{(6.9)} \tau_v^{\square} \theta_{vs}. \end{aligned} \tag{6.30}$$

Constraints (6.26) ensure that each VNF v (whether primary and backup) is installed on a unique server. Constraints (6.27) confirm that if v is installed on s and its dependency v' is installed on s', then the latency $\delta^{\text{server}}_{ss'}$ between server s and s' does not exceed the latency limit $\delta^{\text{L}}_{vv'}$ between v and v'. Constraints (6.28) make sure that a primary VNF v and its backup v^{B} are not installed on the same server.

Resolve the non-linearity

The term δ_γ, as defined in (6.1), depends on the delay $\delta_{v,v'}$ between each connected pair of VNFs v and v', which we now detail in the following computation:

$$\delta_{vv'} = \sum_{s \in S_v} \sum_{s' \in S_{v'}} \delta^{\text{server}}_{ss'} \theta_{vs} \theta_{v's'}. \tag{6.31}$$

To linearize $\delta_{vv'}$, we introduced $\kappa^{vv'}_{ss'} = \theta_{vs}\theta_{v's'}$ with an additional set of linearization constraints:

$$\begin{aligned} \kappa^{vv'}_{ss'} &\leq \theta_{vs} \\ \kappa^{vv'}_{ss'} &\leq \theta_{v's'} \\ \kappa^{vv'}_{ss'} &\geq \theta_{vs} + \theta_{v's'} - 1. \end{aligned}$$

Heuristic

In Section 6.5.2 we proposed a linearization of the delay of a configuration; however, it introduces additional $8|S|^2$ new variables and $24|S|^2$ new constraints, which does not scale well when the number of servers increases.

To address this issue, we follow the procedure described in Figure 6.6b, beginning with the computation of a lower bound for $\text{PP}^{\text{DP\_DELAY}}$. From (6.31), we have:

$$\begin{aligned} \delta_{rc} &\geq \sum_{s \in S^{\text{R}}} \sum_{s' \in S^{\text{E}}} \min_{i \in S^{\text{E}}} \delta^{\text{server}}_{si} \theta_{rs} \theta_{cs'} \\ &\geq \sum_{s \in S^{\text{R}}} \min_{i \in S^{\text{E}}} \delta^{\text{server}}_{si} \theta_{rs} \underbrace{\sum_{s' \in S^{\text{E}}} \theta_{cs'}}_{=1 \text{ according to (6.26)}} \\ &\geq \sum_{s \in S^{\text{R}}} \min_{i \in S^{\text{E}}} \delta^{\text{server}}_{si} \theta_{rs}. \end{aligned} \tag{6.32}$$

Similarly we also have:

$$\delta_{rc} \geq \sum_{s' \in S^{\text{E}}} \min_{i \in S^{\text{R}}} \delta^{\text{server}}_{is'} \theta_{cs'}. \tag{6.33}$$

Combining both (6.32) and (6.33) we have:

$$\delta_{rc} \geq \frac{1}{2}\left(\sum_{s\in S^{\mathrm{R}}} \min_{i\in S^{\mathrm{E}}} \delta^{\mathrm{server}}_{si}\theta_{rs} + \sum_{s'\in S^{\mathrm{E}}} \min_{i\in S^{\mathrm{R}}} \delta^{\mathrm{server}}_{is'}\theta_{cs'}\right). \tag{6.34}$$

Then the overall delay between primary and backup NRT_RIC (r, r^{B}) and O-CU (c, c^{B}) can be expressed as:

$$\begin{aligned}\delta_{rc} + \delta_{rc^{\mathrm{B}}} + \delta_{r^{\mathrm{B}}c} + \delta_{r^{\mathrm{B}}c^{\mathrm{B}}} \geq &\sum_{s\in S^{\mathrm{R}}} \min_{i\in S^{\mathrm{E}}} \delta^{\mathrm{server}}_{si}\theta_{rs} \\ &+ \sum_{s\in S^{\mathrm{R}}} \min_{i\in S^{\mathrm{E}}} \delta^{\mathrm{server}}_{si}\theta_{r^{\mathrm{B}}s} + \sum_{s\in S^{\mathrm{E}}} \min_{i\in S^{\mathrm{R}}} \delta^{\mathrm{server}}_{is}\theta_{cs} \\ &+ \sum_{s\in S^{\mathrm{E}}} \min_{i\in S^{\mathrm{R}}} \delta^{\mathrm{server}}_{is}\theta_{c^{\mathrm{B}}s}.\end{aligned} \tag{6.35}$$

We apply the same logic for the delay between O-CUs and O-DUs, note that now there're cases in which both O-CU and O-DU can be on the same sever and the delay can be 0.

$$\begin{aligned}\delta_{cd} &\geq \sum_{s\in S^{\mathrm{E}}} \sum_{\substack{s'\in S^{\mathrm{E}}\\ s'\neq s}} \min_{\substack{i\in S^{\mathrm{E}}\\ i\neq s}} \delta^{\mathrm{server}}_{si}\theta_{cs}\theta_{ds'} + \underbrace{\sum_{s\in S^{\mathrm{E}}} \delta^{\mathrm{server}}_{ss}\theta_{cs}\theta_{ds}}_{=0 \text{ because } \delta^{\mathrm{server}}_{ss}=0} \\ &\geq \sum_{s\in S^{\mathrm{E}}} \min_{\substack{i\in S^{\mathrm{E}}\\ i\neq s}} \delta^{\mathrm{server}}_{si}\theta_{cs} \sum_{\substack{s'\in S^{\mathrm{E}}\\ s'\neq s}} \theta_{ds'} \\ &\geq \sum_{s\in S^{\mathrm{E}}} \min_{\substack{i\in S^{\mathrm{E}}\\ i\neq s}} \delta^{\mathrm{server}}_{si}\theta_{cs} \left(\underbrace{\sum_{s'\in S^{\mathrm{E}}} \theta_{ds'}}_{=1} - \theta_{ds}\right) \\ &\geq \sum_{s\in S^{\mathrm{E}}} \min_{\substack{i\in S^{\mathrm{E}}\\ i\neq s}} \delta^{\mathrm{server}}_{si}\theta_{cs} - \sum_{s\in S^{\mathrm{E}}} \min_{\substack{i\in S^{\mathrm{E}}\\ i\neq s}} \delta^{\mathrm{server}}_{si}\theta_{cs}\theta_{ds}.\end{aligned} \tag{6.36}$$

Similarly we also have:

$$\delta_{cd} \geq \sum_{s\in S^{\mathrm{E}}} \min_{\substack{i\in S^{\mathrm{E}}\\ i\neq s}} \delta^{\mathrm{server}}_{si}\theta_{ds} - \sum_{s\in S^{\mathrm{E}}} \min_{\substack{i\in S^{\mathrm{E}}\\ i\neq s}} \delta^{\mathrm{server}}_{si}\theta_{cs}\theta_{ds}. \tag{6.37}$$

Combining (6.36) and (6.37) we have:

$$\delta_{cd} \geq \frac{1}{2}\left(\sum_{s\in S^{\mathrm{E}}} \min_{\substack{i\in S^{\mathrm{E}}\\ i\neq s}} \delta^{\mathrm{server}}_{si}\theta_{cs} + \sum_{s\in S^{\mathrm{E}}} \min_{\substack{i\in S^{\mathrm{E}}\\ i\neq s}} \delta^{\mathrm{server}}_{si}\theta_{ds}\right) - \sum_{s\in S^{\mathrm{E}}} \min_{\substack{i\in S^{\mathrm{E}}\\ i\neq s}} \delta^{\mathrm{server}}_{si}\theta_{cs}\theta_{ds}. \tag{6.38}$$

The overall delay between primary and back up O-CU (c, c^{B}) and O-DU (d, d^{B}) can be written as lower bound as follows:

$$\begin{aligned}\delta_{cd} + \delta_{cd^{\mathrm{B}}} + \delta_{c^{\mathrm{B}}d} + \delta_{c^{\mathrm{B}}d^{\mathrm{B}}} \geq & \sum_{s\in S^{\mathrm{E}}} \min_{\substack{i\in S^{\mathrm{E}}\\ i\neq s}} \delta^{\mathrm{server}}_{si}\theta_{cs} \\ & + \sum_{s\in S^{\mathrm{E}}} \min_{\substack{i\in S^{\mathrm{E}}\\ i\neq s}} \delta^{\mathrm{server}}_{si}\theta_{c^{\mathrm{B}}s} + \sum_{s\in S^{\mathrm{E}}} \min_{\substack{i\in S^{\mathrm{E}}\\ i\neq s}} \delta^{\mathrm{server}}_{si}\theta_{ds} \\ & + \sum_{s\in S^{\mathrm{E}}} \min_{\substack{i\in S^{\mathrm{E}}\\ i\neq s}} \delta^{\mathrm{server}}_{si}\theta_{d^{\mathrm{B}}s} \\ & - \sum_{s\in S^{\mathrm{E}}} \min_{\substack{i\in S^{\mathrm{E}}\\ i\neq s}} \delta^{\mathrm{server}}_{si}(\theta_{cs}\theta_{ds} + \theta_{cs}\theta_{d^{\mathrm{B}}s} + \theta_{c^{\mathrm{B}}s}\theta_{ds} + \theta_{c^{\mathrm{B}}s}\theta_{d^{\mathrm{B}}s}).\end{aligned} \tag{6.39}$$

The nonlinear term of the last term of (6.39) can be rewritten as:

$$\begin{aligned}\theta_{cs}\theta_{ds} + \theta_{cs}\theta_{d^{\mathrm{B}}s} + \theta_{c^{\mathrm{B}}s}\theta_{ds} + \theta_{c^{\mathrm{B}}s}\theta_{d^{\mathrm{B}}s} & \\ = (\theta_{cs} + \theta_{c^{\mathrm{B}}s})(\theta_{ds} + \theta_{d^{\mathrm{B}}s}) = \omega_s.\end{aligned} \tag{6.40}$$

Thanks to constraints (6.28), we can linearize (6.40) by:

$$\begin{aligned}\omega_s &\leq \theta_{cs} + \theta_{c^{\mathrm{B}}s} \\ \omega_s &\leq \theta_{ds} + \theta_{d^{\mathrm{B}}s} \\ \omega_s &\geq \theta_{cs} + \theta_{c^{\mathrm{B}}s} + \theta_{ds} + \theta_{d^{\mathrm{B}}s} - 1 \\ \omega_s &\leq \frac{1}{2}(\theta_{cs} + \theta_{c^{\mathrm{B}}s} + \theta_{ds} + \theta_{d^{\mathrm{B}}s}).\end{aligned}$$

The linear lower bound of δ_γ becomes:

$$\begin{aligned}
\delta_\gamma \geq & \sum_{s \in S^{\mathrm{R}}} \min_{i \in S^{\mathrm{E}}} \delta_{si}^{\mathrm{server}} (\theta_{rs} + \theta_{r^{\mathrm{B}}s}) + \sum_{s \in S^{\mathrm{E}}} \min_{i \in S^{\mathrm{R}}} \delta_{is}^{\mathrm{server}} (\theta_{cs} + \theta_{c^{\mathrm{B}}s}) \\
& + \sum_{s \in S^{\mathrm{E}}} \min_{\substack{i \in S^{\mathrm{E}} \\ i \neq s}} \delta_{si}^{\mathrm{server}} (\theta_{cs} + \theta_{c^{\mathrm{B}}s} + \theta_{ds} + \theta_{d^{\mathrm{B}}s}) \\
& - \sum_{s \in S^{\mathrm{E}}} \min_{\substack{i \in S^{\mathrm{E}} \\ i \neq s}} \delta_{si}^{\mathrm{server}} \frac{1}{2} (\theta_{cs} + \theta_{c^{\mathrm{B}}s} + \theta_{ds} + \theta_{d^{\mathrm{B}}s}) \\
= & \sum_{s \in S^{\mathrm{R}}} \min_{i \in S^{\mathrm{E}}} \delta_{si}^{\mathrm{server}} (\theta_{rs} + \theta_{r^{\mathrm{B}}s}) + \sum_{s \in S^{\mathrm{E}}} \min_{i \in S^{\mathrm{R}}} \delta_{is}^{\mathrm{server}} (\theta_{cs} + \theta_{c^{\mathrm{B}}s}) \\
& + \frac{1}{2} \sum_{s \in S^{\mathrm{E}}} \min_{\substack{i \in S^{\mathrm{E}} \\ i \neq s}} \delta_{si}^{\mathrm{server}} (\theta_{cs} + \theta_{c^{\mathrm{B}}s} + \theta_{ds} + \theta_{d^{\mathrm{B}}s}). \qquad (6.41)
\end{aligned}$$

We then use the heuristic H^{RUP} in Algorithm 6.1 to minimize the lower bound of $\mathrm{PP}^{\mathrm{DP\_DELAY}}$. This heuristic is guaranteed to converge, as there always exists a feasible placement for provisioning an SFC configuration.

Algorithm 6.1 Round-up Heuristic H^{RUP}

Require: P': a relaxed pricing problem

1: **for** $v \in V$ **do**
2: Solve P' as LP problem
3: **repeat**
4: $S \leftarrow S_v$
5: $s' \leftarrow \mathrm{argmax}_{s \in S}\ \theta_{vs}$
6: $\theta_{vs'} \leftarrow 1$
7: **until** Latency constraints not violated
8: **end for**

6.5.3 Pricing with Objective 2: Dedicated Protection

Exact solution

Based on the definition of SFC availability from (6.4), we derive the following objective function:

$$\max \quad \text{PP}^{\text{DP\_AVAIL}} = a_r a_c a_d + D_{\text{DP}} \tag{6.42}$$

subject to:

$$\sum_{s \in S} \theta_{vs} = 1 \qquad v \in V^{rc} \tag{6.43}$$

$$\delta_{ss'}^{\text{server}}(\theta_{vs} + \theta_{v's'} - 1) \leq \delta_{vv'}^{\text{L}} \qquad v \in V^{rc}, v' \in V_v^{\text{DEP}}, s \in S_v, s' \in S_{v'} \tag{6.44}$$

$$\theta_{vs} + \theta_{v^{\text{B}}s} \leq 1 \qquad v \in V^{rc,\text{P}}, s \in S_v \tag{6.45}$$

$$\theta_{vs} \in \{0, 1\} \qquad v \in V^{rc}, s \in S_v, \tag{6.46}$$

where D_{DP} is defined in (6.30).

Resolve the non-linearity

Let $a_{ss'}$ denote the probability that both servers s and s' are available. Then $a_{ss'} = 1 - (1 - a_s)(1 - a_{s'})$.

Assuming that the availability of a VNF is equivalent to the availability of the server on which it is deployed, and using Equation (6.3), we introduce an additional set of constraints to compute the availability of each VNF.

By combining the objective of maximizing $\text{PP}^{\text{DP\_AVAIL}}$ with the following additional constraints, we ensure that the availability a_v of VNF v is accurately captured:

$$a_v \leq a_{ss'}(3 - \theta_{vs} - \theta_{v^{\text{B}}s'}) \qquad v \in V^{rc}, s, s' \in S_v, s \neq s'. \tag{6.47}$$

In corporation with the objective function, constraints (6.47) enforce that, for each primary VNF v, its overall availability probability is set as $a_{ss'}$ (a precomputed value/parameter) only when v is installed at s, and its backup instance is installed at s', which means:

$$a_v = \max_{s,s' \in S_v} (a_{ss'} \theta_{v,s} \theta_{v^{\text{B}},s'}). \tag{6.48}$$

The product $a_r a_c a_d$, which appears in the objective of the pricing, can be transformed into a series of additive terms:

$$a_r a_c a_d = \frac{1}{6}\left[(a_r + a_c + a_d)^3 - (a_r + a_c)^3 - (a_c + a_d)^3 \right.$$
$$\left. -(a_d + a_r)^3 + a_r^3 + a_c^3 + a_d^3\right]. \tag{6.49}$$

Each of the cubic terms is then linearized using a piecewise linear (PWL) function g^{PWL} that approximates $g(x) = x^3$. Then (6.49) can be written as:

$$\begin{aligned} a_r a_c a_d &= \frac{1}{6}\left[g(\alpha_{rcd}) - g(\alpha_{rc}) - g(\alpha_{cd}) - g(\alpha_{dr}) + g(a_r)\right. \\ &\quad \left. + g(a_c) + g(a_d)\right] \\ \alpha_{rcd} &= a_r + a_c + a_d \\ \alpha_{uv} &= a_u + a_v \qquad \text{with } u, v \in \{r, c, d\}. \end{aligned} \tag{6.50}$$

The PWL function $g^{\text{PWL}}(x)$ is depicted in Figure (6.7). The points defining the PWL functions are selected at evenly intervals across the input domain, as specified in Table (6.1).

Variables	Domain	Number of intervals
α_{rcd}	$[0, 3]$	8
α_{uv}	$[0, 2]$	5
a_u	$[0, 1]$	5

Table 6.1: Details of g depends on input variables

Heuristic

We use the heuristic H^{RUP} to solve the pricing problems.

6.5.4 Pricing with Objective 1: Shared Protection

Exact solution

$$\min \quad \text{PP}^{\text{SP\_DELAY}} = \delta_\gamma + D_{\text{SP}}, \tag{6.51}$$

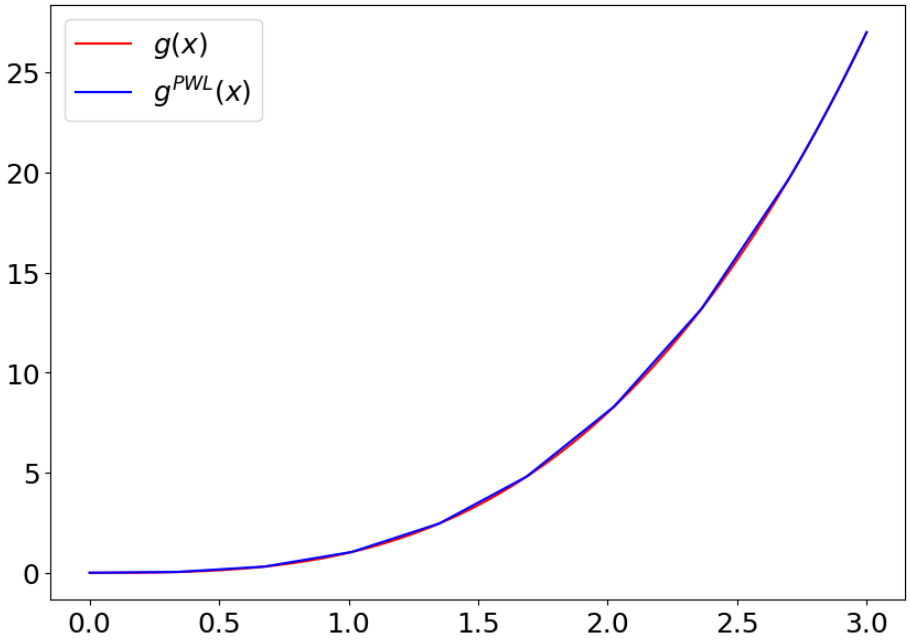


Figure 6.7: PWL function

Algorithm 6.2 Semi Round-up Heuristic H^{SRUP}

Require: P': a relaxed pricing problem

1: **for** $v \in V^{\text{P}}$ **do**
2: Solve P' as LP problem
3: **repeat**
4: $S \leftarrow S_v$
5: $s' \leftarrow \text{argmax}_{s \in S}\ \theta_{vs}$
6: $\theta_{vs'} \leftarrow 1$
7: **until** Latency constraints not violated
8: **end for**
9: For $v \in V^{\text{B}}$, θ_{vs} is decided by solving the pricing problem as a MILP.

subject to:

$$\sum_{s\in S}\theta_{vs}=1 \qquad v\in V \tag{6.52}$$

$$\delta^{\text{server}}_{ss'}(\theta_{vs}+\theta_{v's'}-1)\le\delta^{\text{L}}_{vv'} \qquad v\in V^{\text{P}}, v'\in V^{\text{DEP}}_{v}, s\in S_v, s'\in S_{v'} \tag{6.53}$$

$$\theta_{vs}+\theta_{v^{\text{B}}s}\le 1 \qquad v\in V^{\text{P}}, s\in S_v \tag{6.54}$$

$$\theta_{vs}\in\{0,1\} \qquad v\in V, s\in S_v. \tag{6.55}$$

where D_{SP} is the duality term in shared protection scheme, defined by the following formula:

$$\begin{aligned} D_{\text{SP}} =& \sum_{s\in S} u^{(6.13)}_{c}\theta_{c,s}+\sum_{s\in S} u^{(6.14)}_{r,s}\theta_{r,s} \\ &+\sum_{v\in V^{rc}}\sum_{s\in S^{\text{E}}}\sum_{s'\in S^{\text{E}}}\sum_{\square\in\{\text{CPU,RAM}\}} u^{(6.16)}_{s,s',\square}\tau^{\square}_{v}\theta_{vs}\theta_{v^{\text{B}}s'} \\ &+\sum_{v\in V^{rc}}\sum_{s\in S}\sum_{\square\in\{\text{CPU,RAM}\}} u^{(6.17)}_{s}\tau^{\square}_{v}\theta_{v,s}. \end{aligned} \tag{6.56}$$

Heuristic

Similarly to Section 6.5.2, we can use (6.41) as the lower bound for δ_γ. We also need to linearize the non-linear term of D_{SP}. Notice that $u^{(6.16)}$ is negative in minimization problem and τ is positive, so we have:

$$u^{(6.16)}_{s,s',\square}\,\tau^{\square}_{v}\,\theta_{vs}\theta_{v^{\text{B}}s'}\ge u^{(6.16)}_{s,s',\square}\,\tau^{\square}_{v}\,\frac{1}{2}(\theta_{vs}+\theta_{v^{\text{B}}s'}). \tag{6.57}$$

We then apply the heuristic H^{SRUP} in Algorithm 6.2 to solve the pricing with the new objective being the lower bound of $\text{PP}^{\text{SP\_DELAY}}$.

6.5.5 Pricing with Objective 2: Shared Protection

Exact solution

$$\max \quad \text{PP}^{\text{SP\_AVAIL}}=a_r a_c a_d - D_{\text{SP}} \tag{6.58}$$

subject to:

$$\sum_{s \in S} \theta_{vs} = 1 \qquad v \in V \tag{6.59}$$

$$\delta^{\text{server}}_{ss'}(\theta_{vs} + \theta_{v's'} - 1) \leq \delta^{\text{L}}_{vv'} \qquad v \in V^{\text{P}}, v' \in V^{\text{DEP}}_{v}, s \in S_v, s' \in S_{v'} \tag{6.60}$$

$$\theta_{vs} + \theta_{v^{\text{B}}s} \leq 1 \qquad v \in V^{\text{P}}, s \in S_v \tag{6.61}$$

$$\theta_{vs} \in \{0, 1\} \qquad v \in V, s \in S_v. \tag{6.62}$$

Heuristic

The linear upper bound of $\text{PP}^{\text{SP\_AVAIL}}$ is obtained by applying (6.50) to linearize the availability. For the non-linear term of D_{SP}, notice that in the maximization problem, $u^{(6.16)}$ is positive, which allows us to use the following formulation:

$$u^{(6.16)}_{s,s',\square}\, \tau^{\square}_{v}\, \theta_{vs}\theta_{v^{\text{B}}s'} \geq u^{(6.16)}_{s,s',\square}\, \tau^{\square}_{v}\, (\theta_{vs} + \theta_{v^{\text{B}}s'} - 1). \tag{6.63}$$

Then we apply the heuristic H^{SRUP} to solve the new pricing.

6.5.6 Lagrangian Bound

From Vanderbeck [87], the Lagrangian bound is a valid bound computed by:

$$z_{\text{LB}} = ub + \sum_{\text{PP}} rc_{\text{PP}}, \tag{6.64}$$

where u is the dual vector and b is the constant vector on the right-hand side, rc_{PP} denotes the reduced cost of the pricing problem PP. Since the exact reduced cost of the pricing problem cannot be determined due to its nonlinearity, we proposed a new bound LB$'$ defined as:

$$\begin{aligned} z^{C}_{\text{LB}'} &= ub + \sum_{\text{PP}} \text{UB}^{C}_{\text{PP}} \\ &\geq ub + \sum_{\text{PP}} rc_{\text{PP}} \\ &\geq z^{C}_{\text{LB}} \qquad C \in \{\text{PP}^{\text{DP\_AVAIL}}, \text{PP}^{\text{SP\_AVAIL}}\}, \end{aligned} \tag{6.65}$$

where $\text{UB}^{C}_{\text{PP}}$ is the linear upper bound of objective function C. Similarly, for $C \in \{\text{PP}^{\text{DP\_DELAY}}, \text{PP}^{\text{SP\_DELAY}}\}$:

$$z^{C}_{\text{LB}'} = ub + \sum_{\text{PP}} \text{LB}^{C}_{\text{PP}}, \tag{6.66}$$

where $\text{LB}^{C}_{\text{PP}}$ is the linear lower bound of objective C.

6.6 An ILP Formulation with a Stronger Relaxation Bound

We now show that the dedicated protection model DP_M described in Section 6.4.5 provides a stronger LP that a classical compact Integer Linear Programming approach, as proposed in [80]. We recall it below.

$$\text{OBJ}^{\text{DT\_BIP}} = \max \sum_{v \in V} \sum_{s \in S} t_{vs} \theta'_{vs} \tag{6.67}$$

object to:

$$\delta^{\text{server}}_{ss'} \times (\theta'_{vs} + \theta'_{v's'} - 1) - \delta^{\text{L}}_{vv'} \leq 0 \quad v \in V^{rc}, v' \in V^{\text{DEP}}_{v} \quad s \in S_v, s' \in S_{v'} \tag{6.68}$$

$$\sum_{v \in V} \theta'_{vs} \tau^{\square}_{v} \leq \text{CAP}^{\square}_{s} \quad s \in S \quad \square \in \{\text{CPU}, \text{RAM}\} \tag{6.69}$$

$$\sum_{s \in S^{\text{R}}} \theta'_{vs} = 1 \quad v \in V^{\text{NRT\_RIC}} \tag{6.70}$$

$$\sum_{s \in S^{\text{E}}} \theta'_{vs} = 1 \quad v \in V^{cd} \tag{6.71}$$

$$\theta'_{vs} + \theta'_{v^{\text{B}}s} \leq 1 \quad v \in V^{\text{P}}, s \in S_v, \tag{6.72}$$

where t_{vs} represents the downtime of VNF v on server s, and θ'_{vs} is a binary decision variable equal to 1 if VNF v is deployed on server s. Constraints (6.68) ensure that the interconnection latencies between VNFs do not exceed the specified thresholds. Constraint (6.69) guarantees that resource usage does not surpass the available capacity of each server. Constraints (6.70, 6.71) enforce that each VNF is placed uniquely on its designated server type. Finally, constraints (6.72) prevent a VNF and its backup from being deployed on the same server.

To enable a fair comparison between the two models, we first align their objective functions. Then, following the assumption made in [80], we assume that each NRT_RICis connected to a unique O-CU. An equivalence objective function of $\text{OBJ}^{\text{DT}}_{\text{BIP}}$

for the decomposition model DP_M is:

$$\text{OBJ}^{\text{DT}} = \max \sum_{s \in S} \sum_{v \in V} \sum_{\gamma \in \Gamma} t_{vs} z_\gamma \theta^\gamma_{vs}. \tag{6.73}$$

Theorem 1. *Let* $\overline{\text{OBJ}}^{\text{DT}}$ *and* $\overline{\text{OBJ}}^{\text{DT\_BIP}}$ *denote the optimal values of the continuous relaxation of the formulations* DT *and* DT_BIP*, respectively, then:*

$$\overline{\text{OBJ}}^{\text{DT}} \leq \overline{\text{OBJ}}^{\text{DT\_BIP}}$$

Theorem 1 shows that the continuous relaxation of our model provides a tighter upper bound on the optimal value compared to the relaxation of the model in [80].

Suppose that $(z_\gamma, x_{vs}, \theta^\gamma_{vs})$ is the optimal solution of $\overline{\text{OBJ}}^{\text{DT}}$, note that $z_\gamma, x_{vs} \in [0, 1]$ and $\theta^\gamma_{vs} \in \{0, 1\}$, we define the following:

$$\theta'_{vs} = \sum_{\gamma \in \Gamma} z_\gamma \theta^\gamma_{vs} \quad v \in V, s \in S_v. \tag{6.74}$$

We need to prove that constraints (6.68 - 6.72) are satisfied.

Constraints (6.70, 6.71)

From (6.6), and given that the relationships between (NRT_RIC, O-CU), (O-CU, O-DU) are one-to-one, we have:

$$\sum_{\gamma \in \Gamma} \sum_{s \in S^{\text{R}}} \theta^\gamma_{rs} z_\gamma = 1 \quad r \in V^{\text{P,NRT\_RIC}} \tag{6.75}$$

$$\sum_{\gamma \in \Gamma} \sum_{s \in S^{\text{E}}} \theta^\gamma_{ds} z_\gamma = 1 \quad d \in V^{\text{P,O-DU}}. \tag{6.76}$$

Furthermore, since each primary VNF has a unique corresponding backup VNF, we can also conclude that:

$$\begin{aligned} &\sum_{\gamma \in \Gamma} \sum_{s \in S} \theta^\gamma_{vs} z_\gamma = 1 \qquad && v \in V \\ \Rightarrow &\sum_{s \in S} \theta'_{vs} = 1 && v \in V, \end{aligned} \tag{6.77}$$

which confirms that (6.70, 6.71) are satisfied.

Constraints (6.72)

In the next part, we will prove that (6.72) is satisfied. To support the upcoming analysis, we first present an auxiliary result.

Lemma 2. *Given two VNFs v_1, v_2, two servers s_1, s_2 and a configuration γ such that γ provision both v_1 and v_2. If $\theta^{\gamma}_{v_1 s_1}$ and $\theta^{\gamma}_{v_2 s_2}$ cannot be 1 at the same time, then $\theta^{\gamma}_{v_2 s_2} \leq \sum_{s \neq s_1} \theta^{\gamma}_{v_1 s}$.*

Proof. If $\theta^{\gamma}_{v_2 s_2} = 1$, then in γ, v_2 must be installed on any server other than s_1, then $\sum_{s \neq s_1} \theta^{\gamma}_{v_1 s} = \theta^{\gamma}_{v_2 s_2} = 1$. If $\theta^{\gamma}_{v_2 s_2} = 0$, then $\theta^{\gamma}_{v_2 s_2} \leq \sum_{s \neq s_1} \theta^{\gamma}_{v_1 s}$. From both cases, we can conclude that $\theta^{\gamma}_{v_2 s_2} \leq \sum_{s \neq s_1} \theta^{\gamma}_{v_1 s}$. □

Let s^* and v be an arbitrary server and $v \in V^{\mathrm{P},cd}$. Using (6.74), we have:

$$\begin{aligned} & \theta'_{vs^*} + \theta'_{v^{\mathrm{B}}s^*} \\ =& \sum_{\gamma \in \Gamma} z_\gamma \theta^{\gamma}_{vs^*} + \sum_{\gamma \in \Gamma} z_\gamma \theta^{\gamma}_{v^{\mathrm{B}}s^*} \\ =& \underbrace{\sum_{\gamma \in \Gamma} (z_\gamma \theta^{\gamma}_{vs^*} + z_\gamma \theta^{\gamma}_{v^{\mathrm{B}}s^*})}_{B}. \end{aligned} \tag{6.78}$$

Using Lemma 2, we have $\sum_{s \neq s^*} \theta^{\gamma}_{vs} \geq \theta^{\gamma}_{v^{\mathrm{B}}s^*}$. Then:

$$\begin{aligned} B &\leq \sum_{\gamma \in \Gamma} (z_\gamma \theta^{\gamma}_{vs^*} + z_\gamma \sum_{s \neq s^*} \theta^{\gamma}_{vs}) \\ &\leq \sum_{\gamma \in \Gamma} \sum_{s \in S} z_\gamma \theta^{\gamma}_{vs} = 1. \end{aligned} \tag{6.79}$$

This concludes that the constraints (6.72) are satisfied.

Constraints (6.69)

From (6.9) and (6.74) we can easily see that (6.69) is correct for all edge servers. We now prove that it is also correct for regional servers.

Since the relation between NRT_RIC and O-CU is one-to-one, for an arbitrary NRT_RIC r, $n_r^{\text{O-CU}} = 1$, then (6.7) become:

$$\sum_{\gamma\in\Gamma}\theta^{\gamma}_{rs}z_{\gamma}\leq x_{rs}\qquad r\in V^{\text{NRT\_RIC}},s\in S$$
$$\Rightarrow\theta'_{rs}\leq x_{rs}\qquad r\in V^{\text{NRT\_RIC}},s\in S. \tag{6.80}$$

Combine (6.10) and (6.80), we have:

$$\sum_{r\in V^{\text{NRT\_RIC}}}\tau^{\square}_{r}\theta'_{rs}\leq\sum_{r\in V^{\text{NRT\_RIC}}}\tau^{\square}_{r}x_{rs}\leq\text{CAP}^{\square}_{s}\quad s\in S^{\text{R}}$$
$$\square\in\{\text{CPU},\text{RAM}\}. \tag{6.81}$$

This concludes that constraints (6.69) are satisfied. Finally, we will check the constraints (6.68).

Constraints (6.68)

Consider arbitrary VNF v_1 and its dependence v_2, two servers $s_1\in S_{v_1}$ and $s_2\in S_{v_2}$. Constraints (6.68) become:

$$\delta^{\text{server}}_{s_1s_2}(\theta'_{v_1s_1}+\theta'_{v_2s_2})\leq\delta^{\text{server}}_{s_1s_2}+\delta^{\text{L}}_{v_1v_2}. \tag{6.82}$$

For any configuration γ that provisioning v_1 and v_2, due to way each configuration is generated, we always have:

$$\begin{aligned}
&\delta^{\text{server}}_{s_1s_2}(\theta^{\gamma}_{v_1s_1}+\theta^{\gamma}_{v_2s_2}-1)\leq\delta^{\text{L}}_{v_1v_2}\\
\equiv\quad&\delta^{\text{server}}_{s_1s_2}(\theta^{\gamma}_{v_1s_1}+\theta^{\gamma}_{v_2s_2})\leq\delta^{\text{server}}_{s_1s_2}+\delta^{\text{L}}_{v_1v_2}\\
\equiv\quad&\delta^{\text{server}}_{s_1s_2}(\theta^{\gamma}_{v_1s_1}+\theta^{\gamma}_{v_2s_2})z_{\gamma}\leq(\delta^{\text{server}}_{s_1s_2}+\delta^{\text{L}}_{v_1v_2})z_{\gamma}\\
\equiv\quad&\sum_{\gamma\in\Gamma_{v_1}}\delta^{\text{server}}_{s_1s_2}(\theta^{\gamma}_{v_1s_1}+\theta^{\gamma}_{v_2s_2})z_{\gamma}\\
&\qquad\leq(\delta^{\text{server}}_{s_1s_2}+\delta^{\text{L}}_{v_1v_2})\sum_{\gamma\in\Gamma_{v_1}}z_{\gamma},
\end{aligned} \tag{6.83}$$

where $\Gamma_v=\{\gamma\,|\exists s\in S,\theta^{\gamma}_{vs}=1\}$. Note that (6.74) can also be written as:

$$\theta'_{vs}=\sum_{\gamma\in\Gamma_v}z_{\gamma}\theta^{\gamma}_{vs}\quad v\in V,s\in S_v. \tag{6.84}$$

For an arbitrary configuration $\gamma \in \Gamma_{v_1}$, v_1 must be located on a unique server in γ, which means $\sum_{s \in S} \theta^{\gamma}_{v_1 s} = 1$. Then:

$$\begin{aligned}\sum_{\gamma \in \Gamma_{v_1}} z_\gamma &= \sum_{\gamma \in \Gamma_{v_1}} z_\gamma \sum_{s \in S} \theta^{\gamma}_{v_1 s} = \sum_{s \in S} \sum_{\gamma \in \Gamma_{v_1}} z_\gamma \theta^{\gamma}_{v_1 s} \\ &= \sum_{s \in S} \theta_{v_1 s} = 1. \end{aligned} \tag{6.85}$$

Then (6.83) becomes:

$$\begin{aligned} & \delta^{\text{server}}_{s_1 s_2} \sum_{\gamma \in \Gamma_{v_1}} (\theta^{\gamma}_{v_1 s_1} z_\gamma + \theta^{\gamma}_{v_2 s_2} z_\gamma) \leq \delta^{\text{server}}_{s_1 s_2} + \delta^{\text{L}}_{v_1 v_2} \\ \Longleftrightarrow & \delta^{\text{server}}_{s_1 s_2} (\theta'_{v_1 s_1} + \theta'_{v_2 s_2}) \leq \delta^{\text{server}}_{s_1 s_2} + \delta^{\text{L}}_{v_1 v_2}. \end{aligned} \tag{6.86}$$

This confirms the validity of (6.82), and consequently, the constraints (6.68) are satisfied.

We have shown that the optimal solution of $\overline{\text{OBJ}}^{\text{DT}}$ is also a feasible solution of $\overline{\text{OBJ}}^{\text{DT\_BIP}}$, indicating $\overline{\text{OBJ}}^{\text{DT}} \leq \overline{\text{OBJ}}^{\text{DT\_BIP}}$.

6.7 Numerical Results

We now discuss the validation of the proposed mathematical models. We first describe the methodology used to generate the data. We then analyze the performance of the models, comparing the resource consumption of the two protection schemes, as well as the impact of the objective on these two schemes.

6.7.1 Data Generator

While edge and regional clouds are small data centers today, they are expected to be comprised of many more data centers tomorrow. With this in mind, we generated the data using the following steps.

Step 1: We generate a network topology $G(V, E)$, where E is the set of edges and V is the set of nodes (routers/switches). Each edge is characterized by its length ℓ. We assumed a fixed link capacity of $c = 10^4$ Gb/s and a signal propagation speed of $v_p = 2 \times 10^5$ km/s. A message size of $M = 20$ Mb is used for delay calculation. The delay between 2 nodes $\delta_{vv'}$ is computed by $\delta_{vv'} = t_d + t_p$ where

- t_d is transmission time computed by M/c
- t_p is propagation time computed by ℓ/v_p.

Step 2: A set of nodes from V is selected to host data centers. Each data center follows a Fat-Tree architecture, illustrated in Figure 6.8. The configuration of each server is randomized between (4 TB CPU, 32 GB RAM) and (8 TB CPU, 64 GB RAM).

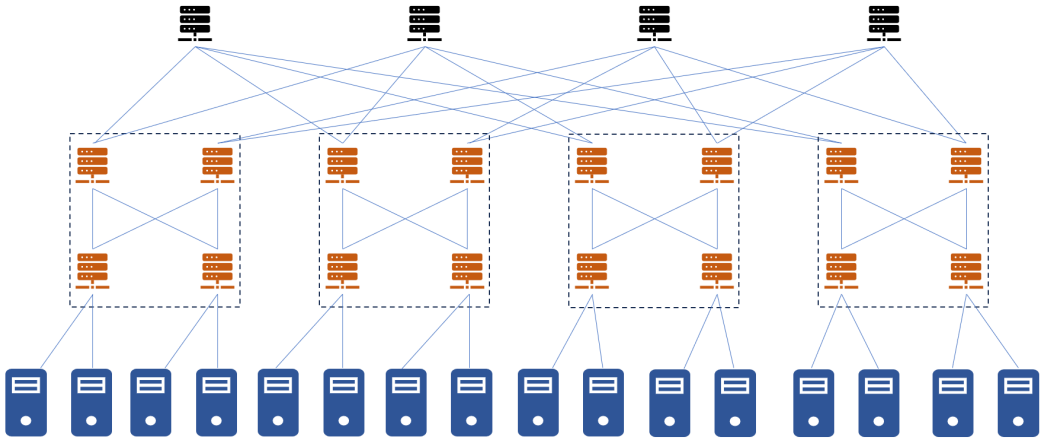


Figure 6.8: Fat tree

Step 3: The latency matrix between each pair of servers is such that the delay between 2 servers s, s' is the weight of the shortest path in which the weight of each edge is the delay calculated from Step 1.

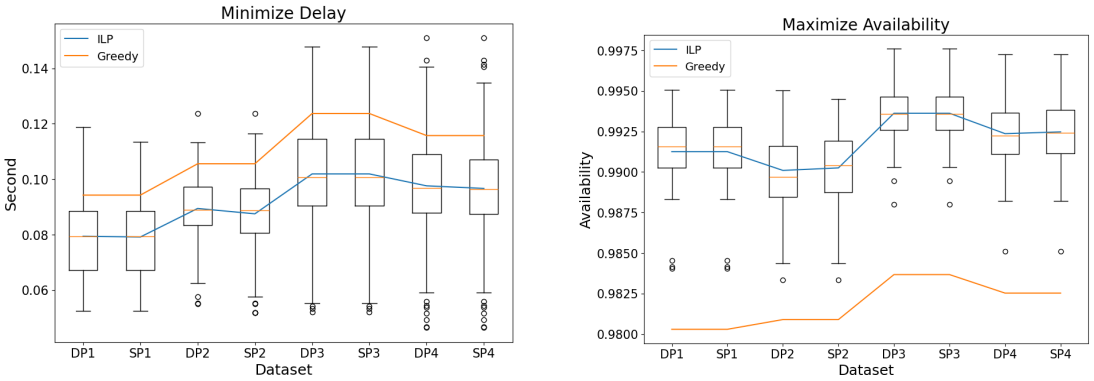
Step 4: Generate Mean Time to Failure (MTTF) and Mean Time to Repair (MTTR). Each data center has a mean MTTF $a_M TTF$ that is randomized between $\min_M TTF =$ 25 months and $\max_M TTF = 36$ months. The MTTF value of each server is randomized using a normal distribution with the mean value is $a_M TTF$ and the standard deviation is $0.2 \times a_M TTF$. The MTTR value of each server is also generated following the normal distribution with the mean value $a_M TTR = 5$ hours and the standard deviation $0.2 \times a_M TTR$. The availability of a server s is then computed as:

$$a_s = \frac{a_M TTF}{a_M TTF + a_M TTR}.$$

Step 5: We generate SFCs chains. Given a number of Near-Realtime RICs, each NRT_RIC is associated with a number of O-CUs (i.e., $n_{\text{O-CU}}$) which is a random value between $\min_{\text{O-CU}}$ and $\max_{\text{O-CU}}$. Some of the specifications of the generated datasets are shown in Table 6.2.

Dataset	#regional servers	#edge servers	#VNFs	#SFCs
1	48	96	160	75
2	120	160	254	120
3	240	384	462	221
4	240	384	590	283

Table 6.2: Dataset details



(a) Delay minimization

(b) Availability maximization

Figure 6.9: Comparison between dedicated and shared protection

6.7.2 Delay vs. Availability

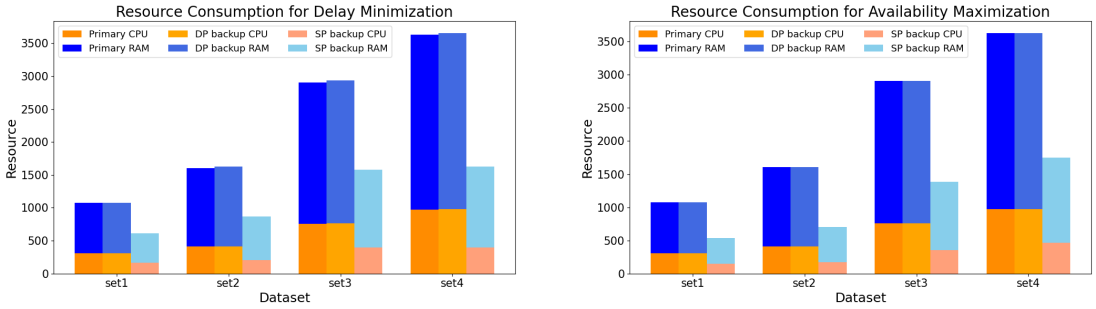
Figures 6.9a and 6.9b illustrate the average delay and average availability per SFC, respectively. On the x-axis, DPi, SPi represent the results of dedicated protection and shared protection models, respectively, for dataset i. The box plots represent the overall delay and availability per SFC, while the line plots show the average delay and availability in Figures 6.9a and 6.9b, respectively. In Figure 6.9a we compare the average results between dedicated protection and shared protection, which showed an improvement of 0.8% on average. The dedicated protection also performed better than the greedy heuristic by 16% on average.

Achieving a small improvement, when maximizing availability, Figure 6.9b pictured a 0.007% improvement between shared protection compared to dedicated protection, and a 1% improvement for dedicated protection compared to greedy heuristic.

We successfully obtained a good bound for the maximization of availability. The

Dataset	Shared Protection			Dedicated Protection		
	ILP	Bound	Gap (%)	ILP	Bound	Gap (%)
1	74.34	74.64	0.39	74.34	75.38	1.40
2	118.83	119.42	0.49	118.81	120.61	1.52
3	219.59	220.48	0.40	219.59	222.68	1.41
4	280.87	282.27	0.50	280.84	285.08	1.51

Table 6.3: Optimality gap for the maximum availability model



(a) Delay minimization

(b) Availability maximization

Figure 6.10: Resource consumption dedicated vs. shared protection

results are presented in Table 6.3, where the gap represents the maximum percentage difference between the ILP result and the optimal solution.

In this study, we explored two extreme objectives. However, in practice, these objectives are usually more balanced, for example, a trade-off between minimizing delay, maximizing availability, and saving resources used for backups with shared protection.

6.7.3 Shared vs. Dedicated

While both protection schemes aim to ensure network function continuity during a single-server failure, Shared Protection (SP) prioritizes resource efficiency by minimizing the backup VNF resource consumption. Figures 6.10a and 6.10b illustrate the comparison of resource usage between Shared Protection (SP) and Dedicated Protection (DP) for both objective functions. In the delay minimization problem, SP utilizes an average of 53% of the CPU and 54% of the RAM compared to DP. Similarly, in the availability maximization problem, SP achieves an average resource

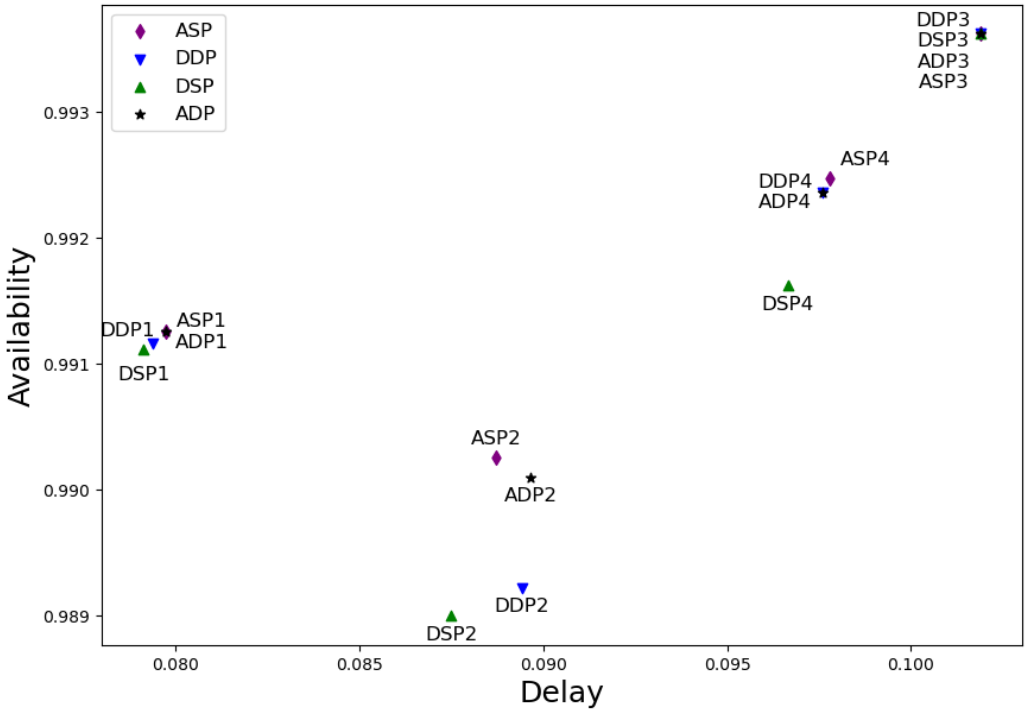


Figure 6.11: Delay vs Availability

usage of 49% for CPU and 48% for RAM relative to DP.

In the dedicated protection scheme, backups are pre-configured, meaning that in the event of a single-server failure, failover occurs almost instantly. In contrast, the shared protection scheme introduces a slight delay, as backup resources need to be activated. However, this minor drawback is offset by the significant resource savings, up to approximately 50%-offered by shared protection.

Setting aside the advantage of resource savings, Figure 6.11 shows no clear dominance between DP_DELAY and SP_DELAY (DDP and DSP), nor between DP_AVAIL, SP_AVAIL (ADP and ASP). For instance, while shared protection offers notable resource efficiency, it may not always yield the lowest latency and highest availability, depending on network topology and traffic distribution. These trade-offs highlight the potential value of hybrid strategies that dynamically select protection schemes based on network conditions and service requirements.

6.8 Conclusions

In this study, we proposed decomposition-based models for VNF provisioning, with dedicated and shared protection mechanisms against service interruptions caused by single-server failures. We consider the objectives of minimizing delay and maximizing availability. In spite of the nonlinear constraints, we manage to design a solution process able to handle quite large number of servers, definitively at least as larger as the number typically found in a edge (today typically in the order one or two racks) or regional cloud of a 5G network. The edge computing market is experiencing a significant surge, with projections for substantial growth in the coming years, and our proposed mathematical model should be able to cope with it.

Our numerical results demonstrate that the proposed models outperform the greedy heuristic, achieving a 16% improvement in delay minimization and a 1% improvement in availability maximization. While the difference between Shared Protection (SP) and Dedicated Protection (DP) in terms of performance is minimal, SP significantly reduces backup VNF resource consumption, saving nearly 50% compared to DP. This could lead in a significant amount of energy, another big concern of future networks.

Acknowledgment

We would like to thank Ciena and Mitacs for their valuable financial and technical support for this project. We would also like to thank Abdallah Shami (professor, University of Western Ontario) for introducing us to the topic of this study, and for Quang Huy Duong for his early comments on the project.

Chapter 7

Conclusion and Future Directions

7.1 Conclusion

To recap, in this thesis we investigate two big topics:

- Provisioning in the physical layer, i.e., Routing and Spectrum Assignment (RSA) problem
- Provisioning in the logical layer, i.e., SFC placement problem

The first topic encompasses the contributions presented in Chapter 3, 4, and 5. In Chapter 3, we introduced two distinct approaches for solving the RSA problem: the link-based (LBD) and node-based (NBD) decomposition models. While the LBD approach demonstrated a faster convergence rate compared to NBD in many cases, it is noteworthy that in certain datasets, the NBD approach exhibited shorter runtime. Moreover, the numerical results highlighted that, when the objective was to maximize network throughput, the First Fit greedy heuristic produced solutions that were on par with both LBD and NBD.

Chapters 4 and 5 addressed the interference-aware RSA problem. In Chapter 4, the solution was achieved by integrating OSNR constraints into the previously established link-based decomposition model. However, owing to the complexity of the subproblems involved, the utilization of Tabu Search became necessary to tackle this challenge. While the proposed solution in Chapter 4 has shown effectiveness compared to greedy algorithms like Best-Fit and First-Fit, it is not without limitations.

Specifically, Tabu Search, utilized to solve the sub-problems, faced challenges in finding optimal solutions, leading to suboptimal results. Additionally, the running time of the solution was impacted by the computational complexity introduced by Tabu Search.

In Chapter 5, we tackled the limitations of Tabu Search in finding optimal solutions for the sub-problems. To overcome this, we proposed a reformulation of the sub-problems as a conflict graph and maximum weight independent set. This new formulation allowed us to achieve optimal solutions for the subproblems. While the running time issue persisted, the quality of the solution was significantly enhanced compared to the previous approach discussed in Chapter 4, with an average improvement of 10%.

The second topic, covered in Chapter 6, examined VNF placement in O-RAN networks under reliability constraints. We proposed decomposition-based models for both dedicated and shared protection schemes, as well as heuristics to address the non-linearities in the sub-problems. The delay minimization models outperformed greedy heuristics by at least 5.6%, while the availability maximization models achieved gains of at least 1%. Although the overall performance difference between the two protection schemes was relatively small, shared protection reduced resource usage by up to 50% compared to dedicated protection–even without explicitly optimizing for resource efficiency.

7.2 Future Works

In the first topic of this thesis, we do not include the launch power of each channel as variables in our model. However, it is worth noting that by considering the launch power as variables, we could potentially enhance our results and incorporate energy-saving objectives into our optimization framework. By explicitly incorporating launch power as a variable, we may be able to further improve the performance and explore energy-efficient solutions.

There are several avenues for further research and development based on the insights gained from this thesis. One promising direction is to delve deeper into the capabilities of Column Generation and explore its potential in solving even more complex large-scale optimization problems. By refining the formulation and leveraging

the benefits of Nested Column Generation, we can potentially extend the algorithm's applicability to include additional variables such as launch power and modulation format. By considering launch power as a variable, we can optimize its allocation and distribution across the network, leading to improved signal quality and potentially reduced power consumption. Similarly, incorporating modulation format as a variable allows for the optimization of data transmission rates and bandwidth utilization, ultimately resulting in enhanced network capacity and spectral efficiency.

To effectively incorporate launch power and modulation format as variables, further investigation and experimentation are needed. This could involve developing new mathematical models and optimization formulations that capture the relationships between these variables and other relevant factors. Additionally, it would be valuable to explore the impact of these variables on different network architectures, traffic patterns, and performance metrics. This empirical analysis would provide valuable insights into the benefits and trade-offs associated with varying launch power and modulation formats.

In the second topic, the results underscore the potential of multi-objective formulations that simultaneously optimize latency, availability, and resource consumption– objectives that better align with practical industrial requirements. One limitation of this study is the omission of dynamic end-user location distributions, which can significantly influence SFC provisioning strategies. Incorporating the geographical distribution of end users and the locations of Radio Units (O-RU) in O-RAN networks could lead to more realistic and effective solutions for real-world deployments.

Bibliography

[1] Sandvine 2024 Global Internet Phenomena Report.

[2] Cisco visual networking index: Forecast and trends, 2017–2022. White Paper, 2019.

[3] F. S. Abkenar and A. G. Rahbar. Study and analysis of routing and spectrum allocation (RSA) and routing, modulation and spectrum allocation (RMSA) algorithms in elastic optical networks (EONs). *Optical Switching and Networking*, 23:5–39, 2017.

[4] T-Systems International AG and M. Jaeger. SNDlib, 2005.

[5] R.W. Alaskar, I. Ahmad, and A. Alyatama. Offline routing and spectrum allocation algorithms for elastic optical networks. *Optical Switching and Networking*, 21:79–92, 2016.

[6] A. Amari, O. A. Dobre, R. Venkatesan, O. S. Sunish Kumar, P. Ciblat, and Y. Jaouën. A survey on fiber nonlinearity compensation for 400 Gb/s and beyond optical communication systems. *IEEE Communications Surveys & Tutorials*, 19(4):3097–3113, 2017.

[7] R. Antil, S B. Pinki, and S. Beniwal. An overview of DWDM technology & network. *Int. J. Sci. Technol. Res*, 1(11):43–46, 2012.

[8] A. Asiri and B. Wang. Deep reinforcement learning for QoT-aware routing, modulation, and spectrum assignment in elastic optical networks. *Journal of Lightwave Technology*, 2024.

[9] L. Askari, M. Tamizi, O. Ayoub, and M. Tornatore. Protection strategies for dynamic vnf placement and service chaining. In *International Conference on Computer Communications and Networks (ICCCN)*, pages 1–9. IEEE, 2021.

[10] C. Barnhart, E. L Johnson, G. L Nemhauser, M. WP Savelsbergh, and P. H Vance. Branch-and-price: Column generation for solving huge integer programs. *Operations research*, 46(3):316–329, 1998.

[11] M. Batayneh, D.A. Schupke, M. Hoffmann, A. Kirstaedter, and B. Mukherjee. On routing and transmission-range determination of multi-bit-rate signals over mixed-line-rate WDM optical networks for carrier ethernet. *IEEE/ACM Transactions on Networking*, 19(5):1304–1316, Oct. 2011.

[12] M. Bianchetti and J. Marenco. Valid inequalities and a branch-and-cut algorithm for the routing and spectrum allocation problem. *Procedia Computer Science*, 195:523–531, 2021. Proceedings of the XI Latin and American Algorithms, Graphs and Optimization Symposium.

[13] S. Bottacchi. *Noise and signal interference in optical fiber transmission systems: an optimum design approach*. John Wiley & Sons, 2008.

[14] Y. C. Chang, K. H. Chang, and T. K. Chang. Applied column generation-based approach to solve supply chain scheduling problems. *International Journal of Production Research*, 51(13):4070–4086, 2013.

[15] B. C. Chatterjee, N. Sarma, and E. Oki. Routing and spectrum allocation in elastic optical networks: A tutorial. *IEEE Communications Surveys & Tutorials*, 17(3):1776–1800, 2015.

[16] B.C. Chatterjee, S. Ba, and E. Oki. Fragmentation problems and management approaches in elastic optical networks: A survey. *IEEE Communications Surveys & Tutorials*, 20(1):183–210, 2018.

[17] X. Chen, B. Li, R. Proietti, H. Lu, Z. Zhu, and S. J. B. Yoo. DeepRMSA: A deep reinforcement learning framework for routing, modulation and spectrum assignment in elastic optical networks. *Journal of Lightwave Technology*, 37(16):4155–4163, 2019.

[18] Y. Cheng, S. Ding, Y. Shao, and C. C. K. Chan. PtrNet-RSA: a pointer network-based QoT-aware routing and spectrum assignment scheme in elastic optical networks. *Journal of Lightwave Technology*, 42(17):5808–5819, 2024.

[19] K. Christodoulopoulos, I. Tomkos, and E. A. Varvarigos. Elastic bandwidth allocation in flexible OFDM-based optical networks. *Journal of Lightwave Technology*, 29(9):1354–1366, 2011.

[20] V. Chvatal. *Linear Programming*. Freeman, 1983.

[21] J. Desrosiers and M. E. Lübbecke. *A Primer in Column Generation*, pages 1–32. Springer US, Boston, MA, 2005.

[22] I. Diarrassouba and Y. Hadhbi. The constrained-routing and spectrum assignment problem: Valid inequalities and branch-and-cut algorithm. In *Combinatorial Optimization*, pages 35–47, Cham, 2022. Springer International Publishing.

[23] Q. H. Duong and B. Jaumard. A nested decomposition model for reliable NFV 5G network slicing. In *INOC*, pages 107–112, 2019.

[24] Q. H. Duong, I. Tamim, B. Jaumard, and A. Shami. A column generation algorithm for dedicated-protection O-RAN VNF deployment. In *International Wireless Communications and Mobile Computing (IWCMC)*, pages 1206–1211, 2022.

[25] Q. H. Duong, I. Tamim, B. Jaumard, and A. Shami. A column generation algorithm for dedicated-protection O-RAN VNF deployment. In *International Wireless Communications and Mobile Computing (IWCMC)*, pages 1206–1211, 2022.

[26] J. Enoch and B. Jaumard. Towards optimal and scalable solution for routing and spectrum allocation. *Electronic Notes in Discrete Mathematics (ENDM)*, 64C:335–344, 2018.

[27] D. Eppstein. Finding the k shortest paths. *SIAM Journal on computing*, 28(2):652–673, 1998.

[28] J. Fan, C. Guan, Y. Zhao, and C. Qiao. Availability-aware mapping of service function chains. In *IEEE Conference on Computer Communications*, pages 1–9, 2017.

[29] J. Fan, M. Jiang, O. Rottenstreich, Y. Zhao, T. Guan, R. Ramesh, S. Das, and C. Qiao. A framework for provisioning availability of NFV in data center networks. *IEEE Journal on Selected Areas in Communications*, 36(10):2246–2259, 2018.

[30] D. Feillet. A tutorial on column generation and branch-and-price for vehicle routing problems. *4or*, 8(4):407–424, 2010.

[31] G. Friesecke, A. S Schulz, and D. Vogler. Genetic column generation: fast computation of high-dimensional multimarginal optimal transport problems. *SIAM Journal on Scientific Computing*, 44(3):A1632–A1654, 2022.

[32] A. Garcia-Saavedra and X. Costa-Pérez. O-RAN: Disrupting the virtualized RAN ecosystem. *IEEE Communications Standards Magazine*, 5(4):96–103, 2021.

[33] D. Gattuso, G. C. Cassone, and D. S. Pellicanò. Assessment of freight traffic flows and harmful emissions in euro-mediterranean context: scenario analyses based on a gravity model. *Journal of Shipping and Trade*, 7(1):13, 2022.

[34] F. Glover. Tabu search: A tutorial. *Interfaces*, 20(4):74–94, 1990.

[35] F. Glover and M. Laguna. Tabu search*. In *Handbook of combinatorial optimization*, pages 3261–3362. Springer, 2013.

[36] R. Goścień and K. Walkowiak. A column generation technique for routing and spectrum allocation in cloud-ready survivable elastic optical networks. *International Journal of Applied Mathematics and Computer Science*, 27(3):591–603, 2017.

[37] R. Goścień, K. Walkowiak, and M. Klinkowski. Tabu search algorithm for routing, modulation and spectrum allocation in elastic optical network with anycast and unicast traffic. *Computer Networks*, 79:148–165, 2015.

[38] A. Gumaste and T. Antony. *DWDM network designs and engineering solutions*. Cisco press, 2003.

[39] M. Hadi and M. R. Pakravan. Resource allocation for elastic optical networks using geometric optimization. *IEEE/OSA Journal of Optical Communications and Networking*, 9(10):889–899, 2017.

[40] A. Hmaity, M. Savi, F. Musumeci, M. Tornatore, and A. Pattavina. Protection strategies for virtual network functions placement and service chains provisioning. *Networks*, 70(4):373–387, 2017.

[41] D.J. Ives, P. Bayvel, and S.J. Savory. Routing, modulation, spectrum and launch power assignment to maximize the traffic throughput of a nonlinear optical mesh network. *Photonic Network Communications*, 29(3):244–256, 2015.

[42] B. Jaumard and M. Daryalal. Scalable elastic optical path networking models. In *International Conference on Transparent Optical Networks - ICTON*, pages 1–4, 2016.

[43] B. Jaumard and Q. A. Nguyen. Interference aware RSA problem. In *International Conference on Optical Network Design and Modeling (ONDM)*, pages 1–6, 2023.

[44] Z. Jia, Q. Wu, C. Dong, C. Yuen, and Z. Han. Column generation for optimization problems in communication networks. *IEEE Network*, pages 1–8, 2022.

[45] M. Jinno. Elastic optical networking: Roles and benefits in beyond 100-Gb/s era. *Journal of Lightwave Technology*, 35(5):1116–1124, 2017.

[46] M. Jinno, B. Kozicki, H. Takara, A. Watanabe, Y. Sone, T. Tanaka, and A. Hirano. Distance-adaptive spectrum resource allocation in spectrum-sliced elastic optical path network [topics in optical communications]. *IEEE Communications Magazine*, 48(8):138–145, 2010.

[47] P. Johannisson and E. Agrell. Modeling of nonlinear signal distortion in fiber-optic networks. *Journal of Lightwave Technology*, 32(23):4544–4552, 2014.

[48] M. Johansson and L. Xiao. Cross-layer optimization of wireless networks using nonlinear column generation. *IEEE Transactions on Wireless Communications*, 5(2):435–445, 2006.

[49] M. Klinkowski, M. Pióro, M. Żotkiewicz, M. Ruiz, and L. Velasco. Valid inequalities for the routing and spectrum allocation problem in elastic optical networks. In *16th International Conference on Transparent Optical Networks (ICTON)*, pages 1–5, 2014.

[50] M. Klinkowski and K. Walkowiak. A simulated annealing heuristic for a branch and price-based routing and spectrum allocation algorithm in elastic optical networks. In *International Conference on Intelligent Data Engineering and Automated Learning (IDEAL)*, pages 290–299, 2015.

[51] M. Klinkowski, M. Żotkiewicz, K. Walkowiak, M. Pióro, M. Ruiz, and L. Velasco. Solving large instances of the RSA problem in flexgrid elastic optical networks. *Journal of Optical Communications and Networking*, 8(5):320–330, 2016.

[52] E. L. Lawler and D. E. Wood. Branch-and-bound methods: A survey. *Operations research*, 14(4):699–719, 1966.

[53] P. Lechowicz, M. Tornatore, A. Włodarczyk, and K. Walkowiak. Fragmentation metrics and fragmentation-aware algorithm for spectrally/spatially flexible optical networks. *Journal of Optical Communications and Networking*, 12(5):133–145, 2020.

[54] K. Li, P. He, and P. N. Ram Kumar. A column generation based approach for an integrated production and transportation scheduling problem with dual delivery modes. *International Journal of Production Research*, 61(16):5483–5501, 2023.

[55] A. Martin. *General mixed integer programming: Computational issues for branch-and-cut algorithms*. Springer, 2001.

[56] M. Mehrabi, H. Beyranvand, and M. J. Emadi. Multi-band elastic optical networks: Inter-channel stimulated Raman scattering-aware routing, modulation level and spectrum assignment. *Journal of Lightwave Technology*, 39(11):3360–3370, 2021.

[57] A. Mohammed and B. Jaumard. Nested column generation algorithm for the routing and spectrum assignment problem in flexgrid optical networks. In *IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*, pages 1–5, 2021.

[58] T. Nassima, B. Malika, and H. H. Eddine. An efficient genetic algorithm for solving spectrum assignment problem in elastic optical networks. In *3rd International Conference on Human-Centric Smart Environments for Health and Well-being (IHSH)*, pages 31–36. IEEE, 2022.

[59] A. M. Newman and M. Weiss. A survey of linear and mixed-integer optimization tutorials. *INFORMS Transactions on Education*, 14(1):26–38, 2013.

[60] D. M. Nguyen, A. N. Le, T. V. H. Pham, H. S. Ngo, and T. H. Dao. An efficient column generation approach for solving the routing and spectrum assignment problem in elastic optical networks. In *6th NAFOSTED Conference on Information and Computer Science (NICS)*, pages 130–135, 2019.

[61] O-RAN Alliance. Cloud architecture and deployment scenarios for O-RAN virtualized RAN, 2022.

[62] University of Texas and Y. Zhang. SNDlib, 2011.

[63] S. Park, H. G. Kim, J. Hong, S. Lange, J. H. Yoo, and J. W. K. Hong. Machine learning-based optimal VNF deployment. In *21st Asia-Pacific Network Operations and Management Symposium (APNOMS)*, pages 67–72. IEEE, 2020.

[64] P. Poggiolini. The GN model of non-linear propagation in uncompensated coherent optical systems. *Journal of Lightwave Technology*, 30(24):3857–3879, December 2012.

[65] P. Poggiolini, G. Bosco, A. Carena, V. Curri, Y. Jiang, and F. Forghieri. The GN-model of fiber non-linear propagation and its applications. *Journal of Lightwave Technology*, 32(4):694–721, 2014.

[66] L. Qu, C. Assi, M. J. Khabbaz, and Y. Ye. Reliability-aware service function chaining with function decomposition and multipath routing. *IEEE Transactions on Network and Service Management*, 17(2):835–848, 2020.

[67] L. Qu, C. Assi, K. Shaban, and M. J. Khabbaz. A reliability-aware network service chain provisioning with delay guarantees in NFV-enabled enterprise datacenter networks. *IEEE Transactions on Network and Service Management*, 14(3):554–568, 2017.

[68] H. T. Quang, O. Houidi, J. Errea-Moreno, D. Verchere, and D. Zeghlache. MAGC-RSA: multi-agent graph convolutional reinforcement learning for distributed routing and spectrum assignment in elastic optical networks. In *European Conference on Optical Communication (ECOC)*, pages 1–4. IEEE, 2022.

[69] C. Rottondi, L. Barletta, A. Giusti, and M. Tornatore. Machine-learning method for quality of transmission prediction of unestablished lightpaths. *Journal of Optical Communications and Networking*, 10:A286, 02 2018.

[70] M. Ruiz, M. Pioro, M. Zotkiewicz, M. Klinkowski, and L. Velasco. Column generation algorithm for RSA problems in flexgrid optical networks. *Photonic Network Communications*, 26:53–64, 2013.

[71] M. Ruiz, M. Zotkiewicz, L. Velasco, and J. Comellas. A column generation approach for large-scale RSA-based network planning. In *International Conference on Transparent Optical Networks - ICTON*, pages 53–64, 2013.

[72] S. Sakai, M. Togasaki, and K. Yamazaki. A note on greedy algorithms for the maximum weighted independent set problem. *Discrete applied mathematics*, 126(2-3):313–322, 2003.

[73] M. Salani, C. Rottondi, and M. Tornatore. Routing and spectrum assignment integrating machine-learning-based QoT estimation in elastic optical networks. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pages 1738–1746. IEEE, 2019.

[74] X. Shang, Y. Huang, Z. Liu, and Y. Yang. Reducing the service function chain backup cost over the edge and cloud by a self-adapting scheme. *IEEE Transactions on Mobile Computing*, 21(8):2994–3008, 2022.

[75] S. Sharma, A. Engelmann, A. Jukan, and A. Gumaste. VNF Availability and SFC Sizing Model for Service Provider Networks. *IEEE Access*, 8:119768–119784, 2020.

[76] S. Shirazipourazad, C. Zhou, Z. Derakhshandeh, and A. Sen. On routing and spectrum allocation in spectrum-sliced optical networks. In *IEEE INFOCOM*, pages 385–389, 2013.

[77] S. K. Singh, R. Singh, and B. Kumbhani. The evolution of radio access network towards open-RAN: Challenges and opportunities. In *IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, pages 1–6. IEEE, 2020.

[78] M. Srinivas and L.M. Patnaik. Genetic algorithms: a survey. *Computer*, 27(6):17–26, 1994.

[79] Statista. Number of Internet of Things (IoT) connected devices worldwide, 2020.

[80] I. Tamim, A. Saci, M. Jammal, and A. Shami. Downtime-aware O-RAN VNF deployment strategy for optimized self-healing in the O-Cloud. In *IEEE Global Communications Conference (GLOBECOM)*, pages 1–6, 2021.

[81] I. Tamim, A. Shami, and L. Ong. ML-Based strategies to optimize O-RAN VNFs for latency and reliability. In *IEEE Future Networks World Forum (FNWF)*, pages 1–7, 2023.

[82] B. Tang, Y. C. Huang, Y. Xue, and W. Zhou. Heuristic reward design for deep reinforcement learning-based routing, modulation and spectrum assignment of elastic optical networks. *IEEE Communications Letters*, 26(11):2675–2679, 2022.

[83] M. Tayyab, A. Tarar, and M. Riaz. Review of gravity model derivations. *Mathematical Theory and Modeling*, 2(9):82–95, 2012.

[84] A. Tomassilli, N. Huin, F. Giroire, and B. Jaumard. Resource requirements for reliable service function chaining. In *IEEE International Conference on Communications (ICC)*, pages 1–7, May 2018.

[85] I. Tomkos, S. Azodolmolky, J. Solé-Pareta, D. Careglio, and E. Palkopoulou. A tutorial on the flexible optical networking paradigm: State of the art, trends, and research challenges. *Proceedings of the IEEE*, 102(9):1317–1337, 2014.

[86] F. Vanderbeck. A nested decomposition approach to a three-stage, two-dimensional cutting-stock problem. *Management Science*, 47(6):864–879, 2001.

[87] F. Vanderbeck and L.A. Wolsey. An exact algorithm for IP column generation. *Operations Research Letters*, 19:151–159, 1996.

[88] M. M. R. Villamayor-Paredes, L. V. Maidana-Benítez, J. Colbes, and D. P Pinto-Roa. Routing, modulation level, and spectrum assignment in elastic optical networks a route-permutation based genetic algorithms. *Optical Switching and Networking*, page 100710, 2022.

[89] Y. Wang, J. Tang, and R. YK Fung. A column-generation-based heuristic algorithm for solving operating theater planning problem under stochastic demand and surgery cancellation risk. *International Journal of Production Economics*, 158:28–36, 2014.

[90] Y. Wu, W. Zheng, Y. Zhang, and J. Li. Reliability-aware VNF placement using a probability-based approach. *IEEE Transactions on Network and Service Management*, 18(3):2478–2491, 2021.

[91] M. Xiao and H. Nagamochi. Exact algorithms for maximum independent set. *Information and Computation*, 255:126–146, 2017.

[92] Z. Xiong, Y. C. Huang, and X. Hu. Graph attention network enhanced deep reinforcement learning framework for routing, modulation, and spectrum allocation in EONs. In *Asia Communications and Photonics Conference (ACP) and International Conference on Information Photonics and Optical Communications (IPOC)*, pages 1–6. IEEE, 2024.

[93] L. Xu, Y. C. Huang, Y. Xue, and X. Hu. Deep reinforcement learning-based routing and spectrum assignment of EONs by exploiting GCN and RNN for feature extraction. *Journal of Lightwave Technology*, 40(15):4945–4955, 2022.

[94] L. Yan and E. Agrell. Capacity scaling of flexible optical networks with nonlinear impairments. In *International Conference on Transparent Optical Networks - ICTON*, pages 1–4, 2017.

[95] L. Yan, E. Agrell, M. Nishan Dharmaweera, and H. Wymeersch. Joint assignment of power, routing, and spectrum in static flexible-grid networks. *Journal of Lightwave Technology*, 35(10):1766–1774, 2017.

[96] L. Yan, E. Agrell, H. Wymeersch, and M. Brandt-Pearce. Resource allocation for flexible-grid optical networks with nonlinear channel model. *IEEE/OSA Journal of Optical Communications and Networking*, 7(11):B101–B108, November 2015.

[97] Q. Yao, H. Yang, B. Bao, J. Zhang, H. Wang, D. Ge, S. Liu, D. Wang, Y. Li, D. Zhang, and H. Li. SNR re-verification-based routing, band, modulation, and spectrum assignment in hybrid C-C+L optical networks. *Journal of Lightwave Technology*, 40(11):3456–3469, 2022.

[98] J. Zhang, Z. Wang, C. Peng, L. Zhang, T. Huang, and Y. Liu. RABA: Resource-aware backup allocation for a chain of virtual network functions. In *IEEE Conference on Computer Communications*, pages 1918–1926, 2019.

[99] P. Zhang, X. Yang, J. Chen, and Y. Huang. A survey of testing for 5g: Solutions, opportunities, and challenges. *China Communications*, 16(1):69–85, Jan 2019.

[100] W. Zhang, S. Yin, Z. Wang, Y. Chai, and S. Huang. Routing, modulation level and spectrum assignment considering nonlinear interference in C+L+S-bands EONs. In *27th OptoElectronics and Communications Conference (OECC) and 2022 International Conference on Photonics in Switching and Computing (PSC)*, pages 1–3, 2022.

[101] Y. Zhao, Q. Zhang, Z. Li, X. Xin, R. Gao, F. Chai, Y. Tao, Q. Tian, F. Tian, D. Chen, et al. Dynamic routing, modulation, and spectrum assignment based on fuzzy logic control in elastic optical networks. *Applied Optics*, 61(1):223–230, 2022.