Designing and Optimization of Heterogeneous OTN/DWDM Networks with Intermediate Grooming

Afonso Mota da Conceição Oliveira

Thesis to obtain the Master of Science Degree in Communication Networks Engineering

Examination Committee

Chairperson: Prof. Paulo Jorge Pires Ferreira
Supervisors: Prof. João José de Oliveira Pires
            Doctor João Manuel Ferreira Pedro
Member of the Committee: Prof. Rui Jorge Morais Tomaz Valadas

October 2013
Abstract

The coexistence of channels with different bitrates in the same optical network OTN/DWDM is a possible scenario as the technological evolution comes to allow higher bitrates. In this document a study and results are presented after the development of an optimization framework for OTN networks. For this framework the intention is to study methods that design a cost optimized optical network that support different services as traffic demands with different bitrates as well as optical channels with a different bitrate. The use of intermediate grooming is also be considered where one traffic flow can be regrouped with others at any intermediary node of its path. The model considering intermediate grooming proposed yields lower network cost values but needs considerably more time to compute. Two heuristic possibilities are compared having them different behaviors. This work considers also the use of different node equipment with different architectures.

Keywords

OTN/DWDM, grooming, ILP, CapEx, intermediate grooming, heuristics
Resumo

A coexistência de canais óticos com diferentes débitos na mesma rede OTN/DWDM é um possível cenário visto que a evolução tecnológica tem vindo a permitir débitos mais altos. Neste documento é elaborado um estudo e são apresentados resultados sobre o desenvolvimento de uma ferramenta de otimização de redes OTN. A intenção desta ferramenta é estudar métodos que projetem uma rede ótica otimizada em termos de custos, tendo em conta diferentes serviços bem como canais de diferentes débitos. A possibilidade de agregação intermédia, onde o tráfego pode ser re-agregado num nó intermediário, será também considerada. O modelo que considera agregação intermédia consegue devolver valores de custo menores mas necessita de mais tempo para correr. Duas possibilidades de heurísticas são apresentadas e comparadas, tendo estas comportamentos diferentes. Este trabalho considera também o uso de diferentes arquiteturas em relação aos equipamentos usados nos nós.

Palavras Chave

OTN/DWDM, agregação, ILP, CapEx, agregação intermédia, heurísticas
## Contents

1 Introduction ................................. 1
   1.1 Motivation .................................. 3
   1.2 Objectives .................................. 4
      1.2.1 Requirements .............................. 4
      1.2.2 Parameters ................................. 4
   1.3 Main Contributions ......................... 5
   1.4 Dissertation Outline ....................... 5

2 State of the art .............................. 7
   2.1 OTN ......................................... 8
      2.1.1 The Role of OTN .............................. 8
      2.1.2 OTN Layered Structure ....................... 9
      2.1.3 Traffic Grooming ............................ 10
      2.1.4 Color and Colorless Concept ............... 10
   2.2 ROADM Architectures ....................... 11
   2.3 Models for Studying OTN/DWDM Networks .......... 14
   2.4 Intermediate Grooming ...................... 15
   2.5 Heuristic Algorithms ....................... 16
   2.6 Summary ..................................... 18

3 Architecture ................................ 21
   3.1 Integer Linear Programming (ILP) Model Implementation For Source Grooming .......... 22
      3.1.1 Previous Calculations ....................... 23
      3.1.2 Formulation ................................ 24
   3.2 Heuristic .................................... 24
      3.2.1 LP+R ...................................... 26
      3.2.2 LRH ...................................... 26
         3.2.2.A Lagrangian Relaxation ................... 26
         3.2.2.B Upper Bound Algorithm .................. 27
         3.2.2.C Multipliers’ Iteration ................... 29
List of Figures

2.1 Service and transport network ........................................ 8
2.2 Optical Transport Network architecture. [22] ................. 9
2.3 Basic mapping and multiplexing stages to accommodate a client signal into an OTU. [17] ................................................. 10
2.4 Mapping/multiplexing of ODUs into OTUs. [17] .............. 11
2.5 Implementation based on wavelength blocking with tunable filters [10] .................................................. 12
2.6 Simplified four-degree ROADM system with Wavelength Selective Switching (WSS) technology. [1] ......................... 13
2.7 Detailed Ring-to-Ring interconnection using PLC based ROADM subsystems [25] ........................................ 13
2.8 Comparison of transponder costs for SLR and MLR networks. [13] .................................................. 14
2.9 Multiplexing (grooming) and inverse-multiplexing (Virtual Concatenation (VCAT)) techniques in OTN. [21] .............. 15
2.10 Flowchart of the LP+R optimization approach. [23] .......... 17
3.1 Framework architecture ................................................... 22
3.2 Aggregations to be considered. Adapted from [17] .......... 22
3.3 Different regenerator permutations in a path ....................... 33
4.1 Comparison between network costs varying traffic volume in EON network ................................................................. 40
4.2 Comparison between the computation times for solving the ILP problem varying traffic volume in EON network ................................................................. 41
4.3 Comparison between network costs resulted from the two works using EON network ................................................................. 42
4.4 Comparison between network costs resulted from the two works using UBN network ................................................................. 42
4.5 LRH upper and lower bound values in each LRH iteration using network EON ................................................................. 43
4.6 LRH upper and lower bound values in each LRH iteration using network EON with only positive values ................................................................. 44
4.7 Comparison between implementations’ network costs varying traffic volume in EON network ................................................................. 46
4.8 Comparison between implementations’ computation times varying traffic volume in EON network ................................................................. 46
List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.9</td>
<td>Comparison between step 1 and step 4 implementations' network costs varying traffic volume in EON network.</td>
<td>49</td>
</tr>
<tr>
<td>4.10</td>
<td>Comparison between step 1 and step 4 implementations' computation times varying traffic volume in EON network with a logarithmic scale on time.</td>
<td>50</td>
</tr>
<tr>
<td>A.1</td>
<td>EON Topology</td>
<td>62</td>
</tr>
<tr>
<td>A.2</td>
<td>UBN Topology</td>
<td>63</td>
</tr>
<tr>
<td>A.3</td>
<td>GEANT Topology</td>
<td>64</td>
</tr>
<tr>
<td>A.4</td>
<td>CORONET Topology</td>
<td>65</td>
</tr>
</tbody>
</table>
List of Tables

2.1 OPU types and bit rates. [9] ......................................................... 9

4.1 Results using the different algorithms in different networks .................. 45
4.2 ILP solving restrained by computation time. ........................................ 47
4.3 Node Architecture Study for UBN. ................................................ 48
List of Abbreviations

TTV  Total Traffic Volume
LRH  Lagrangean Relaxation with Heuristics
VOA  Variable Optical Attenuators
AWG  Arrayed Waveguide Gratings
PLC  Planner Lightwave Circuit
OTM  Optical Terminal Multiplexer
WSS  Wavelength Selective Switching
SLR  Single Line Rate
MLR  Mixed Line Rate
ODTUG Optical channel Data Tributary Unit Group
SDH  Synchronous Digital Hierarchy
CapEx Capital Expenditure
VCAT  Virtual Concatenation
ILP  Integer Linear Programming
LP  Linear Programming
OCh  Optical Channel
OTS  Optical Transport Section
OMS  Optical Multiplex Section
ODU  Optical channel Data Unit
OPU  Optical channel Payload Unit
OTU  Optical channel Transport Unit
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DWDM</td>
<td>Dense Wavelength Division Multiplexing</td>
</tr>
<tr>
<td>WDM</td>
<td>Wavelength Division Multiplexing</td>
</tr>
<tr>
<td>ROADM</td>
<td>Reconfigurable Optical Add-Drop Multiplexer</td>
</tr>
<tr>
<td>O/E/O</td>
<td>Optical-Electrical-Optical</td>
</tr>
<tr>
<td>E/O/E</td>
<td>electrical-optical-electrical</td>
</tr>
<tr>
<td>VoIP</td>
<td>Voice over Internet Protocol</td>
</tr>
<tr>
<td>3R</td>
<td>Reamplification, Reshaping and Retiming</td>
</tr>
<tr>
<td>IPTV</td>
<td>Internet Protocol Television</td>
</tr>
<tr>
<td>OTN</td>
<td>Optical Transport Network</td>
</tr>
<tr>
<td>TDP</td>
<td>Traffic Demand Probability</td>
</tr>
<tr>
<td>TDQi</td>
<td>Traffic Demand Quantity Index</td>
</tr>
<tr>
<td>NSN</td>
<td>Nokia Siemens Networks</td>
</tr>
<tr>
<td>EON</td>
<td>European Optical Network</td>
</tr>
<tr>
<td>UBN</td>
<td>US Backbone Network</td>
</tr>
</tbody>
</table>
# 1 Introduction

## Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 Motivation</td>
<td>3</td>
</tr>
<tr>
<td>1.2 Objectives</td>
<td>4</td>
</tr>
<tr>
<td>1.3 Main Contributions</td>
<td>5</td>
</tr>
<tr>
<td>1.4 Dissertation Outline</td>
<td>5</td>
</tr>
</tbody>
</table>
1. Introduction

Due to huge increase in traffic demands resulting from new services of high bitrate, telecommunications backbone networks need to rely on optical solutions like Optical Transport Network (OTN). In OTN networks one traffic demand is mapped into an unit called Optical channel Payload Unit (OPU), which is still a unit of the electrical domain. After this unit is mapped into more units of the same domain, which are described in Section 2.1.2, the respective signal is converted to the optical domain and an optical channel is generated that will operate at a given wavelength. One optical signal is routed through a Dense Wavelength Division Multiplexing (DWDM) network which includes many fibers and optical nodes.

As optical signals are analogue ones they need to be regenerated to avoid the effects of physical impairments. This regeneration is a process where a signal is converted to the electrical domain and then reconverted to the optical domain again in order to correct the mentioned effects. To do so there is a need for Wavelength Division Multiplexing (WDM) transponders which are equipment capable of emitting and receiving optical signals. The mentioned conversion is called an Optical-Electrical-Optical (O/E/O) conversion and it is a process that increases significantly the network cost due to the use of transponders. Transponders have characteristics that define it’s bitrate and optical reach, which is the maximum length that one emitted signal can travel and be received by other transponder without loss of information.

These transponders are currently evolving and their cost and characteristics are changing. Each of them will have their own price and will offer a different optical reach. Also each one will transmit and receive at a given bitrate.

The optical nodes are usually Reconfigurable Optical Add-Drop Multiplexers (ROADMs) which are equipment belonging to the transport layer that have the capability to add or drop traffic to the infrastructure attached to them in the service layer. In addition they also route traffic in the transport layer. A ROADM has also the ability to route wavelengths, as so optical signals can be routed without O/E/O conversion. For this reason a ROADM is called a colorless equipment.

In an OTN network we might have nodes with a degree greater than two. For those we need a ROADM Multi-Degree that needs a more complex architecture in order to be completely colorless. Some older ROADM do not have the capability to route wavelengths to all directions.

Designing an efficient network requires the use of traffic engineering techniques that can adapt the traffic in an optimized way to have lower costs and work around network limitations. In the case of an OTN there are several variables which can change the network cost. From where and how to regenerate a signal to the bitrate of the channel used numerous choices can be made and the variables are all interconnected. For instance the bitrate of the channel used has a great impact in the need for regeneration of the respective signal. In one optical network there are traffic engineering techniques such as grooming and VCAT.

Grooming consists in aggregating different traffic flows in one optical channel in order to avoid the use of multiple optical channels. The traffic flows are then separated at the destination. There
1.1 Motivation

is also the possibility of separating and re-aggregating the flows at an intermediary node of the communication, as so it is possible to aggregate traffic from different source nodes. This process of re-aggregating the traffic will be referred in this document as intermediate grooming. The case when only grooming at the source is consider will be referred as source grooming.

VCAT consists in separating one big traffic flow into separate flows so that they can be mapped into lower bitrate [OPU]. These lower bitrate flows will be aggregated at the destination node.

This thesis is the continuation of recent works developed by Santos et al which design and optimize heterogeneous optical networks with grooming and VCAT [24] and provide one heuristic algorithm for the respective solution [23].

This work pretends to design and optimize the cost of an OTN/DWDM network which considers the transponder related cost associated to intermediate grooming of the traffic.

1.1 Motivation

With a tremendous traffic demand growth, which is caused by widespread adoption of broadband connectivity and an overuse of network applications (e.g. on-line gaming, Internet Protocol Television (IPTV), cloud storage and other cloud applications), network operators have been feeling the need to renew their core infrastructure in order to support higher bitrates.

In December of 2009, the standard ITU-T G.709 (standard for interfaces of an OTN) was reformulated in order to allow the use of 100Gb/s optical channels [9]. However in order to deploy a 100Gb/s optical channel there is the need to use advanced modulation formats (e.g. PM-DQPSK) characterized not only by their robustness to physical impairments but also their capability to improve the spectral efficiency of the optical signal. But these advantages are reflected in a more complex development of transponders whose cost-effectiveness cannot be easily predicted [21].

In addition the traffic demands variety has also increased. From Voice over Internet Protocol (VoIP) applications to the storage of a big quantity of information in data centers, due to cloud services and others, there is a big variety of services with different needs in terms of bitrate. Grooming techniques can be used in this context to aggregate the traffic in an appropriate way in order to better manage the bandwidth and improve the network efficiency.

As referred above there is the possibility of intermediate grooming. By re-aggregating traffic in an intermediary node we can combine it with other traffic flows trying to use as less optical channels as possible, hence improving the optical channels’ efficiency. However this procedure has a cost. The grooming of traffic demands must take place in electrical domain and as a consequence the number of transponders, which we are trying to minimize, must increase to support additional [O/E/O] conversions. The framework will decide where to do intermediate grooming and which flows to aggregate in order to reduce the transponders’ cost. By considering intermediate grooming the possibilities of aggregation are greatly increased which adds complexity to the
1. Introduction

problem solving.

1.2 Objectives

This section will describe the main objectives of this framework.

1.2.1 Requirements

In this work the main objective is to develop a framework to design OTN/DWDM networks with the aim of optimizing transponder related Capital Expenditure (CapEx) costs. Transponders can only be located at the nodes.

This network should be designed in order to support service types with different rates. The ones to be considered will be services with: 1.25, 2.5, 10, 40 and 100 Gb/s. Hence referred as 1G25, 2G5, 10G, 40G and 100G services, respectively.

The optical channels used will have only two different bitrates: 40 and 100 Gb/s. As so only 40 and 100 Gb/s transponders are going to be used, which are going to be referred as 40G and 100G transponders.

Three scenarios with different node architectures are to be considered. The first one considers that all nodes have colored interfaces. The second considers that there is only wavelength switching in some directions, as so some directions will need O/E/O conversion. The last one considers that there is wavelength switching in all directions.

Finally the framework should also be able to design networks that support intermediate grooming.

With the developed framework it will be possible to analyze how the network design and cost varies with different equipment cost ratios, as well as equipment with different characteristics.

1.2.2 Parameters

The network design framework should receive as input the network physical topology and traffic demands.

The framework to be developed will not have into consideration any precise equipment. Instead the difference between transponder costs and their optical reaches will be tuned by variable parameters. These parameters will be:

- $\alpha$ - Ratio between 100G and 40G technology costs, where 100G transponders are $\alpha$ times more expensive than 40G transponders;

- $\beta$ - Ratio between 40G and 100G technology optical reach, where the 40G transponders signals have $\beta$ times more optical reach than an optical signal generated by a 100G transponder.
1.3 Main Contributions

This work extends the work [24] by introducing new traffic services. Although in this thesis work VCAT is not supported, as it is in [24], there are now numerous combinations of traffic aggregation as different and smaller services are considered. Due to this it was created a new ILP formulation with a different approach.

Also the path calculation algorithm considers the fact that one link may not be considered due it’s length. Transponders have an associated optical reach and if this optical reach is smaller than one link, this link can not be considered as regeneration can only be done at the nodes. As so paths that used by 40G transponders will be different from paths used by 100G ones.

This work also extends the work [23] by introducing a different heuristic to reduce the computation time of the problem to solve. The new heuristic uses a Lagrangian relaxation applied to the formulation. A comparison of the heuristics is described in chapter [2].

This thesis work will also do a study on the impact of different node architectures in the network design and optimization. A similar work and more detailed [12] is described in Section 2.2. In this last work there is no network optimization involved though.

In this work it is considered the possibility of rearranging the traffic flows in different channels to reduce the network’s costs. An analysis on the advantages of considering intermediate grooming is going to be presented as well.

1.4 Dissertation Outline

In chapter 2 this document will present works performed in these areas, explaining also the most important concepts introduced. After this, in chapter 3 the architecture of the work will be presented by describing the formulations and algorithms introduced.

The work’s simulation results are described in chapter 4. Finally a summary and conclusion is presented in chapter 5 as well as an enumeration of possible future work.
1. Introduction
2
State of the art

Contents

2.1 OTN ................................................. 8
2.2 ROADM Architectures .............................. 11
2.3 Models for Studying OTN/DWDM Networks ...... 14
2.4 Intermediate Grooming ............................ 15
2.5 Heuristic Algorithms .............................. 16
2.6 Summary ........................................... 18
2. State of the art

This chapter presents some considerations about OTN networks and describes the state of the art in methodologies for designing and optimizing these networks. The works to be considered are works in models for studying OTN/DWDM networks, heuristic algorithms implemented in order to reduce the solving duration of an ILP problem, description of ROADM architectures and works that consider intermediate grooming.

2.1 OTN

In order to provide a better understanding of the model to calculate the network cost, a study was made about the OTN. OTN is a network composed by a set of optical network elements which are linked by optical fiber. The main intention behind its creation was to combine the benefits of Synchronous Digital Hierarchy (SDH) technology with the bandwidth expansion capabilities that characterize the DWDM technology.

2.1.1 The Role of OTN

Telecommunication networks can be divided into two networks: service network and transport network. The transport network provides paths to the service network, as so the service network can be seen as a client of the transport network. This separation can be observed in Figure 2.1.

OTN is a transport network that serves a variety of service networks. These networks can be networks like mobile networks, cable networks and data networks. Transport networks provide features and guarantees to the service network like routing, protection, supervision and capacity provisioning.
2.1 OTN

2.1.2 OTN Layered Structure

The OTN layered structure is composed of two main domains: electrical domain and optical domain. Figure 2.2 represents this domain separation.

![Figure 2.2: Optical Transport Network architecture. [22]](image)

In the electrical domain we can find three different layers OPU, Optical channel Data Unit (ODU) and Optical channel Transport Unit (OTU). These layers adapt the client signal to fixed throughput units. OPU layer is the responsible for adapting the signals bitrate. It encapsulates the client signal (e.g., SDH, Ethernet) and does any rate justification that is needed.

The OPU header in the frame is valid between the source and destination nodes of the client signal. There are different types of OPUs referred by OPU$^k$ where $k$ is the type specifier. In Table 2.1.2 we can see the bit rates for each OPU$^k$.

<table>
<thead>
<tr>
<th>OPU type</th>
<th>OPU nominal bit rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPU0</td>
<td>238/239 × 1 244 160 kbit/s</td>
</tr>
<tr>
<td>OPU1</td>
<td>2 488 320 kbit/s</td>
</tr>
<tr>
<td>OPU2</td>
<td>238/237 × 9 953 280 kbit/s</td>
</tr>
<tr>
<td>OPU3</td>
<td>238/236 × 39 813 120 kbit/s</td>
</tr>
<tr>
<td>OPU4</td>
<td>238/227 × 99 532 800 kbit/s</td>
</tr>
<tr>
<td>OPU2e</td>
<td>238/237 × 10 312 500 kbit/s</td>
</tr>
<tr>
<td>OPUflex</td>
<td>client bit rate</td>
</tr>
</tbody>
</table>

![Table 2.1: OPU types and bit rates. [9]](table)

Going to a lower layer we have the ODU. This layer’s role is to monitor the connection between the source and destination nodes of the client signal. Another role is to establish the data channel between those two nodes and protect it.

The OTU is responsible for processing the detection and error correction of an optical transmission. The OTU creates the optical channel until the next colored node.
The optical domain layers relate to the optical channel and have defined important procedures to manage the optical signals. The top layer in the electrical domain is the Optical Channel (OCh) which identifies the channel used by a connection between two non-optical ends, for instance two transponders. The Optical Multiplex Section (OMS) comes next and is the one responsible for identifying wavelengths and assigning them to the communication channels. The bottom layer is the Optical Transport Section (OTS) which is responsible for managing signal strengthening in order to avoid effects originated by physical impairments.

2.1.3 Traffic Grooming

There are also higher order and lower order OPU and ODU so that we can group different traffic demands in one optical channel reducing the number of wavelengths needed. A graphical representation of this process is presented in Figure 2.4. As it can be observed after a client signal is mapped indirectly to a lower order ODU, this one can be mapped into an OTU or grouped with other ODU by being grouped in a Optical channel Data Tributary Unit Group (ODTUG) and restarting the mapping process but with higher order units.

![Figure 2.3: Basic mapping and multiplexing stages to accommodate a client signal into an OTU.](image)

Only ODU1, ODU2, ODU3 and ODU4 can be higher order. And only these type of units lower order can be mapped into an OTU. An ODU2e has to be multiplexed into a higher order ODU3 or ODU4 and an ODU0 to one of the higher order ODU. However a recent supplement defines the use of a OTU2e.

Traffic grooming is a traffic engineering process that consists in aggregating traffic demands so that the bandwidth of an optical channel can be used more efficiently. Different possibilities of aggregation within the OTN scope can be found in Figure 2.4.

2.1.4 Color and Colorless Concept

In an OTN there is a big set of network elements. There are two main types that characterize the equipment: color and colorless equipment. The main difference is that color equipment has the capability to do optical-electrical conversion contrary to colorless equipment. To better understand the color/colorless concept and the advantages/disadvantages of an optical-electrical conversion consider the following example: for amplifying an optical signal we could use either an amplifier or a regenerator. An amplifier, as the name says, only amplifies the optical signal, so it's
cheaper but not only it also amplifies the signal noise as it does not do any type of signal reshaping or frame re-timing. On the other hand we have a regenerator that can do a Reamplification, Reshaping and Retiming \(3\text{R}\) regeneration, but has the downside of being a more expensive equipment as it does optical-electrical and electrical-optical conversion.

Even though a node may be colorless, hence equipped with optical bypass equipment, some of the traffic transiting that node may still need \(3\text{R}\) regeneration after some distance. To this maximum distance we call optical reach. Many factors affect the optical reach such as the launched power of the signal and the modulation format of the signal [7] which are determined by the WDM Transponders on the edge of the communication. Of course even random physical impairments in the channel can degrade signal quality and hence shortening the optical reach, but they are not so easily predictable. In order to optimize the CapEx of the network we need to know how to use regeneration wisely.

A regeneration can be done in the middle of a link or at a node. If we do the regeneration in the middle of the link we have use an Optical Terminal Multiplexer (OTM), which is used to split the wavelengths, as we can only regenerate wavelengths separately. But node equipment already has the need to split the optical signal in order to add and drop wavelengths or switch their path. Different wavelengths have different sources and destinations as well as different WDM Transponders that transmit signals with different optical reach. As a result locations where many wavelengths need regeneration at the same place are not so common. A regeneration at the node seems much more effective as there no need for extra OTMs in order to regenerate a small amount of wavelengths.

### 2.2 ROADM Architectures

ROADM are equipment capable of adding and dropping traffic to their tributary interfaces. The tributary interfaces are the ones that connect the Transport Layer with the Service layer. They are...
2. State of the art

connected to the infrastructure in the Service Layer local to the node.

There are ROADM with different degrees. A degree is another term for the number of switching directions. A two-degree ROADM is a simple case to analyse: the traffic is flowing in both directions and the equipment decides which are the wavelengths that need to be extracted or inserted in the fiber. When a ROADM has more than two degrees there is a need to switch the path of each wavelength in different directions.

ROADM technologies have passed through different generations. The existing technologies include wavelength blocking, Planner Lightwave Circuit (PLC) and WSS, which has become the dominant technology [18].

First generation ROADM used wavelength blocking technology which is the simplest. An example is illustrated in Figure 2.5. There are filters which select the wavelengths to be dropped. Followed by a wavelength blocker which blocks the wavelengths that were dropped so that those wavelengths can be used by the traffic that was added [10].

![Figure 2.5: Implementation based on wavelength blocking with tunable filters](image)

Second generation ROADM used PLC technology. The functional technologies of the PLC-based ROADM depend on Arrayed Waveguide Gratings (AWG), thermo-optic switches, Variable Optical Attenuators (VOA) and tap monitors that are used for monitoring power in specific channels. PLC-based ROADM have attracted the most attention because their overall manufacturing approach is more efficient and simpler [25]. Hence being cheaper equipment.

Finally we have ROADM with WSS technology. WSS is an advanced optic fiber module that can select individual wavelengths from multiple input fibers and switch these to a common output fiber (Nx1 WSS). These modules can also be configured in the other direction so that individual wavelengths on a common input fiber can be selectively switched to any of multiple output fibers (1xN WSS) [6]. An example on how the WSS technology is used in a ROADM can be seen in Figure 2.6. As you can observer the same amount of WSS modules as the node degree is required. The Mux/Demux connect to the add/drop interfaces respectively. The WSS modules required are from the type Nx1 where N is the same value as the node degree: one input comes from the Mux connected to the add interfaces an the other inputs come from the other fibers.

PLC-based, as well as wavelength blocking ROADM do not facilitate true optical branching,
2.2 ROADM Architectures

Figure 2.6: Simplified four-degree ROADM system with WSS technology. [1]

in which any wavelength can be directly routed to any desired port without the need to perform optical-electrical conversions. This type of node architecture will be hence referred as Fixed Node Architecture and Flexible Node Architecture to nodes with ROADM with WSS technology.

However these ROADM can still take part as nodes in a mesh network. In [25] Sayeed et al implemented a network structure where two PLC-based ROADM are used to implement a four-degree node. This network's physical topology is described in Figure 2.7. This type of architecture has the downside of needing regeneration when a wavelength needs to go from one ring to another. This node has only two directions that enable optical bypass.

Figure 2.7: Detailed Ring-to-Ring interconnection using PLC based ROADM subsystems [25]

In [12] Serge and Vusirikala studied network planning with the same difference in node architecture as in my thesis work. They consider as network elements PLC-based ROADM which can only be two-degree, ROADM with WSS technology which can have four or more degrees and ROADM that always have optical-electrical conversion.
2. State of the art

They evaluated the implications of different ROADM systems implementations on network architecture and analysed the impact of service type, network size, nodal connectivity and capacity growth on different ROADM networks.

2.3 Models for Studying OTN/DWDM Networks

In this section some works done in optimizing traffic grooming are going to be described. Most of the works use an ILP model in order to get the optimal results.

In [13] Nag and Tornatore developed an ILP to study the cost-effectiveness of using different optical channel bitrates: Mixed Line Rate (MLR). In this work the authors considered the separate use of 10G, 40G or 100G (Single Line Rate (SLR)) optical channels comparing their costs to a network where the three types of channels were used.

The authors concluded that total network costs in terms of regenerators could be lessened if we consider a MLR instead of a SLR network. In Figure 2.8 we can observe their ILP model results for SLR and MLR networks. As you can see networks where different rate optical channels are used tend to return cheaper solutions. As it is the case of this work. Future work by the authors considered also different types of modulation techniques [14].

![Figure 2.8: Comparison of transponder costs for SLR and MLR networks.][13]

In [26] Scheffel et al create a model to minimize the network cost considering costs of not only transponders, but also the muxponders needed for the grooming capabilities.

Recently there was a work done by J. Santos et al [24] that optimizes the network cost in terms of regenerators. The network designing framework developed supports only 40G or 100G equipment as well as 40G and 100G services.

They introduced an ILP model to get the minimum cost results which takes into account grooming and VCAT techniques. As in this work the traffic demands considered are services of 40Gb/s and 100Gb/s, there is only two possibilities of grooming and one of VCAT. This is illustrated in Figure 2.9 where we can see the different possibilities of these traffic engineering methods used in this work. A study was made also about the impact of the use or not of grooming or VCAT techniques.
2.4 Intermediate Grooming

As the formulation is link-path based there is a need to pre-calculate paths for each node pair. In this work 5 candidate paths per node pair were calculated, computed with a k-shortest path algorithm aiming to minimize the hop-count.

With the framework developed the authors studied the network regenerators cost varying certain parameters related to the regenerator relative costs and optical reach. The channel capacity was also considered. A study was made also about the impact of the use or not of grooming or VCAT techniques.

As to regenerators relative costs the authors considered an $\alpha$ value which represented the cost relation between 100G and 40G interfaces. A unitary cost is assumed for 100G transceivers and dividing it by $\alpha$ we get the value for 40G transceivers. The value $\alpha$ is varied between 1 and 4.

In terms of the interfaces optical reach a fixed value is set for 100G interfaces. A value $\beta$ sets the difference in reach between the optical reach of the interfaces. For the study purposes the authors consider the value of $\beta$ either 1 or 1.4 (40% reach increase for 40G interfaces in comparison with 100G).

2.4 Intermediate Grooming

In [15] Necker et al describe a routing algorithm that aggregates the traffic depending on the network usage. It consists in deciding if a new connection shares one of the lightpaths being used, creates a new lightpath for that connection or uses different lightpaths (used or created) to reach the destination. Although this is a routing algorithm and the aim is not to design a network for a static set of traffic demands, it is possible to create a heuristic algorithm based in this work’s implementation.

In [29] Zhu and Mukherjee formulate an ILP to reach their objective which is mainly maximizing the total throughput. The network is already designed and has a fixed number of equipment. The authors focus in using it in the most efficient way possible which is different from this thesis work, but the overall architecture of the ILP can be used.

A traffic demand is carried in a wavelength which can be routed through one or more lightpaths. What the authors do is to create variables for each lightpath and for each lightpath there are
variables for each wavelength that passes through it. The lightpaths are only identified by its ends so the wavelengths can go through different nodes while being attributed to only one lightpath. This variable representation and how these variables are organized can be also used to a cost minimization formulation.

Some works were made on multi-granular networks which formulations or approaches can be adapted in this work as it happens in [11]. Instead of having one traffic demand variable per path, the authors have one traffic demand variable per path and per wavelength used. Which increases the number of variables used but allows to consider a wider set of scenarios.

2.5 Heuristic Algorithms

In order to do an accurate optimization an ILP model is the best approach. However solving ILP starts to be rather time consuming when our network gets bigger. Considering all the nodes, with all the traffic demands and grooming possibilities our system will start to have a lot of variables and constraints. In order to reduce the time spent in the ILP solving approximations or techniques are needed.

In [19] Rahman et al propose a solution using a technique called branch, price and cut which is a branch and bound approach combined with column generation and cutting plane methods.

The branch and bound technique is used to solve ILP problems which consists in recursively solving Linear Programming (LP) relaxations branching sets of variables’ possible values, using extra constraints, in a tree structure. Each node of the tree structure is a problem with a subset of values of the parent’s set. For each node problem an upper and a lower bound are calculated and if one node’s lower bound is greater than other node’s upper bound all of it’s subsets are invalid and hence not calculated [2].

The cutting plane method in an ILP model solving consists in adding constraints in order to reduce the possible variable solution. Usually the cutting planes are used as an inequality defined by the requirement of integer variables. The column generation is a method that allows us to reduce the variables’ values we need to consider by finding the most important variables to the final solution. Branch and Price methods are useful when the number of variables is high compared to the number of constraints as in some problems most of the variables in one constraint end up with the value zero.

If it is a minimization problem the lower bound can be obtained solving the formulation with the respective constraints as a LP problem, which is the method used in this work. For calculating the upper bound the authors calculate the result of the objective function substituting the variables of the current node’s set by integer values that return a feasible solution. The closer the values are to the optimal solution the more effective is the upper bound.

One of the methods is presented by Santos, J. et al [23] and Pedro, J. et al [16] in two works
2.5 Heuristic Algorithms

also about designing and optimization of OTN/DWDM networks. They do an LP relaxation that will be solved in a cycle where they progressively add constraints to the variables until all the resulting variables are integer.

This LP relaxation was done in order to continue the work done in [24]. The LP problem formulation is exactly the same as the one developed for the ILP model previously done. This cyclic strategy, named LP+R, is depicted in the flowchart of Figure 2.10.

As you can see in the flowchart, the strategy consists in solving the LP formulation in each cycle until the resulting variables are all integer. There is a list \( L \) which the purpose is to be filled with rounded values of the variables in order to add the constraints. In step 1 while formulating the problem these constraints are added. After the LP problem solving in step 2 the framework checks if all the resulting variables are integer. If they are the problem was solved, if not the cycle would continue to step 3.

In step 3 the resulting variables are analyzed and a rounding distance \( \Delta \) is calculated. This rounding distance is the difference between the variable ceiling value and the variable value itself. This is calculated so that in step 4 the non-integer variable with the shortest rounding distance to its ceiling value can be selected. This ceiling value is added to the list \( L \) in order to add a constraint in the next iteration. For instance, if variable \( x \) is the one selected in step 4 with a rounding value \( a \) on the next LP formulation there will be a \( x \geq a \) constraint.

![Figure 2.10: Flowchart of the LP+R optimization approach. [23]](image)

In [27] Sridharan et al made a study on designing a survivable WDM network with an ILP model and then proposed a heuristic algorithm based on a LP relaxation technique.

In the LP relaxation technique they used two steps. The first step was a preprocessing step where they identified the possible backup routes for each node pair and also assign to it a set of
2. State of the art

wavelengths. The second step is to formulate the problem. Combining constraints the authors managed to force some variables to have a binary value as a result.

Not only did they develop an LP relaxation technique, but they also explored two ways to reduce the ILP problem size. One of them they called it Demand Normalization Technique. What they did was to normalize the demand sets by finding the greatest common divisor for all the demand requests, dividing each demand set by that factor. This resulted in a scaled-down version of the original problem, which was less difficult to solve.

In [3] Farahmand and Iue do not propose an ILP formulation at all. The authors objective is to study different traffic grooming algorithms in WDM mesh networks with dynamic traffic patterns so that there is no request blocking. They define a routine that consists in placing one demand in the network by calculating a route for the lightpath and then allocating the necessary wavelengths taking into account the ones already occupied.

This work alerted me for the possibility of not using an ILP model. If a sufficient accurate algorithm for placing all the demands in order to reduce the network costs is designed (obeying to specific criteria), a more time efficient solution can be implemented.

In [11] Lee et al combined Lagrange relaxation with some extra heuristics, which is called Lagrangean Relaxation with Heuristics (LRH), in order to more efficiently resolve routing and wavelength assignment for multi-granular optical WDM networks. The main algorithm can be seen as a branch and bound structure, but instead of using branching the Lagrange multipliers are iterated which will result in different lower and upper bound calculations.

Lagrangean relaxation is method used in not only ILPs but also in other cases. Applying it in an ILP brings the advantage to relax constraints that complicate the problem solving. The relaxing process consists in lifting the constraints to the objective function multiplying them by Lagrange multipliers. These Lagrange multipliers can be iterated according to the lower and upper bound calculations. In order to calculate this upper bound value one could use a Lagrangean dual or similar to this case a heuristic algorithm [5].

In order to find a lower bound the Lagrange relaxation technique is used. The heuristic algorithm in the authors LRH model comes for calculating the upper bound. The heuristic algorithm considers values obtained in one of the sub problems of the Lagrange relaxation in order to reduce it’s complexity. Then, repetitively using the Dijkstra’s shortest path algorithm for all the traffic demands, it allocates the necessary wavelengths. The output of this algorithm will return an upper bound.

2.6 Summary

One traffic demand that is routed through an OTN needs to be grouped in one OPU. These can be grouped ultimately to OTUs with higher bitrate to better fit optical channels with bigger capacity.
so that we can reduce our total network cost. Re-arranging the traffic into separate channels in the middle of the path is possible and, well used, may result into smaller costs.

In order to reduce the network cost we need to find where to regenerate the signals and which optical channel to use in order to minimize the network costs. Channels with higher capacity can groom more traffic in one wavelength but usually have smaller optical reaches.

ROADM are important equipment to add-drop and route traffic. First and second generation ROADM can only provide optical bypass from one fiber to another, needing optical-electrical conversion in order to route traffic to a third fiber. Third generation ROADM can provide optical bypass to several fibers.

As the conditions that have influence in the network cost are all connected and multiple combinations exist, solving one ILP problem is possibly the best way to get the optimal value of the network. However ILP with a great amount of variables take time to solve and so the use of heuristic algorithms or techniques to reduce the solving time are encouraged.
2. State of the art
3 Architecture

Contents

3.1 ILP Model Implementation For Source Grooming . . . . . . . . . . . . . . . . . . 22
3.2 Heuristic . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 24
3.3 Node Architectures . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 30
3.4 ILP Model Implementation For Intermediate Grooming . . . . . . . . . . . . . . 31
3.5 Summary . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 35
3. Architecture

The main architecture of the framework is represented on Figure 3.1.

![Figure 3.1: Framework architecture.](image)

The framework receives as input the network topology and a set of traffic demands, after this it will calculate 5 different paths between each node pair \( sd \) and for each channel type: 40G and 100G. There can only be regeneration at the nodes so all links bigger than a channel’s optical reach are discarded.

### 3.1 [LP] Model Implementation For Source Grooming

In order to adapt our traffic demands an OPU0, OPU1, OPU2, OPU3 and OPU4 can be used to map 1G25, 2G5, 10G, 40G and 100G, respectively. As in this work the optical channels’ bitrates to be considered are 40Gb/s and 100Gb/s all the traffic as to be aggregated either into one OTU3 or one OTU4. As so the possible grooming values to be considered are represented in Figure 3.2.

![Figure 3.2: Aggregations to be considered. Adapted from [17].](image)

As referred before, the framework developed does not support VCAT. The ILP formulation implemented is a link-path formulation as so a set number of different paths will be calculated for each source/destination node pair with traffic demands.
Each path with a certain type of channel will already have an associated cost when we implement the formulation. Using a node-link approach the regenerator costs must be included in the formulation itself increasing in this way the number of variables although it would consider more path possibilities. Also it fits better in the whole framework architecture if a link-path approach is used as when calculating paths and costs we can take into account what node architecture are we using. As so we can use the same formulation for all the cases.

After calculating the paths the framework will calculate the cost associated to each path. This cost will depend on the node architecture of the network. There is no paths’ costs calculation if it is intended to use intermediate grooming as the ILP formulated to solve this problem considers separated paths.

### 3.1.1 Previous Calculations

Before formulating the ILP, the framework will need to do some previous calculations. It will need to calculate different paths for each node pair and the values of all the $C_{sd,k}^{ch}$ parameters (See Formulation 1).

Five paths will be calculated using a k-shortest path algorithm minimizing the total path length as each link has it’s length as weight. As we need to regenerate after the signal travels some length we should minimize it. The algorithm to use will be Yen’s algorithm described in [28] which consists in applying successively Dijkstra’s algorithm removing different links in each iteration. Links which are longer than the optical reach of a given optical channel type $ch$ will not be considered.

Different paths is not the only thing that should be calculated before the ILP solving. There is the need to know which is the cost of which path, that is, how may regenerators will it need depending on the technology used (40G or 100G).

Also it is needed to know how many optical channels will a set of traffic demands of the same source/destination node pair need in order to calculate a maximum value for a $n$ index (See Formulation 1).

The following parameters regarding regenerators’ characteristics will be used to calculate necessary values:

- $D_{ch}^{max}$ - optical reach of a transponder of the type $ch$;
- $C_{ch}^{T_Rx}$ - cost of a regenerator of the type $ch$.

For calculating the $C_{ch}^{sd,k}$ value we will need first to count the number of regenerators the path will use: $Reg_{ch}^{sd,k}$. So then we can calculate $C_{ch}^{sd,k} = Reg_{ch}^{sd,k} C_{ch}^{T_Rx}$. For each path $k$ between a node pair $sd$ using one type of transponder $ch$ a number of needed regenerators must be calculated.
3. Architecture

In the scope of this work the regenerators can only be placed at the nodes. The main idea is to put a regenerator in the node before the optical reach of the current lightpath is reached. Taking this into consideration a simple algorithm to calculate the number of regenerators was made and is described in Algorithm [3.1]. In this algorithm the function \( \text{length} \) returns the length of a link, \( \text{path} \) represents the path \( k \) and \( \text{link} \) represents each of the links of the path considered.

**Algorithm 3.1 Calculate \( R_{sd,k}^{ch} \)**

```plaintext
length_sum ← 0
reg_count ← 0
for link ∈ path do
    length_sum ← length_sum + length(link)
    if length_sum > D_{ch}^{max} then
        reg_count ← reg_count + 1
        length_sum ← length(link)
    end if
end for
Reg_{ch} ← reg_count
```

### 3.1.2 Formulation

The ILP model formulation to solve this problem is described in Formulation 1. Equations (3.2) to (3.5) represent grooming constraints. In equations (3.2) and (3.3) all the grooming possibilities are considered for each type of channel. Each service type demand has its own weight on an optical channel and hence the multipliers, for instance a 40G service in a 100G channel uses \( \frac{40}{100} = 0.4 \) of the channel’s bandwidth. The equation (3.4) represents one optical channel rate limitation, i.e., one 40G optical channel can not carry five 10G services as the total rate (50Gb/s) is superior to 40Gb/s.

The condition where, for a set of values \( s, d, ch \) and \( k \), \( x_{ch,1}^{sd,k} = 1 \) and \( x_{ch,2}^{sd,k} = 0 \) will give the same final solution as where \( x_{ch,1}^{sd,k} = 0 \) and \( x_{ch,2}^{sd,k} = 1 \). As so there is no need to check both sets of values which is the purpose of equation (3.5).

Equation (3.6) represents the traffic constraints. For each service type demand between each node pair the sum of all the \( t_{u,ch,n}^{sd} \) variables should be equal to the respective traffic demand value.

Equation (3.7) represents the link capacity. The equation simply prevents a link from carrying more optical channels than the ones permitted.

### 3.2 Heuristic

An ILP problem tends to have an exponential growth in computation time when more and more variables are considered. In order reduce this computation time and its growth with variables the use of a heuristic is needed.
### Formulation 1 ILP formulation considering only source grooming.

#### Indexes

- **sd**: Source and Destination node pair of a traffic demand
- **ij**: Link between nodes **i** and **j**
- **ch**: Bitrate of the channel used (\{(ch)\)G)
- **u**: Type of service identified by the OPU{u} unit with the same bitrate
- **n**: The *n*th optical channel which has the same source and destination going through the same path and using the same type of regenerators

#### Input Parameters

- **T_{sd}^u**: Traffic demands between the node pair **sd** with the rate of an OPU{u}
- **C_{ch}^{sd,k}**: The total regenerator related costs of a path **k** between the node pair **sd** using a channel of the type **ch**
- **W**: Maximum number of optical channels a link can support
- **δ_{ij}**: List of values that are 1 if the *k*th path between the node pair **sd** passes through link **ij** and otherwise

#### Variables

- **x_{ch}^{sd,k,n}**: Binary variable which represents the existence or not of the *n*th optical channel from node **s** to node **d**, that is routed through the *k*th path of bitrate **ch**
- **t_{sd,u,ch,n}**: Integer value which represents how much of a service **u** from node **s** to node **d** goes through the *n*th channel of type \{(ch)\)G (can have a value between 0 and the respective **T_{sd}^u**)

#### Objective

\[
\min z = \sum_{ch} \sum_{sd} \sum_{k} \sum_{n} \left[ c_{ch}^{sd,k} x_{ch}^{sd,k,n} \right]
\]  

(3.1)

#### Constraints

\[
\sum_{k} x_{100,n}^{sd,k} \geq \frac{1}{100} (100 t_{1,100,n}^{sd} + 40 t_{2,100,n}^{sd} + 10 t_{3,100,n}^{sd} + 2.5 t_{4,100,n}^{sd} + 1.25 t_{5,100,n}^{sd}), \forall sd \forall n
\]  

(3.2)

\[
\sum_{k} x_{40,n}^{sd,k} \geq \frac{1}{40} (40 t_{1,40,n}^{sd} + 10 t_{2,40,n}^{sd} + 2.5 t_{3,40,n}^{sd} + 1.25 t_{4,40,n}^{sd}), \forall sd \forall n
\]  

(3.3)

\[
\sum_{k} x_{ch,n}^{sd,k} \leq 1, \forall ch \forall sd \forall n
\]  

(3.4)

\[
x_{ch,n-1}^{sd,k} \geq x_{ch,n}^{sd,k}, \forall ch \forall ch \forall sd \forall n \setminus \{1\}
\]  

(3.5)

\[
\sum_{n} \left( t_{OPU,100,n}^{sd} + t_{OPU,40,n}^{sd} \right) = T_{OPU}^{sd}, \forall sd \forall OPU \in \{0,1,2,3,4\}
\]  

(3.6)

\[
\sum_{sd} \sum_{k} \sum_{n} \left( x_{40,n}^{sd,k} + x_{100,n}^{sd,k} \right) \delta_{ij} \leq W, \forall ij
\]  

(3.7)
3. Architecture

As referred in Section 2.5 a heuristic called LP+R was developed to get an approximated value of an ILP problem to design and optimize an optical network. This heuristic is described in [23]. In this work this heuristic was implemented to fit the formulation presented in Section 3.1.2.

Another heuristic was implemented in this work using as basis the work [11]. This heuristic consists in using a Lagrangian relaxation in order to calculate a lower bound of the problem and using the values obtained in the relaxation in an algorithm to calculate an upper bound. The authors called this implementation LRH. It is a good option as this work’s formulation has constraints that bring a great benefit by being relaxed as it will be explained in Section 3.2.2.A.

3.2.1 LP+R

The LP+R heuristic was implemented in the same way as in [23]. For a detailed description of this heuristic please consult Section 2.5.

3.2.2 LRH

By analyzing the previous formulation a conclusion was reached that the best constraints to relax were (3.3) and (3.2) as they are the constraints that connect the two types of variables. Not only each of the constraints may end up with many variables, but also, as we separate the variables, we can look at the problem as two separate problems.

The constraint (3.7) will also be relaxed in order to provide a faster calculation of the variables $x_{sd,k}^{ch,n}$.

3.2.2.A Lagrangian Relaxation

Constraints (3.3) and (3.2) are relaxed and each constraint is associated with the respective Lagrange multiplier also constraints (3.7) are relaxed and each constraint is associated with the respective $\lambda_{ij}$ multiplier. The resulted objective function to minimize after the relaxation is presented in (3.8).

$$
z = \sum_{ch} \sum_{sd} \sum_{k} \sum_{n} \left( C_{ch}^{sd,k} - \lambda_{sd,n,ch} + \sum_{ij} \delta_{ch,n,ij}^{sd,k} \lambda_{ij} \right) x_{ch,n}^{sd,k} + \sum_{ch} \sum_{sd} \sum_{n} \sum_{u} \left( B_{ch}^{u} \lambda_{sd,n,ch} \right) t_{u,ch,n} - \lambda_{ij} W 
$$

(3.8)

In this equation $B_{ch}^{u}$ represents the division of the bitrate of a service that is mapped into an OPU{$u$} by a channel's bitrate $ch$. For example $B_{40}^{1} = \frac{2.5}{40}$ and $B_{100}^{3} = \frac{40}{100}$. This is used to generalize the divisions presented in (3.3) and (3.2). If the channel’s bitrate is bigger than the bitrate of the OPU that carries the traffic this value is zero as there is no possibility of that channel carrying that service. The only case is $B_{40}^{1} = 0$ as an OPU4 has the bitrate of 100Gps and the channel can only support 40Gps.
3.2 Heuristic

As there are no constraints that combine the two types of variables there is the possibility of separating this problem into two: $Z_1$ and $Z_2$ as can be observed in equations (3.9) to (3.11).

$$z = Z_1 + Z_2 - \lambda_{ij}W.$$  \hspace{1cm} (3.9)

$$Z_1 = \sum_{ch} \sum_{sd} \sum_{k} \sum_{n} \left( C_{ch}^{sd,k} - \lambda_{sd,n,ch} + \sum_{ij} \delta_{ch,n,ij}^{sd,k} \right) x_{ch,n}^{sd,k}$$  \hspace{1cm} (3.10)

$$Z_2 = \sum_{ch} \sum_{sd} \sum_{n} \sum_{u} \left( B_{ch}^{u} t_{u,ch,n} \lambda_{sd,n,ch} \right)$$  \hspace{1cm} (3.11)

In this formulation $Z_1$ will be restricted to the grooming constraints that were not relaxed: (3.4) and (3.5). $Z_2$ will be restricted to the traffic constraints: (3.6). By minimizing $Z_1$ and $Z_2$, $z$ will also be minimized.

$Z_1$ is easy to solve by an algorithm obtaining the same value as solving the respective ILP problem. This can be done using the coefficient of the $x_{ch,n}^{sd,k}$ variable: $C_{ch}^{sd,k} = C_{ch} - \lambda_{sd,n,ch} + \sum_{ij} \delta_{ch,n,ij}^{sd,k} \lambda_{ij}$. For variables $x_{ch,n}^{sd,k}$ grouped by the same value of $sd, ch$ and $n$, differing only in the path $k$, the group variable that is associated with the lower final value of $c_{ch,n}^{sd,k}$ will be 1 and the others 0 due to constraint (3.4). There is also the possibility that for one group all $c_{ch,n}^{sd,k}$ values are positive. If this is the case all the variables from the group will be 0 as it is the set of values that minimizes the objective function. The Algorithm 3.2 describes this process.

The algorithm consists in calculating the $c_{ch,n}^{sd,k}$ value corresponding to each $x_{ch,n}^{sd,k}$ variable. After calculating the value $c_{ch,n}^{sd,k}$ for each group of $x_{ch,n}^{sd,k}$ variables that differ only in the path $k$, we check which is associated with the minimum value of $c_{ch,n}^{sd,k}$. This will be the path chosen by the respective channel unless all $c_{ch,n}^{sd,k}$ values turn out to be positive. If all values for similar variables that differ only in $k$ are positive all variables should be 0 to minimize $Z_1$.

In order to get all paths between the nodes $s, d$ using channels of bitrate $ch$ it is used the term traffic_paths[s, d][ch]. Following this notation, to get all the links between two nodes $i, j$ that belong to a path $k$ it is used the term traffic_path[s, d][ch][k].

On the other hand $Z_2$ would be solved as a LP problem. As an upper bound algorithm is always needed to connect the variables, the processing time needed to solve it as an ILP is not required.

3.2.2.B Upper Bound Algorithm

The upper bound algorithm consists in using the Lagrange multipliers to fill the traffic in optical channels. Each channel used as an attributed variable $x_{ch,n}^{sd,k}$ which minimization depends on
3. Architecture

Algorithm 3.2 Solve $Z_1$

\begin{algorithm}
\begin{algorithmic}
\State \textbf{for all} $(s, d, n) \in \lambda_{s, d, n, c, h}$ \textbf{do}
\State $\text{min} \leftarrow \text{INT\_MAX}$
\State $\text{all\_positive} \leftarrow \text{true}$
\State $\text{minimizer} \leftarrow -1$
\ForAll {$k \in \text{traffic\_paths}[s, d][c, h]$}
\State $z_1 \leftarrow C_{s, d, k} - \lambda_{s, d, n, c, h}$
\ForAll {$ij \in \text{traffic\_path}[s, d][c, h][k]$}
\State $z_1 \leftarrow z_1 + \lambda_{ij}$
\EndFor
\If {$z_1 < 0$}
\State $\text{all\_positive} \leftarrow \text{false}$
\EndIf
\If {$z_1 < \text{min}$}
\State $\text{minimizer} \leftarrow k$
\State $\text{min} \leftarrow z_1$
\EndIf
\EndFor
\ForAll {$k \in \text{traffic\_paths}[s, d][c, h]$}
\If {$\text{all\_positive}$} \textbf{and} $k == \text{minimizer}$
\State $x_{s, d, k} \leftarrow 1$
\Else
\State $x_{s, d, k} \leftarrow 0$
\EndIf
\EndFor
\EndFor
\end{algorithmic}
\end{algorithm}

the calculation of $c_{s, d, k}^{s, d, k}$, By organizing these values by ascending order of the total result and associating each result to a list of respective variables $x_{s, d, k}^{s, d, k}$ that have that result we can choose the channels that will be filled with traffic first.

When calculating $Z_1$ it is stored the value of each $c_{s, d, k}^{s, d, k}$ different result, associating it with the respective $x_{s, d, k}^{s, d, k}$ variable. At the end it is obtained the structure $c_{values}$ that relate the different values obtained to the list of variables which end up with the respective value as different $c_{s, d, k}^{s, d, k}$ expressions can result in equal values.

The upper bound algorithm is described in algorithm 3.3. The main idea is to iterate the $x_{s, d, k}^{s, d, k}$ variables obtained in the $c_{values}$ structure. As this structure was ordered, the variables that have associated the lower $c_{s, d, k}^{s, d, k}$ values are considered first.

The $fill\_traffic$ method gathers the traffic demands from the node pair $s, d$ and aggregates traffic in a total bitrate equal or lower to the channel’s bitrate $c, h$. This method returns the sum of traffic bitrate aggregated to the variable $fill\_info$, if this variable ends up with the value 0 it means that there is no more traffic available that fits in the channel of bitrate $c, h$.

If the path $k$ chosen contains links that have more than $W$ channels passing by, there is a need to find another path through the $find\_new\_path$ method. In the actual implementation there are more statements such as counting the number of channels passing by each link and restrict the variable attribution so that for similar variables, with different paths only, one is selected to have
3.2 Heuristic

Algorithm 3.3 Upper Bound Algorithm

\begin{algorithm}
\begin{algorithmic}
\ForAll{$x_{sd,k,\text{ch,n}} \in \text{c_values}$}
\State $\text{fill_info} \leftarrow \text{fill_traffic}(\text{ch}, \text{trafficList}[s][d])$
\If{$\text{fill_info} \neq 0$}
\State $\text{alternate_path} \leftarrow -1$
\ForAll{$\text{link} \in \text{traffic_path}[s,d][\text{ch}][k]$}
\If{$\text{is_full}(\text{link})$}
\State $\text{alternate_path} \leftarrow \text{find_new_path}(s,d,\text{ch})$
\EndIf
\If{$\text{alternate_path} \neq -1$}
\State $\text{break}$
\EndIf
\EndFor
\EndIf
\EndFor
\If{$\text{alternate_path} == -1$}
\State $x_{\text{ch,n}} \leftarrow 1$
\Else
\State $k_2 \leftarrow \text{alternate_path}$
\State $x_{\text{sd,k}_2,\text{ch,n}} \leftarrow 1$
\EndIf
\EndIf
\end{algorithmic}
\end{algorithm}

the value 1 in order to don’t contradict constraint (3.4).

Also in the implementation the new path is not found during the main iteration. The links which are overused are identified and after the main iteration there are 10 iterations to try and relocate the traffic until there are no overused links. If after 10 iterations no arrangement is found it is assumed that the network planning is not possible. However presenting the algorithm as it is presented in algorithm 3.3 seemed more appropriate and simple to explain the main idea.

3.2.2.C Multipliers’ Iteration

In order to iterate the Lagrange multipliers it was chosen upon using the Subgradient Method at least in a first stage. This method doesn’t guarantee that one iteration proves to be better than the last but guarantees that eventually the method will converge \[20\].

The equations used to iterate the multipliers are presented in equations (3.12) and (3.13). They both use in their calculations the subgradient respective to the iteration $s^u$ represented in equation (3.14). Note that LP problems are usually referred also as $Ax \leq b$ where $x$ is an array with all the variables, $A$ a matrix with the multipliers of each variable in each constraint and $b$ a vector with the constants in each constraint.

\begin{align*}
\lambda_{sd,n,\text{ch}}^{u+1} &= \lambda_{sd,n,\text{ch}}^u + s^u \frac{UB^u - LB^u}{\|s^u\|^2} \quad (3.12) \\
\lambda_{ij}^{u+1} &= \lambda_{ij}^u + s^u \frac{UB^u - LB^u}{\|s^u\|^2} \quad (3.13)
\end{align*}
3. Architecture

\[ s^u = b - Ax^u \] (3.14)

As for a stopping condition the idea was to primarily observe the algorithm working and then deciding on an upper bound value to stop or a maximum iteration number. The value to stop can be a gap percentage value between the lower bound and the upper bound that after some experiments seemed fit. As to the maximum iteration number it would depend on the time needed for each iteration and after some experiments with different networks finding the appropriate value.

3.2.2. Final Heuristic

After running this implementation a conclusion was reached that the first upper bound calculated was always better than the second and following iterations. From the second iteration it was observed that the LRH started to converge but did not reach the value obtained from the first iteration. This results for this observation are represented in Section 4.4.1.

The first iteration turned out to have a more accurate result than the implementation of the LP+R for networks with few traffic demands as it can be observed in Section 4.4.2. For that it was decided to use just the first iteration as the heuristic, it proved to be of really fast execution obtaining approximate values for networks low on traffic demands.

3.3 Node Architectures

As referred before the framework will design different networks each one with different node architectures. In this work networks were compared using a fixed node architecture, flexible node architecture and non-colorless nodes - digital node architecture.

Note that this step does not implies any change in the ILP formulation. Actually, the presence of different node architectures only impact the transponder costs and as a consequence the cost of the paths used as inputs for the formulation.

3.3.1 Flexible Node Architecture

The network with flexible node architecture is the default node type which is the one considered in the previous calculations of the ILP problem with only source grooming. The calculations can be observed in Section 3.1.1.

3.3.2 Fixed Node Architecture

The more complicated scenario is the network with fixed node architecture. For each node it is needed to previously calculate which of the directions should be the ones that do not need O/E/O conversion, in other words then ones that can be optical bypassed. The ideal choice for optical bypassed directions should take into account which paths are used but that will depend on the
chosen directions. Therefore the most accurate way would be to insert this problem into the ILP model formulation which would increase number of variables and constraints.

More efficient solutions, but less accurate in terms of minimizing the transponder related cost, would be either to define random default directions as optical bypassed or to calculate the directions based on the first calculated path of each node pair.

It was chosen upon calculating the optical bypassed directions by using an algorithm with a similar approach as the ReqSetup method in [3]. After calculating the paths for each node pair, each traffic demand is placed in the path connecting the respective node pair. After that the amount of traffic passing through the different directions is counted. The directions which have greater load for each node are the optical bypassed ones.

In order to compare the use or not of an initial algorithm it was also considered the previous calculations where random optical bypassed directions are defined. This way it can be proved that the previous calculation brings advantage. This comparison can be observed in Section 4.5.

3.3.3 Digital Node Architecture

In the network with a non-colorless node architecture there is a pair of transponders for each wavelength that passes through the node. There will always be electrical-optical-electrical (E/O/E) conversions and as so the cost of each path will be the cost of two transponders for each intermediate node of the path plus the cost of the source and destination transponders.

3.4 ILP Model Implementation For Intermediate Grooming

In order to groom the traffic not only at the source, it is needed either to purposely add and drop a traffic flow in the middle of it's lightpath or to take advantage of nodes where regeneration is needed to re-aggregate traffic.

The first case considers a situation where the traffic flowing in channel can be converted and reorganized with another traffic flow if it brings advantage or it is necessary in order to get a feasible solution. This might not bring a great cost gain, but can make it possible to design a network with big traffic demands and small link capacity. The second case is less prone to help with the network design, but gives us a better possibility of reducing costs.

In Section 3.4.1 there is a list of different solutions to consider in order to solve the intermediate grooming problem. Referring the one chosen and why.

3.4.1 Possible Solutions

In order to implement intermediate grooming there are several approaches that can be made. One example is to use the paths calculated in the ILP model without intermediate grooming and, node by node, reconstruct the network where intermediate grooming has cost gains. This solution
would not give us the optimal result, but will give us a faster result. As the alternatives are re-
formulating the problem which will need a considerable more amount of variables.

Different paths have a different nodes in which the regeneration is necessary. It is also pos-
sible to implement a greedy algorithm that can match these nodes with sources and destinations
of other traffic flows. And thus designing a network considering the cases where intermediate
grooming has no costs in a similar way to the work [15]. However using this approach it is possible
to get values even greater than the ones obtained in the previous formulation with no intermediate
grooming.

Another possibility is, as said above, re-formulating the problem and there are different ap-
proaches that can be made. We can, for instance, evaluate the problem node by node using a
node-link type formulation. In this case it is difficult to formulate the aggregation conditions and
costs as they are related with the path chosen. A traffic flow can go through different paths and
the nodes where a regeneration is needed will vary from path to path. However, a formulation that
observes the problem node by node would be most appropriate to consider this problem as what
we want to know is if there is a cost reduction by grooming traffic in one node.

Last but not least is another formulation using a link-path approach, but instead of considering
whole paths we consider sets of lightpaths. A similar approach was done in [29]. As referred
previously, different paths have different nodes in which the regeneration is necessary. There are
different permutations of which nodes should be chosen to regenerate the optical signal. And
between the nodes of each permutation there are the lightpaths to be considered. Each of these
lightpaths has only two transponders as no regeneration takes place.

Of all these possibilities, the one chosen was the last one. This solution provides a good
balance between computational complexity and the accuracy of the result. By using the latter
ILP result and applying one algorithm in order to find a better result, now using intermediate
grooming, it is predicted that the gain would not be significant. The network was already designed
and optimized to the source grooming scenario and the paths chosen might not be the indicated
ones for intermediate grooming. The node-link formulations needs the use of a great amount of
variables which greatly increases the time needed to solve this problem.

3.4.2 Formulation

The formulation constructed is not too different from the one used for only source grooming
(Formulation 1) and is described on Formulation 2. The indexes and parameters are almost
the same with some additions which can be observed in the formulation. Now the variables $x$
correspond to sub-paths, identified by a source/destination pair $(a, b)$ and the path $k$ that connects
those two nodes. These sub-paths are portions of a path that don’t need regeneration and will
be further referred in this document as transparent sub-paths. Figure 3.3 may help understanding
this concept.
3.4 ILP Model Implementation For Intermediate Grooming

For each sub-path one traffic demand can go through different channels identified by the channel bitrate $ch$ and an index $n$ which has the same use of Formulation 1, only now it refers to $ab$ instead of $sd$. The maximum value of $n$ for each $ab$, $k$, $ch$ needs to be previously calculated by summing the values of maximum $n$ related to the traffic demands that have the possibility of using the subpath $ab$.

As to the traffic variables $t$ they split into two kinds: $t_{sd,k,p}^u$ and $t_{sd,r_{ab}}^u$. Traffic demands $t_{ab}^u$ are identified by a source/destination pair $sd$ and the type of service $u$.

In constraint (3.18) it is specified that one traffic demand can go through different paths and within each path it can have different permutations ($p$) of regeneration nodes. Hence the variables $t_{sd,k,p}^u$. We need to separate a traffic demand into these variables so that we can know which portion of a traffic demand goes through a certain transparent sub-path $ab$ in constraint (3.19). Figure 3.3 gives an example of this permutation concept.

Each portion of a traffic demand that passes through a sub-path $ab$ can go in different channels as there are multiple channels that can be routed between it's endpoints. Each channel that can be chosen among it's different paths is represented in a constraint of type (3.16) which as associated a label $r_{ab}^i$. Hence this index in the variables of the type $t_{sd,r_{ab}}^u$ which represent the portion of a traffic demand that goes through the channel chosen in constraint $r_{ab}^i$.

It is in constraint (3.19) that these variables $t_{sd,r_{ab}}^u$ are related to the traffic demands through the use of the variables of type $t_{sd,k,p}^u$. For each possible sub-path $ab$ the both type of variables relative to one service $u$ that can pass through it are equaled.

The other constraints are the same as the ones used in Formulation 1: choosing how to groom the traffic and restrict the channels to the link’s capacity.
3. Architecture

**Formulation 2** ILP formulation considering intermediate grooming

**Extra Indexes**

- \( p \) identifies a different permutation of the placement of regenerators on the nodes of a path \( k \)
- \( ab \) Source and Destination node pair of a transparent subpath
- \( r_i^{ab} \) Identifier corresponding to the \( i^{th} \) restriction corresponding to an \( ab \) pair

**Extra Parameters**

- \( \delta_{ab,k}^{ij} \) List of values that are 1 if the \( k^{th} \) path between the node pair \( ab \) passes through link \( ij \) and 0 otherwise
- \( \delta_{ab}^{sd,u} \) List of values that are 1 if due the paths and permutations calculated the service of type \( u \) between the node pair \( sd \) has the possibility of passing through the sub-path \( ab \) and 0 otherwise
- \( \delta_{ab}^{sd,k,p} \) List of values that are 1 if the permutation \( p \) of the path \( k \) calculated between the node pair \( sd \) has the sub-path \( ab \) and 0 otherwise

**Variables**

- \( x_{sd,k}^{ch,n} \) Binary variable which represents the existence or not of the \( n^{th} \) optical channel from node \( s \) to node \( d \), that is routed through the \( k^{th} \) path of bitrate \( ch \)
- \( t_{sd,k,p}^{u} \) Integer value which represents how much of a service of type \( u \) from node \( s \) to node \( d \) goes through the permutation \( p \) of the path \( k \) (can have a value between 0 and the respective \( T_{sd}^{u} \))
- \( l_{u}^{sd,ri^{ab}} \) Integer value which represents how much of a service of type \( u \) from node \( s \) to node \( d \) goes through the subpath \( ab \) and is also used in restriction \( r_i^{ab} \) (can have a value between 0 and the respective \( T_{sd}^{u} \))

**Objective**

\[
\min z = \sum_{ab} \sum_{k} \sum_{n} \sum_{ch} \left( C^{TxRx}_{ch} x_{ab,k}^{ch,n} \right) \tag{3.15}
\]

**Constraints**

\[
r_i^{ab} \cdot \sum_{k} x_{ab,k}^{ch,n} = \sum_{u} \sum_{sd} \left( B_{ch}^{u} t_{sd,k}^{u} \delta_{sd}^{u} \right), \forall ab \forall ch \forall n \tag{3.16}
\]

\[
\sum_{k} x_{ab,k}^{ch,n} \leq 1, \forall ab \forall ch \forall n \tag{3.17}
\]

\[
\sum_{k} \sum_{p} t_{sd,k,p}^{u} = T_{sd}^{u}, \forall sd \forall u \in \{0,1,2,3,4\} \tag{3.18}
\]

\[
\sum_{sd} \sum_{k} \sum_{p} \left( t_{sd,k,p}^{u} \delta_{ab}^{u} \delta_{sd}^{u} \right) \sum_{ab} \sum_{n} \sum_{ch} x_{ab,k}^{ch,n} \delta_{ab}^{u} \leq W, \forall ij \tag{3.19}
\]

\[
\sum_{ab} \sum_{k} \sum_{n} \sum_{ch} x_{ab,k}^{ch,n} \delta_{ab}^{u} \leq W, \forall ij \tag{3.20}
\]
3.5 Summary

A framework was developed to design and optimize networks with static traffic demands. In order to solve the grooming problems ILP models were developed. For the source grooming solution two heuristics were developed: LRH and LP+R. LRH was not implemented as it was predicted to be.

The choice for a link-path similar formulation was taken because by using a link-path similar formulation we reduce the number of variables hence reducing the solver computation time. Also different scenarios with different node architectures are to be considered. When calculating the possible paths a cost can be attributed to each path depending on the architecture chosen. As so the same calculation is done when using the ILP model or any of the heuristics.

For calculating the paths’ costs of networks with a Fixed Node Architecture two solutions are proposed in which one of them has an algorithm to define which are the optical bypassed directions in each node. The other solution is to choose those directions randomly. The two solutions are going to be compared in Chapter 4.

The ILP model considering intermediate grooming is similar but will require a higher amount of variables and as so it is expected that the computation time spent will be higher than the ILP model that considers only source grooming.
3. Architecture
4 Results

Contents

4.1 Testing Environment .................................................. 38
4.2 Testing Framework .................................................... 38
4.3 Step 1 - Source Grooming Framework .......................... 39
4.4 Step 2 - Heuristics ...................................................... 43
4.5 Step 3 - Node Architecture Study ............................... 48
4.6 Step 4 - Framework With Intermediate Grooming ............ 49
4.7 Summary ................................................................. 50
4. Results

In this chapter the experiments in order to evaluate the framework steps are presented and detailed. In each step the tests differ as they have different goals. The first framework evaluations are done to analyze the first ILP model and compare it with the work of João Santos et al [24] using different parameters of $\alpha$ and $\beta$. The following compares the values obtained and computation time by solving the problem as an ILP or using the different heuristics: LP+R and LRH.

After that the results of networks using different architectures are going to be presented. Also it will be mentioned the calculation of Fixed Node Architecture networks with different algorithms for calculating colorless directions. Last but not least the results for solutions with intermediate grooming are compared with the results for solutions with only source grooming.

For all the tests that do not refer the parameters, the default values are:

- $\alpha = 2$
- $\beta = 1.4$
- $W = 100$
- $D_{\text{max}}^{100} = 2000\text{km}$

4.1 Testing Environment

The simulations were run on a laptop with an intel processor i7 and 4G of RAM running Windows 7 64bits.

For the ILP model solving CPLEX v12.2 was used.

While measuring computation times non-essential applications were closed and no interaction was made with the computer.

4.2 Testing Framework

In order to test the developed framework an to analyze it’s results there is a need to select networks and traffic demands to serve as input. In order to observe different behaviors and detect some patterns different networks were chosen and considered different traffic demands. The traffic demands need also to have some numeric distinction in order to get better conclusions. This section describes what was chosen and why.

4.2.1 Networks

In the experiments for this work four networks were considered. These networks were chosen among a set that was provided by João Santos and João Pedro of Nokia Siemens Networks (NSN).

The networks to be considered are European Optical Network (EON), US Backbone Network (UBN), CORONET and GÉANT. As some of these networks have a big dimension, for instance
4.3 Step 1 - Source Grooming Framework

the CORONET topology considered has 75 nodes, a software was developed to auto create the network topology graphs. This way the topologies can be presented with all the links’ length information.

The generated topologies are presented in Appendix A.

4.2.2 Traffic Demands

In order to reach good conclusions several traffic demands should be considered. The networks considered are large networks and filling the traffic demands manually can be a long time process. For that it was developed a script in Python to fill in the traffic demands.

Also to be easier to make comparisons there should be numeric ordinal values that characterize the traffic demands filled. This way we can compare not only what happens when the number of nodes and links are higher but also what happens when the number of traffic demands evolve which, for instance, makes possible to organize the information by using graphics.

The script developed receives as arguments the network topology and two numeric values. The first is the probability of having a traffic demand between two nodes: Traffic Demand Probability (TDP) hence referred as TDP and represented in percentage. For instance in a four node topology there are \( \binom{4}{2} = 6 \) node pairs, if this probability is 50% then around 3 node pairs would have traffic demands assigned for a certain service.

The second number is an index that decides the maximum number of different traffic flows belonging to the same service: Traffic Demand Quantity Index (TDQi) hence referred as TDQi. As different services have a significant difference in bitrate, the maximum possible amount of traffic flows per demand differ from service to service. In this case 1G25 services will have the same maximum value as the index passed as argument. For 2G5 services the maximum value is half of the index value, for 10G a third, for 40G a forth and 100G a fifth (using integer division).

The script has also been programmed to generate traffic using just one parameter: the Total Traffic Volume (TTV) in Gbps. After giving this parameter the script will distribute this total of traffic through the different node pairs and services.

4.3 Step 1 - Source Grooming Framework

In this section the framework with only source grooming will be analyzed and compared with the work of Joao Santos. So far it is only considered the use of ILP to calculate the network cost, the heuristic results are going to be covered in Section 4.4.

4.3.1 Source Grooming Framework Study

Here some results of the framework with only source grooming are going to be presented.
4. Results

One of the experiments to test this framework was to see its behavior when the traffic volume increased. For that the script was used to distribute the traffic using the parameter TTV between the values 500Gb and 30500Gb with an interval of 500Gb. The maximum of 30500Gb is due to the fact that traffic demands with higher traffic volume caused errors in the framework due to memory limit.

4.3.1.A Cost

Figure 4.1 represents the total network cost calculated depending on the traffic volume variation. The growth of the network cost value with the traffic volume was expected and can be observed.

![Network Cost vs Traffic Volume](chart.png)

Figure 4.1: Comparison between network costs varying traffic volume in EON network.

By observing the chart it is also noticed that, in some cases, higher traffic volumes yield lower cost values. This can happen when the cost depends greatly on how the traffic demands are distributed. Some traffic demand distributions can be groomed in a more efficient way and so lower network costs can be obtained by considering a higher total traffic volume.

4.3.1.B Computation Time

Figure 4.2 represents the computation time needed to solve the ILP problem depending on the traffic volume variation. Computation time also suffers from the same effect. Although it has the tendency to increase if the framework receives traffic demands with higher total bitrate, in some cases the opposite happens. For instance the computation times registered using TTV between 25000Gbps and 30000Gbps: the computation time has a tendency to increase when given a higher traffic volume, but there are variations in which we get a lower computation time with a higher traffic volume. This happens for a set of reasons.
From one perspective we have other processes that are running in the same machine as the framework. These processes need to use resources that the framework could use and they don’t always use it using the same CPU utilization, for instance.

From other perspective different traffic demands with the same amount of traffic yield different formulations. Some can have fewer variables and hence the increase in speed when solving the ILP problem. For instance if a traffic demand has only 100G traffic it will require less variables than a traffic demand with the same traffic amount with all traffic types. Not only one variable includes more traffic, but also 40G channels are not to be considered in this case.

![Computation Time](image)

Figure 4.2: Comparison between the computation times for solving the ILP problem varying traffic volume in EON network.

### 4.3.2 Comparison With Previous Work

In this section we compare this thesis work with the work of João Santos et al. In this evaluation it is ignored the VCAT option and the possibility of having traffic demands with bitrate 1G25, 2G5 and 10G in order to have compatibility.

It is expected that this work yields higher network costs as the paths calculated in this work don’t consider links larger than the transponders optical reach. The compared work does a more simple calculation of the paths costs and might consider those cases as the calculation is based in the total length of the path and the optical reach, not considering the links individually.

For this comparison two networks were considered EON and UBN. These networks were chosen has they have links with big length (biggest is 2600). Figure 4.3 and Figure 4.4 refer to the results of the network costs using the two works varying the values of $\alpha$, $\beta$ and $W$ different scenarios:

- $a - \alpha = 1, \beta = 1$ and $W = 100$;
4. Results

- b - $\alpha = 1, 5$, $\beta = 1$ and $W = 100$;
- c - $\alpha = 1, \beta = 1, 4$ and $W = 100$;
- d - $\alpha = 1, 5, \beta = 1, 4$ and $W = 100$;

![UBN Network](image)

Figure 4.3: Comparison between network costs resulted from the two works using EON network.

![EON Network](image)

Figure 4.4: Comparison between network costs resulted from the two works using UBN network.

In order to obtained these results it was necessary to use the value $D_{100}^{\max} = 2000km$. Simulations were also run using the value $D_{1000}^{\max} = 1000km$, but for all the scenarios in this current work no feasible solution was found as there are multiple links bigger than 1000km or even 1400km ($1000.\beta$). Some node pairs didn’t have a possible path between them.
4.4 Step 2 - Heuristics

In this section the heuristics included in this work (LP+R and LRH) are going to be analyzed by being compared with the ILP. Computation times and calculated network costs are going to be measured using different networks and traffic demands as input.

4.4.1 LRH Convergence and Iteration

Here the LRH convergence is going to be analyzed and so it will be explained why the algorithm suffered alterations from the version proposed initially.

Figure 4.5 represents the lower and upper bound value calculated in each iteration. This experiment was done using the EON network and the traffic demand generated by using the values $TDP = 50\%$ and $TDQi = 5$.

![LRH Iteration](image)

Figure 4.5: LRH upper and lower bound values in each LRH iteration using network EON

As it can be observed the lower bound converges as it is expected by iterating the multipliers with the subgradient method. Figure 4.6 represents only the positive values and adds the optimal value, calculated with the ILP model, in order to better understand the values.

The first value obtained by the upper bound was 629 being the optimal value 618. The value obtained in the second iteration was 881 and from there on the upper bound values calculated started decreasing. After some iterations the value stabilizes in 759 which is a value greater than the value obtained in the first iteration.

After conducting this experiment in other networks with different traffic demands, it was observed that the value calculated in the first iteration was always the smallest value.

This shows that the upper bound algorithm developed is better used once than several times iterating the Lagrangian Multipliers.
4. Results

Figure 4.6: LRH upper and lower bound values in each LRH iteration using network EON with only positive values

4.4.2 ILP vs LP+R vs LRH

Here we compare the values obtained by the implementation of the ILP Model, the LP+R heuristic and the LRH heuristic. In order find patterns in the time needed to solve the problem and value obtained in function of the networks used and the quantity of traffic demands several simulations were made.

Figures 4.8 and 4.7 represent charts comparing computation times and the network costs final values, respectively, of the different implementations.

As we can observe in Figure 4.8 in terms of computation time the LRH implementation is the fastest in general. The ILP implementation with lighter traffic demands (traffic demands with lower traffic volume) starts by solving the problem fast, even faster than LP+R but when we give a higher traffic volume in the input the computation time raises at a high rate. This can be observed, for example, with the values using the UBN network.

In conclusion LRH has a great computation time with a small growth when the traffic volume as input increases. LP+R has a bigger computation time growth but it is, as well as LRH, similar to a linear growth. ILP computation time growth is not linear and is more similar to an exponential growth. As so for bigger networks or a bigger traffic volume easily the time needed to solve a problem starts to be too long.

In terms of network cost accuracy LRH is more accurate with lighter traffic demands. By increasing the traffic volume the LP+R implementation starts to have more accurate results than LRH.

Table 4.1 shows the traffic volume and network topology effect on the final computed network cost and computation time needed to solve the problem. In some cases the framework returned
errors due memory limit. For those situations the last value obtained was recorded. The values presented in the table with an * are values that were obtained from that recorded value and so might not be the optimal ones.

As it can be observed in Coronet, which has 75 nodes, in some cases LRH takes longer to solve the problem than ILP. However the computation time growth with the traffic volume is lower than the LP+R for the few data collected.

LRH and LP+R are two different approaches. They are better used when the traffic demands are of higher volume or when the networks have a big dimension. In these conditions LRH is a good approach if it is wanted fast results, but LP+R is a better approach if it is wanted more accurate results.
4. Results

**Network Cost Comparison**

Figure 4.7: Comparison between implementations’ network costs varying traffic volume in EON network.

**Computation Time Comparison**

Figure 4.8: Comparison between implementations’ computation times varying traffic volume in EON network.

4.4.3 Time Restrained ILP vs LRH

In this section simulations are run using LRH. The time needed to solve a problem using LRH will be a time restraint to the ILP so it is possible to know the minimum cost the ILP can find taking the same time as the LRH.

Table 4.2 shows the values obtained with this experiment. Different times for LRH and restrained ILP happen because what is restricted is the ILP problem solving and not also the time needed to formulate the problem.
4.4 Step 2 - Heuristics

<table>
<thead>
<tr>
<th>Network</th>
<th>TDP (%)</th>
<th>TDQi</th>
<th>Restricted ILP</th>
<th>LRH</th>
<th>Unrestrained ILP</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Time (s)</td>
<td>Result</td>
<td>Time (s)</td>
</tr>
<tr>
<td>UBN</td>
<td>5</td>
<td>10</td>
<td>0,221</td>
<td>392</td>
<td>0,09</td>
</tr>
<tr>
<td></td>
<td></td>
<td>20</td>
<td>-</td>
<td>-</td>
<td>0,109</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>10</td>
<td>0,199</td>
<td>1864</td>
<td>0,105</td>
</tr>
<tr>
<td></td>
<td></td>
<td>20</td>
<td>-</td>
<td>-</td>
<td>0,273</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>10</td>
<td>-</td>
<td>-</td>
<td>0,265</td>
</tr>
<tr>
<td></td>
<td></td>
<td>20</td>
<td>-</td>
<td>-</td>
<td>0,384</td>
</tr>
<tr>
<td>EON</td>
<td>5</td>
<td>10</td>
<td>0,156</td>
<td>314</td>
<td>0,045</td>
</tr>
<tr>
<td></td>
<td></td>
<td>20</td>
<td>-</td>
<td>-</td>
<td>0,047</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>10</td>
<td>0,289</td>
<td>924</td>
<td>0,07</td>
</tr>
<tr>
<td></td>
<td></td>
<td>20</td>
<td>-</td>
<td>-</td>
<td>0,078</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>10</td>
<td>-</td>
<td>-</td>
<td>0,156</td>
</tr>
<tr>
<td></td>
<td></td>
<td>20</td>
<td>-</td>
<td>-</td>
<td>0,15</td>
</tr>
<tr>
<td>GEANT</td>
<td>5</td>
<td>10</td>
<td>0,327</td>
<td>410</td>
<td>0,266</td>
</tr>
<tr>
<td></td>
<td></td>
<td>20</td>
<td>0,531</td>
<td>664</td>
<td>0,327</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>10</td>
<td>1,326</td>
<td>1778</td>
<td>0,577</td>
</tr>
<tr>
<td></td>
<td></td>
<td>20</td>
<td>-</td>
<td>-</td>
<td>0,68</td>
</tr>
</tbody>
</table>

Table 4.2: ILP solving restrained by computation time.

In most cases of the ones experimented the ILP could not get an upper bound within time. As so no result was reached. Also in other networks using a TDP higher than 25% no result was reached within the limited time. Coronet network was not considered in this case as for the values that it was going to be tested all ILP solving computation times were smaller than LRH.

It can be observed that for GEANT network more results could be obtained. This is due to the fact that it is a bigger network (32 nodes) and so for lighter traffic demands the difference in computation times from the ILP implementation to the LRH implementation is smaller.

With this experiment it was pretended to see if an ILP approach could return a lower network cost than LRH taking the same time to solve. ILP problems often take a long time to solve because although they have found the optimal solution they need to be certain that the solution is optimal using the branch and bound approach. As so it was possible that the ILP approach could get better network cost results using the same computation time as LRH.

As it can be observed only using light traffic demands as input to the framework this theory has some effect. Also when using the traffic demands where restricted ILP got better values, the ILP is in fact the best option (as the computation time needed is of mere seconds) and so there is no need for the use of LRH.
4. Results

4.5 Step 3 - Node Architecture Study

In this section it is going to be analyzed the difference in transponders associated cost in networks with different node architectures. Using different node architectures imply nodes with different costs as it is referred in Section 2.2 but that is not considered in this work.

<table>
<thead>
<tr>
<th>TDP (%)</th>
<th>TDQi</th>
<th>Node Architecture</th>
<th>$D_{\text{max}}^{100} = 1500\text{km}$</th>
<th>$D_{\text{max}}^{100} = 2000\text{km}$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>$\alpha = 1$</td>
<td>$\alpha = 1.5$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Flexible</td>
<td>4068</td>
<td>3616</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fixed</td>
<td>4260</td>
<td>3861.33</td>
</tr>
<tr>
<td></td>
<td>20%</td>
<td>Randomly Fixed</td>
<td>4452</td>
<td>4036</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Digital</td>
<td>4804</td>
<td>4320</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2528</td>
<td>2158.67</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fixed</td>
<td>2724</td>
<td>2340</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Randomly Fixed</td>
<td>2832</td>
<td>2414.67</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Digital</td>
<td>3112</td>
<td>2621.33</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>Flexible</td>
<td>936</td>
<td>765,333</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fixed</td>
<td>1008</td>
<td>841,333</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Randomly Fixed</td>
<td>1020</td>
<td>854,667</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Digital</td>
<td>1168</td>
<td>965,333</td>
</tr>
<tr>
<td></td>
<td>5%</td>
<td>Flexible</td>
<td>712</td>
<td>581,333</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fixed</td>
<td>788</td>
<td>653,333</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Randomly Fixed</td>
<td>800</td>
<td>662,667</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Digital</td>
<td>896</td>
<td>736</td>
</tr>
</tbody>
</table>

Table 4.3: Node Architecture Study for UBN.

Table 4.3 represents the values obtained using the different architectures and different $\alpha$ values in the UBN network. The Fixed architecture line has the final network cost values using a fixed node architecture and uses the algorithm referred in Section 3.3.2 to calculate the directions of a node which are optical bypassed. The Randomly Fixed architecture line has the final network cost values using a fixed node architecture and selects the optical bypassed directions randomly.

As expected and supporting what is theoretically correct a digital node architecture is the one that returns a bigger network cost followed by a fixed node architecture and finally the flexible node architecture.

By the results it can be observed that, when calculating the cost of a fixed node architecture, the algorithm referred brings advantage when compared to a random selection. In all the calculated values the Fixed architecture returned lower values than the Randomly Fixed.

Looking at the values it can also be observed that the optical reach has a smaller impact in networks with digital node architecture when comparing to the other node architectures considered. It also has a smaller impact in networks with a fixed node architecture when comparing to a
4.6 Step 4 - Framework With Intermediate Grooming

In this section it is going to be analyzed the cost gains of using intermediate grooming and the disadvantages of using this implementation. It is expected that this implementation yields lower costs as it considers more possibilities to groom traffic. It is also expected that it needs a longer computation time to solve the problem as the ILP formulation is more complex and requires more variables than the Step 1 implementation for the same input.

In order to obtain more results the number of paths considered when considering intermediate grooming changed to 3. This way less variables are needed and more results can be obtained.

4.6.1 Network Costs

Figure 4.9 is a chart comparing the costs obtained by the framework with and without intermediate grooming. From the chart it can be observed that when using intermediate grooming the framework yields lower costs.

![Network Cost Comparison](image)

Figure 4.9: Comparison between step 1 and step 4 implementations’ network costs varying traffic volume in EON network.

Also the difference between the costs calculated is greater when the framework receives traffic demands that have higher traffic volume. This can be easily explained as with greater traffic volumes there are more possibilities of grooming.

It can be also observed that the cost calculated using the implementation with traffic grooming as a more linear growth with traffic volume. As referred before the cost of the network using only grooming at the source depends not only of the traffic volume, but also how that traffic is arranged.
4. Results

By introducing intermediate grooming a wider set of grooming possibilities are considered where the traffic going through a lightpath can be groomed at any node. Due to this fact how the total traffic is organized in the traffic demands has smaller impact in the calculated network cost.

4.6.2 Computation Times

Figure 4.10 is a chart comparing the computation times obtained by the framework with and without intermediate grooming. As the values observed had a significant difference between them it was opted to present the computation time values in a logarithmic scale (base 10).

![Computation Time Comparison](chart.png)

Figure 4.10: Comparison between step 1 and step 4 implementations’ computation times varying traffic volume in EON network with a logarithmic scale on time.

As it was expected using the ILP formulation that supports intermediate grooming has a great impact on the time needed to solve the problem. By increasing the same amount of traffic the solution with intermediate grooming stepped from taking around 0,1 seconds to above 1000 seconds (2076,44s to be exact) while the solution with no intermediate grooming stepped from taking around 0,01 to around 0,1.

Not only the new formulation takes, in general, longer to compute but also the difference will be higher and higher when raising the total traffic volume to be considered.

4.7 Summary

The computation time needed to solve the ILP problems considered is an issue. Although it has a considerable low value when it comes to small networks or a lighter traffic demand it starts to grow rapidly as bigger networks or heavier traffic demands are considered. This growth is not
linear and is more similar to an exponential growth.

In terms of heuristics when varying traffic demands from lighter to heavier ones both have a growth that is similar to a linear growth even though LRH takes less time to compute and has a slower growth than LP+R. However for heavier traffic demands LP+R produces better results.

When using second generation ROADM a good algorithm to select the optical bypassed directions has impact in the network cost. Also non-optical nodes or second generation ROADMs bring more disadvantages to the network cost when the optical reach between two nodes is higher.

By considering intermediate grooming better network cost results are obtained. However by comparing the two ILP solutions it can be observed that the solution that considers intermediate grooming needs, in general, a higher computation time to solve the problem. Also the solution that considers intermediate grooming has a greater computation time growth as heavier traffic demands are considered.
4. Results
5

Conclusions
5. Conclusions

In the thesis work a framework was developed to design and optimize the cost of an OTN/DWDM network. This work extends works by João Santos et al. by introducing the possibility of more services as input and the consideration for intermediate grooming of traffic. Also a new heuristic was considered and compared to LP+R. In addition this thesis work considers different types of node architectures.

This work was separated into different phases. The first phase was to upgrade the ILP model in order to consider the 5 different services.

The second was to build a heuristic to reduce the computation time of the problem solving. A heuristic was built based on the work [11] from Lee et al. called LRH. This heuristic consists in an algorithm to calculate an upper bound to the problem and a Lagrangian relaxation to calculate a lower bound. The upper bound algorithm depends on the Lagrange multipliers which are iterated yielding different values for the lower and upper bound. In this work another upper bound algorithm was designed in order to adapt to the problem. Although it was also based in the Lagrangian relaxation theory and dependent on the Lagrange multipliers, this algorithm proved to yield better results when not iterated.

This solution was then compared to the LP+R solution introduced by João Santos. For heavier traffic demands, which is when the heuristics bring a great advantage in terms of computation time, LP+R tends to yield more accurate results and LRH tends to yield results faster.

The impact of a node architecture in the transponder related network cost was also studied and was the third phase of this work. One of the conclusions was that when designing a network with second generation ROADMs a good choice of the optical bypassed direction is important for a lower transponder related network cost. In this work a solution is proposed to set these optical bypassed directions when the traffic demands are known.

Also the impact of a different node architecture on transponder related network cost is higher when the optical reach of the equipment considered is bigger. For instance when choosing between second and third generation ROADMs if they have equally an optical reach of 1000km the difference in the transponder cost will be smaller than if they had both an optical reach of 2000km.

Last but not least the final phase was to make the framework consider intermediate grooming. By considering intermediate grooming lower costs were obtained. Also a more steady growth of the network cost with the total traffic volume as input can be observed, being this cost a value easier to predict.

5.1 Future work

The study of different node architectures took only into account the transponder related cost. Second generation ROADMs are cheaper equipment but require more transponders. Also third generation ROADM costs depend on the number of different directions it needs to consider. A
study could complete the third phase of this work where all this details and equipment costs were taken into account.

The ILP model considering intermediate grooming has a great computation time growth when considering heavier traffic demands. The module may require a big memory/storage to return feasible results if the total traffic volume considered is too high. Also it may require more time than the one a person or company is willing to wait.

The heuristics proposed are both adaptable to this problem seeing the formulations are similar. However they were not adapted in this work. A future work can adapt the heuristics used in this work and even compare them with a new one, for instance.
5. Conclusions
Bibliography


Appendix A
In this appendix the topology of the used networks are presented. The link labels represent the connection length in km. Note that the graphs’ edges length don’t correspond to the fiber connection length between two nodes and that the node placement is not the same as it occurs in the real network implementation.

Figure A.1: EON Topology.
Figure A.2: UBN Topology.
Figure A.3: GEANT Topology.
Figure A.4: CORONET Topology.